SFX

Advanced User's Guide

Version 4

The bridge to knowledge

CONFIDENTIAL INFORMATION

The information herein is the property of Ex Libris Ltd. or its affiliates and any misuse or abuse will result in economic loss. DO NOT COPY UNLESS YOU HAVE BEEN GIVEN SPECIFIC WRITTEN AUTHORIZATION FROM EX LIBRIS LTD.

This document is provided for limited and restricted purposes in accordance with a binding contract with Ex Libris Ltd. or an affiliate. The information herein includes trade secrets and is confidential.

DISCLAIMER

The information in this document will be subject to periodic change and updating. Please confirm that you have the most current documentation. There are no warranties of any kind, express or implied, provided in this documentation, other than those expressly agreed upon in the applicable Ex Libris contract. This information is provided AS IS. Unless otherwise agreed, Ex Libris shall not be liable for any damages for use of this document, including, without limitation, consequential, punitive, indirect or direct damages.

Any references in this document to third-party material (including third-party Web sites) are provided for convenience only and do not in any manner serve as an endorsement of that third-party material or those Web sites. The third-party materials are not part of the materials for this Ex Libris product and Ex Libris has no liability for such materials.

TRADEMARKS

"Ex Libris," the Ex Libris Bridge to Knowledge , Primo, Aleph, Voyager, SFX, MetaLib, Verde, DigiTool, Rosetta, bX, URM, Alma , and other marks are trademarks or registered trademarks of Ex Libris Ltd. or its affiliates.

The absence of a name or logo in this list does not constitute a waiver of any and all intellectual property rights that Ex Libris Ltd. or its affiliates have established in any of its products, features, or service names or logos.

Trademarks of various third-party products, which may include the following, are referenced in this documentation. Ex Libris does not claim any rights in these trademarks. Use of these marks does not imply endorsement by Ex Libris of these third-party products, or endorsement by these third parties of Ex Libris products.

Oracle is a registered trademark of Oracle Corporation.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd.

Microsoft, the Microsoft logo, MS, MS-DOS, Microsoft PowerPoint, Visual Basic, Visual C++, Win32,

Microsoft Windows, the Windows logo, Microsoft Notepad, Microsoft Windows Explorer, Microsoft Internet Explorer, and Windows NT are registered trademarks and ActiveX is a trademark of the Microsoft Corporation in the United States and/or other countries.

Unicode and the Unicode logo are registered trademarks of Unicode, Inc.

Google is a registered trademark of Google, Inc.

Document released: March 2024

Web address: http://www.exlibrisgroup.com

Table of Contents

Appendixes

Updates to This Guide

This guide is being reissued due to the following changes:

- **Command-Line DataLoader** on page **107** was updated.

1

SFX Environment

This section describes the SFX environment and the layout of the SFX installation, databases, and the directory tree. This section also explains how to navigate the SFX server and how the database is set up.

This section includes:

SFX Installations

In SFX, the following levels of granularity exist:

- **Installation** – One or more SFX installations can exist on one server.
- **Customer** – The institution or consortium that has SFX. A customer can have access to one or multiple instances.
- **Instance** – An SFX environment to be localized by an institution.
- **Institute** – A sub-division within an instance, with its own configuration settings different from the instance default (regarding activation, coverage, proxy or linking parameter information). An institute can be assigned to one or more groups.

These levels are used in the `OWNER` and `AVAILABLE_FOR` fields in the SFX database to indicate which customer, instance, or institute owns and is entitled to see each database item. For more information about these database fields, see **Appendix A: Database Tables** on page **255**.

The customer name is a unique customer code assigned by Ex Libris and is included in the SFX license of each instance. This code can be either an individual customer code or a consortium customer code. When an instance is created, the customer name is retrieved from the license and stored in the following configuration file of the global instance:

```
config/instances.config_
```

IMPORTANT:
This file should not be edited.

From SFX 4.1.1 and later, the customer name for newly created instances is stored in the local instance `CONTROL` table. When retrieving the customer name, SFX first checks the `CONTROL` table and then the `sfxglb41` configuration file `instances.config_`.

Multiple SFX installations can exist on the same server, but instance names and Apache ports should be unique within the server. Each installation is located at `/exlibris/sfx_ver/sfx4_x` (where 'x' is the installation number).

By default, the first installation created is called `sfx4_1` and is located at `/exlibris/sfx_ver/sfx4_1`. The global instance of this installation is called `sfxglb41`.

Additional installations are called `sfx4_2`, `sfx4_3`, and so forth.

In the example below, two installations were created:

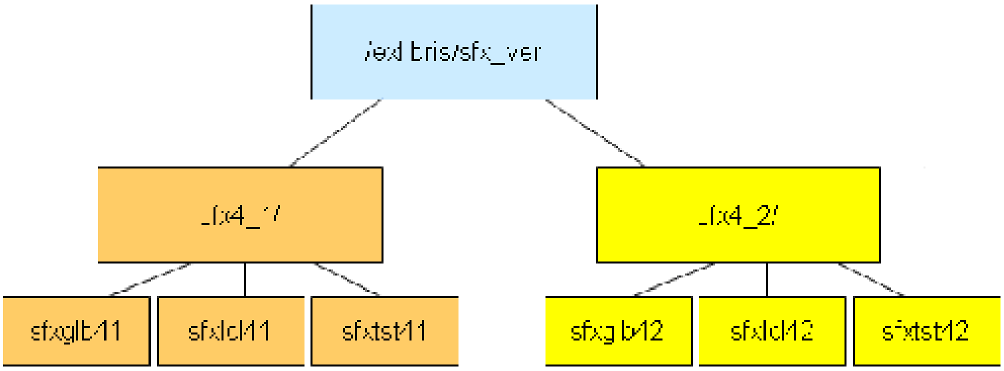


Figure 1: SFX Installations

In the description of the SFX directory tree structure (see **SFX Directory Tree Structure** on page 17), an installation in the `sfx4_1` directory is used as an example.

SFX Database Structure

The SFX database contains two types of information: global and local.

- Global information – Data about resources that exist in the information world and are "globally" relevant

  Examples are: target, target service, and portfolio information, including names, descriptions, global coverage, and linking information.

- Local information – customer, "context"-specific data, such as activation information, local coverage, and customer-specific linking information. Local information includes overrides of global data.

Global information is stored in the global schema (`sfxglb4x`). Only one copy of the global information exists per SFX installation. This global information is largely maintained by Ex Libris as part of the global KnowledgeBase. Additionally, any locally created database items that describe global information (for example, targets, target services, portfolios, or objects) are stored in the global schema.

Local information is stored only in the local schema(s) of each instance. The activation information for any global database item is stored in the INVENTORY tables, whereas other localization information and overrides of global data are stored in the LINKING_INFO tables.

A list of database tables and fields, as well as a section of the ERD diagram that shows the connections between the most important database tables, can be found in **Appendix A: Database Tables** on page 255.

For more information on the SFX databases and MySQL, refer to the **SFX Environment** section of the *SFX System Administration Guide*.

SFX Directory Tree Structure

A default SFX installation package includes some application directories (`app`, `data`, and `proxy`) and the following three instances:

- `sfxglb41` – This is the global instance.
- Two local instances:
  - `sfxtst41` – This instance is used as a testing environment. This is a clean copy of the SFX instance.
  - `sfxlcl41` – This instance is used for production.

Additional local instances may be created within the installation.

Do not make an localizations in the `sfxglb41` instance. Instead, perform file localizations in the local instances, after the symbolic link to the global instance has been broken.

The resulting tree structure is as follows:

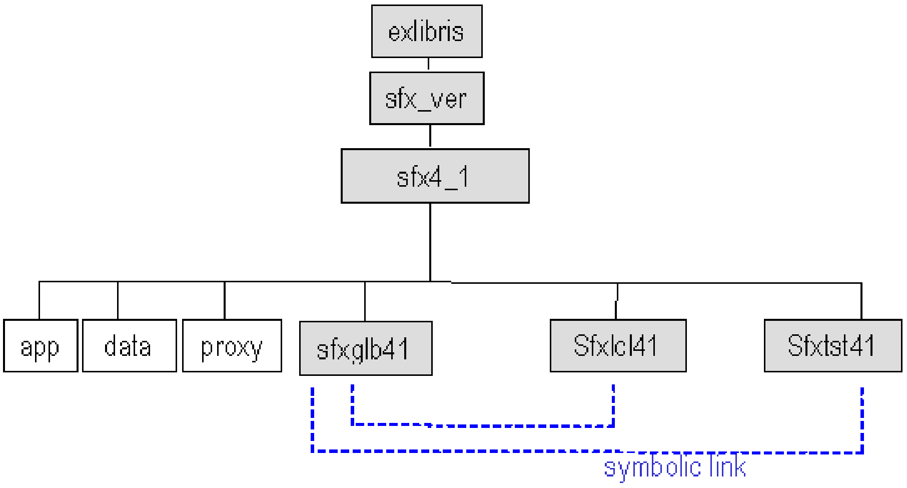


Figure 2: SFX Tree Structure

For more information on the SFX databases and MySQL, refer to the **SFX Environment** section of the *SFX System Administration Guide*.

Application Directories

The following are the application directories in SFX.

- /exlibris/sfx_ver/sfx4_x/app – Supporting Software

  This directory contains all supporting software required for running SFX (MySQL, Perl programming environment, Java, and an Apache Web server).

  The app directory contains the following subdirectories:

Table 1. Application Directory

Subdirectory	Description
/exlibris/sfx_ver/sfx4_x/app/apache	Contains all software required to run and administer an Apache HTTP dæmon
/exlibris/sfx_ver/sfx4_x/app/mysql	Contains all software required to run and administer a MySQL database
/exlibris/sfx_ver/sfx4_x/app/openssl	Contains software for HTTPS
/exlibris/sfx_ver/sfx4_x/app/oracle	Contains an Oracle table client used for Verde-SFX synchronization

Table 1. Application Directory

Subdirectory	Description
`/exlibris/sfx_ver/sfx4_x/app/perl`	Contains all Perl software and modules required for SFX
`/exlibris/sfx_ver/sfx4_x/app/python`	Contains Python, a programming language used by Sphinx
`/exlibris/sfx_ver/sfx4_x/app/sphinx`	Contains Sphinx, an SQL full-text search engine used for Author searching
`/exlibris/sfx_ver/sfx4_x/app/utils`	Contains utilities such as UNZIP
`/exlibris/sfx_ver/sfx4_x/app/yaz`	Contains a YAZ z39.50 client

- `/exlibris/sfx_ver/sfx4_x/data` – This directory contains MySQL databases.
- `/exlibris/sfx_ver/sfx4_x/proxy` – This directory contains reverse proxy Apache files.

Default Instances

The following are the default instances in SFX.

- `sfxglb4x` (Global)

  The purpose of the global instance is to provide the customer with the most up-to-date KnowledgeBase, programs, and configuration files.

- `sfxlcl4x` (Local) and `sfxtst41` (Test)

  The files in the local and test instance use UNIX symbolic links to point to files in the global instance. Ex Libris regularly updates files on the `sfxglb41` instance and because of the relationship between the `sfxlcl41` instance and the `sfxglb41` instance, the `sfxlcl41` instance is immediately updated as well (same for `sfxtst41`).

Tree Structure of Each SFX Instance

Each SFX instance has the following directories and subdirectories:

Table 2. Tree Structure of Each SFX Instance

Directory	Description
`/exlibris/sfx_ver/sfx4_x/<instance>/admin`	Contains command line tools

Table 2. Tree Structure of Each SFX Instance

Directory	Description
/exlibris/sfx_ver/sfx4_x/<instance>/archive	Contains statistics archive files
/exlibris/sfx_ver/sfx4_x/<instance>/cgi/core	Contains SFX core CGI programs (encrypted), such as sfxmenu.cgi, sfxresolver.cgi, and sfxctrl.cgi
/exlibris/sfx_ver/sfx4_x/<instance>/cgi/public	Contains CGI programs required for certain targets (not encrypted, intended to be editable)
/exlibris/sfx_ver/sfx4_x/<instance>/config	Contains all configuration files
/exlibris/sfx_ver/sfx4_x/<instance>/dbs	Contains SQL table definitions and the scratch directory
/exlibris/sfx_ver/sfx4_x/<instance>/export	Contains export files from KBTools
/exlibris/sfx_ver/sfx4_x/<instance>/home	Home directory of the instance
/exlibris/sfx_ver/sfx4_x/<instance>/lib	Contains Perl modules, target parsers, source parsers, and target displayer programs
/exlibris/sfx_ver/sfx4_x/<instance>/logs	Contains SFX request logs when redirection is used
/exlibris/sfx_ver/sfx4_x/<instance>/sfxadmin	Contains programs for the SFX Admin Center
/exlibris/sfx_ver/sfx4_x/<instance>/templates	Contains HTML pages for the SFX menu screen, the HTML update reports, the HTML electronic journal lists, and SFX Admin
/exlibris/sfx_ver/sfx4_x/<instance>/convert_DB	Contains data related to the SFX V3 to SFX V4 database upgrade
/exlibris/sfx_ver/sfx4_x/<instance>/upgrade	Contains data related to the SFX V3 to SFX V4 upgrade process

Sfxmenu Aliases

When logging on to the server, several aliases listed in the following table can be used to navigate through the `sfxmenu` directory structure.

Table 3. Sfxmenu Aliases

Alias	Description
`Slib <instance>`	Sets the current `sfxmenu` instance to `<instance>`
`cv`	Changes the directory to the `sfx4_x` directory: `/exlibris/sfx_ver/sfx4_x/`
`sx`	Changes the directory to the sfxmenu root directory for the current instance: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/`
`lib`	Changes the directory to the `lib` directory for the current instance: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/lib`
`cn`	Changes the directory to the configuration directory containing all configuration files: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/config`
`cgi`	Changes the directory to: `/exlibris/sfx_ver/sfx4_x/<instance>/cgi`
`da`	Changes the directory to the `admin` directory containing administrative tools and programs: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/admin`
`ds`	Changes directory to the `scratch` directory containing the input and output of the programs in the `prog` directory: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/dbs/scratch`
`dt`	Changes the directory to the `tab` directory containing all SQL scripts to create database tables and indexes: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/dbs/sqltab`
`sp`	Changes the directory to the `SourceParser` directory containing all SourceParser modules: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/lib/Parsers/SourceParser`

Table 3. Sfxmenu Aliases

Alias	Description
`tp`	Changes the directory to the `target/TargetParser` directory containing all TargetParser modules: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/lib/Parsers/TargetParser`
`td`	Changes the directory to the `TargetDisplayer` directory containing all TargetDisplayer modules: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/lib/Parsers/TargetDisplayer`
`tm`	Changes the directory to the `templates` directory: `/exlibris/sfx_ver/sfx4_x/<sfx_instance>/templates`
`lg`	`/exlibris/sfx_ver/sfx4_x/<sfx_instance>/logs`
`ph`	`/exlibris/sfx_ver/sfx4_x/app/perl/bin`
`mh`	`/exlibris/sfx_ver/sfx4_x/app/mysql`
`ah`	`/exlibris/sfx_ver/sfx4_x/app/apache/bin`

Mod_Perl and Multiple Apache Daemons

SFX uses Mod_perl to compile code in order to optimize performance. To be able to use Mod_perl, SFX employs a reverse proxy setup. In this setup, different SFX instances on the same server share the same binaries of Apache, but each instance has its own Apache daemon, as follows:

- In the `sfxglb41` instance, the following configuration file lists the Apache configuration of the different instances:

  `/exlibris/sfx_ver/sfx4_1/proxy/config/reverse_proxy_httpd.config`

- In the local SFX instances, the following configuration files contain Apache daemon information:
  - `/exlibris/sfx_ver/sfx4_x/<instance>/config/httpd.conf`
  - `/exlibris/sfx_ver/sfx4_x/<instance>/config/local_httpd.config`

To the outside world, all SFX instances have the same port.

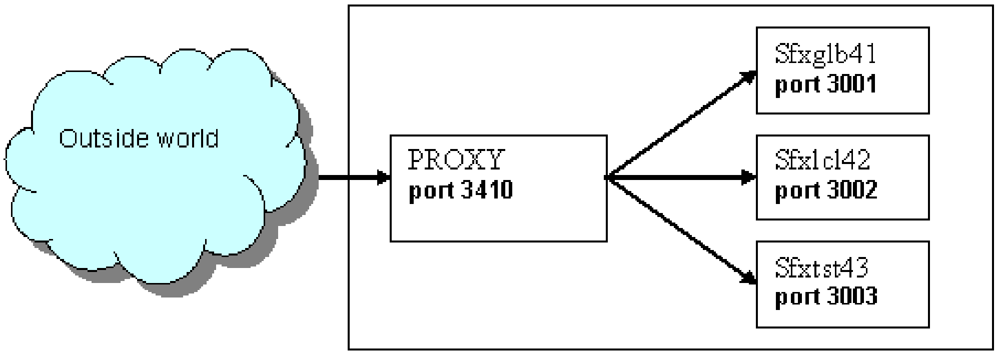


Figure 3: SFX Instances

For the `/exlibris/sfx_ver/sfx4_1` installation, the default reverse proxy port is 3410. This port can be changed during or after SFX installation.

For the `/exlibris/sfx_ver/sfx4_2` installation, the default reverse proxy port is 3420.

UNIX File Map and User Shortcuts

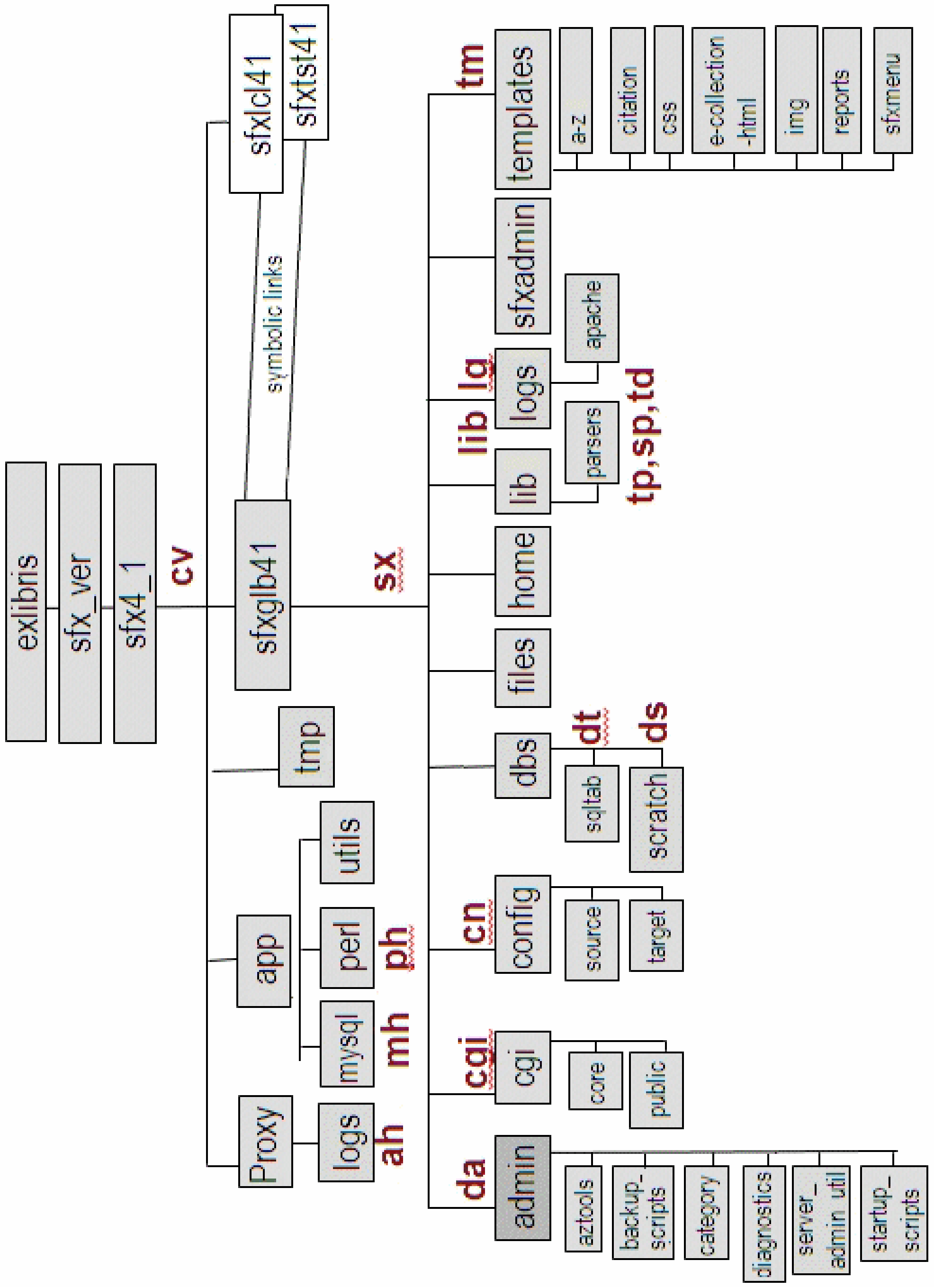


Figure 4: UNIX File Map and User Shortcuts

2

Configuring the SFX Menu

This section includes:

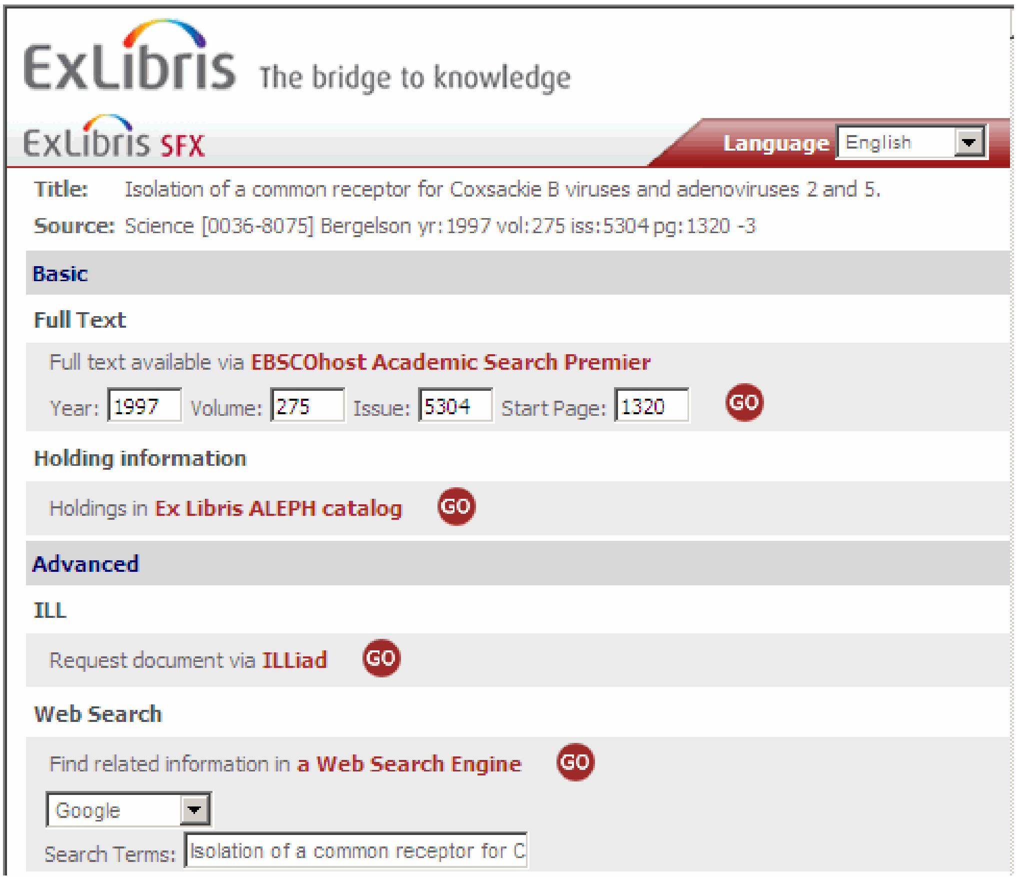
Advanced SFX Menu Template Configuration

This section provides the following information regarding the advanced SFX menu template configuration:

- **Related Objects in XML, OpenURL, and Consortium API** on page **47**

SFX Menu Example

In the example below, the SFX menu uses the advanced SFX menu template. The services on the SFX menu are grouped into Basic and Advanced.



Title: Isolation of a common receptor for Coxsackie B viruses and adenoviruses 2 and 5.

Source: Science [0036-8075] Bergelson yr:1997 vol:275 iss:5304 pg:1320 -3

Basic

Full Text

Full text available via **EBSCOhost Academic Search Premier**

Year: 1997 Volume: 275 Issue: 5304 Start Page: 1320 GO

Holding information

Holdings in **Ex Libris ALEPH catalog** GO

Advanced

ILL

Request document via **ILLiad** GO

Web Search

Find related information in **a Web Search Engine** GO

Google

Search Terms: Isolation of a common receptor for C

Figure 5: Advanced SFX Menu Display

Basic services can be defined in the `basic_grouping.config_ file` in your instance. All other services display in the Advanced services group.

You can choose whether to make the headers for these groups collapsible by editing the `ctx_object.config` file in your instance.

Each of the individual services listed in the example is comprised of the following parts:

- The text describing the type of service, such as **Full text available via X**.

  Changes can be made to this text by editing the `services_public_name.config_file` in your instance.

- The name of the target (displayed in bold).

  You can change the name of a target by editing the target's `PUBLIC_NAME` field in your instance of the KBManager.

Defining the Basic Group of Services

In the `basic_grouping.config_` file, you can define the SFX services to be displayed in the top part of the SFX menu and the order in which they are displayed. The configuration file is used if you want to group services in the Basic or Advanced sections of the SFX menu. (For more information, see **Grouping Services on an SFX Menu Screen** on page **28**).

NOTES:

- The `basic_grouping.config_` file overrides the order of services configured in the `services_public_name.config_` file (for more information, see **Defining the Order of Groups of Services and Text Describing Each Group** on page **29**) even if grouping is not used; therefore, it is important to define the order of services in the SFX menu in the `basic_grouping.config_` file.
- The only service that cannot be configured in the `basic_grouping.config_` file is the bX `getRecommendation` service. This service uses very specific display programs, and its display is controlled by the UNIX configuration file `config/bx.config`. More information about the configuration of the bX recommendation service can be found in the *bX–SFX Configuration Guide*.

To edit the `basic_grouping.config_` file:

1. Telnet to the SFX server and log on as `sfxlcl41`.
2. Enter `cn`.
3. Enter `vi basic_grouping.config_`.
4. You are prompted to break the symbolic link. Enter `y`.
5. In the "`basic_grouping`" section, enter the target services.

```
Section "basic_grouping"
    getFullTxt
    getHolding
EndSection
```

IMPORTANT:

The services not listed in "`basic_grouping`" automatically display as part of the Advanced group.

If desired, you may change the default names for the Basic and Advanced groups of services. See **Changing the .tmpl Files that Constitute the SFX Menu** on page **35**.

6 Save your changes.

Grouping Services on an SFX Menu Screen

In the `ctx_object.config` file, you can specify:

- whether SFX services should be grouped on the SFX menu into Basic and Advanced services
- the option to allow users to collapse and expand these groups on the SFX menu

To edit the `ctx_object.config` file:

1 Connect to the SFX server and log on as the instance user, such as `sfxlcl41`.

2 Enter `cn`.

3 Enter `vi ctx_object.config`.

4 Go to the "`html_menu`" section.

```
Section "html_menu"
    debugging        "1"
    log              "$ENV{SFXCTRL_HOME}/logs/sfxmenu"
    timer            "1"
    keep_log         "1"
    debug_ip_range   "*.*.*.*"
    grouping         "Y"
    collapsible      "Y"
    user_must_pick_single_object "1"
    template         "$ENV{SFXCTRL_HOME}/templates/sfxmenu/
sfxmenu.tmpl"
EndSection
```

5 Set the `grouping` and `collapsible` options.

If `grouping` is set to `Y`, the services are grouped into Basic and Advanced services as defined in the `basic_groupings.config_` file.

Setting `grouping` to `Y` also enables the option to make these groups collapsible on the SFX menu. Set the `collapsible` option to "`Y`" to make groups collapsible on the SFX menu.

If `grouping` is set to `N`, the target services display on the menu without a header or footer for Basic and Advanced services.

6 Save the file.

Defining the Order of Groups of Services and Text Describing Each Group

In the "group_names" section of the services_public_name.config_ file, you can specify or change:

- The names for the headings that display for each group of services.
- The order in which the groups of services display.

NOTES:

- When using the advanced menu, the order of services is configured in the services_public_name.config_ file. However, the services listed in the basic_grouping.config_ configuration file are displayed before any other services - even when grouping into basic and advanced sections is not used. SFX first displays the services listed in basic_grouping.config_ (using the order defined in services_public_name.config_) and afterwards displays the services not listed in basic_grouping.config_ (using the order defined in services_public_name.config_).
- The only service that cannot be configured in the services_public_name.config_ file is the bX getRecommendation service. This service uses very specific display programs, and its display is controlled by the UNIX configuration file config/bx.config. More information about the configuration of the bX recommendation service can be found in the *bX–SFX Configuration Guide*.

To edit the "group_names" section of the services_public_name.config_ file:

1. Telnet to the SFX server and log on as sfxlcl41.
2. Enter cn.
3. Enter vi services_public_name.config_.
4. You are prompted to break the symbolic link. Enter y.

5. Go to the “group_names” section.

```
Section "group_names"
    getFullTxt          "Full Text"
    getSelectedFullTxt  "Full text"
    getDOI              "Full text"
    getMessageNoFullTxt "Full text"
    getHolding          "Holding information"
    getDocumentDelivery "ILL"
    getBookReview       "Reviews"
    getAbstract         "Abstract"
    getTOC              "Table of contents"
    getAlikeRecord      "Reviews"
    getAuthor           "Author"
    getCitedAuthor      "Author"
    getCitedReference   "Reference"
    getReference        "Reference"
    getCitedBook        "Citation information"
    getCitedGenome      "Citation information"
    getCitedRecord      "Citation information"
    getCitedJournal     "Citation information"
    getAuthorEmail      "Author"
    getSubject          "Subject"
    getWebService       "Web Service"
    getWebSearch        "Web Search"
EndSection
```

6. Change the name or phrase in quotation marks to change the group name displayed in the SFX menu.

7. Change the order of the service types to your preference.

   The order of the list of service types displayed on the SFX menu is determined by the order of the service types as entered in the “group_names” section.

   If the getFullTxt service is the first one entered in the “group_names” section, it is the first service that displays on the SFX menu if a FullTxt service is available for the link source.

IMPORTANT:

If you remove the name or phrase in quotation marks, the group name is removed from the SFX menu. The services are still grouped and ordered, but the name of the grouping does not display.

In addition, do not remove the name of the service, such as getFullTxt.

8. Save the file.

Defining the Text Describing Each Service

In the “services” section of the services_public_name.config_ file, you can specify or change the default text describing each service as it displays on the SFX menu.

To edit the “services” section of the services_public_name.config_ file:

1. Telnet to the SFX server and log on as sfxlcl41.
2. Enter cn.
3. Enter vi services_public_name.config_.
4. Go to the “services” section.

```
Section "services"
        getFullTxt          "Full text available via"
        getSelectedFullTxt  "Full text available via"
        getDOI              "Full text available via"
        getMessageNoFullTxt "No Full text available"
        getHolding          "Holdings in"
        getDocumentDelivery "Request document via"
       getBookReview       "Read the review or buy this book
at"
        getAbstract         "Abstract available in"
        getTOC              "Table of contents from"
        getAlikeRecord      "This record at"
        getAuthor           "Other articles by this author? -
- using"
        getCitedAuthor      "Is this author cited? -- using"
        getCitedReference   "Is this article cited? -- using"
        getCitedJournal     "This journal in"
        getReference        "References at"
        getCitedBook        "This book in"
        getCitedGenome      "Get sequence information from"
        getCitedRecord      "This record in"
        getAuthorEmail      "Find this author's email in"
        getSubject          "Find this subject in"
        getWebService       "Find related information in"
        getWebSearch        "Find related information in"
EndSection
```

5. Change the default text to your preferences.
6. Save the file.

Defining the Order for Targets for the Same Service Type

If several target services of the same type are displayed in the SFX menu, they are by default displayed in alphabetical order by the target name. You can change the order of the display in the “sort” section of the sort_services.config_ file.

To edit the "sort" section of the sort_services.config_ file:

1 Telnet to the SFX server and log on as sfxlcl41.

2 Enter cn.

3 Enter vi sort_services.config_.

4 Go to the "sort" section.

5 Change the sort arrangement to match your preferences using the example below as a guide.

```
Section "sort" Type=LIST
#getHolding services
LOCAL_CATALOGUE_ENDEAVOR_VOYAGER
LIBRARY_OF_CONGRESS
#getFullTxt service
SWETSWISE
SPRINGER_LINK_JOURNALS
EndSection
```

In this "sort" example:

- You have activated the following getHolding target services:
  - A link to your local OPAC
  - A link to the Library of Congress

  You want the local OPAC link to display first and the Library of Congress link to display second.
- You have activated full-text access to journals via SwetsWise and Springer. If more than one full-text link displays in the SFX menu, you want the link to the SwetsWise service to display first and the link to the Springer service to display second.

IMPORTANT:

All target services not listed in the "sort" section are sorted alphabetically.

When specifying a target in the "sort" section, use the correct TARGET name found in the KBManager.

6 Save your changes.

Force Targets to Display at the Bottom of a List of Services

You may want to suppress certain targets so that they always display at the bottom of the list of services in the SFX menu. A target defined in the bottom_list.config_ file displays at the bottom of each list of services in which it is found. These changes can be made in the "bottom" section.

To edit the "bottom" section of the bottom_list.config_ file:

1 Telnet to the SFX server and log on as sfxlcl41.

2 Enter cn.

3 Enter vi bottom_list.config_.

4 Go to the "bottom" section.

```
Section "bottom" Type=LIST
LIBRARY_OF_CONGRESS
EndSection
```

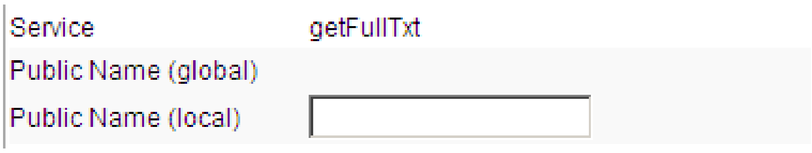
5 Specify the target to display at the bottom.

6 Save the file.

Changing the Text Explaining a Particular Service

Use the KBManager to change the default explanation text for one particular service. To do this, enter the text in the PUBLIC_NAME field of the TARGET_SERVICE table. The text you enter in the PUBLIC_NAME field overrides the default explanation text defined in the services_public_name.config_ file.

For example, if you have created a getWebService service that allows the user to send an e-mail with comments or questions to the library, according to the services_public_name.config_ file, the default text for this service is **Find related information in**. If, instead of the default text, you want the SFX menu to display **Send questions or comments to**, enter this text in the PUBLIC_NAME field of your new service. The new text overrides the default text specified in the services_public_name.config_ file.



Service	getFullTxt
Public Name (global)	
Public Name (local)	

Figure 6: Changing the Text Explaining a Particular Service

HTML Templates of the Advanced Menu Template

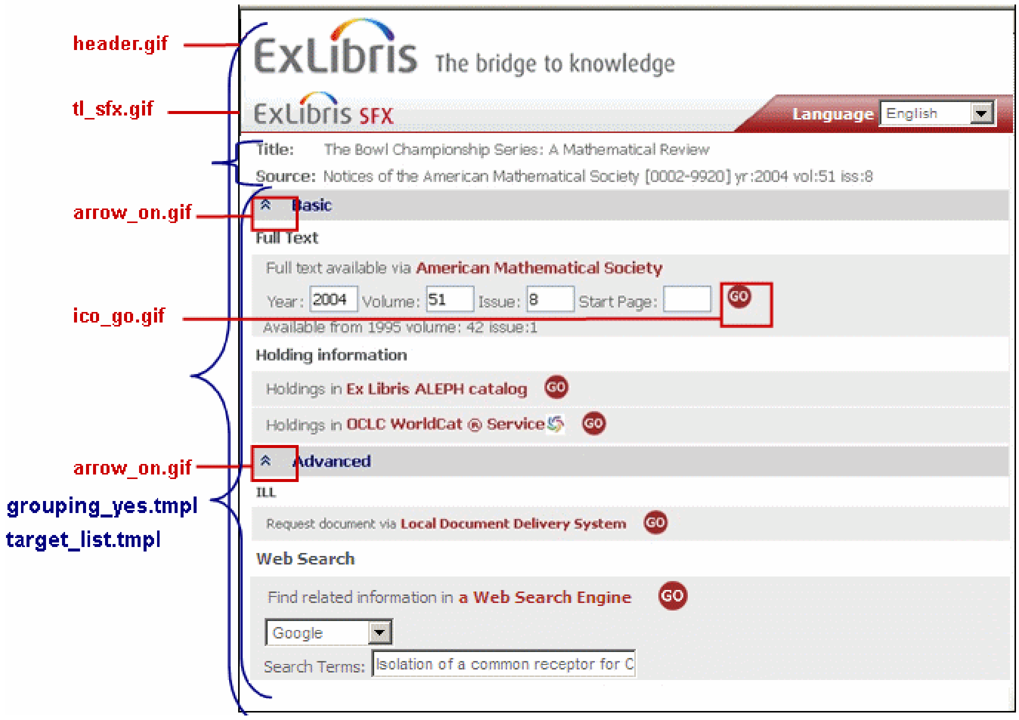
The SFX menu is created from the following .tmpl files that are located in /exlibris/sfx_ver/sfx4_1/<instance>/templates/sfxmenu.

- banner.tmpl– This file is used to create the reference box at the top of SFX menu.
- sfxmenu.tmpl– This file points to the CSS style sheet and grouping templates.

- grouping_yes.tmpl– This file contains the layout instructions for menus when grouping is set to yes and points to the target_list.tmp file.
- grouping_no.tmpl– This file contains the layout instructions for menus when grouping is set to no and points to the target_list.tmp file.
- target_list.tmpl – This file contains Perl instructions to create service links and references target displayers.

The display characteristics of the SFX menu are defined in the following .css file that is referenced by sfxmenu.tmpl and the HTML style sheet:

/exlibris/sfx_ver/sfx4_1/<instance>/templates/css/
publicstyle.css



Title: The Bowl Championship Series: A Mathematical Review
Source: Notices of the American Mathematical Society [0002-9920] yr:2004 vol:51 iss:8

Basic

Full Text

Full text available via American Mathematical Society

Year: 2004 Volume: 51 Issue: 8 Start Page:

Available from 1995 volume: 42 issue:1

Holding information

Holdings in Ex Libris ALEPH catalog

Holdings in OCLC WorldCat ® Service

Advanced

ILL

Request document via Local Document Delivery System

Web Search

Find related information in a Web Search Engine

Google

Search Terms: Isolation of a common receptor for C

Figure 7: SFX Menu Display Characteristics

Changing Images in the SFX Menu

The following .gif files provide the default images used in the SFX menu:

- tl_sfx.gif
- header.gif
- arrow_on.gif
- arrow_off.gif

These files are located in /exlibris/sfx_ver/sfx4_1/<instance>/
templates/img/sfxmenu.

Optionally, you may use your custom images instead of these default images.

To use custom images instead of default images:

1. Create backups of the default images.
   a. Log on to your server as the instance user.
   b. Copy the following files from `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/sfxmenu`:
      - `cp header.gif header-bak.gif`
      - `cp tl_sfx.gif tl_sfx-bak.gif`
      - `cp arrow_on.gif arrow_on-bak.gif`
      - `cp arrow_off.gif narrow_off-bak.gif`
2. Create new images and save them with the same names as the ones you are replacing.
3. FTP your new `.gif` files to `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/sfxmenu`.

NOTE:

> The location and names of the images should not be changed. Rather, the original images should be replaced with your local images. This eliminates the need to make changes to any corresponding `.tmpl` files (see **Changing the .tmpl Files that Constitute the SFX Menu** on page **35**).

Changing the .tmpl Files that Constitute the SFX Menu

This section describes how to edit the `.tmpl` files that constitute the SFX menu. Any changes made to the `.tmpl` files are preserved when the SFX software is updated.

Each `.tmpl` file contains parameters that refer to the SFX data elements to be displayed in the SFX menu. For example, in `banner.tmpl`, you find `<TMPL_VAR NAME="ATITLE">` and `<TMPL_VAR NAME="ISSN">` to refer to metadata that displays in the top part of the SFX menu.

When SFX generates an SFX menu, `<TMPL_VAR NAME="ATITLE">` is replaced by the article title and the journal ISSN. Similarly, JournalTitle/BookTitle, date, volume, issue, and start page information are displayed when available.

CAUTION:

> It is recommended that you not change these parameters because they are used by the SFX programs to generate a valid SFX menu.

If you do need to make changes, create backup files of the `.tmpl` files.

To back up files:

1. Log on to your SFX server as the user of the instance where you want to make changes (such as the `sfxlcl41` user) to make changes to the `sfxlcl41` instance.
2. Enter `sx`.
3. Enter `cd templates/sfxmenu`.
4. Enter `cp xxxx.html xxxx-bak.html`.

   For example:

   ```
   cp banner.tmpl banner-bak.tmpl
   ```

The main `.tmpl` files that constitute the SFX menu are:

- `banner.tmpl`
- `sfxmenu.tmpl`
- `grouping_yes.tmpl`
- `grouping_no.tmpl`
- `target_list.tmpl`

The `sfxmenu.tmpl` file allows you to specify or change the following:

- The referenced Cascading Style Sheets used when generating the menu html.

  ```
  <LINK REL="stylesheet" TYPE="text/css" href="/<TMPL_VAR
  INSTANCE>/css/public/style.css">
  ```

- The image across the top of the SFX menu. By default, the `header.gif` file is used.

  ```
  <IMG SRC="/<TMPL_VAR INSTANCE>/img/sfxmenu/header.gif"
  WIDTH="455" HEIGHT="48" BORDER="0" ALT="ExLibris Electronic
  Library">
  ```

- The SFX icon located at top of menu.

  ```
  <IMG SRC="/<TMPL_VAR INSTANCE>/img/sfxmenu/tl_sfx.gif"
  WIDTH="103" HEIGHT="35" BORDER="0" ALT="SFX">
  ```

To change the image:

1. Enter `vi sfxmenu.tmpl`.

   The system prompts you for the following:

   ```
   [break symbolic link 'sfxmenu.tmpl' ?] [y/N]
   ```

2. Enter `y` and the file is created as a local file.

3. Make your changes to the `sfxmenu.tmpl` file.

   The `banner.tmpl` file allows you to change or specify the text used in the banner to display journal and article information.

4. Save your changes.

To change the `banner.tmpl` file:

1. Enter `vi banner.tmpl`.

   The system prompts you for the following:

   ```
   [break symbolic link 'banner.html'?] [y/N]
   ```

2. Enter `y` and the file is created as a local file.
3. Make your changes to the `banner.tmpl` file.

   The `grouping_yes.tmpl` file allows you to change or specify the text used to describe the Basic and Advanced service groupings when used in the SFX menu. A different name can be defined for collapsible menus.

```
<TD class="GroupTitle" height="22" width="100%">
<TMPL_IF COLAPSIBLE_ENABLED><a
href="javascript:toggleBasic();" class="GroupTitle">Basic</a>
<TMPL_ELSE>
Basic
</TMPL_IF></TD>

<TD class="GroupTitle" height="22" width="100%">
<TMPL_IF COLAPSIBLE_ENABLED><a
href="javascript:toggleAdvanced();"
class="GroupTitle">Advanced</a>
<TMPL_ELSE>
Advanced
</TMPL_IF>
</TD>
```

4. Save your changes.

To change the `grouping_yes.tmpl` file:

1. Enter `vi grouping_yes.tmpl`.

   The system prompts you for the following:

   ```
   [break symbolic link 'grouping_yes.html' ?] [y/N]
   ```

2. Enter `y` and the file is created as a local file.

3. Make your changes to the `grouping_yes.tmpl` file.

   The `target_list.tmpl` file allows you to change or specify the button to go to a particular target. By default, the `ico_go.gif` image is used.

```
<IMG SRC="/<TMPL_VAR NAME='INSTANCE'>/img/sfxmenu/ico_go.gif"
WIDTH="20" HEIGHT="20" BORDER="0" ALT="GO" HSPACE="10">
```

4. Save your changes.

To change the `target_list.tmpl` file:

1. Enter `vi target_list.tmpl`.

   The system prompts you for the following:

   `[break symbolic link 'target_list.html'?] [y/N]`

2. Enter `y` and the file is created as a local file.
3. Make your changes to the `target_list.tmpl` file.

   Make target names clickable.

   It is possible to make the name of each target and/or the text describing each target clickable. To do this, you need to edit the target name section of the `target_list.tmpl` file.

   If you want the target name to be hyperlinked, add the lines as shown in the example below.

   - Without hyperlinks in the service and target name:

```
<table border=0>
<tr>
<td nowrap width="1%" class="Label"><TMPL_VAR
NAME='SERVICE_NAME'>
<span class="LabelHighlightInner"><TMPL_VAR
NAME='TARGET_NAME'></span>
</td>
```

- With hyperlinked target name:

```
<table border=0>
<tr>
<td nowrap width="1%" class="Label"><TMPL_VAR
NAME='SERVICE_NAME'>
<TMPL_IF IS_LINKABLE>
<A HREF="javascript:openWindow(this,'<TMPL_VAR
NAME=FORM>');">
<span class="LabelHighlightInner"><TMPL_VAR
NAME='TARGET_NAME'></span></A>
<TMPL_ELSE>
<span class="LabelHighlightInner"><TMPL_VAR
NAME='TARGET_NAME'></span>
</TMPL_IF>
</td>
```

If you want both the target name and the text describing the target to be hyperlinked, add the lines shown in the example below:

- Without hyperlinks in the target name:.

```
<table border=0>
<tr>
<td nowrap width="1%" class="Label">
<TMPL_VAR NAME='SERVICE_NAME'>
<span class="LabelHighlightInner">
<TMPL_VAR NAME='TARGET_NAME'></span>
</td>
```

- With hyperlinked service and target name:

```
<table border=0>
<tr>
<td nowrap width="1%" class="Label">
<TMPL_IF IS_LINKABLE>
<A HREF="javascript:openWin(this);window.document.<TMPL_VAR
FORM>.submit();">
<TMPL_VAR NAME='SERVICE_NAME'>
</A>
<TMPL_ELSE>
<TMPL_VAR NAME='SERVICE_NAME'>
</TMPL_IF>
<TMPL_IF IS_LINKABLE>
<A HREF="javascript:openWin(this);window.document.<TMPL_VAR
FORM>.submit();">
<span class="LabelHighlightInner"><TMPL_VAR
NAME='TARGET_NAME'></span>
</A>
<TMPL_ELSE>
<TMPL_VAR NAME='TARGET_NAME'>
</TMPL_IF>
</td>
```

The `style.css` file allows you to change or specify the style (fonts, colors, and layout) of the SFX menu.

To change or specify the style of the SFX menu:

1. Enter `cd /exlibris/sfx_ver/sfx4_1/sfxinstance/templates/css/public/`.
2. Enter `vi style.css`.

   The system prompts you for the following:

   ```
   [break symbolic link 'style..css'?] [y/N]
   ```
3. Enter `y` and the file is created as a local file.
4. Make your changes to the `style.css` file.

Creating Language-Specific Menu Templates

It is possible to use and create language-specific `.tmpl` files when an SFX menu is generated.

The SFX installation comes with a set of language templates for English that is used as the default and has no file extensions. Additionally, there are language template sets for Dutch and Spanish. More language template sets can be created. It is possible to define which set of templates are the defaults.

When a user chooses available languages from a drop-down list on the SFX menu, the defined language templates are consulted and a new menu is generated based on these. If the program cannot find a template file with a language extension, it uses the default set of SFX menu template files.

For example, the following default template files are consulted when a user chooses **Espanol** from the SFX menu:

```
/exlibris/sfx_ver/sfx4_1/instance/templates/sfxmenu

banner.tmpl.es
sfxmenu.tmpl.es
target_list.tmpl.es
error.tmpl.es
grouping_yes.tmpl.es
grouping_no.tmpl.es
/exlibris/sfx_ver/sfx4_1/instance/config/
services_public_name.config_.es
```

To define the default language if it is different from English:

1. In the /sfx_ver/sfx4_1/<instance>/config/map_languages.config_ configuration file, add a default_language section.
2. Specify the default language of your choice.

   See the example below where the Dutch language templates are used as the default set.

```
Section "default_language"
    default Dutch
EndSection

Section "map_languages"
    Dutch   "du"
    Spanish "sp"
EndSection
```

3. Save your changes.

To specify additional language-specific templates:

1. Create additional language-specific .tmpl files.
   a. Telnet to your SFX server and log on as sfxlcl41 (or the instance where you want to use the language-specific SFX menu).
   b. Enter sx.
   c. Enter cd templates/sfxmenu.
   d. Enter cp banner.tmpl banner.tmpl.<language extension>.

      For example, Enter cp banner.tmpl banner.tmpl.ger for German.
   e. Enter vi banner.tmpl.<language extension>.
   f. Make the changes you want, and save the changes by typing :wq.
   g. Repeat this process for additional menu templates.

      When German is selected from the SFX menu, the SFX program uses the .tmpl files with the .ger extension. If it cannot find a .tmpl file with a .ger extension, it uses the default .tmpl files (without extensions).
2. Specify the language name and its .tmpl extension that is to be used in the map_languages.config file.
   a. Telnet to the SFX server and log on as sfxlcl41 (or the instance where you want to use the language-specific SFX menu).
   b. Enter cn.
   c. Enter vi map_languages.config.
   d. Edit the "map_languages" section.

e Add the language name as it displays in the SFX menu drop-down list.

f Tab and add the extension of your choice. In the example below, "ger" is used for German.

```
Section "map_languages"
    Espanol "es"
    German "ger"
    Dutch   "du"
EndSection
```

g Save your changes, and quit the file by typing `:wq`.

NOTE:

Verify that the `sfx_language_code_to_sfxmenu` section exists in the `map_languages.config_` file. Do not remove this section from the file. For more information, see **Setting user_profile Cookies** on page **207**.

3 Add a new language option to the drop-down navigation list of the SFX menu in `/exlibris/sfx_ver/sfx4_1/instance/templates/sfxmenu/sfxmenu.tmpl`.

a Telnet to the SFX server and log on as `sfxlcl41` (or the instance where you want to use the language-specific SFX menu).

b Enter `sx`.

c Enter `cd templates/sfxmenu`.

d Enter `vi sfxmenu.tmpl`.

e Add the name that you defined in the `map_language.config` file to the following section of the `sfxmenu.tmpl` file.

```
Language
<SELECT NAME="language" onChange=set_user_profile()>
<OPTION value="English"<TMPL_VAR NAME='English'
ESCAPE='HTML'>>English</OPTION>
<OPTION value="Espanol"<TMPL_VAR NAME='Espanol'
ESCAPE='HTML'>>Espanol</OPTION>
<OPTION value="German"<TMPL_VAR NAME='German'
ESCAPE='HTML'>>German</OPTION>
</SELECT>
```

In this example, German was added.

Overview of SFX Menu-Related Features, Files, and Tools

Table 4. Where Found in SFX Admin

Where Found in SFX Admin	Function
Setup & Administration > Configuration: Menu Configuration	Display logic Suppressing a particular target when another specific one is present
KBManager: TARGET Public Name	Controls target display name on menu
KBManager: `TARGET_SERVICE` Public Name	Controls `OVERRIDE` default service wording See `services_public_name.config_` below

Table 5. Where Configuration Files for Instance - Shortcut: cn Found

Where Configuration Files for Instance - Shortcut: cn Found	Function
`services_public_name.config_`	Text describing service and/or order of service - Text for "`group name`" - Regrouping of service on the menu
`ctx_object.config`	Specify grouping menu into Basic and Advanced sections Make headers for groups collapsible
`basic_grouping.config_`	Basic services can be defined All other services fall into extended services
`sort_services.config_`	Sorting Push to top of list (services of the same type)
`bottom_list.config_`	Sorting Push to bottom of list (services of the same type)

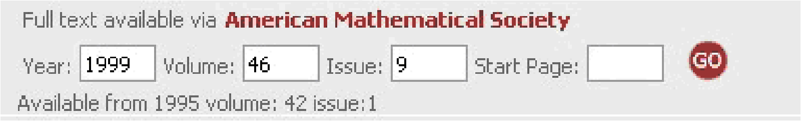
Table 6. Where Template Files - Shortcut:tm Found

Where Template Files - Shortcut:tm Found	Function
`/css/public/style.css`	Central style sheet
`/img/sfxmenu`	Storage of the images for banner, header, arrows, and so on
`/sfxmenu/sfxmenu.tmpl`	Central file References all the images Can remove the language drop-down in the upper-right corner
`/sfxmenu/grouping_yes.tmpl`	Change text describing “`Basic`” and “`Advanced`” sections
`/sfxmenu/target_list.tmpl`	Can make the target names clickable hyperlinks

Target Displayers

A target displayer is a program that determines what a service looks like on the SFX menu.

Each service type has a default target displayer. For example, a full-text service typically has four small input boxes and looks like the following example in the advanced menu.



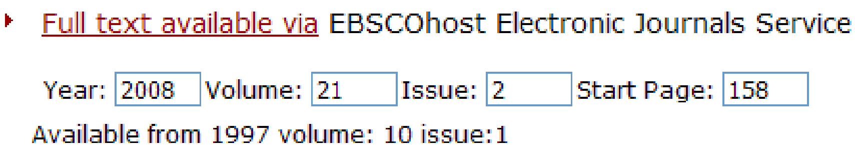
Full text available via **American Mathematical Society**

Year: 1999 Volume: 46 Issue: 9 Start Page:

Available from 1995 volume: 42 issue:1

Figure 8: Full-Text Service in the Advanced Menu

In the simplified menu, the displayer looks like the following:



Full text available via EBSCOhost Electronic Journals Service

Year: 2008 Volume: 21 Issue: 2 Start Page: 158

Available from 1997 volume: 10 issue:1

Figure 9: Full -Text Service in the Simplified Menu

The default target display is used by most existing `TARGET_SERVICES`.

In the KBManager, you can see which target display is used.

- When a TARGET_SERVICE uses the default parser, the DISPLAYER field in the KBManager is left blank.
- When a TARGET_SERVICE uses a special display parser, the name of the parser is filled in the DISPLAYER field of that particular TARGET_SERVICE.

For example, the getWebSearch service for the Web Search Engine target does not use volume or start page information, so a special displayer shows a drop-down list of different search engines instead of the input boxes.

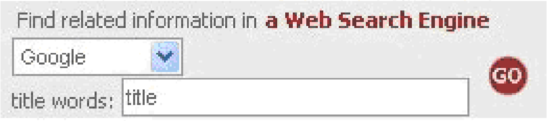


Figure 10: Drop-Down List of Search Engines

In the displayer field of the TARGET_SERVICE for WWW_SEARCH_ENGINES, you see WWW::ALL.

This refers to the location of a specific target displayer on the SFX server in /exlibris/sfx_ver/sfx4_1/instance/lib/Parsers/TargetDisplayer/WWW/ALL.pm.

Another example of a service that uses a special display parser is UNION_ALL_CATALOGUES, with a drop-down list of different union catalogs.

To edit or view the target displayers:

1. Telnet to the SFX server and log on as sfxlcl41.
2. Enter tp.
3. Enter cd ../TargetDisplayer/.

   This is where you find the different target displayers.

   If you would like to prevent the year/volume/issue/startpage input boxes from displaying completely for a particular target service, you can apply a special NO_FILL_IN displayer.

To prevent the year/volume/issue/startpage input boxes from displaying completely:

1. Click the **E** button on the target service level of the target of your choice.
2. Go to the **Displayer** field (the next field under parse_param).
3. Enter FT::NO_FILL_IN in the **Displayer** field.
4. Save your changes.

Using a Special Displayer to Show Date Threshold Information in the SFX Menu

In some `SOURCES`, very little metadata (no date, volume, issue, or page information) is available but links to full-text services display on the SFX menu.

This is true for:

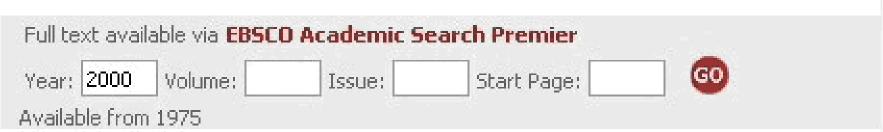
- OPACs
- An HTML list of electronic journals on a customer's library pages
- The CitationLinker form
- A journal index, such as Ulrichs periodicals database

To provide more information about full-text availability, an extra line with availability information can be added to each full-text service in the SFX menu. (The same availability information is also displayed in the DirectLink sidebar, if the sidebar is in use in the SFX instance.) This special displayer shows the relevant `OBJECT_PORTFOLIO` date and time difference (`timediff`) thresholds in the SFX menu. The format of these thresholds is changed to make them more user-friendly.

In the example below, the following date threshold exists in the Academic Search Premier `getFullTxt OBJECT_PORTFOLIO` for the American Heart Journal:

```
$obj->parsedDate(">=",1975,140,1)
```

The special displayer looks like the following:



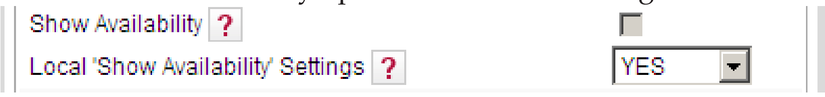
Full text available via **EBSCO Academic Search Premier**

Year: 2000 Volume: Issue: Start Page:

Available from 1975

Figure 11: Special Displayer

To use this special displayer, you need to access the KBManager and go to the List of Sources. Click the E **Edit** button to edit the source for which you want the availability information to be displayed in the SFX menu, and click the check box for the Show Availability option as in the following:



Show Availability ? □

Local 'Show Availability' Settings ? YES

Figure 12: Show Availability

You can use following OpenURL to create new screen shot of the SFX menu:

```
http://il-sfx07:3410/sst1_4?url_ver=Z39.88-
2004&url_ctx_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Actx&ctx_ver=Z
39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-
8&rfr_id=info%3Asid%2Fsfxit.com%3Akbmanager&sfx.ignore_date_t
hreshold=1&rft.object_id=954925246074
```

```
$obj->parsedDate(">",1975,140,1) && $obj-
>parsedDate("<",2015,200,1)
```

is displayed as:

```
Available after 1975 volume: 140 issue:1 until (not including)
2015 volume: 200 issue:1
```

and

```
$obj->parsedDate(">=",1975,140,1) && $obj-
>parsedDate("<=",2015,200,1)
```

is displayed as:

```
Available from 1975 volume: 140 issue:1 until (and including)
2015 volume: 200 issue:1
```

Related Objects in XML, OpenURL, and Consortium API

In all cases where SFX does not respond by presenting the SFX menu, the behavior for related objects is described below:

- XML OpenURL

  Only close relations are included in the XML responses. This happens only if the related object display is turned on for the SFX menu. There is no indication that these services originate from related objects.

- Consortia API

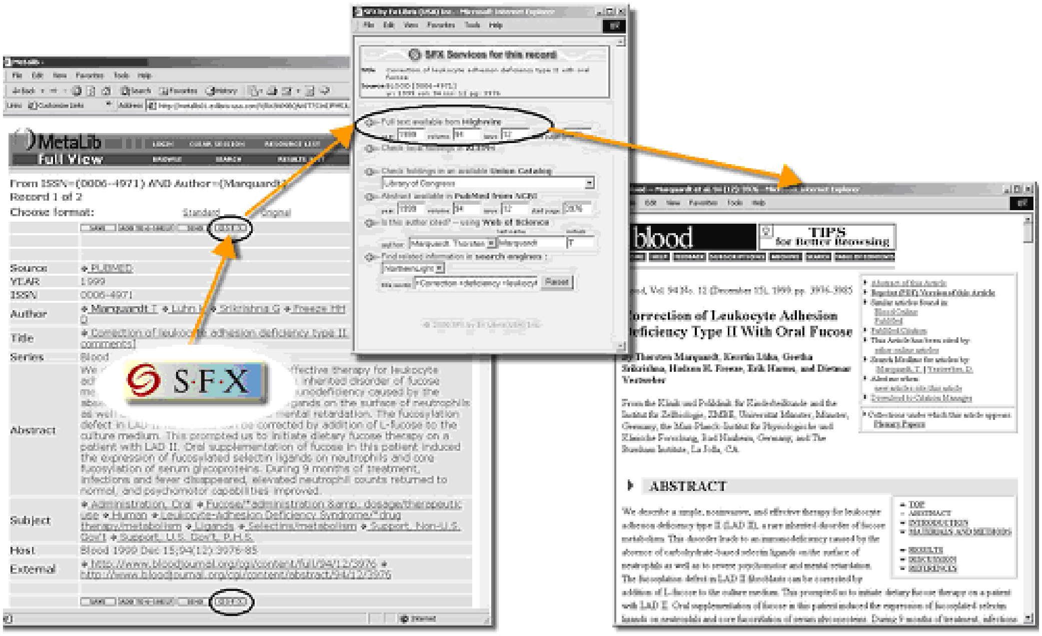
  When retrieving additional services using consortia API, services from the related objects are displayed based on configuration both in the local and remote instance.

  - If the related object display is turned on in the local instance only, only related services from the local instance are included in the SFX menu.
  - If the related object display is turned on in both the local and remote instance, related services from both instances are included in the SFX menu.

SFX Window Behavior

By default, three windows are open during the SFX flow: the source, the SFX menu, and the target windows.

- Source window – the database from which you begin your search. If you click the SFX button in the top bar of the record, the SFX menu opens.
- SFX menu window – contains a list of available services. When you click one of the services, the target window opens.
- Target window – the service requested in the SFX menu. In the following example, the user requests a full-text service, so the target window displays the HighWire full-text version corresponding to the original abstract record.



MetaLib
Full View
S·F·X
SFX Services for this record
Full text available from HighWire
Check holdings in an available Union Catalog
Abstract available in PubMed from NCBI
Is this author cited? – using Web of Science
Find related information in search engines :
blood
TIPS for Better Browsing
ABSTRACT

From ISSN=(0006-4971) AND Author=(Marquardt)
Record 1 of 2
Choose format: Standard Original

Source	PUBMED
YEAR	1999
ISSN	0006-4971
Author	Marquardt T, Luhn K, Srikrishna G, Freeze HH
Title	Correction of leukocyte adhesion deficiency type II [comments]
Series	Blood
Abstract	...effective therapy for leukocyte ... inherited disorder of fucose ... unodeficiency caused by the ... gands on the surface of neutrophils as well ... mental retardation. The fucosylation defect in LAD II ... be corrected by addition of L-fucose to the culture medium. This prompted us to initiate dietary fucose therapy on a patient with LAD II. Oral supplementation of fucose in this patient induced the expression of fucosylated selectin ligands on neutrophils and core fucosylation of serum glycoproteins. During 9 months of treatment, infections and fever disappeared, elevated neutrophil counts returned to normal, and psychomotor capabilities improved.
Subject	Administration, Oral; Fucose/*administration &amp; dosage/therapeutic use; Human; Leukocyte-Adhesion Deficiency Syndrome/*drug therapy/metabolism; Ligands; Selectins/metabolism; Support, Non-U.S. Gov't; Support, U.S. Gov't, P.H.S.
Host	Blood 1999 Dec 15;94(12):3976-85
External	http://www.bloodjournal.org/cgi/content/full/94/12/3976 ; http://www.bloodjournal.org/cgi/content/abstract/94/12/3976

Correction of Leukocyte Adhesion Deficiency Type II With Oral Fucose

Figure 13: SFX Windows

If you click the SFX button for a second record in the source window, a new SFX menu is opened in a new window or new browser tab (depending on browser settings).

When another service is clicked in the SFX menu, a new the target window is opened in a new window or new browser tab (depending on browser settings).

Changing the Size of the Target Window

It is possible to change the size of the target window that is created after clicking a link to the target from the SFX menu.

To change the size of the target window:

1 Log on to the SFX server as the instance user.

2 Navigate to the template directory:

```
cd /exlibris/sfx_ver/sfx4_1/<instance>/templates/js/sfxmenu
```

3 Access the JavaScript files listed below. In both files, you need to change the current window size (marked in red) to the window size of your choice.

- `main.js`

```
function openWin(obj)
var options =
"toolbar=yes,location=yes,directories=yes,buttons=yes,status=yes";
options +=
",menubar=yes,scrollbars=yes,resizable=yes,width=800,height=600";

function openWin_location
var options =
"toolbar=yes,location=yes,directories=yes,buttons=yes,status=yes";
options +=
",menubar=yes,scrollbars=yes,resizable=yes,width=800,height=600";
```

- `gbs.js`

```
function openWindow(obj, form_name)
var options =
"toolbar=yes,location=yes,directories=yes,buttons=yes,status=yes";
options +=
",menubar=yes,scrollbars=yes,resizable=yes,width=800,height=600";
```

NOTE:
When changing the size of the target window, remember that your window size may display differently on computers with different screen resolutions.

4 Save your changes.

Including COinS Representation in the SFX Menu

It is possible to include a COinS representation of the SFX ContextObject in the SFX menu HTML span tag. COinS stands for ContextObject in Span. COinS allows citation metadata to be in embedded into HTML so that the client-side software (such as browser plug-ins and extensions such as LibX and Zotero) can make use of this metadata.

A COinS is an OpenURL ContextObject inside a hidden `<span>` tag of the HTML page.

The specification of COinS can be found in http://ocoins.info/.

For example:

For the OpenURL http://demo.exlibrisgroup.com:3210/demo?genre=article&title=Proceedings%20of%20the%20second%20ACM%20SIGPLAN%20international%20%20&issn=&date=1997&volume=&issue=&spage=&atitle=A%20practical%20subtyping%20system%20for%20Erlang&aulast=Marlow&auinit=S, the COinS representation looks like the following:

```
<span class="Z3988" title="ctx_ver=Z39.88-2004&amp;url_ver=Z39.88-
2004&amp;ctx_enc=info%3Aofi%2Fenc%3AUTF-
8&amp;ctx_id=10_1&amp;rft.auinit=S&amp;rft.genre=article&amp;rft.ye
ar=1997&amp;rfr_id=info%3Asid%2Fopenly%3Aopenurlref&amp;rft.aulast=
Marlow&amp;rft.atitle=A%20practical%20subtyping%20system%20for%20Er
lang&amp;rft.jtitle=Proceedings%20of%20the%20second%20ACM%20SIGPLAN
%20international&amp;rft.date=1997&amp;rft_val_fmt=info:ofi/
fmt:kev:mtx:journal">
</span>
```

The strength of COinS is that it can provide an alternative to the conventional way of an OpenURL-enabling of resources, namely, contacting vendors and turning on SFX buttons in their databases.

Freely available resources typically do not have subscribers and have difficulty discovering a user’s institutional affiliation, which is needed to create an OpenURL. By including COinS metadata, these resources can be made OpenURL-compliant. Client-side software uses COinS to add links to this non-subscription-based content.

COinS can also be used for many other purposes. For a list of COinS implementations, refer to http://ocoins.info/. A short description of some COinS tools is also provided in **Applications that Use COinS** on page **51**.

Note that the quality and quantity of the metadata in the ContextObject determines whether the COinS information is useful. It is not possible to guarantee that every OpenURL creates a usable COinS representation.

Setup

To enable the inclusion of the COinS in the SFX menu, it is necessary to add a configuration option to the following file:

`/exlibris/sfx_ver/sfx4_1/<instance>/config/ctx_object.config`

Add the following line in the “`html_menu`” section:

```
include_coins   "Y"
```

The "html_menu" section looks like the following:

```
 debugging       "0"
    log             "$ENV{SFXCTRL_HOME}/logs/sfxmenu"
    keep_log        "0"
    debug_ip_range  "*.*.*.*"
    grouping        "Y"
    collapsible     "Y"
    user_must_pick_single_object "1"
    template        "$ENV{SFXCTRL_HOME}/templates/sfxmenu/
sfxmenu.tmpl"
    verde_link      "0"
    include_coins   "Y"
EndSection
```

After the setup, send an OpenURL to the SFX instance and check that the HTML source includes the following tag:

```
<span class="Z3988" title"[ctx obj]">
```

See the following example of an OpenURL to be used for testing:

```
<base-URL>?id=pmid:15486829
```

Applications that Use COinS

- LibX

  LibX is a browser plug-in for Firefox and Internet Explorer that provides direct access to your library's resources. It includes a toolbar and a right-click context menu, and supports searches against the library catalog (OPAC) as well as against an OpenURL linking server. Information about LibX can be found at http://libx.org.

- OpenURL Referrer

  This extension enables the browser to display OpenURL links on Web pages with CoinS embedded in the HTML source code. If COinS is turned on for the SFX menu, you see an SFX button in the menu banner.

  The extension is available for both IE 6+7 and Firefox. Information can be found at http://openly.oclc.org/openurlref/.

- Zotero

  Zotero is a library metadata utility extension for Firefox. Its purpose is to help you collect, manage, and cite your research sources. Information about Zotero can be found at http://zotero.org.

Configuring the SFX Menu Interface Language

You can configure the SFX menu to be available for more than one language.

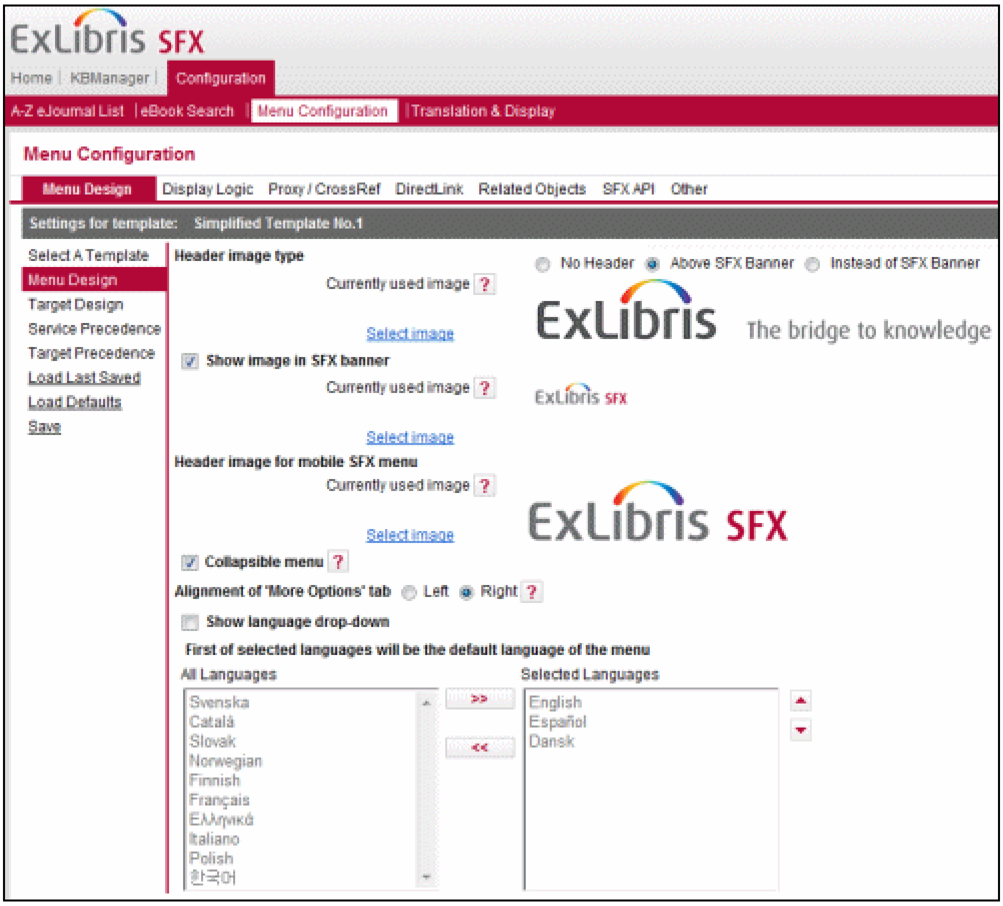
Configuring the SFX Menu Language in the SFX Admin Center

The language configuration depends on the SFX menu template selected.

Simplified Menu Template

To configure the simplified menu template of the SFX menu to be available for more than one language:

1 In the SFX Admin Center, click **Set-up and Administration** > **Configuration** > **Menu Configuration** > **Menu Design**. The following is displayed:



Settings for template: Simplified Template No.1

Header image type ○ No Header ● Above SFX Banner ○ Instead of SFX Banner

Currently used image ?

☑ Show image in SFX banner

Currently used image ?

Header image for mobile SFX menu

Currently used image ?

☑ Collapsible menu ?

Alignment of 'More Options' tab ○ Left ● Right ?

☐ Show language drop-down

First of selected languages will be the default language of the menu

All Languages	Selected Languages
Svenska Català Slovak Norwegian Finnish Français Ελληνικά Italiano Polish 한국어	English Español Dansk

Figure 14: Menu Design

2. Select **Show language drop-down**.
3. Select the languages that you want to appear in the language drop-down from the All Languages pane and click the right arrow >> to move them to the Selected Languages pane. The first language in the Selected Languages pane is the default language.

NOTE:
If you only move one language to the Selected Languages pane, the SFX menu appears only in that language and no drop-down list appears.

Advanced Menu Template

For information concerning configuring the advanced menu template of the SFX menu to be available for more than one language, see **Creating Language-Specific Menu Templates** on page **40**.

Setting the SFX Menu Language in an External Application

The following two methods exist to set the SFX menu language from an external application:

- via the `language` user profile cookie set via an external application, the SFX menu, or the SFX A-Z language drop-down list.
- via the `req.language` OpenURL element

For both methods, the language has to be set up according to the configuration described in **Configuring the SFX Menu Language in the SFX Admin Center** on page **52**.

NOTE:
If a language is set in an external application that is not configured in the SFX Admin Center, the SFX menu default language is used.

Setting the SFX Menu Language via a User Profile Cookie

For information on setting the SFX menu language via a user profile cookie, see **Setting user_profile Cookies** on page **207**.

NOTE:
If the user's browser has an SFX language cookie, the cookie takes precedence over the OpenURL `req.language` parameter, except when an OpenURL is sent from a Primo source, in which case the `req.language` parameter sent by Primo takes precedence over the language cookie.

Setting the SFX Menu Language via the req.language OpenURL Element

You can set the SFX menu language via the following OpenURL element: `req.language=<ISO 639-2(B) language code>`. SFX supports the 3-letter language codes of the ISO 639-2 (B) standard. SFX defaults to the default language configured if a language that is not supported is set.

The following OpenURL example sets the SFX menu interface language to Spanish:

```
http://demo.exlibrisgroup.com:3210/demo?ctx_ver=Z39.88-
2004&url_ver=Z39.88-2004&ctx_enc=UTF-8&rft.issn=0894-
9867&rft.year=2011&rft_id=info%3Apmid%2F21812037&rft.atitle=P
redicting%20pediatric%20posttraumatic%20stress%20disorder%20a
fter%20road%20traffic%20accidents%3A%20the%20role%20of%20pare
ntal%20psychopathology.&rft.issue_start=4&rft.object_id=95492
5559531&rft.volume_start=24&rft.volume=24&rft.month=8&rft.gen
re=article&rft.issue=4&rft.doi=10.1002%2Fjts.20667&rft_val_fm
t=info:ofi/fmt:kev:mtx:journal&req.language=spa
```

SFX Menu for Mobile Devices

You can change the display of the mobile SFX menu by changing the following files in the following locations:

NOTE:
It is recommended to make changes only to images and CSS files. If you make changes to the template files, you will need to manually implement in the localized template files any future updates made to the template files distributed as part of a revision update.

- Mobile SFX menu header images:

  `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/simplified_template1/mobile/default/`

  `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/simplified_template1/mobile/custom/`

NOTES:

- Do not change the images in this directory. Instead, use **SFX Admin Center > Configuration > Menu configuration > Menu Design** to upload and select the header images.
- A different header image is used for the CitationLinker and multi-object window. For this header image, upload the image in the location specified below.

- Location of other mobile SFX menu images (including service type icons):

  `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/`

- CSS files for the mobile SFX menu:
  - Global css files that should not be changed:

    ```
    /exlibris/sfx_ver/sfx4_1/<instance>/templates/css/public/
    mobile/style.css
    ```

    ```
    /exlibris/sfx_ver/sfx4_1/<instance>/templates/css/public/
    mobile/style_rtl.css [Hebrew css]
    ```

  - Local CSS files that can be used to override the global CSS classes:

    ```
    /exlibris/sfx_ver/sfx4_1/<instance>/templates/css/public/
    mobile/custom.css
    ```

    ```
    /exlibris/sfx_ver/sfx4_1/<instance>/templates/css/public/
    mobile/custom_rtl.css [Hebrew css]
    ```

- Mobile SFX menu template files:

  ```
  /exlibris/sfx_ver/sfx4_1/<instance>/templates/
  simplified_template1/mobile
  ```

- Mobile target displayers:

  ```
  /exlibris/sfx_ver/sfx4_1/<instance>/templates/
  simplified_template1/services/mobile/
  ```

- Mobile version of the DirectLink banner and multi-object window templates and images:
  - Images for multi-object window and DirectLink banner mobile version (including header image):

    `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/`

  - Header image name location and file name. (Note that this image is shared by the mobile versions of the CitationLinker, DirectLink banner, and multi-object window):

    ```
    /exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/
    logo_sfx.png
    ```

  Recommended image size: 213 pixels wide, 52 pixels high

- Templates for the mobile version of the SFX menu – multi-object menu and DirectLink banner:

  `/exlibris/sfx_ver/sfx4_1/<instance>/templates/multiobject/mobile`

- Templates and images for the CitationLinker mobile version:
  - Images for the CitationLinker mobile version (including header images):

    `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/`

  - Citationlinker title image located on the left side of the screen:

    `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/logo_citation_linker.png`

    Recommended image size: 238 pixels width, 52 pixels height

  - SFX logo image on the right side of the screen:

    `/exlibris/sfx_ver/sfx4_1/<instance>/templates/img/mobile/logo_sfx.png`

    Recommended image size: 213 pixels width, 52 pixels height

- Templates for the CitationLinker mobile version:

  `/exlibris/sfx_ver/sfx4_1/<instance>/templates/citation/mobile`

3

A-Z eJournal List and eBook Search

This section includes:

NOTES:

- For information on the deep link that enables you to link to the SFX A-Z List and eBook Search with two additional parameters, refer to the EL Commons Web site http://www.exlibrisgroup.org.
- For general information regarding the A-Z List and eBook Search, refer to the *SFX General User's Guide*.

Advanced Customization: HTML Template Files and Header Files

The A-Z eJournal List and the eBook Search interfaces share the same template files.

It is possible to change the display of the A-Z List and eBook Search by making changes to the following template files in the `/templates/azlist_ver3` directory:

- `templates/azlist_ver3/head.tmpl`
- `templates/azlist_ver3/browse_detail.tmpl`

- `templates/azlist_ver3/table_view.tmpl`
- `templates/azlist_ver3/detail_view.tmpl`
- `templates/azlist_ver3/set_detail.tmpl`
- `templates/azlist_ver3/openurl.tmpl`
- `templates/azlist_ver3/openurl_filtered.tmpl`
- `templates/azlist_ver3/title_head.tmpl`
- `templates/azlist_ver3/navigation_bar.css.tmpl`
- `templates/azlist_ver3/info_button.tmpl`
- `templates/azlist_ver3/navigation_go_button.tmpl`

It is possible to change the display of the A-Z List by making changes to the following template files in the `/templates/azlist_ver3` directory.

```
templates/azlist_ver3/category.tmpl
templates/azlist_ver3/locate_browse.tmpl
templates/azlist_ver3/locate_set.tmpl
templates/azlist_ver3/title_browse.tmpl
templates/azlist_ver3/title_set.tmpl
```

It is possible to change the display of the mobile A-Z List and eBook Search by making changes to the following template files in the `/templates/azlist_ver3` directory:

- `az_mobile.tmpl`
- `mobile_info_button.tmpl`

NOTE:

When customizing your A-Z List templates, it is recommended that you not remove placeholders for the textual elements in the SFX A-Z List that are controlled by the SFX Admin Center Translations & Display section. Removing the placeholders results in the textual element not being displayed.

For example:

```
<TMPL_VAR NAME='MESSAGE("AZ Main","Result-
>DisplayingRelatedJournal")' ESCAPE=HTML>
```

Where to Customize and What to Customize

- The Search form features and settings, as well as the data elements included among the search results, are all configured in:
    - For the A-Z eJournal list – **SFX Admin Center > Setup & Administration > Configuration > A-Z List**
    - For the eBook Search – **SFX Admin Center > Setup & Administration > Configuration >eBook Search**

- All interface text is defined in the **SFX Admin Center > Setup & Administration > Configuration > Translations & Display > AZ Main** tab.
- Images displayed in the SFX A-Z List are, by default, saved in the `templates/img/azlist_ver3` directory.

NOTE:

The top banner image is called `a-z_banner.gif` in the `.tmpl` files, but can be found in the `img/azlist_ver3` directory as `header.gif`.

- The headers in the A-Z title tab are defined in the `/config/headers` directory. For each header set, a configuration file exists that defines each header and the characters that are filed under that header.
- Other elements are controlled by the template (`.tmpl`) files. The files identified in the diagrams below are in the `templates/azlist_ver3` directory.
- The banner image in the A-Z and eBook search mobile interface is listed in the `templates/azlist_ver3/az_mobile.tmpl` file. Replace the image `img/azlist_ver3/logo_sfx.png`. The recommended image size is 213 pixels (width) and 52 pixels (height).

NOTE:

The location and names of images should not be changed. Rather, the original images should be replaced with your local images. This eliminates the need to make changes to any corresponding `.tmpl` files.

Title Search ("Starts with"): Detail View

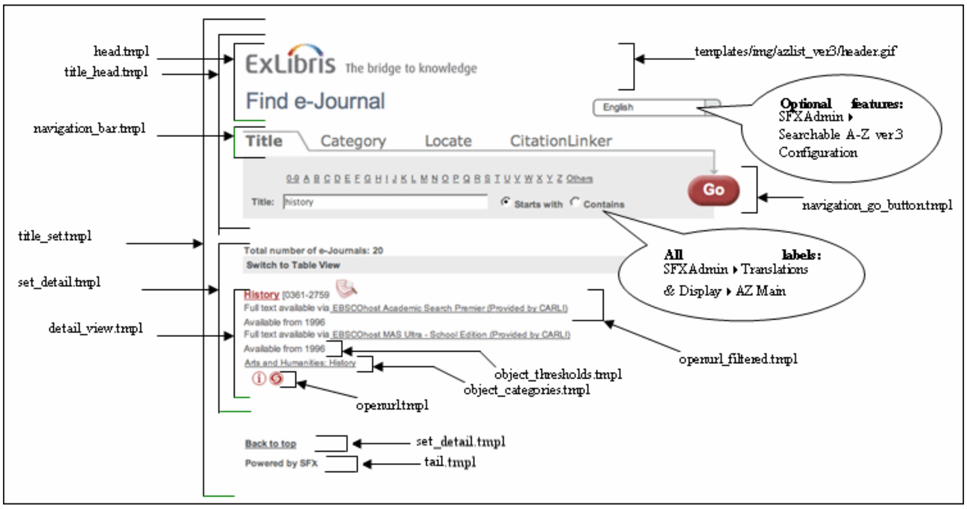

head.tmpl
title_head.tmpl
navigation_bar.tmpl
title_set.tmpl
set_detail.tmpl
detail_view.tmpl
ExLibris The bridge to knowledge
Find e-Journal
English
Title
Category
Locate
CitationLinker
0-9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Others
Title: history
Starts with
Contains
Go
Total number of e-Journals: 20
Switch to Table View
History [0361-2759
Full text available via EBSCOhost Academic Search Premier (Provided by CARLI)
Available from 1996
Full text available via EBSCOhost MAS Ultra - School Edition (Provided by CARLI)
Available from 1996
Arts and Humanities: History
Back to top
Powered by SFX
templates/img/azlist_ver3/header.gif
Optional features: SFXAdmin ▸ Searchable A-Z ver3 Configuration
navigation_go_button.tmpl
All labels: SFXAdmin ▸ Translations & Display ▸ AZ Main
openurl_filtered.tmpl
object_thresholds.tmpl
object_categories.tmpl
openurl.tmpl
set_detail.tmpl
tail.tmpl

Figure 15: Title Search: Detail View

Title Browse ("Starts with"): Detail View

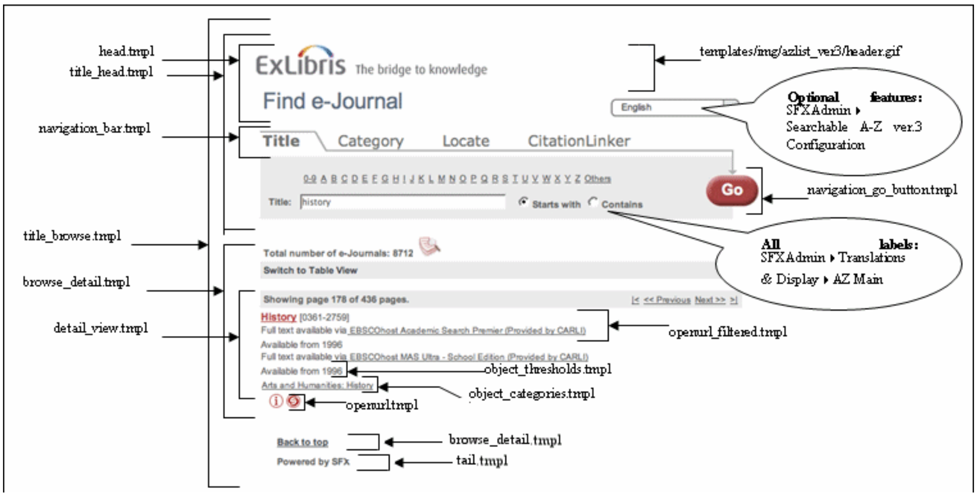

head.tmpl
title_head.tmpl
navigation_bar.tmpl
title_browse.tmpl
browse_detail.tmpl
detail_view.tmpl
ExLibris The bridge to knowledge
Find e-Journal
English
Title
Category
Locate
CitationLinker
0-9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Others
Title: history
Starts with
Contains
Go
templates/img/azlist_ver3/header.gif
Optional features: SFXAdmin ▸ Searchable A-Z ver.3 Configuration
navigation_go_button.tmpl
All labels: SFXAdmin ▸ Translations & Display ▸ AZ Main
Total number of e-Journals: 8712
Switch to Table View
Showing page 178 of 436 pages.
|< << Previous Next >> >|
History [0361-2759]
Full text available via EBSCOhost Academic Search Premier (Provided by CARLI)
Available from 1996
Full text available via EBSCOhost MAS Ultra - School Edition (Provided by CARLI)
Available from 1996
Arts and Humanities: History
openurl_filtered.tmpl
object_thresholds.tmpl
object_categories.tmpl
openurl.tmpl
Back to top
Powered by SFX
browse_detail.tmpl
tail.tmpl

Figure 16: Title Browse: Detail View

Title Search ("Starts with"): Table View

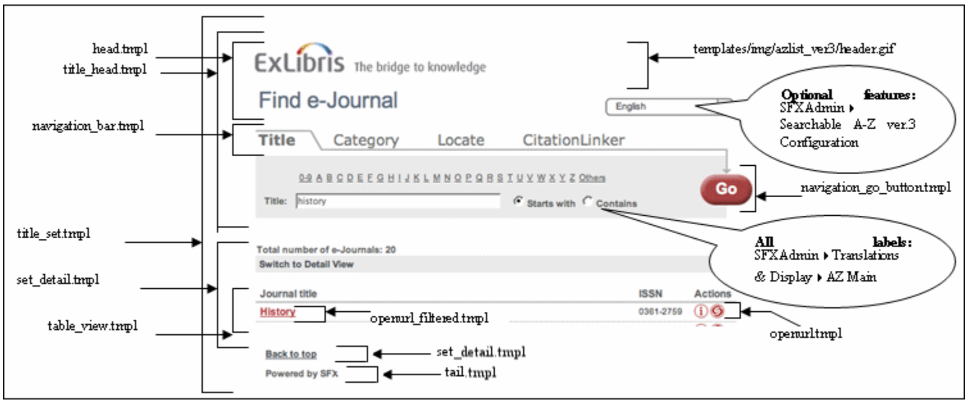


Figure 17: Title Search: Table View

Title Browse ("Starts with"): Table View

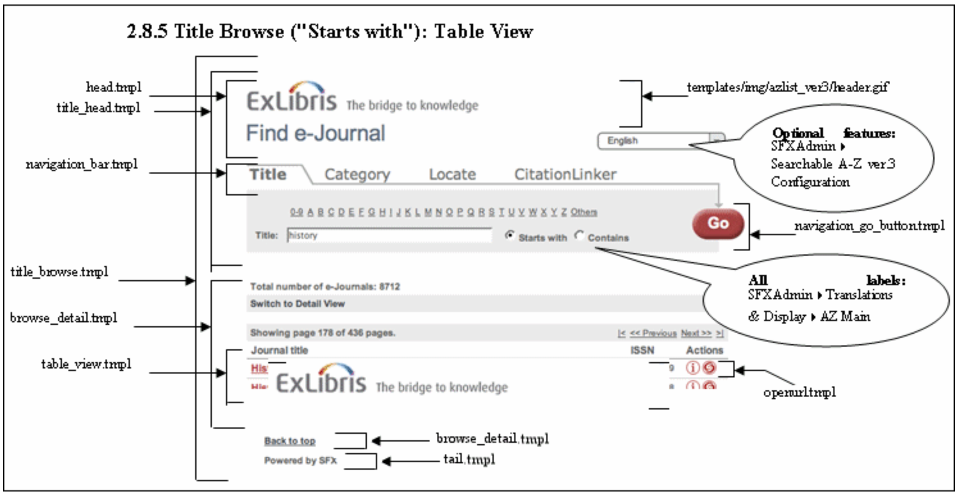

2.8.5 Title Browse ("Starts with"): Table View
head.tmpl
title_head.tmpl
navigation_bar.tmpl
title_browse.tmpl
browse_detail.tmpl
table_view.tmpl
templates/img/azlist_ver3/header.gif
Optional features: SFXAdmin ▸ Searchable A-Z ver3 Configuration
navigation_go_button.tmpl
All labels: SFXAdmin ▸ Translations & Display ▸ AZ Main
openurl.tmpl
browse_detail.tmpl
tail.tmpl
ExLibris The bridge to knowledge
Find e-Journal
English
Title
Category
Locate
CitationLinker
0-9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Others
Title: history
Starts with
Contains
Go
Total number of e-Journals: 8712
Switch to Detail View
Showing page 178 of 436 pages.
|< << Previous Next >> >|
Journal title
ISSN
Actions
Back to top
Powered by SFX

Figure 18: Title Browse: Table View

Category: Detail View

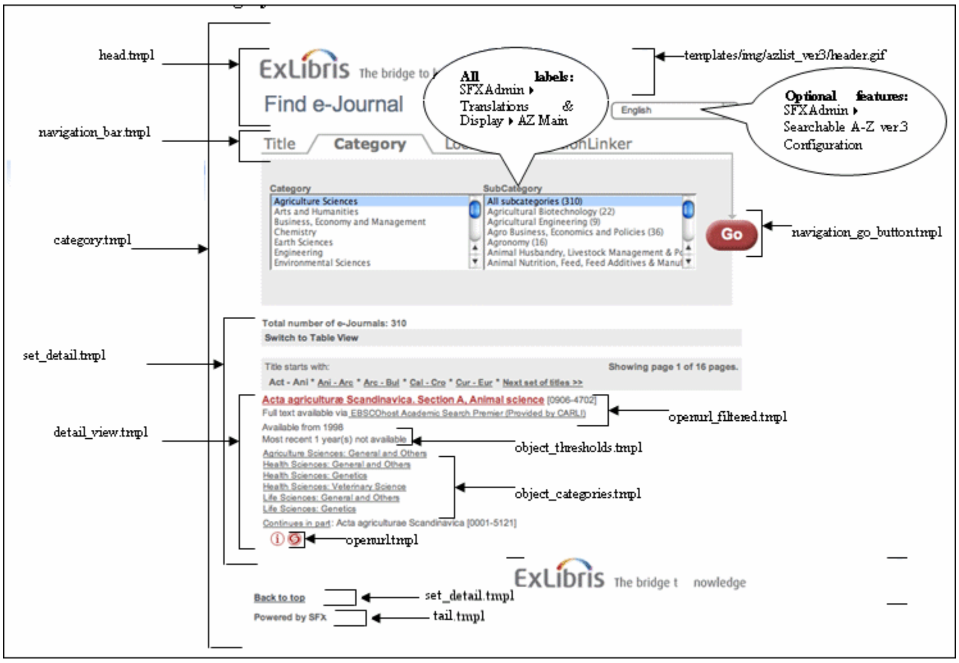
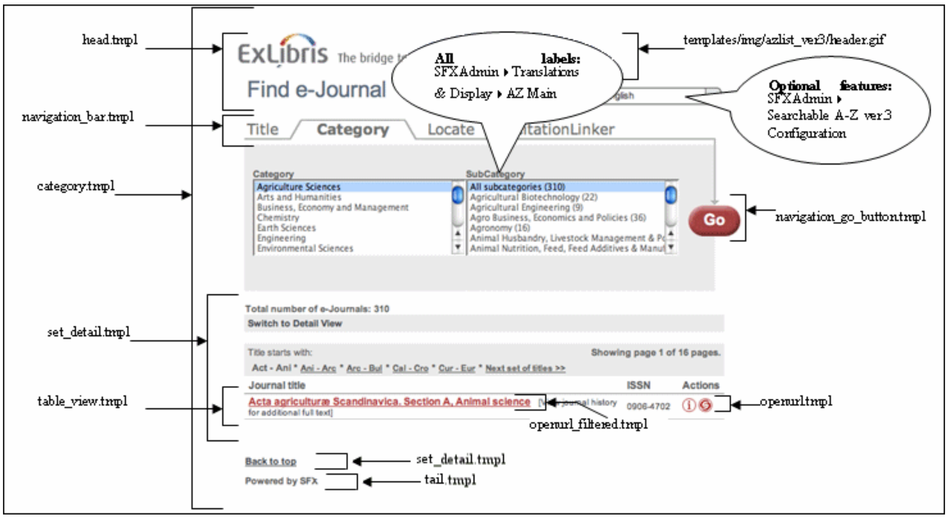

head.tmpl
navigation_bar.tmpl
category.tmpl
set_detail.tmpl
detail_view.tmpl
templates/img/azlist_ver3/header.gif
All labels: SFXAdmin ▸ Translations & Display ▸ AZ Main
Optional features: SFXAdmin ▸ Searchable A-Z ver3 Configuration
navigation_go_button.tmpl
openurl_filtered.tmpl
object_thresholds.tmpl
object_categories.tmpl
openurl.tmpl
set_detail.tmpl
tail.tmpl
ExLibris The bridge to knowledge
Find e-Journal
English
Title
Category
Locate
CitationLinker
Category
Agriculture Sciences
Arts and Humanities
Business, Economy and Management
Chemistry
Earth Sciences
Engineering
Environmental Sciences
SubCategory
All subcategories (310)
Agricultural Biotechnology (22)
Agricultural Engineering (9)
Agro Business, Economics and Policies (36)
Agronomy (16)
Animal Husbandry, Livestock Management & P
Animal Nutrition, Feed, Feed Additives & Manu
Go
Total number of e-Journals: 310
Switch to Table View
Title starts with:
Showing page 1 of 16 pages.
Act - Ani * Ani - Arc * Arc - Bul * Cal - Cro * Cur - Eur * Next set of titles >>
Acta agriculturæ Scandinavica. Section A, Animal science [0906-4702]
Full text available via EBSCOhost Academic Search Premier (Provided by CARLI)
Available from 1998
Most recent 1 year(s) not available
Agriculture Sciences: General and Others
Health Sciences: General and Others
Health Sciences: Genetics
Health Sciences: Veterinary Science
Life Sciences: General and Others
Life Sciences: Genetics
Continues in part: Acta agriculturae Scandinavica [0001-5121]
Back to top
Powered by SFX

Figure 19: Category: Detail View

Category: Table View



Category: Agriculture Sciences, Arts and Humanities, Business, Economy and Management, Chemistry, Earth Sciences, Engineering, Environmental Sciences

SubCategory: All subcategories (310), Agricultural Biotechnology (22), Agricultural Engineering (9), Agro Business, Economics and Policies (36), Agronomy (16), Animal Husbandry, Livestock Management & Po, Animal Nutrition, Feed, Feed Additives & Manuf

Total number of e-Journals: 310

Switch to Detail View

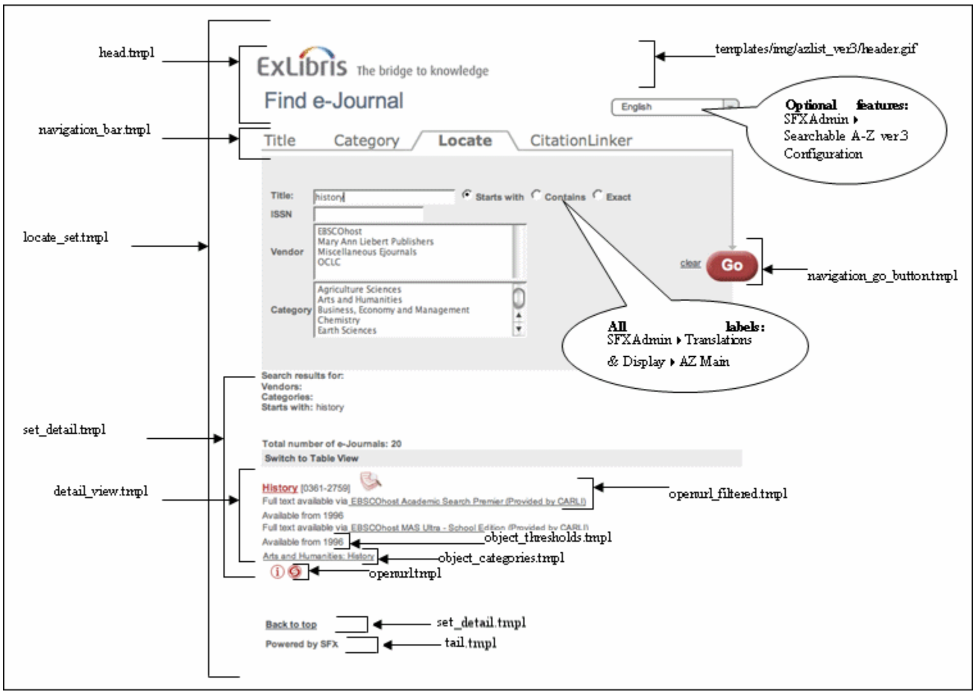
Title starts with: Showing page 1 of 16 pages.

Act - Ani * Ani - Arc * Arc - Bul * Cal - Cro * Cur - Eur * Next set of titles >>

Journal title	ISSN	Actions
Acta agriculturæ Scandinavica. Section A, Animal science [View journal history for additional full text]	0906-4702	ⓘ

Figure 20: Category: Table View

Locate Search ("Starts with"): Detail View



Title: history ● Starts with ○ Contains ○ Exact

ISSN

Vendor: EBSCOhost, Mary Ann Liebert Publishers, Miscellaneous Ejournals, OCLC

Category: Agriculture Sciences, Arts and Humanities, Business, Economy and Management, Chemistry, Earth Sciences

Search results for:
Vendors:
Categories:
Starts with: history

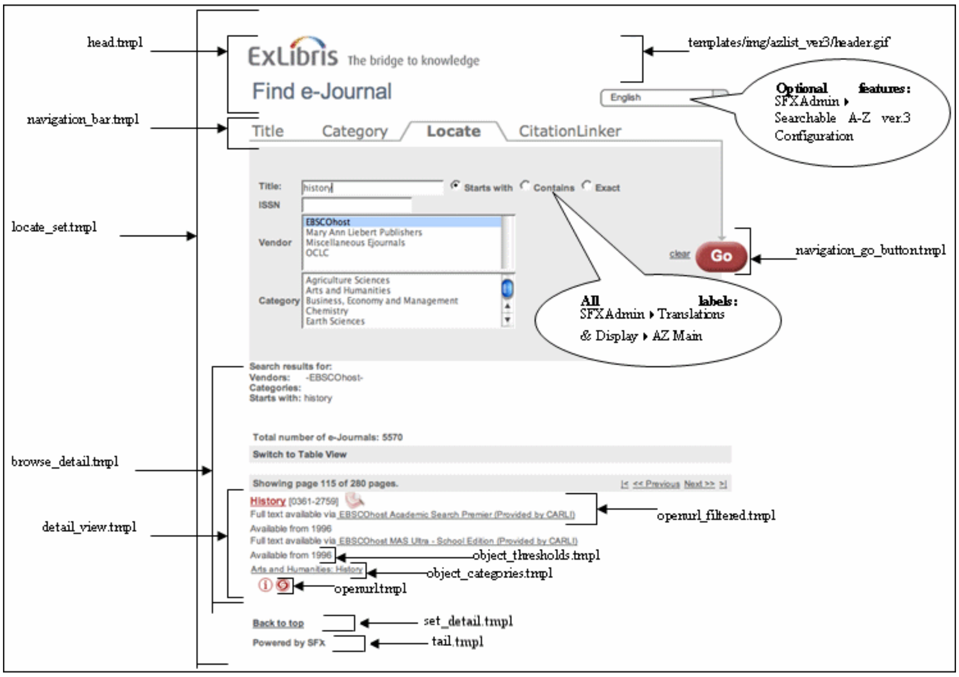
Total number of e-Journals: 20

Switch to Table View

History [0361-2759]
Full text available via EBSCOhost Academic Search Premier (Provided by CARLI)
Available from 1996
Full text available via EBSCOhost MAS Ultra - School Edition (Provided by CARLI)
Available from 1996
Arts and Humanities: History

Figure 21: Locate Search: Detail View

Locate Browse ("Starts with"): Detail View



ExLibris The bridge to knowledge

Find e-Journal

English

Title Category Locate CitationLinker

Title: history ● Starts with ○ Contains ○ Exact

ISSN

Vendor: EBSCOhost, Mary Ann Liebert Publishers, Miscellaneous Ejournals, OCLC

Category: Agriculture Sciences, Arts and Humanities, Business, Economy and Management, Chemistry, Earth Sciences

clear Go

Search results for:
Vendors: -EBSCOhost-
Categories:
Starts with: history

Total number of e-Journals: 5570

Switch to Table View

Showing page 115 of 280 pages. |< << Previous Next >> >|

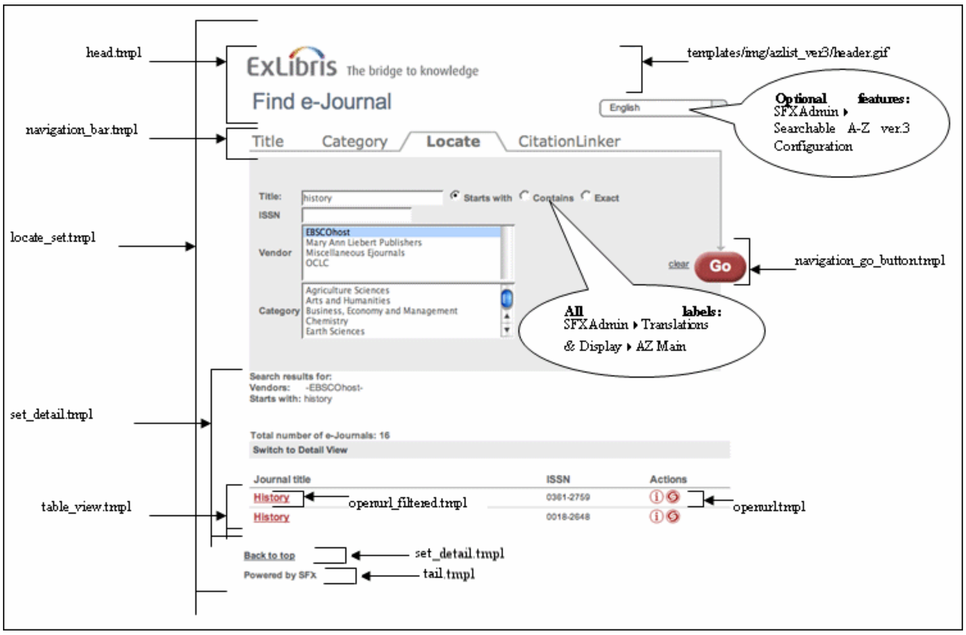
History [0361-2759]
Full text available via EBSCOhost Academic Search Premier (Provided by CARLI)
Available from 1996
Full text available via EBSCOhost MAS Ultra - School Edition (Provided by CARLI)
Available from 1996
Arts and Humanities: History

Back to top

Powered by SFX

Figure 22: Locate Browse: Detail View

Locate Search ("Starts with"): Table View



ExLibris The bridge to knowledge

Find e-Journal

English

Title Category Locate CitationLinker

Title: history ● Starts with ○ Contains ○ Exact

ISSN

Vendor: EBSCOhost, Mary Ann Liebert Publishers, Miscellaneous Ejournals, OCLC

Category: Agriculture Sciences, Arts and Humanities, Business, Economy and Management, Chemistry, Earth Sciences

clear Go

Search results for:
Vendors: -EBSCOhost-
Starts with: history

Total number of e-Journals: 16

Switch to Detail View

Journal title	ISSN	Actions
History	0361-2759	ⓘ Ⓢ
History	0018-2648	ⓘ Ⓢ

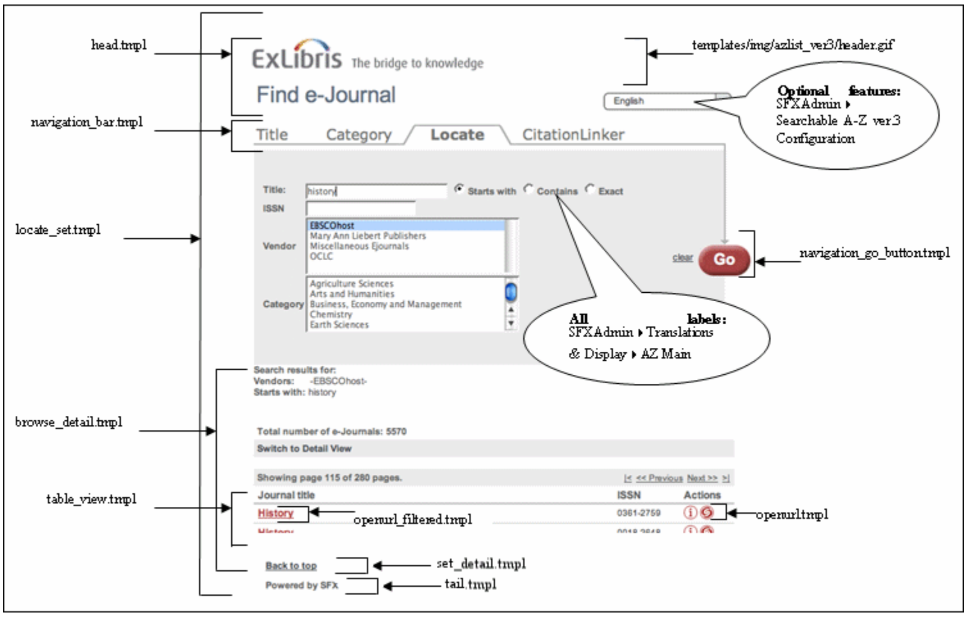
Back to top

Powered by SFX

head.tmpl
navigation_bar.tmpl
locate_set.tmpl
set_detail.tmpl
table_view.tmpl
templates/img/azlist_ver3/header.gif
Optional features: SFXAdmin ▸ Searchable A-Z ver.3 Configuration
navigation_go_buttontmpl
All labels: SFXAdmin ▸ Translations & Display ▸ AZ Main
openurl_filtered.tmpl
openurltmpl
set_detail.tmpl
tail.tmpl

Figure 23: Locate Search: Table View

Locate Browse ("Starts with"): Table View



Title: history ● Starts with ○ Contains ○ Exact

Vendor: EBSCOhost, Mary Ann Liebert Publishers, Miscellaneous Ejournals, OCLC

Category: Agriculture Sciences, Arts and Humanities, Business, Economy and Management, Chemistry, Earth Sciences

Search results for:
Vendors: -EBSCOhost-
Starts with: history

Total number of e-Journals: 5570

Switch to Detail View

Showing page 115 of 280 pages. |< << Previous Next >> >|

Journal title	ISSN	Actions
History	0361-2759	

Figure 24: Locate Browse: Table View

CitationLinker

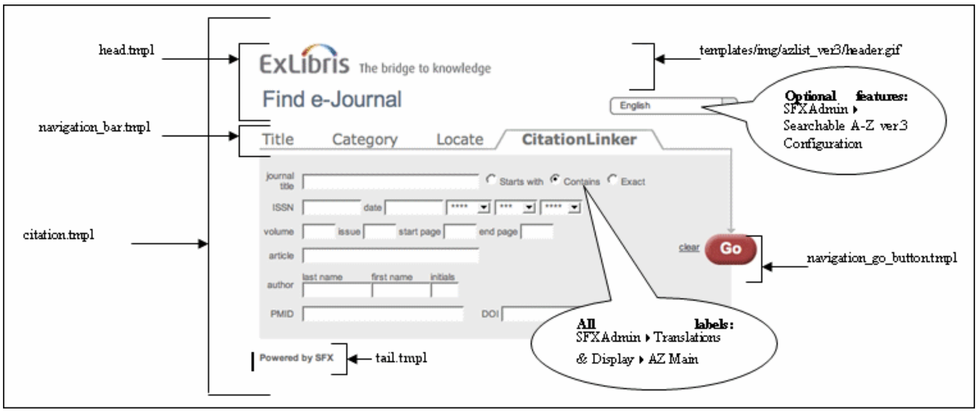
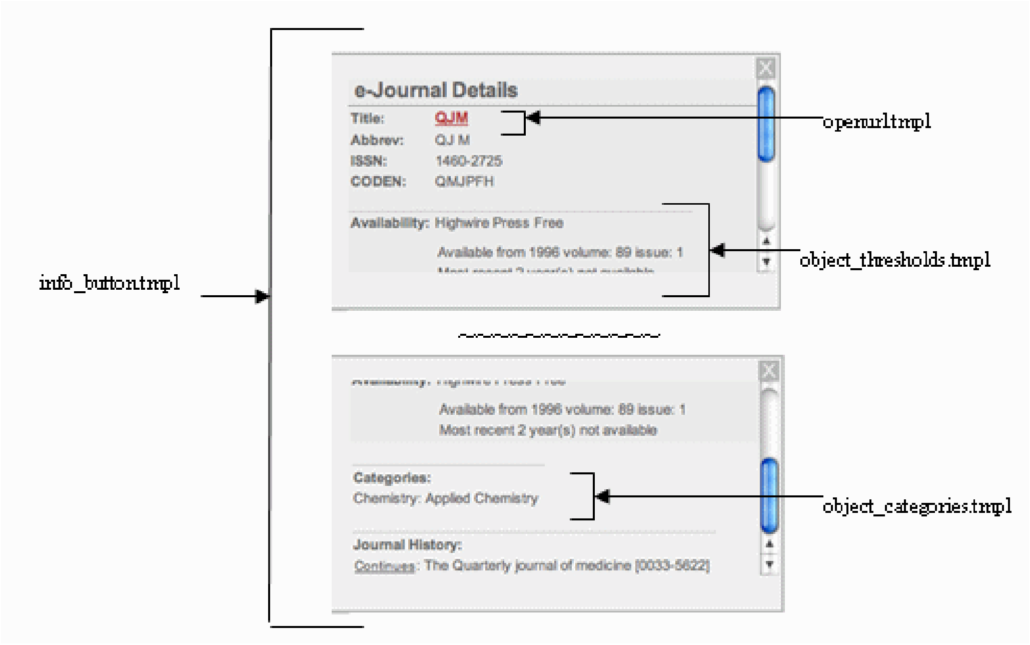


Figure 25: CitationLinker

Info Window



e-Journal Details

Title: QJM
Abbrev: QJ M
ISSN: 1460-2725
CODEN: QMJPFH

Availability: Highwire Press Free
Available from 1996 volume: 89 issue: 1
Most recent 2 year(s) not available

Categories:
Chemistry: Applied Chemistry

Journal History:
Continues: The Quarterly journal of medicine [0033-5622]

Figure 26: Info Window

A-Z List for Mobile Devices

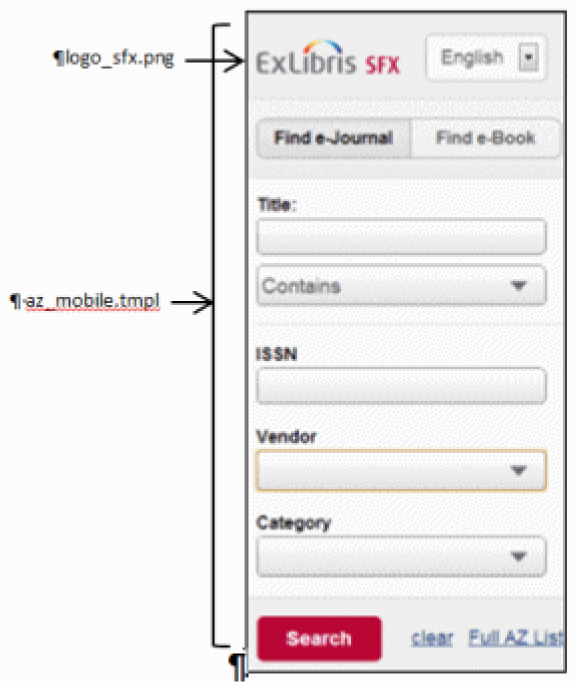


Figure 27: A-Z List for Mobile Devices

A-Z List and Institutes Using the Consortium API

For the A-Z list, no special configuration is needed in the `config/consortium.config` file. However, since the Rapid Service Indicator (RSI) index is used to build the A-Z list, you need to verify that RSI has been configured correctly in `config/consortium.config`, and that the RSI index has been created, first in the shared instance, and then in the local instances. Make sure that all of the institutes and groups for which you want to create A-Z lists that include services from the shared instance are defined in the local instance (including the group assignments for each institute).

For additional information concerning RSI and A-Z List setup, refer to the *Using SFX in a Consortium Environment* document.

For more information on building the RSI index, refer to the **Rapid Service Indicator** section of the *SFX System Administration Guide*.

A-Z List and CJK Support

This section includes the following information regarding CJK language support for the A-Z List:

- Simplified Chinese and Traditional Chinese Titles
- Mainland China
- Taiwan
- Korea
- Japan
- Limitations in the Current CJK A-Z List Implementation

Simplified Chinese and Traditional Chinese Titles

Chinese objects in the SFX KnowledgeBase are assigned the language value `chi`. They will have, if required, both a Simplified and Traditional Chinese title. For example:

```
Title - MAIN - Simplified
                                阿坝科技

Title - MAIN - Traditional
                                阿壩科技

Title - ROMANIZATION - Pinyin   a ba ke ji
```

Object ID	1000000000256408
Object type	JOURNAL
Language	Chinese
Peer reviewed	NO

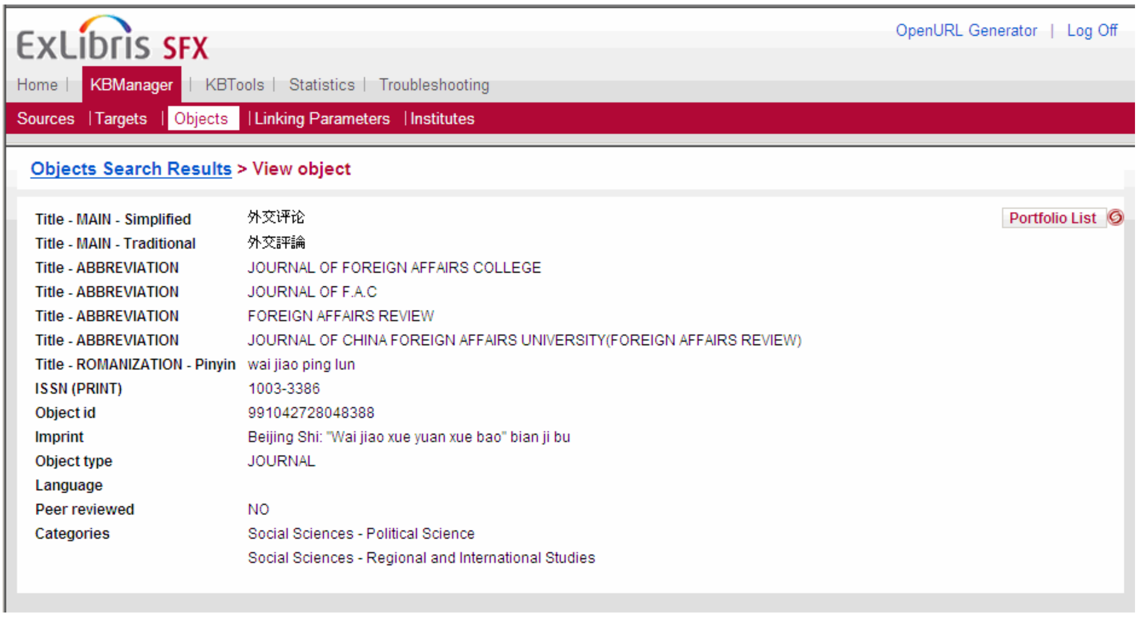
In the SFX menu and A-Z List, only the appropriate Chinese title is displayed, depending on the country settings in the default A-Z profile.

If **Mainland China** is selected as the country in the default A-Z profile, Simplified Chinese titles are displayed.

If **Taiwan** is selected as the country, the Traditional Chinese titles are displayed.

If the SFX KB contains only a Simplified or Traditional Chinese title, this title is displayed for both Mainland China and Taiwan country settings.

In KBManager, both Chinese titles (if they exist) will be displayed in the results display of the Search Object search form.


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | KBTools | Statistics | Troubleshooting
Sources | Targets | Objects | Linking Parameters | Institutes
Portfolio List

Objects Search Results > View object

Title - MAIN - Simplified	外交评论
Title - MAIN - Traditional	外交評論
Title - ABBREVIATION	JOURNAL OF FOREIGN AFFAIRS COLLEGE
Title - ABBREVIATION	JOURNAL OF F.A.C
Title - ABBREVIATION	FOREIGN AFFAIRS REVIEW
Title - ABBREVIATION	JOURNAL OF CHINA FOREIGN AFFAIRS UNIVERSITY(FOREIGN AFFAIRS REVIEW)
Title - ROMANIZATION - Pinyin	wai jiao ping lun
ISSN (PRINT)	1003-3386
Object id	991042728048388
Imprint	Beijing Shi: "Wai jiao xue yuan xue bao" bian ji bu
Object type	JOURNAL
Language	
Peer reviewed	NO
Categories	Social Sciences - Political Science Social Sciences - Regional and International Studies

Figure 28: Chinese Object

A search of Chinese titles can be performed on both the Simplified and Traditional titles. When searching the complete Simplified or Traditional title, no truncation should be used.

Mainland China

The following items are relevant for mainland China:

- **A-Z Searching** – When the country chosen in the default A-Z profile is **China** and the A-Z List contains both Latin and Chinese journal titles, it is recommended to display both Latin and Pinyin headers in the A-Z List.

  When both headers display, an additional **Pinyin** check box displays. Every time a Pinyin search term is typed, the **Pinyin** check box should be selected to indicate that Pinyin is used. This is required to distinguish between Latin and Chinese titles. The result set corresponds to the check box choice and contains either Pinyin or Latin titles.

  The default state of the Pinyin check box (selected or cleared) can be defined in the A-Z Configuration section of the SFX Admin Center.

  A search of Chinese titles can be performed on both the Simplified and Traditional titles.
- **Display of Search Results** – When a Chinese search term is entered—in either Simplified or Traditional Chinese—the results are displayed in Simplified Chinese if the Simplified Chinese title exists in the SFX KnowledgeBase.
- **Sorting** – When the country chosen is **China**, the Chinese titles in the A-Z List are sorted according to Pinyin.

Taiwan

The following items are relevant for Taiwan:

- **Display of Search Results** – When **Taiwan** is the country chosen and a Chinese search term is entered in either Simplified or Traditional Chinese, the results are displayed in Traditional Chinese if the Traditional Chinese title exists in the SFX KnowledgeBase.
- **Sorting** – Chinese titles in the A-Z List are sorted by stroke sequence.

Korea

The following items are relevant for Korea:

- **Searching** – When the country chosen is **Korea** and the A-Z List contains Chinese titles, the **Limit search to Chinese titles** check box needs to be selected when searching Chinese titles. This is needed to be able to distinguish between Chinese journals and Korean journals with Chinese characters in the title (since the sorting and search normalizations for both types of objects work differently).
- **Sorting** – When the country chosen is **Korea** and a search term is entered in either Hangul or Hanja, the results of the Korean titles are sorted by Hangul.

Japan

- **Display of Search Results** – When the country chosen is **Japan** and the A-Z List contains Japanese as well as Latin titles, it is recommended that both Latin and the two Hiragana headers in the SFX A-Z List be displayed.
- **Searching** – It is possible to use Hiragana, Katakana, Kanji, and Latin scripts in a search.

  The title search term is used without transliteration being performed (no Katakana-to-Hiragana or Kanji-Hiragana conversions).
- **Sorting** – The sorting of search results, as well as the grouping into headers for Japanese titles, is performed using normalized Katakana/Hiragana titles.

  The main title in the SFX KnowledgeBase is used for sorting and can contain either Hiragana or Katakana characters. The Katakana characters are converted to Hiragana during the A-Z index build.

Limitations in the Current CJK A-Z List Implementation

The following limitations exist in the current CJK A-Z List implementation:

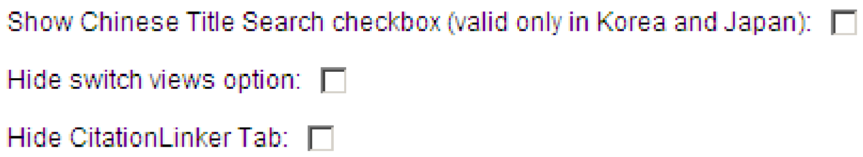
- Journals in the SFX KnowledgeBase that are missing required title information for display or sorting are not included in the A-Z List.
- For CJK journal objects, the A-Z List does not currently support searching based on non-CJK titles (such as abbreviated titles). Searching can only be done on CJK titles in the SFX KnowledgeBase.

Removing the CitationLinker Tab from the A-Z List

By default, the CitationLinker tab is displayed in the A-Z List.

To hide the CitationLinker tab from the A-Z List users:

1 In the **SFX Admin Center > Setup & Administration > Configuration > A-Z List > DISPLAY Section**, select the **Hide CitationLinker Tab** check box.



Show Chinese Title Search checkbox (valid only in Korea and Japan): □

Hide switch views option: □

Hide CitationLinker Tab: □

Figure 29: Hide CitationLinker Tab

When this option is selected, the A-Z List displays as follows:

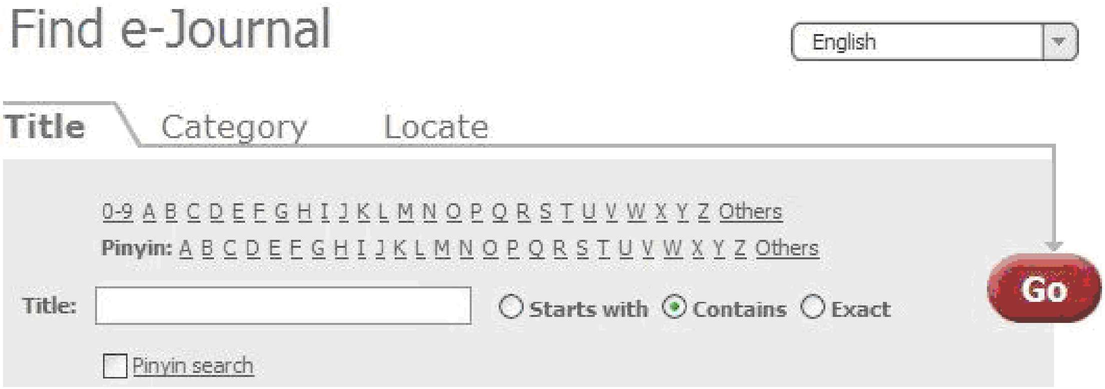

Find e-Journal
English
Title
Category
Locate
0-9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Others
Pinyin: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Others
Title:
○ Starts with ● Contains ○ Exact
Go
□ Pinyin search

Figure 30: A-Z List

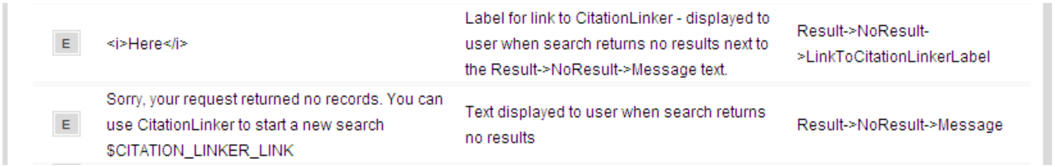
2 Change the wording of the text displayed to the user when a search returns no results. By default, the text provides a link to the CitationLinker tab with the message:

Sorry, your request returned no records. You can use CitationLinker to start a new search here.

a Go to the **SFX Admin Center > Setup & Administration > Configuration > Translations & Display > AZ Main** tab.

b For each language used in the A-Z List, edit the following elements:

- Remove the link from the Result->NoResult->Message element.
- Delete the contents of the Result->NoResult->LinkToCitationLinkerLabel.



E	<i>Here</i>	Label for link to CitationLinker - displayed to user when search returns no results next to the Result->NoResult->Message text.	Result->NoResult->LinkToCitationLinkerLabel
E	Sorry, your request returned no records. You can use CitationLinker to start a new search $CITATION_LINKER_LINK	Text displayed to user when search returns no results	Result->NoResult->Message

Figure 31: CitationLinker Label

c Remove the link from the message text and change it to the following:

Sorry, your request returned no records.

A-Z List Browser Toolbar

The SFX A-Z List browser toolbar is available as an add-on component for Microsoft Internet Explorer that allows end users to quickly search the SFX A-Z List from anywhere on the Web.

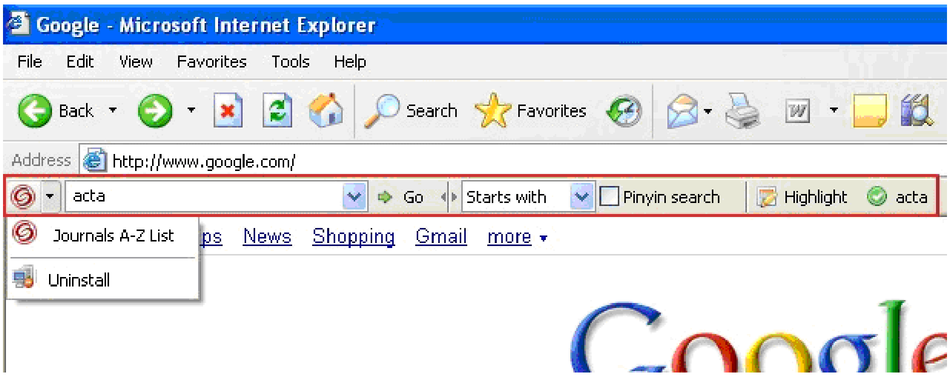


Figure 32: SFX A-Z List Toolbar

The toolbar is supported by Microsoft Internet Explorer versions 5.5, 6, 7, and 8, but not later versions.

After installing the A-Z List browser toolbar, the following options are available:

- A search box to perform an A-Z List title search using one of the following search types:
  - Starts with
  - Contains
  - Exact
- The Pinyin/Chinese check box (presented only if needed)
- A link to the library home page and/or the A-Z List of the library by clicking the SFX logo

All links displaying in this drop-down list can be configured in **SFX Admin Center > Setup & Administration > Configuration > A-Z List**.

Implementing the A-Z List Browser Toolbar

You enable the toolbar in a configuration file and then configure it in the SFX Admin Center.

To enable the toolbar:

1. Connect to the SFX server and log on as the instance user.
2. Enter `cn`.
3. Enter `vi ctx_object.config`.

4 At the end of the file, add the following lines:

```
Section "azlist"
    enable_toolbar   "Y"
EndSection
```

5 Enter `wq` to save the file and exit.

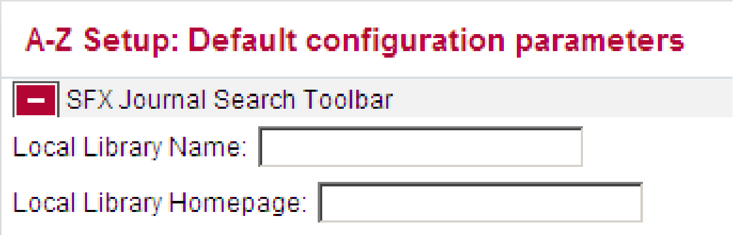
To configure the toolbar in the SFX Admin Center:

1 Go to the **SFX Admin Center** > **Setup & Administration** > **Configuration** > **A-Z List** section of the instance.

2 Define the links in the toolbar. This can be done in the default profile or any other A-Z List profile.

3 Configure the following parameters for the SFX Journal Search Toolbar:

- **Local Library Name** – Enter the link name.
- **Local Library Homepage** – Enter the link URL.



A-Z Setup: Default configuration parameters

SFX Journal Search Toolbar

Local Library Name:

Local Library Homepage:

Figure 33: Default Configuration Parameters

4 Save your changes.

NOTE:

The toolbar configuration is saved in an XML form on the SFX server. This configuration is checked for updates whenever the browser toolbar is used.

For performance reasons, this configuration is held in the client-side cache and checked once a day. This means that any configuration changes made in the SFX Admin Center only take effect a day later in the end user's browser.

The toolbar content XML can be viewed at:
`<SFX_AZ_URL>?param_perform_value=toolbar&action=configure`

Installing the A-Z List Browser Toolbar

When you want to release the toolbar to end users, specify the following link to the toolbar installer:

```
<SFX_AZ_URL>?param_perform_value=toolbar&action=install
```

This can be added to the library Web page on which you want to provide instructions to your users on how to install the A-Z List browser toolbar.

For example:

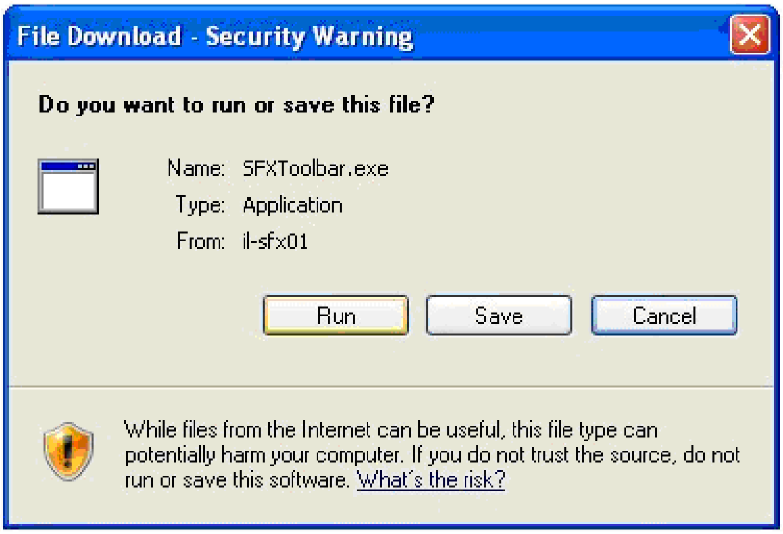
http://demo.exlibrisgroup.com:3210/demo/az?param_perform_value=toolbar&action=install

The following is an example of an HTML page with this link:

```
<html>
<head>
<title>Test A-Z toolbar</title></head>
<body>
To install the A-Z toolbar, please click <a href='http://il-
sfx01:3210/lr_3/
az?param_perform_value=toolbar&action=install'>here</a>
</body>
</html>
```

To install the toolbar:

1 Click the link to display the following dialog box:



File Download - Security Warning

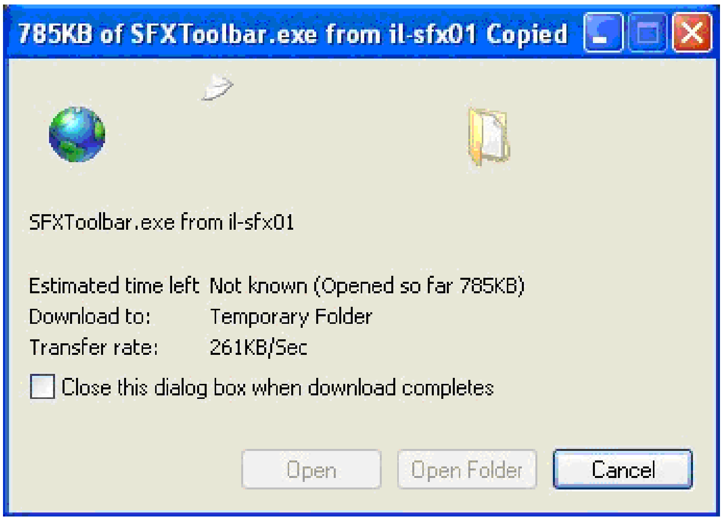
Do you want to run or save this file?

Name: SFXToolbar.exe
Type: Application
From: il-sfx01

While files from the Internet can be useful, this file type can potentially harm your computer. If you do not trust the source, do not run or save this software. What's the risk?

Figure 34: Run File

2 Click **Run** to run the file or **Save** to save the file and double-click it later at the saved location. The following dialog box displays:



785KB of SFXToolbar.exe from il-sfx01 Copied

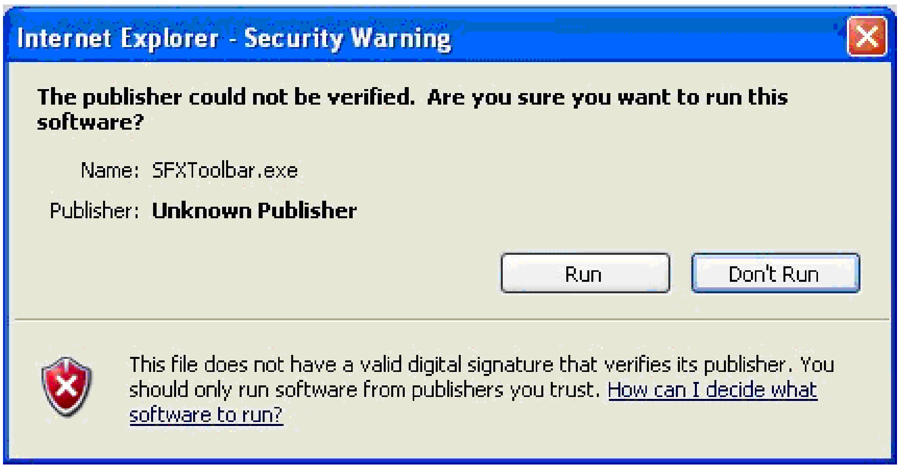
SFXToolbar.exe from il-sfx01

Estimated time left Not known (Opened so far 785KB)
Download to: Temporary Folder
Transfer rate: 261KB/Sec

□ Close this dialog box when download completes

Figure 35: Downloading File

3 When the following dialog box displays, click **Run**.



Internet Explorer - Security Warning

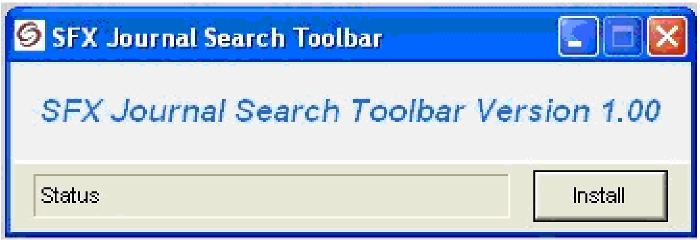
The publisher could not be verified. Are you sure you want to run this software?

Name: SFXToolbar.exe
Publisher: **Unknown Publisher**

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust. How can I decide what software to run?

Figure 36: Unknown Publisher

4 Click **Install** to start the installation process.

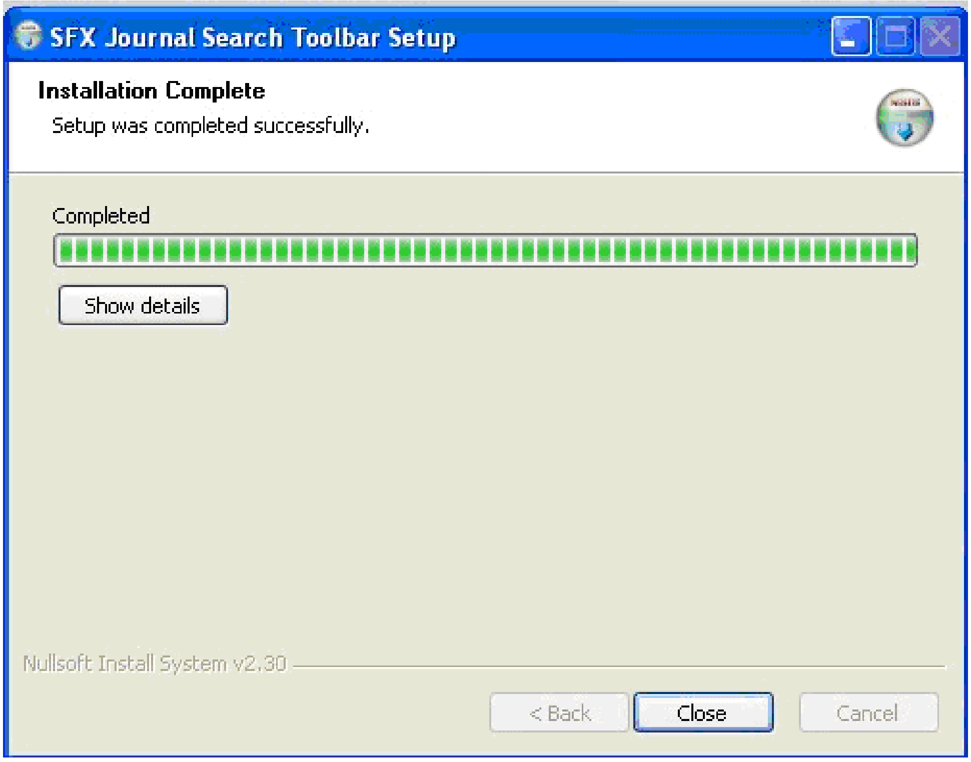


SFX Journal Search Toolbar

SFX Journal Search Toolbar Version 1.00

Status

Figure 37: Install


SFX Journal Search Toolbar Setup

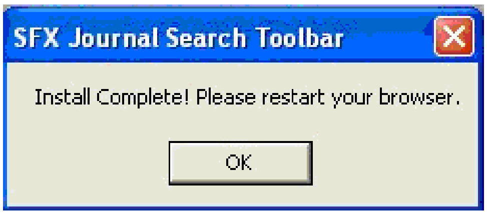
Installation Complete

Setup was completed successfully.

Completed

Figure 38: Installation Complete

5 Restart the browser when the following message displays:


SFX Journal Search Toolbar

Install Complete! Please restart your browser.

Figure 39: Install Complete

The A-Z toolbar displays as follows:

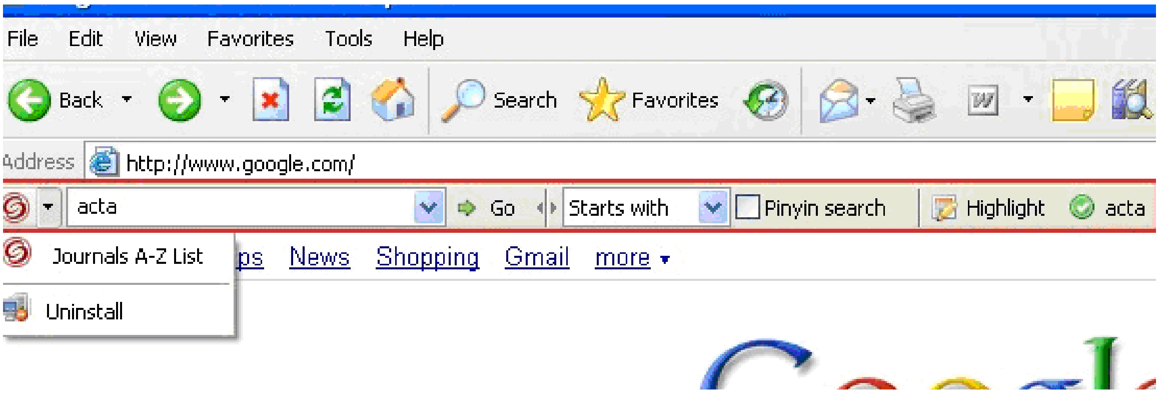

File Edit View Favorites Tools Help
Back
Search
Favorites
Address http://www.google.com/
acta
Go
Starts with
Pinyin search
Highlight
acta
Journals A-Z List
Uninstall
ps News Shopping Gmail more

Figure 40: A-Z Toolbar

Removing the A-Z List Browser Toolbar

You can uninstall the A-Z List Browser toolbar.

To uninstall the toolbar:

1. Click the SFX logo.
2. Select **Uninstall**.
3. Restart the browser to complete the toolbar uninstall process.

Automatically Updating the A-Z List Toolbar

The A-Z List browser toolbar contains an automatic update mechanism to make sure that when updates to the A-Z List toolbar are released by Ex Libris, toolbar users are alerted that updates are available and are able to install them.

Each toolbar installation contains the toolbar version number. It is compared with a file on the SFX server that contains the current toolbar version number. The file is saved in the following location on the SFX server:

```
./lib/AZLIST_VER3/Toolbar/version.txt
```

If updates are needed, this file is updated by the revision update process. Once the revision is applied on the SFX server, each locally installed toolbar detects the change and alerts the user.

The end user can also initiate a check for updates by using the following option directly from the browser toolbar.

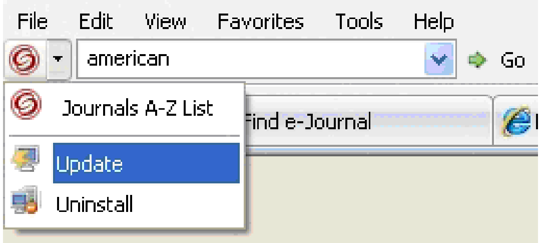


Figure 41: Update Option

Solr Index Configuration

In the global instance, the following configuration file determines the setup of the Solr index: `/exlibris/sfx_ver/sfx4_1/sfxglb41/config/solr_index.config_`

In this configuration file, all information is set automatically when creating the Solr core, including the URL, port, and encrypted password. Before starting to use the Solr service, set the `active` parameter to `Y` in the `solr_startup_params` section (the default is `N`):

```
Section "solr_startup_params"
        active  "N"
        (...)
Section
```

Set the `check_memory` parameter to `Y` to have a memory check performed before starting the Solr service to make sure there is enough memory for the service (default is `Y`). Set the parameter to `N` to skip the memory check.

```
Section "solr_startup_params"
        (...)
        check_memory "Y"
Section
```

For information concerning the server requirements for Solr, refer to **Appendix D eBook Search Set-Up Instructions** of the *SFX System Administration Guide*.

Additionally, it is possible to set the bulk_size parameter in the complete_build_params section to define the maximum amount of changes before a complete build is needed:

```
# Specify how many objects will be processed in each bulk
# Bigger bulks may be faster but will consume more memory
Section "build_params"
        bulk_size       1000
EndSection
```

If you use the eBook search, it is recommended that you configure a cron job to run a script that restarts Solr automatically if it is down. It is recommended to have the script run at least once an hour. Running the script and restarting Solr have no impact on SFX performance.

To set up the cron job:

1 Log on as sfxglb41.

2 Type crontab -e.

3 Add the following line after the existing tasks:

```
<scheduling time/date/frequency> tcsh -c 'source ~sfxglb41/
.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/init_scripts/
solr_daemon.pl'
```

4 Save and exit the file.

For example, the following line runs the script once an hour at the beginning of the hour:

```
0 * * * * tcsh -c 'source ~sfxglb41/.cshrc; /exlibris/sfx_ver/
sfx4_1/sfxglb41/admin/init_scripts/solr_daemon.pl'
```

For more information on cron jobs, including the structure of the scheduling time/date definition, see http://en.wikipedia.org/wiki/Cron.

4

CitationLinker

This section includes:

NOTES:

- For information on the deep link that enables you to link to the SFX CitationLinker form and prepopulate the form with metadata, refer to the EL Commons Web site http://www.exlibrisgroup.org.
- For general information on the CitationLinker, refer to the *SFX General User's Guide*.

Pointing to the CitationLinker

Point your users to the following for the default version of CitationLinker:

http://server_name:port/instance_name/cgi/core/citation-linker.cgi

This links your users to the article form that further links them to the journal and book forms.

Changing the Look of the Existing CitationLinker

You can change buttons, the `.gif` files, and the `.tmpl` files that make up the CitationLinker.

Elements in the CitationLinker are governed by the `style.css` file (also used with the SFX menu `.tmpl` files) located in:

```
/exlibris/sfx_ver/sfx4_1/<instance_name>/templates/css/public
```

To change the look of the CitationLinker:

1 Edit the `.tmpl` files on the SFX server as required.

NOTE:
Do not change the FORM and INPUT TYPE tags.

The `.tmpl` files are available at:

```
/exlibris/sfx_ver/sfx4_1/<instance_name>/templates/citation
```

The `.tmpl` files include:

- `Article.tmpl`
- `Book.tmpl`
- `head.tmpl`
- `Journal.tmpl`
- `tail.tmpl`

2 Create backups of the existing `.gif` files in:

```
/exlibris/sfx_ver/sfx4_1/<instance_name>/templates/img/
citation
```

- `mv ico_go.gif ico_go-bak.gif`
- `mv sfx.gif sfx-bak.gif`
- `mv title.gif title-bak.gif`

3 FTP any different `.gif` files that you want to use to:

```
/exlibris/sfx_ver/sfx4_1/cs_3/templates/img/citation
```

You need to use the names `ico_go.gif`, `sfx.gif`, and `title.gif` for your files.

Incorporating the CitationLinker in Existing Web Pages

You can incorporate the CitationLinker in existing Web pages.

To incorporate the CitationLinker in existing Web pages:

1 Copy the template files (`Article.tmpl`, `Book.tmpl`, `Journal.tmpl`, `head.tmpl`, and `tail.tmpl`) to your local Web server.

2 Place the contents of `head.tmpl` and `tail.tmpl` on the top and bottom, respectively, of each of the other pages and change the extensions from `.tmpl` to `.html`. You may delete `head.tmpl` and `tail.tmpl`. Make changes to the HTML as required.

IMPORTANT:
Do not edit the existing FORM and INPUT TYPE tags except as described below.

In each of the HTML files, you need to enter the hard-coded address of your SFX server, and possibly your library Web server, in the forms as described in the following steps.

NOTE:
In the examples, the library's Web server is http://library.university.edu and the library's SFX server is http://sfx.university.edu:9003/sfx_local.

3 Edit the Javascript references.

This Javascript causes a smaller window, containing the SFX menu, to open after the user submits the form.

Change the following:

```
  <script type="text/javascript" language="JavaScript" src="/
<TMPL_VAR INSTANCE>/js/common/main.js"></script>
    <script type="text/javascript" language="JavaScript"
src="/<TMPL_VAR INSTANCE>/js/common/date.js"></script>
    <script type="text/javascript" language="JavaScript"
src="/<TMPL_VAR INSTANCE>/js/citation/main.js"></script>
```

to:

```
  <script type="text/javascript" language="JavaScript"
src="http://sfx.university.edu:9003/sfx_local/js/common/
main.js"></script>
    <script type="text/javascript" language="JavaScript"
src="http://sfx.university.edu:9003/sfx_local/js/common/
date.js"></script>
    <script type="text/javascript" language="JavaScript"
src="http://sfx.university.edu:9003/sfx_local/js/citation/
main.js"></script>
```

Change the image references to reference your images. The images included in the files as delivered provide the top header `.gif` file and the GO! submit button.

4 Edit the form tag.

For each of the HTML files, the following tag should be changed to include the reference to the SFX `citation-linker.cgi` script.

Change the following:

```
<form name=OpenURL target="SFXmenu"
onSubmit="OpenWindow('','SFXmenu');">
```

to:

```
<form name=OpenURL target="SFXmenu" action="http://
server_address:port/instance-name/cgi/core/citation-
linker.cgi" onSubmit="OpenWindow('','SFXmenu');">
```

For example:

```
<form name=OpenURL target="SFXmenu" action="http://
sfx.university.edu:9003/sfx_local/cgi/core/citation-
linker.cgi" onSubmit="OpenWindow('','SFXmenu');">
```

5. Edit the form INPUT tag.

   The following tag:

```
<input type=hidden name="__base_url" value="/<TMPL_VAR
INSTANCE><TMPL_VAR __INST_URL_PART>">
```

should be changed to:

```
<input type=hidden name="__base_url" value="http://
server_address:port/instance-name">
```

For example:

```
<input type=hidden name="__base_url" value="http://
sfx.university.edu:9003/sfx_local">
```

6. Edit all `.html` files (`Article.html`, `Book.html`, and `Journal.html`) to change the references to the location of the files on your server, or use relative links. Each `.html` file has links to the other two `.html` files.

   For example, the `Article.html` file:

```
<tr>
 <td class="Block">&nbsp;&nbsp;<strong>Article</
strong>&nbsp;&nbsp;</td>
 <td>&nbsp;&nbsp;&nbsp;</td>
 <td class="GroupTitle">&nbsp;&nbsp;<a class="ABC"
href="?rft.genre=journal">Journal</a>&nbsp;&nbsp;</td>
 <td>&nbsp;&nbsp;&nbsp;</td>
 <td class="GroupTitle">&nbsp;&nbsp;<a class="ABC"
href="?rft.genre=book">Book</a>&nbsp;&nbsp;</td>
</tr>
```

should be changed to:

```
<tr>
 <td class="Block">&nbsp;&nbsp;<strong>Article</
strong>&nbsp;&nbsp;</td>
 <td>&nbsp;&nbsp;&nbsp;</td>
 <td class="GroupTitle">&nbsp;&nbsp;<a class="ABC"
href="Journal.html">Journal</a>&nbsp;&nbsp;</td>
 <td>&nbsp;&nbsp;&nbsp;</td>
 <td class="GroupTitle">&nbsp;&nbsp;<a class="ABC"
href="Book.html">Book</a>&nbsp;&nbsp;</td>
</tr>
```

7 Clean up the input tags.

Each file (`Article.html`, `Book.html`, and `Journal.html`), as delivered, contains form INPUT tags that contain OpenURL 1.0 elements. You should remove these elements from the INPUT tags so that your form looks good when hosted on your own server. You can find these elements by looking for `<TMPL_VAR ESCAPE='HTML' NAME=` when contained within an INPUT tag.

Change the following:

```
<input
    type=text
    name="rft.atitle"
    size="33"
    value="<TMPL_VAR ESCAPE='HTML' NAME='rft.atitle'>">
```

to:

```
<input
    type=text
    name="rft.atitle"
    size="33"
    value="">
```

5

SFX Targets and Target Parsers

A target parser is a program that creates a URL to go to a specific place on the Internet, such as a publisher's Web site or the abstract of a particular journal article. In the SFX environment, each target service and object portfolio uses a target parser to create a URL which the target service and object portfolio resolve.

This section includes:

NOTES:

- For general information on SFX targets and target parsers, refer to the *SFX General User's Guide*.
- The SFX administrator can choose to use either local or global parser/ parse_param combinations. No mixture of the two is possible, whether in the Target Service table or in the Object Portfolio table.

PARSE_PARAM Fields

`PARSE_PARAM` fields contain information used by target parsers to create URLs to which the SFX target service resolves. They have the syntax: `<parameter_name>=<parameter_value>`. If there are multiple parameters, use an ampersand (`&`) to combine them into one statement.

The following PARSE_PARAM fields exist in the database:

- In the Target Service table:
  - Parameters that apply to the whole target service
  - Syntax – url=base_URL
- In the Object Portfolio table:
  - Parameters that apply to one particular object portfolio
  - Syntax – jkey=fill_in_your_jkey_here

To identify the information/components required for the URL structure of a particular TARGET_SERVICE or OBJECT_PORTFOLIO, review the examples below.

Examples of PARSE_PARAM Fields

This section includes several PARSE_PARAM field examples for your review.

American Chemical Society Example

The following information is needed for the American Chemical Society example:

- The PARSE_PARAM field in the getFullTxt target service needs to contain:

  url= http://pubs3.acs.org/acs/journals/toc.page
- The PARSE_PARAM field in the object portfolio for the *Accounts of Chemical Research* journal must contain the following information:

  jkey=achre4

Sometimes the URL structure of a target service is more complicated and additional information must be provided in the PARSE_PARAM fields.

In the PARSE_PARAM field of the target service, more than one URL can be entered. This is necessary, for example, when one URL is used to link to a journal's home page and another URL to link to the abstract level of the journal.

American Physical Society Example

The PARSE_PARAM field of the full-text service of the American Physical Society (PROLA archive) contains the following:

```
url1=http://ojps.aip.org/journal_cgi/dbt &
url2=http://publish.aps.org/abstract &
url3=http://prola.aps.org
```

In the PARSE_PARAM field of the OBJECT_PORTFOLIO, more than one journal key can be entered. This is necessary when a different jkey is used for different levels of linking. The different jkeys in the PARSE_PARAM field are separated by commas. For example, the *Physical Review Letters* is part of the American

Physical Society `getFullTxt` service and has the following information in the `PARSE_PARAM` field:

```
jkey=prl,PRL,PRLTAO
```

American Journal of Psychiatry Example

In the `PARSE_PARAM` field of the `OBJECT_PORTFOLIO`, it is sometimes necessary to indicate that a journal (or group of journals) uses a slightly different URL structure than other journals belonging to the same `TARGET_SERVICE`. These journals have a `code=1,2,3,…` in the `PARSE_PARAM` field. For example, the *American Journal of Psychiatry* is part of the HighWire Press `getFullTxt` service and has the following `PARSE_PARAM` field:

```
jkey=ajp.psychiatryonline & code=3
```

NOTES:

Characters that are part of the `<parameter_name>` or `<parameter_value>` should be URL-encoded in case they are not safe to use in a URL without URL encoding. More information on URL encoding can be found at http://www.blooberry.com/indexdot/html/topics/urlencoding.htm. See the following examples:

- Example 1:

  The following is a Jkey element that includes a URL with two ampersands:

  ```
  jkey=http://www.evaluationcanada.ca/
  site.cgi?section=4&ssection=3&_lang=an
  ```

  The following is the same URL with encoded ampersands:

  ```
  jkey=http://www.evaluationcanada.ca/
  site.cgi?section=4%26ssection=3%26_lang=an
  ```

- Example 2:

  The following is a URL element after URL encoding that includes URLs with spaces and ampersands:

  ```
  url1=http://www.university.corg?a=xx%26b=yy & url2=http://
  www.library.net/book%20title/
  ```

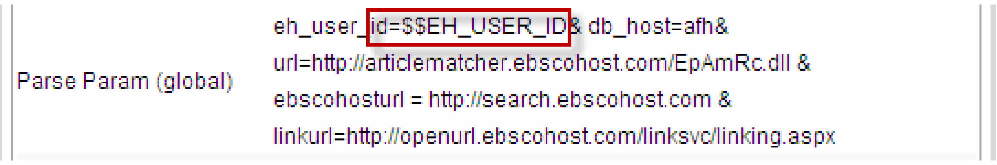
  Note that the ampersand used to combine URL1 and URL2 is not URL-encoded. This is not required.

Linking Parameter Table

The Linking Parameter table is used to store local server, customer authentication information, and any other variables necessary to link to specific targets.

In each target service for which linking parameter information is required, a L/P button is displayed in:

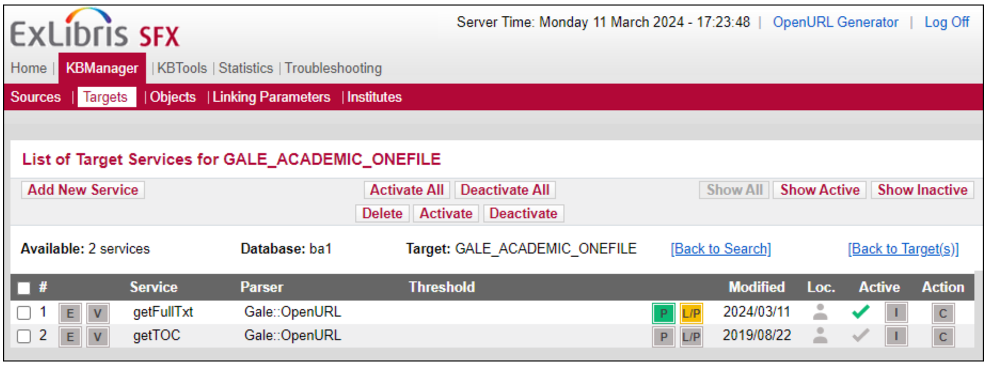
- Edit Target Service
- List of Target Services



Parse Param (global)	eh_user_id=$$EH_USER_ID& db_host=afh&
	url=http://articlematcher.ebscohost.com/EpAmRc.dll &
	ebscohosturl = http://search.ebscohost.com &
	linkurl=http://openurl.ebscohost.com/linksvc/linking.aspx

Figure 42: Edit Target Service

Click the L/P button to fill in the required authentication information for the `ID_VALUE` field. The help file next to each target service `PARSE_PARAM` field provides more information about the linking parameter information that needs to be provided.


ExLibris SFX
Server Time: Monday 11 March 2024 - 17:23:48 | OpenURL Generator | Log Off
Home | KBManager | KBTools | Statistics | Troubleshooting
Sources | Targets | Objects | Linking Parameters | Institutes
Add New Service
Activate All | Deactivate All
Delete | Activate | Deactivate
Show All | Show Active | Show Inactive
[Back to Search]
[Back to Target(s)]

List of Target Services for GALE_ACADEMIC_ONEFILE

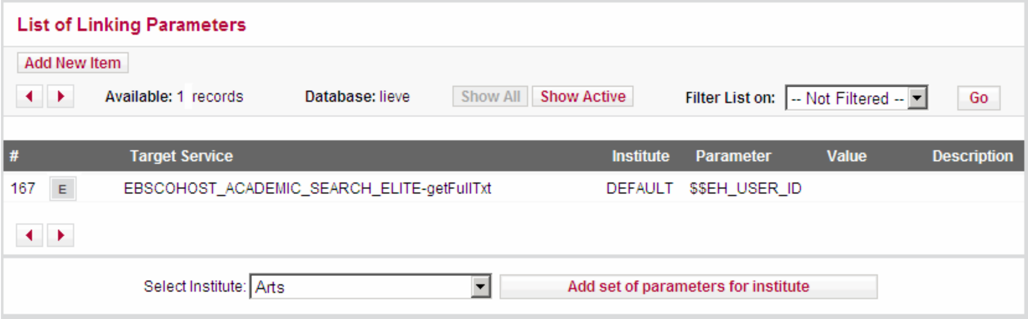
Available: 2 services **Database:** ba1 **Target:** GALE_ACADEMIC_ONEFILE

□ #			Service	Parser	Threshold			Modified	Loc.	Active		Action
□ 1	E	V	getFullTxt	Gale::OpenURL		P	L/P	2024/03/11		✓	I	C
□ 2	E	V	getTOC	Gale::OpenURL		P	L/P	2019/08/22		✓	I	C

Figure 43: List of Target Services

The Linking Parameters button is displayed in yellow if at least one L/P has been filled in for either DEFAULT or institute/Group.

See the example below for the `EBSCO_ACADEMIC_SEARCH_ELITE` full-text target service.



List of Linking Parameters

Add New Item

Available: 1 records Database: lieve Show All Show Active Filter List on: -- Not Filtered -- Go

#		Target Service	Institute	Parameter	Value	Description
167	E	EBSCOHOST_ACADEMIC_SEARCH_ELITE-getFullTxt	DEFAULT	$$EH_USER_ID		

Select Institute: Arts Add set of parameters for institute

Figure 44: EBSCO_ACADEMIC_SEARCH_ELITE Full-Text Target Service

If the `PARSE_PARAM` field includes a `$$PARAMETER` value, the **+L/P** button appears when adding a local target service, indicating that a linking parameter should be added.

Storing Institute-Specific Linking Parameters for a Particular Target

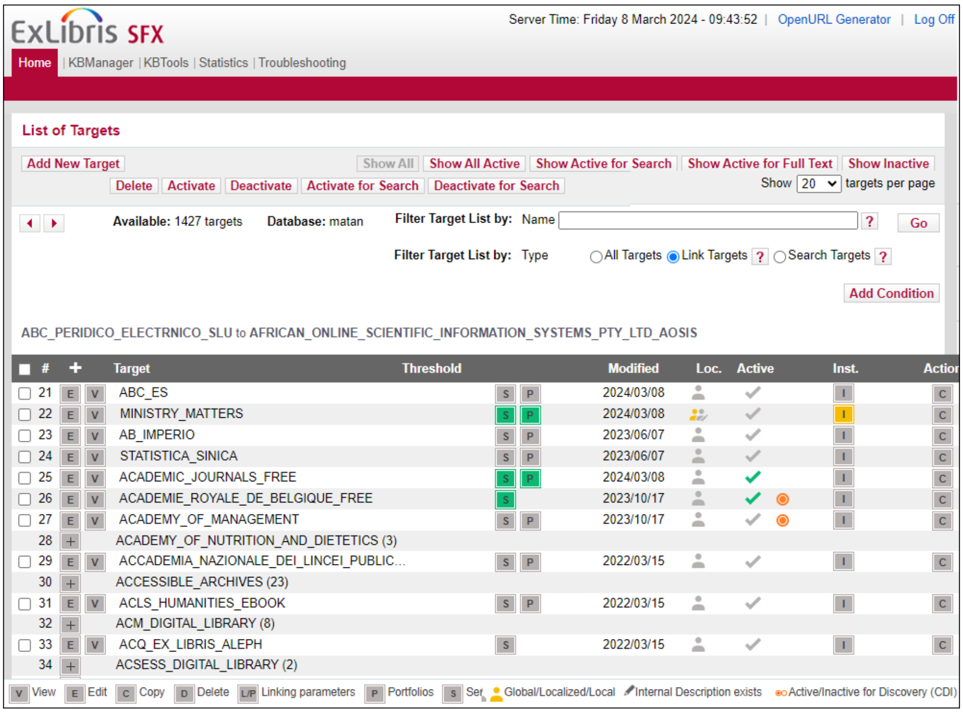
It is possible to specify institute-specific linking parameters for a particular target service.

The following fields should be specified:

- Target service name
- Name – The content of this field (in the given example: `$$EH_USER_ID`) matches the `$$` parameter in the `parse_param` field. When SFX creates a specific target URL, the `$$` parameter in the `parse_param` field is replaced with the content of the corresponding `Value` field.
- Value – The actual user name, password, or other local parameters for the specific institute.

To set up institute-specific linking parameters:

1 From the KBManager section in the SFX Admin Center, click **Targets**. The list of targets is displayed.



Server Time: Friday 8 March 2024 - 09:43:52 | OpenURL Generator | Log Off

Home | KBManager | KBTools | Statistics | Troubleshooting

List of Targets

Add New Target   Show All | Show All Active | Show Active for Search | Show Active for Full Text | Show Inactive

Delete | Activate | Deactivate | Activate for Search | Deactivate for Search   Show 20 targets per page

Available: 1427 targets   **Database:** matan   **Filter Target List by:** Name [ ] ? Go

Filter Target List by: Type ○ All Targets ● Link Targets ? ○ Search Targets ?

Add Condition

ABC_PERIDICO_ELECTRNICO_SLU to AFRICAN_ONLINE_SCIENTIFIC_INFORMATION_SYSTEMS_PTY_LTD_AOSIS

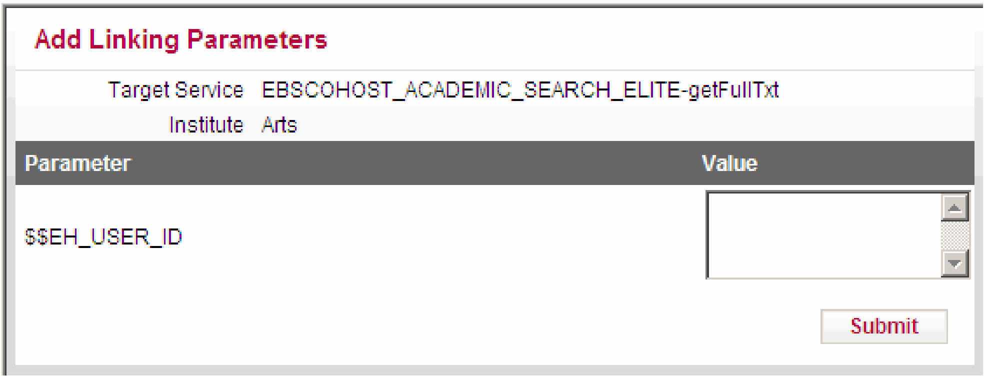
□	#	+	Target	Threshold	Modified	Loc.	Active		Inst.	Actior
□	21	E V	ABC_ES	S P	2024/03/08	👤	✓		I	C
□	22	E V	MINISTRY_MATTERS	S P	2024/03/08	👥	✓		I	C
□	23	E V	AB_IMPERIO	S P	2023/06/07	👤	✓		I	C
□	24	E V	STATISTICA_SINICA	S P	2023/06/07	👤	✓		I	C
□	25	E V	ACADEMIC_JOURNALS_FREE	S P	2024/03/08	👤	✓		I	C
□	26	E V	ACADEMIE_ROYALE_DE_BELGIQUE_FREE	S	2023/10/17	👤	✓	●	I	C
□	27	E V	ACADEMY_OF_MANAGEMENT	S P	2023/10/17	👤	✓	●	I	C
	28	+	ACADEMY_OF_NUTRITION_AND_DIETETICS (3)							
□	29	E V	ACCADEMIA_NAZIONALE_DEI_LINCEI_PUBLIC...	S P	2022/03/15	👤	✓		I	C
	30	+	ACCESSIBLE_ARCHIVES (23)							
□	31	E V	ACLS_HUMANITIES_EBOOK	S P	2022/03/15	👤	✓		I	C
	32	+	ACM_DIGITAL_LIBRARY (8)							
□	33	E V	ACQ_EX_LIBRIS_ALEPH	S	2022/03/15	👤	✓		I	C
	34	+	ACSESS_DIGITAL_LIBRARY (2)							

Figure 45: Target List

2 Click the **Service** button S next to the target service to which you want to add different linking parameters for each of the different institutes.

3 Click the **L/P** button L/P . The List of Linking Parameters dialog box is displayed.

4 Select the institute from the drop-down list and click the **Add Set of Parameters for Institute** button. The Add Linking Parameters dialog box opens:



Add Linking Parameters

Target Service EBSCOHOST_ACADEMIC_SEARCH_ELITE-getFullTxt

Institute Arts

Parameter	Value
$$EH_USER_ID	

Figure 46: Add Linking Parameters Dialog Box

5 Enter the authentication information in the **Value** field.

6 Click **Submit** to save the information.

Repeat this procedure for each of the institutes for which specific user name and password information or other local information needs to be specified.

Using the Generic Target Parser

Using the generic target parser, you can create a target parser “on the fly” (without writing a Perl script). The generic parser can be used to test URLs, store syntax, and add new journals to the SFX database.

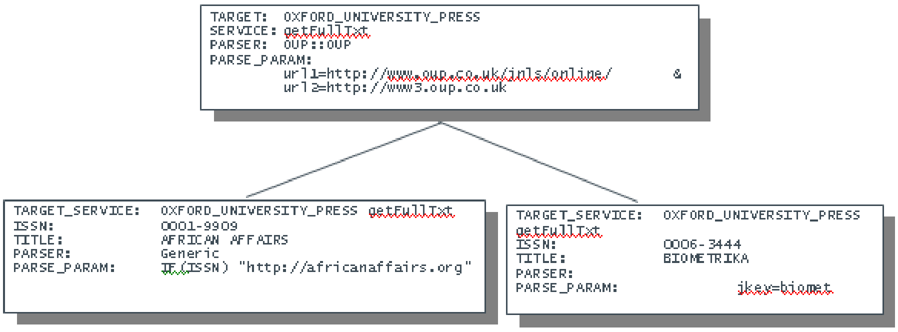
The generic target parser can be used both on the OBJECT_PORTFOLIO and TARGET_SERVICE level.

To use the generic target parser:

1 Enter `Generic` in the Parser field.

2 Enter the URL information using the generic target parser syntax in the `PARSE_PARAM` field.

The following example uses the generic target parser on the OBJECT_PORTFOLIO level:



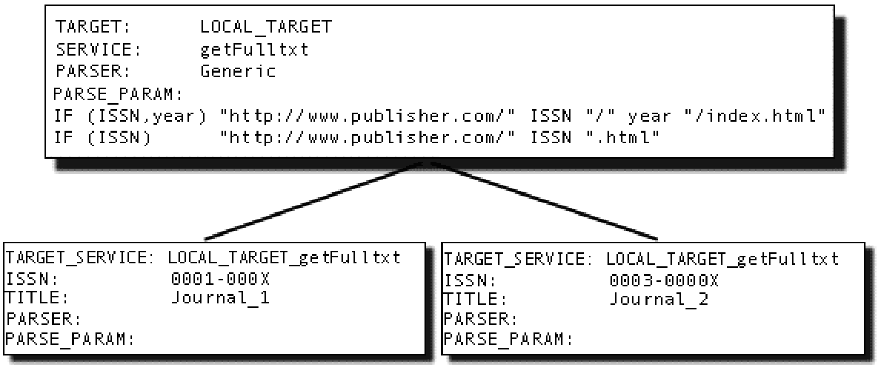
```
TARGET:   OXFORD_UNIVERSITY_PRESS
SERVICE:  getFullTxt
PARSER:   OUP::OUP
PARSE_PARAM:
          url1=http://www.oup.co.uk/jnls/online/        &
          url2=http://www3.oup.co.uk
```

```
TARGET_SERVICE:   OXFORD_UNIVERSITY_PRESS  getFullTxt
ISSN:             0001-9909
TITLE:            AFRICAN AFFAIRS
PARSER:           Generic
PARSE_PARAM:      IF (ISSN) "http://africanaffairs.org"
```

```
TARGET_SERVICE:   OXFORD_UNIVERSITY_PRESS
getFullTxt
ISSN:             0006-3444
TITLE:            BIOMETRIKA
PARSER:
PARSE_PARAM:                jkey=biomet
```

Figure 47: Generic Target Parser in the OBJECT_PORTFOLIO

The following example uses the generic target parser on the TARGET_SERVICE level:



```
TARGET:      LOCAL_TARGET
SERVICE:     getFulltxt
PARSER:      Generic
PARSE_PARAM:
IF (ISSN,year) "http://www.publisher.com/" ISSN "/" year "/index.html"
IF (ISSN)      "http://www.publisher.com/" ISSN ".html"
```

```
TARGET_SERVICE:  LOCAL_TARGET_getFulltxt
ISSN:             0001-000X
TITLE:            Journal_1
PARSER:
PARSE_PARAM:
```

```
TARGET_SERVICE:  LOCAL_TARGET_getFulltxt
ISSN:             0003-0000X
TITLE:            Journal_2
PARSER:
PARSE_PARAM:
```

Figure 48: Generic Target Parser in the TARGET_SERVICE

Generic Target Parser Syntax

This section describes the generic target parser syntax.

- Generic Target Parser Syntax Example on page 92
- Generic Target Parser Syntax Example for Static Web Pages on page 93

Generic Target Parser Syntax Example

In the following example, the generic target parser syntax describes how to link to:

- A specific journal article
- A particular year of a journal
- The home page of a journal

The link that is created depends on the amount of information available in the context object (CTXO) (see **ContextObject** on page **209**).

For example:

```
IF(ISSN,year,startPage)
"http://www.publisher.com/" ISSN "/" year "/" startPage
IF(ISSN,year)
"http://www.publisher.com/" ISSN "/" year "/index.html"
IF(ISSN)
"http://www.publisher.com/" ISSN "/"
```

Generic Target Parser Syntax Example for Static Web Pages

Another popular use of the generic target parser is to create targets that are static Web pages, such as a library's FAQ page or a page describing the library's digital reference service. For example:

```
IF ()
"http://www.your.library.edu/FAQ.html”
IF ()
"http://www.your.library.edu/askalibrarian.html”
```

Variables in the `PARSE_PARAM` field are replaced with the relevant content of the context object (CTXO). The generic target parser returns the first condition that matches. This means that if only an ISSN is available, the last row is matched. When users click this service on the SFX menu, they are sent to the home page of the journal.

Functions

The following functions can be used in the generic target parser syntax:

- `REPLACE(string,search,replace)]`

  For example:

```
IF (ISBN)
"http://www.amazon.com/search.cgi?isbn=" REPLACE(ISBN,"-","")
```

- `[SUBSTR(string,from,length)]`

  For example:

```
IF (year)
"http://www.thisyear.com/" SUBSTR(year,2,2)
```

- [URLENCODE(string)]

  Example:

```
IF (articleTitle)
"http://www.google.com/search?q=" URLENCODE(articleTitle)
IF (@abbrevTitle)
"http://www.google.com/search?q=" @abbrevTitle[0]
```

Target Parser Information

This section contains information about target parsers and recommendations on how to write them.

NOTE:
Perl knowledge is required to write target parsers.

The SFX installation includes a template target parser that is used as an example.

The template is located in: `/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/TargetParser/TEMPLATE/TEMPLATE.pm`.

```
# Version: $Id: TEMPLATE.pm,v 1.6
package Parsers::TargetParser::TEMPLATE::TEMPLATE;
use base qw(Parsers::TargetParser);
use URI;
use strict;
sub <service_same> {
 my ($this) = @_;
my $ctx_obj = $this->{ctx_obj}; # retrieve context object
  my $value = $ctx_obj->get('<ctx_object attribute>');
my $svc = $this->{svc}; # retrieve service object
  my $value = $svc->parse_param('<param name as it appears
in the Parse Param>');
my $base_url = "";
my $uri = "";
my %query = ();
if (length $value) {
 $query{'name'} = $value; # name and value as it should
appears in the url result
}
$uri = URI->new($base_url);
$uri->query_form(%query);
return $uri;
}
1;
```

Target Parser Location

All target parsers can be found in the following location:

`/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/TargetParser/<target_parser_name>`.

For example:

```
/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/TargetParser/
TEMPLATE/TEMPLATE.pm
```

Target Parser Name Convention

The following describes the target parser naming convention:

- Directory – Directory with the name of the target, in uppercase letters
- File name – `<target_name>.pm` (recommended to use uppercase letters)

For example:

```
TEMPLATE/TEMPLATE.pm
```

The package name should be constructed from the target parser name and listed in the first line of the target parser. See line 1 of the template:

```
Parsers::TargetParser::TEMPLATE::TEMPLATE;
```

Subroutine Name

The target parser should contain a subroutine for every target service associated with the target. The subroutine name is identical to the `serviceType`. For example:

```
getFullTxt, getTOC, getAbstract, getHolding etc'
```

Constructing the Target URL

The target URL is divided into two parts:

- The base URL – the part up to the question mark (?). For example:

  `http://......?`

- The query string – the part after the question mark (?) in `name=value pairs` separated by the ampersand (&). For example:

  `a=b&c=d&e=f…`

  (see line 3 in the target parser template)

In order to construct a URL, the target parser should use the URI module. Objects of this class represent Uniform Resource Identifier references.

The base URL is the input to the constructor method. For example:

```
my $uri = URI->new($base_url);
```

See line 16 in the target parser template.

The query string is implemented through a hash. The query string key is the name, and its value is the name's value. For example:

```
$query{'volume_id'} = $volume;
$query{'issue_id'} = $issue;
```

The constructed hash should be given as input to the `query_form`. For example:

```
$uri->query_form(%query);
```

The target parser should return the URI object.

See line 18 in the target parser template.

For more information, refer to: http://search.cpan.org/~gaas/URI-1.35/URI.pm#SYNOPSIS.

Integrating Metadata and Parameters into URL

In order to integrate the `ContextObject` into the query string, use the `ContextObject` object. For example:

```
my $ctx_obj = $this->{ctx_obj};
```

See line 7 in the target parser template.

See **ContextObject** on page **209** for more information about the `ContextObject` and `ContextObject` attributes.

To retrieve attributes from the `ContextObject`, use the get method of the context object in the following way:

```
my $attribute_value = $ctx_obj->get(\<attribute_name>');
```

A list of all `ContextObject` attributes that can be used with the get method, can be found in `/exlibris/sfx_ver/sfx4_1/<instance>/config/ctx_attribute.config`.

See line 8 in the target parser template.

For example:

```
my $issue = $ctx_obj->get('rft.issue');
```

To set the `ctx_obj` attributes, use the set method of the context object in the following way:

```
$ctx_obj->set(\<attribute_name>')=<value>
```

`<value>` can be a string or a reference to an array when `<attribute_name>` is an array.

NOTE:
If `<attribute_name>` is an array such as `rft.aulast`, the `$attribute_value` is a reference to an array. To retrieve the actual value, use `$attribute_value->[$i]` where `$i` is the array index.

In order to integrate the parameters stored in the SFX database `PARSE_PARAM` field into the URI, use the `PARSE_PARAM` method of the service object. The service object is the metadata container that includes all the information for the `TARGET_SERVICE` that was chosen (including the `TARGET` and `TARGET_SERVICE` public names, `PARSE_PARAM` information, and so forth).

You first need to retrieve the service object.

See line 10 in the target parser template.

For example:

```
my $svc = $this->{svc}; # services
my $jkey  = $svc->parse_param('jkey');
my $host  = $svc->parse_param('url')
```

Syntax Checking

After writing the target parser, check its syntax by running the following command:

```
Perl -cw <TARGET_PARSER_name>;
```

Debugging

In order to debug the target parser, turn on target debugging.

To turn on target debugging:

1 Choose **Debugging** in the Troubleshooting section of the SFX Admin Center.

2 Turn on the **Debugging in TARGET SCREEN** option.

To test debugging once it has been activated:

1 Send an OpenURL that brings up the target service in the SFX menu.

2 Click **Go** to activate the target parser.

Debugging information is printed in the target window. The bottom part of the debugging information displays the target URL constructed by the target parser.

See **Debugging Tool** on page **141** for more information.

The target parser is also traced in the Apache error log, and the `warn` command can be used in the code for debugging purposes.

Target Parser Example

See the example in `/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/TargetParser/MASSON/E2MED.pm`.

```
# Version: $Id: E2MED.pm,v 1.3 2006/05/24 06:39:35 nk_3 Exp $
package Parsers::TargetParser::MASSON::E2MED;

use base qw(Parsers::TargetParser);
use URI;
use strict;

sub getFullTxt {    # the service name

   my ($this) = @_;

  my $svc = $this->{svc}; # retrieve service object
    my $jkey          = $svc->parse_param('jkey'); # retrive
parse_param values of target service
   my $host      = $svc->parse_param('url');

   my $uri = "";
   my %query = ();

  if (length $jkey) {
       $uri = URI->new("$host/masson/portal/bookmark");  #
cunstruct the URI object  with the base url as input.
       $query{'Global'} = 2; # Populate the hash for the
query_form URI method.
       $query{'Page2'} = 1;
$query{'MenuIdSelected'} = 102;
     $query{'MenuItemSelected'} = 0;
     $query{'MenuSupportSelected'} = 0;
     $query{'Product2'} = $jkey;
     $query{'ProductCode2'} = $jkey;

      $uri->query_form(%query);  # set the query string using
query_form method
      return $uri;
   }

$uri = URI->new($host);  );  # construct the URI object with
the base url.
   return $uri;
}
1;
```

NOTES:

- The query string part of the target URL is URL-encoded by the URI object. As a result, it is not necessary to handle URL encoding in the target parser.
- In order to apply the new target parser, the Apache instance must be restarted. Restarting Apache involves a short period of downtime. For more information on restarting apache, refer to the **Start/Stop Services** section of the *SFX System Administration Guide*.

GET and POST Notes

Nearly all of the standard Ex Libris-supported SFX target parsers are written to create a final URL in the form `GET`.

When a local developer wants to create a URL using the `POST` form in a local target parser, it is recommended that you:

- Have the target parser return the URL of a Web page that posts the URL to the target
- Have the target parser generate a URL that calls a CGI that generates the appropriate POST request.

  Refer to the following global targets that use this approach:

  - LOCAL_FEEDBACK
  - DOCDEL_LOCAL

  These targets have corresponding CGI scripts in the `/cgi/public` directory.

6

SFX Admin Center Configuration Files

The `sfxctrl.config` and `ctx_login.config` configuration files located in the `/config` directory of your UNIX file system can be used to change the SFX Admin Center settings.

This section includes:

Editing Configuration Files

The `sfxctrl.config` and `ctx_login.config` files are located in the `/config` directory of your UNIX file system.

To edit these files:

1. Telnet to the SFX server.
2. Log on to the SFX instance with which you want to work.
3. Enter `cn` to go to the configuration directory.

sfxctrl.config

In the `sfxctrl.config` configuration file, the following items can/need to be changed in the “`extra`” section:

- `mailfrom/mailcc`

  Use the `mailfrom` and `mailcc` parameters to identify the e-mail addresses to be used as the default from and carbon copy (cc:) e-mail addresses. These email addresses display when a user clicks the **E-Mail Ex Libris** button in the KBManager.

```
Section "extra"
    sfx_help_setting      "CRM"
    sfx_help_CRM_address  "http:// support.exlibrisgroup.com"
    help_email            ""
    mailto                "sfx-changes@exlibris-usa.com"
    mailfrom              ""
    mailcc                ""
    sort_portfolio        "object_identifier"
    default_sort_portfolio_max_num "500"
    session_expire        "4h"
    temp_table_type       "DEFAULT"
    use_RSI               "Y"
EndSection
```

- `sort_portfolio`

  The `sort_portfolio` option determines the layout of the OBJECT_PORTFOLIO list in the KBManager.

  The default for this option is `object_identifier`. This means that the OBJECT_PORTFOLIO list is sorted numerically by ISSN.

  The `sort_portfolio` option can be set to `0` to increase the display speed or performance of the SFX Admin Center.

  Valid values for sort_portfolio are by_title, object_identifier, or 0. If the OBJECT_PORTFOLIO list is longer than the number specified in the default_sort_portfolio_max_num parameter, the list will initially be shown unsorted. Once the customer clicks the ID or Title column header, the object portfolio list will be sorted, regardless of the default_sort_portfolio_max_num parameter value.

NOTE:
If the setting is `0`, the OBJECT_PORTFOLIO list is not sorted.

- `session_expire`

  The `session_expire` parameter defines the timeout length for the SFX Admin Center sessions. The amount of time can be set by specifying the length of time followed by a single character indicating the type of time, as follows:

  - Seconds (s)
  - Minutes (m)
  - Hours (h)
  - Weeks (w)
  - Months (M)

- Years (y)

For example:

```
session_expire        "30m"
session_expire        "3w"
```

If no value is defined for session_expire, the default value is 4h (four hours).

- sfx_help_setting/sfx_help_CRM_address/help_email

  The sfx_help_setting, sfx_help_CRM_address, and help_email parameters determine the screen that is displayed when a user clicks the **Contact Support** link in the SFX Admin Center.

  The sfx_help_setting can have one of two values:

  - CRM
  - local

  If the value is set to CRM, the logon page to the Ex Libris CRM E-Service Support Center opens. For example:

```
sfx_help_setting      "CRM"
   sfx_help_CRM_address  "http://support.exlibrisgroup.com"
   help_email              ""
```

  If the value is set to local, a help form opens that you can submit to the designated support e-mail address in your library. The local option is recommended for consortia in which individual institutes do not have a CRM logon, but contact a central administrator for assistance. With the local option, you also need to enter an e-mail address of your choice for the help_email setting. For example:

```
sfx_help_setting      "local"
   sfx_help_CRM_address  ""
   help_email            "admin@consortium.edu"
```

ctx_login.config

In the ctx_login.config configuration file, the following items can/need to be changed in the "login" section:

- password

  The password parameter must be completed for the database (also called mysql).

- timeout

  The timeout parameter can be set in minutes. In the example below, timeout is set for 15 minutes.

```
Section "login"
    database    "sfxlcl41"
    driver      "mysql"
    hostname    "localhost"
    port        "3310"
    username    "sfxlcl41"
    password    "6726174676c6431333535303236303"
    timeout     "10"
    oneadmin    "0"
    seq_mode    "local"
    socket      "/exlibris/sfx_ver/sfx4_1/tmp/socket/
mysql.sock"
    instance    "sfxlcl41"
EndSection
```

- oneadmin

  When oneadmin is set to 0, more than one person is allowed to use the SFX Admin Center simultaneously.

  If this option is set to 1, only one person can access the SFX Admin Center at any given time.

config/admin.config

The option records_per_page can be set via the config/admin.config file of the local instance to define the default number of object portfolios displayed in in the List of Object portfolios in KBmanager.

You can define the number of sources, target and portfolios to display by default when accessing a list of portfolios by choosing from the following predefined list of values: 10/20/ 50/100/200.

```
Configuration section:
Section "default"
#allowed values for 'records_per_page' are limited to 10, 20,
50, 100, or 200
records_per_page "20"
EndSection
```

The config/admin.config file has configuration options for setting the time-out for the SFX Admin Center and whether it is possible to log on with more than one simultaneous user:

```
Section "login"
expire "+1h"
multilogin "0"
EndSection
```

In the section kbmanager, you can configure the following:

- The default search fields displayed in the KBManager > Search Objects page, as well as the order and the number of the search fields.
- The display of an active portfolio indicator if an object found via the KBManager > Search Objects page has at least one active portfolio

```
Section "kbmanager"
# Possible options for object_search:
#       issn,isbn,lccn,coden,doi,local,object_id,publisher
#
publication_place,oclc_nr,object_type,object_language,title,
#
category,author_personal,author_corporate,author_meeting
# You can chose as many as you need.
# for example:
#       object_search         "Title,ISSN,ISBN"
# Incorrect options as well duplicate fields will be ignored.
        object_search         "Title,ISSN,ISBN"
# Possible options for active_op_indicator: "Y" or "N"
# Setting it to "Y" will provide indicator in "Search Objects"
screen whether the object has at least one active portfolio.
        active_op_indicator   "Y"
EndSection
```

7

KBTools Command Lines

The command line versions of the KBTools can be useful for SFX administrators who prefer to work in the UNIX environment, and for crontab scheduling purposes.

This section includes:

NOTE:

For information regarding KBTools' Web interfaces, refer to the *SFX General User's Guide.*

Command-Line DataLoader

The following contains information about the command-line version of the DataLoader:

```
Usage: ./dataloader.pl

--mode=<mode>

Values: portfolios, update_objects, add_remove_objects,
target_cdi_discovery_activation

--file=<file>
```

```
Data file. Full path, or found in '/exlibris/sfx_ver/sfx4_1/
sst1_4/dbs/scratch'.

--target_service=<ts> (portfolios only)

Name of target-service(s). Required.

--sub_target_service=<ts> (portfolios only)

Name of sub-target-service.

Should be used in case --target_service is specified which has
multiple sub-target-services.

The sub-target-service specified should be connected to the
Target Service specified in the --target_service.

--all_sub_targets (portfolios only)

Should be used in case --target_service is specified which has
multiple sub-target-services.

Will activate all subtargets.

--institute=<institute>
(portfolios,target_cdi_discovery_activation only)

Optional name of institute/group.

--column1=<type>

Type of Primary Key column. Required for portfolios,
update_objects, add_remove_objects.
Do not use for target_cdi_discovery_activation

Possible values for portfolios, update_objects:

ISSN, ISBN, LCCN, OBJECT_ID

Possible values for add_remove_objects:

OBJECT_TYPE (for load_type=add)

OBJECT_ID (for load_type=remove)

--column#=<type> (# = 2-9)

Types of other columns. To skip a column in the data file,

don't specify a --column# option for that column.

Possible values for portfolios:

THRESHOLD, TARGET_PARSER_PROGRAM, PARSE_PARAM,
ACTIVATION_STATUS, AUTHENTICATION_NOTE, GENERAL_NOTE,
INTERNAL_DESCRIPTION

Possible values for update_objects:

OBJECT_TYPE, LOCAL, TITLE_LANGUAGE, TITLE_TYPE, TITLE_VALUE,
TITLE_NON_FILING_CHAR, LANGUAGE, PUBLICATION_DATE,
PUBLICATION_PLACE, PUBLISHER
```

```
Possible values for add_remove_objects:

LOCAL, TITLE_LANGUAGE, TITLE_TYPE, TITLE_VALUE,
TITLE_NON_FILING_CHAR, LANGUAGE, ISSN, EISSN,

ISBN_ELECTRONIC, ISBN_HARDBACK, ISBN_PAPERBACK, LCCN, CODEN,
DOI, OCLC_NR, PUBLICATION_DATE, PUBLICATION_PLACE, PUBLISHER

--kbart_format (portfolios only)

This option indicates that --column definition is not used -
but kbart format is used instead

--kbart_activation_status=<status>

Possible value for <status> = ACTIVE/INACTIVE

This option is required in case kbart_format is used

--kbart_ignore_coverage_depth

Optional - when included - coverage_depth will be ignored
during load

--kbart_parse_param=<jkey>

Possible value for <jkey> = title_id / title_url

Optional - when included - PARSE_PARAM field will be loaded
with either jkey=title_url or jkey=title_id if exists

--report_mode

Report what would be done with the data file,

but do not actually update the Database.

--add (portfolios only)

Add portfolios to the target-service.

--update (portfolios only)

Update portfolios for the target-service.

--remove (portfolios only)

Remove portfolios from the target-service.

--load_type=<load_type> (update_objects,
add_remove_objects, target_cdi_discovery_activation))

Possible values for update_objects: keep, overwrite, remove

Possible values for add_remove_objects: add, remove

Possible values for target_cdi_discovery_activation:
activate_targets_cdi_discovery,
deactivate_targets_cdi_discovery

--activate (portfolios only)
```

```
Activate the chosen target and target_services

--quiet

Don't print summary on success

OR

./dataloader.pl --help

OR

./dataloader.pl --list

--change_activation_status_multi_ops (update portfolios only)

Allow updating multiple OPs (under same object and same
target-target_service) by Parse Params and Threshold matching

Note: For threshold matching, update can only be for
availability_status (active/inactive)
```

Command-Line Information

The following files contain command-line information:

- Cleaned input file – `/exlibris/sfx_ver/sfx4_1/<instance>/dbs/scratch/<input_file>.ok`
- Report of loading – `/exlibris/sfx_ver/sfx4_1/<instance>/dbs/scratch/<input_file>.ok.err`
- Log file of dataloader – `/exlibris/sfx_ver/sfx4_1/<instance>/dbs/scratch/dataloader.pl.log`

Command-Line Examples

To run the DataLoader program on a file called `testAmerican_Medical_Association` that exists in the scratch directory, use the following command:

```
./dataloader.pl --mode=portfolios --
file=testAmerican_Medical_Association --
target_service=AMERICAN_MEDICAL_ASSOCIATION-getFullTxt --
column1=ISSN -column2=AVAILABILITY --column3=THRESHOLD --
update --activate
```

In this example, the DataLoader is used to update the OBJECT_PORTFOLIOs for the American Medical Association target (activate and add a local threshold). At the same time, the target and target_service will be activated.

The tab-delimited TXT input file looks like the following:

```
0098-7484 ACTIVE $obj->parsedDate(">=",2002,131,1)
```

The following is an example for commandline dataloader to add objects:

```
./dataloader.pl --mode=add_remove_objects --load_type=add --
file=test10.txt --column1=OBJECT_TYPE --column2=ISBN_HARDBACK
-column3=TITLE_VALUE --column4=PUBLICATION_PLACE --
column5=PUBLISHER --column6=PUBLICATION_DATE
```

An example for commandline dataloader in kbart format is the following:

```
./dataloader.pl --mode=portfolios --file=/exlibris/sfx_ver/
sfx4_1/sst1_4/dbs/scratch/dataloader.txt --
target_service=ADISONLINE-getFullTxt --all_sub_targets --
kbart_format --kbart_activation_status=ACTIVE --
kbart_ignore_coverage_depth --kbart_parse_param=title_url --
add
```

The tab-delimited TXT input file looks like the following (only the first four columns are displayed here):

```
publication_title   print_identifier   date_first_issue   num_first_
                                       _online            vol_online

Philosophical       0370-2316          1665               1
Transactions
```

For a complete list of the columns of the KBART format, see the **KBART** section of the *SFX General User's Guide*.

Command-Line Lookup Tool

The following contains information about the lookup tool:

```
Help information:
./lookup.pl
--input-file=<input_file> --input-content=<field_name>
--lookup-content=<field_name>
input_file location defaults to /exlibris/sfx_ver/sfx4_1/
dana_3/dbs/scratch if no path is given.
Possible field_name values are: ISSN, eISSN, ISBN, LCCN,
TITLE, OBJECT
```

Command-Line Lookup Tool Examples

The following are lookup tool examples:

```
./lookup.pl --input-file=test1.txt --input-content=TITLE --
lookup-content=ISSN
./lookup.pl --input-file=test1.txt --input-content=TITLE --
lookup-content=LCCN
```

Command-Line Export Tool

The command-line version of the Export tool allows you to run all of the advanced export options that can be performed from the Export Tool option of the SFX Admin Center. Additionally, it is possible to run the basic export query `export_urls`. No other basic export options can be performed from the command-line version of the Export tool. All other basic export options can be run only via the Export Tool option of the SFX Admin Center. For more information, see **Appendix C: External Export** on page **295**.

The command-line export tool is located in: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/admin/kbtools`.

This tool allows you to do the following actions in a command-line mode:

- Create different types of advanced export files.
- Export URLs from active targets, target services, and active object portfolios. (This can also be performed from the basic export section of the SFX Admin Center).

To get to the tool:

1. From the UNIX command prompt, type `da`.
2. Type `cd kbtools/`.
3. Type `./export.pl [Options]`.

NOTE:

If disk space is an issue, you can compress the Google export file – `institutional_holding.xml` – as a ZIP file by adding the syntax `--compress` at the end of the export statement.

The following is a list of all the options that can be used in the command-line export tool:

```
Usage:  ./export.pl

Basic Queries:

        --basic

            Values: urls

        --institute

            Optional name of institute/group to filter
results.

            Multiple option.

--service
```

```
            Optional name of service(s) to export.  Multiple
option.

            If not given, export ALL services.

        --output_dir        (profile or manual)

            Optional alternate location of output files.

            Output dir defaults to

            '/exlibris/sfx_ver/sfx4_1/sfxlcl41/dbs/scratch'.

Advanced Queries:

        --mode

            Values: profile, manual

        --output_dir        (profile or manual)

            Optional alternate location of output files.

            Output dir defaults to

            '/exlibris/sfx_ver/sfx4_1/sfxlcl41/dbs/scratch'.

--filename                      (manual only)

            Specify output file name. If filename value is
contains

 a relative or absolute path,

           this is the same as specifing --output_dir. i.e.,

 --filename=/some/path/to/file.xml

           is the same as specifying --output_dir=/some/path/
to

            --filename=file.xml

           This option is not available for the Google export.


   --profile           (profile only)

            Name of profile.  Required.

        --institute         (manual only)

            Optional name of institute/group to filter
results.

            Multiple option.

        --format            (manual only)
```

```
            Required.  Values: TXT, HTML, XML, ALEPH_SEQ,
PUBMED, KBART

            GOOGLE

            Output file format.

        --external_input    (profile or manual)

            Optional name of external TXT format export file
to include.

Location defaults to

            '/exlibris/sfx_ver/sfx4_1/sfxlcl41/dbs/scratch'

            if no path is given

        --target            (manual only)

            Optional name of target to export.  Multiple
option.

            If not given, export ALL active targets/
target_services.

        --service           (manual only)

            Optional name of service to export.  Multiple
option.

            If not given, export ALL services from the
following:

            Values: getFullTxt, getHolding, getTOC,

getAbstract

        --object_type

            Specify types of objects to export

            (e.g., JOURNAL, BOOK). Multiple option.

            If not given defaults to JOURNAL

        --compare           (profile or manual)

            For XML export only.

            Optional name of previous TXT format export

file to

           compare to. Must be located in '/exlibris/sfx_ver/
sfx4_1/sfxlcl41/dbs/scratch/e_collection_update'

  For profile, only used and is required if profile
```

```
            is 'compare'.
        --prefix            (manual only)
            For XML export only.
            Optional file prefix.
        --link_text         (manual only)
            For XML export only. '856 $y' tag
        --inst_info         (manual only)
            For XML export only. '852 $a' tag
        --target_note       (manual only)
            For XML export only. '245 $h' tag
        --base_url          (manual only)
            For XML export: '856 $u' tag
            For PUBMED export:  required Base URL.
    --gif_url           (manual only)
            For PUBMED export only.
            Required gif URL.
        --provider_id       (manual only)
            For PUBMED export only.
            Required ProviderId.
        --no_issn_info  (manual only)
          For GOOGLE export, do not include issn
      --no_portfolio_info (manual only)
          For GOOGLE export, do not include 866 a
        --[no]include_categories        (profile or manual)
         For XML export, will include (or exclude) categories
in
          the export file. Not included by default
        --[no]include_local_attributes  (profile or manual)
          For XML export, will include (or exclude)
local_attributes
          in the export file. Not included by default
        --[no]include_servicetype  (profile or manual)

          Include or exclude "sfx.servicetype=yes" from 856
url.

                  Included by default.

        --[no]institute_info  (profile or manual)

          For XML export, will include (or exclude)
institute_info in

          the export file.

          Not included by default

        --[both]original_title_format

          Which title will be included in export in case of
Chinese

          object

          Values: both, china, taiwan

      --include_alternative_titles (profile or manual)

        For TXT, XML exports, will include (or exclude) alternative title

        in the export file.

        Not included by default

  Display list of target names:

        --show_targets

        --compress           (google only)

        ZIP the output file

        --exclude_not_changed_objects - to exclude from the
export file objects that were not changed, both when manually
exporting or using profiles
```

Command-Line Export Tool Examples

The following are export tool examples:

- `./export.pl --mode=manual --format=TXT --target=AMERICAN_MATHEMATICAL_SOCIETY  --service=getFullTxt`
- `./export.pl --mode=manual --format=ALEPH  --service=getFullTxt`

  `--output_dir=/exlibris/sfx_ver/sfx4_1/<instance>/`

- `./export.pl --mode=manual --format=XML  --service=getFullTxt`

  `---filename=/exlibris/sfx_ver/sfx4_1/<instance>/lieve.xml  --target=NATURE`

  `--target=ERUDIT`
- `./export.pl --mode=manual --format=XML --service=getHolding --institute=InstA`

  `--prefix=OPAC-XML --base_url=http://sfxserver:3210/<instance> --link_text="SFX linking" --inst_info=InstitutionA --target_note="[electronic content]"`
- `./export.pl --mode=manual --format=XML --service=getFullTxt`

  `--base_url http://sfxserver:3210/<instance> --noinclude_servicetype  --include_categories`

  `--external_input=additional.txt --compare=TEST-e-collection.20060613120211`
- `./export.pl --mode=profile --profile=metalib`
- `./export.pl --mode=profile --profile=primo --compare=e-collection.20120321165005`
- `./export.pl --mode=manual --format=KBART --service=getFullTxt`
- The following is an example to create Google Export file with holdings information for three institutes:

  `./export.pl --mode=manual --format=GOOGLE --object_type=JOURNAL --object_type=BOOK --institute=<institute_name_1> --institute=<institute_name_2> --institute=<institute_name_3> --service=getFullTxt`
- `./export.pl --format=XML --mode=manual --compare=e-collection.20110612125533 -exclude_not_changed_objects`

NOTE:

The default location for export output is generally the scratch directory of the instance. However for Google Scholar exports, the file must be placed in the `/export` directory of the instance for proper crawling by Google Scholar. Use the parameter `output_dir` in the tool to specify the `/exlibris/sfx_ver/sfx4_1/<instance_name>/export` (where `sfx4_x` is the name of the SFX installation; the default is `sfx4_1`). For example: `./exlibris/sfx_ver/sfx4_1/sfxlcl41/admin/kbtools/ export.pl --mode=manual --format=GOOGLE --output_dir=/ exlibris/ sfx_ver/sfx4_1/sfxlcl41/export`

NOTES:

- If no `--institute=<institute>` is included, anything active for either default or any of the institutes/groups is exported.

- If one or more `--institute=<institute>` is specified, anything active for the institute(s) or active for DEFAULT or a group the institute is affiliated with and not explicitly inactive for the institute is exported.
- If one or more `--institute=<group>` is specified, anything active for the group(s) is exported.
- If `--institute=DEFAULT` is specified, anything active for DEFAULT is exported.

Command-Line TargetMatcher

The following is a description of the TargetMatcher program:

```
./target_matcher.pl


--file             Name of input file, file should exist in
                   scratch directory.


--target_service   Name of target service
                   format: TARGET_INTERNAL_NAME-SERVICE
                   e.g CAMBRIDGE_UNIVERSITY_PRESS-getFullTxt
                   multiple argument


--key              primary key, valid values: ISSN,LCCN, ISBN


--mode             type of action
                   valid values: report,activation


--target_type      match against group of targets
                   valid values:
                   ACTIVE,INACTIVE,SPECIFIC,ALL,
                 ACTIVE_FULLTXT, INACTIVE_FULLTXT,ALL_FULLTXT
```

Running the TargetMatcher Program Examples

To run the TargetMatcher program in report mode on a file called `test.txt` that exists in the scratch directory and checks the content of `test.txt` against all the active targets in the SFX database, use the following command:

```
./target_matcher.pl --file=test.txt --key=ISSN --mode=report
--target_type=ACTIVE
```

To run the TargetMatcher program in active mode on a file called `test.txt` that exists in the scratch directory and checks the content of `test.txt` against all the active full-text targets in the SFX database, use the following command:

```
./target_matcher.pl --file=test.txt --key=ISBN --
mode=activation --target_type=ACTIVE_FULLTXT
```

To run the TargetMatcher program in active mode on a file called `test.txt` that exists in the scratch directory and checks the content of `test.txt` against the following targets in the SFX database, use the command below:

- AMERICAN_INSTITUTE_OF_PHYSICS_OJPS-getFullTxt
- AMERICAN_CHEMICAL_SOCIETY-getFullTxt

```
./target_matcher.pl --file=test.txt --key=ISSN --
mode=activation
--target_type=SPECIFIC --
target_service=AMERICAN_INSTITUTE_OF_PHYSICS_
OJPS-getFullTxt --target_service=AMERICAN_CHEMICAL_SOCIETY-
getFullTxt
```

Log and Report Files

In the scratch directory of your instance, the following files are created by the program:

- In report mode:
  - `target_matcher.pl.result` – contains a list of created files
  - Export files per target service for which hits were found
  - `<input_file>.ok.err` – contains an analysis of input data (only contains information if input data is incorrect)
  - `target_matcher.pl.log` – a general log file
- In active mode:
  - `target_matcher.pl.result` – contains a list of all targets, target services, and portfolios activated
  - `<input_file>.ok.err` – contains an analysis of input data (only contains information if input data is incorrect)

- `<input_file>.ok.portfolios.out` – a complete file with all portfolios
- `target_matcher.pl.log` – a general log file

Command-Line Category Tool

The command-line Category Tool allows you to export and load category information.

To use this tool:

1. Type `da`.
2. Type `cd category`.
3. Type `./category_export <options>` to start the export or type `./category_load <options>` to start loading.

Category Export

The following is the help screen for the Category Export Tool:

```
category_export [--help] <command> [--input=<file name>]

--help                                 Print this screen

--output=<file name | 'standardoutput'>  File where the data
will be stored or

                                         'standardoutput' to
print to screen

--export_global                       Export global and translated
categories

--export_non_translated               Export global categories for
which no

                                      translated counterpart exists

--export_local_without_assigned  Export local categories
without assignments

--export_local_with_assigned     Export local categories with
assignments

--export_local_with_count        Export local categories with
assignment

                                record count

--export_all                     Export all category
information for list of

                                 objects (using input file)
```

```
--input=<file name>                 File containing data for --
export_all command

--key= (issn|lccn|object_id)
```

The following are examples of Category Export output:

- `/category_export --output=standardoutput --export_global`
- `./category_export --output=../dbs/scratch/test1.txt --export_global`
- `./category_export --output=standardoutput --export_non_translated`
- `./category_export --output=../dbs/scratch/test2.txt --export_non_translated`
- `./category_export --output=standardoutput --export_local_without_assigned`
- `./category_export --output=standardoutput --export_local_with_count`
- `./category_export --output=standardoutput --export_all --input=export9.txt --key=object_id`

Category Load

The following is the help screen for the Category Load Tool:

```
category_load [--help] <command> --input=<file name> [--
replace]

 --help                        Print this screen

 --input=<file name>          File containing the records to
import

 --encoding=<utf8 | latin1>    Encoding of the input file



command:

 --load_translated      Import translated
categories\subcategories from file

 --load_local           Import local categories\subcategories
from file

 --load_global_assign   Import local category assignments to
global

 --load_local_assign    Import local category assignments to
local



options:
 --key=<ISSN_ID|ISBN_ID|LCCN_ID|OBJECT_ID>   Key included in
input file for

                                             adding local
assignments

 --option_local=<add|addAndRemove|update>    Options for
adding                        local


categories\subcategories

 --option_assign=<add|replace>               Options for
adding local


assignments
```

The following are examples of Category Load output:

-
  ```
  ./category_load --load_translated --input=../dbs/scratch/
  test1.txt --encoding=utf8
  ```

-
  ```
  ./category_load --load_local --input=../dbs/scratch/test2.txt
  --encoding=utf8 --option_local=add
  ```

-
  ```
  ./category_load --load_local --input=../dbs/scratch/
  local_cat1.txt --encoding=utf8 --option_local=add
  ```

-
  ```
  ./category_load --load_local --input=../dbs/scratch/test2.txt
  --encoding=utf8 --option_local=addAndRemove
  ```

-
  ```
  ./category_load --load_local_assign --input=../dbs/scratch/
  local_cat_assign4.txt --encoding=utf8 --key=LCCN_ID --
  option_assign=add
  ```

-
  ```
  ./category_load --load_local_assign --input=../dbs/scratch/
  local_cat_assign2.txt --encoding=utf8 --key=ISSN_ID --
  option_assign=add
  ```

-
  ```
  ./category_load --load_global_assign --input=../dbs/scratch/
  test3.txt --encoding=utf8 --key=OBJECT_ID --option_assign=add
  ```

-
  ```
  ./category_load --load_global_assign --input=../dbs/scratch/
  global_cat_assign1.txt --encoding=utf8 --key=ISSN_ID --
  option_local=add
  ```

-
  ```
  ./category_load --load_local_assign --input=../dbs/scratch/
  local_cat_assign3.txt --encoding=utf8 --key=OBJECT_ID  --
  option_assign=add
  ```

-
  ```
  ./category_load --load_local_assign --input=../dbs/scratch/
  local_cat_assign3.txt --encoding=utf8 --key=OBJECT_ID  --
  option_assign=replace
  ```

Command-Line Tool: change_availability.pl

The `change_availability.pl` script is intended for large-scale batch activation and deactivation in the KBManager that involves many targets at once. This is typically useful during the initial setup of SFX or after major changes in your institution's electronic holdings. This script can be used either for full text activations, or for CDI Discovery activations.

To start this tool:

1. Enter `da`.
2. Enter `cd kbtools`.
3. Enter `./change_availability.pl [option]`.

Examples

- `./change_availability.pl --ACTION=ACTIVE --ALL_TARGETS`

  This makes all targets, target services, and portfolios active.

- `./change_availability.pl --ACTION=INACTIVE --ALL_TARGETS`

  This makes all targets, target services, and portfolios inactive.

-
  ```
  ./change_availability.pl --ACTION=ACTIVE --ALL_FREE_TARGETS
  --EXCLUDE_T=AMAZON
  ```

  This activates all targets, target services, and portfolios where target service has a status of `free`, except for the `AMAZON` target.

- `./change_availability.pl`

  ```
  --TARGET_SERVICE=JSTOR_COMPLETE-getFullTxt
  --TARGET_SERVICE=PROJECT_MUSE_STANDARD_COLLECTION-getFullTxt
  --AUTO_ACTIVE=YES --ACTION=ACTIVE
  ```

  This activates the targets in the command, their full-text target services, and full-text portfolios. It also sets the `AutoActive` option to `yes` so that portfolios added in future to these target services are active.

- `./change_availability.pl`

  ```
  --TARGET=JSTOR_COMPLETE --
  TARGET=PROJECT_MUSE_STANDARD_COLLECTION --
  ACTION=SEARCH_ACTIVE
  ```

  This activates the two targets listed in the command for CDI Discovery.

The following is a complete list of options:

```
./change_availability.pl [command]
    commands:
```

```
--ACTION                      ACTIVE/INACTIVE/SEARCH_ACTIVE/
                              SEARCH_INACTIVE - mandatory option.

                              Note: SEARCH_ACTIVE and
                              SEARCH_INACTIVE can only be used for
                              targets that can be activated for CDI
                              search/discovery.
                              It cannot be used in combination with
                              other options related to target
                              services or portfolios.
--AUTO_ACTIVE                 Value can be YES/NO. Sets AUTOACTIVE
                              for target services which were
                              Activated/Deactivated.
--ALL_FREE_TARGETS            Activate/Deactivate ALL targets,
                              target services and portfolios where
                              target service has status 'free'.

                              Should not be combined with options:
                              --ALL_TARGETS, --ALL_TARGET_SERVICES,
                              --TARGET, --TARGET_SERVICE,
                              --PORTFOLIOS
--ALL_TARGETS                 Activate/Deactivate ALL targets,
                              target services and portfolios.

                              Should not be combined with options:
                              --ALL_FREE_TARGETS,
                              --ALL_TARGET_SERVICES, --TARGET,
                              --TARGET_SERVICE, --PORTFOLIOS
--ALL_TARGET_SERVICES         Activate/Deactivate ALL
                              target_services and portfolios.

                              Should not be combined with options:
                              --ALL_FREE_TARGETS,
                              --ALL_TARGET_SERVICES,
                              --TARGET, --TARGET_SERVICE
                              --PORTFOLIOS
--ALL_SERVICE=<service>       Activate/Deactivate ALL targets,
                              target services and portfolios from a
                              specific service type (repeatable
                              option).

                              Should not be combined with options:
                              --TARGET_SERVICE, --PORTFOLIOS
```

--TARGET=<target>	Activate/Deactivate specific target and all target services and portfolios connected that to target (repeatable option). Should not be combined with options: --ALL_FREE_TARGETS, --ALL_TARGET_SERVICES, --TARGET_SERVICE --PORTFOLIOS
--TARGET_SERVICE=<target service>	Activate/Deactivate specific target_service and all targets and portfolios connected to it (repeatable option). Should not be combined with options: --ALL_FREE_TARGETS, --ALL_TARGET_SERVICES, --TARGET --PORTFOLIOS
--PORTFOLIOS=<target service>	Activate/Deactivate only the portfolios of a specific target_service (repeatable option). Should not be combined with options: --ALL_FREE_TARGETS, --ALL_TARGET_SERVICES, --TARGET, --TARGET_SERVICE

The following options can be used in combination with the options above - to further limit the items that will be activated/ deactivated:

--OBJECT_TYPE=<object_type>	Activate/Deactivate only the portfolios linked to object of selected type (repeatable option).
--EXCLUDE_T=<target>	Excludes the target (with all target services and portfolios connected) listed from being activated or deactivated. Repeatable option - cannot be used together with --TARGET and --ALL_TARGET_SERVICES options.
--EXCLUDE_TS=<target_service>	Excludes the target service (and its portfolios) listed from being activated or deactivated. Repeatable option - cannot be used together with --TARGET_SERVICE option.
--EXCLUDE_OP=<target_service>	Excludes the portfolios connected to the target service listed from being activated or deactivated. Repeatable option - cannot be used together with --PORTFOLIOS option.

```
Note: It is possible to use '%' to indicate truncation in the
target internal name.

Example: ./change_availability.pl --TARGET=EBSCO% --
ACTION=INACTIVE will deactivate all targets, target services and
portfolios for targets with internal name starting with EBSCO.
```

Use the percent sign (%) to indicate truncation in the target internal name. For example:

```
./change_availability.pl --TARGET=EBSCO% --ACTION=INACTIVE
```

This deactivates all the targets, target services, and portfolios for targets with an internal name starting with `EBSCO`.

ONIX Import Tool: onix_import.pl

SFX supports the import of serials data in SOH (Serials Online Holdings) 1.0 format. A data file provided in the SOH format can be used to localize or update a local library's SFX configurations.

The ONIX import tool is currently available in command-line form.

Mapping is available for the SOH-formatted files from EBSCO Information Systems. As more providers make files available in the SOH format, applicable mapping files become available in SFX. A library may also create its own local mapping file. Mapping files are placed in the `/exlibris/sfx_ver/ sfx4_1/ <sfx_instance>/config/onix` directory.

Overview (from http://www.editeur.org/onixserials.html)

ONIX for Serials uses the design principles and many of the elements defined in ONIX for Books to communicate information about serial products and subscriptions. The SOH (Serials Online Holdings) is a format for communicating electronic serial holdings details from publication access management systems to user libraries.

The Serials Online Holdings (SOH) format is designed to convey information about online serial resources from suppliers such as hosting services, publication access management services, agents, or publishers to customers in subscribing libraries. Each instance of the format shows the holdings of a particular subscribing organization or library available via, or known to, the sending party.

To run the tool:

1. Place the SOH XML file obtained from the provider in the SFX file system for the instance. A recommended location is `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/dbs/scratch`.

   The command-line ONIX import tool is located in `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/admin/kbtools`.

2. Access the tool by entering the following commands at the UNIX prompt:

   a. `da`

   b. `cd kbtools/`

```
usage:./onix_import.pl [--options] FILE.xml

--vendor=<vendor>   which vendor mapping table to use (will
prompt
        if not supplied)

--report_mode       don't change anything, just report what
would
        happen

--deactivation, --nodeactivation     deactivate all relevant
targets, target
              services and portfolios (enabled by
              default)

--loadlocal, --noloadlocal load local thresholds if onix
value
                             differs from global (enabled by
default)

--list              display a list of available vendors

--log_file=<logfile>

--verbose, --noverbose         default is off

--help

--debug
```

The following are several examples:

```
./onix_import.pl /exlibris/sfx_ver/ sfx4_1/sfxlcl4/dbs/
scratch/

ebsco_onix.20050731.xml --vendor=ebsco



./onix_import.pl /exlibris/sfx_ver/ sfx4_1/sfxlcl41/dbs/
scratch/

ebsco_onix.20050731.xml --vendor=ebsco --noloadlocal
```

```
./onix_import.pl /exlibris/sfx_ver/ sfx4_1/sfxlcl41/dbs/
scratch/

ebsco_onix.20050731.xml --vendor=ebsco --verbose
```

Reports

A report of the load is created in the same directory as the load file using the load file name with the suffix `ok.err`.

There is also a log file created with the name `onix_import.log`.

For example:

```
/exlibris/sfx_ver/ sfx4_1/sfxlcl41/dbs/scratch/
ebsco_onix.20050731.xml.ok.err

/exlibris/sfx_ver/ sfx4_1/sfxlcl41/dbs/scratch/
onix_import.log
```

The report provides a list of the items loaded, and the log file reports on any non-fatal XML errors.

8

User Administration

This section includes:

Overview of User Administration

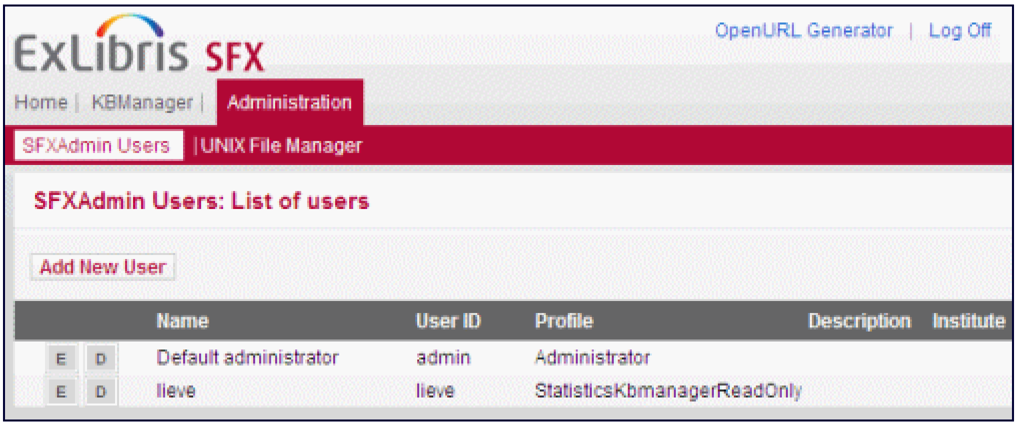
The purpose of the User Administration module is to provide a facility for the SFX system administrators to create and delete users and manage the different access rights of SFX Admin Center users.

Adding a New User/Specifying Access Rights

If you are a system administrator, you can add a new user to the SFX Admin Center and configure permissions for this user.

To add a new user to the SFX Admin Center and configure permissions for this user:

1 Click **SFXAdmin Users** under **Setup & Administration > Administration**. The SFXAdmin Users: List of users screen displays.


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | Administration
SFXAdmin Users |UNIX File Manager
SFXAdmin Users: List of users
Add New User

		Name	User ID	Profile	Description	Institute
E	D	Default administrator	admin	Administrator		
E	D	lieve	lieve	StatisticsKbmanagerReadOnly		

Figure 49: List of Users

2 Click the **Add New User** button.

3 Specify the following information for each user:

- Name
- User ID
- Password – must be at least 6 characters and include at least one number and one letter.
- Profile
- Description (optional)

4 Click **Submit** to save the new user.

The new user's profile is saved to the `USERS` table in the database.

Description of User Profiles

There are predefined profiles that are set in SFX. These include the following:

Table 7. Predefined Profiles

Predefined Profiles	Description
StatisticsOnly	Provides access to Statistics, OpenURL generator, and the Customer Center logon page

Table 7. Predefined Profiles

Predefined Profiles	Description
StatisticsKbmanagerReadOnly	Provides read-only access to KBManager, access to Statistics, OpenURL generator, and the Customer Center logon page
StatisticsKbmanagerWrite	Provides access to KBManager, KBUpdate reports, Statistics, OpenURL generator, and the Customer Center logon page
StatisticsKbmanagerKBTools	Provides access to KBManager, KBUpdate reports, KBTools, the Troubleshooting section, and Statistics
Administrator	Provides access to all sections
AdministratorLight	Provides access to the light administration interface

These predefined profiles are set in a file called `user_permissions.config`. This file is located in the `/config` directory on your SFX server. You can change the access to any modules for any of these profiles by editing the `user_permissions.config` file.

Module Definitions

The `user_permissions.config` file contains definitions for all the predefined profiles listed and provides the ability to define custom profiles. The following table contains a list of SFX module definitions:

Table 8. SFX Module Definitions

Component	Defined In ...
KBManager:	
Sources	`KBMANAGER.WRITE/KBMANAGER.READ`
Targets	`KBMANAGER.WRITE/KBMANAGER.READ`
Search Objects	`KBMANAGER.WRITE/KBMANAGER.READ`
Linking Parameters	`KBMANAGER.WRITE/KBMANAGER.READ`
Institutes	`INSTITUTE.WRITE`
Contribution	`COMMUNITY_ZONE.WRITE`
KBTools:	
DataLoader	`DATALOADER.WRITE`
Export Tool	`EXPORT.WRITE/EXPORT.READ`

Table 8. SFX Module Definitions

Component	Defined In ...
Collection Tool	`COLLECTION.READ`
Statistics:	
Queries	`STATISTICS.READ`
Scheduled Queries List	`STATISTICS.READ`
UStat	
Troubleshooting:	
OpenURL Generator	`CITATIONSFORMS.READ`
Debugging	`DEBUGGING.WRITE`
Contact Support	`SFXHELP.WRITE`
KB Change Requests	`CRM_SI_CREATION.WRITE`
Additional KBTools:	
Look-Up Tool	`LOOKUP.READ`
MARCIt! Tool	`MARC_ENHANCEMENT.READ` (service activated on Ex Libris MARCit! Server via license)
TargetMatcher	`TARGET_MATCHER.WRITE`
Threshold Tool	`THRESHOLD.WRITE`
Category Tool	`CATEGORY.WRITE`
Localization Manager	`LOCALIZATION.WRITE`
Configuration:	
A-Z List	`AZ.WRITE`
Menu Configuration	`CUSTOM.WRITE`
Translations & Display	`MULTILANGUAGE.WRITE`
KBUpdate Reports:	
HTML Reports	`KB_REPORTS.WRITE`
Text Reports	`KB_REPORTS.WRITE`
Administration:	
SFXAdmin Users	`ADMIN.WRITE`
UNIX File Manager	`FM.WRITE`

Table 8. SFX Module Definitions

Component	Defined In ...
Task Manager	`TASK_MANAGER.WRITE`
Services:	
Google Scholar Registration	`GOOGLE_SCHOLAR.WRITE`
bX Configuration	`BX.WRITE`
Discovery:	
Configuration	`PRIMO_CENTRAL.WRITE`
Not available in SFXAdmin Center - Deprecated	`ONIX.WRITE`

Custom Profile

There is another profile (in addition to the predefined profiles) in the `<instance>/config/user_permissions.config` file that you may customize to suit your needs.

`No (N)` is the default access to all of the modules set in the custom profile.

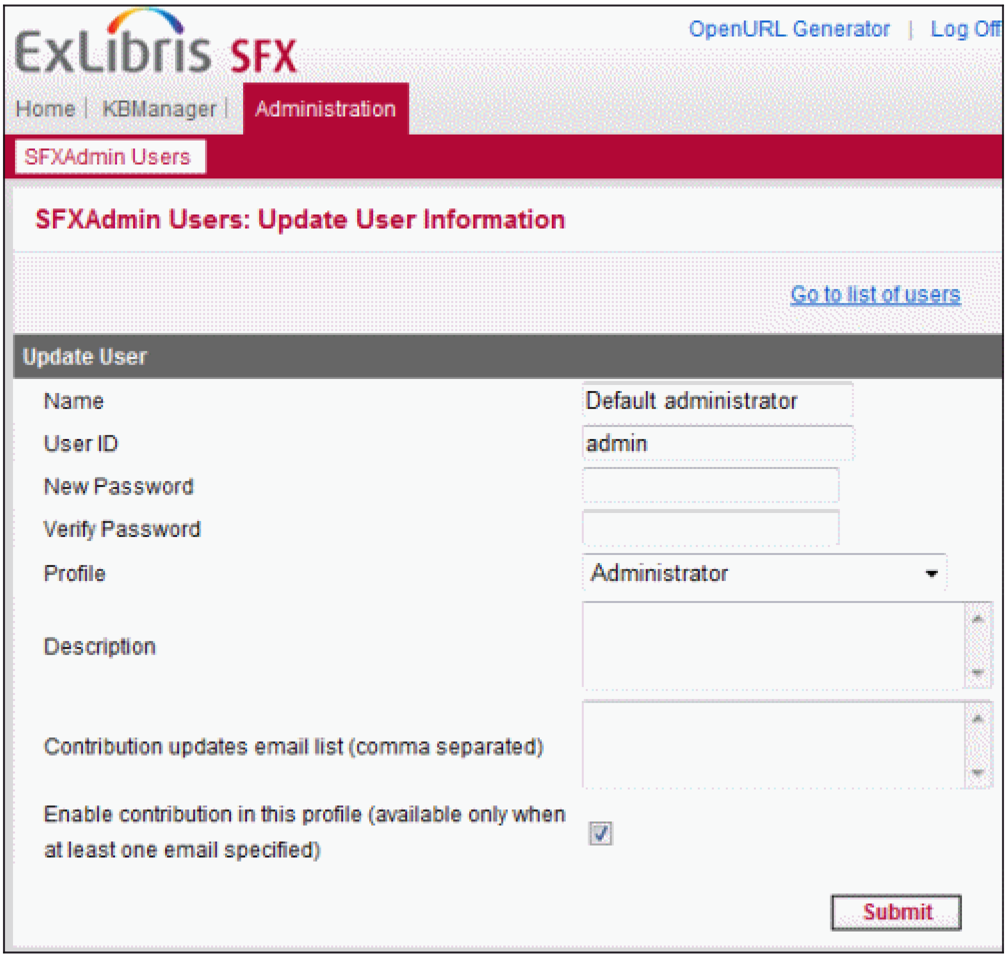
```
Section "Custom"

KBMANAGER.WRITE          "N"
KBMANAGER.READ           "N"
DATALOADER.WRITE         "N"
STATISTICS.READ          "N"
CITATIONSFORMS.READ      "N"
LOOKUP.READ              "N"
EXPORT.READ              "N"
ADMIN.WRITE              "N"
THRESHOLD.WRITE          "N"
TARGET_MATCHER.WRITE     "N"
DEBUGGING.WRITE          "N"
KB_REPORTS.WRITE         "N"
SFXHELP.WRITE            "N"
CATEGORY.WRITE           "N"
AZ.WRITE                 "N"
INSTITUTE.WRITE          "N"
CUSTOM.WRITE             "N"
GOOGLE_SCHOLAR.WRITE     "N"
MSN_SCHOLAR.WRITE        "N"
CRM_SI_CREATION.WRITE    "N"
MULTILANGUAGE.WRITE      "N"
EndSection
```

Edit the "`Custom`" section of the `<instance>/config/user_permissions.config` file to customize the access rights for the custom profile. Replace the default `N` (No) with `Y` (Yes) to grant access to one or more sections/components in the SFX Admin Center.

Configuring Customer Contributions

You can configure SFX to allow customer contribution for a specific profile. When editing a profile, the following is displayed:


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | Administration
SFXAdmin Users
Go to list of users
Submit

SFXAdmin Users: Update User Information

Update User

Name	Default administrator
User ID	admin
New Password	
Verify Password	
Profile	Administrator
Description	
Contribution updates email list (comma separated)	
Enable contribution in this profile (available only when at least one email specified)	☑

Figure 50: Edit User Profile

To configure customer contribution, enter an e-mail address in the **Contribution updates email list** field to receive reports of contribution updates and select the **Enable contribution in this profile** checkbox.

For more information concerning customer contributions, see the **Customer Contributions** section of the *SFX General User's Guide*.

9

Troubleshooting and Debugging

This section includes:

NOTE:

For general information regarding troubleshooting and debugging, refer to the *SFX General User's Guide*.

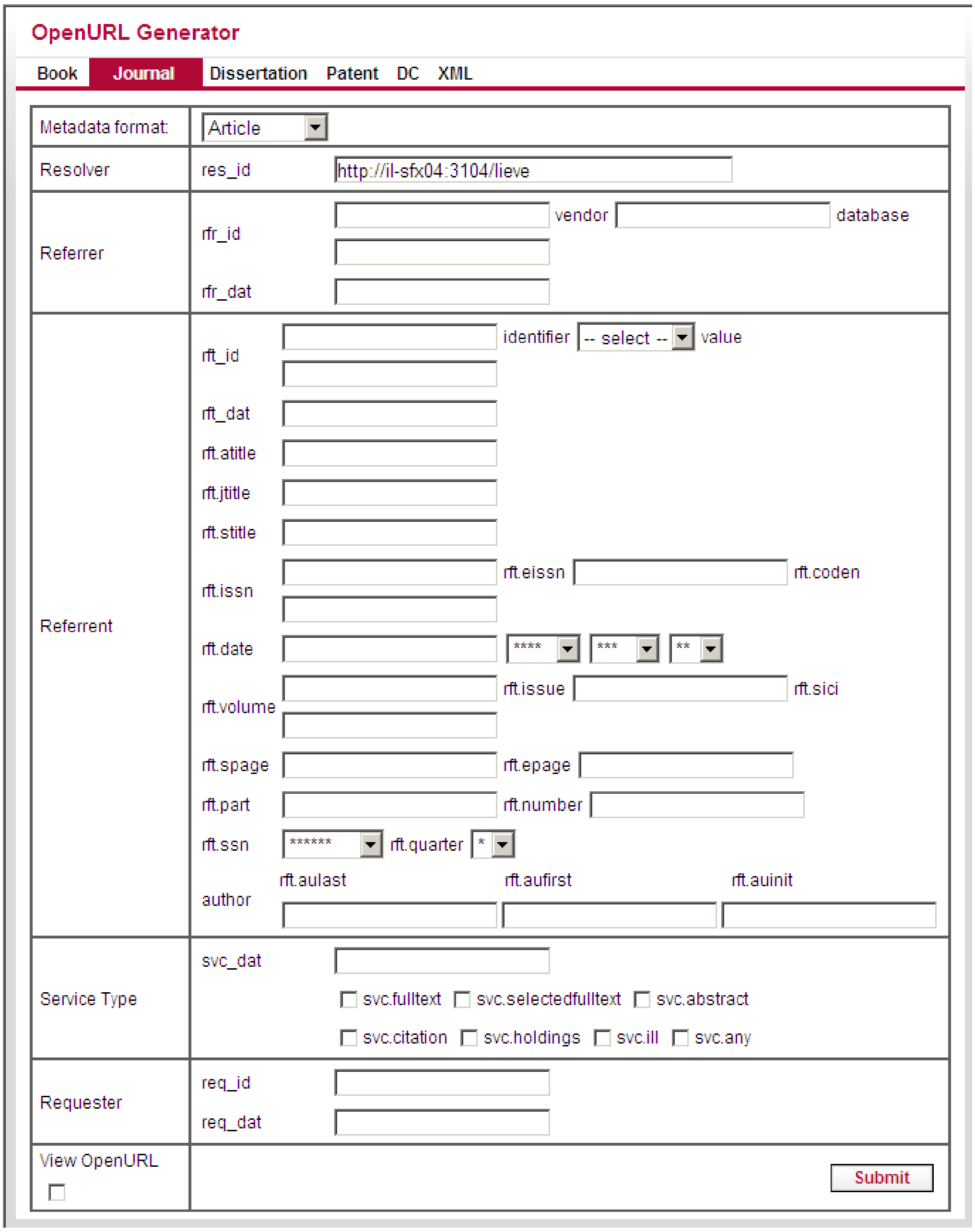
OpenURL Generator

The OpenURL generator is a program that allows you to create an OpenURL based on metadata that you enter. The Web form that the OpenURL uses is specific for each instance. You can access the form by going to http://server_name:port/instance_name/cgi/core/openurl-generator.cgi. The form contains fields that ensure proper URL syntax.

You can use the OpenURL generator to:

- Test the data in the SFX database
- Check TARGET_PARSERS
- Access the SFX database directly and simulate SFX requests without using an A&I database or full-text repository as a source
- Allow end users or librarians to request services for citations they have in hand

The OpenURL generator uses the `DEFAULT SOURCE` in the SFX database.



OpenURL Generator

Book | Journal | Dissertation | Patent | DC | XML

Metadata format:	Article ▾
Resolver	res_id http://il-sfx04:3104/lieve
Referrer	rfr_id vendor database rfr_dat
Referent	rft_id identifier -- select -- ▾ value rft_dat rft.atitle rft.jtitle rft.stitle rft.issn rft.eissn rft.coden rft.date **** ▾ *** ▾ ** ▾ rft.volume rft.issue rft.sici rft.spage rft.epage rft.part rft.number rft.ssn ****** ▾ rft.quarter * ▾ author rft.aulast rft.aufirst rft.auinit
Service Type	svc_dat □ svc.fulltext □ svc.selectedfulltext □ svc.abstract □ svc.citation □ svc.holdings □ svc.ill □ svc.any
Requester	req_id req_dat
View OpenURL □	Submit

Figure 51: OpenURL Generator

Using the OpenURL Generator to Test Your SFX Database

When you configure the SFX database or make changes to it, it is important to test the results of your work. The OpenURL generator enables you to create an SFX menu with metadata you provide.

Using the OpenURL generator, you can:

- Check to see which services display on the SFX menu for a particular journal
- Review the order of the services
- Check to see whether the thresholds you defined are working properly

OpenURL Example: Activating a New Journal

You have just activated a new Wiley journal because the library has recently acquired an online subscription to it. You used KBManager to activate the object portfolio for the journal.

To make sure that a full-text service for this journal shows up in the SFX menu:

1 On the OpenURL generator Web page, enter the ISSN of the journal you just activated.

You can fill in this information in the **Referrent** section of the OpenURL Generator.

2 Enter the year information.

You can use a randomly chosen year. You can also use the date threshold information in the KBManager.

3 Select the **View OpenURL** check box.

4 Click **Submit**.

5 Review the OpenURL that is to be created.

6 Click to access this OpenURL.

The SFX menu for the journal displays, taking into account the year, volume, and issue information you provided. The full-text link to the Wiley journal should display in the SFX menu.

7 Click the button for the Wiley full-text service to check if the link to the Wiley journal actually works.

You can test the date threshold for this particular journal by going back to the OpenURL generator and entering the same ISSN with more information, such as year, volume, and issue.

To test if a threshold works, you can enter a year value that is lower than the threshold value. For example, if the threshold in the KBManager is 1997, you can enter 1996). With this test, the full-text link to Wiley should not display in the SFX menu.

OpenURL Example: Checking All Services for a Particular Journal

To check all services for a particular journal:

1. In KBManager, go to the list of object portfolios for one particular journal.
2. Click the **SFX** button.
3. When the SFX menu displays for the journal, compare the list of services in the SFX menu with the list of services you find in the KBManager list of object portfolios.

XML Tab of the OpenURL Generator

The XML tab of the OpenURL Generator allows the library to submit simple XML to the SFX server in order to retrieve either an SFX menu or XML output. The library can use the XML output for its own purposes (write programs to parse information for its own use.)

XML Input

Given the following citation, see the corresponding OpenURL below:

Temperature dependence of Brillouin frequency, power, and bandwidth in panda, bow-tie, and tiger polarization-maintaining fibers / Qinrong Yu

Optics letters 2004 29: 17-18

```
http://10.1.1.67:9003/sfx_local?ctx_tim=2004-10-
20T13%3A22%3A7CDT&rft.volume=29&rft.issn=0146-
9592&rft.genre=article&ctx_ver=Z39.88-
2004&svc_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Asch_svc&rft.issue=1&url_ctx
_fmt=10_1&rft.aufirst=Qinrong&rft.atitle=Temperature+dependence+of+Brillouin+freque
ncy%2C+power%2C+and+bandwidth+in+panda%2C+bow-
tie%2C+and+tiger+polarization-
maintaining+fibers.&rft_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Ajournal&rft.da
te=2004&ctx_id=10_1&url_ver=Z39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-
8&rft.aulast=Yu&res_id=http%3A%2F%2F10.1.1.67%3A9003%2Fsfx_local&rft.epage=18&
rft.spage=17
```

The corresponding OpenURL in an XML format is:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ctx:context-objects xmlns:ctx="info:ofi/fmt:xml:xsd:ctx"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="info:ofi/fmt:xml:xsd:ctx http://
www.openurl.info/registry/docs/info:ofi/fmt:xml:xsd:ctx">
    <ctx:context-object timestamp="2004-01-16T12:13:00Z"
version="Z39.88-2004" identifier="123">
        <ctx:referent>
            <ctx:metadata-by-val>
                <ctx:format>info:ofi/fmt:xml:xsd:journal</
ctx:format>
                <ctx:metadata>
                    <rft:journal xmlns:rft="info:ofi/
fmt:xml:xsd:journal" xsi:schemaLocation="info:ofi/
fmt:xml:xsd:journal http://www.openurl.info/registry/docs/info:ofi/
fmt:xml:xsd:journal">
                        <rft:author rank="1">
                            <rft:aulast>Yu</rft:aulast>
                            <rft:aufirst>Qinrong</rft:aufirst>
                        </rft:author>
                        <rft:atitle>Temperature dependence of
Brillouin frequency, power, and bandwidth in panda, bow-tie, and
tiger polarization-maintaining fibers.</rft:atitle>
                        <rft:title>Opt Lett</rft:title>
                        <rft:date>2004-12-31</rft:date>
                        <rft:volume>29</rft:volume>
                        <rft:issue>1</rft:issue>
                        <rft:pages>17/18</rft:pages>
                        <rft:spage>17</rft:spage>
                        <rft:epage>9</rft:epage>
                    </rft:journal>
                </ctx:metadata>
            </ctx:metadata-by-val>
            <ctx:metadata-by-ref>
            </ctx:metadata-by-ref>
        </ctx:referent>
        <ctx:referrer>
        </ctx:referrer>
    </ctx:context-object>
</ctx:context-objects>
```

XML Output

The response sent by SFX to the application that initiated the request is a response in an XML format that contains:

- Metadata received by the SFX server
- SFX services, if they exist

For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <sfx_menu>
  <ctx_obj_set><ctx_object_1><perldata> <hash> <item
key="rft.issn">0146-9592</item> <item key="ctx_ver">Z39.88-2004</
item> <item key="rft.year">2004</item> <item key="rft.pages">17/
18</item> <item key="rft.eissn">1539-4794</item> <item
key="@rfr_id"> <item key="@rft_id"> <array> </array> </item> <item
key="rft.atitle">Temperature dependence of Brillouin frequency,
power, and bandwidth in panda, bow-tie, and tiger polarization-
maintaining fibers.</item> <item
key="sfx.response_type">simplexml</item> <item
key="rft.object_type">JOURNAL</item> <item
key="rft.issue_start">1</item> <item key="@rft.abbrev"> <array>
<item key="0">OPT LETT</item> <item key="1">OPTICS LETT</item> </
array> </item> <item key="rft.object_id">954925474435</item> <item
key="rft.volume_start">29</item> <item key="rft.volume">29</item>
<item key="sfx.request_id">1001000000000668</item> <item
key="rft.month">12</item> <item key="rft.genre">journal</item>
<item key="req.user_group"></item> <item key="rft.issue">1</item>
<item key="req.faculty"></item> <item key="sfx.openurl">http://
<OpenURLValuefrom example
above>;sfx.response_type=simplexml;__submit=Submit</item> <item
key="req.language">English</item> <item key="rft.title">Optics
letters</item> <item key="@rfe_id"> <array> </array> </item> <item
key="@rft.aufirst"> <array> <item key="0">Qinrong</item> </array>
</item> <item key="@rft.aulast"> <array> <item key="0">Yu</item> </
array> </item> <item key="rft.day">31</item> <item
key="sfx.sourcename">DEFAULT</item> <item
key="rft.coden">OPLEDP</item> <item key="rft.jtitle">Optics
letters</item> <item key="rft_val_fmt">journal</item> <item
key="rft.date">2004-12-31</item> <item key="rft.epage">18</item>
<item key="rft.spage">17</item> </hash> </perldata> </
ctx_object_1></ctx_obj_set>
- <targets>
- <target>
  <target_name>LOCAL_CATALOGUE_ENDEAVOR_VOYAGER</target_name>
  <target_public_name>Library Catalog (UNISTAR)</
```

```
target_public_name>
  <target_service_id>110986400596330</target_service_id>
  <service_type>getHolding</service_type>
  <parser>ENDEAVOR::VOYAGER</parser>
  <parse_param>url=</parse_param>
  <proxy>no</proxy>
  <crossref>no</crossref>
  <note />
  <authentication />
  <char_set />
  <displayer />
  <target_url />
  </target>
  <authentication />
  <char_set>UTF8</char_set>
  <displayer />
  <target_url />
  </target>
  </targets>
  </sfx_menu>
```

Debugging Tool

The debugging tool allows the SFX administrator to determine the source of errors that occur in the SFX flow. Debugging can be used in the:

- SFX menu
- Target window

The debugging information can be restricted to one IP address or a range of IP addresses. This prevents end users from seeing the debugging information in a live/production SFX environment.

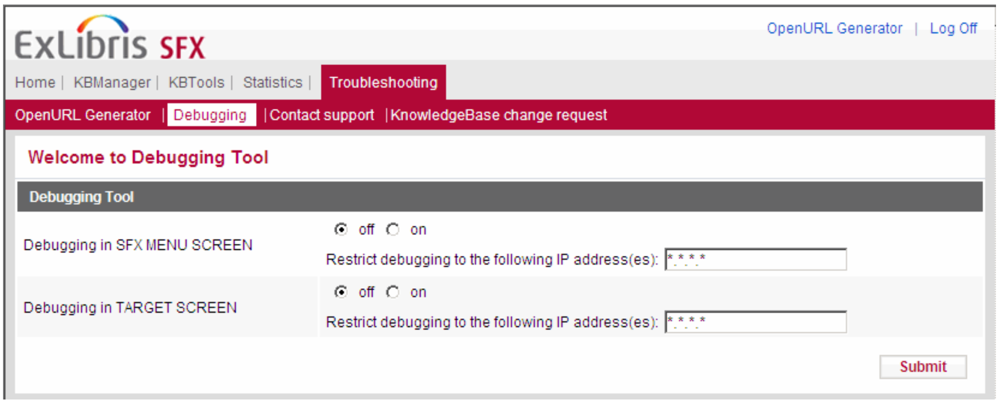
See the following examples:

```
No IP restriction:         *.*.*.*

One IP address:            123.12.151.23

IP range:                  123.12.151.*

IP range:                  123.12.151.23-25
```


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | KBTools | Statistics | Troubleshooting
OpenURL Generator | Debugging | Contact support | KnowledgeBase change request

Welcome to Debugging Tool

Debugging Tool

Debugging in SFX MENU SCREEN ● off ○ on
Restrict debugging to the following IP address(es): *.*.*.*

Debugging in TARGET SCREEN ● off ○ on
Restrict debugging to the following IP address(es): *.*.*.*

Figure 52: Debugging Tool

Checking the SFX Menu

Turning on debugging in the SFX menu allows the SFX administrator to see if the information being sent through the SFX server passes or fails the thresholds set in the SFX database. This helps to determine why an expected target service is not displaying in the SFX menu. The following is an example of HTML source code for the SFX menu:

```
 'ctx_enc' => 'UTF-8',
                 'ctx_id' => '10_1',
                 'rft.year' => '2003',
                 'sfx.openurl' => 'http://10.1.2.225:3210/
cs_3?ctx_tim=2004-10-20T10%3A57%3A8EDT;rft.issn=0000-
0019;rft.genre=journal;rft_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Aj
ournal;rft.date=2003;ctx_id=10_1;ctx_enc=info%3Aofi%2Fenc%3AUTF-
8;ctx_ver=Z39.88-2004;url_ver=Z39.88-
2004;svc_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Asch_svc;res_id=http
%3A%2F%2F10.1.2.225%3A3210%2Fcs_3;url_ctx_fmt=10_1',
-------------- SFX Metadata Augmentation-------------------

ContextObject::Engine::OpenURL::augment_object_attributes: Setting
'rft.title' in ctx obj to: The publishers weekly
ContextObject::Engine::OpenURL::augment_object_attributes: From the
db: $VAR1 = {
          'TITLE_ABBREV' => 'PUBLISHERS WEEKLY'
        };

ContextObject::Engine::OpenURL::augment_object_attributes: Setting
'@rft.abbrev' in ctx obj to: PUBLISHERS WEEKLY
ContextObject::Engine::OpenURL::augment_object_attributes: From the
db: $VAR1 = {
          'OBJECT_TYPE' => 'JOURNAL'
        };
```

```
   -------------------Threshold Checking------------------


'target_internal_name' => 'GALEGROUP_INFOTRAC_EXP_ACAD_ASAP',
                   'ss_threshold_local' => undef,
                   'target_service_id' => '110997392153001',
                   'op_threshold_global' => '$obj-
>parsedDate(">=",1987,undef,undef)',
                   'ts_threshold_global' => '',
```

The above example of HTML source code for the SFX menu shows the `ContextObject`. It also includes references to the augmentation process and threshold checking (at target, target service, or object portfolio levels).

NOTE:

The SFX menu debugging information is normally hidden in comments in the HTML source code of the SFX menu. However if direct linking is active, the debugging information displays directly in the window that opens. Therefore, if both direct linking and SFX menu debugging are turned on, the SFX menu debugging should be restricted by IP address to prevent it from displaying to end users.

Checking Target Parsers

Turning on debugging in the target window allows the SFX administrator to see if the appropriate SFX target parser is being used. It also shows the resulting URL that is created by this target parser, which helps determine if the parser is functioning properly.

For example:

```
------------------Context Object section--------------------

SFXResolver::Engine::restore_request: Final version of ctx obj:
$VAR1 = bless( {
                 'rft.issn' => '0000-0019',
                 'ctx_tim' => '2004-10-20T11:3:34CDT',
                 'rft.genre' => 'journal',
                 'svc_val_fmt' => 'info:ofi/fmt:kev:mtx:sch_svc',
                 'ctx_ver' => 'Z39.88-2004',
                 'rft.year' => '2003',
                 'sfx.openurl' => 'http://10.1.1.67:9003/
sfx_local?ctx_tim=2004-10-20T11%3A3%3A34CDT;rft.issn=0000-
0019;rft.genre=journal;rft_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Aj
ournal;rft.date=2003;ctx_id=10_1;ctx_enc=info%3Aofi%2Fenc%3AUTF-
8;ctx_ver=Z39.88-2004;url_ver=Z39.88-
2004;svc_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amtx%3Asch_svc;res_id=http
%3A%2F%2F10.1.1.67%3A9003%2Fsfx_local;url_ctx_fmt=10_1',
                 'rft.title' => 'The publishers weekly',
                 'url_ctx_fmt' => '10_1',
                 '@rfe_id' => [],
                 '@rft_id' => [],
                 'rft.coden' => 'PWEEAD',
                 'sfx.sourcename' => 'DEFAULT',
                 'rft.jtitle' => 'The publishers weekly',
                 'rft_val_fmt' => 'journal',
                 'rft.date' => '2003',
                 'sfx.sid' => 'DEFAULT',
                 'ctx_id' => '10_1',
                 'ctx_enc' => 'UTF-8',
                 'url_ver' => 'Z39.88-2004',
                 '@req.institutes' => [
                                        'Medical'
                                      ],
  'rft.object_type' => 'JOURNAL',
                 '@rft.abbrev' => [
                                    'PUBLISHERS WEEKLY'
                                  ],
                 '@res_id' => [
                                'http://10.1.1.67:9003/sfx_local'
                              ],
                 'rft.object_id' => '954921332001'
               }, 'ContextObject::Generic' );

-------------- Target Parser Retrival Section-----------------

SFXResolver::Engine::revive_target_service: Context service
found:$VAR1 = bless( {
                 'tmp_ctx_obj_id' => 1,
                 'ss_threshold_global' => undef,
                 'parser' => 'Bulk::BULK',
                 'authentication' => '',
                 'TS_AVAILABILITY' => 'ACTIVE',
                 'op_status' => undef,
                 'target_public_name' => 'Free E- Journals',

-----------------Target URL Retrieval------------------------

SFXResolver::Engine::redirect_to_target_url: Target URL: 'http://
www.publishersweekly.com/'.
```

The above example shows the result of clicking the **Full Text Available From Free e-Journals** button in the SFX menu while debugging is activated in the target window. These sections of the debugging mode show that the information passed through the `ContextObject`, activated the free e-journal target parser, and returned the target URL for the chosen article.

If the debugging screen displays an error message that no parser was found:

1. Check the parser field in the target service using the KBManager. If this field is empty, this is the cause of the problem.

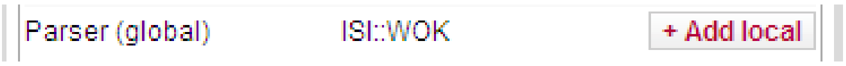


Figure 53: ISI::WOS

2. If the field is not empty, check the target parser `.pm` file on your SFX server. The name of the target parser and the name of the directory that stores this parser should correspond to the data in the parser field of the target service.

   In the above example, the directory name should be `ISI` and the parser name should be `WOS.pm`.

3. Check the first line of the target parser. This line should contain the exact directory name and file name of the target parser in the following format:

```
package TargetParser::ISI::WOS;
```

Error Messages in the Debugging Tool

If the error message **Could not change debugging mode of...**, displays in the debugging tool:

1. Telnet to the SFX server and log on to the SFX instance for which you want to activate debugging.
2. Enter `cn` to go to the configuration directory.
3. Enter the following:

```
chmod 775 ctx_object.config

chmod 775 sfxresolver.config
```

Configuring Contact Support

You can configure the automated sign-in for SalesForce by providing your institution's SalesForce CRM credentials in your SFX instance.

This enables the following features:

- The Contact Support link located in the SFX Admin Center main menu under Data Management > Troubleshooting
- The SFX Contribution feature in KBManager

To configure the automated sign-in for the Contact Support link:

1 Log on to the SFX server as an instance user (for example: `sfxlcl41`).

2 Start the Server Admin Utility program:

```
da
./server_admin_util
```

3 Select **3 Change Passwords** from the main menu.

4 Select **5 Generate encrypted password for use in configuration files**.

5 Enter your current USERNAME (not password) for the Ex Libris Customer Center / SalesForce.

For example:

```
Generate encrypted password for use in configuration files
Please select [1-5,q,m]: 5
Please enter password to encrypt:****** <password, no output
visible>
[Enter]
Confirm password:****** <password, no output visible>
[Enter]
Your encrypted password is: "378686c673535303236303"*
```

6 Edit the CRM configuration file:

```
cn
vi sfx_crm.config
```

7 Copy and paste the encrypted password into the Contacts section of `sfx_crm.config` as the `crm_encrypted_pass` value:

For example:

```
#contact_name "email" "crm_user_name" "crm_encrypted_pass"
Customer_Lastname "name@institution.edu" "name@sf.com"
"378686c673535303236303"
```

8 Enter value for `crm_user_name` in the Contacts section of `sfx_crm.config`. Use the contact's entire SalesForce email address (as shown on the My Profile page in SalesForce).

9 Enter the SaleForce URL in the `server_url` section of the `sfx_crm.config` file:

```
server_url "http://exlibris.force.com/antwoord/services/
apexrest/xmlservice/"
```

10 Save and exit the `sfx_crm.config` file:

```
:wq
```

NOTE:
Separate all entries in the Contacts section of `sfx_crm.config` by a tab.

10

SFX Statistics Module

This section includes:

NOTE:

For general information regarding the SFX Statistics module, refer to the *SFX General User's Guide*.

Stat Tables

The log information is stored in a number of tables that are part of the SFX database. Two online and two offline tables are used.

- The online tables (`STAT_OBJECT` and `STAT_TARGET_SERVICE`) are used during the SFX request time. Details regarding all the interactions with the SFX end user are stored in these tables.
- The offline tables (`STAT_OBJECT_OFFLINE` and `STAT_TARGET_SERVICE_OFFLINE`) are used by the statistics queries that the librarian or SFX administrator runs.

Data is moved from the online table to the offline table using either a command-line tool or a cron job.

Why Stat Queries Are Run on the Offline Statistics Tables

The runtime for statistics queries varies depending on the amount of data.

IMPORTANT:

The database engine locks the table in order to keep data integrity when reading from tables. Therefore, if the SFX administrator runs the queries directly on the online table, SFX end users may have long wait periods for their SFX menu to display. For this reason, offline tables are used in the Statistics module.

You can and may want to remove information from the offline stat tables and archive the data into `TXT` files when the offline tables contain a large amount of statistical information.

The diagram below illustrates a complete flow of the statistics information on the SFX server:

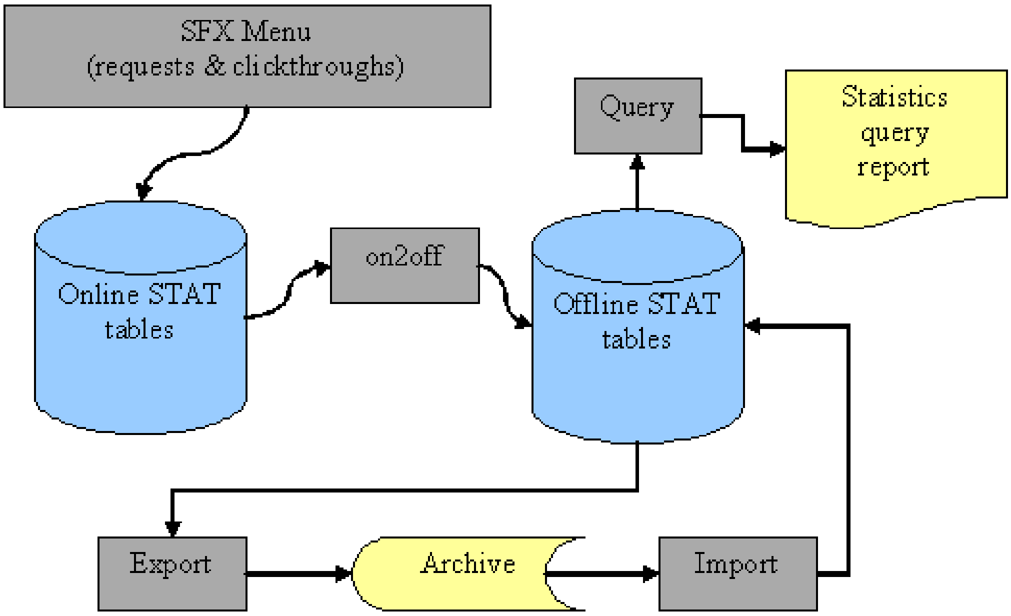
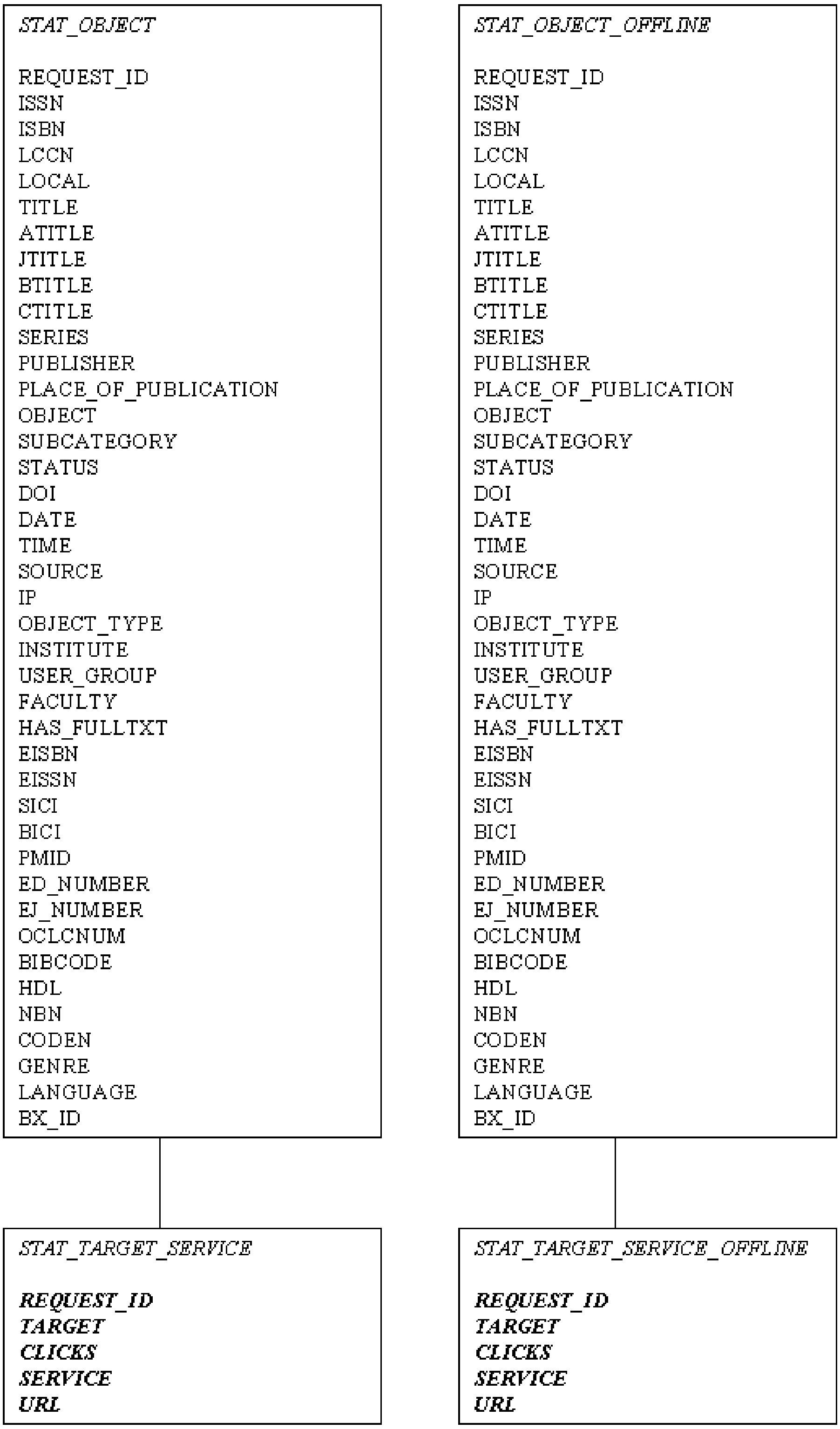


Figure 54: Flow of Statistics Information

Note how the arrows point in the direction of the statistical data flow. The gray squares indicate the processes that move the data, the blue cylinders represent the online and offline tables, and the yellow figures indicate the statistical data stored in a static medium such as a text or an HTML file.

The following illustration describes the online and offline tables:



STAT_OBJECT

REQUEST_ID
ISSN
ISBN
LCCN
LOCAL
TITLE
ATITLE
JTITLE
BTITLE
CTITLE
SERIES
PUBLISHER
PLACE_OF_PUBLICATION
OBJECT
SUBCATEGORY
STATUS
DOI
DATE
TIME
SOURCE
IP
OBJECT_TYPE
INSTITUTE
USER_GROUP
FACULTY
HAS_FULLTXT
EISBN
EISSN
SICI
BICI
PMID
ED_NUMBER
EJ_NUMBER
OCLCNUM
BIBCODE
HDL
NBN
CODEN
GENRE
LANGUAGE
BX_ID

STAT_OBJECT_OFFLINE

REQUEST_ID
ISSN
ISBN
LCCN
LOCAL
TITLE
ATITLE
JTITLE
BTITLE
CTITLE
SERIES
PUBLISHER
PLACE_OF_PUBLICATION
OBJECT
SUBCATEGORY
STATUS
DOI
DATE
TIME
SOURCE
IP
OBJECT_TYPE
INSTITUTE
USER_GROUP
FACULTY
HAS_FULLTXT
EISBN
EISSN
SICI
BICI
PMID
ED_NUMBER
EJ_NUMBER
OCLCNUM
BIBCODE
HDL
NBN
CODEN
GENRE
LANGUAGE
BX_ID

STAT_TARGET_SERVICE

REQUEST_ID
TARGET
CLICKS
SERVICE
URL

STAT_TARGET_SERVICE_OFFLINE

REQUEST_ID
TARGET
CLICKS
SERVICE
URL

Guidelines for SFX Statistics Maintenance

As with most software, the SFX Statistics module should be monitored and maintained. The following major tasks should be considered when maintaining your SFX statistical data:

- Moving data from the online to offline tables
- Running your queries
- Archiving the offline data

The following are recommendations concerning these tasks:

- Moving data from the online to the offline tables

  As described in **Stat Tables** on page **149**, the statistics queries are always run on the offline tables. You should move data from the online tables to the offline tables as often as possible.

  Note that the live/production SFX menu writes to the online tables when library end users click the SFX menu items. Therefore, if you want to store all possible interaction data for a given instance, you must move from the online to offline process during a time when there are no users connected to that particular instance (if possible). In most cases, a good time to move data from online to offline is as early as possible in the morning.

- Running your queries

  Running statistics queries irregularly can be a time-consuming task if the offline table is not maintained correctly. The SFX Apache Web server is set by default to time out after 300 seconds. As a result, the display of queries you submit via the Statistics module in the SFX Admin Center may be affected.

  For example, the average SFX Apache server times out after a query involving approximately 1 million records. The results of such a query are not viewable via the SFX Admin Center. This estimate may vary depending on your hardware and system load. It is therefore important that you run your queries on a regular basis and that you clean out the offline table once the data has been analyzed. If you experience timeouts that cause queries involving large amounts of data to not result in an HTML display of results via the SFX Admin Center, you may want to schedule the queries to receive the results via e-mail and archive data to reduce the burden on your offline table.

- Archiving the offline data

  It is recommended that you archive statistics data from the offline tables to prevent poor performance of your queries. The recommended practice is to not store more than 1 million records in the offline tables.

The following table contains guidelines—based on the overall activity of your SFX installation—as to how often to move data from the online table to the offline table, run your queries, and archive your offline data.

Table 9. SFX Guidelines

SFX Requests/ Week	Online to Offline	Run Queries	Archive
Less than 1,500	yearly	monthly/per semester/yearly	yearly
1,501 - 20,000	yearly	monthly/per semester	monthly/yearly
20,001 - 70,000	twice a year	weekly/monthly	monthly
More than 70,000	a few times a year/ monthly	weekly/monthly	twice a month

To help ensure the smooth functioning of the SFX Statistics module:

1. Set up a scheduled job (cron job) to move the online statistics data to the offline table every day/week/month depending on the usage load of your SFX server.
2. Run statistics every week/month using the SFX Admin Center Web interface or the scheduled query options.
3. After running the statistical queries, move the data stored in the offline table to an archive file and save this archive file to a location outside the server.
4. Choose meaningful names for the archive files in order to easily retrieve them in case they need to be reloaded into the offline table for further analysis.
5. Make sure, before reloading old archived information into the offline table, that all the information in the offline table is archived and well-labeled. It is recommended that you empty the offline table before reloading old archived information to ensure that the results of your queries are accurate.

Troubleshooting SFX Statistics

The following problem/solution describes a possible e-mail query/error message issue.

- **Problem** – E-Mail queries are not received and/or the browser displays an error message when the e-mail option is chosen from the standard query display.

- **Possible Solution** – Check the SMTP configuration on the server.

  The SFX Statistics module requires that your SFX server's SMTP (outgoing mail server) be configured correctly and running in order to send statistics query results via e-mail.

  If, for some reason, the e-mail fails before sending, you may find the query's generated file under the `/exlibris/sfx_ver/sfx4_1/tmp` directory, with the name of the file in the following format depending on the output that was chosen:

  ```
  <instance>_<querynumber>_<yearmonthday>_<time>.html
  ```

  or

  ```
  <instance>_<querynumber>_<yearmonthday>_<time>.txt
  ```

11

UNIX File Manager

This section includes:

IMPORTANT:

For security reasons, the use of the UNIX file manager is not supported by Ex Libris if SSL is not configured and in use for the particular instance. For more information about configuring SSL, consult the *SFX System Administration Guide*.

UNIX File Manager Description

The UNIX file manager allows you to access the UNIX files on the SFX server from the SFX Admin Center interface for a particular instance.

The tool allows you to:

- View, edit, delete, copy, and rename existing files
- Add new files and directories in the `/exlibris/sfx_ver/sfx4_1/<instance>` location on the SFX server and all its subdirectories.

You can access the UNIX file manager from the **Setup & Administration > Administration** section of the SFX Admin Center.

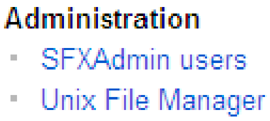


Figure 55: AdministrationTools

Different icons are used to distinguish between the following:

- Directory

- Text file

- Non-text file

The permission of the directories and files is displayed when moving your mouse over the file or directory name (as tooltips).

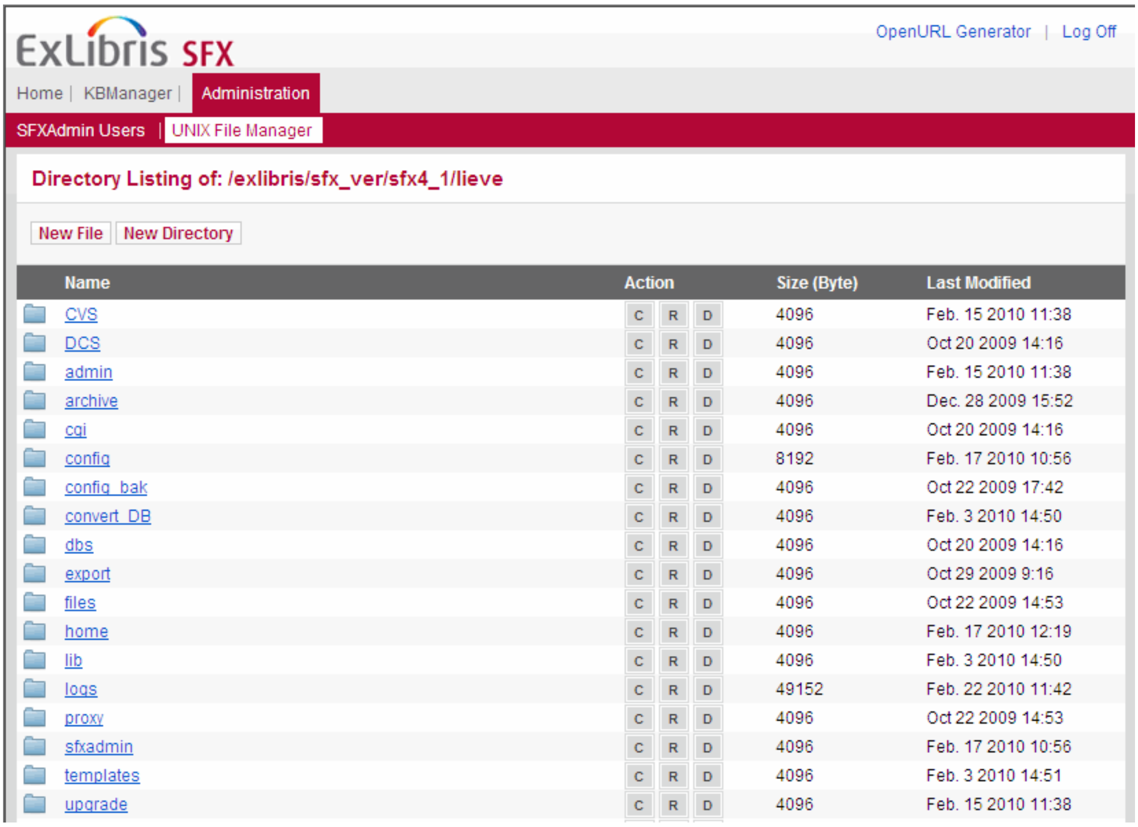
For each file, the file size and last modified time are listed.

When a specific file name is clicked, a pop-up window allows the user to edit the file. The file can be saved after modification if it is writable. If the file is read-only, a warning is provided and all the modifications are discarded.

When a specific directory name is clicked, the user is presented with the files in this directory. To go back to the parent directory, click the **Parent Directory** link.

New files and directories can be created in the current directory by clicking the **New File** or **New Directory** buttons.

After certain files are saved, you receive the following warning message: **Changes might not take effect until the httpd server is restarted**.


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | Administration
SFXAdmin Users | UNIX File Manager
New File
New Directory

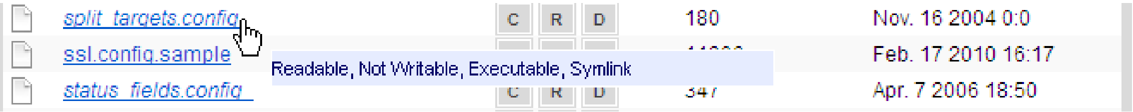
Directory Listing of: /exlibris/sfx_ver/sfx4_1/lieve

Name	Action	Size (Byte)	Last Modified
CVS	C R D	4096	Feb. 15 2010 11:38
DCS	C R D	4096	Oct 20 2009 14:16
admin	C R D	4096	Feb. 15 2010 11:38
archive	C R D	4096	Dec. 28 2009 15:52
cgi	C R D	4096	Oct 20 2009 14:16
config	C R D	8192	Feb. 17 2010 10:56
config_bak	C R D	4096	Oct 22 2009 17:42
convert_DB	C R D	4096	Feb. 3 2010 14:50
dbs	C R D	4096	Oct 20 2009 14:16
export	C R D	4096	Oct 29 2009 9:16
files	C R D	4096	Oct 22 2009 14:53
home	C R D	4096	Feb. 17 2010 12:19
lib	C R D	4096	Feb. 3 2010 14:50
logs	C R D	49152	Feb. 22 2010 11:42
proxy	C R D	4096	Oct 22 2009 14:53
sfxadmin	C R D	4096	Feb. 17 2010 10:56
templates	C R D	4096	Feb. 3 2010 14:51
upgrade	C R D	4096	Feb. 15 2010 11:38

Figure 56: SFX File Manager

Some files are not editable by the UNIX file manager because they are symbolically linked to files in `sfxglb41`.

In the example below, the `sort_services.config_` file is not editable because the file is symbolically linked.



split_targets.config	C R D	180	Nov. 16 2004 0:0
ssl.config.sample	[illegible]	[illegible]	Feb. 17 2010 16:17
status_fields.config_	C R D	347	Apr. 7 2006 18:50

Readable, Not Writable, Executable, Symlink

Figure 57: Non-editable File

To edit a non-editable file:

1 Click the **Copy** button to make a local copy of the file (with a different name), and make the necessary changes.

2 Click the **Delete** button to remove the original symbolically linked file.

3 Click the **Rename** button to give the local file the same name as the original file.

UNIX File Manager Setup

You can set up access to the UNIX file manager.

To set up access to the UNIX file manager:

1. Open the following file that provides access to the UNIX file manager: `/exlibris/sfx_ver/sfx4_1/<instance>/config/user_permissions.config`

2. In this file, change the value for your profile to “`y`” for `FM.WRITE`. See the following example:

```
Section "Administrator"
KBMANAGER.WRITE         "Y"
KBMANAGER.READ          "Y"
DATALOADER.WRITE        "Y"
STATISTICS.READ         "Y"
CITATIONSFORMS.READ     "Y"
LOOKUP.READ             "Y"
EXPORT.READ             "Y"
EXPORT.WRITE            "Y"
ADMIN.WRITE             "Y"
FM.WRITE                "Y"
THRESHOLD.WRITE         "Y"
TARGET_MATCHER.WRITE    "Y"
MARC_ENHANCEMENT.READ   "N"
DEBUGGING.WRITE         "Y"
KB_REPORTS.WRITE        "Y"
SFXHELP.WRITE           "Y"
CATEGORY.WRITE          "Y"
AZ.WRITE                "Y"
INSTITUTE.WRITE         "Y"
CUSTOM.WRITE            "Y"
EndSection
```

12

SFX KnowledgeBase Update

This section includes:

NOTE:

For general information on the SFX KnowledgeBase update, refer to the *SFX General User's Guide*.

Location of Files Related to the Update

Update files are located as follows:

- All version 4 revisions can be found on the Ex Libris FTP server in `sfx/revisions_v4`. Each revision is stored in a separate directory.

  For example, the KBDB revision with Release Number 20100500 is stored in `sfx/revisions_v4/20100500/sfx-kbdb-update-20100500.run`.
- A list of all version 4 revisions and the order in which they need to be applied can be found in `sfx/revisions_v4/list/RevisionList.pm`.
- On the local SFX server, the `exlibris/sfx_ver/sfx4_1/.sfx_history` file contains information about which revision packages have been applied for each instance.
- Revision packages are downloaded to the local SFX server during the revision update in the `/exlibris/sfx_ver/sfx4_1/tmp/rev-up-downloads` directory.

- The report files can be found in the `/exlibris/sfx_ver/sfx4_1/tmp` directory.

  It is best to clean this directory after the update but retain the latest backup of the database.

- The `ftp.config` configuration file contains the information necessary to FTP the KnowledgeBase package from the FTP site. The file can be found in the `/exlibris/sfx_ver/sfx4_1/sfxglb41/config` directory.

  The configuration file contains the host name, user name, and password (in encrypted form) to connect to the Ex Libris FTP server. Additionally, the configuration file contains an option to define whether active or passive FTP should be used. By default, passive FTP is used and the value of the `passive` parameter is either blank or `1`. If it is necessary to use active FTP, change the `passive` parameter to `0`.

- At the start of the revision update, a backup is made of the prerevision state of the SFX server.

  Each backup is stored as a compressed `.tgz` file in `/exlibris/sfx_ver/sfx4_1/tmp/backup`.

- The following options can be used when running the revision update (`rev-up`) program:

```
rev-up [options]
Options:
--status   shows current state of server
--available         fetches list of available updates from Ex Libris
--justget=ID      downloads update(s), if not specified, 'all' is
     assumed
--justgetall      downloads all relevant update(s)
--justapply=ID    applies update(s), if not specified, 'all' is
     assumed
--applyall applies all relevant update(s)
--instance=NAME   name of instance to update, not a good idea for
     sw updates
--defaults assume that the answer to any question is the default
--nolog, --log    disable logging
--logfile  full path of log file
--manualdownload assumes all files already downloaded into
Directory specified by rev-up.conf
--type=type restricts the process to run only on a type of
revision, i.e. kbdb or sw
--help     brief help message
--man      full documentation
--apply_v3_all  downloads and applies all updates up to the latest
released revision in V3
```

To get information on the latest KB Revision and Service Pack applied on the SFX server, type the command `ver` (this works from any location in the SFX file system, as any user).

The following is displayed:

```
KBDB: <KBDB revision number> KBSW: <KBSW revision number>;
SW: <SP number>; EXTRA: <last 'extra' package applied'>
```

For example:

```
KBDB: 20134600; KBSW: 20134600; SW: 4.6.0; EXTRA: 4.6.0
```

Run the revision update program with the `--apply_v3_all` option after installing SFX 4 and before starting the upgrade to ensure that SFX 4 is up-to-date with the latest revision released for SFX 3 (a prerequisite for the SFX 3 to SFX 4 database migration).

When running the revision update on your SFX 4 installation with the `--apply_v3_all` option, the program retrieves the `/sfx/upgrade_3_to_4/revision_info_ver3_and_ver4.txt` file from the Ex Libris FTP server. This file contains a list of SFX 3 and SFX 4 revisions and maps the SFX 4 KBDB revision number to the latest SFX 3 KBDB revision released to customers.

Running this option is needed to make sure SFX 3 and SFX 4 are synchronized.

Manual FTP

If, for some reason, you are not allowed to FTP the revision package automatically during the `revision_update` program, you need to download the necessary files manually and place them in the `/exlibris/sfx_ver/sfx4_1/tmp` directory before you can apply the SFX update.

To manually download the files:

1. Download the `RevisionList.pm` file located in `ftp.exlibris-usa.com/sfx/revisions_v4/list` and place the file in the `/exlibris/sfx_ver/sfx4_1/tmp/rev-up-downloads` directory.
2. If you are using the `--apply_v3_all` option (as described in the previous section) in combination with manual FTP, download from the FTP server the file that lists the SFX 4 and corresponding SFX 3 revisions and place the file in the `/exlibris/sfx_ver/sfx4_1/tmp/rev-up-downloads` directory:

```
cd /exlibris/sfx_ver/sfx4_1/tmp/rev-up-downloads
ftp ftp.exlibris-usa.com
cd /sfx/upgrade_3_to_4/
get revision_info_ver3_and_ver4.txt
bye
```

3. Run the `revision_update` program to create a list of necessary software updates.

a Log on to the SFX server as the `sfxglb41` user.

b Enter `da` to go to the `/admin` directory.

c Start the software update program by entering the following:

```
./revision/rev-up --available --manualdownload
```

A report is created to list the revisions needed in each instance.

4 Using the report, manually download all the following update packages, where `<mmddyyyy>` is the revision date of each package listed in the report file:

- `sfx-kbdb-update-<mmddyyyy>.run` located at `ftp.exlibrisusa.com/sfx/revisions_v4/<mmddyyyy>`
- `sfx-kbsw-update-<mmddyyyy>.run` located at `ftp.exlibrisusa.com/sfx/revisions_v4/<mmddyyyy>`
- `sfx-sw-update-<VERSION.SP.minorRelease>.run` located at `ftp.exlibris-usa.com/sfx/revisions_v4/<mmddyyyy>`
- `sfx-extra_<VERSION.SP.minorRelease>.run` located at `ftp.exlibris-usa.com/sfx/revisions_v4/<mmddyyyy>`
- tar files associated with the extra package located at `ftp.exlibris-usa.com/sfx/revisions_v4/<mmddyyyy>`

  Note that you can reduce download time of the operating system specific tar files (for Linux or Solaris) by downloading only the tar file appropriate for the operating system of your SFX installation.

  If an extra package is included in the revision update, do the following before running the rev-up program with the --`manualdownload` option (as the `sfxglb41` user):

```
    type 'Setenv manualdownload 1'
```

If the revision update is run in batch mode (for more information, see **Running the KnowledgeBase Update in Batch Mode** on page **163**), with the `manualdownload` option for a revision with an extra package, add the following line to the `sfxglb41.cshrc` file before starting the rev-up program:

```
    type 'Setenv manualdownload 1'
```

5 Once the files are placed in the `/exlibris/sfx_ver/sfx4_1/tmp/rev-up-downloads` directory, run the `revision_update` program using `./revision/rev-up --manualdownload`.

```
SFX Revision Update

$Id: rev-up,v 1.6 2005/06/01 04:19:15 am_3 Exp $
Logging at '/exlibris/sfx_ver/sfx4_1/tmp/rev-up-2005-05-
31T23:24:40.log'.

Looking for 'RevisionList.pm'... found
Looking for 'sfx-kbdb-update-20050601.run'... found
Looking for 'sfx-sw-update-20050601.run'... found
```

Running the KnowledgeBase Update in Batch Mode

To run the KnowledgeBase update in batch mode, you need the UNIX root password and the database passwords of all the instances you want to update.

To run the KnowledgeBase update in batch mode:

1 Log on as `sfxglb41`.

2 Type `da`.

3 Type `su`.

4 Enter the UNIX root password.

5 Enter the `/bin/tcsh -c "source <home directory of sfx global>/.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb/admin/revision/rev-up"` command to start the `revision_update` program.

For example:

```
/bin/tcsh -c "source /exlibris/sfx_ver/sfx4_1/sfxglb41/home/
.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/revision/rev-
up"
```

IMPORTANT:

There must be a space between `source` and the path to the environment variables of the global instance. The environment variables of the global instance must be loaded in order for the update to be run as root.

Additional Options

This section describes four possible options for running the KnowledgeBase update:

OPTION 1

If you want to run the update in the background, log on as a root user and send the process to the background.

On a Linux machine, add `&` to the command. For example:

```
/bin/tcsh -c "source /exlibris/sfx_ver/sfx4_1/sfxglb41/home/
.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/revision/rev-
up" &
```

On a Solaris machine, press CTRL-Z to stop the process and enter `bg` to send it to the background. For example:

```
/bin/tcsh -c “source /exlibris/sfx_ver/sfx4_1/sfxglb41/home/
.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/revision/rev-
up"

Ctrl+Z

bg
```

OPTION 2

If you want to run the update at certain time, such as 1:00 AM on the 15th of every month, set the command in a cronjob of root. Root permissions and knowledge of the crontab command are required. You may need to contact your system administrator to set this up.

Your `crontab -l` may look like the following for a system in which the update is going to run at 1:00 AM on the 15th of every month:

```
[root@carreta root]# crontab -l

00 01 15 * * /bin/tcsh -c "source /exlibris/sfx_ver/sfx4_1/
sfxglb41/home/.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/
admin/revision/rev-up"
```

OPTION 3

You can run the update in `nohup`, enabling the process to continue even if the terminal window is closed.

NOTE:
To run the `nohup` command, you need the root password.

To run the `nohup` command:

1. Log on as `sfxglb41`.
2. Enter `da`.
3. Enter `su`.
4. Enter the root password.
5. Paste the `nohup` command line.

To verify that the `nohup` is running correctly:

1. Log on to another `sfxglb41` session.
2. Enter `da`.
3. Enter `su`.
4. Enter the root password.
5. Enter `vi nohup.out`.

You see the files being downloaded from the FTP site and the software and KB packages being applied.

The `nohup.out` file should be deleted before each new update. Otherwise, the new data is appended to the same file.

The following is an example of a Linux `nohup` command:

```
nohup /bin/tcsh -c "source /exlibris/sfx_ver/sfx4_1/sfxglb41/
home/.cshrc; /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/
revision/rev-up" &
```

The following is an example of a Solaris `nohup` command:

```
nohup /bin/tcsh -c "source /exlibris/sfx_ver/sfx4_1/sfxglb41/
home/.cshrc;  /exlibris/sfx_ver/sfx4_1/sfxglb41/admin/
revision/rev-up"
```

OPTION 4

If you do not have root access to the SFX server, you can run the software and KB update separately. Root access is required only for the software part of the update, not for the KB part which is more time-consuming.

To run the KB update without root access:

1. Enter `./revision/rev-up --type=sw .`

   If you run this command as `sfxglb41`, you are prompted for the UNIX passwords for the different SFX instances.

2 Enter `./revision/rev-up --type=kbdb`.

If you run this command as `sfxglb41`, you are not prompted for any passwords.

Related options include:

```
Rev-up --nointeractive
```

This option skips the creation of symbolic links from the local instances to new files in `sfxglb41`, which is the part of the software update for which root access is required.

This option can be used in cases where no root access is allowed, but the symbolic links need to be created afterwards.

Revision Update Backup and Rollback

When you start the revision update program, a backup of all directories and database tables that may be affected by the update is automatically performed. This backup allows a rollback to the prerevision state of the server in case a problem occurs during the revision update. A rollback allows all software, KB software, and KB database package changes to be rolled back if a problem occurs.

If multiple KB database packages are applied simultaneously, the revision update program makes an additional temporary backup of each database before it is updated. This temporary backup is made from the second KB database package forward and is removed from the server after the database is successfully updated. The purpose of this temporary backup is to make it easier to do a one-instance rollback.

Contact your local Ex Libris Support Center if you encounter problems while applying an update. The Support Center can provide advice on whether a roll-back is needed and whether to use a complete rollback or a one-instance rollback.

Rollback instructions and details about the backup and rollback procedure via the Server Admin Utility can be found in the **Revision Backup/Rollback** section of the *SFX System Administration Guide*.

Description of the Full-Revision Update Backup

During the revision update backup, the following data is saved:

- All SFX databases that are updated during the revision

NOTE:

This is not a full backup. Only the tables altered by the revision update are backed up.

- All software files in the following folders of the `sfxglb41` instance:
    - `admin`
    - `cgi`
    - `lib`
    - `config`
    - `dbs/sqltab`
    - `sfxadmin`
    - `templates`
- `.sfx_history` file (includes a list all revisions applied on each instance)

Location/Format of the Backup

Each backup is stored as a compressed `.tgz` file in `/exlibris/sfx_ver/sfx4_1/tmp/backup`.

NOTES:

- Third party applications located in the `/exlibris/sfx_ver/sfx4_1/app` directory of the SFX server are not part of the backup/rollback procedure. This directory contains all supporting software required to run SFX (MySQL, Perl, Java, and Apache). When updates are needed for this directory, the revision update includes an extra package. For these extra packages, no rollback is provided. Therefore, when an extra package is part of a revision, the revision release note instructions recommend that you make a full backup of the SFX system before the revision update program is run.
- The SFX server requires the correct file permissions on the `sfxglb41` instance directories to ensure a successful backup. If the backup fails because of incorrect file permissions, the program reports the problem and stops. You need to resolve these file permission problems before you can run the revision update program.

The full backup is not performed automatically if the revision update program did not finish successfully for all instances in a previous run of the revision. If SFX detects that not all SFX databases are up-to-date with the same revision number and the revision update program is run as `sfxglb41`, you are asked whether or not a full backup should be taken. Since a full backup of the prerevision state in which all instances are in sync most likely still exists, no additional backup is needed in most cases.

For example:

```
IMPORTANT NOTE:
.sfx_history file is out of sync
Would you like to run a full backup? [y/N]
```

If SFX detects that not all SFX databases are up-to-date with the same revision number and the revision update program is run as root, the backup is not performed.

Description of the Temporary Database Backup

When the revision update program detects that multiple KB database packages need to be applied on the SFX server, the revision update program makes an additional temporary backup of each database before it is updated. The purpose of this temporary backup is to make it easier to do a one-instance rollback.

This temporary backup procedure starts from the time the second KB database package is applied. Before each database is updated, a temporary backup is taken. Once the database is successfully updated, the temporary backup is removed.

NOTE:

This is not a full backup. Only the tables altered by the revision update are backed up.

Location/Format of the Temporary Database Backup

The backup is stored as a mysqldump `.sql` file in `/exlibris/sfx_ver/sfx4_1/tmp/applykbdb-bak`.

13

DOI/CrossRef and NCBI Pubmed Setup

This section includes:

Overview of DOI and CrossRef

A **DOI** (digital object identifier) is a permanent identifier given to a Web file or other Internet document. For example: `10.1103/PhysRevE.62.1457`

In this example, the 10.1103 prefix identifies the publisher and the part after the "/" contains the DOI suffix—in this case, the journal title, volume, and start page information that identifies a particular article published in Physical Reviews E.

DOI Linking

DOIs are submitted to a centrally managed directory and can then be used in a URL that contains the address of the directory plus the DOI.

If you want to use DOI linking to go to a particular article published in Physical Reviews E, use the following URL:

http://dx.doi.org/10.1103/PhysRevE.62.1457

The **DOI system** was conceived by the Association of American Publishers in partnership with the Corporation for National Research Initiatives and is now administered by the International DOI Foundation. Essentially, the DOI system is a scheme for Web page redirection by a central manager.

CrossRef

CrossRef is an initiative of the Publishers International Linking Association (PILA). It is the official DOI registration agency for scholarly and professional publications, including journals, books, and other content types.

Source: http://whatis.techtarget.com

How CrossRef/DOI and SFX Work Together

This section summarizes the different ways in which SFX and CrossRef/DOI can work together:

- When the SFX server receives a DOI, DOI linking can be provided in the SFX menu screen. To use this functionality, it is necessary to activate the DOI target in the SFX KnowledgeBase. This ensures that when the SFX server receives a DOI, a target offering DOI linking is presented in the SFX menu screen.
- SFX can query the CrossRef reference database with citation metadata and DOI information. This capability allows for enhancement of metadata and improved linking capabilities for certain targets.

Each of the above options is explained in greater detail in the following sections.

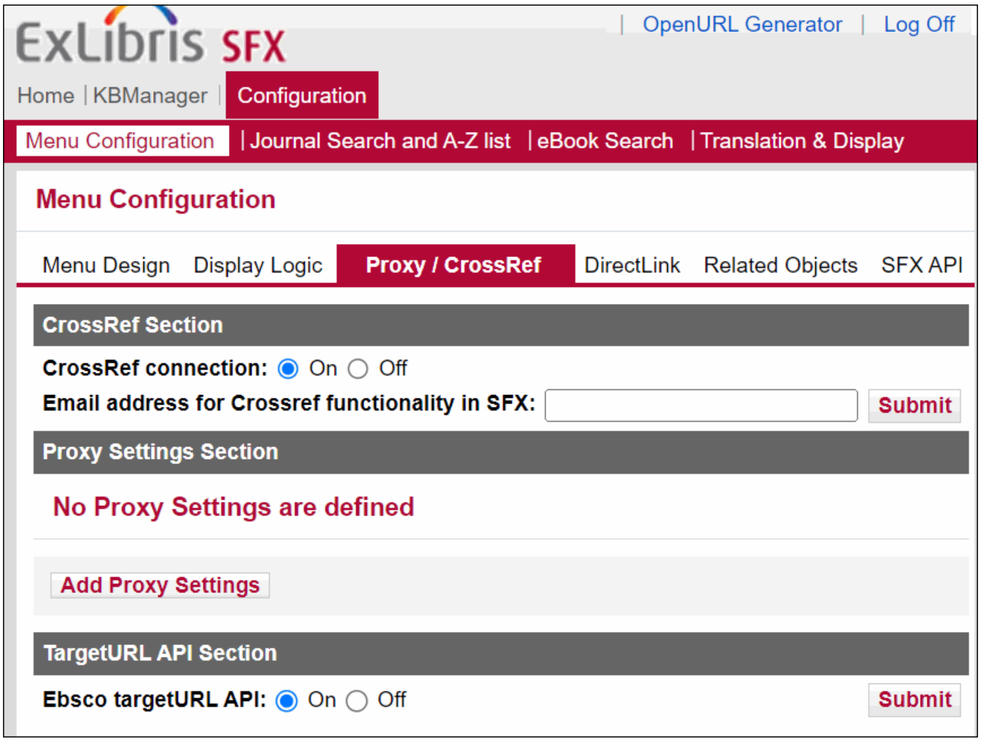
The CrossRef Connection Option in the SFX Admin Center

The connection to CrossRef is enabled in the **SFX Admin Center > Configuration > Menu Configuration > Proxy/CrossRef** tab. The CrossRef connection can be turned on or off.

By default, the CrossRef connection is turned on, which means that a lookup will be performed in CrossRef when appropriate. CrossRef can be used in either of the following cases:

- to look up metadata using DOI information in the OpenURL to enhance the ContextObject
- to look up DOI information based on metadata available to improve target linking

When the CrossRef connection option is turned off, no lookup is performed in CrossRef for the SFX instance. This option should be used in the case of significant CrossRef server downtime.


ExLibris SFX
| OpenURL Generator | Log Off
Home | KBManager | Configuration
Menu Configuration | Journal Search and A-Z list | eBook Search | Translation & Display
Menu Design
Display Logic
Proxy / CrossRef
DirectLink
Related Objects
SFX API
Submit
Add Proxy Settings
Submit

Menu Configuration

CrossRef Section

CrossRef connection: ● On ○ Off

Email address for Crossref functionality in SFX:

Proxy Settings Section

No Proxy Settings are defined

TargetURL API Section

Ebsco targetURL API: ● On ○ Off

Figure 58: Crossref Section

Activating the DOI Target

When the SFX server receives a DOI, DOI linking can be provided in the SFX menu screen.

To use this functionality, it is necessary to activate the DOI target in the SFX KnowledgeBase. This ensures that when the SFX server receives a DOI, a target offering DOI linking is presented in the SFX menu screen.

To activate the DOI target:

1. Access the SFX Admin Center of your instance and select **KBManager > Targets**.
2. Enter `DOI` in the **Filter Target List** box.
3. Activate the DOI target and target service.

To test the newly activated target, use the following OpenURL:

http://<sfx_server>:<port>/
<sfx_instance>?genre=article&id=doi:10.1103%2FPhysRevE.62.1457

The new target should be presented in the SFX menu screen.

Using CrossRef Connection to Enable Article-Level Linking

Some SFX targets require the presence of a DOI in order to create an article-level link. However, the SFX source does not necessarily send the DOI as metadata on the OpenURL (for example, when the DOI may not be available in the source document). In such circumstances, the DOI can be retrieved by SFX from the CrossRef database. The metadata received via the OpenURL is sent to the CrossRef database and where a match is found, the corresponding DOI is sent back to SFX and is used to link to the target.

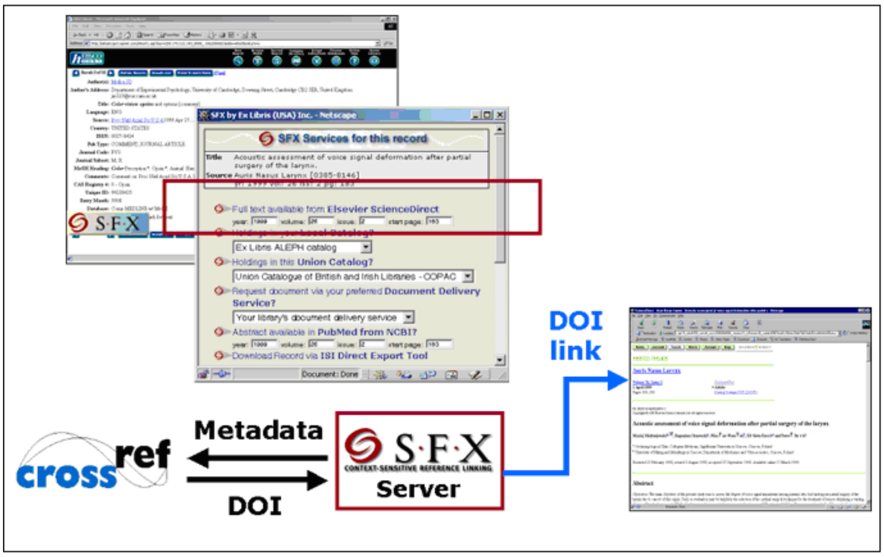
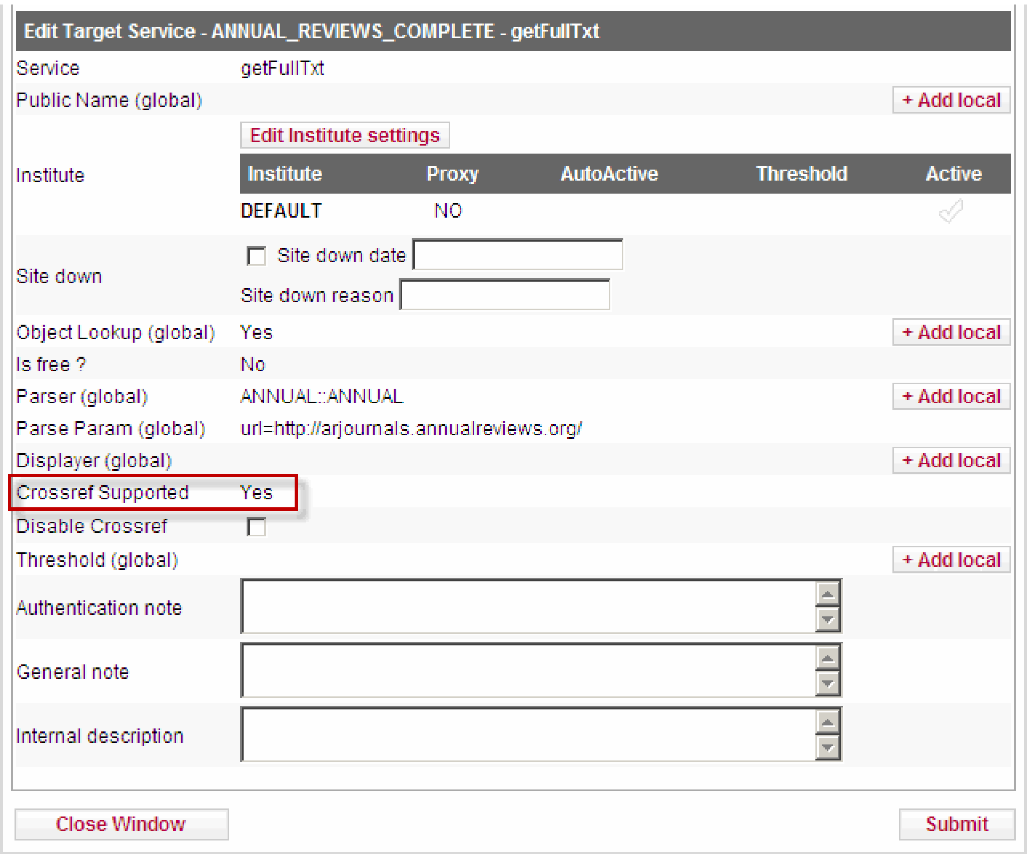

SFX by Ex Libris (USA) Inc. - Netscape
SFX Services for this record
Title Acoustic assessment of voice signal deformation after partial surgery of the larynx.
Source Auris Nasus Larynx [0385-8146]
Full text available from Elsevier ScienceDirect
year: 1999 volume: 26 issue: 2 start page: 183
Holdings in your Local Catalog?
Ex Libris ALEPH catalog
Holdings in this Union Catalog?
Union Catalogue of British and Irish Libraries - COPAC
Request document via your preferred Document Delivery Service?
Your library's document delivery service
Abstract available in PubMed from NCBI?
year: 1999 volume: 26 issue: 2 start page: 183
Download Record via ISI Direct Export Tool
Document: Done
S·F·X
crossref
Metadata
DOI
S·F·X CONTEXT-SENSITIVE REFERENCE LINKING
Server
DOI link

Figure 59: DOI Link

Each target in the SFX KnowledgeBase for which a DOI can be used for article-level linking contains a field called **Crossref Supported** with a value **Yes** in the View and Edit windows of the target service.

An example is shown below.



Edit Target Service - ANNUAL_REVIEWS_COMPLETE - getFullTxt

Service	getFullTxt	
Public Name (global)		+ Add local
Institute	Edit Institute settings	
Site down	□ Site down date Site down reason	
Object Lookup (global)	Yes	+ Add local
Is free ?	No	
Parser (global)	ANNUAL::ANNUAL	+ Add local
Parse Param (global)	url=http://arjournals.annualreviews.org/	
Displayer (global)		+ Add local
Crossref Supported	Yes	
Disable Crossref	□	
Threshold (global)		+ Add local
Authentication note		
General note		
Internal description		

Institute	Proxy	AutoActive	Threshold	Active
DEFAULT	NO			✓

Figure 60: CrossRef Supported - Example

This field ensures that if a DOI is not part of the ContextObject, SFX sends metadata to Crossref and retrieves the matching DOI. It is possible to disable this Crossref retrieval by selecting the **Disable Crossref** check box.

The CrossRef retrieval occurs only if the following conditions are met:

- The **Crossref Supported** field of the target service is set to **Yes** and the Crossref retrieval is not disabled.
- An email address has been entered in the Menu configuration > Crossref/ Proxy section of SFXAdmin (see instructions below).
- The CrossRef connection is set to **On** (as described in **The CrossRef Connection Option in the SFX Admin Center** on page 170).

To use crossref connection to enable article-level linking:

1. Make sure that CrossRef is supported for the given target service and that it is not disabled.
2. Enter your library email address in the Menu Configuration > Crossref/ Proxy section of SFXAdmin.

This allows Crossref to contact you in case of problems with linking and replaces the username and password Crossref required previously.

Using CrossRef Connection to Enhance SFX Services

To use CrossRef Connection to enhance SFX services (unless you performed the procedure at the end of the Using CrossRef Connection to Enable Article-Level Linking section, immediately above):

1. Make sure that CrossRef connection is enabled.
2. Enter your library email address in the **Menu Confguration > Crossref/ Proxy** section of SFXAdmin.

Providing this email address allows Crossref to contact you in case of problems with linking and replaces the username and password Crossref required previously.

CrossRef Timeout Settings

Two timeout settings exist in the `/config/crossref.config` configuration file. These settings ensure that during CrossRef service downtime, there is a minimal effect on the SFX server performance. The default timeout value is set to five seconds for both settings. This can be changed by editing the `/config/crossref.config` configuration file.

```
Section "parsePrivateID"
    module       "Parsers::SourceParser::OpenURL_1_0::doi"
    user         ""
    pass         ""
    base         "idea"
    type         "xml"
    path         "/query/xref.cgi"
    sid          "null:null"
    type         "xml"
    doiurl       "http://dx.doi.org"
    xrefurl      "www.crossref.org"
    timeout      "5"
EndSection

Section "fetchRecord"
    module       "NetWrap::HTTP"
    host         "doi.crossref.org"
    timeout      "5"
EndSection
```

The first timeout value, under the parsePrivateID section, controls the timeout for DOI lookup (DOI lookup based on metadata available to improve target linking, as described in **Using CrossRef Connection to Enable Article-Level Linking** on page 172).

The second timeout value, under the fetchRecord section, applies to the metadata lookup (when looking up metadata using DOI information in the OpenURL to enhance the ContextObject, as described in **Using CrossRef Connection to Enhance SFX Services** on page 174).

If these timeout entries do not exist in your instance, it is recommended to add them based on the example above.

To add timeout entries:

1 Enter `cn`.

2 Enter `vi crossref.config`.

3 In the parsePrivateID section, below the `xrefurl` entry, ensure that the following line exists and add it if it does not exist:

```
timeout     "5"
```

4 Save the file and quit.

DOI and CrossRef Troubleshooting

To troubleshoot problems, see **Tool to Verify Setup of PubMed and CrossRef/ DOI Source Fetching** on page 229.

NCBI Pubmed

PubMed is a free search engine accessing primarily the MEDLINE database of references and abstracts on life sciences and biomedical topics. The United States National Library of Medicine (NLM) at the National Institutes of Health maintains the database as part of the Entrez system of information retrieval.

A PMID (PubMed identifier or PubMed unique identifier) is a unique integer value, assigned to each record in the NCBI Pubmed database. For example: 27252163.

To view the record for this PMID in Pubmed, see:

https://www.ncbi.nlm.nih.gov/pubmed/27252163

When the SFX server receives a PMID as part of the OpenURL, SFX queries Pubmed to fetch the corresponding citation metadata for this record. This capability allows for the enhancement of the metadata and for improved linking capabilities. To fetch the metadata, SFX uses the Pubmed API.

Customers expecting to send more than three PMID requests from SFX to Pubmed per second need to create an API key and store it in the Pubmed source configuration file at:

`config/source/OpenURL_0_1/pmid.config_`

`config/source/OpenURL_1_0/pmid.config_`

Information about the API key API can be found at:

https://ncbiinsights.ncbi.nlm.nih.gov/2017/11/02/new-api-keys-for-the-e-utilities/

The key should be stored in the `api_key` field in:

`/exlibris/sfx_ver/sfx4_1/<instance>/config/source/OpenURL_0_1/pmid.config_`

`/exlibris/sfx_ver/sfx4_1/<instance>/config/source/OpenURL_1_0/pmid.config_`

The following is an example of he configuration file with `api_key` filled in:

```
PubMed configuration
#
Section "source"
id "PUBMED"
EndSection
Section "parsePrivateID"
module "Parsers::SourceParser::OpenURL_0_1::pmid"
EndSection
Section "fetchRecord"
module "NetWrap::HTTP"
host https://www.ncbi.nlm.nih.gov
timeout "3"
api_key "XXXXXX"
EndSection
Section "parseRecord"
module "Parsers::SourceParser::OpenURL_0_1::pmid"
EndSection
```

14

Plug-In Feature

This section includes:

Overview of the Plug-in Feature

“Traditional” thresholds check whether conditions are met based on the content of the ContextObject (within SFX).

The plug-in feature introduces a new type of threshold, one which determines if a condition was met based on a value returned by a program.

For example, if you want a holding link for a particular book or journal to appear in the institution’s SFX menu screen only if the target catalog actually has a record for this particular book or journal, you must set a plug-in threshold for the getHolding target service.

When a user sends an SFX request, SFX determines the services that are candidates for display in the SFX menu. Before displaying these services, however, SFX checks whether the different conditions—determined by the corresponding thresholds, including the plug-in threshold—are met.

To determine whether the plug-in threshold is met, SFX runs the program specified in the plug-in threshold. This program checks whether the target catalog has a record for the particular journal/book contained in the SFX request.

If the catalog has the book/journal, the program returns the value **1**. Otherwise, the program returns the value **0**.

If the program returns **1**, the condition in the plug-in threshold is met and the getHolding service is displayed in the SFX menu.

If the program returns **0**, the condition in the plug-in threshold is not met and the getHolding service is not displayed in the SFX menu.

Implementation of the Plug-in Feature

You can implement the plug-in feature.

To implement a plug-in threshold:

1 Write a stand-alone program that checks the status of a condition. The program should return **0** or **1**.

The location of the program should be:

```
/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/PlugIn
```

NOTE:
When writing a program, make sure all plug-in subroutines are called **lookup**.

2 Set a plug-in threshold on the target or target service level. For example:

```
$obj->plugIn(name of plug-in program)
```

The following plug-in programs are available in SFX:

- `Aleph.pm` – checks whether a specific journal /book exists in the Aleph catalog (version 14.2 or later)
- `z3950_lookup.pm` – checks whether a specific journal/book exists in a catalog based on Z39.50
- `syndetics.pm` – checks whether the book review for a particular book exists in Syndetics, based on an ISBN search

Plug-In Parallel Processing

The following section describes the Parallel Plug-In Processing feature.

Overview

The parallel plug-in processing feature aims to reduce the response time of the SFX Menu when multiple plug-ins are used by running several plug-ins in parallel instead of consecutively.

This feature is optional. It will apply to those plug-ins enabled by the configuration file for parallel processing.

In a simple scenario, where each target service has only one plug-in and there are no dependencies on data that was fetched by another plug-in, all plug-ins can run in parallel. In other cases, parallel processing is more complicated, for example in case of conflicts, where plug-ins use and update the same CTXO attribute, or in cases where the display logic excludes services against another and both use a plug-in.

All settings are controlled by the following configuration file:

```
/exlibris/sfx_ver/sfx4_1/<instance name>/config/
parallel_plugin.config_
```

The default number of plug-ins that can be run in parallel is 10. This number is configurable.

There are two limitations to parallel processing:

- If the same item has a threshold using several plug-ins, the plug-ins run sequentially.
- If SFX handles multiple context objects – for example for several related objects in one request – the parallel processing is done for each object, but context objects are handled sequentially.

Configuring Plug-in Parallel Processing

The `parallel_plugin.config_` configuration file contains sections for all plug-ins that are globally supported for this feature. You can add sections for local plug-ins. Plug-ins can be excluded from or included in the parallel processing.

Plug-ins that are not listed in the configuration file are processed sequentially, after the plug-ins that are listed are processed. The same applies to plug-ins that are listed but disabled for parallel processing.

The configuration file is used to control the following settings:

Table 10. Plug-In Parallel Processing Configuration Settings

Parameter	Description
General	

Table 10. Plug-In Parallel Processing Configuration Settings

Parameter	Description
`enable_parallel_processing`	Possible Values: Y/N Enable/disable the parallel functionality across the system.
max_plugin_excecutions_in_parallel	Default Value: 10 The maximum number of plug-ins that can run in parallel.
For each Plug-In	
`Active`	Possible Values: Y/N Activate the plug-in module in parallel mode.
`insert_data_order`	The plug-in with the higher number for this parameter inserts its fields first. (This can be overwritten by later plug-in activity if conflicting data is fetched. This applies where several plug-ins populate the same field in the CTXO).
`multiple_running`	Possible Values: Y – the plug-in should run in parallel every time it runs N – the plug-in should run in parallel only the first time it runs. This may be desired if the same plug-in is used by several active target services, and the first request already returns all the data needed. In this case there is no point in running the plug-in several times in parallel. This applies for example to the EBSCO API plug-in.
`fields_from_ctx_obj`	Insert the names of the fields from the ctx object with a semicolon (;) separator between them, for example: `sfx.aleph_title_index;fetchId`. This is needed because at the end of the parallel processing the CTXOs of all the plug-ins that ran in parallel are merged into one CTXO. The fields you add to this parameter tell SFX what fields in the CTXO to populate.

Plug-in Program for Aleph

The plug-in program for Aleph uses the X-server functionality (available in Aleph version 14.2.4 and later). The program checks the Aleph catalog for the existence of a bibliographic record for a specific item. Based on the reply of the X-server, the plug-in program decides whether or not to show a link to the catalog in the SFX menu.

Additionally, it is possible to configure the Aleph plug-in to retrieve Aleph summary holding statements to be displayed in the SFX menu.

Requirements

- For ALEPH 14.2.4 through 15, the Aleph license must include Aleph X-server details. Contact your Aleph support team for more information.
- From Aleph 16.01 onwards, the following additional requirements exist:
  - A special/separate license (from the standard Aleph license) is required for using the X-server.
  - The existence of a staff user WWW-X (similar to Z39 Server).
  - server_ip_allowed with special X entries.

Setup

You can set up the plug-in program.

To set up the Aleph plug-in program:

1. Log on to your server as the instance user (for example, `sfxlcl41`).
2. Navigate to the `config` directory by typing `cn`.
3. Open the `aleph_xserver_lookup.config` file:

```
vi aleph_xserver_lookup.config
```

4. In this file, a number of configuration elements need to be entered. If you want the plug-in to check the catalog using ISSN or ISBN identifiers only, enter the URL, ISSN, ISBN index names, and the database name, as well as the URL for your catalog's X-server. Additionally, specify the timeout value (in seconds).

   For example:

```
Section "x_server"
    url         "http://www.university.edu:8991/X"
    issn_index  "ISSN"
    isbn_index  "ISBN"
    full_title_index   ""    #e.g. TIT
    word_title_index   ""   #e.g. WTI
    database    "USM01"
    timeout "5"
EndSection
```

5. If no ISSN or ISBN is available in the OpenURL, the plug-in program can be configured to check the catalog using title and author information found in the OpenURL.

- Title search – To configure this option, set the `title_option` parameter to `Y`, as in the example below. In addition, provide the codes of the Aleph Title, Title word, and Author indexes. Indexes whose codes are not provided in the configuration file are not used for lookup.

  When searching using the title, a search is first performed on the Aleph full title index using the book or journal title. If this search retrieves zero results, a second search is performed on the Aleph title word index using a limited number of the longest words in the title. The number of words that are used for searching is defined by the `num_of_words_for_search` parameter. The default number is 4.

  If the book or journal title is longer than 10 words, it is not used for searching and a search is performed using the title's longest words in the Aleph title word index.

- Author search – To activate searching using author information, populate the `author_index` parameter of the configuration file with the Aleph author index code. In addition, if author information is included in the OpenURL and the type of object searched is a book, the first author's last name will be searched in the Aleph author index.

  The following is an example of this section in the `aleph_xserver_lookup.config` file::

```
Section "x_server"
    url         "http://www.university.edu:8991/X"
    issn_index  "ISSN"
    isbn_index  "ISBN"
    full_title_index   "TIT"        #e.g. TIT
    word_title_index   "WTI"        #e.g. WTI
    database    "USM01"
EndSection

Section "searching"
    title_option             "Y"
    author_index             "WAU"
    num_of_words_for_search  "4"
EndSection
```

6 Access the SFX Admin Center. In KBManager, locate the relevant target service (for example, LOCAL_CATALOGUE_EX_LIBRIS_ALEPH). In the target service, add the following threshold:

```
$obj->plugIn('ALEPH')
```

If a global threshold exists, it is recommended that you use the following:

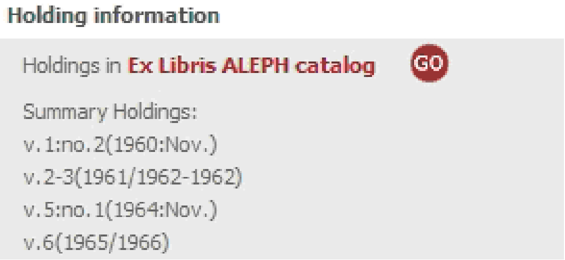
```
GLOBAL && $obj->plugIn('ALEPH')
```

NOTE:
Use single quotation marks around the word `ALEPH` in the threshold field.

7 Test the above by creating an OpenURL (use the OpenURL generator) for a journal or a book that exists in the Aleph OPAC. The link to the Aleph OPAC should appear in the SFX menu screen.

Summary Holdings Display

The Aleph plug-in can be configured to retrieve the Aleph summary holding statements for display in the SFX menu.



Holding information

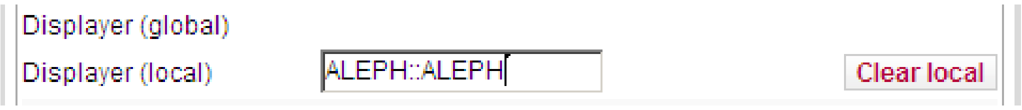
Holdings in Ex Libris ALEPH catalog

Summary Holdings:
v.1:no.2(1960:Nov.)
v.2-3(1961/1962-1962)
v.5:no.1(1964:Nov.)
v.6(1965/1966)

Figure 61: Holding Information

For this feature to work, the following conditions must be met:

- The Aleph OPAC needs to contain **Summary holding statements**—for example, in the 852 and 866 MARC fields.
- The Aleph X-server needs to be configured to include these fields in the X-server **present** response. (More information about setting this up can be found in the *Summary Holdings to Display in SFX: Aleph Side Configuration* document located in KB item 16384-42722 *How to Display Aleph Summary Holdings in the SFX Menu*.)
- In KBManager, a local target displayer should be added to the Aleph getHolding target service. The value should be set to: `ALEPH::ALEPH`.



Displayer (global)

Displayer (local) ALEPH::ALEPH

Figure 62: ALEPH::ALEPH

To enable the summary holding statement display in the SFX menu, perform the following SFX configuration:

1 Log on to your server as the instance user (for example, `sfxlcl41`).

2 Navigate to the config directory by typing `cn`.

3 Open the `aleph_xserver_lookup.config` file :

```
vi aleph_xserver_lookup.config
```

4 In the Display section, turn on the summary holding statement display option by setting `display_summary_holdings` to `Y`, and specify which MARC field/subfield(s) contain the summary holdings statement.

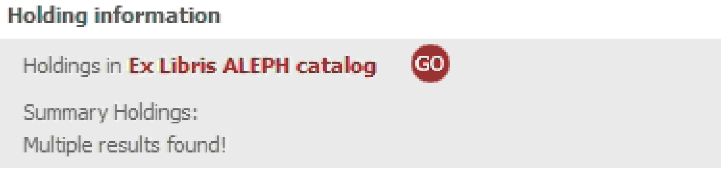
In addition, using the `cut_off_num` parameter, you can determine the maximum number of holding statements to be shown in the SFX menu.

For example:

```
Section "display"
marc_field                "852$a 866$a"
display_summary_holdings  "Y"
cut_off_num               "10"
EndSection
```

NOTES:

If there is more than one match for the OPAC Plug-in search, the following message is displayed:



Holding information

Holdings in Ex Libris ALEPH catalog

Summary Holdings:
Multiple results found!

Figure 63: Multiple Results Found

Plug-in Program Based on Z39.50

This plug-in program uses Z39.50 to query the catalog and determine (based on the reply) whether or not to show a link to the catalog in the SFX menu screen. In addition, it is possible to configure the Z39.50 plug-in to retrieve summary holding statements to be displayed in the SFX menu.

NOTE:
The display summary holdings feature does not work when there is more than one record available for the object.

To start the program:

STEP 1: Provide Local Z39.50 Information in z3950_lookup.config

1. Log on as `sfxlcl41`.
2. Enter `cn`.
3. Enter `vi z3950_lookup.config`.
4. Enter the Z39.50 server address, port, and database name.
5. Specify whether the ISSN and ISBN query should be performed with or without dashes (where `0` means without dashes and `1` means with dashes).

   For example::

```
Section "z3950_server"
host                " 111.22.3.44"
port                "210"
base                "innopac"
syntax              "USMARC"
EndSection

Section "dashes"
isbn_dashes         "0"
issn_dashes         "1"
EndSection
```

The following are additional configuration options:

- If no ISSN or ISBN is available in the OpenURL, the plug-in program can be configured to check the catalog using title information found in the OpenURL. To configure this option, set the `title_option` parameter to `Y`, as in the example below.

```
Section "searching"
title_option         "Y"
EndSection
```

- The plug-in can be configured to retrieve the catalog summary holding statements for display in the SFX menu. For this feature to work, the following conditions must be met:
  - The OPAC needs to contain **Summary holding statements**—for example, in the 852 and 866 MARC fields.
  - The Target Displayer for the Aleph getHolding target service needs to be set to: `Z3950::Z3950`.

    To start using this option, you must configure the Display section of the `z3950_lookup.config` file. Set the value of display_summary_holdings to Y and specify which MARC field/ subfield(s) contain the summary holdings statement.

In addition, the following two parameters can be configured:

- cut_off_num – This parameter determines the maximum number of holding statements to be shown in the SFX menu.
- charset – This optional parameter defines the character set of the z39.50 response. When blank, the default latin1 is used. Use utf8 or another character set if the z39.50 response is sent in a different character set.

  The following is an example of the z3950_lookup.config, where both title and display of summary holding options are turned on:

```
Section "z3950_server"
host                " 111.22.3.44"
port                "210"
base                "innopac"
timeout             "5"
syntax              "USMARC"
EndSection

Section "dashes"
isbn_dashes         "0"
issn_dashes         "1"
EndSection

Section "searching"
title_option         "Y"
EndSection

Section "display"
display_summary_holdings  "Y"
marc_field                "866$a 852$a"
cut_off_num               "10"
# optional parameter, can be "", "utf8", "latin1", etc.
# defines how to interpret raw data received from catalog
charset ""
EndSection
```

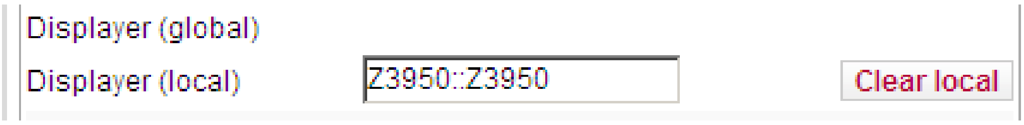
STEP 2

In KBManager, add the following threshold in the target service level of the catalog target:

```
$obj->plugIn('Z3950Lookup')
```

NOTE:
If a global threshold exists, it is recommended that you use: GLOBAL && $obj->plugIn('Z3950Lookup')

When the option to display summary holdings is turned on, the following target displayer is added in the target service level of the catalog target:



Displayer (global)

Displayer (local) Z3950::Z3950

Figure 64: Z3950::Z3950

STEP 3

Now that the plug-in feature is implemented, test it using an OpenURL of a journal or book that is available in the OPAC. Verify that the target shows up in the SFX menu screen.

You can also test this by using an OpenURL of a journal or book that is not available in the OPAC. The catalog target should not show up in the SFX menu screen.

If you encounter problems with the Z39.50 plug-in, it is recommended that you use the Z39.50 diagnostics tool (part of the server_admin_util) described in the *SFX System Administration Guide*.

NOTE:

It is possible to configure the Z39.50 plug-in to retrieve summary holding statements to be displayed in the SFX menu. For more information, see **Summary Holdings Display** on page **183**.

Plug-in Program for SYNDETICS

You do not have to make any changes to ensure that the SYNDETICS target uses the plug-in.

In KBManager, note that the TARGET_SERVICE global threshold for the SYNDETICS target is:

```
$obj->need('ISBN') && $obj->plugIn('syndetics')
```

A configuration file exists in each local instance to determine the time-out – the number of seconds SFX waits for a response from Syndetics before it is considered non-responsive and a failed threshold check.

The following is the location and file name:

/exlibris/sfx_ver/sfx4_1/<instance>/config/syndetics.config_

```
Section "general_configuration"
#    set timeout value
        timeout "10"
```

For more information about the Syndetics target, see the *SFX Target Configuration Guide*.

Plug-in Program for AIRway Project

This plug-in is to be used only with the AIRWAY_PROJECT target. When run, this plug-in sends a request to the AIRway Project server and checks whether it contains a record for the requested item. You do not need to make changes to the AIRway Project target in order to use the plug-in. The target service global threshold for the AIRway Project target is `$obj->plugIn` (AIRWAYLookup) in KBManager.

Plug-in Program for Ebsco

General Information

SFX can use an API on EBSCOHOST to pre-check the availability of an item and retrieve the direct URL. This mechanism improves the availability indication in cases where not all articles in a particular issue are available on the EBSCO platform. SFX only displays the full text link if the API provides a positive response. In addition, the retrieved URL increases the accuracy of links, especially if there are more than one article published on the same page. SFX uses a plug-in to retrieve the response from EBSCO full text targets, based on information in the OpenURL. Specifically it helps with the following:

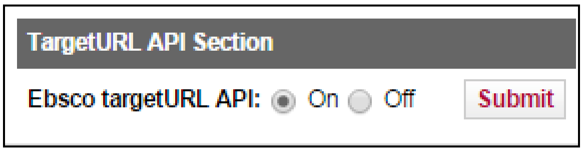
- Metadata mismatches between OpenURL sources and Ebsco targets
- Incomplete coverage in some Ebsco targets (only some articles are available in Ebsco in full text, not the complete issue)

NOTE:

The use of the EBSCO API is optional. Refer to **Plug-in Details** on page **191** for conditions and limitation of its use.

To configure the Ebsco plug-in in your local SFX instance:

1 Select **On** for the Ebsco targetURL API (**SFX Admin Center > Configuration > Menu configuration > Proxy/CrossRef**):



TargetURL API Section

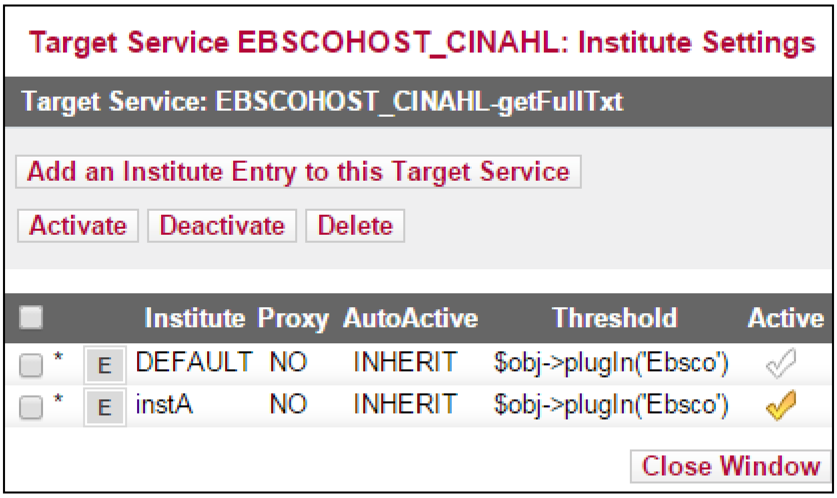
Ebsco targetURL API: ● On ○ Off

Figure 65: Ebsco targetURL API

2 Fill in the $$API_USER_ID linking parameter in the L/P area of the target service with the relevant value for your institution as received from the vendor. For more information about the $$API_USER_ID, see **API_USER_ID Linking Parameter** on page 190.

NOTE:
This linking parameter needs to be requested from Ebsco and needs to be filled in before the Ebsco plug-in will work.

3 If an institute specific activation exists for Ebsco target services, make sure that the threshold $obj->plugIn(Ebsco) exists also for the institute target service activation settings, as in the following image. For default activation, the plug-in thresholds already exist for each Ebsco getFullTxt target service.



Target Service EBSCOHOST_CINAHL: Institute Settings

Target Service: EBSCOHOST_CINAHL-getFullTxt

□			Institute	Proxy	AutoActive	Threshold	Active
□	*	E	DEFAULT	NO	INHERIT	$obj->plugIn('Ebsco')	✓
□	*	E	instA	NO	INHERIT	$obj->plugIn('Ebsco')	✓

Figure 66: Ebsco Threshold

NOTE:
In SP 4.8.3, the default EBSCO plug-in threshold $obj->plugIn(Ebsco) is automatically copied to all existing institute activation settings.

4 Check the set-up using the following OpenURL (or an alternative article level OpenURL) to make sure that the linking to Ebsco targets in the SFX menu is working properly:

```
<SFX base_url>?ctx_ver=Z39.88-2004&ctx_enc=info:ofi/enc:UTF-
8&ctx_tim=2015-04-02--10:0948&url_ver=Z39.88-
2004&url_ctx_fmt=infofi/fmt:kev:mtx:ctx&rft_val_fmt=info:ofi/
fmt:kev:mtx:journal&rft.genre=article&rft.atitle=Effects%20of
%20the%20Fitkids%20exercise%20therapy%20program%20on%20health
-related%20fitness%2C%20walking%20capacity%2C%20and%20health-
related%20quality%20of%20life&rft.jtitle=Physical%20Therapy&r
ft.aulast=Kotte&rft.auinit=E&rft.auinitm=M&rft.au=Kotte%2C%20
Elles%20M.%20W.&rft.date=2014&rft.volume=94&rft.issue=9&rft.s
page=1306&rft.epage=1318&rft.pages=1306-
1318&rft.issn=00319023&rft.eissn=15386724&rft_id=info:doi/
10.2522/ptj.20130315
```

For more information on the limitations of the Ebsco plug-in, refer to **Ebsco Plug-in Limitations** on page **192**.

API_USER_ID Linking Parameter

The API_USER_ID is an EBSCO authentication string that is customer specific and consists of the following three parts, separated by periods:

`<custid>.<groupid>.<profileid>`

- where `custid` is the EBSCOadmin customer id
- where `groupid` is the Group ID in EBSCOhost where the database resides
- where `profileid` is the Profile ID in EBSCOhost where the database resides

Example: s123456.main.eit

The structure of this authentication string is identical to the target information for Z39.50 connections.

Contact your EBSCO representative if you do not already have this information.

Note that Ex Libris uses the EBSCOhost API (EIT) to retrieve the list of databases per customer. Therefore, the API_USER_ID should be the same profile used with EIT, and your EIT profile should include all the full text databases.

NOTES:

- More information about setting up an EBSCOhost API (EIT) account can be found at http://support.ebsco.com/eit/ws_admin.php
- More information about EBSCOhost API (EIT) can be found at http://support.ebsco.com/eit/ws_api_info.php#b

Plug-in Details

The Ebsco plug-in consists of:

- A plug-in program: lib/Parsers/PlugIn/Ebsco.pm

  This program does the following:

  - Sends requests to Ebsco API
  - Stores article level links returned by the Ebsco API in CTXO so it can be used by the Ebsco target parser
  - Translates the response from Ebsco API as either 0 or 1 for the plug-in threshold $obj->plugIn(Ebsco)
- A plug-in threshold added to all full text Ebsco target services:

  ```
  $obj->plugIn(Ebsco)
  ```

Conditions for Using the Ebsco Plug-in

The Ebsco plug-in is used for one or more Ebsco targets that are active in the SFX instance if the following conditions are met:

- The SFX Menu configuration option **Ebsco targetURL API** is set to **On** in **SFX Admin Center > Configuration > Menu configuration > Proxy/ CrossRef**.
- The linking parameter $$API_USER_ID has been be filled in for the Ebsco targets. More information about the linking parameter can be found in the *SFX Target and Alma E-Collection Configuration Guide*. Note that this linking parameter can be defined for DEFAULT or can be institute specific, if institutes have been defined in the SFX instance.
- The Ebsco target service has threshold $obj->plugIn(Ebsco). In the SFX KnowledgeBase, a global threshold $obj->plugIn(Ebsco) exists for all Ebsco getFullTxt target services for which the Ebsco API can be used.
- The link in the SFX menu for Ebsco is an article level link, and the following metadata elements exist in the ContextObject:

  ```
  (rft.issn or rft.jtitle or rft.title) AND (rft.date and
  rft.volume and rft.issue) AND (rft.spage or rft.au or
  rft.atitle)
  ```

If the Ebsco API request cannot be sent because conditions are not met, the regular Ebsco target linking is used.

If the conditions are met, the Ebsco plug-in is used and an API request is sent to Ebsco.

If a URL is returned by the Ebsco API, it is used as a targetURL. If no URL is returned (and no server errors or time-outs occur), the Ebsco target is not displayed in the SFX menu – the Ebsco PlugIn returns `0` and threshold fails.

Ebsco Plug-in Limitations

The following limitations apply to the Ebsco plug-in:

- The Ebsco plug-in cannot be used selectively for individual EBSCO targets. If activated, it is used for all EBSCO targets.
- The Ebsco plug-in requires an Ebsco API user_id, stored in the SFX KnowledgeBase as a linking parameter (L/P). Only one Ebsco API user_id should be used per SFX instance. An exception to this rule are consortia sites using either model 2 (Using a Shared Instance) or model 3 (Single SFX Instance - Multiple Institutions Sharing the Same Instance):
    - In model 3 (Single SFX Instance - Multiple Institutions Sharing the Same Instance), each consortium member should be represented via an institute. One Ebsco API user ID should be defined per institute and the Ebsco targets should be activated per institute (not for default).
    - In model 2 (using a shared instance), the Ebsco API is invoked only in the instance responsible for creating the SFX menu (in most cases this will be the local instance, not the shared instance). For Ebsco targets activated in the shared instance, the Ebso API user ID should be added in the corresponding local instances (even if they are inactive).
- If the Ebsco plug-in is used, only one Ebsco package of the same type is displayed in the SFX menu. For example, if you have a subscription to both Ebsco Academic search Complete and Ebsco Academic search Elite, only the Ebsco Academic search Elite target is displayed in the SFX menu. The link returned by the Ebsco API is to the one with the lowest database code alphabetically. In case of a subscription to different types of Ebsco packages (for example, Academic search Elite and Business Source Premier), this restriction does not apply. Both packages are displayed in the SFX menu.
- For some Ebsco portfolios, the year information in different SFX sources varies, making it problematic to do a pre-check via the EBSCO API. For these portfolios, an exception was added to not use EBSCO API to check EBSCO targets before it is displayed in the SFX menu – based on a flag in the Object Portfolio: `parse_param: exception=no_date`.

15

Institute Feature

This section includes:

Overview of the Institute Feature

The institute feature in SFX is used to store information about the context of the user and more specifically, the institution to which the user belongs. This is useful in consortia or group environments, when the context of the user has ramifications for the access to different SFX services.

The following is a description of the institute workflow:

1. When SFX receives an OpenURL, it determines the institute to which the SFX user belongs.
2. SFX includes the value of the institute in the ContextObject.
3. SFX checks the SFX KnowledgeBase for specific institute settings. Some targets, for example, may be available for only certain institutes.
4. Based on the information in step 3, an SFX menu appropriate for the institute of the SFX user is presented.

Currently, institute information about the SFX user can be derived from the following:

- A request that is forwarded from one instance to another in a consortium model
- A qualified base URL: http://<sfx_server>:<port>/<instance>-<institute>

- Part of an OpenURL element named `sfx.institute`
- The user's IP address
- A cookie set using the `user_profile` cookie script (For more information, see Setting user_profile Cookies on page 207).
- Part of an OpenURL element named `req.institutes`

The order in which the different methods are listed is also the order in which SFX checks for institute information.

Note that it is possible to change the order of the IP and cookie method checking. This may be useful in the following scenario: A user comes from an IP defined as part of an IP institute and has a cookie that identifies him as belonging to a different institute. By default, the system determines affiliation by IP (first method checked by SFX where institute affiliation can be established). If the cookie should take preference, the order can be changed by editing the `config/institute.config_` file.

To have SFX check the cookie first, the `select_inst_by` section should be configured as follows:

```
Section "select_inst_by"
        IP      "0"
        cookie  "1"
EndSection
```

To have SFX check the IP first, the `select_inst_by` section should be configured as follows (this is the default setting):

```
institute.config_, with this section:
Section "select_inst_by"
        IP      "1"
        cookie  "0"
EndSection
```

Defining Institutes and Institute Groups

When using the institute feature in SFX, the first step is to define institute names in the Institute section of KBManager.

Select **Institutes** on the main page of the SFX Admin Center to access the Institutes section.

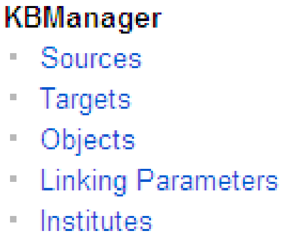


Figure 67: KBManager

The Institutes section can also be accessed from the top KBManager toolbar.

Initially, no institutes are defined in the Institute section. Click **Add Institute/ Group** to begin defining an institute.

In the Add Institute/Group window, specify the following:

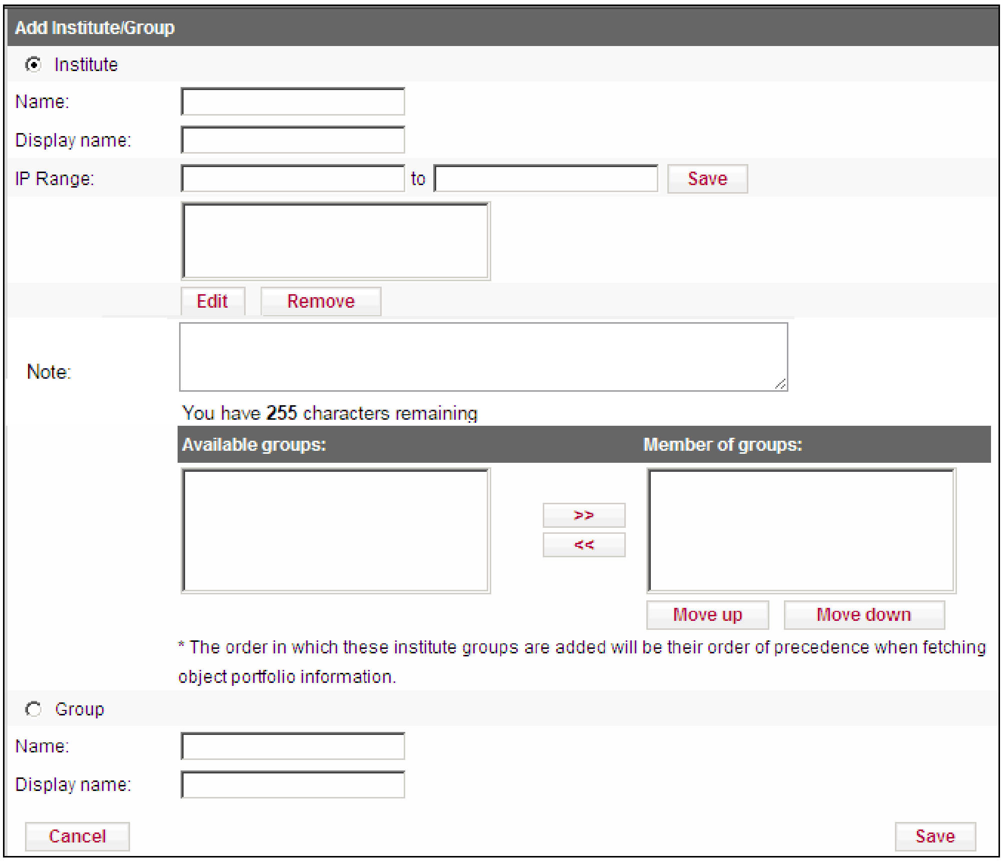
- The name of the institute (mandatory)

NOTES:

- The institute name cannot be changed once it is created. However, the display name (see below) can be changed after it is submitted.
- Do not use spaces or special characters such as !?‘;:" when choosing the institute name.

- A display name of the institute (mandatory)
- (Optional) The IP range(s) to used in identifying SFX users that are part of this institute. IP information must be specified only if IP is the method used to identify users. If cookies or consortia information are used to identify users, IP information is not required.
- (Optional) A note containing information about the institute.
- (Optional) Group information (the use of groups is explained in the following section).

Click **Save** at the bottom of the window to store the newly defined institute.



Add Institute/Group

● Institute

Name:

Display name:

IP Range: to

Note:

You have 255 characters remaining

Available groups: Member of groups:

* The order in which these institute groups are added will be their order of precedence when fetching object portfolio information.

○ Group

Name:

Display name:

Figure 68: Add Institute/Group

IP Information

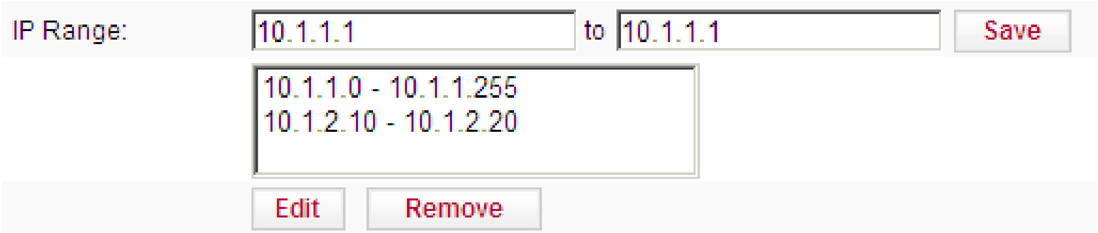
To define IP ranges, specify the start and end of the IP range and click **Save**. The information is then listed in the list box. It is possible to define multiple ranges per institute by repeating this process.

To remove or edit IP ranges, select the IP range in the list box and click the **Edit** or **Remove** buttons.

The following are examples of IP ranges:

- One IP address: `10.1.1.1-10.1.1.1`

- A range of IP addresses: `10.1.2.0-10.1.2.255; 10.1.3.10-10.1.3.21`

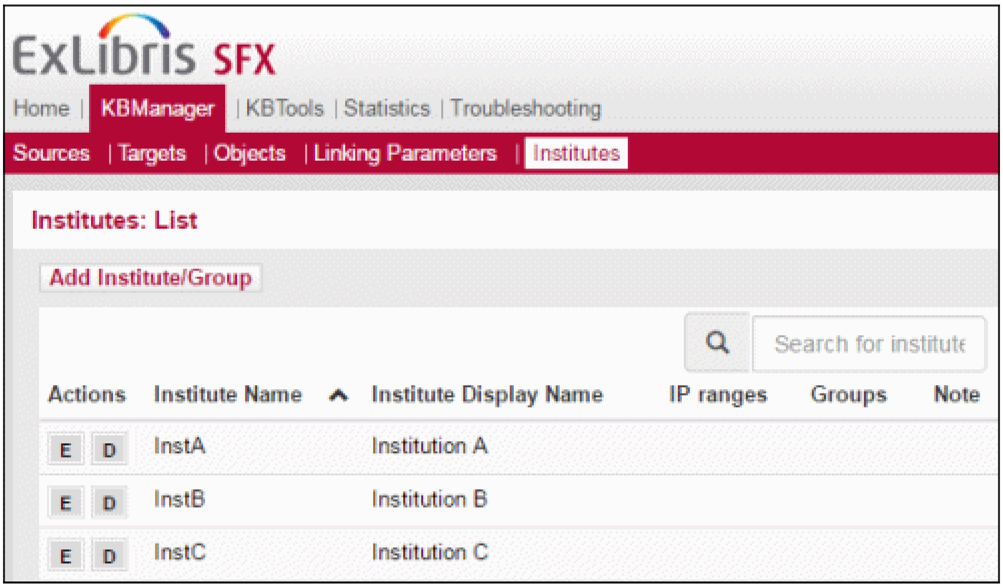


IP Range: 10.1.1.1 to 10.1.1.1

10.1.1.0 - 10.1.1.255
10.1.2.10 - 10.1.2.20

Figure 69: IP Range

Once the different institute names have been defined, the Institute section in KBManager will look like the following:



Institutes: List

Actions	Institute Name	Institute Display Name	IP ranges	Groups	Note
E D	InstA	Institution A			
E D	InstB	Institution B			
E D	InstC	Institution C			

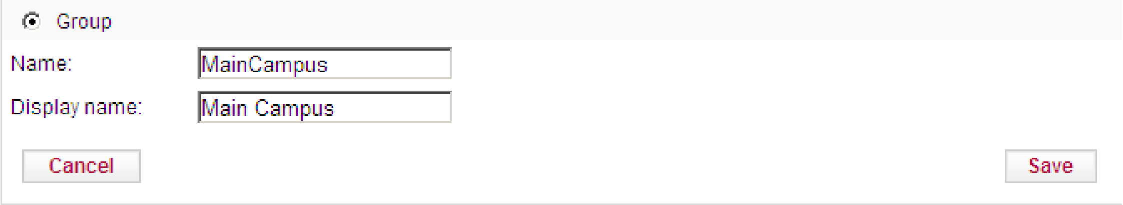
Figure 70: Institutes List

It is possible to group the different institutes. These groups can be used in KBManager in the same way as individual institutes—that is, you can activate institute-specific targets or define user name/password information specific to each institute or group.

To define a group:

1. Click the **Add Institute/Group** link.
2. Select **Group** at the bottom of the Add Institute/Group window.
3. Specify the group name.

4 Click **Save** at the bottom of the window to store the newly defined group.



● Group

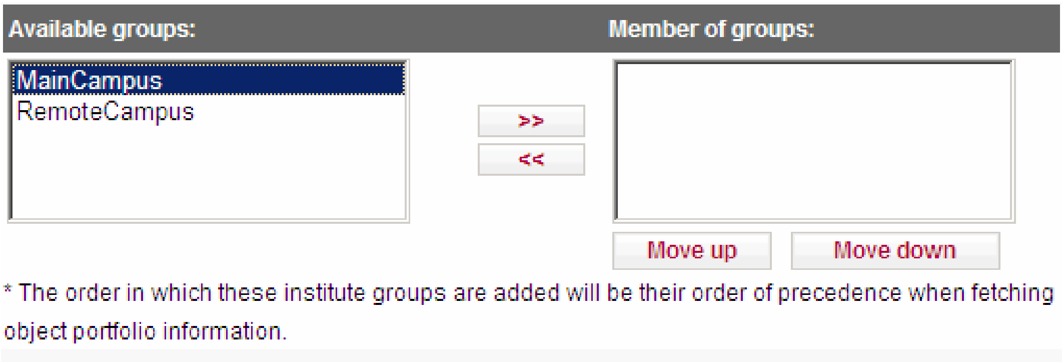
Name: MainCampus

Display name: Main Campus

Figure 71: Define a Group

To assign an institute to a group:

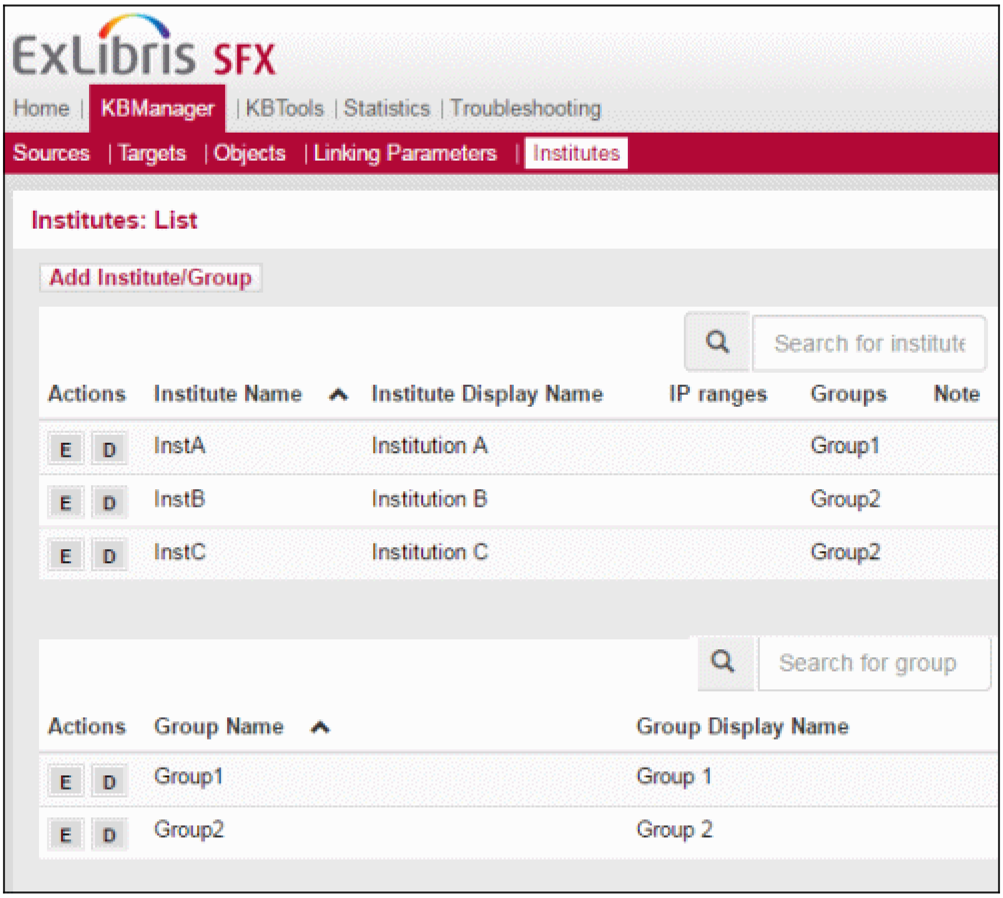
In the Edit or Add Institute/Group window, select a group in the **Available groups** list and use the arrows to move the group to the box on the right:



* The order in which these institute groups are added will be their order of precedence when fetching object portfolio information.

Figure 72: Select Group

After defining groups and assigning institutes to groups, the lists of institutes and groups looks like the following:


ExLibris SFX
Home | KBManager | KBTools | Statistics | Troubleshooting
Sources | Targets | Objects | Linking Parameters | Institutes
Add Institute/Group
Search for institute
Search for group

Institutes: List

Actions	Institute Name	Institute Display Name	IP ranges	Groups	Note
E D	InstA	Institution A		Group1	
E D	InstB	Institution B		Group2	
E D	InstC	Institution C		Group2	

Actions	Group Name	Group Display Name
E D	Group1	Group 1
E D	Group2	Group 2

Figure 73: Institute Updated

The Edit and Delete buttons can be used on each of the institutes and groups that have been defined.

NOTES:

- Deleting institute or group information automatically removes all the institute or group settings in the SFX database.
- It is possible to create different groups and assign an institute to multiple groups.
- The lists of institutes and groups are searchable and sortable.

Institute Feature Testing and Troubleshooting

After the institutes and groups have been defined, it is important to test whether the institute information is stored in the ContextObject. The test depends on the method in which institute information is passed on to the SFX server.

The User's IP Address

You can test the setup by creating a test institute with your own IP address.

When you send an OpenURL to the SFX server, you should see your test institute as one of the attributes in the ContextObject (the ContextObject data is shown in the HTML source for any SFX menu).

Using the user_profile Cookie Script

Use the following URL to set institute information using a cookie:

http://<sfx_server>:<port>/<instance>/cgi/public/user_cookie.cgi?usergroup=GUEST&language=NED&faculty=EXT&institute=test&source_id=&ml_session_id=&redirect= http://<sfx_server>:<port>/<instance>?rft.issn=0000-0019%26rft.date=2003

When you send an OpenURL to the SFX server, you should be able to see your institute test as one of the attributes in the ContextObject (the ContextObjectObject data is shown in the HTML source for any SFX menu).

Receiving Institute Information from Another SFX Instance

After the setup of consortium model 2 (where OpenURLs are forwarded to another instance), the `instance_self_name` is added to the ContextObject as an institute.

When you send an OpenURL to the SFX server, you should see your institute as one of the attributes in the ContextObject (the ContextObjectObject data is shown in the HTML source for any SFX menu).

Using Institute Information in KBManager to Store Information

Once the institute names have been defined, they can be used in KBManager when activating institute specific targets or defining user name/password information specific to each institute.

To define institute specific settings, use the **Institute** field in the TARGET, TARGET_SERVICE, or OBJECT_PORTFOLIO tables in KBManager.

It is possible to restrict or limit access for a particular TARGET, TARGET_SERVICE, or OBJECT_PORTFOLIO based on institute information.

The following general rules apply:

- If no institute or group information is assigned to a target, target service, or portfolio, the default activation status (active/inactive) is used.
- Specific institute or group information can be assigned to a target, target service, or portfolio. This option is used for the particular institute or group, if the availability information is different from the default activation settings.

For example:

- A specific target is available for everyone in a consortium sharing one SFX instance, except for one consortium member (defined as a specific institute to be called **arts**).

  You configure this by adding specific institute information for **arts** to the target and setting the target to **inactive** for **arts**. All other consortia members use the default setting (which is **active**).

  The following are some examples of the above scenario:

  - All libraries except the Science library of a particular institution have access to MCB Emerald journals.
  - All members of a library consortium except one have access to the Ebsco Academic Search Elite database.

- A specific target is available for only one member in a consortium sharing one SFX instance (defined as a specific institute to be called **arts**).

  You configure this by adding specific institute information for **arts** to the target and setting the target to **active** for **arts**. All other consortia members use the default setting (which is **inactive**).

  The following are some examples of the above scenario:

  - Only the Science library of a particular institution has access to Elsevier Science Direct journals.
  - Only a few members of a library consortium have access to Swets journals.

Institute thresholds can be set in the following database tables:

- TARGET
- TARGET_SERVICE
- OBJECT_PORTFOLIO

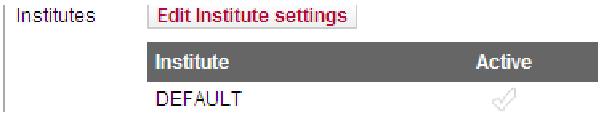
Setting Up Institute Localization

In KBManager, the Edit window of the TARGET, TARGET_SERVICE, and OBJECT_PORTFOLIO tables contains an **Institute** field.

NOTE:

> If no institutes have been defined in an SFX instance, the **Institute** field does not appear. It is always necessary to define the institute names first.

To configure institute-specific information, click the **Edit Institute Settings** link in the Edit window of a particular target, target service, or object portfolio.

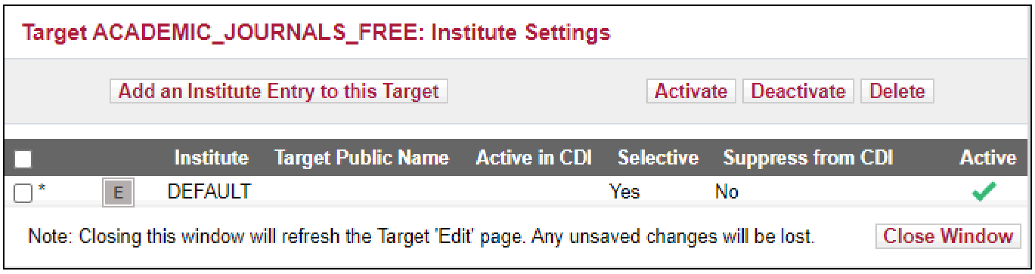


Institutes

Institute	Active
DEFAULT	✓

Figure 74: Edit Institute Settings

A list of currently defined institute settings appears in a separate browser window. In this window, it is possible to view, activate/deactivate, and remove institute-specific information. In the following example, no specific institute information has been added, so only the default activation is displayed:



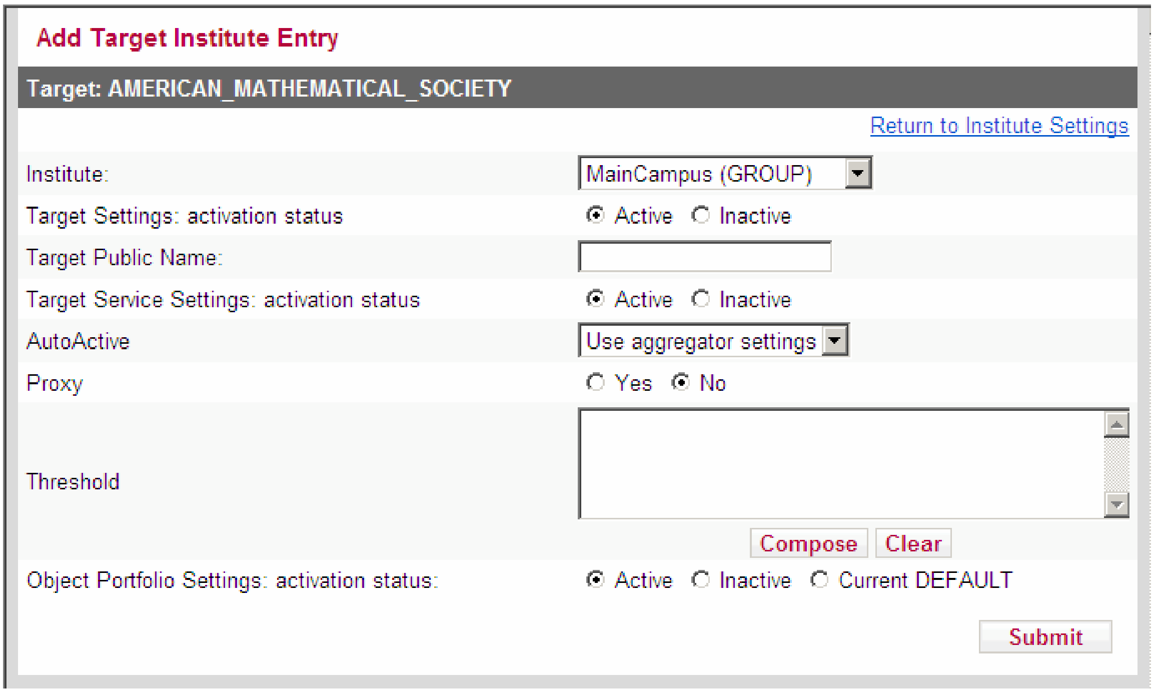
Target ACADEMIC_JOURNALS_FREE: Institute Settings

☐		Institute	Target Public Name	Active in CDI	Selective	Suppress from CDI	Active
☐ *	E	DEFAULT			Yes	No	✓

Note: Closing this window will refresh the Target 'Edit' page. Any unsaved changes will be lost.

Figure 75: Add an Institute Entry to This Target

To add institute settings, click the **Add an Institute Entry to this Target** link. Another window opens:



Add Target Institute Entry

Target: AMERICAN_MATHEMATICAL_SOCIETY

Return to Institute Settings

Institute:	MainCampus (GROUP)
Target Settings: activation status	● Active ○ Inactive
Target Public Name:	
Target Service Settings: activation status	● Active ○ Inactive
AutoActive	Use aggregator settings
Proxy	○ Yes ● No
Threshold	
	Compose Clear
Object Portfolio Settings: activation status:	● Active ○ Inactive ○ Current DEFAULT

Figure 76: Add Target Institute Entry

To define a new institute-specific entry:

1 Choose an institute or group name.

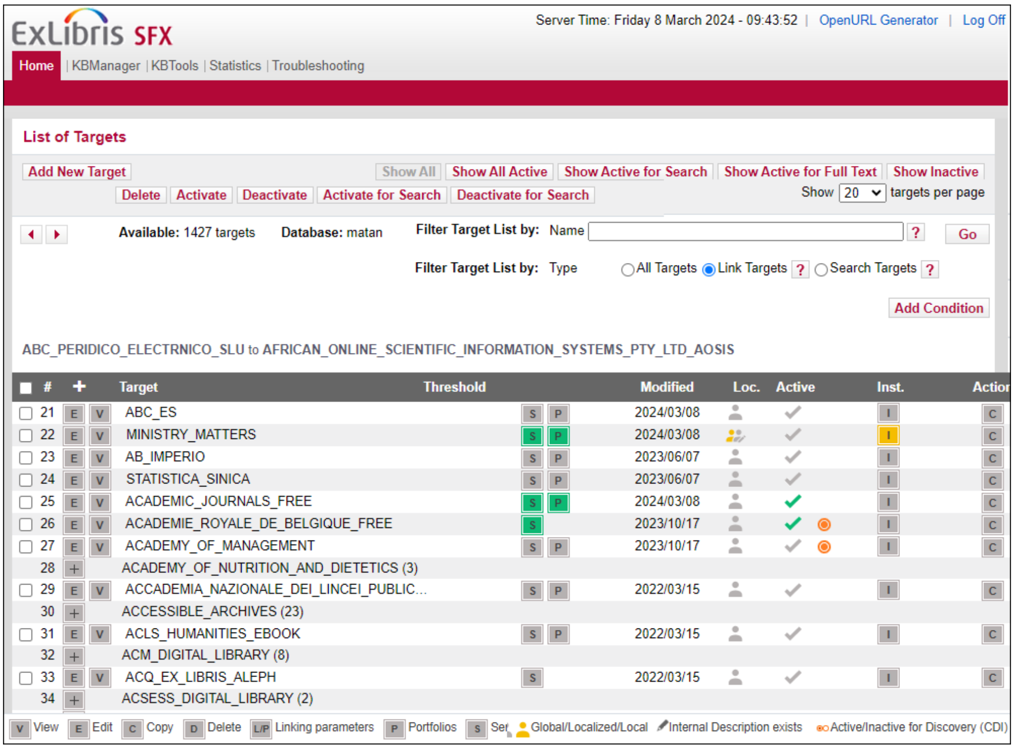
2 Define the availability information of target/ target service/object portfolio for this institute or group.

3 If you are defining a target or target service, you must also set the autoActive and Proxy settings.

4 Click **Submit** to save the new institute information.

Viewing Institute Localization

If institute-specific information has been defined for a target, target service, or object portfolio, the institute column is displayed. If there are no active institute settings, the **I** in the column is gray. If there are active institute settings, the **I** is yellow. Clicking the **I** opens the Institute Information window for this target.



Server Time: Friday 8 March 2024 - 09:43:52 | OpenURL Generator | Log Off

Home | KBManager | KBTools | Statistics | Troubleshooting

List of Targets

Add New Target | Show All | Show All Active | Show Active for Search | Show Active for Full Text | Show Inactive

Delete | Activate | Deactivate | Activate for Search | Deactivate for Search

Show 20 targets per page

Available: 1427 targets **Database:** matan

Filter Target List by: Name ? Go

Filter Target List by: Type ○ All Targets ● Link Targets ? ○ Search Targets ?

Add Condition

ABC_PERIDICO_ELECTRNICO_SLU to AFRICAN_ONLINE_SCIENTIFIC_INFORMATION_SYSTEMS_PTY_LTD_AOSIS

□	#	✚	Target	Threshold	Modified	Loc.	Active		Inst.	Actior
□	21	E V	ABC_ES	S P	2024/03/08	👤	✓		I	C
□	22	E V	MINISTRY_MATTERS	S P	2024/03/08	👥	✓		I	C
□	23	E V	AB_IMPERIO	S P	2023/06/07	👤	✓		I	C
□	24	E V	STATISTICA_SINICA	S P	2023/06/07	👤	✓		I	C
□	25	E V	ACADEMIC_JOURNALS_FREE	S P	2024/03/08	👤	✓		I	C
□	26	E V	ACADEMIE_ROYALE_DE_BELGIQUE_FREE	S	2023/10/17	👤	✓	●	I	C
□	27	E V	ACADEMY_OF_MANAGEMENT	S P	2023/10/17	👤	✓	●	I	C
	28	+	ACADEMY_OF_NUTRITION_AND_DIETETICS (3)							
□	29	E V	ACCADEMIA_NAZIONALE_DEI_LINCEI_PUBLIC...	S P	2022/03/15	👤	✓		I	C
	30	+	ACCESSIBLE_ARCHIVES (23)							
□	31	E V	ACLS_HUMANITIES_EBOOK	S P	2022/03/15	👤	✓		I	C
	32	+	ACM_DIGITAL_LIBRARY (8)							
□	33	E V	ACQ_EX_LIBRIS_ALEPH	S	2022/03/15	👤	✓		I	C
	34	+	ACSESS_DIGITAL_LIBRARY (2)							

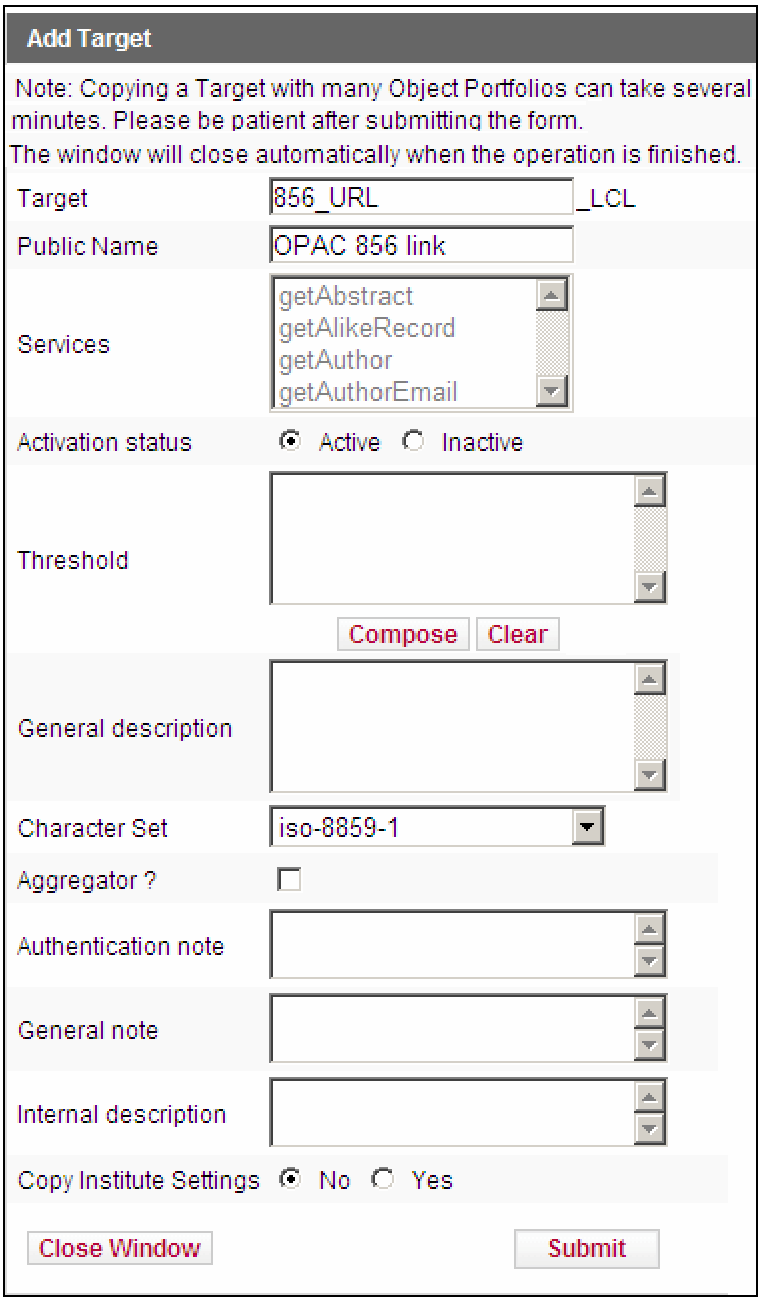
V View E Edit C Copy D Delete L/P Linking parameters P Portfolios S Ser 👤 Global/Localized/Local ✎ Internal Description exists ●○ Active/Inactive for Discovery (CDI)

Figure 77: List of Targets

The **Show Active**, **Show All**, and **Show Inactive** filter buttons in the list of targets, target services, and portfolios will take into account availability information from default and institute-specific settings. For example, even though the Wiley target is not active, the target is active for one of the institutes defined in the SFX instance. As a result, the target is listed if you click the **Show Active** button.

Copying Institute Settings When Copying a Target, Target Service, or Object Portfolio

In KBManager, when copying a target, target serice, or object portfolio, there is an option to copy institute settings.



Add Target

Note: Copying a Target with many Object Portfolios can take several minutes. Please be patient after submitting the form.
The window will close automatically when the operation is finished.

Target	856_URL _LCL
Public Name	OPAC 856 link
Services	getAbstract getAlikeRecord getAuthor getAuthorEmail
Activation status	● Active ○ Inactive
Threshold	
General description	
Character Set	iso-8859-1
Aggregator ?	□
Authentication note	
General note	
Internal description	
Copy Institute Settings	● No ○ Yes

Figure 78: Copy Institute Settings

Select **Yes** to copy the institute activation and institute specific threshold information to the newly created target, target service, or object portfolio.

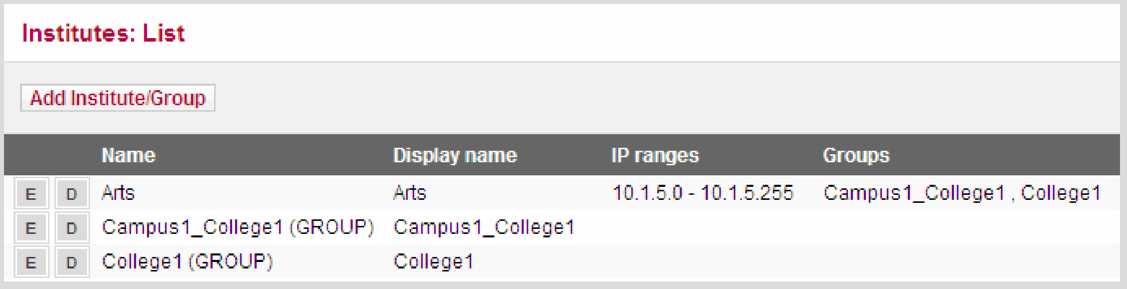
Storing Institute Information in Different Database Tables

It is possible to combine institute settings in different database tables and to assign both group and institute-specific information. To make sure the SFX software can handle a situation where availability settings exist for an institute and one or more groups to which an institute belongs, a feature called **Ranking** is used.

Ranking works in the following way:

- If institute settings exist for a specific target, target service, or object portfolio, these take preference over any group settings.
- If multiple group settings have been defined, one group takes preference over another by taking into account the order in which an institute has been assigned to groups.

For example, the institute **arts** is first part of the **Campus1_College1** group**,** and then the **College1** group.

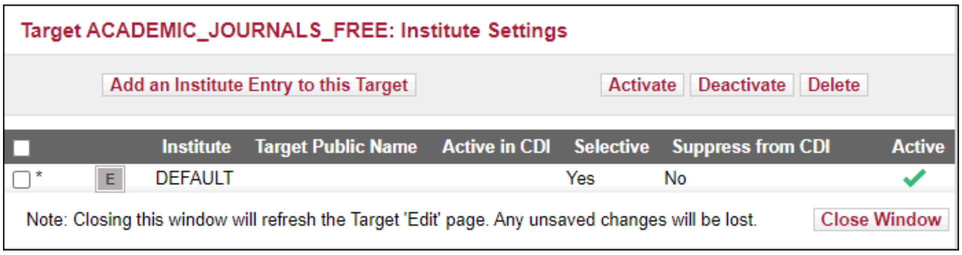


Institutes: List

		Name	Display name	IP ranges	Groups
E	D	Arts	Arts	10.1.5.0 - 10.1.5.255	Campus1_College1 , College1
E	D	Campus1_College1 (GROUP)	Campus1_College1		
E	D	College1 (GROUP)	College1		

A particular target has the following institute-specific activation:

- Active setting for **Campus1_College1**
- Inactive setting for **College1**



Target ACADEMIC_JOURNALS_FREE: Institute Settings

■		Institute	Target Public Name	Active in CDI	Selective	Suppress from CDI	Active
□ *	E	DEFAULT			Yes	No	✓

Note: Closing this window will refresh the Target 'Edit' page. Any unsaved changes will be lost.

Figure 79: Target: Institute Settings

The SFX software will take into account the active setting for **Campus1_College1** and ignore the inactive setting for **College1** .

Setting user_profile Cookies

An external database can send user information to SFX using a cookie named **user-Profile**.

To achieve this, the external database should send a request to the SFX server for the creation of the cookie with the appropriate user information in the following way:

http://<sfx_server>:<port>/<sfx_instance>/cgi/public/user_cookie.cgi?usergroup=<user_group>&language=<language>&faculty=<faculty>&institute=<institute>&redirect=<url_to_go_back_to>

NOTE:

For `language`, use the values between quotes in the following section of the `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/config/map_languages.config_` file.

```
Section "sfx_language_code_to_sfxmenu"
      dut "Dutch"
      spa "Spanish"
      eng "English"
EndSection
```

For example:
http://demo.exlibrisgroup.com:3210/demo/cgi/public/user_cookie.cgi?usergroup=GUEST&language=English&faculty=LAW&institute=test&redirect= http://demo.exlibrisgroup.com:3210/demo?issn=0000-0019%26date=2003

Once SFX receives the above request, it creates a cookie named **user-Profile** that contains the content of the user group, faculty, language, and institute fields and redirects the user to the URL specified in the **redirect** field.

Every time a user accesses SFX, SFX attempts to read the user-Profile query. If the cookie is available, SFX adds the content of the cookie to the appropriate ContextObject attribute:

- Faculty maps to the **Faculty** attribute.
- Institute maps to the **Institute** attribute.
- Language maps to the **Language** attribute.
- User group maps to the **UserGroup** attribute.

To test the cookie setting:

1 Use the following URL:

http://<sfx_server>/<sfx_instance>/user_cookie?usergroup=<user_group>&language=<language>&faculty=<faculty>&institute=<institute>&redirect=<url_to_go_back_to>

where all the info between <> should be replaced with actual information.

2 Send an OpenURL to your instance with some metadata. For example:

http://<sfx_server>:<port>/<sfx_instance>?genre=journal&issn=0000-0019

3 Check the HTML source of the SFX menu screen and verify that the ContextObject now contains usergroup, language, faculty, and institute information.

For example:

```
<!-- <ctx_object_1>$VAR1 = bless( {
      |req.user_group| => |GUEST|,
      |req.faculty| => |LAW|,
      |req.language| => |English|,
      |@req.institutes| => [|test|],
       }, |ContextObject::Generic| );
</ctx_object_1> //-->
```

16

ContextObject

This section includes:

Overview of the ContextObject

The ContextObject is used in the following ways in SFX:

- As a metadata container
- To support the evaluation of thresholds that are set in the SFX database
- To support the computation of link-to URLs via target parsers

The ContextObject is an instance of a Perl module named ContextObject::Generic. This instance is created by the `sfxmenu.cgi` CGI program when it parses the OpenURL.

Metadata from the OpenURL and from additional source parsing is stored in the ContextObject::Generic instance. As such, the ContextObject::Generic instance is a container of metadata of the object for which services have been requested.

The ContextObject::Generic instance is also used to evaluate thresholds when computing the bundle of services that are available for the object. Thresholds set in the SFX database using KBManager are conditions on the content of a ContextObject::Generic instance.

For instance, if the following threshold is set for the **Highwire** target:

```
$obj->need("year",">",1999)
```

the Highwire services can be displayed only if the ContextObject::Generic instance **$obj** contains a **year** attribute with a value greater than **1999**.

The `login.cgi` program uses the ContextObject::Generic instance to launch a TargetParser module. This module will use the attributes of the ContextObject::Generic instance to create a URL that links to a target resource.

When the SFX menu is presented to the user, the ContextObject information is printed in the HTML source for debugging purposes. More information about using the ContextObject to debug SFX problems can be found in **Troubleshooting and Debugging** on page **135**.

Transporting Metadata: OpenURL Syntax Description

More information about the OpenURL standard can be found at: http://library.caltech.edu/openurl/ and http://www.niso.org/ (standard Z39.88)

Hybrid OpenURLs

In SFX, if an OpenURL contains both 1.0 and 0.1 elements, the 0.1 elements will be ignored (1.0 being the preferred format).

A URL is treated as 1.0 if either `url_ver=Z39.88-2004&` or `ctx_ver=Z39.88-2004` elements exist.

OpenURL v1.0: SFX Specific Elements

The following section describes OpenURL v1.0: SFX specific elements.

SFX KB Object ID

It is possible to include an SFX object ID from the SFX KB in an OpenURL. Including this element will result in an SFX menu for the specific object from the SFX KB. This is useful in cases where no other unique identifiers (such as ISSN) are available, to reduce the occurrence of multi-object menus.

Element to include: `rft.object_id`

For example:

http://demo.exlibrisgroup.com:3210/demo?url_ver=Z39.88-2004&url_ctx_fmt=infofi%2Ffmt%3Akev%3Amtx%3Actx;ctx_ver=Z39.88-2004;ctx_enc=info%3Aofi%2Fenc%3AUTF-8;rfr_id=info%3Asid%2Fsfxit.com%3Akbmanager&sfx.ignore_date_threshold=1&rft.object_id=954925388218

NOTES:

- This element is used by SFX sources such as the SFX A-Z List and the **Test SFX** button in KBManager.
- When used in combination with DirectLinking set to **yes** and a serviceType restriction, this parameter ensures that a particular OpenURL links directly to one service (example: FullText) provider, without displaying an SFX menu. This occurs because only one service appears in the SFX menu, triggering DirectLinking.

For example:

http://demo.exlibrisgroup.com:3210/demo?url_ver=Z39.88-2004;url_ctx_fmt=infofi/fmt:kev:mtx:ctx;ctx_enc=info:ofi/enc:UTF-8;rfr_id=info:sid/sfxit.com:azlist;sfx.ignore_date_threshold=1;rft.object_id=954925388218;svc.fulltext=yes

SFX Object Portfolio ID

It is possible to include an SFX object portfolio ID in an OpenURL. This restricts the SFX menu to display only one service: the service associated with the object portfolio ID specified in the OpenURL.

Element to include: `rft.object_portfolio_id`

For example:

http://demo.exlibrisgroup.com:3210/demo?url_ver=Z39.88-2004&url_ctx_fmt=infofi%2Ffmt%3Akev%3Amtx%3Actx;ctx_ver=Z39.88-2004;ctx_enc=info%3Aofi%2Fenc%3AUTF-8;rfr_id=info%3Asid%2Fsfxit.com%3Akbmanager&sfx.ignore_date_threshold=1&rft.object_id=954925388218&rft.object_portfolio_id=110981564625337

NOTE:

When used in combination with DirectLinking set to **yes**, this parameter ensures that a particular OpenURL links directly to one provider, without displaying an SFX menu. This occurs because only one service appears in the SFX menu, triggering DirectLinking.

Ignore SFX KB Date Thresholds

The SFX server, when calculating which services to include in the SFX menu, takes into account any thresholds defined in the SFX KB.

For certain sources, this may not be appropriate—for example, if these sources are known not to include date information. This is the case, for example, for

OPAC systems, where the records for journals may be related to the print version and not the electronic one.

To make sure services are not excluded from the SFX menu when coming from these sources, it is possible to include an element that triggers SFX not to take into account date thresholds in the SFX KB.

Element to include: `sfx.ignore_date_threshold=1`

For example:

http://demo.exlibrisgroup.com:3210/demo?url_ver=Z39.88-2004&url_ctx_fmt=infofi%2Ffmt%3Akev%3Amtx%3Actx;ctx_ver=Z39.88-2004;ctx_enc=info%3Aofi%2Fenc%3AUTF-8;rfr_id=info%3Asid%2Fsfxit.com%3Akbmanager&sfx.ignore_date_threshold=1&rft.object_id=954925388218&rft.object_portfolio_id=110981564625337

Type of Response

NOTE:
This element can be used only for XML 1.0 OpenURLs (SFX API).

Element to include: `sfx.response_type`

Possible Values: `html/sfx_api_0_1_xml/simplexml/multi_obj_xml/service_exist`

- `html` – SFX presents the SFX menu.
- `sfx_api_0_1_xml` – SFX responds in the OpenURL 0.1 XML response. (This format was the format available in SFX version 2.0 API and is intended for backward compatibility.)
- `simplexml` – SFX responds in the OpenURL 1.0 XML response, presenting in XML format a list of targets with target URLs for all services that should appear in the SFX menu. `simplexml` format supports XML files that contain only one ContextObject.
- `multi_obj_xml` – The `multi_obj_xml` format is the same as the `simplexml` format, except that it supports XML files containing one or more ContextObjects.
- service_exist – SFX responds by indicating whether services exist for a particular ContextObject. This format supports XML files that contain one or more ContextObjects. No target URLs or details about services are provided, which makes this format lightweight and fast. It can be used in combination with the ServiceType restriction (for instance FullText) to include full-text indicator buttons for records with full-text in SFX.

The following is an example of a response:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ctx_obj_set>
  <ctx_obj index="02" id="954925373792">
  <service_exist>
  <services>no</services>
  </service_exist>
  </ctx_obj>
  <ctx_obj index="01" id="954925469303">
  <service_exist>
  <services>yes</services>
  </service_exist>
  </ctx_obj>
</ctx_obj_set>
```

ContextObject Attributes

The current implementation of the Generic::Request instance includes all tags that are defined in the OpenURL 1.0 standard (SAP1 and SAP2). Additionally, the ContextObject contains OpenURL 0.1 tags (to ensure backward compatibility) and a number of SFX-specific attributes.

A list of all currently supported attributes can be found in the `/exlibris/sfx_ver/sfx4_1/<instance>/config/ctx_attribute.config_` file. This file lists all attributes that can be set by a ContextObject::Generic instance. The first column specifies the name of the attribute. The second column is a regular expression giving the boundary conditions on the attribute. If the values stored in the ContextObject::Generic do not match the regular expression, these values are ignored.

The suffix `config_` is important. The underscore at the end indicates that the files will not be copied to the local instances. Instead, a symbolic link is used from the `sfxlcl41` to the `sfxglb41` instance. The attributes mentioned above should not be edited.

Some attribute names start with a **@** sign. This is a reminder for the programmer that the value of these attributes is not a string but an array reference.

Support for additional local attributes can be added by editing the following configuration file on the server:

```
/exlibris/sfx_ver/sfx4_1/<sfx_instance>/config/
ctx_attrib_local.config
```

Resolution of OpenURLs containing Object Portfolio IDs

SFX is able to resolve OpenURLs that contain an object portfolio ID in addition to an ISSN or object ID. SFX restricts the menu services to the provider (target) connected to the portfolio. If direct linking is turned on (see the **Direct Link** section of the *SFX General User's Guide*), SFX forwards the user directly to the URL built by the target parser for that provider.

Example OpenURL including Object Portfolio ID

http://sfx.university.edu:3210/sfxlcl41?url_ver=Z39.88-2004&url_ctx_fmt=infofi/fmt:kev:mtx:ctx&ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-2004&rfr_id=info:sid/sfxit.com:azlist&rft.issn=0000-0019&rft.object_portfolio_id=111030372947000&svc.fulltext=yes&sfx.ignore_date_threshold=1

Components

Table 11. Components

Component	Description
`http://sfx.university.edu:3210/sfxlcl41?`	Base_URL for the SFX server
`url_ver=Z39.88-2004&url_ctx_fmt=infofi/fmt:kev:mtx:ctx&ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-2004 &`	OpenURL 1.0 administrative metadata (required)
`rfr_id=info:sid/sfxit.com:azlist &`	Referrer ID. Indicates the source of the OpenURL. Optional but recommended for statistics purposes. (The example shown will enable date availability to be shown in SFX menu, if direct linking is not enabled.)

`rft.issn=0000-0019 &`
or
`rft.object_id=954921332001 &`
	Value for ISSN or object ID. It is mandatory to use **one** of these identifiers in the OpenURL.

`rft.object_portfolio_id=111030372947000`	Object portfolio ID. This is a mandatory parameter.
`svc.fulltext=yes & sfx.ignore_date_threshold=1`	Additional specifications to 1) limit services to full-text only and 2) allow the date threshold to be passed regardless of the absence of the `specific rft.date` parameter in the OpenURL. **NOTES:** When restricting the type of services that are displayed in the SFX menu to full text only (using the OpenURL 1.0 format `svc.fulltext=yes`), the following SFX serviceTypes are included: ■ For the HTML menu: `getFulltxt`, `getSelectedFullTxt`, and `getMessageNoFullTxt` ■ For all other outputs: (including simplexml and other xml formats, serivce_exist, etc.) `igetFulltxt` and `getSelectedFullTxt`

Retrieving Object Portfolio IDs for Use in OpenURLs

SFX object portfolio IDs are available via export from the SFX KnowledgeBase. More information about export tools in general can be found in the **Export Tool** section of the *SFX General User’s Guide*.

Recommended exports that contain object portfolio IDs along with ISSNs and target information are:

- Advanced exports in text-tab-delimited
- XML formats

Exports can be run via the SFX Admin Center Web interface in the Export Tool GUI or via the UNIX command line with the `export.pl` script (described in **Command-Line Export Tool** on page **112**).

In the text export, the following columns are relevant:

- Column 4: ISSN
- Column 12: Object Portfolio ID

In the XML export, the relevant data is organized as follows:

Table 12. XML Report

ISSN	`<datafield tag="022" ind1="" ind2="">`
	`<subfield code="a">0009-2347</subfield>`

Object ID	`<datafield tag="090" ind1="" ind2="">`
	`<subfield code="a">954925376877</subfield>`

Object Portfolio ID	`<datafield tag="866" ind1="" ind2="">`
	`<subfield code="z">110996846962000</subfield>`

If relevant ISSNs and object portfolio IDs are placed in the OpenURL template below, actionable OpenURLs are formed.

```
<BASE_URL>?url_ver=Z39.88-2004&url_ctx_fmt=infofi/
fmt:kev:mtx:ctx&ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-
2004&rfr_id=info:sid/
sfxit.com:azlist&rft.issn=<ISSN>&rft.object_portfolio_id=<OBJ
ECT_PORTFOLIO_ID>&svc.fulltext=yes&sfx.ignore_date_threshold=
1
```

It is also possible to create a link with object ID, using the following OpenURL template:

```
<BASE_URL>?url_ver=Z39.88-2004&url_ctx_fmt=infofi/
fmt:kev:mtx:ctx&ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-
2004&rfr_id=info:sid/
sfxit.com:azlist&rft.object_id=<Object_ID>&rft.object_portfol
io_id=<OBJECT_PORTFOLIO_ID>&svc.fulltext=yes&sfx.ignore_date_
threshold=1
```

Image-Based Linking

When SFX receives an OpenURL with a proprietary parameter, SFX returns a .gif instead of an SFX menu. This .gif differs, depending on whether or not a particular type of service is present.

For example, SFX returns:

- a transparent .gif, if no full-text services are available (a transparent .gif is invisible to the end user, so it looks as if no image is returned)
- a full-text indicator button if full-text is available – Behind the full-text indicator button is a standard OpenURL with a service type, so direct linking to full-text can occur.

The proprietary parameter can be:

- For OpenURL 0.1: either `_response_type=image-large` or `_response_type=image-small`
- For OpenURL 1.0: either `sfx.response_type=image-large` or `sfx.response_type=image-small`

When used as a means to present full-text indicator buttons, the `_response_type=image` parameter is always used in conjunction with the `_service_type` parameter.

Configuration of the "look and feel" of the images is defined in each SFX instance, in the `config/response_type_img.config_` file.

IMPORTANT:

The only source currently using image-based linking is Elsevier Scopus. This feature can be used for sources other than Elsevier Scopus, but should be used only for individual records, not for result lists. Using this feature for more than one record at a time may result in slowness or even downtime of the SFX server, as this may drastically increase the number of SFX requests received by SFX.

Image-Based Linking for Each Search Result in the Scopus Brief Result Display

- OpenURL returns a (large, Scopus-specific) button if at least one full-text service is available.

  For example:

  http://demo.exlibrisgroup.com:9003/demo?sid=Elsevier:Scopus&issn=0003-0007&date=2002&__service_type=getFullTxt&__response_type=image-large

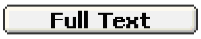
  

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_fulltxt.gif`

- OpenURL returns a transparent button if no full-text service is available.

  For example:

  http://demo.exlibrisgroup.com:9003/demo?sid=Elsevier:Scopus&issn=1520-765X&date=2002&__service_type=getFullTxt&__response_type=image-large

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_not_found.gif`

Image-Based Linking in the Journal Title List from Scopus

- OpenURL returns a small blue image if at least one full-text service is available.

  For example:

  http://demo.exlibrisgroup.com:9003/demo?sid=Elsevier:ScopusBrowse&issn=0003-0007&__service_type=getFullTxt&__response_type=image-small

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_fulltxt_sm.gif`

- OpenURL returns a small white image if no full-text is available.

  For example:

  http://demo.exlibrisgroup.com:9003/demo?sid=Elsevier:ScopusBrowse&issn=1520-765&__service_type=getFullTxt&__response_type=image-small

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_not_found_sm.gif`

Image-Based Linking Without a Specified ServiceType

- OpenURL returns Scopus-specific SFX button if at least one SFX service is available.

  For example:

  http://demo.exlibrisgroup.com:9003/demo?sid=Elsevier:Scopus&issn=1520-765X&date=2002&__response_type=image-large

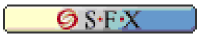
  

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_default.gif`

- OpenURL returns a transparent button if no SFX service is available.

  Image used: `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/templates/img/sfxmenu/service_not_found.gif`

Additional OpenURL 1.0 Functionality

If OpenURL contains sid/rfr.id information and SFX finds the corresponding source name, it is possible to use a source-specific image during image-based linking. If SFX finds a source-specific image, this image is used. If no source-specific image is found, the default image is shown.

Setup

Configuration file with images: `config/response_type_img.config_`

For example:

```
<base-url>?url_ver=Z39.88-2004&url_tim=2004-01-
09&ctx_ver=Z39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-
8&ctx_tim=2002-03-20T08%3A55%3A12Z&rft_id=info:oai/
arXiv.org:hep-th%2F0404069
```

The rft_id is: `info:oai/arXiv.org:hep-th%2F0404069`

SFX finds the corresponding SOURCE_ID: ARXIV

Add the following line to the section “`image`”:

```
getFullTxt_large.ARXIV    "$ENV{SFXCTRL_HOME}/templates/img/
sfxmenu/arxiv_fulltxt.gif"
```

The configuration file should look like the following:

```
Section "image"

default_large        "$ENV{SFXCTRL_HOME}/templates/img/
sfxmenu/service_default.gif"
default_small        "$ENV{SFXCTRL_HOME}/templates/img/
sfxmenu/service_default_sm.gif"
notFound_large    "$ENV{SFXCTRL_HOME}/templates/img/sfxmenu/
service_not_found.gif"
notFound_small   "$ENV{SFXCTRL_HOME}/templates/img/sfxmenu/
service_not_found_sm.gif"

getFullTxt_large     "$ENV{SFXCTRL_HOME}/templates/img/
sfxmenu/service_fulltxt.gif"

getFullTxt_large.ARXIV
"$ENV{SFXCTRL_HOME}/templates/img/sfxmenu/arxiv_fulltxt.gif"
```

As a result, an OpenURL with `sfx.response_type=image-large`:

```
<base-url>?url_ver=Z39.88-2004&url_tim=2004-01-
09&ctx_ver=Z39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-
8&ctx_tim=2002-03-20T08%3A55%3A12Z&rft_id=info:oai/
arXiv.org:hep-th%2F0404069&sfx.response_type=image-large
```

should result in the source-specific image specified in configuration file:

17

Local Source Configuration

This section includes:

The SFX Flow

When an OpenURL is received by SFX from a specific source, SFX performs one of the following actions:

- Interprets metadata sent in the OpenURL
- Augments the SFX request in a particular way for this specific source
- Shows an SFX menu with specific services on it

Configuration files and source parsers written for this specific source are stored on the SFX server and called when SFX receives an OpenURL from this source.

SFX scripts identify which source configuration files and which source parser to use in the following way:

OpenURL sid → source.dir → configuration files → source parsers

Metadata is transported to the SFX server via the OpenURL. The `Origin-Description/sid` (OpenURL 0.1) or `referrer-ID/rfr_id` (OpenURL 1.0) of an OpenURL contains information about the database from which the OpenURL originated. For example:

http://sfxserver.somewhere.edu/
sfxlcl41?sid=SilverPlatter:MX&pid=myid=12345678

The value of the `Origin-Description/sid` (OpenURL 0.1) or `referrer-ID/rfr_id` (OpenURL 1.0) is used for a lookup function in a file called `source.dir`. In turn, for each source listed in the `source.dir`, there is a reference to the location of a source configuration file. The configuration file for a specific source contains information about the location and the name of the source parser to be used, if any.

In SFX, which can interpret OpenURLs sent in either 0.1 or 1.0 format, there are corresponding 0.1 and 1.0 `source.dir` files, source configuration files, and/or source parsers depending on the format of the vendor or database's OpenURL. For more information about `source.dir` files, see **Source Identifiers (SIDs)** on page **224**. For more information about source configuration files, see **Source Configuration Files** on page **226**. For more information about source parsers, see **Source Parsers** on page **227**.

Transportation of Metadata to SFX

Metadata sent to SFX via an OpenURL 0.1 consists of two parts:

- an optional, yet highly recommended ORIGIN-DESCRIPTION
- a mandatory OBJECT-DESCRIPTION

The OBJECT-DESCRIPTION itself can contain up to three zones:

- a GLOBAL-IDENTIFIER-ZONE
- a OBJECT-METADATA-ZONE
- a LOCAL- IDENTIFIER-ZONE

The following is a graphical representation of the OpenURL:

```
        ORIGIN-DESCRIPTION  '&' OBJECT-DESCRIPTION
                        /    |    \
               ,-------'     |     `-------
GLOBAL-IDENTIFIER-ZONE       |       LOCAL-IDENTIFIER-ZONE
                             |
                  OBJECT-METADATA-ZONE
```

Since the SFX system relies heavily on metadata to deliver services, the intention is to obtain as much metadata as possible for an object for which services are requested.

The metadata is handled as follows:

- Part of that metadata is immediately available from the OBJECT-METADATA-ZONE of the OpenURL for this object. However, in many cases, additional metadata can be obtained by importing it from an information resource.
- If the OpenURL contains a GLOBAL-IDENTIFIER-ZONE, the metadata is imported from an authoritative namespace (for instance PubMed, DOI/X, and so forth). The tools to do this—that is, namespace-specific source parsers—come with the SFX package and should not be edited.
- If the OpenURL contains a resource-specific LOCAL-IDENTIFIER-ZONE, the metadata is fetched from the resource using resource-specific source parsers. These source parsers must reside in the following locations:
  - For OpenURL 0.1:

    /exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/OpenURL_0_1
  - For OpenURL 1.0:

    /exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/OpenURL_1_0
- If an OpenURL contains a LOCAL-IDENTIFIER-ZONE, metadata can be transported to the SFX server in one of three ways:
  - The LOCAL-IDENTIFIER-ZONE contains a record identifier by which a source parser can fetch a record from the origin-resource (described in ORIGIN-DESCRIPTION) via the Z39.50 or HTTP protocol.
  - The LOCAL-IDENTIFIER-ZONE contains object-metadata in a syntax that is specific for the origin-resource.
  - The LOCAL-IDENTIFIER-ZONE contains a combination of the above.

Example – OpenURL Sending Only a Record ID

The following is an example of an OpenURL that sends only a record ID to SFX:

http://sfxserver.somewhere/menu?sid=EBSCO:MRD&pid=myid:178912781

The ORIGIN-DESCRIPTION of this OpenURL is `sid=EBSCO:MRD`. This informs SFX to search for an `EBSCO:MRD` source parser on disk. SFX looks for this in `/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/OpenURL_0_1` (for OpenURL 0.1) or `/exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/OpenURL_1_0` (for OpenURL 1.0).

The LOCAL-IDENTIFIER-ZONE of this OpenURL is `pid=myid:178912781`. The EBSCO:MRD source parser receives the pid value `myid:178912781` as input and uses this record ID:

- to fetch a record from the EBSCO service

- to parse the record
- to fill out the attributes of a ContextObject (CTXO) instance

Example – OpenURL Sending Object Metadata

The following OpenURL sends object-metadata to SFX via a LOCAL-IDENTIFIER-ZONE. For example:

http://sfxserver.somewhere/menu?sid=XYZ:SoulNews&pid=ISSN%3D0012-1289%26VOL%3D45%26ISS%3D1%26YEAR%3D1974%26AUTH%3DBrown,%20James%26TITLE%3DI%20Feel%20Good

The ORIGIN-DESCRIPTION of the OpenURL informs SFX that `XYZ:SoulNews` should be used as a key to find an appropriate source parser on disk.

The LOCAL-IDENTIFIER-ZONE contains object-metadata, uses a syntax that is specific for the resource named `XYZ:SoulNews`, and is URL-encoded, as required. The decoded representation is:

```
ISSN=0012-1289&VOL=45

&ISS=1&YEAR=1974&AUTH=Brown, James&TITLE=I Feel Good
```

In this case, the XYZ:SoulNews source parsers must be able to read the specific syntax used by the resource XYZ:SoulNews and use this input to parse the content of the LOCAL-IDENTIFIER-ZONE and set the ISSN, volume, issue, year, @authLast, @authFirst, and articleTitle attributes of the ContextObject (CTXO).

Example – OpenURL Not Requiring a Specific Source Parser

The following OpenURL does not require a specific source parser:

http://sfxserver.somewhere/menu?issn=1234-5678&date=1998&volume=12&issue=2&spage=134&aulast=Brown&aufirst=James

This OpenURL does not contain a LOCAL-IDENTIFIER-ZONE. It contains object-metadata in the standardized syntax of the OBJECT-METADATA-ZONE. Therefore, no specific source parser is required to parse this URL.

Source Identifiers (SIDs)

The following two files include a list of all the source identifiers (SIDs) that SFX can currently handle when parsing OpenURLs:

- `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_v0_1.dir_`
- `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_v1_0.dir_`

NOTE:
Separate files are used for OpenURL 1.0 and OpenURL 0.1 formats.

For each source, the files contain:

- the start of the SID sent in the OpenURL
- the name and location of the configuration file

These files contain globally-recognized SIDs that are used for all Ex Libris customers. They are symbolically linked to the `sfxglb41` instance and are updated by Ex Libris with any new sources that are added to the SFX configuration. Do not break the symbolic link to these files.

Local versions of these files can be used to specify local SID information. The following are the locations of the SID directory files:

- For OpenURL 0.1:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_v0_1.dir_`

  includes the following file that can be locally modified:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_local_v0_1.dir`

- For OpenURL 1.0:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_v1_0.dir_`

  includes the following file that can be locally modified:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/source_local_v1_0.dir`

The following is an example of a portion of the `source_v0_1.dir` file:

```
FirstSearch.*               config/source/OpenURL_0_1/
                            firstsearch.config_
Gale.*                      config/source/OpenURL_0_1/
                            gale.config_
HWW.*                       config/source/OpenURL_0_1/
                            wilsonweb.config_
III:innopac                 config/source/OpenURL_0_1/
                            inov.config_
```

The first column is a regular expression to be matched against the `sid`. The second column contains the name and location of the source configuration file to be used.

Any additions to the local `source.dir` files, `source_local_v0_1.dir` and `source_local_v1_0.dir`, should follow the pattern displayed above.

Source Configuration Files

Source configuration files and source parsers are usually global in nature, and contain parameters that are relevant for all SFX customers. The source configuration files can be found in the following locations:

- For OpenURL 0.1:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/OpenURL_0_1`

- For OpenURL 1.0:

  `/exlibris/sfx_ver/sfx4_1/<instance>/config/source/OpenURL_1_0`

The source configuration file informs SFX which source parser to use in order to accomplish each of the following steps:

1. Parse the content of the LOCAL-IDENTIFIER-ZONE
2. Import a record from a link source, if necessary
3. Parse the record imported in step 2, if necessary.

The following is an example of a source configuration file:

```
# MetaLib configuration
#

Section "source"
        id              "METALIB"
EndSection

Section "parsePrivateID"
        module
"Parsers::SourceParser::OpenURL_0_1::metalib"
EndSection

Section "fetchRecord"
        module          "NetWrap::HTTP"
        host             $0
        timeout         "30"
EndSection

Section "parseRecord"
        module
"Parsers::SourceParser::OpenURL_0_1::metalib"
EndSection
```

The configuration file for each source is divided into four sections:

- The `source` section informs the SFX program of the origin of the SFX request. The `id` value in this section is used as a key for the lookup of a source in the SOURCE table of the SFX database.

- The `parsePrivateId` section informs the SFX program which Perl module must be dynamically loaded to parse the LOCAL-IDENTIFIER-ZONE. This Perl module must be a subclass of `SourceParser` and must contain a `parsePrivateId` function. As arguments, this function receives a `Generic::Request` instance, a HASH reference containing optional parameters, and the content of the OpenURL's `pid` field.
- The `fetchRecord` section sends a command to SFX as to which Perl module should be dynamically loaded to fetch a record from a link source. In the above example, the `NetWrap::HTTP` module will be used. This is a Perl module provided by SFX to fetch records from a Z39.50 database. SFX also provides a `NetWrap::Z3950` module to fetch records from databases that support Z39.50. If you want to implement your own `fetchRecord` module, it must be a subclass of SourceParser and must contain a `fetchRecord` function. As arguments, this function receives a `Generic::Request` instance and a HASH reference containing optional parameters. The `$` signs in the source configuration files are replaced with the actual values by breaking the symbolic link and making the configuration file a local one. For example, in the `metalib.config` file:

  ```
  $0 is replaced by: metalib.university.edu
  ```
- The `parseRecord` section informs SFX which Perl module must be dynamically loaded to parse the record returned from the `fetchRecord` module. This Perl module must be a subclass of `SourceParser` and must contain a `parseRecord` function. As arguments, this function receives a `Generic::Request` instance, a HASH reference containing optional parameters, and the return value of the `fetchRecord` module.

Any locally created source configuration files should follow the model shown above.

Source Parsers

The SFX package includes source parsers for all available sources. The following sources do not use a source parser, since the information providers have implemented only the metadata zone of the OpenURL and special interpretation of the OpenURL is not required for them:

ABC CLIO

AMS MATHSCINET

ANNUAL_REVIEW

ARXIV

BIOMEDCENTRAL

GALEGROUP INFOTRAC

HLAS

INGENTA

INSTITUTE OF PHYSICS JOURNALS

ISI RESEARCHSOFT ENDNOTE

BLACKWELL_SYNERGY
CIOS
CISTI SOURCE
CONTENTSCAN
DIALOG_AT_SITE
EBSCO
EI VILLAGE
ELSEVIER_SCIENCE_DIRECT
JSTOR6
NCBI_PUBMED
NISC BIBLIOLINE
PCI
RLG
WILSONWEB
ZETOC

A source parser is a Perl module that is a subclass of the `SourceParser` module. In the SFX environment, the source parsers can be found in the following location:

- For OpenURL 0.1:

  ```
  /exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/
  OpenURL_0_1
  ```

- For OpenURL 1.0:

  ```
  /exlibris/sfx_ver/sfx4_1/<instance>/lib/Parsers/SourceParser/
  OpenURL_1_0
  ```

The function of the `SourceParser` module is to set the attributes of the `ContextObject` instance.

The metadata to be translated into the attributes of a `ContextObject` instance originates from an OpenURL.

Location of the SFX Button

This is the button that information providers will typically display as the source link to the SFX menu.

- The SFX button can be accessed via this URL:

  http://<sfx_server_name>:port/<sfx_instance>/sfx.gif

  For example:

  http://demo.exlibrisgroup.com:8888/demo/sfx.gif

- The SFX button can be found in the following location on the server:

  ```
  /exlibris/sfx_ver/sfx4_1/<instance_name>/templates/img/
  sfxmenu/sfx.gif
  ```

- Additional SFX buttons can be found in the same directory. These buttons are customized for specific source databases:

  ```
  sfx.gif (default)width=54, height=19
  sfx_proquest.gifwidth=90, height= 32 (with transparent banner)
  sfx_firstsearch.gifwidth=45, height=16 (smaller size)
  ```

Tool to Verify Setup of PubMed and CrossRef/DOI Source Fetching

The purpose of this tool is to verify the status and setup of Pubmed and Crossref metadata fetching.

The following files are involved:

- `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/cgi/public/profile_checker.cgi`
- `/exlibris/sfx_ver/sfx4_1/<sfx_instance>/config/profile_checker.config_`

The XSD schema is:

http://demo.exlibrisgroup.com:9003/schema/profile_checker.xsd

The following usages are available:

- `<SFX_server>:<port>/<instance>/cgi/public/profile_checker.cgi?svc_dat=all_services`
- `<SFX_server>:<port>/<instance>/cgi/public/profile_checker.cgi?svc_dat=pubmed_service`
- `<SFX_server>:<port>/<instance>/cgi/public/profile_checker.cgi?svc_dat=doi_service`

For example:

http://demo.exlibrisgroup.com:9003/demo/cgi/public/profile_checker.cgi?svc_dat=all_services

Setup

Setup

It is recommended to provide contact information in case you encounter problems with Pubmed and Crossref/DOI source fetching.

To provide contact information :

1. Log on as the instance user (for example, `sfxlcl41`) .
2. Enter `cn`.
3. Enter `vi profile_checker.config_`
4. Enter your name, e-mail, and telephone information.

18

CookiePusher

This section includes:

Overview of CookiePusher

In order to allow delivery of context-sensitive services via an SFX framework, information providers must provide the following:

- The implementation of a technique to make the resource differentiate between a user with access to a service component that can deliver context-sensitive services and a user without access to a service component that can deliver context-sensitive services.
- An OpenURL for each metadata-object for users with access to a service component. For information on the OpenURL syntax, see **Transporting Metadata: OpenURL Syntax Description** on page 210.

Recognizing a User Who Wants SFX Services

In this section, the CookiePusher is described as one of many mechanisms through which a problem can be addressed. Inclusion of the SFX server location in a registered user profile can, for instance, be another way to deal with the problem. Yet another mechanism is to keep a table that links IP addresses to URLs of service components. Information providers who want to allow for context-sensitive services for their information resources can decide which mechanism best addresses a problem in their environment.

Introducing the CookiePusher

The CookiePusher offers a pragmatic solution whereby an information resource can be dynamically notified about the existence and the location of an SFX server. From the information contained in the OpenURL, SFX can deliver context-sensitive services to the end user. Input relating to the existence and location of the user's SFX server is essential for the information resource. It enables the resource:

- to include the OpenURL if the user has access to an SFX server, and to point the OpenURL to the appropriate URL—the **BASE-URL**—of the SFX server
- to not include an OpenURL if the user does not have access to an SFX server or another OpenURL-compliant service component

The idea behind the solution is that a user does not directly connect to an information resource. Rather, a detour is taken via a script that is installed in the same domain as the information resource. This script, the CookiePusher, takes as input certain parameters, two of which are essential for the functioning of the solution:

- `BASE-URL` – the BASE-URL of the SFX server (for more information, see **Transporting Metadata: OpenURL Syntax Description** on page **210**)
- `Redirect` – the desired address (URL) of the information resource

For example:

http://www.iop.org/EJ/sfx_cookie?BASE-URL=http://sfxserver.university.edu/sfx_local&Redirect=http://www.iop.org/EJ/

To use the CookiePusher mechanism, a simple adaptation at the side of the information resource is required, as well as at the side of the institution that runs the SFX server accessing the information resource.

SFX Sources Using the CookiePusher Script

The CookiePusher script is installed in the domain of the information provider. When a user of the information resource targets the CookiePusher, the information resource can determine the following:

- The location of the user's SFX server, read from the `BASE-URL` parameter of the URL by which the user connects
- The URL within the information system to which the user wants to connect, read from the `Redirect` parameter of the URL by which the user connects

Based on the first of the above, the information resource can register this specific user preference appropriately. For instance, it can set a long-lasting cookie in the user's browser, containing the `BASE-URL` of the user's SFX server. This enables the information resource to detect the user's preference from the browser during all future visits. This is the approach shown in the following sample CookiePusher script:

```
#!/sfx/app/perl/bin/perl
  #
  # Version:   $Id: pushcookie.cgi,v 1.1 2000/04/05 12:03:51
sfx Exp $
  #
  # Authors:   Patrick Hochstenbach,
<Patrick.Hochstenbach@rug.ac.be>
  #
  #
  use     CGI;
  my      $cookie_path     = '/cgi-bin';
  my      $cookie_expires = '+6m';  # Expires in 6 months
  my      $query           = new CGI;
  {
          my $BASE_URL = $query->param('BASE-URL');
          my $Redirect       = $query->param('Redirect');
          my $cookie         = $query->cookie(
                          -name    => 'user-OpenURL',
                          -value   => $BASE_URL,
                          -path    => $cookie_path,
                          -expires => $cookie_expires,
                          );
          print $query->redirect(
                          -uri    => $Redirect,
                          -cookie => $cookie
                          );
  }
```

The information system does not necessarily have to transform the knowledge of the BASE-URL of a user's SFX server into a cookie. For example, it can add the information to other session-related information or store it in the user's profile.

Parameters expressing preferences of the institution, other than BASE-URL and Redirect, can potentially be provided via the CookiePusher mechanism. However, during the initial deployment of the SFX solution, it is recommended that these preferences be hard-coded. The information provider is encouraged to support the following preferences:

- Insertion of an SFX button in the output of the information resource. For each SFX server, the clickable image to be hyperlinked with an OpenURL is always located at BASE-URL/sfx.gif .
- Opening of the SFX menu screen – Clicking the SFX button (the OpenURL) opens a browser window with the following properties:
  - window name = SFXmenu
  - location = no
  - status = yes

- menubar = no
- scrollbar = yes
- resizable = yes
- width = 460
- height = 420

Institutions Using an SFX Server with CookiePusher

If an information provider installs a CookiePusher script as a means to detect the institution's preferences regarding an SFX server, the institution must make sure it connects its users to this resource via its CookiePusher. This has to be done for all SFX sources that have implemented the CookiePusher script. There are two ways in which an institution can do this:

- Enhancement of the Institutional Menu System
- Creation of a Special HTML Page

Enhancement of the Institutional Menu System

An institution can enhance the institutional menu system so that it points to the information resource's CookiePusher script rather than to its start-up URL. The following example shows how an institution can use this technique and insert a

URL in its menu system that targets an information system via its CookiePusher.

```
If an institution's OpenURL-aware service component is
located at:
http://sfxserver.university.edu/sfx_local

If the URL used in the institution's menu-system to connect to
an information resource is:
http://www.info.com/cgi-bin/start.cgi

If this information resource's CookiePusher is at:
http://www.info.com/cgi-bin/pushcookie.cgi

Then registration of the location of the institution’s SFX
server within the information resource is achieved by editing
the institution's menu-system, changing the URL to:

http://www.info.com/cgi-bin/pushcookie.cgi?BASE-URL=http://
sfxserver.university.edu/sfx_local&Redirect=http://
www.info.com/cgi-bin/start.cgi

In reality, this URL must be UTF-8 encoded:
http://www.info.com/cgi-bin/pushcookie.cgi?BASE-URL=http%3A%2F%2F
sfxserver.university.edu%2Fsfx_local&Redirect=http%3A%2F%2Fww
w.info.com%2Fcgi-bin%2Fstart.cgi
```

Creation of a Special HTML Page

An institution can set up a special HTML page to which the users who want to take advantage of the SFX server are asked to connect. Connecting to this page automatically registers the location of the user’s SFX server in all the information resources that use the CookiePusher.

For each information resource, this HTML page should contain a link to an image that is stored on the information resource’s server. Instead of a direct link to the image, a detour is taken via the resource’s CookiePusher script.

When the HTML page loads and the image is requested from the information provider’s server, the cookie is set in the user’s browser. The following example shows how an institution can use this technique to create an HTML page that connects to the CookiePushers for all their OpenURL-enabled resources.

```
If an institution's SFX server is located at:
http://sfxserver.university.edu/sfx_local

If the institution has access to two information resources
that have installed a CookiePusher:
CookiePusher for resource 1 is at http://www.info.com/cgi-
bin/pushcookie.cgi
CookiePusher for resource 2 is at http://www.moreinfo.com/
cgi-bin/cookieset.cgi

Then the installation should locate a readily accessible
image in each of those information resources, for instance:
For resource 1 at http://www.info.com/images/info.gif
For resource 2 at http://www.moreinfo.com/gifs/welcome.gif

Registration of the location of the institution’s SFX server
within both information resources can be achieved by creating
a HTML page that contains the following lines:

<img src = "http://www.info.com/cgi-bin/
pushcookie.cgi?Redirect=http://www.info.com/images/
info.gif&BASE-URL=http://sfxserver.university.edu/sfx_local">


<img src = "http://www.moreinfo.com/gifs/
welcome.gif?Redirect=http://www.moreinfo.com/gifs/
welcome.gif&BASE-URL=http://sfxserver.university.edu/
sfx_local">

Again, these URLs must be UTF-encoded.
```

NOTE:

When using the CookiePusher mechanism, the privacy settings of the user’s browser are important, since the browser needs to allow the setting of third-party cookies. By default, the privacy settings do not allow the setting of third-party cookies. Ensure that you override the default settings and configure the browser to accept all cookies.

CookiePusher Compatibility

The default privacy preferences in Internet Explorer 6 and later prevent setting the BASE-URL cookie in a third-party context. Users setting cookies from DOI, arXiv, and IOP have reported this problem.

Description of Problem: Symptoms and Scenarios

- Setting the BASE-URL cookie by embedding the `cookiepusher.cgi` request in an image does not work in IE 6 and later. The browser shows the icon of an eye in the status bar. The HTML code that represents this scenario is:

```
<img src="http://www.doi.org/cgi-bin/pushcookie.cgi?BASE-
URL=http://demo.exlibris
group.com:9003/demo&Redirect=http://www.doi.org/images/
banner_foundation.gif">
```

- Setting the BASE-URL cookie by redirecting the user to `cookiephusher.cgi` within a Web page with frames does not work in IE 6 and later. The browser shows the icon of an eye in the status bar. The HTML code that represents this scenario is:

```
<!-- JavaScript begins ...
function jump (page) {
  window.location = page;
}
// JavaScript ends -->
<FORM NAME="jumpform">
<select name="pages"
onChange="window.status=this.options[this.selectedIndex].text
">
<option SELECTED value="http://sfx2.exlibrisgroup.com:9003/
cgibin/pushcookie.cgi?BASE-URL=http%3A//
sfx2.exlibrisgroup.com%3A9003/sfx_local&amp;Redirect=http%3A/
/www.sfxit.com/no_p3p_compliant.html">
Ex Libris SFX Server
</select>
</span>
<INPUT VALUE="Go" TYPE=BUTTON onClick =
"jump(document.jumpform.pages.options[pages.selectedIndex].va
lue)">
<br>
</FORM>
```

Technical Explanation of the Problem

According to the default IE 6 (and later) settings, which most users never change, third-party cookies are blocked when they do not have compact policies or when they have unsatisfactory compact policies. The BASE-URL cookie is blocked in the scenarios described above because they are being pushed in a third-party context and do not have P3P compact policies.

Third-Party Cookies

Cookies are associated with a Web page or with an image or other object embedded in a Web page. When a page or object is served, the server adds a special header that "sets" the cookie on the user's machine. Sometimes, Web pages include images, frames, or other content that is located on a site with a different domain name than the page in which it is embedded. For example, it is quite common for Web sites to embed banner advertisements that are served by an ad company. If any of these "third-party" images or objects set cookies, they are referred to as third-party cookies. Since the SFX CookiePusher resides in a domain such as doi.org and tries to push a cookie in a Web site such as library.edu, IE 6 (and later) considers it a third-party cookie.

Preventing IE 6 (and Later) from Blocking My Cookies

To prevent IE 6 (and later) from blocking cookies, you must make sure that all of the cookies, especially in a third-party context, have P3P compact policies associated with them, and that those compact policies are considered satisfactory by IE 6 (and later).

P3P

The full name for P3P is the Platform for Privacy Preferences Project. P3P 1.0 is an official "recommendation" of the World Wide Web Consortium (W3C) that was approved in April 2002. P3P provides a standard way for Web sites to encode their privacy policies in a computer-readable XML format. This allows P3P-enabled Web browsers and other P3P user agents to fetch P3P privacy policies automatically, parse them, and compare them with a user's privacy preferences. P3P user agents can use the information in a P3P policy to provide a summarized version of Web site privacy policies to users. For example, IE 6 (and later) offers a Privacy Report option from the View menu.

The P3P 1.0 Specification also defines an abbreviated version of a P3P policy—called a **compact policy**—that can be transmitted in HTTP headers when cookies are set. Some P3P-enabled browsers, such as IE 6 (and later), use the information in P3P compact policies to make cookie-blocking decisions.

Solution

A new version of the CookiePusher CGI, called `p3p_pushcookie.cgi`, has been created. This new version pushes not only the standard BASE-URL cookie, but sends P3P Compact Policies as part of the HTTP header. The new CGI is fully compliant with the IE 6 (and later) default privacy values. The two third-party scenarios described above are no longer a problem.

The compact policy implemented by the new CookiePusher CGI is:

```
CP="NOI DSP CURa ADMa DEVa TAIa OUR IND COM NAV OTC"
```

The following table explains the meaning of each field in the compact policy.

```
FieldMeaning
CP=This is the compact policy header; it indicates that what
follows is a P3P compact policy.
NOINo identifiable information is collected, so no access is
possible.
DSPThe policy contains at least one dispute-resolution
mechanism.
CURaThe data is used for completion of the current activity.
ADMaThe data is used for site administration.
DEVaThe data is used for research and development.
TAIaThe data is used for tailoring the site.
OURThe data is given to ourselves and our agents.
INDThe data will be kept indefinitely.
COMComputer information is collected.
NAVNavigation and clickstream data is collected.
OTCOther types of information are collected.
```

The P3P Compact Policy included in the `p3p_pushcookie.cgi` is enough to solve the problem with IE 6 (and later) and third-party cookies. Nevertheless, the standard P3P recommends that an XML and an HTML representation of the policy should be posted on the Web site where CGI resides. If someone wants an XML and HTML representation of the compact policy hard-coded in `p3p_pushcookie.cgi`, the following files have been created:

```
/exlibris/sfx_ver/sfx4_1/<instance>/templates/p3p_samples/
p3p.xml
/exlibris/sfx_ver/sfx4_1/<instance>/templates/p3p_samples/
sfx_p3p_cookiepusher.html
/exlibris/sfx_ver/sfx4_1/<instance>/templates/p3p_samples/
sfx_p3p_cookiepusher.xml
```

The P3P standard specifies that these files should be accessible via http://host.edu:9003/w3c/.

References

- W3C Platform for Privacy Preferences 1.0 Specification:

  http://www.w3.org/TR/P3P/
- O'Reilly Network: Help! IE 6 Is Blocking My Cookies:

  http://www.oreillynet.com/pub/a/javascript/2002/10/04/p3p.html

19

SFX APIs

This section includes basic information on the following API options. For details on each of these APIs, see the EL Commons Web site (www.exlibrisgroup.org):

- OpenURL
- Rapid Service Indicator
- Journal Subscription (Deprecated)

OpenURL

This API enables the sending of an OpenURL 1.0 request to SFX in XML format in order to receive context-sensitive linking information.

The response sent by SFX to the application that initiated the request is a response in XML format that includes the following:

- The metadata received by the SFX server and stored in the ContextObject
- SFX services, if they exist

XML requests should be sent to the following address/program:

```
<SFX server>:<port>/<sfx_instance>
```

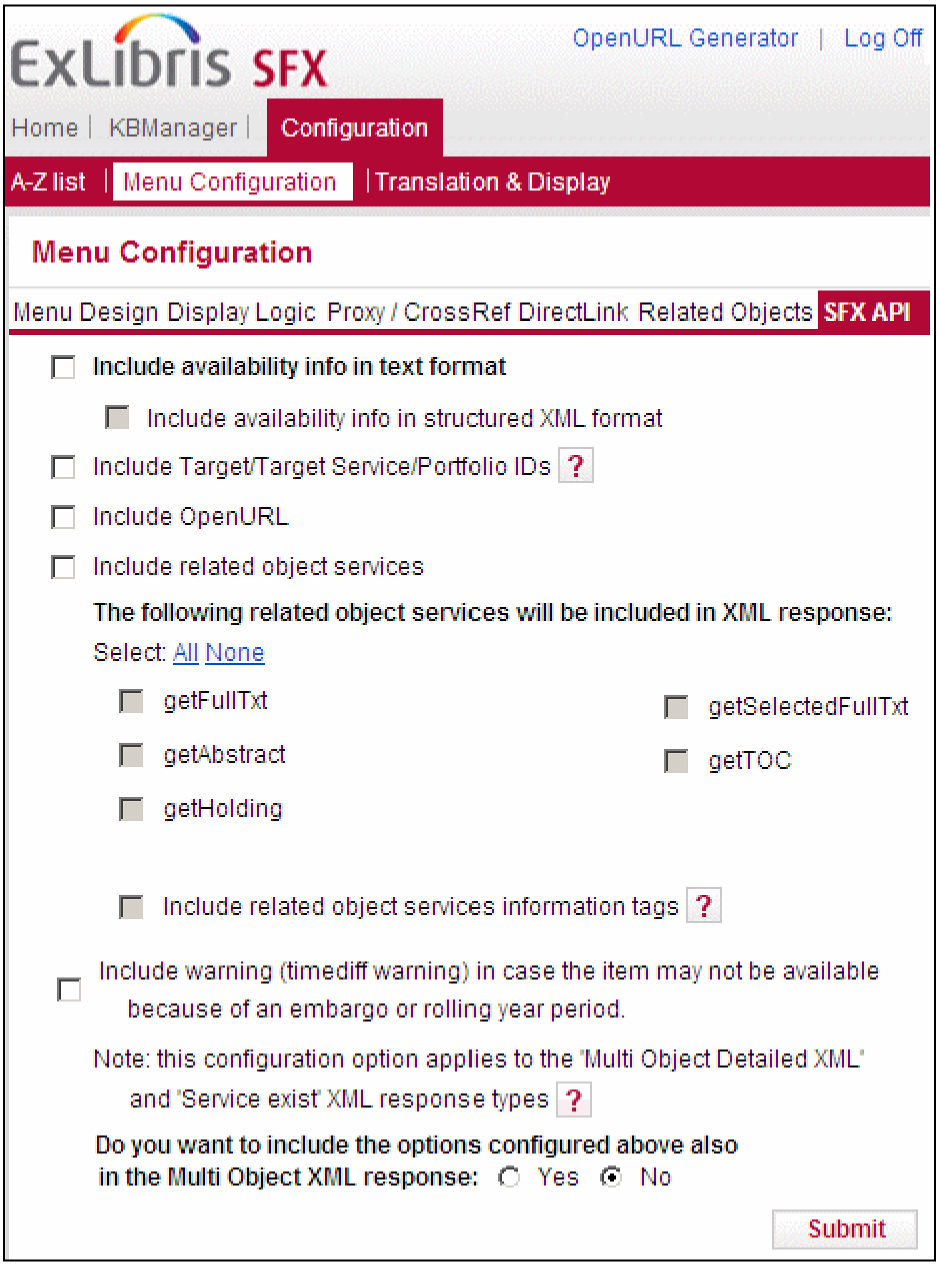
Configuring the Multi-Object Detailed XML Response Type

The Multi Object Detailed XML response type provides advanced options for the inclusion of additional details in the XML response. It is the most comprehensive response in XML format.

Details that can be added include:

- Availability (coverage statement) in the API response, both in structured XML and in a user-readable text string
- SFX KnowledgeBase target, target service, and portfolio identifiers.
- Journal relationship information in the API response and an indication when a target is for a close or remote related object

To configure the Multi-Object Detailed XML, click **Menu Configuration** from the Configuration section of the Setup & Administration area. Click the **SFX API** tab. The following is displayed:


ExLibris SFX
OpenURL Generator | Log Off
Home | KBManager | Configuration
A-Z list | Menu Configuration | Translation & Display
Menu Configuration
Menu Design Display Logic Proxy / CrossRef DirectLink Related Objects SFX API
Submit

□ **Include availability info in text format**

□ Include availability info in structured XML format

□ Include Target/Target Service/Portfolio IDs ?

□ Include OpenURL

□ Include related object services

The following related object services will be included in XML response:

Select: All None

□ getFullTxt □ getSelectedFullTxt

□ getAbstract □ getTOC

□ getHolding

□ Include related object services information tags ?

□ Include warning (timediff warning) in case the item may not be available because of an embargo or rolling year period.

Note: this configuration option applies to the 'Multi Object Detailed XML' and 'Service exist' XML response types ?

Do you want to include the options configured above also in the Multi Object XML response: ○ Yes ● No

Figure 80: Configuring the Multi-Object Detailed XML

This section allows the Multi-Object Detailed XML to include a number of additional XML tags in its response. Additionally, it is possible to configure the existing Multi-Object XML response to behave identically to the new Multi Object Detailed XML.

The following is a list of all advanced options that can be configured for the Multi-Object Detailed XML API response.

- **Include availability info in text format** – select this option to include availability (coverage statement) in the SFX Multi-Object API response in text format. For each target, all active parsedDate and timeDiff thresholds are included in the coverage statement. This is the same text string that appears in the A-Z list, SFX menu, and in XML advanced exports. In the following example, the text “Available from 1976 volume: 1 issue: 1 “is included as follows:

```
<coverage>
<coverage_text>
<threshold_text>
<coverage_statement> Available from 1976/01/01 volume: 1
issue:
1</coverage_statement>
</threshold_text>
</coverage_text>
</coverage>
```

- **Include availability info in structured XML format** – select this option to have the availability statement be in an XML structure containing informational threshold elements. For example:

```
<coverage>
<from>
<year>1976</year>
<month>01</month>
<day>01</day>
<volume>1</volume>
<issue>1994</issue>
</from>
<embargo />
</coverage>
```

- **Include Target/Target Service/Portfolio IDs** – select this option add the identification numbers of the target, target service and object portfolio for each target in the response. If this option is cleared, only the `target_service.internal_id` is included in the SFX API response. The following is an example of an XML response with this option selected:

```
<target>
  <target_name>FACTIVA</target_name>
  <target_public_name>Factiva</target_public_name>
  <object_portfolio_id>111037302894000</object_portfolio_id>
  <target_id>111037301464000</target_id>
  <target_service_id>111037301464002</target_service_id>
</target>
```

- **Include openURL** – Clear this option to exclude the element `<item key="sfx.openurl">` in the section `<ctx_obj_attributes>` of the XML response. This can reduce the size of the XML response greatly. We recommend excluding the OpenURL element except when needed in the API response, since its existence can significantly increase the size of the `<ctx_obj_attributes>` section.

- **Include related object services** – Select this option to include relation information in the Multi-Object Detailed XML API. If this option is cleared, no related object information is included in the XML. If this option is selected, target services for the orginal object sent in the OpenURL, as well as services for close and remotely related objects are included, with the following new tags:

    - For objects sent in the OpenURL: `<is_related>no</is_related>`
    - For closely related objects: `<is_related>close</is_related>`
    - For remotely related objects: `<is_related>remote</is_related>`

    Additionally, when this option is selected, the following options exist:

    - The following list of services can be defined for which the related object behavior should apply:
        - getFullTxt
        - getSelectedFullTxt
        - getAbstract
        - getTOC
        - getHolding
    - **Include related object services information tags** – select this option to include additional tags for remote relation types, such as the type of relation and the related object information.

    The following is an example of a target that is not for a related object:

```
<target>
  <target_name>LOCAL_CATALOGUE_EX_LIBRIS_ALEPH</target_name>
  <target_public_name>Ex Libris ALEPH catalog</
target_public_name>
  <target_service_id>110974980650101</target_service_id>
  <is_related>no</is_related>
</target>
```

The following is an example of a target for a close type of related object:

```
<target>
  <target_name>WILEY_INTERSCIENCE_ANALYT_SCIENCE_BACKFI</
target_name>
  <target_public_name>Wiley InterScience Analytical Sciences
   Backfiles</target_public_name>
 <object_portfolio_id>1000000000709579</object_portfolio_id>
  <target_id>1000000000000511</target_id>
  <target_service_id>1000000000001038</target_service_id>
  <is_related>close</is_related>
</target>
```

The following is an example of a target for a remote type of related object:

```
target>
  <target_name>PUBMED_CENTRAL_JOURNALS</target_name>
  <target_public_name>PubMed Central</target_public_name>
 <object_portfolio_id>1000000000548543</object_portfolio_id>
  <target_id>111058563444000</target_id>
  <target_service_id>111058563444001</target_service_id>
  <is_related>remote</is_related>
  <related_service_info>
     <relation_type>CONTINUES</relation_type>
     <related_object_issn>0267-0623</related_object_issn>
     <related_object_title>British medical journal</
related_object_title>
     <related_object_id>954927635438</related_object_id>
 </related_service_info>
</target>
```

- **Include warning (timediff warning) in case the item may not be available because of an embargo or rolling year period** – a warning is displayed if only year (no month/day) is sent in the request and that year is equal to the year the embargo/rolling year threshold for a specific porfolio starts.

  This option corresponds to the warning for the SFX menu descibed in the **Other Menu Configurations** section of the *SFX General User's Guide*.

- **Do you want to include the options configured above also in the Multi Object XML response** – Select this option to include the options configured above also in the Multi-Object XML response. If this option is selected, the Multi-Object XML response type is identical to the behavior of the Multi-Object Detailed XML response type. This eliminates the need to change the API request when you want to utilize the Multi-Object Detailed XML response type.

Rapid Service Indicator

The following section describes the Rapid Service Indicator.

Overview

The Rapid Service Indicator API (RSI) offers an interface for querying the SFX KnowledgeBase for the availability of specific journals and books identified by identifiers such as ISSN/ISBN/LCCN/OBJECT_ID/CODEN.

It replaces the Journal Subscription API (JSI), which will be phased out.

The improvements in the RSI over the JSI include:

- RSI includes not only journals, but also books.
- RSI allows searching not just by ISSN, but also by OBJECT_ID, ISBN, LCCN, and CODEN.
- RSI has full support for consortia APIs. It also takes into account institute-specific activation and allows the sending of IP information so that SFX can determine the appropriate institute.
- RSI can respond with **YES/NO** and also a third value, **MAYBE**, when multiple objects are found.

XML requests should be sent to the following address/program:

```
<SFX server>:<port>/<sfx_instance>/cgi/core/rsi/rsi.cgi
```

The request can be sent using the GET or POST method and the `request_xml` parameter. The POST method is recommended due to the length of the XML query.

For more information on building the RSI index, refer to the **Rapid Service Indicator** section of the *SFX System Administration Guide*.

Configuring the Rapid Service Indicator

In SFX 4, an additional configuration file can be used to configure which types of services are included in the RSI index. By default, all services defined to be included in the A-Z list profile services are included in the RSI index.

This is necessary because the A-Z index, which builds on the data in the RSI index, may contain, in addition getFullTxt services, also the following additional services: getSelectedFullTxt, getAbstract, getTOC, and getHolding.

The following is the location of the configuration file:

```
/exlibris/sfx_ver/sfx4_1/<local instance>/config/rsi.config
```

This file contains the list of service types that are included in the RSI index. Since the RSI index is used for both the RSI API and the A-Z list, it is recommended to set all service types to `Y` that you want to use in either the RSI API or A-Z list.

The following is an example of the configuration file:

```
Section "retrievalType"
# Set next parameter to 1 for automatically detect serviceType
settings defined in A-Z configuration (all profiles taken
into account)
# If you set this parameter to 0 or no A-Z profiles defined in
current instance serviceTypes will be taken from next section
("serviceTypes")
Automatically 1
EndSection

Section "serviceTypes"
getFullTxt Y
getSelectedFullTxt N
getAbstract N
getTOC N
getHolding N
EndSection
```

NOTE:

If the SFX environment has been set up with consortia model 2 (consortia API) and you want services other than full text to be included in the local instance A-Z list (with, for example, abstract services from the shared instance being included in the local instance), it is recommended to manually configure the shared instance service type settings (since the A-Z configuration in the shared instance might not correspond to the service types required in the A-Z lists of the local instances).

For more information on the RSI, see the Developer's Network.

Separate Indexes for Serials and Monographs

Following the release of SFX 4.3, there is an option to create separate RSI indexes for serials and monographs. It is recommended to create separate indexes, as this is required for the eBook Search to work. There are two benefits to the new functionality:

- It is possible to perform an incremental build for the monograph index where the existing RSI index is updated with all of the changes that were made in the SFX KnowledgeBase since the last time the index was built, but the whole index is not rebuilt. This build takes much less time than a complete build.
- The mechanism for including activation data from the shared instance in consortia model 2 has been improved. Instead of retrieving and including the shared data in the local instance RSI during the build (old mechanism), the eBook RSI integrates the RSI data from the local and shared instance on-the-fly during the RSI query (which reduces RSI build time).

Previous Functionality

The following characterize the previous functionality where serials and monographs could only be in the same RSI index:

- Data is stored in local instance `RAPID_SERVICE_INDICATOR` table.
- When an RSI API is received, SFX checks the RSI index, which includes both journals and books.
- Only a complete build is available – no incremental build.
- For consortia API, data from a shared instance is included in the RSI index from the local instance during the build.

New Functionality

The following characterize the new functionality where serials and monographs can be in separate RSI indexes:

- RSI ejournals and RSI ebooks are stored in separate DB tables.
- When an RSI API request is received, both the journal and book RSI indexes are checked.

If RSI ejournals are stored in local instance `RAPID_SERVICE_INDICATOR` table:

- Only a complete build is possible – no incremental build.
- For consortia API, data from a shared instance is included in the RSI index from the local instance during the build.

If RSI ebooks are stored in the `IRSI_INSTITUTIONS` and `IRSI_<instance>` global instance tables:

- Both complete build and incremental build are available
- For an incremental build, the `TRACKING_TABLE` table and triggers are used.
- For consortia API, data from a shared instance is not included in the RSI index from the local instance during a build. Instead, on-the-fly querying of the local and shared RSI tables is performed. For a remote shared set-up, replication is used.

NOTE:

The RSI API does not work if an RSI book index exists, but an RSI index for journals does not exist.

Configuring Separate Indexes for Serials and Monographs

To configure SFX to create separate indexes for serials and monographs, set the `Separate_RSI_for_Books` parameter in the `config/rsi.config` file to `Y`. For example:

```
Section "monograph_parameters"
      Separate_RSI_for_Books "Y"
EndSection
```

The `Separate_RSI_for_Books` parameter affects the behavior of the RSI API and RSI build in the following manner:

- If the parameter set to `Y`:

  RSI API queries both the serial and the monographs RSIs. When building the serial RSI index, book object holdings information is no longer included in the index.

- If the parameter is set to `N`:

  RSI API queries only the serial RSI index (and disregards the monograph RSI index, if it exists). When building the serial RSI index, SFX includes both journal and book holdings data in the index.

The `Separate_RSI_for_Books` parameter does not affect the build of the monograph RSI index. This index can be built in any case.

Complete Monograph Index Build

The complete build is needed in the following cases:

- No RSI index data exists yet for the instance (a first build)

  This means that the `tracking_id` value in the `CONTROL` table is not defined.

- More than 10,000 changes or more than 20% of the total amount of objects have changed for the instance in `TRACKING_TABLE`. In this case, it is faster to do a complete build.
- Activation changes are made to targets or target services.
- After a KnowledgeBase update has been applied in the SFX installation.

NOTE:
Only object portfolio changes are taken into account, not target or target service changes.

The following scenarios track changes by storing the current value in the `CONTROL` table.

- If one of following changes in institute set-up in the local instance:
  - An institute or group is added or deleted.

  - A group affiliation for an institute is added or deleted.
- If one of following occurs in the consortia API set-up of the local instance:
  - The `active` parameter in the Consortium section of the `ctx_object.config` file is changed from `Y` to `N` or from `N` to `Y`.
  - The value of the `rsi_consortium_members` parameter in the `config/consortium.config` file has changed.

With a complete build, the `TRACKING` table, which contains a list of changes, is ignored. The complete build may be quite heavy on system resources (depending on the number of eBook activations). It is recommended to not run this process during peak time.

Incremental Monograph Index Build

During an incremental build, the TRACKING_TABLE table lists all of the changes to the instance. This table is updated by the triggers of the local and global database tables.

During an incremental build, SFX makes a list of objects for which activation needs to be recalculated. For this list, the default institute and group activations are all recalculated and the existing RSI index is updated.

The incremental build is fast, depending on the number of changes. Build time can be 10 minutes or less.

The maximum amount of changes before a complete build is needed is defined in the `config/rsi.config` configuration file, in the following section:

```
Section "complete_build_params"
# The parameters below determine the maximum amount of changes
allowed for incremental build of the monograph RSI and A-Z
index.
# If the number of changes in the SFX KB exceeds the
parameters below, a complete build is required.
        max_op_changes "10000"
        max_changes_percent "20"
EndSection
```

Journal Subscription (Deprecated)

The Journal Subscription API offers an interface for querying the SFX KnowledgeBase for the availability of specific journals identified by their ISSN.

It allows sending an ISSN along with year, institute, issue, or volume details to SFX, which replies whether there are full-text or other services active for this journal, within these restrictions. This API supports requests for one or more ISSNs.

XML requests should be sent to the following address/program:

```
<SFX server>:<port>/<sfx_instance>/cgi/core/
journal_subscription.cgi
```

The request can be sent using the GET or POST method and the `request_xml` parameter. The POST method is recommended due to the length of the XML query.

20

Appendixes

This guide contains the following appendixes:

A

Database Tables

All KB_ and LCL_ tables have the following fields:

Table 13. KB_ and LCL_ Tables

Name	Description
OWNER	For customer owned data: should be displayed as R.customer.instance For all data distributed by ExL, the owner should be `otb`
AVAILABLE_FOR	In LCL_ tables, the field is used to indicate to whom the local data pertains – either DEFAULT or a specific group/institute. The format is either R.customer.instance or R.customer.instance.institute In KB_ tables, the field is used to indicate for whom the entry is visible in KBManager. The format is R.customer.instance or R.customer.instance.institute
CREATION_DATE	Date the DB entry was created
CREATED_BY	Name of the SFX administrator user that created the DB entry
LAST_UPDATE_DATE	Date the DB entry was last updated
LAST_UPDATED_BY	Name of the SFX administrator user that last updated the DB entry

All KB_ tables have the following fields:

Table 14. KB_ Tables

Name	Value	Description
VERSION_NUMBER		DCS version number
RELEASE_NUMBER		Revision release number

Table 14. KB_ Tables

Name	Value	Description
CRUD_TYPE		Create-Read-Update-Delete - stores latest change of DB record

Only the KB_ tables listed in **Table 15** have the following additional fields:

- KB_INTERFACES
- KB_OBJECTS
- KB_OBJECT_IDENTIFIERS
- KB_OBJECT_PORTFOLIOS
- KB_RELATIONS
- KB_TARGETS
- KB_TARGET_SERVICES
- KB_MULTILANGUAGE

Table 15. KB_ Tables

Name	Value	Description
STATUS	ACTIVE, WITHDRAWN	The purpose of this field is to store the life cycle information of the item. Initially it is used for `withdrawn` versus `live` status, but in the future it can also be used for `under construction`, `approved`, etc.
STATUS_DATE		Date of the last status change
DISTRIBUTION_STATUS	READY FOR DISTRIBUTION NOT FOR DISTRIBUTION	Purpose of this field is to indicate whether the record is ready for distribution or not
DISTRIBUTION_STATUS _DATE		Date of the last DISTRIBUTION_STATUS change

List of KB_ Tables

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
KB_AUTHORS	AUTHOR_ID		Internal ID
	OBJECT_ID		Internal ID of the object
	AUTHOR_SIGNI FICANCE	MAIN ADDITIONAL (default) ALTERNATIVE	1. MAIN authors will only be loaded if the source is a MARC records (where 100 is main author, 700 are additional authors). In vendor files, we cannot assume that first author listed is the main author. In this case, all authors will be additional authors. Note: main versus additional is important for articles, not so much for books. If we make a distinction, is is more for display in KBManager, etc when displaying only the first author. 2. ALTERNATIVE authors are not loaded using MARC records, but are authors we would like to store as author variations available for searching. These authors will not be displayed in end-user applications (such as the SFX menu and SFX menudisplayers, target parsers, the A-Z list, CitationLinker multi-object window). They will not be included in Export formats. 3. ALTERNATIVE authors will be used for CJK author information, in case more than one author value is needed for searching.
	AUTHOR_TYPE	PERSONAL CORPORATE MEETING	Type of author can be personal, corporate, or meeting

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	AUTHOR_FIRST_NAME		Contains the first name of the author names whose parsing of full name into author_first and author_last was successful - for display purposes.
	AUTHOR_LAST_NAME		Contains the last name of the author names whose parsing of full name into author_first and author_last was successful - for display purposes.
	AUTHOR_FULL_NAME		Contains the full author name as recieved from load file – for display purposes
	AUTHOR_FIRST_NAME_NORMALIZED		Contains the first name of the author names whose parsing of full name into author_first and author_last was successful - after normalization.
	AUTHOR_LAST_NAME_NORMALIZED		Contains the last name of the author names whose parsing of full name into author_first and author_last was successful - after normalization.
	AUTHOR_FULL_NAME_NORMALIZED		Contains the full author name as recieved from load file – after normalization.

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	FULL_NAME_FORMAT	For PERSONAL author_type: FORENAME, SURNAME, FAMILY NAME, Default is SURNAME For CORPORATE, MEETING author_type: INVERTED, JURISDICTION, DIRECT ORDER Default is DIRECT ORDER	Description of each type – from MARC standard: Forename - Forename or a name consisting of words, initials, letters, etc. that are formatted in direct order. For authors of the type 'Forename', the Firstname field will be empty. The whole string is included as 'Lastname'. Surname - Single or multiple surname formatted in inverted order or a single name without forenames that is known to be a surname. Family name - Name represents a family, clan, dynasty, house, or other such group and may be formatted in direct or inverted order. Inverted - Corporate name or Meeting name starting with a personal name in inverted order. Jurisdiction - Name of a jurisdiction that is also an ecclesiastical entity or is a jurisdiction name under which a corporate name/meeting name or a title of a work is entered. Direct Order - Corporate or Meeting Name in direct order

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	COMMENT	LATIN (default), TRADITIONAL-CHI SIMPLIFIED-CHI, PINYIN KOREAN-UNSPEC HANGUL, JAPANESE-UNSPEC HIRAGANA	Will store type of CJK author – only used for ALTERNATIVE authors
KB_CAT	CAT_ID		Internal ID
	CAT_NORMALIZED		will store the global category information provided by ExL - after normalization
	CAT_DISPLAY		will store the global category information provided by ExL - for display purposes
KB_CAT_ASSIGN	CAT_ASSIGN_INTERNAL_ID		Internal ID
	CAT_SUBCAT_INTERNAL_ID		Internal ID of a global category/ subcategory combination
	OBJECT_ID		Internal ID of the object
KB_CAT_SUBCAT	CAT_SUBCAT_INTERNAL_ID		Internal ID
	CAT_ID		Internal ID of a global category
	SUBCAT_ID		Internal ID of a global subcategory
KB_INTERFACES	INTERFACE_ID		Internal ID

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	INTERFACE_NAME		will store Interface name used for BO display purposes
	INTERFACE_PUBLIC_NAME		will store Interface name used for display purposes for end-user (e.g. in A-Z list - locate tab - vendor list)
KB_LINKING_PARAMETERS	LINKING_PARAM_ID		Internal ID
	ITEM_TYPE	TARGET_SERVICE OBJECT_PORTFOLIO	Linking parameters can be connect to either target service or portfolio
	ITEM_ID		Internal ID of the target service or portfolio - in case of global entry
	LOCAL_ITEM_ID		Internal ID of the target service or portfolio - in case of locally created entry
	PARAMETER_NAME		Linking parameter name
	DESCRIPTION		Description of linking parameter - added to make it easier for SFX administrator to fill in the value
	IS_OTB_VALUE		Defines whether linking parameter has an out-of-the-box value or not
	OTB_VALUE		Stores out-of-the-box value
KB_MULTILANGUAGE	MULTILANGUAGE_INTERNAL_ID		Internal ID
	MULTILANGUAGE_GROUP		SFX application in which translation element is used
	MULTILANGUAGE_MESSAGE_CODE		HTML template code to refer to this translation element

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	MULTILANGUAGE_MESSAGE_DETAILS		Description for this translation element
	MULTILANGUAGE_LANGUAGE_CODE		Language of the multilanguage translation element - using 3-letter ISO 639-2 standard language code
	MULTILANGUAGE_MESSAGE_TEXT		Actual value for this translation element
KB_OBJECTS	OBJECT_ID		Internal ID
	OBJECT_TYPE	BOOK CD CONFERENCE DATABASE DISSERTATION DOCUMENT JOURNAL MANUSCRIPT NEWSPAPER PROCEEDING REPORT SERIES TRANSCRIPT WIRE VIDEO AUDIO SCORE	Type of object
	LANGUAGE		Language of the object - using 3-letter ISO 639-2 standard language code
	PEER_REVIEWED	YES NO	indicates whether object uses peer-review mechanism before allowing articles to be published

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	MMS_ID		ID required for sync with MMS
KB_OBJECT_ATTRIBUTES_METADATA	OBJECT_TYPE	BOOK CD CONFERENCE DATABASE DISSERTATION DOCUMENT JOURNAL MANUSCRIPT NEWSPAPER PROCEEDING REPORT SERIES TRANSCRIPT WIRE VIDEO AUDIO SCORE	Table not in use - intended to store allowed attributes for each type of object
	TABLE_NAME		
	ATTRIBUTE_TYPE		
	MANDATORY		
KB_OBJECT_IDENTIFIERS	OBJECT_IDENTIFIER_ID		Internal ID
	OBJECT_ID		Internal ID of the object

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	TYPE	ISSN ISBN LCCN OCLC_NR CODEN DOI	Identifier type
	SUBTYPE	PRINT ELECTRONIC INCORRECT PRINT_HBK PRINT_PBK	Identifier subtype
	VALUE		Value
KB_OBJECT_PORTFOLIOS	OP_ID		Internal ID
	TARGET_ID		Internal ID of the Target
	TARGET_SERVICE_ID		Internal ID of the Target service
	OBJECT_ID		Internal ID of the object
	TARGET_PARSER_PROGRAM		Stores the name of the target parser program that will create the URLs to link to a particular TARGET_SERVICE
	PARSE_PARAM		Contains information that is used by the Target Parser to create URLs. You fill in parameters that apply to the specific portfolio for example, jkey
	DESCRIPTION		General description - will be used by the kb team to describe the portfolio (for ExL distributed targets) - can be used by customers to describe locally created portfolio

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	KB_INTERNAL_DESCRIPTION		Internal field for KB team - not included in revision update package - used to store test account info, internal notes
	LINKING_LEVEL	ARTICLE ARTICLE_DOI ISSUE VOLUME JOURNAL CITATION BOOK DATABASE INHERIT	
	THRESHOLD		Conditions placed on a portfolio. Thresholds determine whether or not a given service is displayed in the SFX menu screen.
KB_PUBLISHERS	PUBLISHER_ID		Internal ID
	PUBLISHER_NAME_DISPLAY		Publisher name information - used for display purposes
	PUBLISHER_NAME_NORMALIZED		Publisher name information - normalized version
	DATE_OF_PUBLICATION		Date of publication
	PLACE_OF_PUBLICATION_DISPLAY		Publisher place of publication information - used for display purposes
	PLACE_OF_PUBLICATION_NORMALIZED		Publisher place of publication information - normalized version
	OBJECT_ID		Internal ID of the object

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
KB_RELATIONS	RELATION_ID		Internal ID
	PRIMARY_OBJECT_ID		Internal ID of the primary object for this relation
	SECONDARY_OBJECT_ID		Internal ID of the secondary object for this relation
	RELATION_TYPE	PROCEEDING_SERIES PROCEEDING_ITEM OTHER_EDITION SERIES,SERIES_ITEM CONTINUES CONTINUED_BY CONTINUES_IN_PART CONTINUED_IN_PART_BY MERGED_INTO FORMED_BY_THE_UNION_OF SPLIT_INTO ABSORBED_BY ABSORBED ABSORBED_IN_PART_BY ABSORBED_IN_PART RELATED SUPPLEMENT SUPPLEMENT_PARENT TRANSLATION_ENTRY	

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
KB_SORTING_ORDER	TYPE		Table used to to determine what takes preference when displaying id-info in kbmanager
	SUBTYPE		
	ORDER		
KB_SOURCES	SOURCE_ID		Internal ID
	SOURCE_NAME		name of the SOURCE
	SOURCE_PARSER_PROGRAM		the name of the source as listed in the source configuration file - source ID section.
	SHOW_AVAILABILITY	YES NO	setting this option to 'Yes' will make sure the availability information (available from 2009, 12, 1 until …') is shown in the SFX menu screen for a particular SOURCE
	DESCRIPTION		General description - will be used by the kb team to describe the source (for ExL distributed sources) - can be used by customers to describe locally created source
KB_SOURCE_SERVICES	SOURCE_SERVICE_ID		Internal ID
	SOURCE_ID		Internal ID of the source

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	SERVICE_TYPE	getFullTxt getSelectedFullTxt getDOI getMessageNoFullTxt getHolding getDocumentDelivery getBookReview getAbstract getTOC getAlikeRecord getAuthor getCitedAuthor getCitedReference getReference getCitedBook getCitedGenome getCitedRecord getCitedJournal getAuthorEmail getSubject getWebService getWebSearch getRecommendation	Type of service
KB_SUBCAT	SUBCAT_ID		Internal ID
	SUBCAT_NORMALIZED		will store the global subcategory information provided by ExL - after normalization
	SUBCAT_DISPLAY		will store the global subcategory information provided by ExL - for display purposes

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
KB_TARGETS	TARGET_ID		Internal ID
	TARGET_NAME		the name of the target as displayed in SFX Admin Center Back Office interfaces
	TARGET_PUBLIC_NAME		the name of the TARGET as displayed to the user in the SFX menu screen. NOTE: the character # should not be used in the public name of the TARGET as this will cause problems when exporting database information.
	DESCRIPTION		General description - will be used by the kb team to describe the target / target_service (for ExL distributed targets) - can be used by customers to describe locally created target
	KB_INTERNAL_DESCRIPTION		Internal field for KB team - not included in revision update package - used to store test account info, internal notes

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	AGGREGATOR	0, 1	Indicates whether a target is aggregator or not. This will be used (1) with aggregator versus non-aggregator display logic rules and (2) during post revision update scripts - where if this field is set to 'Aggregator - Y', NEW Object Portfolios will be added as ACTIVE, in case the KB_TARGET_SERVICE has a LCL_SERVICE_INVENTORY record with ACTIVATION_STATUS set to be ACTIVE. (3) when adding portfolios in KBmanager The customer can change the behavior of aggregator automatic activation by using the LCL_SERVICE_INVENTORY field Autoactive. This field can be INHERIT/YES/NO. INHERIT means inherit value from the KB_TARGETS Aggregator field. If field is set to 'YES', this means that during updates, new OP will be added as ACTIVE, in case the KB_TARGET_SERVICE has a LCL_SERVICE_INVENTORY record with ACTIVATION_STATUS set to be ACTIVE. If field is set to 'NO', this means that during updates, new OP will be added as INACTIVE.

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	SITE_AVAILABILITY	SITE DOWN SITE UP	This field will be for future use. This field will not be displayed in the KBManager UI in SFX version 4.0
	TARGET_CHARACTER_SET		can be used to specify a different Character Set (other than LATIN-1) which needs to be used for a particular TARGET.
	TARGET_TYPE	BOOK JOURNAL, MIXED A&I WEB SERVICE MIXED	
	THRESHOLD		Conditions placed on a target. Thresholds determine whether or not a given service is displayed in the SFX menu screen.
	LOGIC_UPDATE_DATE		Date of last complete update: this is the date of the last complete overhaul of the target
KB_TARGET_SERVICES	TARGET_SERVICE_ID		Internal ID
	TARGET_ID		Internal ID of the target

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	SERVICE_TYPE	getFullTxt getSelectedFullTxt getDOI getMessageNoFullTxt getHolding getDocumentDelivery getBookReview getAbstract getTOC getAlikeRecord getAuthor,getCitedAuthor getCitedReference getReference getCitedBook getCitedGenome getCitedRecord getCitedJournal getAuthorEmail getSubject getWebService getWebSearch getRecommendation	Type of Service
	SERVICE_PUBLIC_DESCRIPTION		Contains the text describing a specific service, for example, "Full-text available via". This text will override the default explanatory text defined in the 'Services Public Names' tab (for simplified menu) or services_public_names.config_ for that service type.

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	DESCRIPTION		General description - will be used by the kb team to describe the target_service (for ExL distributed targets) - can be used by customers to describe locally created target service
	KB_INTERNAL_ DESCRIPTION		Internal field for KB team - not included in revision update package - used to store test account info, internal notes
	TARGET_PARSER_PROGRAM		stores the name of the target parser program that will create the URLs to link to a particular TARGET_SERVICE
	PARSE_PARAM		contains information that is used by the target Parser to create URLs. You fill in parameters that apply to the whole target Service, for example, base URL
	TARGET_DISPLAYER_PROGRAM		used to specify a special display type for a particular TARGET_SERVICE. If the field is left blank, the default display is used. For example, the UNION_CATALOGUE's getHolding service uses a DISPLAYER to create a drop-down list of different union catalogues in the SFX menu screen.

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	OBJECT_LOOKUP	YES, NO	Object-lookup should be 'Yes" when objects are attached to this target service. This means that the target service only shows up in the SFX menu screen if the object requested (for example, a particular journal) is part of the object portfolio of that particular target service. If Object-lookup is 'No', this means no objects are connected to a particular target service. The target service will show up in the SFX menu screen, regardless of the object you start from. This is used for more general, 'searchoriented' target services, such as a getHolding or getWebSearch service.
	LINKING_LEVEL	ARTICLE ARTICLE_DOI ISSUE VOLUME JOURNAL CITATION BOOK DATABASE INHERIT	
	IS_FREE		indicates whether all the portfolios for a target service are accessible freely, without subscription fee.

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	CROSSREF_SUPPORTED	YES, NO	Indication whether Crossref lookup can be used for this target service to improve target linking, as part of the revision update. Customers do not have to do anything if they want to use crossref for a particular target service in v4. Can be overridden by ROSSREF_ENABLED field in LCL_LINKING_TARGET_SERVICE table Purpose of this field is to provide a local override option – in case customers want to disable crossref lookup for a particular target service.
	THRESHOLD		Conditions placed on a target service. Thresholds determine whether or not a given service is displayed in the SFX menu screen.
	AUTOLOADER_USED		internal KB team field - indicates whether autoloader is used to update this target service
	LOGIC_UPDATE_DATE		date of last complete update: this is the date of the last complete overhaul of the target service
KB_TITLE	TITLE_ID		Internal ID
	OBJECT_ID		Internal ID of the object

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	TITLE_TYPE	MAIN TRANSLATION ROMANIZATION WRITING_SYSTEM ABBREVIATION UNIFORM ALTERNATIVE	title type
	TITLE_SUB_TYPE	SIMPLIFIED TRADITIONAL PINYIN HIRAGANA KATAKANA HANGUL	title subtype
	TITLE_VALUE		title value
	NON_FILLING_CHAR		stores the number of characters at the beginning of a title which should be ignored when sorting. By default or when no non-filing character is loaded, the value will be '0'.
	TITLE_LANGUAGE		stores the title language - using 3-letter ISO 639-2 standard language code
KB_TITLE_KEYS	TITLE_KEY_ID		Internal ID
	OBJECT_ID		Internal ID of the object
	TITLE_ID		Internal ID of the corresponding title
	TITLE_KEY_USE	SEARCH, SORT	defines what title will be used for - searching or sorting purposes

Table 16. KB_Tables

Table Name	Field Name	Field Value	Field Description
	TITLE_KEY_TYPE	generalDiacritics diacriticsGerman diacriticsSwedish diacriticsDanish diacriticsUSA Hangul pinyinNormalization StrokeSequence StrokeCount katakana Hiragana koreanNormalization JapaneseNormalization ChineseNormalization TaiwaneseNormalization	title key type
	TITLE_KEY_VALUE		title value
	NON_FILLING_CHAR		stores the number of characters at the beginning of a title which should be ignored when sorting. By default or when no non-filing character is loaded, the value will be ‘0’.

List of LCL_ Tables

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
LCL_CAT	CAT_ID		Internal ID
	CAT_NORMALIZED		will store the local category information - after normalization
	CAT_DISPLAY		will store the local category information - for display purposes
LCL_CAT_ASSIGN	CAT_ASSIGN_INTERNAL_ID	CAT_ASSIGN_ID	Internal ID
	CAT_SUBCAT_INTERNAL_ID		Internal ID of a local category/subcategory combination
	OBJECT_ID		Internal ID of the object
LCL_CAT_SUBCAT	CAT_SUBCAT_INTERNAL_ID	CAT_SUBCAT_ID	Internal ID
	CAT_ID		Internal ID of a local category
	SUBCAT_ID		Internal ID of a local subcategory
LCL_LINKING_PARAMETERS	INTERNAL_ID		Internal ID
	LINKING_PARAM_ID		Internal ID of KB_LINKING_PARAMETERS entry for which value is stored
	VALUE		Stores linking parameter value
LCL_MULTILANGUAGE	MULTILANGUAGE_INTERNAL_ID		Internal ID

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	MULTILANGUAGE_GROUP		SFX application in which translation element is used
	MULTILANGUAGE_MESSAGE_CODE		HTML template code to refer to this translation element
	MULTILANGUAGE_MESSAGE_DETAILS		Description for this translation element
	MULTILANGUAGE_LANGUAGE_CODE		Language of the multilanguage translation element - using 3-letter ISO 639-2 standard language code
	MULTILANGUAGE_MESSAGE_TEXT		Actual value for this translation element
LCL_OBJECT_IDENTIFIERS	OBJECT_IDENTIFIER_ID		Internal ID
	OBJECT_ID		Internal ID of the object
	TYPE		Identifier type - table is currently only used to store LOCAL attribute
	SUB_TYPE		Identifier subtype
	VALUE		Value
LCL_OBJECT_PORTFOLIO_INVENTORY	INTERNAL_ID		Internal ID
	OP_ID		Internal ID of KB_OBJECT_PORTFOLIOS entry for which activation information is stored
	ACTIVATION_STATUS	ACTIVE, INACTIVE	Activation status
	DEACTIVATION_REASON		Reason for deactivation

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
LCL_OBJECT_PORTFOLIO_LINKING_INFO	INTERNAL_ID		Internal ID
	OP_ID		Internal ID of KB_OBJECT_PORTFOLIOS entry for which activation information is stored
	AUTHENTICATION_NOTE		Authentication note - will be displayed in SFX menu to endusers
	PUBLIC_NOTE		General note - will be displayed in SFX menu to endusers
	INTERNAL_DESCRIPTION		Internal description - used by SFX administrator for internal purposes
	TARGET_PARSER_PROGRAM		Override of KB_ TARGET_PARSER_PROGRAM value
	PARSE_PARAM		Override of KB_ PARSE_PARAM value
	THRESHOLD		Override of KB_ THRESHOLD value
LCL_SERVICE_INVENTORY	INTERNAL_ID		Internal ID
	TARGET_ID		Internal ID of KB_TARGETS entry for which activation information is stored
	TARGET_SERVICE_ID		Internal ID of KB_TARGET_SERVICES entry for which activation information is stored

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	AUTO_ACTIVE	YES, NO, INHERIT	If field is set to 'YES', this means that during updates, new OP will be added as ACTIVE, in case the KB_TARGET_SERVICE has a LCL_SERVICE_INVENTORY record with ACTIVATION_STATUS set to be ACTIVE. If field is set to 'NO', this means that during updates, new OP will be added as INACTIVE. Ifi field is set to 'INHERIT', this means inherit behavior from value from the KB_TARGETS AGGREGATOR field.
	ACTIVATION_STATUS	ACTIVE, INACTIVE	Activation status
	DEACTIVATION_REASON		Reason for deactivation
	SITE_DOWN	0, 1	Field is either set in KBManager or triggered by Verde soap service. When a TARGET_SERVICE has status SITE DOWN, then SFX menu will display an icon that shows that the target service is down. Additionally, the 'Go' button and link to target window not be presented, instead a text saying site down will appear
	SITE_DOWN_DATE		Site_down_date can be used to describe, for the SFX administrator, the date of the downtime.

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	SITE_DOWN_REASON		site_down_reason can be used to describe, for the SFX administrator, in more details the reason of the downtime.
LCL_SERVICE_LINKING_INFO	INTERNAL_ID		Internal ID
	TARGET_SERVICE_ID		Internal ID of KB_TARGET_SERVICES entry for which activation information is stored
	SERVICE_PUBLIC_DESCRIPTION		Text describing service - will be displayed in SFX menu to endusers - overrides default text describing service
	AUTHENTICATION_NOTE		Authentication note - will be displayed in SFX menu to endusers
	PUBLIC_NOTE		General note - will be displayed in SFX menu to endusers
	INTERNAL_DESCRIPTION		Internal description - used by SFX administrator for internal purposes
	TARGET_PARSER_PROGRAM		Override of KB_TARGET_PARSER_PROGRAM value
	PARSE_PARAM		Override of KB_PARSE_PARAM value
	TARGET_DISPLAYER_PROGRAM		Override of KB_THRESHOLD value
	PROXY_ENABLED		Used for selective proxy use
	OBJECT_LOOKUP	Y, N, INHERIT	Local field to override KB_SOURCE_SERVICES OBJECT_LOOKUP field

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	CROSSREF_ENABLED	INHERIT, NO	Local field to override KB_SOURCE_SERVICES CROSSREF_ENABLED field
	THRESHOLD		Override of KB_ THRESHOLD value
LCL_SOURCE_LINKING_INFO	INTERNAL_ID		Internal ID
	SOURCE_ID		Internal ID of KB_SOURCES entry for which activation information is stored
	INTERNAL_DESCRIPTION		Internal description - used by SFX administrator for internal purposes
	SHOW_AVAILABILITY	YES, NO, INHERIT	Field to override the KB_SOURCES SHOW_AVAILABILITY value (by setting this option to 'Yes' will make sure the availability information (available from 2009, 12, 1 until …') is shown in the SFX menu screen for a particular SOURCE)
LCL_SOURCE_SERVICE_INVENTORY	INTERNAL_ID		Internal ID
	SOURCE_ID		Internal ID of KB_SOURCES entry for which activation information is stored
	SOURCE_SERVICE_ID		Internal ID of KB_SOURCE_SERVICES entry for which activation information is stored
	ACTIVATION_STATUS	ACTIVE, INACTIVE	Activation status

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	DEACTIVATION_REASON		Reason for deactivation
LCL_SUBCAT	SUBCAT_ID		Internal ID
	SUBCAT_NORMALIZED		will store the local subcategory information - after normalization
	SUBCAT_DISPLAY		will store the local subcategory information - for display purposes
LCL_TARGET_INVENTORY	INTERNAL_ID		Internal ID
	TARGET_ID		Internal ID of KB_TARGETS entry for which activation information is stored
	ACTIVATION_STATUS	ACTIVE, INACTIVE	Activation status
	DEACTIVATION_REASON		Reason for deactivation
	SITE_DOWN	Y, N	Field for future use only - not displayed in KBManager
	SITE_DOWN_REASON	SITE_NOT_AVAILABLE	Field for future use only - not displayed in KBManager
	SITE_DOWN_DATE		Field for future use only - not displayed in KBManager
LCL_TARGET_LINKING_INFO	INTERNAL_ID		Internal ID
	TARGET_ID		Internal ID of KB_TARGET
	TARGET_PUBLIC_NAME		Override of KB_TARGET public name
	AUTHENTICATION_NOTE		Authentication note - will be displayed in SFX menu to endusers

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	PUBLIC_NOTE		General note - will be displayed in SFX menu to endusers
	INTERNAL_DESCRIPTION		Internal description - used by SFX administrator for internal purposes
	THRESHOLD		Override of KB_THRESHOLD value
LCL_TITLE	INTERNAL_ID		Internal ID
	TITLE_ID		Internal ID of the corresponding KB_TITLE entry
	OBJECT_ID		Internal ID of the object
	TITLE_VALUE		title value
	NON_FILING_CHAR		stores the number of characters at the beginning of a title which should be ignored when sorting. By default or when no non-filing character is loaded, the value will be '0'.
LCL_TITLE_KEYS	TITLE_KEY_ID		Internal ID
	OBJECT_ID		Internal ID of the object
	TITLE_ID		Internal ID of the corresponding title
	TITLE_KEY_USE	SEARCH', 'SORT', 'BOTH	defines what title will be used for - searching or sorting purposes

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	TITLE_KEY_TYPE	generalDiacritics', 'diacriticsGerman', 'diacriticsSwedish', 'diacriticsDanish', 'diacriticsUSA, Hangul', 'PinyinNormalization', 'StrokeSequence', 'StrokeCount', 'Katakana', 'Hiragana', 'KoreanNormalization, JapaneseNormalization', 'ChineseNormalization, TaiwaneseNormalization	title key type
	TITLE_KEY_VALUE		title value
	NON_FILING_CHAR		stores the number of characters at the beginning of a title which should be ignored when sorting. By default or when no non-filing character is loaded, the value will be '0'.
LCL_TRANS_CAT	CAT_ID		Internal ID of KB_CAT table
	CAT_TRANS_NORMALIZED		will store the translated version of the global category information provided by ExL - after normalization
	CAT_TRANS_DISPLAY		will store the translated version of the global category information provided by ExL - for display purposes

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
LCL_TRANS_SUBCAT	SUBCAT_ID		Internal ID of KB_SUBCAT table
	SUBCAT_TRANS_NORMALIZED		will store the translated version of the global subcategory information provided by ExL - after normalization
	SUBCAT_TRANS_DISPLAY		will store the translated version of the global subcategory information provided by ExL - for display purposes
CONTROL	CONTROL_INTERNAL_ID		Internal ID of CONTROL table
	CONTROL_GROUP		Group of configuration settings
	CONTROL_SECTION		Section within configuration group
	INSTITUTE_ID		Institute ID information
	CONTROL_NAME		Configuration setting name
	CONTROL_VALUE		Configuration setting value
	CONTROL_VARTYPE	SCALAR, ARRAY	
INSTITUTE_IP	INSTITUTE_IP_ID		Internal ID of INSTITUTE_IP table
	INSTITUTE_ID		
	INSTITUTE_IP_FROM		
	INSTITUTE_IP_TO		
INSTITUTE	INSTITUTE_ID		Internal ID of INSTITUTE table

Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
	INSTITUTE_NAME		
	INSTITUTE_TYPE	I, G	
SYNONYM	INSTITUTE_ID		Internal ID of SYNONYM table
	SYNONYM_ID		
	SYNONYM_RANK		
SFX_REQUEST	INTERNAL_ID		Internal ID of KB_SFX_REQUEST table
	SERIALIZED		
	SERVICES		
	MODIFIED		
STAT_OBJECT and	REQUEST_ID		Internal ID
STAT_OBJECT_OFFLINE	ISSN		
	ISBN		
	LCCN		
	LOCAL		
	TITLE		
	ATITLE		
	JTITLE		
	BTITLE		
	CTITLE		
	SERIES		
	PUBLISHER		
	PLACE_OF_PUBLICATION		

Table 17. LCL_Tables

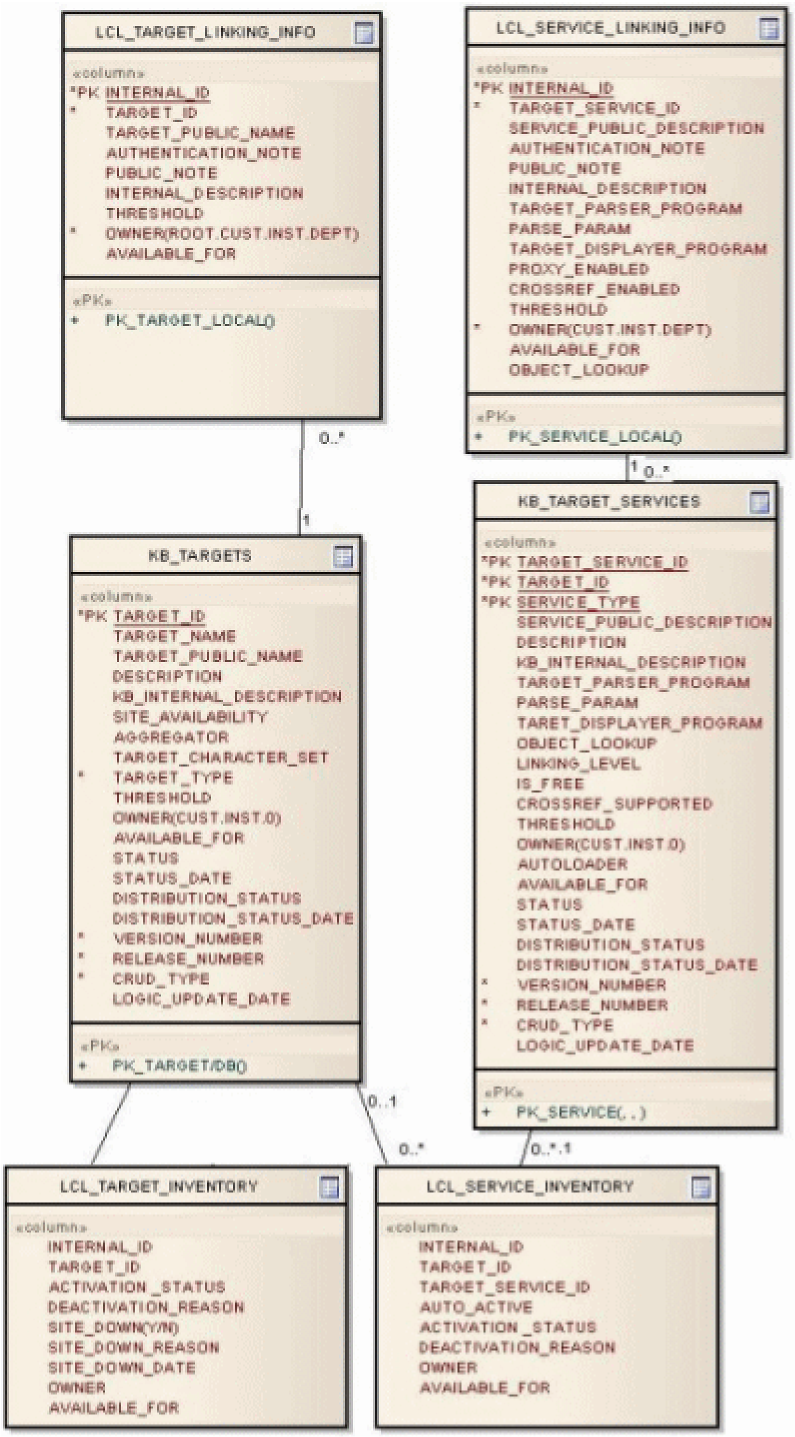
Table Name	Field Name	Field Value	Field Description
	OBJECT		
	SUBCATEGORY		
	STATUS		
	DOI		
	DATE		
	TIME		
	SOURCE		
	IP		
	OBJECT_TYPE		
	INSTITUTE		
	USER_GROUP		
	FACULTY		
	HAS_FULLTXT		
	DATE_OF_PUBLICATION		
	SPAGE		
	EPAGE		
	PRESENTATION_FORMAT		
	SESSION_ID		
	OPEN_URL		
STAT_REPEATABLES and	INTERNAL_ID		Internal ID
STAT_REPEATABLES_OFFLINE	REQUEST_ID		
	ATTRIBUTE		
	VALUE		

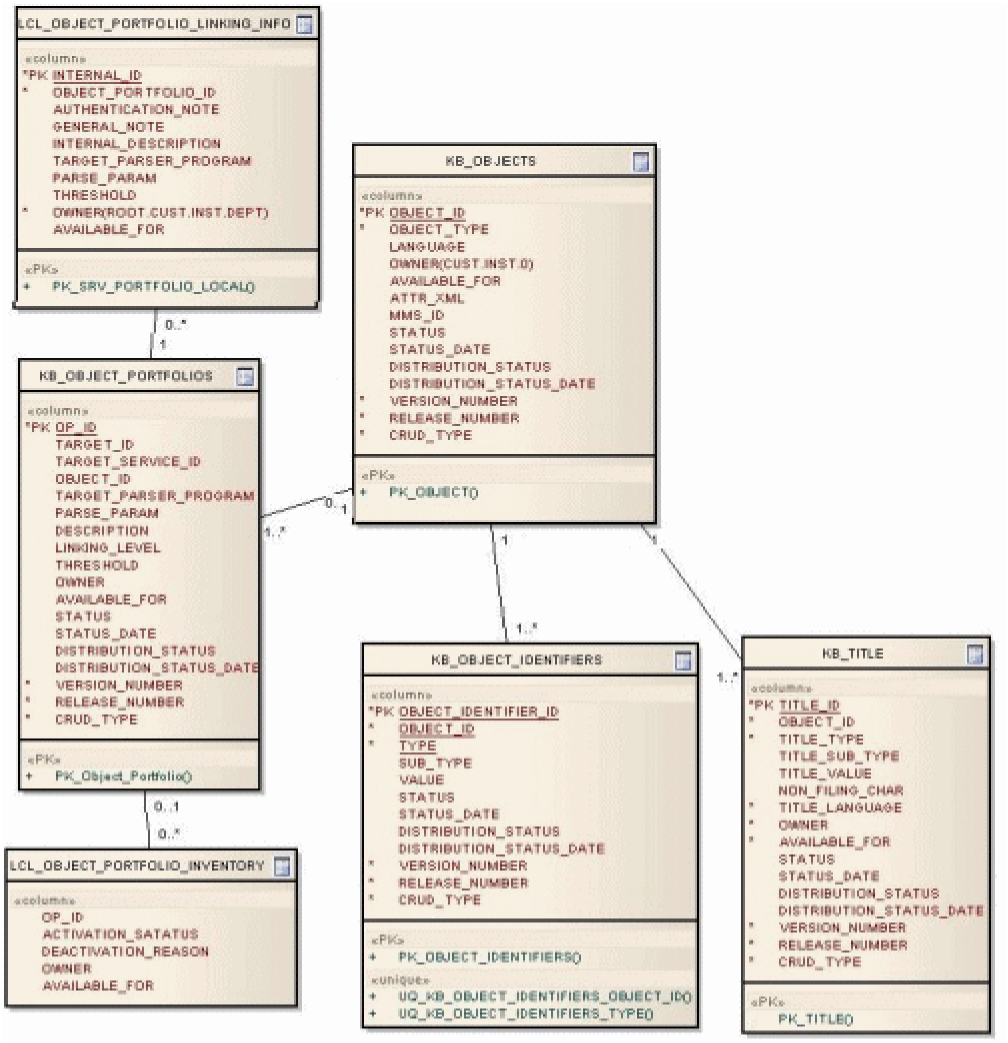
Table 17. LCL_Tables

Table Name	Field Name	Field Value	Field Description
STAT_TARGET_SERVICE and	REQUEST_ID		
STAT_TARGET_SERVICE_OFFLINE	TARGET		
	CLICKS		
	SERVICE		
	URL		
USERS	INTERNAL_ID		Internal ID of USERS table
	USER_ID		
	PASSWORD		
	PROFILE		
	NAME		
	DESCRIPTION		
	INSTITUTE_ID		
	SESSION_ID		

ERD Diagram

The following diagram illustrates the connections between objects (with title and Identifier information), targets, target services and object portfolios. It includes connections from the `global KB_ tables` to `LCL_ tables` with inventory and local linking information.


LCL_TARGET_LINKING_INFO
«column»
*PK INTERNAL_ID
* TARGET_ID
TARGET_PUBLIC_NAME
AUTHENTICATION_NOTE
PUBLIC_NOTE
INTERNAL_DESCRIPTION
THRESHOLD
* OWNER(ROOT.CUST.INST.DEPT)
AVAILABLE_FOR
«PK»
+ PK_TARGET_LOCAL()
LCL_SERVICE_LINKING_INFO
«column»
*PK INTERNAL_ID
* TARGET_SERVICE_ID
SERVICE_PUBLIC_DESCRIPTION
AUTHENTICATION_NOTE
PUBLIC_NOTE
INTERNAL_DESCRIPTION
TARGET_PARSER_PROGRAM
PARSE_PARAM
TARGET_DISPLAYER_PROGRAM
PROXY_ENABLED
CROSSREF_ENABLED
THRESHOLD
* OWNER(CUST.INST.DEPT)
AVAILABLE_FOR
OBJECT_LOOKUP
«PK»
+ PK_SERVICE_LOCAL()
0..*
1
1
0..*
KB_TARGETS
«column»
*PK TARGET_ID
TARGET_NAME
TARGET_PUBLIC_NAME
DESCRIPTION
KB_INTERNAL_DESCRIPTION
SITE_AVAILABILITY
AGGREGATOR
TARGET_CHARACTER_SET
* TARGET_TYPE
THRESHOLD
OWNER(CUST.INST.0)
AVAILABLE_FOR
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
LOGIC_UPDATE_DATE
«PK»
+ PK_TARGET/DB()
KB_TARGET_SERVICES
«column»
*PK TARGET_SERVICE_ID
*PK TARGET_ID
*PK SERVICE_TYPE
SERVICE_PUBLIC_DESCRIPTION
DESCRIPTION
KB_INTERNAL_DESCRIPTION
TARGET_PARSER_PROGRAM
PARSE_PARAM
TARET_DISPLAYER_PROGRAM
OBJECT_LOOKUP
LINKING_LEVEL
IS_FREE
CROSSREF_SUPPORTED
THRESHOLD
OWNER(CUST.INST.0)
AUTOLOADER
AVAILABLE_FOR
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
LOGIC_UPDATE_DATE
«PK»
+ PK_SERVICE(, ,)
0..1
0..*
0..*.1
LCL_TARGET_INVENTORY
«column»
INTERNAL_ID
TARGET_ID
ACTIVATION_STATUS
DEACTIVATION_REASON
SITE_DOWN(Y/N)
SITE_DOWN_REASON
SITE_DOWN_DATE
OWNER
AVAILABLE_FOR
LCL_SERVICE_INVENTORY
«column»
INTERNAL_ID
TARGET_ID
TARGET_SERVICE_ID
AUTO_ACTIVE
ACTIVATION_STATUS
DEACTIVATION_REASON
OWNER
AVAILABLE_FOR


LCL_OBJECT_PORTFOLIO_LINKING_INFO
«column»
*PK INTERNAL_ID
* OBJECT_PORTFOLIO_ID
AUTHENTICATION_NOTE
GENERAL_NOTE
INTERNAL_DESCRIPTION
TARGET_PARSER_PROGRAM
PARSE_PARAM
THRESHOLD
* OWNER(ROOT.CUST.INST.DEPT)
AVAILABLE_FOR
«PK»
+ PK_SRV_PORTFOLIO_LOCAL()
0..*
1
KB_OBJECT_PORTFOLIOS
«column»
*PK OP_ID
TARGET_ID
TARGET_SERVICE_ID
OBJECT_ID
TARGET_PARSER_PROGRAM
PARSE_PARAM
DESCRIPTION
LINKING_LEVEL
THRESHOLD
OWNER
AVAILABLE_FOR
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
«PK»
+ PK_Object_Portfolio()
0..1
0..*
LCL_OBJECT_PORTFOLIO_INVENTORY
«column»
OP_ID
ACTIVATION_SATATUS
DEACTIVATION_REASON
OWNER
AVAILABLE_FOR
1..*
0..1
KB_OBJECTS
«column»
*PK OBJECT_ID
* OBJECT_TYPE
LANGUAGE
OWNER(CUST.INST.0)
AVAILABLE_FOR
ATTR_XML
MMS_ID
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
«PK»
+ PK_OBJECT()
1
1..*
1
1..*
KB_OBJECT_IDENTIFIERS
«column»
*PK OBJECT_IDENTIFIER_ID
* OBJECT_ID
* TYPE
SUB_TYPE
VALUE
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
«PK»
+ PK_OBJECT_IDENTIFIERS()
«unique»
+ UQ_KB_OBJECT_IDENTIFIERS_OBJECT_ID()
+ UQ_KB_OBJECT_IDENTIFIERS_TYPE()
KB_TITLE
«column»
*PK TITLE_ID
* OBJECT_ID
* TITLE_TYPE
TITLE_SUB_TYPE
TITLE_VALUE
NON_FILING_CHAR
* TITLE_LANGUAGE
* OWNER
* AVAILABLE_FOR
STATUS
STATUS_DATE
DISTRIBUTION_STATUS
DISTRIBUTION_STATUS_DATE
* VERSION_NUMBER
* RELEASE_NUMBER
* CRUD_TYPE
«PK»
PK_TITLE()

B

List of Service Types

Service	Description
getAbstract	Supported by services that display a short content description of an article
getAlikeRecord	Supported by services that display a similar record to the one requested
getAuthor	Supported by services that display information about the publications of an author given his/her name
getAuthorEmail	Supported by services that display information about the e-mail address of an author given his/her name
getBookReview	Supported by services that display a short review of a book
GetCitedAuthor	Supported by services that display information on the number of citations for an author
getCitedBook	Supported by services that display information on the number of citations for a book
getCitedGenome	Supported by services that display genome information given a genUID
getCitedJournal	Supported by services that display information on the number of citations for a journal
getCitedRecord	Supported in its current version only by Medline or the PubMed database where a record is displayed given a medUID
getCitedReference	Supported by services that display information on the number of citations for an article
getDocumentDelivery	Supported by services that offer a document delivery service
getDOI	Supported by services that display the electronic content of an article or book based on the DOI value

Service	Description
getFullTxt	Supported by services that display the electronic content of an article or book
getHolding	Supported by services that display holdings information for articles, journals, books
getMessageNoFullTxt	Provides an option for the display of a **No Full-Text Available** message in the SFX menu
getPayPerView	Supported by services that offer a pay per view service
getRecommendation	Supported by services that offer recommendations, such as bX
getReference	Supported by services that display the references of an article
getSelectedFullTxt	Supported by services for databases that do not have cover-to-cover representation of their journals. This service type will allow SFX to add more content, and give the library the opportunity to arrange how this content is presented to the patron.
getSubject	Supported by services that display subject information
getTOC	Supported by services that display the table of contents of a journal or a book (given publication year, volume, issue)
getWebSearch	Supported by services that start searches on the Internet
getWebService	Supported by services that offer general Internet services

C

External Export

This section includes:

Introduction

This option in SFX allows you to initiate from an external application, for example MetaLib version 3, an export of data from your SFX database to the external application. This feature is not available in MetaLib version 4 or later.

Setup

In the SFX server, set up the following files to enable external export.

reverse_proxy_httpd.config__

1 Type the following as the `sfxglb41` user.

```
cn
vi reverse_proxy_httpd.config__
```

2 Add the following line for each instance (if it does not already exist):

```
ProxyPass   /external_export/<instance>  http://
localhost:<local_apache_port>/<instance>/sfxadmin/
external_export.cgi
```

3 Restart the Apache server for the instance, using the Server Admin Utility.

a Type `da` to move to the `admin` directory.

b Type `./server_admin_util`

c Type **1 Stop/Start services**.

d Type **2 Proxy Apache stop**, and after a few seconds type **1 Proxy Apache start**.

local_httpd.conf

1 Telnet to the SFX server and type the following as the instance user (for example, `sfxlcl41`)

```
cd /exlibris/sfx_ver/sfx4_1/<instance>/config  (or use the
shortcut cn)

vi local_httpd.config
```

2 Add the following script alias on one line to the `local_httpd.config` file. Replace `<instance>` with the actual instance name.

```
ScriptAlias /external_export/<instance> "/exlibris/sfx_ver/
sfx4_1/<instance>/sfxadmin/external_export.cgi"
```

3 Restart the Apache server for the instance, using the Server Admin Utility.

a Type `da` to move to the admin directory.

b Type `./server_admin_util`

c Type **1 Stop/Start services**.

d Type **7 Instance Apache stop** and and after a few seconds type **6 Instance Apache start**.

external_export.config

Specify the information about the external application initiating the export of the SFX data in a configuration file of your SFX instance.

The location and name of file is:

`/exlibris/sfx_ver/sfx4_1/<instance>/config/`
`external_export.config`

The configuration file contains the following information:

```
#Section "<host_name>"
#    username "<user>"
#    password "<password>"
#    type     "<secure/insecure>"
#    export   "<Y/N>"
#EndSection
```

1 Remove all `#` from the configuration file.

2 Fill in the MetaLib server address, MetaLib FTP username, and password.

3 Select the FTP method: secure (SFTP) or insecure (FTP).

4 Choose `Y` to allow external export.

The password stored in the configuration file needs to be in an encrypted format. Use the Server Admin Utility tool to encrypt your password before you store it in this file.

1 Type `da` to move to the admin directory for the instance.

2 Type: `./server_admin_util`

3 Type: **3 Change passwords** then **4 Generate encrypted password for use in configuration files**

The following is an example input:

```
Section "1.111.11.1"
    username "metalib_ftp"
    password "b616d7c7d65633535303236303"
    type     "secure"
    export   "Y"
EndSection
```

Toubleshooting

Use this option to troubleshoot problems with the External Export tool.

To troubleshoot problems with the External Export tool:

1 Set up the `external_export.config` file on the SFX server to test an external export from the SFX server to the SFX server itself.

The file is located at: `/exlibris/sfx_ver/sfx4_1/<instance_name>/config/external_export.config` a Remove the comment marks from each line.

b Fill in information about the SFX server and instance where you want to perform the test.

For example:

```
#Section "metalib.uni.edu"
#   username "metalib_ftp"
#   password "xxxxxx"
#   type     "secure"
#   export   "Y"
#EndSection
Section "sfx.uni.edu"
 username "sfxlcl41"
   password "xxxxxxx"
   type     "secure"
   export   "Y"
EndSection
```

2 Run the Check External Export tool:

a Log on to the SFX server as an instance user.

b Type `da`.

c Type `./server_admin_util` and select **6 Diagnostics** and **4 Check External Export**.

d Specify the base URL of the instance that you are testing. The following is displayed:

```
Diagnostics
  1 Check for e-mail capabilities
  2 Check for secure FTP capabilities
  3 Check IP address
  4 Check external export
  5 Check Z3950 capabilities
Please select [1-5,q,m]: 4
Please enter the SFX Instance Base URL: http://
sfx.uni.edu:9003/sfxlcl41
Making request. This may take a few moments...
External Export succeeded! The file that was created:
e-collection.03312004135818.xml-marc.gz
Target location of output file: HOME directory of
'sfxlcl41' at 'sfx.uni.edu'
```

The following table describes possible error messages and the problems to which they correspond:

Table 18. Check External Export Error Messages

Error Message	Problem
Failed to contact the SFX Instance: Not Found (404)	■ External_export alias is missing from `local_httpd`.config. ■ Apache has not been restarted after an alias was added.
External Export failed! The error that was returned: 10.1.2.231 not authorized to use `external_export` (Check `external_export.config` for a section named '10.1.2.231')	This message occurs if no information is specified in `external_export.config`.
External Export failed! The error that was returned: Could not instantiate Net::SFTP or could not establish secure connection to '10.1.2.231' with <user>. You should check the user's credentials or that the server's signature has not changed.	■ Incorrect password in `external_export.config`. ■ Problem with SFTP (if one of the Perl libraries required for SFTP is missing)

Table 18. Check External Export Error Messages

Error Message	Problem
The file permissions in the scratch or `scratch/e_collection_update` directory are too restricted External Export failed! The error that was returned: Cannot open `../dbs/scratch/e-collection` for writing	This message occurs in case the file permissions in the scratch or `scratch/e_collection_update` directory are too restricted.
External Export failed! The error that was returned: Could not upload`../dbs/scratch/e-collection.03252004141754.xml-marc.gz` into `e-collection.03252004141754.xml-marc.gz.`	This message occurs if the file permissions for the destination directory are too restricted.

D

Glossary

Term	Description
Ajax	Asynchronous JavaScript and XML – a broad group of Web technologies that can be used to implement a Web application that communicates with a server in the background, without interfering with the current state of the page. It is used in SFX to run the auto-complete feature. Auto-complete involves the program predicting a word or phrase that the user wants to type (such as a search for a journal title).
API	Application Programming Interface – a source code-based specification intended to be used as an interface by software components to communicate with each other as opposed to user interfaces that facilitate interaction between people and computers. Documentation on SFX APIs can be found in El Commons.
CJK	CJK is a collective term for Chinese, Japanese, and Korean, which is used in the field of software and communications internationalization. The number of characters required for complete coverage of these languages cannot fit in the 256-character code space of 8-bit character encodings. They require at least a 16-bit fixed width encoding or multi-byte variable-length encodings. Support for CJK languages requires language and country specific rules for handling the searching, sorting, and display of CJK characters.
COinS	ContextObjects in SPAN – a method to embed bibliographic metadata in the HTML code of Web pages. SFX uses the metadata to retrieve a journal article. For more information, see http://ocoins.info/.

Term	Description
CrossRef	CrossRef is the official Digital Object Identifier (DOI) Registration Agency of the International DOI Foundation. SFX can query the CrossRef reference database with citation metadata and DOI information. This capability allows for enhancement of metadata and improved linking capabilities for certain targets.
CTXO - ContextObject	An information construct that binds a description of a primary entity — the referenced resource — together with descriptions of entities that indicate the context. In SFX, the ContextObject is used to store the metadata sent to SFX by a source via an OpenURL and any additional data SFX retrieves from the SFX database or via an external lookup. SFX uses the ContextObject to find services in the SFX database.
DOI	Digital Object Identifier – a permanent identifier given to a Web file or other Internet document. Referring to an online document by its DOI provides more stable linking than simply referring to it by its URL, because if its URL changes, the publisher need only update the metadata for the DOI to link to the new URL.
HTML	HyperText Markup Language – the main markup language for Web pages. HTML elements are the basic building-blocks of Web pages.
ISBN	International Standard Book Number – a unique numeric commercial book identifier.
ISSN	International Standard Serial Number – a unique numeric identifier used to identify a print or electronic periodical publication.

Term	Description
KB or KnowledgeBase	The SFX KnowledgeBase is a database and a set of programs that contain knowledge about linking resources. It consists of a comprehensive collection of sources, services, targets, objects, and object portfolios that can be used by the SFX link resolver. This information can be used by an institution to define its digital collection. All information that is part of the original KnowledgeBase is called global. It is globally relevant and is updated by Ex Libris by means of KnowledgeBase updates (released weekly). When a new SFX instance is installed, the SFX KnowledgeBase associated with this instance needs to be customized or localized. All electronic resources that are part of the digital collection of an institution's library need to be described or represented in the SFX database before SFX linking is possible. Localization is performed in three steps: ■ Activating global targets, target services, and object portfolios that are part of the digital collection of an institution's library ■ Adding local information to these global targets, target services, and object portfolios (local coverage, proxy, and local server information) ■ Creating new local targets, target services, and object portfolios if they are not part of the global KnowledgeBase (optional)
LCCN	Library of Congress Control Number – a serially based system of numbering cataloging records in the Library of Congress in the United States. LCCNs are used by librarians to access the associated bibliographic record in the database of the Library of Congress and to obtain information on various book titles in other databases, especially for books that have been published in the United States.
MARC	MAchine-Readable Cataloging – a data format and set of related standards used by libraries to encode and share information about books and other material they collect.
OAI	Open Archives Initiative – an initiative to build a framework to enhance access to archives containing digital content in order to increase the availability of scholarly communication.

Term	Description
OCLC	Online Computer Library Center – a computer library service and research organization that provides bibliographic, abstract, and full-text information.
OCLC nr	Online Computer Library Center number – a unique number used by OCLC to identify objects. The OCLC Control Number is used to enable communication between WorldCat and local catalog systems and can be used as a search key in bibliographic information in non-OCLC resources (for example, Google Books).
ONIX	ONline Information eXchange – XML formats used for capturing metadata relating to books and publications and the licenses under which libraries and other institutions use digital resources.
OpenURL	OpenURL is a standardized format of a Uniform Resource Locator (URL) intended to enable Internet users to more easily find a copy of a resource that they are allowed to access. Although OpenURL can be used with any kind of resource on the Internet, it is most heavily used by libraries to help connect patrons to subscription content. The OpenURL standard is designed to enable linking from information resources such as abstracting and indexing databases (sources) to library services (targets), such as academic journals, whether online, in print, or other formats. The linking is mediated by link resolvers or link-servers that parse the elements of an OpenURL and provide links to appropriate targets available through a library using an OpenURL knowledge base. The National Information Standards Organization (NISO) has developed OpenURL and its data container (the ContextObject) as American National Standards Institute (ANSI) standard Z39.88.
PUBMED / PMID	A free database of references and abstracts on life sciences and biomedical topics. A PMID (PubMed identifier or PubMed unique identifier) is a unique number assigned to each PubMed record.
XML	Extensible Markup Language – a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Term	Description
Z39.50	A client–server protocol for searching and retrieving information from remote computer databases. Interlibrary catalogue searches for interlibrary loans are often implemented with Z39.50 queries. Z39.50 is widely used in library environments and is often incorporated into integrated library systems and personal bibliographic reference software. Interlibrary catalogue searches for interlibrary loans are often implemented with Z39.50 queries.
UTF-8	UCS Transformation Format — 8-bit is a multibyte character encoding for Unicode. UTF-8 can represent every character in the Unicode character set. It is a variable-width encoding, with each character represented by one to four bytes.