How to Export KB Resources to Primo

Version 4.x

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 , Primo, Aleph, Alephino, Voyager, SFX, MetaLib, Verde, DigiTool, Preservation, Rosetta, URM, ENCompass, Endeavor eZConnect, WebVoyáge, Citation Server, LinkFinder and LinkFinder Plus, 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: December 18, 2010

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

Table of Contents

1

Overview

This section includes:

Objectives

The objectives are for Primo (the Harvester) to initiate a request for KnowledgeBase Resources and Categories from MetaLib (the Repository), and for MetaLib to transfer the data in OAI-PMH format to Primo using a standard file transfer protocol (FTP/SFTP).

General Description

MetaLib provides several methods for exporting the resouces:

- Via a batch process which can be scheduled on a regular basis, including a parameter to determine the content of the export:
- Full local KB export from all the local institutions in the MetaLib institutional setup
- Export of the local KBs for a specific group of institutions
- Export of the local KBs for a specific institution.
- Via the MetaLib Management Interface, for export of resources from a specific institution on demand.

MetaLib operates the utility as a fresh export. This means that each process exports all of the active resources for all of the selected institutions.

For each institution, a separate XML output file is created, containing the exported records. The XML file contains a header record, which is based on the

OAI-PMH protocol, with an element describing a unique identifier and a status per each record.

NOTE:

All local fields are retained as part of the export, including authentication data.

The XML files are packaged per institution into zipped tar files, each with a unique name.

At the end of the export process, the export packages are copied to a pre-defined dedicated directory for delivery purposes (FTP/SFTP).

2

Setting up the Export Process

This section includes:

Configuring the Export Directory

Before activating the export process for the first time, it is necessary to configure the dedicated directory for the exported files.

To configure the export directory:

1 Define the environment parameter **`primo_export`** in the `./metalib_conf/metalib_start` file, as follows:

```
cd $metalib_conf
```

```
vi metalib_start
```

2 Add the following line to the end of the file:

```
setenv primo_export $dat01_dev/dat01/export
```

3 Exit and save the changes to the file.

4 Enter the following commands in order for MetaLib to recognize the new parameter:

```
source metalib_start
```

```
metalib_conf_create
```

NOTE:

The value defined above is the default directory.

When using NFS (Network File System), it is possible to define any other directory recognized by the network.

Running the Export via Cron

The Export KnowledgeBase Resources to Primo utility can be set up to run automatically as a cron job. The cron job script (primo_export.cron) is located under the ./aleph/proc directory.

Make sure that this file is executable. If it is not executable, use the chmod command to set it as follows:

```
chmod +x $aleph_proc/primo_export.cron
```

The primo_export.cron script requires input parameters for execution. The parameters are in the following format, where <*LIBRARY*>=DAT01 and <*SELECTION*>={ALL | GROUPNAME | INSTITUTION_CODE}:

```
<LIBRARY>,<SELECTION>,<email>
```

To add the export cron job to your cron jobs list, log in as a metalib user:

For SUN machines, enter the following commands:

```
setenv EDITOR vi
crontab -e
```

For Linux machines, enter the following command:

```
crontab –e -u metalib
```

Add the cron job, as required.

Example 1: Full Export from All the Institutions

For all institutions, the following cron job runs the primo_export.cron every day at 6 a.m.:

```
# Run primo_export.cron every day at 6 a.m. for all institutions:
0 6 * * * /exlibris/metalib/m4_1/aleph/proc/primo_export.cron
"DAT01,ALL,central.admin@myuni.co.ac"
```

Example 2: Export of Institutions Affiliated with a Group

For **mygroup** group, the following cron job runs the primo_export.cron every day at 6 a.m.:

```
# Run primo_export.cron every day at 6 a.m. for mygroup:
0 6 * * * /exlibris/metalib/m4_1/aleph/proc/primo_export.cron
"DAT01,MYGROUP,central.admin@myuni.co.ac"
```

Example 3: Single Institution

For the **openu** institution, the following cron job runs the `primo_export.cron` every day at 6 a.m.:

```
# Run primo_export.cron every day at 6 a.m. for openu:
0 6 * * * /exlibris/metalib/m4_1/aleph/proc/primo_export.cron
"DAT01,OPENU,central.admin@myuni.co.ac"
```

The procedure creates the `p_primo.log` log file under the `./dat01/scratch` directory and sends an e-mail message on completion if the recipient's address is supplied.

3

Running the Export from the Management Interface (/M)

This document contains the instructions used to export the KB resources to Primo. In the following example, the management user is Central Administration and the group is CENTRAL.

To export resources from a specific institution on demand:

1 Log in to the Management Interface (/M) as Institutional or Group Administrator.

2 Click **GO TO ONGOING MENU** to display the MetaLib Ongoing Main menu.

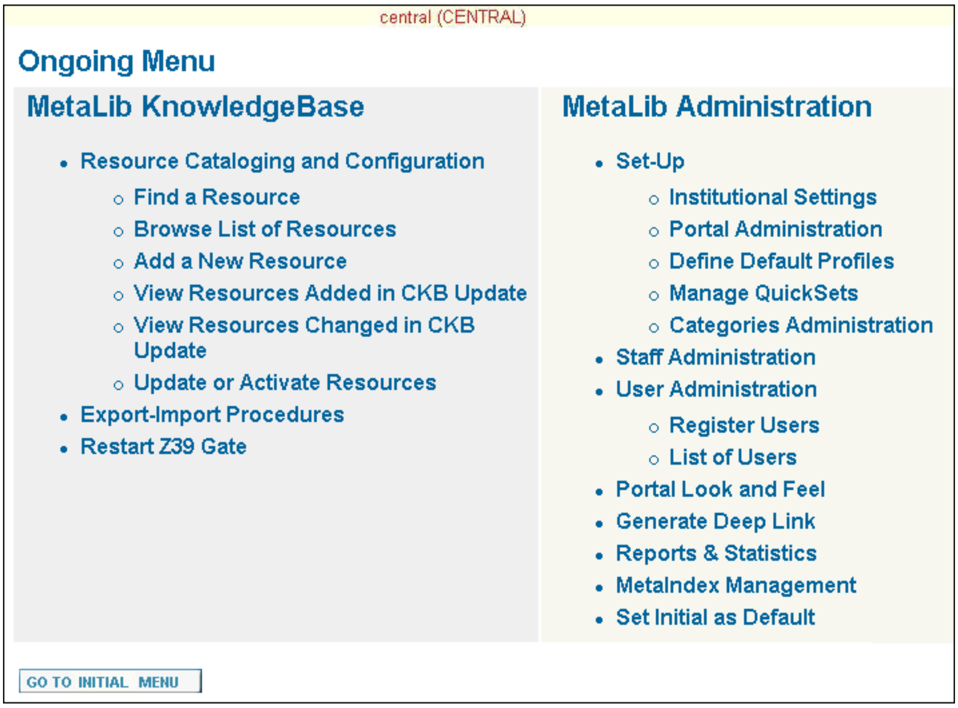
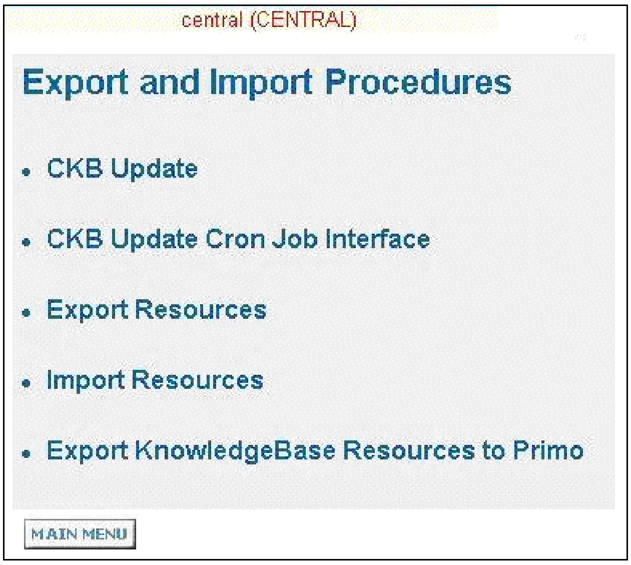


Figure 1: MetaLib Ongoing Main Menu

3 Click **Export-Import Procedures** to access the Export and Import Procedures page.



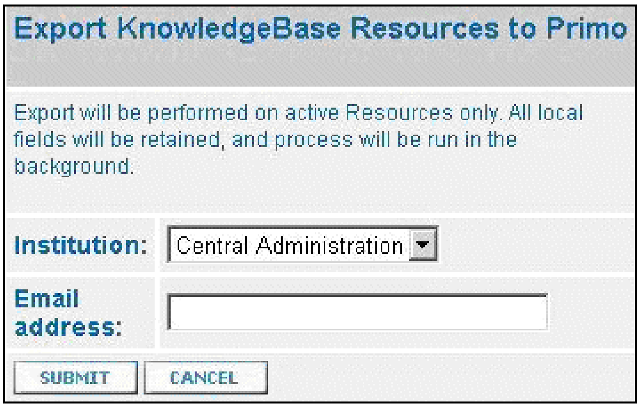
central (CENTRAL)

Export and Import Procedures

- CKB Update
- CKB Update Cron Job Interface
- Export Resources
- Import Resources
- Export KnowledgeBase Resources to Primo

Figure 2: Export and Import Procedures Page

4 Click **Export KnowledgeBase Resources to Primo** to display the .



Export KnowledgeBase Resources to Primo

Export will be performed on active Resources only. All local fields will be retained, and process will be run in the background.

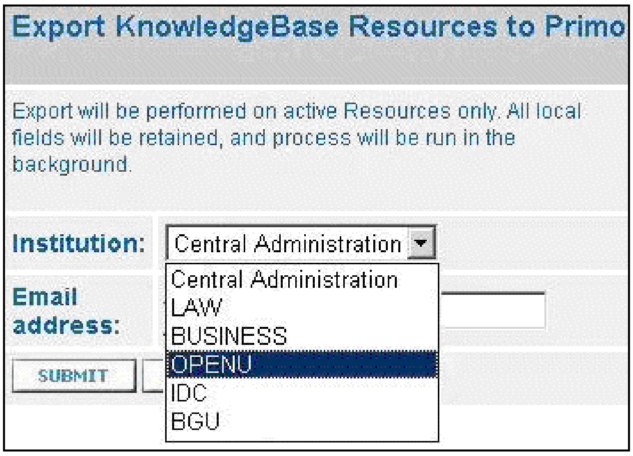
Institution: Central Administration

Email address:

Figure 3: Export KnowledgeBase Resources to Primo Page

5 If you are an Institutional administrator, enter a valid e-mail address and click **SUBMIT**.

If you are a Group administrator, you must first select an institution from the Institution drop-down list, as shown in the following figure.



Export KnowledgeBase Resources to Primo

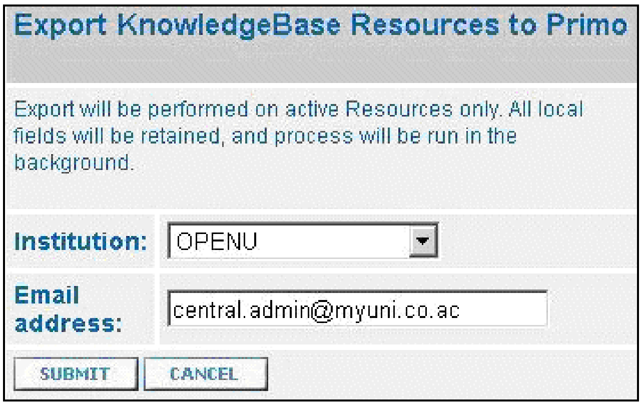
Export will be performed on active Resources only. All local fields will be retained, and process will be run in the background.

Institution: Central Administration

Email address:

Figure 4: List of Institutions in Group

Select the required institution from the list—for example, **OPENU**.



Export KnowledgeBase Resources to Primo

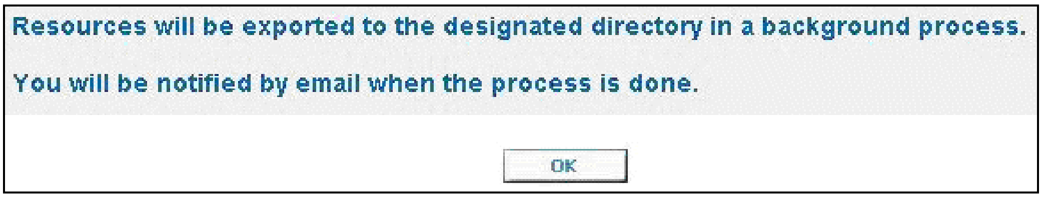
Export will be performed on active Resources only. All local fields will be retained, and process will be run in the background.

Institution: OPENU

Email address: central.admin@myuni.co.ac

Figure 5: Institution Selected for Export

6 Click **SUBMIT** to proceed with the export.



Resources will be exported to the designated directory in a background process.

You will be notified by email when the process is done.

Figure 6: Export Message

7 Click **OK**to start the export and return to the Export and Import Procedures page.

If you specified an e-mail address, a status message is sent to the recipient after the process completes. For example:

```
Subject: MetaLib Export to Primo: Fri May  4 12:20:44 IDT 2007

Message body:

MetaLib Export to Primo (Fri May  4 12:16:42 IDT 2007)
-------------------------------------------------------------------
----------------
Export to Primo completed successfully from the following
institutions:

OPENU
```

Figure 7: Example of E-Mail on Completion of Export

The exported data, which is in XML format, is placed in compressed tar files under the designated output directory. The general format of the output file is:

`<`*`INST`*`>.<`*`DATE_TIME`*`>.tar.gz.`

For the above example, the following file was created and transferred to the `./dat01/export` directory:

`OPENU.20070504130515.tar.gz`

A

XML Output File Format

The XML record for each resource is made up of several elements. The main element is the <knowledge_unit>.

Each knowledge unit contains the following elements:

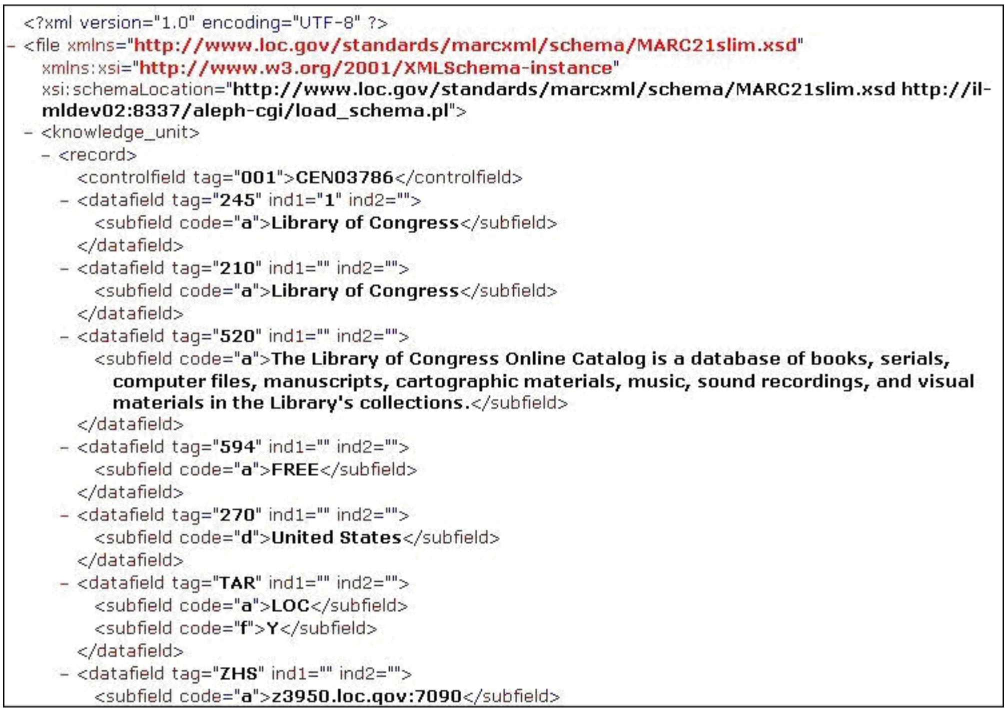
- <record>—This element contains all the fields from the IRD (Information Resource Description)
- <z58_base>—This element contains all the fields from the configuration record. The format of the sub-elements is dependent on the access method.
- <category>—If the resource is assigned to one or more categories, this element is displayed with the main and subcategories.

The general structure of the XML file is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<file xmlns="http://www.loc.gov/standards/marcxml/schema/
MARC21slim.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.loc.gov/standards/marcxml/schema/
MARC21slim.xsd
http://il-mldev02:8337/aleph-cgi/load_schema.pl">
<knowledge_unit>
<record>
  <controlfield tag="001">ABCnnnnn</controlfield>
  <datafield tag="245" ind1="1" ind2=" ">
    <subfield code="a">Resource Name</subfield>
  </datafield>
...

</record>

<z58 base='CONFCODE'>
  <field name='z58_base'>CONFCODE</field>
  <field name='z58_access_method'>Access Method</field>
…

</z58>
```

```
<category>
<main>main1</main>
<sub>sub1</sub>
…
</category>
</knowledge_unit>
```

The following figure shows an example of the header record and some of the elements in the XML file from the OPENU institution for the Library of Congress resource:



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <file xmlns="http://www.loc.gov/standards/marcxml/schema/MARC21slim.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.loc.gov/standards/marcxml/schema/MARC21slim.xsd http://il-
    mldev02:8337/aleph-cgi/load_schema.pl">
  - <knowledge_unit>
    - <record>
        <controlfield tag="001">CEN03786</controlfield>
      - <datafield tag="245" ind1="1" ind2="">
          <subfield code="a">Library of Congress</subfield>
        </datafield>
      - <datafield tag="210" ind1="" ind2="">
          <subfield code="a">Library of Congress</subfield>
        </datafield>
      - <datafield tag="520" ind1="" ind2="">
          <subfield code="a">The Library of Congress Online Catalog is a database of books, serials,
            computer files, manuscripts, cartographic materials, music, sound recordings, and visual
            materials in the Library's collections.</subfield>
        </datafield>
      - <datafield tag="594" ind1="" ind2="">
          <subfield code="a">FREE</subfield>
        </datafield>
      - <datafield tag="270" ind1="" ind2="">
          <subfield code="d">United States</subfield>
        </datafield>
      - <datafield tag="TAR" ind1="" ind2="">
          <subfield code="a">LOC</subfield>
          <subfield code="f">Y</subfield>
        </datafield>
      - <datafield tag="ZHS" ind1="" ind2="">
          <subfield code="a">z3950.loc.gov:7090</subfield>
```

Figure 8: Example of <record> in XML file