Voyager®
Primo® Integration User's Guide

July 2007
Field Test

Ex Libris (USA) Inc.
This document contains proprietary information of
Ex Libris (USA) Inc. and is not to be disclosed or used
except in accordance with applicable agreements.

Copyright © 2007 Ex Libris (USA) Inc.
Unpublished and Not for Publication
All Rights Reserved

Copyright © 2007 Ex Libris (USA) Inc.

This material is protected by the copyright laws of the United States and other countries. It may not be reproduced, distributed, or altered in any fashion by any entity (either internal or external to Ex Libris (USA) Inc.), except in accordance with applicable agreements, contracts or licensing.

CONFIDENTIAL INFORMATION

The information herein is the property of Ex Libris (USA) Inc. or its affiliates and any misuse or abuse will result in economic loss.

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

Notice

Every effort was made to ensure that the information in this document was complete and accurate at the time of printing. However, information is subject to change.

Trademarks

Voyager is a registered trademark of Ex Libris (USA) Inc.
Primo is a registered trademark of Ex Libris Limited
WebVoyáge is a trademark of Ex Libris (USA) Inc.
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.
Sun, Sun Microsystems, the Sun logo, Solaris, ONC+, NFS, SunVTS, VIS, JumpStart, Sun Fire, StorEdge, Sun Fireplane Interconnect, WebNFS, Java, Solstice, Solstice AdminSuite, Sun Management Center, Sun Cluster, JVM, ShowMeHow, ShowMe TV, Solaris Webstart, Sun Remote Systems Control, Solaris PC Netlink, and Open Windows are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Contents

About This Document

- Purpose ix
- Intended Audience ix
- Reason for Reissue x
- Document Summary x
- Conventions Used in This Document x
- Document Reproduction/Photocopying xi
- Comment on This Document xii
  - To Submit Comments by E-mail xii

1 Getting Started

- Introduction 1-1
- Purpose of this Chapter 1-1
- Prerequisite Skills and Knowledge 1-1
- Before You Begin 1-2
  - Checklist 1-2

2 Data Extraction

- Introduction 2-1
- Purpose of this Chapter 2-1
- Data Extraction Process Overview 2-1
  - Initial Primo Data Load 2-2
    - Initial Load Generation 2-2
  - Ongoing Updates to Primo Database 2-2
  - Special Purpose Database Extractions 2-4
  - Primo Transaction 2-4
- PrimoExp.ini Configuration File 2-4
  - Stanza Definitions for PrimoExp.ini 2-11
    - [Item Statuses] 2-12

FLDTST

Contents

[Excluded Happening Locations] 2-14
[Excluded Operator Ids] 2-14
[Institution Id] 2-14
[Location to Primo] 2-14
[Enrichment Tags] 2-15
[PrimoExp] 2-16
[XMLWriterProtocol] 2-20
- Pprimoexp Batch Job 2-21
  Scheduling 2-21

3 Patron Services

- Introduction 3-1
- Patron Authentication (Patron Directory Services) 3-1
  Configuration 3-2
  PDS Invoked 3-2
- Patron Information (Primo Library Card) 3-4
- Patron Requests (GetIt!) 3-4

IN Index IN-1

FLD TST

Figures

2 Data Extraction

2-1. PrimoExp.ini 2-5

3 Patron Services

3-1. Primo Sign In / PDS Authentication 3-2
3-2. Get Avail Requests (prompting sign in) 3-3
3-3. Login dialog box 3-4
3-4. Select A Request options 3-5

FLD TST

Figures

FLD TST

Tables

2 Data Extraction

2-1. [Item Statuses] parameters 2-13
2-2. [PrimoExp] parameter/command descriptions 2-16
2-3. [XMLWriterProtocol] parameter descriptions 2-20

FLD TST

Tables

FLD TST

About This Document

Purpose

The purpose of the *Voyager® Primo® Integration User's Guide* is to provide you with the information that you need to do the following.

- Set up the `PrimoExp.ini` configuration file.
- Run the Pprimoexp batch file.
- Set up user authentication with Patron Directory Services.
- Set up the system for processing user requests.

Intended Audience

This document is written for a technical audience responsible for the following.

- Setup of the `PrimoExp.ini` configuration file.
- Initial upload of Voyager data into Primo.
- Writing scripts and scheduling the batch job for running updates.
- User authentication setup.
- Review/analysis of message logs as needed.

FLDTST

Reason for Reissue

This is the first release of this user's guide.

Document Summary

Chapter 1	"Getting Started" Chapter 1 This chapter describes the prerequisites for getting started with the Voyager Primo integration setup.
Chapter 2	"Data Extraction" Chapter 2 This chapter provides information needed to process the initial data extraction from Voyager and ongoing updates.
Chapter 3	"Patron Services" Chapter 3 This chapter provides information about patron services such as user authentication and processing user requests.
Index	The Index is an alphabetical, detailed cross-reference of topics.

Conventions Used in This Document

The following conventions are used throughout this document:

- Names of commands, variables, stanzas, files, and paths (such as `/dev/tmp`), as well as selectors and typed user input, are displayed in `constant width` type.
- Commands or other keyboard input that must be typed exactly as presented are displayed in **`constant width bold`** type.
- Commands or other keyboard input that must be supplied by the user are displayed in ***`constant width bold italic`*** type.
- System-generated responses such as error messages are displayed in `constant width` type.
- Variable *portions* of system-generated responses are displayed in *`constant width italic`* type.
- Keyboard commands (such as **Ctrl** and **Enter**) are displayed in **bold**.
- Required keyboard input such as "Enter **`vi`**" is displayed in **`constant width bold`** type.
- Place holders for variable portions of user-defined input such as **`ls -l`** ***`filename`*** are displayed in ***`italicized constant width bold`*** type.

FLD TST

- The names of menus or status display pages and required selections from menus or status display pages such as “From the **Applications** drop-down menu, select **System-wide**,” are displayed in **bold** type.
- Object names on a window’s interface, such as the **Description** field, the **OK** button, and the **Metadata** tab, are displayed in **bold** type.
- The titles of documents such as *Acquisitions User’s Guide* are displayed in *italic* type.
- Caution, and important notices are displayed with a distinctive label such as the following:

NOTE:
Extra information pertinent to the topic.

IMPORTANT:
Information you should consider before making a decision or configuration.

CAUTION:
Information you must consider before making a decision, due to potential loss of data or system malfunction involved.

TIP:
Helpful hints you might want to consider before making a decision.

RECOMMENDED:
Preferred course of action.

OPTIONAL:
Indicates course of action which is not required, but may be taken to suit your library’s preferences or requirements.

Document Reproduction/Photocopying

Photocopying the documentation is allowed under your contract with Ex Libris (USA) Inc. It is stated below:

All documentation is subject to U.S. copyright protection. CUSTOMER may copy the printed documentation only in reasonable quantities to aid the employees in their use of the SOFTWARE. Limited portions of documentation, relating only to the public access catalog, may be copied for use in patron instruction.

Comment on This Document

Please contact Customer Support to provide us with your feedback. For Customer Support contact information, see SupportWeb at:

`http://www.exlibrisgroup.com/support_center.htm`.

To Submit Comments by E-mail

To submit comments by e-mail, please send your message to:

`docmanager@endinfosys.com`

FLD TST

Getting Started

1

Fld Tst

Introduction

The *Voyager*® *Primo*® *Integration User's Guide* is designed to provide you with the information that you need to successfully utilize Voyager data in a Primo user environment. The following components are a part of the Voyager Primo integration.

- `PrimoExp.ini` for data extraction.
- User authentication with Patron Directory Services (PDS).
- User information provided with Library Card.
- User request processing with Primo's GetIt! function.

Purpose of this Chapter

The purpose of this chapter is to identify the skills and preparations necessary for setting up your environment for Voyager Primo integration.

Prerequisite Skills and Knowledge

To use this document effectively and set up your systems for Voyager Primo integration, you need to have a general understanding of the following.

- Microsoft® Windows-based applications.

- XML and XML-format data as well as editors and related software.
- At least one language or program for writing scripts.
- UNIX® operating system commands and file system.
- Oracle® Database system, including SQL and SQL*Plus.
- Apache web server systems and related inetd services.
- At least one UNIX-based text editor such as `ed` or `vi`.
- Local procedures.

This user's guide assumes you have a working knowledge of Primo setup and access to Primo documentation to include the *Primo Interoperability Guide*.

Before You Begin

Before you can begin your Voyager Primo integration, you need to have the following software installed.

- Voyager (the base product)
- Primo
- Voyager Primo database extraction component

Checklist

In preparation for setting up Voyager Primo integration, gather the following information.

- Identify the elements needed to customize the configuration file. See PrimoExp.ini Configuration File on page 2-4 for additional information.
- Identify the schedule/times for processing updates.
- Identify excluded happening locations.
- Identify excluded operator IDs.
- Identify institution ID.
- Identify statuses to be made available to Primo.

FLD TST

Data Extraction

2

Contents

Introduction **2-1**

Purpose of this Chapter **2-1**

Data Extraction Process Overview **2-1**

- Initial Primo Data Load 2-2
  - Initial Load Generation 2-2
- Ongoing Updates to Primo Database 2-2
- Special Purpose Database Extractions 2-4
- Primo Transaction 2-4

PrimoExp.ini Configuration File **2-4**

- Stanza Definitions for PrimoExp.ini 2-11
  - [Item Statuses] 2-12
  - [Excluded Happening Locations] 2-14
  - [Excluded Operator Ids] 2-14
  - [Institution Id] 2-14
  - [Location to Primo] 2-14
  - [Enrichment Tags] 2-15
  - [PrimoExp] 2-16
  - [XMLWriterProtocol] 2-20

Pprimoexp Batch Job **2-21**

- Scheduling 2-21

FLD TST

Contents

FLD TST

Data Extraction

2

FLD TST

Introduction

Primo discovery takes advantage of the rich data stored in the Voyager database. To do this, bibliographic data is extracted from the Voyager database for an initial load and then subsequently kept current via regularly scheduled updates with Primo. This chapter discusses the data extraction process.

Purpose of this Chapter

The purpose of this chapter is to describe the following.

- Components required for extracting Voyager data for Primo to harvest.
- `PrimoExp.ini` configuration.
- `Pprimoexp` batch file.

Data Extraction Process Overview

The Voyager Primo integration data extraction tools are used for the following activities.

- Initial load of Voyager data for a new installation of Primo.
- Ongoing updates to the Primo database for an existing installation of Primo harvesting Voyager data.

- Other special purposes

Initial Primo Data Load

For the initial Primo database load, you need to do the following.

- Tailor the `PrimoExp.ini` configuration file to your environment/needs.
- Run the `Pprimoexp` batch job.

  Using the parameters set in `PrimoExp.ini`, this step identifies the Voyager records to be uploaded to the Primo database.

  A separate script, `bundlePrimo.ksh`, wraps the Voyager records in a `tar.gz` file for Primo to access.
- Permit Primo to securely access the directory in which the generated files from Voyager are placed. This is the output location for the `tar.gz` files.

> **NOTE:**
> A number of options exist for moving the data to the Primo server for the initial load such as using FTP, a shared NFS mount, manually copying the data, or utilizing a cron/script to copy the data. For additional information, see your implementation support contact to discuss the alternatives for Primo to access Voyager data.

Initial Load Generation

When identifying bibliographic records for the initial data load, the following criteria is used to identify records to be excluded.

- If a bibliographic record is suppressed from the OPAC, a MARCXML record is not generated.
- If a bibliographic record was most recently modified at one of the excluded happening locations as identified in the `PrimoExp.ini` configuration file, a MARCXML record is not generated.
- If a bibliographic record was most recently modified by one of the excluded operators as identified in the `PrimoExp.ini` configuration file, a MARCXML record is not generated.

See PrimoExp.ini Configuration File on page 2-4 for more information about configuring the `PrimoExp.ini` file.

Ongoing Updates to Primo Database

You need to evaluate how frequently to process updates to the Primo database. Consider the following.

FLDTST

- Level of activity within your Voyager database.
- Frequency of bibliographic record changes.
- Frequency of item status changes.
- Time of day when system usage is greatest.
- Time of day when system usage is the least.

One option to consider for ongoing updates to the Primo database is to create and maintain multiple versions of the `PrimoExp.ini` configuration file to serve different purposes as follows.

1. Bibliographic record changes.
2. Holdings availability changes.

Primo uses separate pipes for harvesting Voyager data.

One pipe is used to harvest Voyager data with bibliographic record changes. This is the publishing pipe.

The other pipe is used to harvest Voyager data with holdings availability changes. This is the availability pipe.

Given this approach for harvesting data, you need to do the following.

- Set up two separate directories for Primo to access the generated `tar.gz` files, one to be used for the Primo publishing pipe for Voyager bibliographic record changes and the other for the Primo availability pipe for Voyager holdings availability changes.
- Set up two separate `PrimoExp.ini` files, one to generate records with only bibliographic record changes and the other with only holdings availability changes.

  NOTE:
  The type of content in these generated records is the same. However, different records get generated for Primo harvesting as identified by the triggers (parameters) specified in the `PrimoExp.ini` file. See Table 2-2 on page 2-16 for examples of parameters that affect which records are generated.

  This approach implies that corresponding copies of `Pprimoexp` batch files are named to match such as `Pprimoexp.pub` (for publishing) and `Pprimoexp.ava` (for availability).

Creating separate directories, `PrimoExp.ini` files, and `Pprimoexp` batch files enables your flexibility in scheduling production processing of data changes in the Voyager Primo integrated environment. This allows you to schedule Primo

FLDTST

harvesting of bibliographic record changes from Voyager at an appropriate time and separate from scheduling harvesting of Voyager data with holdings availability changes.

Special Purpose Database Extractions

Aside from the initial Primo data load and ongoing updates, you may want to set up the Voyager Primo integration tools for other special purposes that meet your institution's specific requirements.

For example, you may want to set up the Voyager Primo integration tools to create a testing subset before doing a complete initial load of the database. This can be done using parameters like `BibRangeBegin` and `BibRangeEnd` or `BibsFromFile`. See Table 2-2 on page 2-16 for more information about these parameters.

Primo Transaction

Primo originates an FTP transaction to Voyager to harvest the records selected by the `Pprimoexp` processing for inclusion in the Primo database. Subsequently within Primo, these records are normalized into the Primo Normalized XML (PNX) format.

NOTE:
A number of options exist for moving the data to the Primo server such as using a shared NFS mount, manually copying the data, or utilizing a cron/script to copy the data. See your implementation support contact to discuss alternatives for harvesting data.

PrimoExp.ini Configuration File

Primo uses data extracted from the Voyager database. The configuration file, `PrimoExp.ini` (see Figure 2-1), is used to define what is extracted. It is located in `/m1/voyager/yyydb/ini` where `yyydb` is the database name.

The `PrimoExp.ini` configuration file defines the following elements.

- Item statuses to check for initial load and ongoing updates to the Primo database.
- Happening locations for exclusion. Records associated with these happening locations are excluded from the initial load and/or updates for Primo harvesting.

FLDTST

- Records to be excluded from the initial load and/or ongoing updates determined by operator ID that processed the record. See [Excluded Operator Ids] on page 2-14.
- Institution ID to be included in the availability tag.
- Location definitions for consistency within the Primo database. See [Location to Primo] on page 2-14.
- Tags to identify unique field usage within your institution's database specific to the database extract. See [Enrichment Tags] on page 2-15.
- Extract processing considerations such as date/time, record ID(s), filters, and so on. See [PrimoExp] on page 2-16.
- Additional processing considerations for uploading the Voyager records selected by `PrimoExp.ini`. See [XMLWriterProtocol] on page 2-20.

```
Line#
 1    # Configuration for Voyager Export for Primo
 2
 3
 4    # Voyager status maps to (A)vailable or (U)navailable
             for Primo
 5    [Item Statuses]
 6    Not Charged=A
 7    Charged=U
 8    Renewed=U
 9    Overdue=U
10    Recall Request=U
11    Hold Request=U
12    On Hold=U
13    In Transit=U
14    In Transit Discharged=U
15    In Transit On Hold=U
16    Discharged=A
17    Missing=U
18    Lost--Library Applied=U
```

Figure 2-1. PrimoExp.ini

FLD TST

```
Line#
  19    Lost--System Applied=U
  20    Claims Returned=U
  21    Damaged=U
  22    Withdrawn=U
  23    At Bindery=U
  24    Cataloging Review=A
  25    Circulation Review=A
  26    Scheduled=U
  27    In Process=U
  28    Call Slip Request=U
  29    Short Loan Request=U
  30    Remote Storage Request=U
  31
  32    [Excluded Happening Locations]
  33    ACQ
  34    ACQ PER
  35    ACQ CONT
  36    ACQ APP
  37    ACQ DOC
  38    ACQ GE
  39    ACQ TEMP
  40    ACQ TIP
  41    coecirc
  42    scicirc
  43    geocirc
  44    illcirc
  45    lrccirc
  46
  47    [Excluded Operator Ids]
  48
  49    [Institution Id]
  50    # This value will be included in the Availability tag
```

FLD TST

Figure 2-1. PrimoExp.ini (Continued)

```
Line#
  51    # as subfield a
  52    Id=@DBNAME@
  53
  54    #map of location strings to primo strings
  55    [Location to Primo]
  56
  57    [Enrichment Tags]
  58    # tags added to marc record with additional data
  59    InsertAVATag=949
  60    InsertAUTTag=950
  61    InsertSUBTag=951
  62
  63
  64    [PrimoExp]
  65    # note that stanza.name can be used with -- on the
              command line
  66    # argv(case insensitive) override
  67    # ini(case insensitive) override
  68    # env (all uppercase)
  69    # -B
  70    # BibRangeBegin
  71    #BibRangeBegin=1
  72    # -E
  73    #BibRangeEnd=200
  74    #
  75    # -F
  76    #BibsFromFile=./test/bibs.txt
  77    #
  78    # -C
  79    # changes since a date or date & time
  80    #ChangedSince=YYYYMMDD.HHMMSS
  81    # or
```

FLDTST

Figure 2-1. PrimoExp.ini (Continued)

```
Line#
  82    # changes since the last (saved) run
  83    # current run time will be saved
  84    ChangedSince=./test/work/since.txt
  85    #
  86    # Whether to include bibs affected by heading changes
  87    # -H Y|N
  88    HeadingChanges=Y
  89    #
  90    # Whether to filter records to prevent sending records
                that have
  91    # had no change in their availability tags.  Defaults to
                N.
  92    # -A Y|N
  93    #AvailFilter=N
  94    #
  95    # for filenames
  96    # note that $str$ will be passed to date format so that
                you can
  97    # specify date in the these files
  98    # -L
  99    LogFile=primo.export.$yyMMdd$.log
 100    #LogFile=/dev/null
 101    LogFileDir=/m1/voyagerdev/2006.5/dev65ncdb/logs
 102    # -v
 103    # the logging level
 104    LogLevel=10
 105    # whether to duplicate log output to stdout
 106    LogToStdOut=N
 107    # -d
 108    # DataBase=
 109    #
 110    # DatabaseHost
 111    DatabaseHost=@DBHOST@
```

Figure 2-1. PrimoExp.ini (Continued)

FLD TST

```
Line#
 112  #DatabaseHost=208.178.237.40
 113  # DatabasePort
 114  # -e
 115  #UserPass=@DBUSER@/@DBPASS@
 116  Database=VGER
 117  Protocol=com.endinfosys.voyager.extract.OAIPMHXmlWriter
              Protocol
 118
 119  # must do items, then mfhds, then bibs
 120  Task=com.endinfosys.voyager.extract.ExtractItemsPrimo
 121  Task=com.endinfosys.voyager.extract.ExtractMFHDsPrimo
 122  Task=com.endinfosys.voyager.extract.ExtractBibsPrimo
 123  DoItems=Y
 124  DoMfhds=Y
 125
 126  DelBibsDir=/m1/voyagerdev/2006.5/dev65ncdb/rpt
 127  DelBibFile=deleted.bib.marc
 128  DelMFHDsDir=/m1/voygerdev/2006.5/dev65ncdb/rpt
 129  DelMFHDsFile=deleted.mfhd.marc
 130  DelItemsDir=/m1/voygerdev/2006.5/dev65ncdb/rpt
 131  DelItemsFile=delete.item
 132
 133  [XMLWriterProtocol]
 134  # OAIPMHXmlWriterProtocol will include the bib id
 135  #  as the last element in the file name before
 136  #  the .xml extension.
 137  # Note the path must be coordinated with the
              bundlePrimo.ksh
 138  #  script execution.
 139  File=./test/work/primo.export.$yyMMddhhmmss$.xml
 140  XSL=
 141  # max number of records to group in a single tar file
```

Figure 2-1. PrimoExp.ini (Continued)

Fld Tst

```
Line#
  142  #  all in single group if 0 or undefined
  143  recsPerGroup=1000
  144  # prefix for record identifiers, must be present, may be
           empty
  145  idPrefix=
  146
  147  #
  148  # EVERYTHING BELOW THIS LINE IS INTERNAL DO NOT MODIFY
  149  #
  150
  151  [OAIPMHXmlWriterProtocol]
  152  BibUpdate=com.endinfosys.voyager.extract.WriterTrans
  153  BibUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
  154  BibUpdate=com.endinfosys.voyager.primo.SummarizeItemsTr
           ans
  155
  156  BibDelete=com.endinfosys.voyager.extract.WriterTrans
  157  BibDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
  158  BibDelete=com.endinfosys.voyager.primo.SummarizeItemsTr
           ans
  159
  160  ItemUpdate=com.endinfosys.voyager.extract.WriterTrans
  161  ItemUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdat
           e
  162  ItemUpdate=com.endinfosys.voyager.primo.SummarizeItemsT
           rans
  163
  164  ItemDelete=com.endinfosys.voyager.extract.WriterTrans
  165  ItemDelete=com.endinfosys.util.TransUpdateRec:FlagDelet
           e
  166  ItemDelete=com.endinfosys.voyager.primo.SummarizeItemsT
           rans
  167
  168  MFHDUpdate=com.endinfosys.voyager.extract.WriterTrans
```

Figure 2-1. PrimoExp.ini (Continued)

FLDTST

```
Line#
 169  MFHDUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdat
          e
 170  MFHDUpdate=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 171
 172  MFHDDelete=com.endinfosys.voyager.extract.WriterTrans
 173  MFHDDelete=com.endinfosys.util.TransUpdateRec:FlagDelet
          e
 174  MFHDDelete=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 175
 176  [TransUpdateRec]
 177  FlagUpdate=ASSIGN,action,m
 178
 179  FlagDelete=ASSIGN,action,d
 180  FlagDelete=ASSIGN,bib.000:5,d
 181  FlagDelete=MOVE,bib.035:a,bib_id
```

Figure 2-1. PrimoExp.ini (Continued)

Stanza Definitions for PrimoExp.ini

This section defines the stanzas in the `PrimoExp.ini` file that need to be customized for Voyager Primo integration.

The following is a list of the stanzas in `PrimoExp.ini`.

- [Item Statuses]
- [Excluded Happening Locations]
- [Excluded Operator Ids]
- [Institution Id]
- [Location to Primo]
- [Enrichment Tags]
- [PrimoExp]
- [XMLWriterProtocol]

FLDTST

[Item Statuses]

The `[Item Statuses]` stanza identifies which item statuses in Voyager are defined as Available (A) or Unavailable (U) for Primo's purposes. A change in status causes a record to be selected for the availability pipe.

FLD TST

See Table 2-1 for a list of the item statuses and default settings stored in the `[Item Statuses]` stanza of `PrimoExp.ini`.

Table 2-1. [Item Statuses] parameters

Parameter	Default Setting
Not Charged=	A
Charged=	U
Renewed=	U
Overdue=	U
Recall Request=	U
Hold Request=	U
On Hold=	U
In Transit=	U
In Transit Discharged=	U
In Transit On Hold=	U
Discharged=	A
Missing=	U
Lost--Library Applied=	U
Lost--System Applied=	U
Claims Returned=	U
Damaged=	U
Withdrawn=	U
At Bindery=	U
Cataloging Review=	A
Circulation Review=	A
Scheduled=	U
In Process=	U
Call Slip Request=	U
Short Loan Request=	U
Remote Storage Request=	U

FLD TST

CAUTION:
Changes to these parameters after your institution has an established

ongoing update/upload process may cause a significant data upload to the Primo database utilizing system resources as needed to complete the process.

[Excluded Happening Locations]

The `[Excluded Happening Locations]` stanza is used to identify the records associated with these locations to be excluded from consideration for uploading to the Primo database. These are location codes as established in Voyager System Administration.

The locations identified in the default configuration file are simply there as an example. Delete any that do not apply to your institution and enter your own location codes. Enter one location code per line.

[Excluded Operator Ids]

The `[Excluded Operator Ids]` stanza is used to identify the records associated with these operator ID(s) to be excluded from consideration for uploading to the Primo database. Enter one operator ID per line.

[Institution Id]

The `[Institution Id]` stanza is used to specify your institution. The institution ID is included as a subfield in the Availability tag of records prepared for the Primo database and is used by Primo to determine ownership of the records.

The Voyager institution ID is set one time at system installation. Verify this setting to insure its accuracy.

[Location to Primo]

The `[Location to Primo]` stanza is used to maintain consistency within Primo for item location codes. Primo is capable of retrieving records from many sources which may not be consistent in naming item location codes. The `[Location to Primo]` stanza can also be used by libraries to collapse locations or split locations for display in Primo.

Enter the Voyager item location code first and the Primo equivalent using the following format.

[Voyager Location Name]=[Primo Location Name]

See the following for an example.

FLD TST

```
Music Reserve=Reserve

Science Reserve=Reserve
```

Each Voyager/Primo item location code entry should be on a separate line in the configuration file.

NOTE:
Locations that are not listed in the `[Location to Primo]` stanza pass to Primo unchanged.

[Enrichment Tags]

The `[Enrichment Tags]` stanza is used to specify the tags used to include the availability, author enrichment, and subject enrichment information in the extract.

By default, the configuration file makes the following assignments.

```
InsertAVATag=949

InsertAUTTag=950

InsertSUBTag=951
```

The 949 code identifies the placement of availability information in the MARCXML output.

The 950 code identifies the placement of name/author cross-reference information.

The 951 code identifies the placement of subject cross-reference information.

The defaults in the `[Enrichment Tags]` stanza assume that your institution is not already using 949, 950, and 951 in the MARC records of your database.

If, for example, your institution currently uses MARC 949 for another purpose, you can specify a different, unused 9XX number for the `InsertAVATag` in the `[Enrichment Tags]` stanza.

The same logic applies to the `InsertAUTTag` and the `InsertSUBTag` parameters.

Changing the default requires a matching change in the Primo configuration for harvesting.

Fld Tst

[PrimoExp]

The `[PrimoExp]` stanza is used to identify database extract processing considerations such as number of records to process, where to log messages, whether to use filters and so on.

The parameters identified in the `[PrimoExp]` stanza may be entered on the command line or set within the `PrimoExp.ini` configuration file. Parameters specified on the command line override any settings within the configuration file.

See Table 2-2 for a description of these parameters.

IMPORTANT:
Do not edit the following lines of the PrimoExp.ini configuration file as numbered in Figure 2-1 on page 2-5: 116 - 124 and 145 to the end of the configuration file.

Table 2-2. [PrimoExp] parameter/command descriptions

Parameter/Command	Description
BibRangeBegin= -B	Use this parameter to specify the bibliographic record ID number for the beginning of a range of bibliographic record IDs. Use this with -E. Used in combination with -E, you are able to process a subset of records from the entire database. **IMPORTANT:** *HeadingChanges must be set equal to N when BibRangeBegin is used.*
BibRangeEnd= -E	Use this parameter to specify the bibliographic record ID number for the end of a range of bibliographic record IDs. Use this with -B. Used in combination with -B, you are able to process a subset of records from the entire database. **IMPORTANT:** *HeadingChanges must be set equal to N when BibRangeEnd is used.*

FLDTST

Table 2-2. [PrimoExp] parameter/command descriptions

Parameter/Command	Description
BibsFromFile= -F	Use this parameter to identify the name of a file that contains bibliographic records to be processed/stored in a file that is separate from the database. This assumes that the file is formatted with one bibliographic record ID per line/row.
ChangedSince= -C	Use this parameter to indicate which Voyager records to evaluate for export to Primo based on a date/timestamp format as follows. `YYYYMMDDHHMMSS` where ... `YYYY`=year `MM`=Month `DD`=Day `HH`=Hours `MM`=Minutes SS=Seconds If no time is specified, 12:00AM is the default setting. Alternatively, store a date such as ***`20070711`*** and time in a file called `since.txt` with a path of your choosing. For example: `ChangedSince=./test/work/since.txt`
HeadingChanges= -H	Use this parameter to indicate the following. • Yes (`Y`), find changed headings and include all records affected. • No (`N`), do not look for changed headings. This parameter should be set to `Y` when the `PrimoExp.ini` file is used to harvest records for the Primo publishing pipe. This parameter should be set to `N` when the `PrimoExp.ini` file is used to harvest records for the Primo availability pipe. **IMPORTANT:** *The HeadingChanges parameter requires that a setting is specified for the ChangedSince parameter. HeadingChanges can only be process if there is a ChangedSince date.*

FLD TST

Table 2-2. [PrimoExp] parameter/command descriptions

Parameter/Command	Description
AvailFilter= -A	Use this parameter to indicate which Voyager records to evaluate for export to Primo based on availability changes. `Y`=Yes (filter for availability changes) `N`=No (do not filter for availability changes) This parameter should be set to `Y` when the `PrimoExp.ini` file is used to harvest records for the Primo availability pipe. This parameter should be set to `N` when the `PrimoExp.ini` file is used to harvest records for the Primo publishing pipe.

LogFile=
-L
	Use this parameter to set the file name for the log file.
	For example:
	`primo.export.$yyMMdd$.log`
	The system automatically updates the `$yyMMdd$` with the date/time component of the file name.
	To specify that no log file is to be created, use the following parameter setting.
	`LogFile=/dev/null`
	 TIP:
	Use a naming convention that reflects whether the PrimoExp.ini configuration file is being used for Voyager records to be exported to the Primo publishing pipe or the Primo availability pipe.

LogFileDir=	Use this parameter to set the path for the log file.
LogLevel= -v	Leave the `LogLevel` parameter setting as specified in the configuration file unless directed otherwise by Ex Libris support staff.
LogToStdOut=	Leave the `LogToStdOut` parameter setting as specified in the configuration file unless directed otherwise by Ex Libris support staff.

FLDTST

Table 2-2. [PrimoExp] parameter/command descriptions

Parameter/Command	Description
Database= -d	Use this parameter to set the Oracle database name. The default is **VGER**. The `-d` command is used to combine the Database, DatabaseHost, and DatabasePort parameters into a single string. For example: `-d 208.178.237.40:1521@VGER`
DatabaseHost= -d	Use this parameter to set the database host IP address (the same as in `voyager.env`). The `DatabaseHost` parameter is set one time at system installation. Verify this setting to insure its accuracy. The `-d` command is used to combine the Database, DatabaseHost, and DatabasePort parameters into a single string. For example: `-d 208.178.237.40:1521@VGER`
DatabasePort= -d	Use this parameter to set the database port. The `-d` command is used to combine the Database, DatabaseHost, and DatabasePort parameters into a single string. For example: `-d 208.178.237.40:1521@VGER`
UserPass= -e	Use this parameter to specify the Oracle database user name and password. For example: `-e user/pass`
DelBibsDir=	Edit the parameter provided in the `PrimoExp.ini` file to match your Voyager configuration.
DelBibFile=	Leave this parameter setting as specified in the configuration file.
DelMFHDsDir=	Edit the parameter provided in the `PrimoExp.ini` file to match your Voyager configuration.
DelMFHDsFile=	Leave this parameter setting as specified in the configuration file.
DelItemsDir=	Edit the parameter provided in the `PrimoExp.ini` file to match your Voyager configuration.
DelItemsFile=	Leave this parameter setting as specified in the configuration file.

Fld Tst

[XMLWriterProtocol]

The `[XMLWriterProtocol]` stanza defines the storage location of the XML records, number of records in the tar.gz file, and so forth for the XML records targeted for export to the Primo database.

See Table 2-3 for a description of the parameters used in the `[XMLWriterProtocol]` stanza.

Table 2-3. [XMLWriterProtocol] parameter descriptions

Parameter	Description

File=	Use this parameter to identify the file path for storing the XML files generated from harvesting Voyager records for export to the Primo database.
	
	TIP:
	Use a naming convention that reflects whether the PrimoExp.ini configuration file is being used for Voyager records to be exported to the Primo publishing pipe or the Primo availability pipe.
	
	For example:
	`./VPrimoExport/Pub/primo.export.$yyMMddhhmmss$.xml`
	`./VPrimoExport/Avail/primo.export.$yyMMddhhmmss$.xml`
	
	NOTE:
	The path you specify needs to be coordinated with the execution of the `bundlePrimo.ksh` script. The `bundlePrimo.ksh` script wraps the individual records into the tar.gz files that are accessed by the Primo publishing and availability pipes for import.

XSL=	Leave this parameter blank. No setting required.
recsPerGroup=	Use this parameter to identify the number of records you prefer to be saved in each tar.gz file. Check to verify that this setting is consistent with the Primo limit.
idPrefix=	Leave this parameter blank. No setting required.

FLD TST

Pprimoexp Batch Job

The `Primoexp` batch job creates XML files for export that conform to the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) schema and places them in the location identified by the ***`File=`*** parameter set in the `[XMLWriterProtocol]` stanza. See Table 2-3.

Scheduling

The Voyager Primo integration setup is designed to give you the flexibility to establish more than one production schedule.

- One production job may be set to run once per day to allow for Primo harvesting of Voyager data with bibliographic record changes through the publishing pipe.

  IMPORTANT:
  This requires an update to the bibliographic index in the Primo environment.

- Another production schedule may be set to run several times per day such as once per hour to allow for exporting Voyager holdings records with availability changes.

  NOTE:
  Voyager availability data harvested through the Primo availability pipe only updates the availability information without changing the bibliographic information in Primo.

The `Pprimoexp` batch job schedule determines when Voyager records are examined for export based on the criteria set in the `PrimoExp.ini` file.

Separately, the Primo publishing and availability pipes are scheduled to FTP whatever `tar.gz` files are available for export. For more information refer to Schedule Tasks in the *Primo Administrator Guide*.

FLD TST

FLD TST

Patron Services

3

Contents

Introduction **3-1**

Patron Authentication (Patron Directory Services) **3-1**

- Configuration 3-2
- PDS Invoked 3-2

Patron Information (Primo Library Card) **3-4**

Patron Requests (GetIt!) **3-4**

FLD TST

Contents

FLD TST

Patron Services

3

Introduction

The following patron services are provided by Voyager Primo integration.

- Patron authentication with Patron Directory Services.
- Patron information with Primo Library Card.
- Patron requests with Primo GetIt!.

Patron Authentication (Patron Directory Services)

Consistent with other Ex Libris products, Primo utilizes Patron Directory Services (PDS) to process patron authentication.

PDS does not have a patron database of its own. Instead, it provides the flexibility to pass authentication credentials to a designated target or redirect the user to an external authentication page.

The Voyager Primo environment uses the Primo Remote CGI Hook to enable user authentication with the Voyager patron database. It provides the following functions.

- Patron authentication using credentials sent from PDS.
- Borrower information sent to PDS.

FLD TST

- PDS Single Sign On (SSO) support for Primo.

Configuration

There are two server configuration components needed to enable user authentication with PDS.

1. The main PDS configuration table called tab_service.<institute> needs to be configured for communicating with the Voyager web service application.

   More information about PDS configuration is available in the *Patron Directory Services* manual.

2. The Voyager server XML needs to be configured with database connection information such as username, password, host, SID (system ID), and so forth. This can be set up once and is handled by the Ex Libris Voyager installation team.

PDS Invoked

Primo invokes PDS when users attempt to sign in. See Figure 3-1.

Figure 3-1. Primo Sign In / PDS Authentication

If a patron has not signed in prior to selecting **Get Avail Requests** (see Figure 3-2), clicking this option displays the Login dialog box. See Figure 3-3.

FLD TST

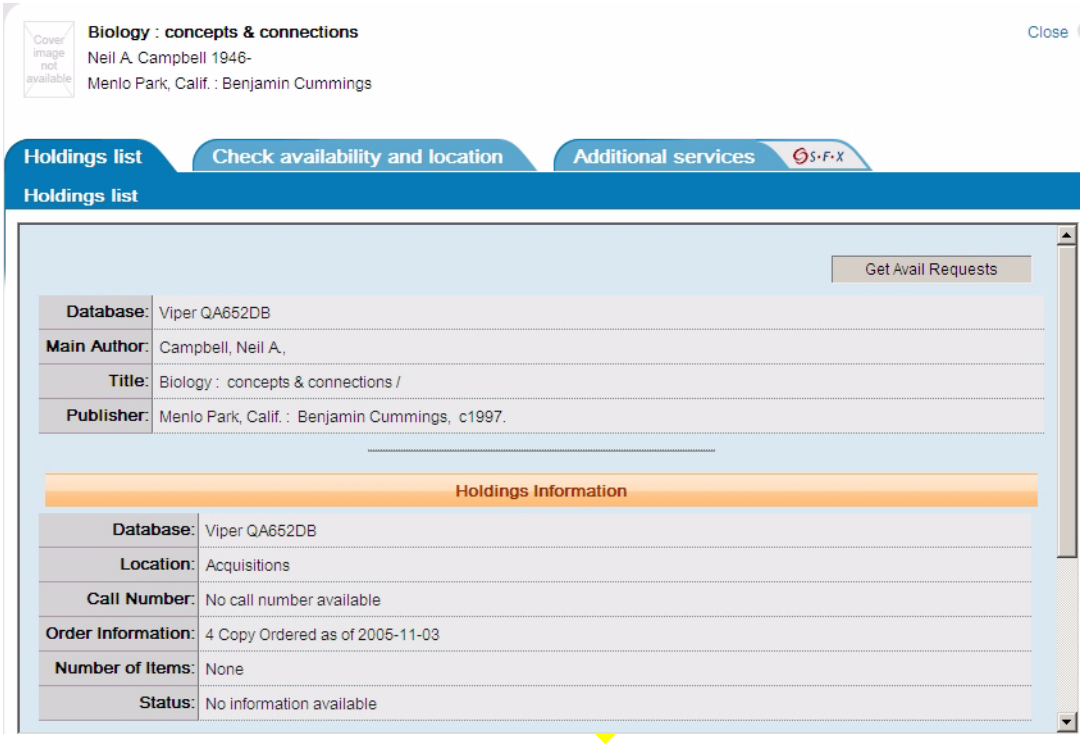

Figure 3-2. Get Avail Requests (prompting sign in)

FLD

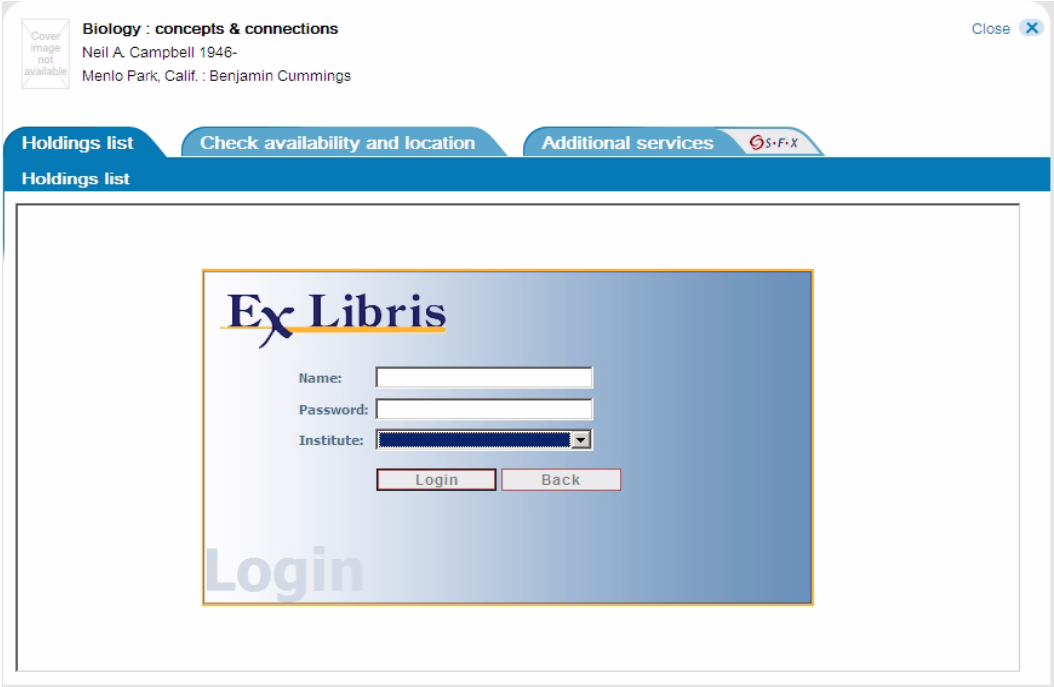

Figure 3-3. Login dialog box

Patron Information (Primo Library Card)

In the Voyager Primo environment, the Primo Library Card utilizes a deep link to Voyager that provides patrons with the following.

- Display of patron information (similar to the Patron Information Page in WebVoyáge™.
- Option to renew charged items.
- Option to cancel requests.

This functionality is accessed through the Library Card link in Primo.

Patron Requests (GetIt!)

When the patron identifies a record of interest in Primo (from the Voyager database), clicking the GetIt! link displays the **Holdings List** tab in Primo.

The **Holdings List** tab display contains the Voyager bibliographic, holdings, and item information. There is also the **Select a Request** drop-down list (see Figure 3-4) providing the patron with a number of request options if the patron has signed in.

If the patron has not signed in, **Get Avail Requests** displays. See PDS Invoked on page 3-2 for the sign in and PDS authentication steps.

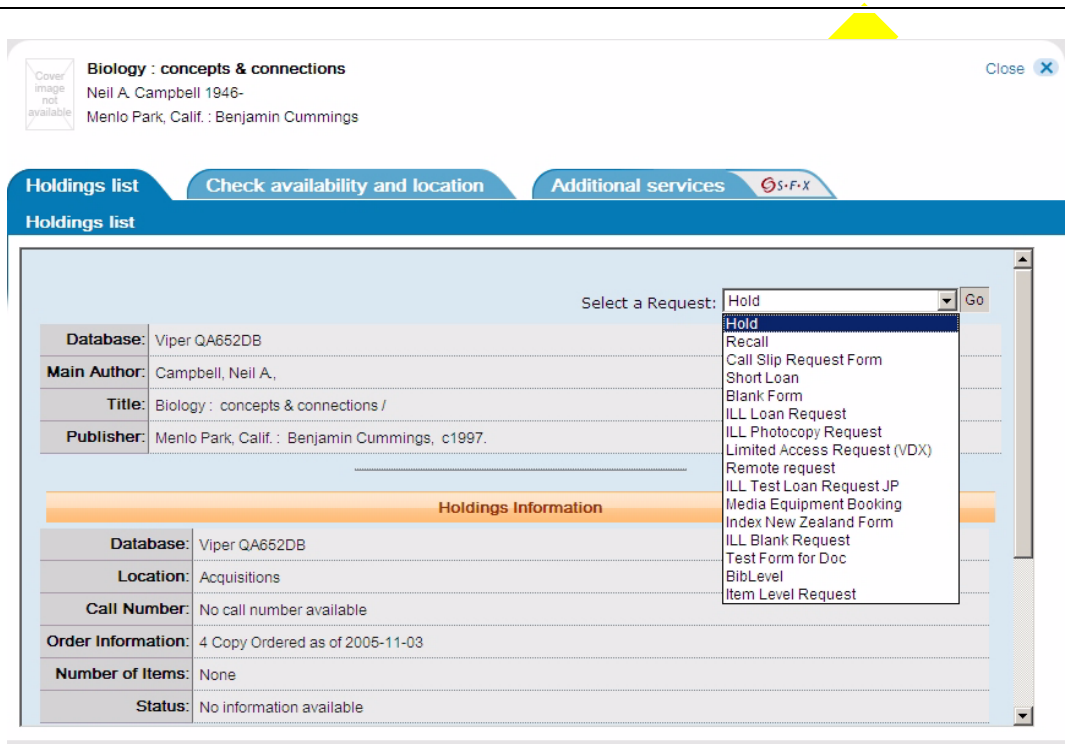

Figure 3-4. Select A Request options

The GetIt! link is configured by the Primo system administrator with the necessary linking information to the Voyager system. See the *Primo Administrator Guide* for details regarding how to configure GetIt!. This provides a deep link that enables the patron to confirm availability and process requests.

NOTE:
Requests in Voyager are only processed after PDS has processed patron authentication.

FLD TST

Index

A

about this document, ix
At Bindery=, 2-13
audience
    of this document, ix
AvailFilter=, 2-18

B

batch job
    Primoexp, 2-21
BibRangeBegin=, 2-16
BibRangeEnd=, 2-16
BibsFromFile=, 2-17

C

Call Slip Request=, 2-13
Cataloging Review=, 2-13
ChangedSince=, 2-17
Charged=, 2-13
checklist, 1-2
Circulation Review=, 2-13
Claims Returned=, 2-13
comments
    about this document, xii
configuration file
    PrimoExp.ini, 2-4
conventions used
    in this document, x

D

Damaged=, 2-13
data extraction, 2-1
Database=, 2-19
DatabaseHost=, 2-19
DatabasePort=, 2-19
DelBibFile=, 2-19
DelBibsDir=, 2-19
DelItemsDir=, 2-19
DelItemsFile=, 2-19
DelMFHDsDir=, 2-19
DelMFHDsFile=, 2-19
Discharged=, 2-13
document summary, x

E

[Enrichment Tags], 2-15
[Excluded Happening Locations], 2-14
[Excluded Operator Ids], 2-14

F

feedback, customer, xii
File=, 2-20

G

GetIt!, 3-4
Getting Started, 1-1
    prerequisite skills and knowledge, 1-1
getting started
    checklist, 1-2

FLD TST

H

Heading Changes=, 2-17
Hold Request=, 2-13

I

idPrefix=, 2-20
In Process=, 2-13
In Transit Discharged=, 2-13
In Transit On Hold=, 2-13
In Transit=, 2-13
initial data load, 2-2
InsertAUTTag, 2-15
InsertAVATag, 2-15
InsertSUBTag, 2-15
[Institution Id], 2-14
intended audience
  of this document, ix
[Item Statuses], 2-12
[Item Statuses] parameter
  At Bindery=, 2-13
  Call Slip Request=, 2-13
  Cataloging Review=, 2-13
  Charged=, 2-13
  Circulation Review=, 2-13
  Claims Returned=, 2-13
  Damaged=, 2-13
  Discharged=, 2-13
  Hold Request=, 2-13
  In Process=, 2-13
  In Transit Discharged=, 2-13
  In Transit On Hold=, 2-13
  In Transit=, 2-13
  Lost--Library Applied=, 2-13
  Lost--System Applied=, 2-13
  Missing=, 2-13
  Not Charged=, 2-13
  On Hold=, 2-13
  Overdue=, 2-13
  Recall Request=, 2-13
  Remote Storage Request=, 2-13
  Renewed=, 2-13
  Scheduled=, 2-13
  Short Loan Request=, 2-13
  Withdrawn=, 2-13

L

Library Card, 3-4
[Location to Primo], 2-14
LogFile=, 2-18
LogFileDir=, 2-18
LogLevel=, 2-18
LogToStdOut=, 2-18
Lost--Library Applied=, 2-13
Lost--System Applied=, 2-13

M

Missing=, 2-13

N

Not Charged=, 2-13

O

On Hold=, 2-13
Overdue=, 2-13

P

patron authentication, 3-1
Patron Directory Services (PDS), 3-1
patron information, 3-4
patron requests, 3-4
PDS, 3-1
photocopying
  documentation, xi
PNX, 2-4
Pprimoexp, 2-21
Primo database
  initial data load, 2-2

FIELD TEST

ongoing updates, 2-2
Primo Library Card, 3-4
Primo Normalized XML, 2-4
[PrimoExp], 2-16
[PrimoExp] parameter
AvailFilter=, 2-18
BibRangeBegin=, 2-16
BibRangeEnd=, 2-16
BibsFromFile=, 2-17
ChangedSince=, 2-17
Database=, 2-19
DatabaseHost=, 2-19
DatabasePort=, 2-19
DelBibFile=, 2-19
DelBibsDir=, 2-19
DelItemsDir=, 2-19
DelItemsFile=, 2-19
DelMFHDsDir=, 2-19
DelMFHDsFile=, 2-19
HeadingChanges=, 2-17
LogFile=, 2-18
LogFileDir=, 2-18
LogLevel=, 2-18
LogToStdOut=, 2-18
UserPass=, 2-19
PrimoExp.ini, 2-4
purpose
of this document, ix

R

Recall Request=, 2-13
recsPerGroup=, 2-20
reissue
reason for, x
Remote Storage Request=, 2-13
Renewed=, 2-13
reproduction, of documentation, xi

S

Scheduled=, 2-13
scheduling, 2-21
Short Loan Request=, 2-13
stanza
[Enrichment Tags], 2-15
[Excluded Happening Locations], 2-14
[Excluded Operator Ids], 2-14
[Institution Id], 2-14
[Item Statuses], 2-12
[Location to Primo], 2-14
[PrimoExp], 2-16
[XMLWriterProtocol], 2-20

T

tab_service.<institute>, 3-2

U

UserPass=, 2-19

W

Withdrawn=, 2-13

X

[XMLWriterProtocol], 2-20
[XMLWriterProtocol] parameter
File=, 2-20
idPrefix=, 2-20
recsPerGroup=, 2-20
XSL=, 2-20
XSL=, 2-20

FLD TST