How to Set Up Support for Nedap RFID in Aleph

Version 20.0

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, URM, Voyager, ENCompass, Endeavor eZConnect, WebVoyage, 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: March 1, 2009

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

Table of Contents

1 General

Radio Frequency Identification (RFID) is an automatic identification method, relying on storing and remotely retrieving data using devices called RFID tags or transponders. An RFID tag is an object that can be attached to or incorporated into a product for the purpose of identification using radio waves. Chip-based RFID tags contain silicon chips and antennas.

RFID tags can be used in various implementations for attaching identification or other information to any object. Once in their receiving range, radio-enabled machines can identify the object or read other information that is attached to the object and use it a large range of purposes.

RFID tags are mainly used in libraries for the following purposes:

- Automatically identifying an item during the check-in and check-out process.

  Replacing barcode scanners and other traditional methods of identifying items, an RFID-enabled circulation desk can use the RFID reader to identify the item that is on the check-out or check-in counter by reading its RFID tag. This can save the library staff the time and effort that barcode scanning entails.

- Using security gates to detect an un-authorized check-out.

  RFID-enabled security gates can be placed at the library entrances. The gates can read the RFID tag of every item that comes into their radio range and detect whether or not the item has been tagged as ‘allowed for check-out’. If an item leaves the library without the RFID tag being updated, the gates will issue an alarm for the staff to act upon.

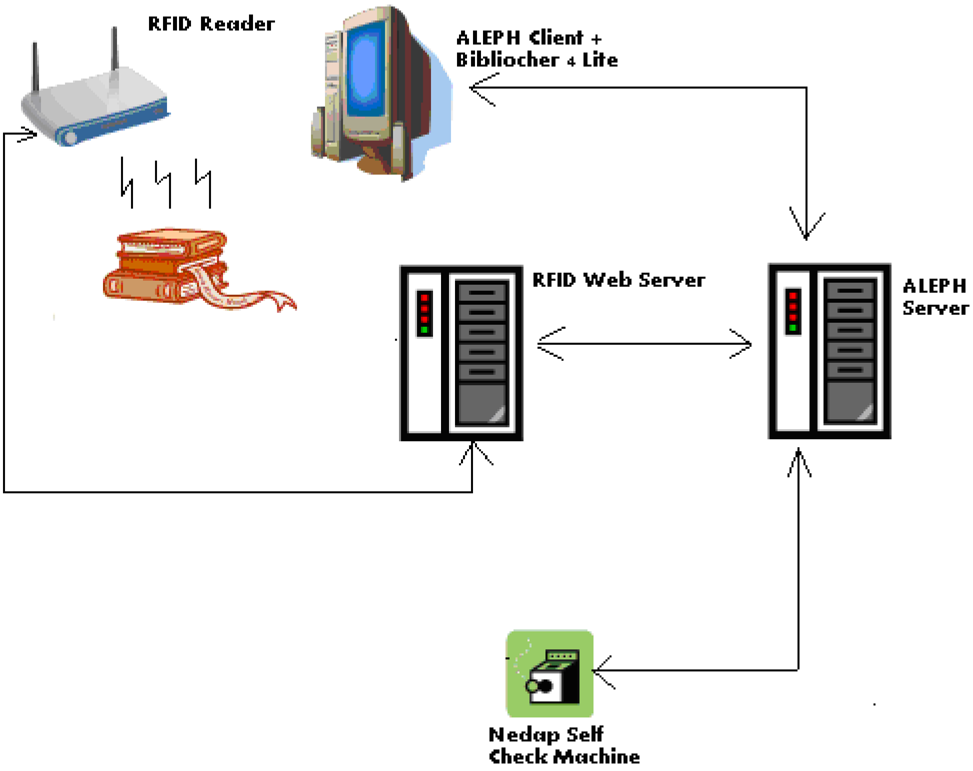
- Inventory management

  RFID-enabled machines are very helpful for processing inventory management actions. For example, automatic sorting of items can be done by RFID-enabled machines that sort items based on information stored on the RFID tags. Also, a physical inventory check can be much easier to perform if the item barcodes are read by an RFID-enabled scanner rather than a barcode scanner.

RFID tags and readers of different vendors are gaining popularity. The purpose of this document is to describe how ALEPH can be set up to work with in a library that is RFID-enabled, where the RFID environment is based on Nedap RFID tags (http://www.Nedaplibrary.com/).

2 System architecture

The following diagram shows how the ALEPH system and the RFID readers interact:



The following is a description of the participating elements and their relationships:

- ALEPH Client + Bibliochecker4Lite

  The ALEPH client station is equipped with the Nedap Bibliochecker4Lite software. The Bibliochecker4Lite software is the component which updates ALEPH with the information on the item that has been detected by the RFID Reader.

- RFID Reader

  The Nedap RFID reader is able to read the RFID tag information that is found within its radio range and passes the information to the ALEPH\Bibliochecker4Lite client. Communication with the RFID reader is done over UDP.

Note

For the system to work properly, only one RFID tag can be in the radio range of the RFID Reader when it is being used.

- ALEPH Server

  The ALEPH server interacts with the ALEPH client to execute the check-out\check-in actions and the item update actions. When actions that affect the RFID tag are triggered, the ALEPH server updates the RFID Web Server on the details of the action and on the RFID reader that is responsible for updating the RFID tag.

- RFID Web Server

  This is a server running the Nedap RFID Web Server software. This server receives from the ALEPH server information on where the updated RFID tag is found (that is, which RFID reader should act on the update request), and on what the update action is (enabling security bit, updating item information, and so on). The RFID Web Server triggers the write action of the RFID Reader, and obtains success or failure information which is reported back to the ALEPH server.

The communication between the ALEPH server and the RFID Web Server is done based on SOAP messages, as described in the Nedap RFID Web Server WSDL document.

3 Functionality

The Nedap RFID-enabled ALEPH system supports the following RFID-based actions:

- Identifying items
- Checking-out and checking-in items.
- RFID-enabled self check actions.
- Update of the RFID tags when updating the item information.

3.1 Identifying items

The RFID-enabled item is identified by the GUI modules in the following manner:

The ALEPH client has the Bibliocheck4Lite software running. The Bibliocheck4LiteSettings.xml file installed on your workstation must contain pre-defined settings such as the following in order to carry out the following actions:

Disclaimer

Third party products such as Nedap are not under the control of Ex Libris, and Ex Libris is not responsible for any changes or updates to Nedap. Furthermore, Ex Libris is not responsible for providing any end user support with respect to Nedap. All end user support requests should be addressed to Nedap.

- Define the Bibliocheck4Lite software to connect to the RFID Reader using UDP.

```
<RfConnection>
        UDPSocketConnection
</RfConnection>
```

- Define the UDP location of the RFID reader.

```
<UDPSocketConnection>
    <ReaderAddress>
        IP address of the RFID Reader
    </ReaderAddress>
    <ReaderPort>
        79
    </ReaderPort>
</UDPSocketConnection>
```

- Define the Bibliocheck4Lite software to only read the tag identification.

```
<AutoMode>
        Manual
</AutoMode>
```

- Define the Bibliocheck4Lite software to always poll the RFID Reader.

```
<AutoStart>
        True
</AutoStart>
```

- Define the input string’s delimiter.

```
<DelimiterString>
        \r
</DelimiterString>
```

In your ALEPH GUI, the cursor is placed in the toolbar in which the item barcode is to be read. For example, if the item is required to be checked out to a patron, the Circulation Loan tab will be activated, and the patron ID will be read. Following that, the cursor is placed in the item bar.

The RFID-tagged item is placed in range of the RFID reader. The item ID is automatically read into the active bar. In the example above, the barcode that has been read by the reader is read into the Item Bar of the Circulation Loan Tab. The ALEPH module workflow can continue as if the item barcode was manually scanned in.

3.2 Check-out and check-in

Whenever an item is checked out the RFID Reader will be triggered to turn off the security bit.

Whenever an item is checked in the RFID Reader will be triggered to turn on the security bit.

If during the check-out or check-in actions the RFID tag has been successfully updated, the GUI can issue a success message such as:

RFID transponder successfully updated.

Whether or not this message is issued depends on the value of the *SuccessMessage* parameter, found under the *[RfidMedia]* section of circ.ini.

If during the check-out or check-in actions the RFID tag has not been successfully updated, the GUI issues an error message, such as

Failed to update RFID Transponder.

The error message is included in a dialog box that allows you to carry out two possible actions:

- To close the error dialog box without further action, click **OK**.
- To retry the RFID update action, by resending the SOAP message click **Retry Transponder**.

3.3 Self-check

Self-check is done by using Nedap RFID-enabled self-check machines. The self-check machines can read and update the RFID tags, and update the ALEPH server by using SIP2 transactions.

3.4 Registering item information

When updating the item, the current item's data is written to the transponder which is in range of the RFID reader.

If the RFID tag has been successfully updated then the GUI can issue a success message such as:

RFID transponder successfully updated.

Whether or not this message is issued depends on the value of the *SuccessMessage* parameter, found under the *[RfidMedia]* section of circ.ini, catalog.ini and acq.ini.

If the RFID tag has not been successfully updated, then the GUI will issue an error message, such as:

Failed to update RFID Transponder.

The error message is included in a dialog box that allows you to carry out two possible actions:

- To close the error dialog box without further action, click **OK**.
- To retry the RFID update action, by resending the SOAP message click **Retry Transponder**.

4 Setting up Nedap RFID

4.1 Configuring the RFID web server

Your ALEPH installation contains a table, named tab_rfid.conf, for defining RFID settings. It is located in the ADM library's data_tab directory. Like tab_ncip.conf, the table defines RFID settings in a free *variable=value* format, for example:

```
[General]
rfid_server_address = 10.1.235.173
timeout = 060
Nedap_item_format = FrenchLabel_native
```

rfid_server_address

This variable identifies the IP address on which the RFID Web Server is found.

time_out

This three-digit variable defines the number of milliseconds which the RFID Web server will wait for a response from the RFID Reader before issuing a timeout.

Nedap_item_format

This variable identifies the write data format that is used when activating the RFID Web Server for an item update action. Possible values are:

FrenchLabel_native - Writes data according to the FrenchLabel data form FR01. The system sends the WriteFrenchLabel_nativeSoapIn and waits for an incoming WriteFrenchLabel_nativeSoapOut message with the action result.

The information to be written into the SOAP message's tags is set in the tab_rfid and tab_rfid_translate tables (see 4.2 Setting up writing of action data below).

4.2 Setting up writing of action data

The data format that is used when updating RFID tag information is configured in the Nedap_item_format variable of the **tab_rfid.conf** table. The ALEPH item information that will be written onto the tag is configured in the **tab_rfid** and **tab_rfid_translate** tables.

4.2.1 tab_rfid

The tab_rfid table is used for translating ALEPH information into the SOAP message components:

- Column 1 lists the element names of the update-item message. In other words, if the FrenchLabel_native format is defined in tab_rfid.conf, then the configurable elements of the WriteFrenchLabel_native element will be listed in column 1 of this table.

- Column 2 lists the information that will populate the update-item message element. The ALEPH information that is available for configuration is derived from the Z30 (Item records) Oracle table. The information is configured in this table by specifying the field name. Constants can also be used as input for SOAP message components.

- Column 3 defines if the code of the item field is to be translated to name (e.g. sublibrary code to sublibrary name). This option is available for:
  - LibraryCode tag – using the SUB-LIB section of tab_rfid_translate
  - LogisticPartGroup tags – using the LOGISTIC section of tab_rfid_translate

- Column 4 defines the character conversion routine that is used. The default value is UTF_TO_RFID.

Here is an example of a setup for the FrenchLabel_native format:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!
DocumentNumber             z30-doc-number
LibraryCode                z30-sub-library                 Y
ItemNr                     z30-barcode
TotalItems                 5
Usage                      0
```

4.2.2 tab_rfid_translate

The tab_rfid_translate table is used for translating ALEPH item information from the ALEPH format into a format that can be used in the Nedap SOAP message component. The table is used in conjunction with the tab_rfid table in the following manner:

> Any information that is defined in column 3 of tab_rfid as Y will be translated by the corresponding tab_rfid_translate function. The LibraryCode and the LogisticPartGroup fields (that is LogisticPartGroup1, LogisticPartGroup2, LogisticPartGroup3, LogisticPartGroup4 and LogisticPartGroup5) are the only fields that can be translated by this table.
>
> If LibraryCode is defined in column 3 of tab_rfid as requiring translation, it will be translated by the SUB-LIB section of tab_rfid_translate table.
>
> If one of the LogisticPartGroup fields is defined in column 3 of tab_rfid as requiring translation, it will be translated by the LOGISTIC section of tab_rfid_translate table.

- Column 1 - defines the section.
- Column 3 - defines the ALEPH item information that will be translated.
- Column 2 is the Nedap message value into which that information will be translated.

A sample tab_rfid_translate table follows:

```
!!!!!!!!!!!!!!!-!!!!!-!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
SUB-LIB      01    WID      Main Library
SUB-LIB      02    MED      Medicine Library
SUB-LIB      03    HYL      East Asian Library

LOGISTIC     01    GEN      General Collection
LOGISTIC     02    ARC      Archives Collection
LOGISTIC     01    WID      Main Library
LOGISTIC     05    HYL      East Asian Library
LOGISTIC     01    BOOK     Book
LOGISTIC     02    ISSUE    Issue
```

4.3 Setting RFID reader location and vendor

The [RfidMedia] section in alephcom.ini defines the location of the station’s RFID reader in the RfidReaderAddress variable and the system’s RFID vendor in the Vendor variable. The possible values for the Vendor variable are Nedap and Bibliotheca.

```
[RfidMedia]
RfidReaderAddress = 10.1.235.173
Vendor=Nedap
```

4.4 Activating reader and success messages

To activate the RFID transponder update from the ALEPH client, set the ActivateReader parameter, found under the *[RfidMedia]* section of circ.ini, catalog.ini and acq.ini, to Y.

To issue a message indicating a successful write action to the RFID Reader, set the *SuccessMessage* parameter, found under the *[RfidMedia]* section of circ.ini, catalog.ini and acq.ini, to Y.

```
[RfidMedia]
ActivateReader=Y
SuccessMessage=Y
INITIATED-RFID=N
```

To add an additional RFID button to the items list, allowing the updating of the RFID tag without making a change in the item information, set the INITIATED-RFID parameter, found under the [RfidMedia] section of circ.ini, catalog.ini, and acq.ini to Y.