Voyager®
Primo® Integration User's Guide

September 2007

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Unpublished and Not for Publication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 - Excluded Records 2-2
  - Ongoing Updates to Primo Database 2-3
    - Pipes 2-3
  - Special Purpose/Selective Database Extractions 2-4
  - Primo Transaction 2-4
- Configuration Files 2-5
  - Availability Configuration File 2-6

Contents

Publishing Configuration File 2-13
Selective Configuration File 2-21
- Stanza Definitions for Voyager Primo Integration Configuration Files 2-29

[Item Statuses] 2-29
[Excluded Happening Locations] 2-31
[Excluded Operator Ids] 2-31
[Institution Id] 2-31
[Location to Primo] 2-31
[Enrichment Tags] 2-32
[PrimoExp] 2-33
[XMLWriterProtocol] 2-37

- Pprimoexp Batch Job 2-38

Scheduling 2-38

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

Figures

2 Data Extraction

2-1. PrimoExp-Availability.ini Example 2-6
2-2. PrimoExp-Publishing.ini Example 2-13
2-3. PrimoExp-Selective.ini Example 2-21

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

Figures

Tables

2 Data Extraction

2-1. [Item Statuses] parameters 2-30
2-2. [PrimoExp] parameter/command descriptions 2-33
2-3. [XMLWriterProtocol] parameter descriptions 2-37

Tables

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/modify configuration files.
- Run batch files.
- 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.

- Configuration file setup.
- Initial upload of Voyager data into Primo.
- Writing scripts and scheduling batch jobs for running updates.
- User authentication setup.
- Review/analysis of message logs as needed.

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.

- 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@exlibris-usa.com`

Getting Started

1

Contents

Introduction **1-1**

Purpose of this Chapter **1-1**

Prerequisite Skills and Knowledge **1-1**

Before You Begin **1-2**

- Checklist 1-2

Contents

Getting Started

1

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.

- Configuration files for data extraction.
- Batch files to run jobs.
- 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.
- UNIX® operating system commands and file system.
- 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.

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 integration feature component

Checklist

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

- Identify the elements needed to customize configuration files.
- 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.

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 - Excluded Records 2-2
- Ongoing Updates to Primo Database 2-3
  - Pipes 2-3
- Special Purpose/Selective Database Extractions 2-4
- Primo Transaction / Record Normalization 2-4

Configuration Files **2-5**

- Availability Configuration File 2-6
- Publishing Configuration File 2-13
- Selective Configuration File 2-21

Stanza Definitions for Voyager Primo Integration Configuration Files **2-29**

- [Item Statuses] 2-29
- [Excluded Happening Locations] 2-31
- [Excluded Operator Ids] 2-31
- [Institution Id] 2-31
- [Location to Primo] 2-31
- [Enrichment Tags] 2-32
- [PrimoExp] 2-33
- [XMLWriterProtocol] 2-37

Pprimoexp Batch Job **2-38**

- Scheduling 2-38

Contents

Data Extraction

2

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 components needed for extracting Voyager data for Primo to harvest.

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 configuration file to your environment/needs.

  Use the `PrimoExp-Publishing.ini` as a starting point. See Publishing Configuration File on page 2-13.

- Run the `Pprimoexp` batch job.

  Using the parameters set in the configuration file, this step identifies the Voyager records to be uploaded to the Primo database.

  A separate script, `PbundlePrimo`, wraps the Voyager records in a `tar.gz` file for Primo to access. The `tar.gz` files are located in `/m1/voyager/yyydb/primo/Publishing/bundled` where `yyydb` is the database name.

- 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 - Excluded Records

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 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 configuration file, a MARCXML record is not generated.

See Stanza Definitions for Voyager Primo Integration Configuration Files on page 2-29 for more information about customizing the configuration file.

Ongoing Updates to Primo Database

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

- 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.

Pipes

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, separate and customized versions of configuration files are provided to you for publishing and availability purposes as follows.

- `PrimoExp-Publishing.ini`. See Publishing Configuration File on page 2-13 for more information.
- `PrimoExp-Availability.ini`. See Availability Configuration File on page 2-6 for more information.

These are located in `/m1/voyager/yyydb/ini` where `yyydb` is the database name.

These configuration files are the same except for the parameter setting differences required for each purpose. See Stanza Definitions for Voyager Primo Integration Configuration Files on page 2-29 for more information about the parameters that affect which records are generated for harvesting.

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 configuration files.

Corresponding copies of `Pprimoexp` batch files for publishing and availability purposes can be created in order to automate the harvesting process using a cron. See Pprimoexp Batch Job on page 2-38 for more information about `Pprimoexp` batch processing.

Having separate functional configuration files, batch files, and directories for generated files enables your flexibility in scheduling production processing of data changes in the Voyager Primo integrated environment. This allows you to schedule Primo 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/Selective 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`.

The `PrimoExp-Selective.ini` configuration file is provided as one method for creating a testing subset. It is located in `/m1/voyager/yyydb/ini` where `yyydb` is the database name.

For Selective record processing and storing `tar.gz` files, the following directory is provided where `yyydb` is the database name.

- `/m1/voyager/yyydb/primo/Selective/bundled`.

  Insure that Primo has access to this directory.

Primo Transaction / Record Normalization

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.

Configuration Files

Primo uses data extracted from the Voyager database. Configuration files are used to define what is extracted. The following customized configuration files are provided with your installation.

- `PrimoExp-Availability.ini`. See Availability Configuration File on page 2-6.
- `PrimoExp-Publishing.ini`. See Publishing Configuration File on page 2-13.
- `PrimoExp-Selective.ini`. Selective Configuration File on page 2-21.

They are located in `/m1/voyager/yyydb/ini` where `yyydb` is the database name.

These configuration files are provided as examples for publishing, availability, and selective requirements with parameters set to what many customers use. However, copies of the configuration files may be made and tailored to meet your unique needs.

The configuration files define 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.
- 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 2-31.
- Institution ID to be included in the availability tag.
- Location definitions for consistency within the Primo database. See [Location to Primo] on page 2-31.
- Tags to identify unique field usage within your institution's database specific to the database extract. See [Enrichment Tags] on page 2-32.
- Extract processing considerations such as date/time, record ID(s), filters, and so on. See [PrimoExp] on page 2-33.
- Additional processing considerations for uploading the Voyager records identified by the configuration file processing. See [XMLWriterProtocol] on page 2-37.

See Stanza Definitions for Voyager Primo Integration Configuration Files on page 2-29 for a complete description of configuration file options.

Availability Configuration File

The following is the customized availability configuration file provided at installation.

```
Line#
  1   # Configuration for Voyager Export for Primo Availability
  2
  3   [Item Statuses]
  4   # Voyager status maps to (A)vailable or (U)navailable for
          Primo
  5   Not Charged=A
  6   Charged=U
  7   Renewed=U
  8   Overdue=U
  9   Recall Request=U
 10   Hold Request=U
 11   On Hold=U
 12   In Transit=U
 13   In Transit Discharged=U
 14   In Transit On Hold=U
 15   Discharged=A
 16   Missing=U
 17   Lost--Library Applied=U
 18   Lost--System Applied=U
 19   Claims Returned=U
 20   Damaged=U
 21   Withdrawn=U
 22   At Bindery=U
 23   Cataloging Review=A
 24   Circulation Review=A
 25   Scheduled=U
 26   In Process=U
```

Figure 2-1. PrimoExp-Availability.ini Example

```
Line#
  27  Call Slip Request=U
  28  Short Loan Request=U
  29  Remote Storage Request=U
  30
  31  [Excluded Happening Locations]
  32  # Items whose most recent activity took place in one of
          these locations will not be exported.
  33
  34  [Excluded Operator Ids]
  35  # Items whose most recent activity was recorded by one of
          these operators will not be exported.
  36
  37  [Institution Id]
  38  # This value will be included in subfield a of each
          Availability field
  39  Id=@DBNAME@
  40
  41  [Location to Primo]
  42  # This section maps Voyager locations to Primo locations.
  43
  44
  45  [Enrichment Tags]
  46  # tags added to marc record with additional data
  47  # AVA tag contains the availability information
  48  # AUT tag contains name cross-reference information
  49  # SUB tag contains subject cross-reference information
  50  InsertAVATag=949
  51  InsertAUTTag=950
  52  InsertSUBTag=951
  53
  54
  55  [PrimoExp]
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
  56    # these parameters provide defaults for this run of the
                job
  57    # each of them can be overridden using a command-line
                parameter
  58
  59    # note that stanza.name can be used with -- on the command
                line
  60    # argv(case insensitive) override
  61    # ini(case insensitive) override
  62    # env (all uppercase)
  63
  64    # BibRangeBegin and BibRangeEnd are the beginning and
                ending bibliographic ID numbers to export.
  65    # -B overrides this parameter on the command line
  66    # BibRangeBegin
  67    #BibRangeBegin=1
  68    # -E overrides this parameter on the command line
  69    #BibRangeEnd=200
  70    #
  71
  72    # BibsFromFile reads the file at the supplied path for a
                list of bibliographic record ids to export.
  73    # The file should contain one bibliographic record ID per
                line.
  74    # -F overrides this parameter on the command line
  75    #BibsFromFile=@VOYAGER@/@DATABASE@/local/PrimoBibs.txt
  76    #
  77
  78    # ChangedSince exports records changed since the supplied
                date.
  79    # ChangedSince may be either
  80    #          a timestamp in YYYYMMDD.HHMMSS format
  81    #          or a path to a file containing a timestamp in
                YYYYMMDD.HHMMSS format on a single line
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
 82  # If a path to a file is supplied, the current run time
         will be saved in the file when
 83  # the job finishes.
 84
 85  # -C overrides this parameter on the command line
 86  # changes since a date or date & time
 87  #ChangedSince=YYYYMMDD.HHMMSS
 88  # or
 89  # changes since the last (saved) run
 90  # current run time will be saved
 91  ChangedSince=@VOYAGER@/@DATABASE@/primo/Availability/
         ChangedSince.txt
 92  #
 93  # HeadingChanges determines whether to export
         bibliographic records whose cross-references have
         changed
 94  # -H Y|N overrides this parameter on the command line
 95  HeadingChanges=N
 96  #
 97  # AvailFilter detemines whether to export records whose
         availability fields have not changed since the
         last time
 98  # the job ran. If AvailFilter is Y, only records whose
         availability
 99  # has changed are exported; if AvailFilter is N, all
         changed records are exported.
100  # Whether to filter records to prevent sending records
         that have
101  # had no change in their availability tags.  Defaults to
         N.
102  # -A Y|N overrides this parameter on the command line
103  AvailFilter=Y
104
105  # Logfile is the location of the logfile.
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
 106  # Note that $str$ will be passed to java's Date.Format()
          function to control the datestamps embedded in
          the file name.
 107
 108  # -L overrides this parameter on the command line
 109  LogFile=@VOYAGER@/@DATABASE@/primo/Availability/logs/
          primo.export.$yyMMdd$.log
 110  #LogFile=/dev/null
 111
 112  # -v
 113  # the logging level
 114  LogLevel=10
 115
 116  # LogToStdOut determines whether or not to write the log
          output to standard output, as well as the log
          file.
 117  # whether to duplicate log output to stdout
 118  LogToStdOut=N
 119
 120  # Database and the other Database parameters determine
          which database to connect to.
 121  # Ordinarily these are set by the wrapper script based on
          the environment.
 122  # -d
 123  # DataBase=
 124  #
 125  # DatabaseHost
 126  DatabaseHost=@DBHOST@
 127  #DatabaseHost=208.178.237.40
 128  # DatabasePort
 129  # -e
 130  #UserPass=@DBUSER@/@DBPASS@
 131  Database=VGER
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
 132  Protocol=com.endinfosys.voyager.extract.OAIPMHXmlWriterPr
          otocol
 133
 134  # must do items, then mfhds, then bibs
 135  Task=com.endinfosys.voyager.extract.ExtractItemsPrimo
 136  Task=com.endinfosys.voyager.extract.ExtractMFHDsPrimo
 137  Task=com.endinfosys.voyager.extract.ExtractBibsPrimo
 138  DoItems=Y
 139  DoMfhds=Y
 140
 141  # The Del* Dir and File are used to determine when records
          are deleted.
 142  # These are standard Voyager locations.
 143  DelBibsDir=@VOYAGER@/@DATABASE@/rpt
 144  DelBibFile=deleted.bib.marc
 145  DelMFHDsDir=@VOYAGER@/@DATABASE@/rpt
 146  DelMFHDsFile=deleted.mfhd.marc
 147  DelItemsDir=@VOYAGER@/@DATABASE@/rpt
 148  DelItemsFile=delete.item
 149
 150  [XMLWriterProtocol]
 151  # OAIPMHXmlWriterProtocol will include the bib id
 152  #  as the last element in the file name before
 153  #  the .xml extension.
 154  # Note the path must be coordinated with the
          bundlePrimo.ksh
 155  #  script execution.
 156  # Note that $str$ will be passed to java's Date.Format()
          function to control the datestamps embedded in
          the file name.
 157
 158  File=@VOYAGER@/@DATABASE@/primo/Availability/exports/
          primo.export.$yyMMddhhmmss$.xml
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
 159  XSL=
 160  # max number of records to group in a single tar file
 161  #  all in single group if 0 or undefined
 162  recsPerGroup=1000
 163  # prefix for record identifiers, must be present, may be
          empty
 164  idPrefix=
 165
 166  #
 167  # EVERYTHING BELOW THIS LINE IS INTERNAL DO NOT MODIFY
 168  #
 169
 170  [OAIPMHXmlWriterProtocol]
 171  BibUpdate=com.endinfosys.voyager.extract.WriterTrans
 172  BibUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 173  BibUpdate=com.endinfosys.voyager.primo.SummarizeItemsTran
          s
 174
 175  BibDelete=com.endinfosys.voyager.extract.WriterTrans
 176  BibDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 177  BibDelete=com.endinfosys.voyager.primo.SummarizeItemsTran
          s
 178
 179  ItemUpdate=com.endinfosys.voyager.extract.WriterTrans
 180  ItemUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 181  ItemUpdate=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 182
 183  ItemDelete=com.endinfosys.voyager.extract.WriterTrans
 184  ItemDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 185  ItemDelete=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 186
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

```
Line#
  187  MFHDUpdate=com.endinfosys.voyager.extract.WriterTrans
  188  MFHDUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
  189  MFHDUpdate=com.endinfosys.voyager.primo.SummarizeItemsTra
           ns
  190
  191  MFHDDelete=com.endinfosys.voyager.extract.WriterTrans
  192  MFHDDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
  193  MFHDDelete=com.endinfosys.voyager.primo.SummarizeItemsTra
           ns
  194
  195  [TransUpdateRec]
  196  FlagUpdate=ASSIGN,action,m
  197
  198  FlagDelete=ASSIGN,action,d
  199  FlagDelete=ASSIGN,bib.000:5,d
  200  FlagDelete=MOVE,bib.035:a,bib_id
  201
```

Figure 2-1. PrimoExp-Availability.ini Example (Continued)

Publishing Configuration File

The following is the customized publishing configuration file provided at installation.

```
Line#
    1  # Configuration for Voyager Export for Primo Publishing
    2
    3  [Item Statuses]
    4  # Voyager status maps to (A)vailable or (U)navailable for
           Primo
    5  Not Charged=A
```

Figure 2-2. PrimoExp-Publishing.ini Example

```
Line#
  6     Charged=U
  7     Renewed=U
  8     Overdue=U
  9     Recall Request=U
  10    Hold Request=U
  11    On Hold=U
  12    In Transit=U
  13    In Transit Discharged=U
  14    In Transit On Hold=U
  15    Discharged=A
  16    Missing=U
  17    Lost--Library Applied=U
  18    Lost--System Applied=U
  19    Claims Returned=U
  20    Damaged=U
  21    Withdrawn=U
  22    At Bindery=U
  23    Cataloging Review=A
  24    Circulation Review=A
  25    Scheduled=U
  26    In Process=U
  27    Call Slip Request=U
  28    Short Loan Request=U
  29    Remote Storage Request=U
  30
  31    [Excluded Happening Locations]
  32    # Items whose most recent activity took place in one of
              these locations will not be exported.
  33
  34    [Excluded Operator Ids]
  35    # Items whose most recent activity was recorded by one of
              these operators will not be exported.
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
 36
 37  [Institution Id]
 38  # This value will be included in subfield a of each
         Availability field
 39  Id=@DBNAME@
 40
 41  [Location to Primo]
 42  # This section maps Voyager locations to Primo locations.
 43
 44
 45  [Enrichment Tags]
 46  # tags added to marc record with additional data
 47  # AVA tag contains the availability information
 48  # AUT tag contains name cross-reference information
 49  # SUB tag contains subject cross-reference information
 50  InsertAVATag=949
 51  InsertAUTTag=950
 52  InsertSUBTag=951
 53
 54
 55  [PrimoExp]
 56  # these parameters provide defaults for this run of the
         job
 57  # each of them can be overridden using a command-line
         parameter
 58
 59  # note that stanza.name can be used with -- on the command
         line
 60  # argv(case insensitive) override
 61  # ini(case insensitive) override
 62  # env (all uppercase)
 63
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
 64  # BibRangeBegin and BibRangeEnd are the beginning and
         ending bibliographic ID numbers to export.
 65  # -B overrides this parameter on the command line
 66  #BibRangeBegin=1
 67  # -E overrides this parameter on the command line
 68  #BibRangeEnd=200
 69  #
 70
 71  # BibsFromFile reads the file at the supplied path for a
         list of bibliographic record ids to export.
 72  # The file should contain one bibliographic record ID per
         line.
 73  # -F overrides this parameter on the command line
 74  #BibsFromFile=@VOYAGER@/@DATABASE@/local/PrimoBibs.txt
 75  #
 76
 77  # ChangedSince exports records changed since the supplied
         date.
 78  # ChangedSince may be either
 79  #          a timestamp in YYYYMMDD.HHMMSS format
 80  #          or a path to a file containing a timestamp in
         YYYYMMDD.HHMMSS format on a single line
 81  # If a path to a file is supplied, the current run time
         will be saved in the file when
 82  # the job finishes.
 83
 84  # -C overrides this parameter on the command line
 85  # changes since a date or date & time
 86  #ChangedSince=YYYYMMDD.HHMMSS
 87  # or
 88  # changes since the last (saved) run
 89  # current run time will be saved
 90  ChangedSince=@VOYAGER@/@DATABASE@/primo/Publishing/
         ChangedSince.txt
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
  91  #
  92  # HeadingChanges determines whether to export
          bibliographic records whose cross-references have
          changed
  93  # -H Y|N overrides this parameter on the command line
  94  HeadingChanges=Y
  95  #
  96  # AvailFilter detemines whether to export records whose
          availability fields have not changed since the
          last time
  97  # the job ran. If AvailFilter is Y, only records whose
          availability
  98  # has changed are exported; if AvailFilter is N, all
          changed records are exported.
  99  # Whether to filter records to prevent sending records
          that have
 100  # had no change in their availability tags.  Defaults to N.
 101  # -A Y|N overrides this parameter on the command line
 102  AvailFilter=N
 103
 104  # Logfile is the location of the logfile.
 105  # Note that $str$ will be passed to java's Date.Format()
          function to control the datestamps embedded in the
          file name.
 106
 107  # -L overrides this parameter on the command line
 108  LogFile=@VOYAGER@/@DATABASE@/primo/Publishing/logs/
          primo.export.$yyMMdd$.log
 109  #LogFile=/dev/null
 110
 111  # -v
 112  # the logging level
 113  LogLevel=10
 114
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
 115  # LogToStdOut determines whether or not to write the log
          output to standard output, as well as the log
          file.
 116  # whether to duplicate log output to stdout
 117  LogToStdOut=N
 118
 119  # Database and the other Database parameters determine
          which database to connect to.
 120  # Ordinarily these are set by the wrapper script based on
          the environment.
 121  # -d
 122  # DataBase=
 123  #
 124  # DatabaseHost
 125  DatabaseHost=@DBHOST@
 126  #DatabaseHost=208.178.237.40
 127  # DatabasePort
 128  # -e
 129  #UserPass=@DBUSER@/@DBPASS@
 130  Database=VGER
 131  Protocol=com.endinfosys.voyager.extract.OAIPMHXmlWriterPr
          otocol
 132
 133  # must do items, then mfhds, then bibs
 134  Task=com.endinfosys.voyager.extract.ExtractItemsPrimo
 135  Task=com.endinfosys.voyager.extract.ExtractMFHDsPrimo
 136  Task=com.endinfosys.voyager.extract.ExtractBibsPrimo
 137  DoItems=Y
 138  DoMfhds=Y
 139
 140  # The Del* Dir and File are used to determine when records
          are deleted.
 141  # These are standard Voyager locations.
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
  142  DelBibsDir=@VOYAGER@/@DATABASE@/rpt
  143  DelBibFile=deleted.bib.marc
  144  DelMFHDsDir=@VOYAGER@/@DATABASE@/rpt
  145  DelMFHDsFile=deleted.mfhd.marc
  146  DelItemsDir=@VOYAGER@/@DATABASE@/rpt
  147  DelItemsFile=delete.item
  148
  149  [XMLWriterProtocol]
  150  # OAIPMHXmlWriterProtocol will include the bib id
  151  #  as the last element in the file name before
  152  #  the .xml extension.
  153  # Note the path must be coordinated with the
           bundlePrimo.ksh
  154  #  script execution.
  155  # Note that $str$ will be passed to java's Date.Format()
           function to control the datestamps embedded in the
           file name.
  156
  157  File=@VOYAGER@/@DATABASE@/primo/Publishing/exports/
           primo.export.$yyMMddhhmmss$.xml
  158  XSL=
  159  # max number of records to group in a single tar file
  160  #  all in single group if 0 or undefined
  161  recsPerGroup=1000
  162  # prefix for record identifiers, must be present, may be
           empty
  163  idPrefix=
  164
  165  #
  166  # EVERYTHING BELOW THIS LINE IS INTERNAL DO NOT MODIFY
  167  #
  168
  169  [OAIPMHXmlWriterProtocol]
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
 170  BibUpdate=com.endinfosys.voyager.extract.WriterTrans
 171  BibUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 172  BibUpdate=com.endinfosys.voyager.primo.SummarizeItemsTran
          s
 173
 174  BibDelete=com.endinfosys.voyager.extract.WriterTrans
 175  BibDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 176  BibDelete=com.endinfosys.voyager.primo.SummarizeItemsTran
          s
 177
 178  ItemUpdate=com.endinfosys.voyager.extract.WriterTrans
 179  ItemUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 180  ItemUpdate=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 181
 182  ItemDelete=com.endinfosys.voyager.extract.WriterTrans
 183  ItemDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 184  ItemDelete=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 185
 186  MFHDUpdate=com.endinfosys.voyager.extract.WriterTrans
 187  MFHDUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 188  MFHDUpdate=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 189
 190  MFHDDelete=com.endinfosys.voyager.extract.WriterTrans
 191  MFHDDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 192  MFHDDelete=com.endinfosys.voyager.primo.SummarizeItemsTra
          ns
 193
 194  [TransUpdateRec]
 195  FlagUpdate=ASSIGN,action,m
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

```
Line#
 196
 197  FlagDelete=ASSIGN,action,d
 198  FlagDelete=ASSIGN,bib.000:5,d
 199  FlagDelete=MOVE,bib.035:a,bib_id
 200
```

Figure 2-2. PrimoExp-Publishing.ini Example (Continued)

Selective Configuration File

The following is the customized selective configuration file provided at installation.

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

Figure 2-3. PrimoExp-Selective.ini Example

```
Line#
 18  Lost--System Applied=U
 19  Claims Returned=U
 20  Damaged=U
 21  Withdrawn=U
 22  At Bindery=U
 23  Cataloging Review=A
 24  Circulation Review=A
 25  Scheduled=U
 26  In Process=U
 27  Call Slip Request=U
 28  Short Loan Request=U
 29  Remote Storage Request=U
 30
 31  [Excluded Happening Locations]
 32  # Items whose most recent activity took place in one of
         these locations will not be exported.
 33
 34  [Excluded Operator Ids]
 35  # Items whose most recent activity was recorded by one
         of these operators will not be exported.
 36
 37  [Institution Id]
 38  # This value will be included in subfield a of each
         Availability field
 39  Id=@DBNAME@
 40
 41  [Location to Primo]
 42  # This section maps Voyager locations to Primo
         locations.
 43
 44
 45  [Enrichment Tags]
 46  # tags added to marc record with additional data
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 47   # AVA tag contains the availability information
 48   # AUT tag contains name cross-reference information
 49   # SUB tag contains subject cross-reference information
 50   InsertAVATag=949
 51   InsertAUTTag=950
 52   InsertSUBTag=951
 53
 54
 55   [PrimoExp]
 56   # these parameters provide defaults for this run of the
          job
 57   # each of them can be overridden using a command-line
          parameter
 58
 59   # note that stanza.name can be used with -- on the
          command line
 60   # argv(case insensitive) override
 61   # ini(case insensitive) override
 62   # env (all uppercase)
 63
 64   # BibRangeBegin and BibRangeEnd are the beginning and
          ending bibliographic ID numbers to export.
 65   # -B overrides this parameter on the command line
 66   #BibRangeBegin=1
 67   # -E overrides this parameter on the command line
 68   #BibRangeEnd=200
 69   #
 70
 71   # BibsFromFile reads the file at the supplied path for a
          list of bibliographic record ids to export.
 72   # The file should contain one bibliographic record ID
          per line.
 73   # -F overrides this parameter on the command line
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 74  BibsFromFile=@VOYAGER@/@DATABASE@/local/PrimoBibs.txt
 75  #
 76
 77  # ChangedSince exports records changed since the
         supplied date.
 78  # ChangedSince may be either
 79  #          a timestamp in YYYYMMDD.HHMMSS format
 80  #          or a path to a file containing a timestamp in
         YYYYMMDD.HHMMSS format on a single line
 81  # If a path to a file is supplied, the current run time
         will be saved in the file when
 82  # the job finishes.
 83
 84  # -C overrides this parameter on the command line
 85  # changes since a date or date & time
 86  #ChangedSince=YYYYMMDD.HHMMSS
 87  # or
 88  # changes since the last (saved) run
 89  # current run time will be saved
 90  #ChangedSince=@VOYAGER@/@DATABASE@/primo/Selective/
         ChangedSince.txt
 91  #
 92  # HeadingChanges determines whether to export
         bibliographic records whose cross-references
         have changed
 93  # -H Y|N overrides this parameter on the command line
 94  HeadingChanges=N
 95  #
 96  # AvailFilter detemines whether to export records whose
         availability fields have not changed since the
         last time
 97  # the job ran. If AvailFilter is Y, only records whose
         availability
 98  # has changed are exported; if AvailFilter is N, all
         changed records are exported.
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 99  # Whether to filter records to prevent sending records
         that have
100  # had no change in their availability tags.  Defaults to
         N.
101  # -A Y|N overrides this parameter on the command line
102  AvailFilter=N
103  #
104
105  # Logfile is the location of the logfile.
106  # Note that $str$ will be passed to java's Date.Format()
         function to control the datestamps embedded in
         the file name.
107
108  # -L overrides this parameter on the command line
109  LogFile=@VOYAGER@/@DATABASE@/primo/Selective/logs/
         primo.export.$yyMMdd$.log
110  #LogFile=/dev/null
111
112  # -v
113  # the logging level
114  LogLevel=10
115
116  # LogToStdOut determines whether or not to write the log
         output to standard output, as well as the log
         file.
117  # whether to duplicate log output to stdout
118  LogToStdOut=N
119
120  # Database and the other Database parameters determine
         which database to connect to.
121  # Ordinarily these are set by the wrapper script based
         on the environment.
122  # -d
123  # DataBase=
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 124 #
 125 # DatabaseHost
 126 DatabaseHost=@DBHOST@
 127 #DatabaseHost=208.178.237.40
 128 # DatabasePort
 129 # -e
 130 #UserPass=@DBUSER@/@DBPASS@
 131 Database=VGER
 132 Protocol=com.endinfosys.voyager.extract.OAIPMHXmlWriter
         Protocol
 133
 134 # must do items, then mfhds, then bibs
 135 Task=com.endinfosys.voyager.extract.ExtractItemsPrimo
 136 Task=com.endinfosys.voyager.extract.ExtractMFHDsPrimo
 137 Task=com.endinfosys.voyager.extract.ExtractBibsPrimo
 138 DoItems=Y
 139 DoMfhds=Y
 140
 141 # The Del* Dir and File are used to determine when
         records are deleted.
 142 # These are standard Voyager locations.
 143 DelBibsDir=@VOYAGER@/@DATABASE@/rpt
 144 DelBibFile=deleted.bib.marc
 145 DelMFHDsDir=@VOYAGER@/@DATABASE@/rpt
 146 DelMFHDsFile=deleted.mfhd.marc
 147 DelItemsDir=@VOYAGER@/@DATABASE@/rpt
 148 DelItemsFile=delete.item
 149
 150 [XMLWriterProtocol]
 151 # OAIPMHXmlWriterProtocol will include the bib id
 152 #  as the last element in the file name before
 153 #  the .xml extension.
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 154  # Note the path must be coordinated with the
          bundlePrimo.ksh
 155  #  script execution.
 156  # Note that $str$ will be passed to java's Date.Format()
          function to control the datestamps embedded in
          the file name.
 157
 158  File=@VOYAGER@/@DATABASE@/primo/Selective/exports/
          primo.export.$yyMMddhhmmss$.xml
 159  XSL=
 160  # max number of records to group in a single tar file
 161  #  all in single group if 0 or undefined
 162  recsPerGroup=1000
 163  # prefix for record identifiers, must be present, may be
          empty
 164  idPrefix=
 165
 166  #
 167  # EVERYTHING BELOW THIS LINE IS INTERNAL DO NOT MODIFY
 168  #
 169
 170  [OAIPMHXmlWriterProtocol]
 171  BibUpdate=com.endinfosys.voyager.extract.WriterTrans
 172  BibUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdate
 173  BibUpdate=com.endinfosys.voyager.primo.SummarizeItemsTr
          ans
 174
 175  BibDelete=com.endinfosys.voyager.extract.WriterTrans
 176  BibDelete=com.endinfosys.util.TransUpdateRec:FlagDelete
 177  BibDelete=com.endinfosys.voyager.primo.SummarizeItemsTr
          ans
 178
 179  ItemUpdate=com.endinfosys.voyager.extract.WriterTrans
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

```
Line#
 180  ItemUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdat
          e
 181  ItemUpdate=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 182
 183  ItemDelete=com.endinfosys.voyager.extract.WriterTrans
 184  ItemDelete=com.endinfosys.util.TransUpdateRec:FlagDelet
          e
 185  ItemDelete=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 186
 187  MFHDUpdate=com.endinfosys.voyager.extract.WriterTrans
 188  MFHDUpdate=com.endinfosys.util.TransUpdateRec:FlagUpdat
          e
 189  MFHDUpdate=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 190
 191  MFHDDelete=com.endinfosys.voyager.extract.WriterTrans
 192  MFHDDelete=com.endinfosys.util.TransUpdateRec:FlagDelet
          e
 193  MFHDDelete=com.endinfosys.voyager.primo.SummarizeItemsT
          rans
 194
 195  [TransUpdateRec]
 196  FlagUpdate=ASSIGN,action,m
 197
 198  FlagDelete=ASSIGN,action,d
 199  FlagDelete=ASSIGN,bib.000:5,d
 200  FlagDelete=MOVE,bib.035:a,bib_id
 201
```

Figure 2-3. PrimoExp-Selective.ini Example (Continued)

Stanza Definitions for Voyager Primo Integration Configuration Files

This section defines the configuration file stanzas that need to be customized for Voyager Primo integration.

The following is a list of the stanzas.

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

[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.

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

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

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.

```
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.

[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 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.

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.*
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.

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

Parameter/Command	Description
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 configuration file is used to harvest records for the Primo publishing pipe. This parameter should be set to N when the configuration 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.*

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 configuration file is used to harvest records for the Primo availability pipe. This parameter should be set to `N` when the configuration 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 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.

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 configuration file to match your Voyager configuration.
DelBibFile=	Leave this parameter setting as specified in the configuration file.
DelMFHDsDir=	Edit the parameter provided in the configuration file to match your Voyager configuration.
DelMFHDsFile=	Leave this parameter setting as specified in the configuration file.
DelItemsDir=	Edit the parameter provided in the configuration file to match your Voyager configuration.
DelItemsFile=	Leave this parameter setting as specified in the configuration file.

[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 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.

Pprimoexp Batch Job

The `Pprimoexp` 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.

The `Pprimoexp` batch job is located in `/m1/voyager/yyydb/sbin` where `yyydb` is the database name.

`Pprimoexp` runs the `PbundlePrimo` script that handles, for example, bundling individual records into `tar.gz` files and cleans up the exports directory.

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 configuration 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*.

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**

Contents

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 provides the following PDS authentication functions against the Voyager patron database.

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

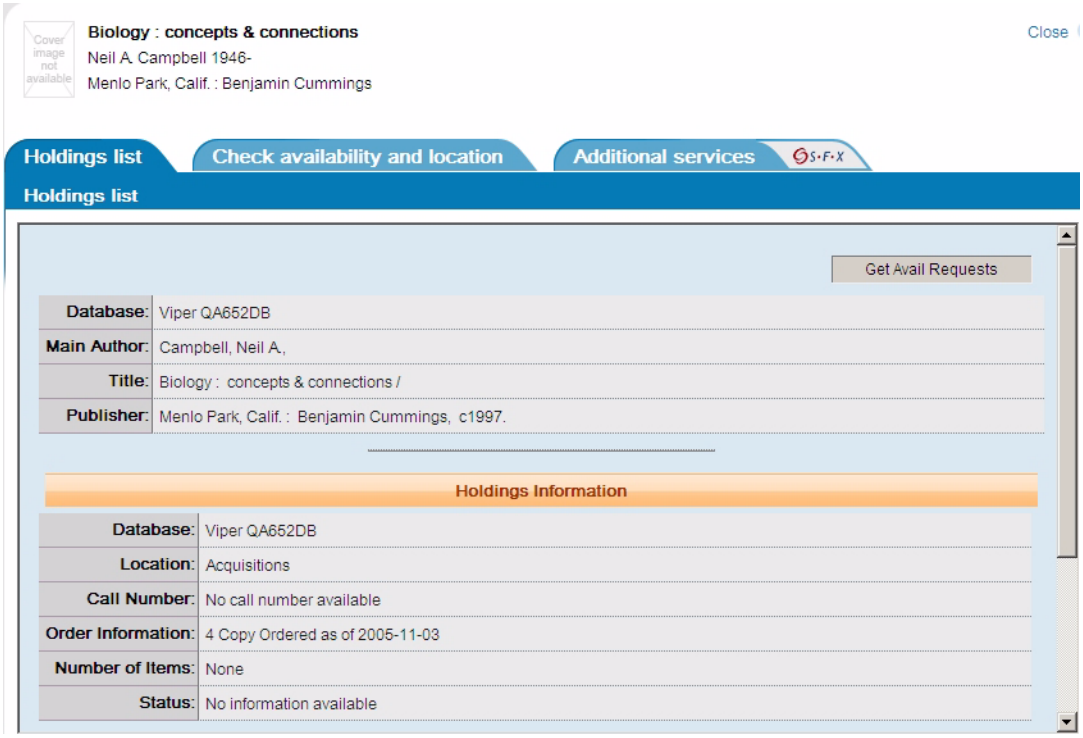

Figure 3-2. Get Avail Requests (prompting sign in)

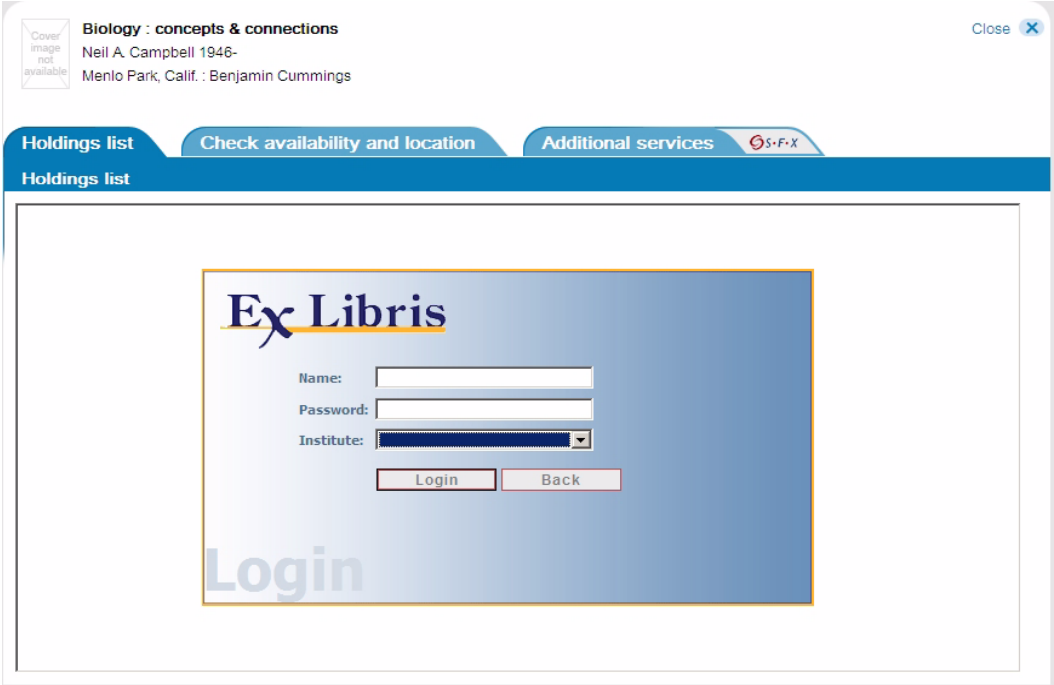

Figure 3-3. Login dialog box

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.
- Option to renew charged items.
- Option to cancel requests.

This functionality is accessed through the My 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.

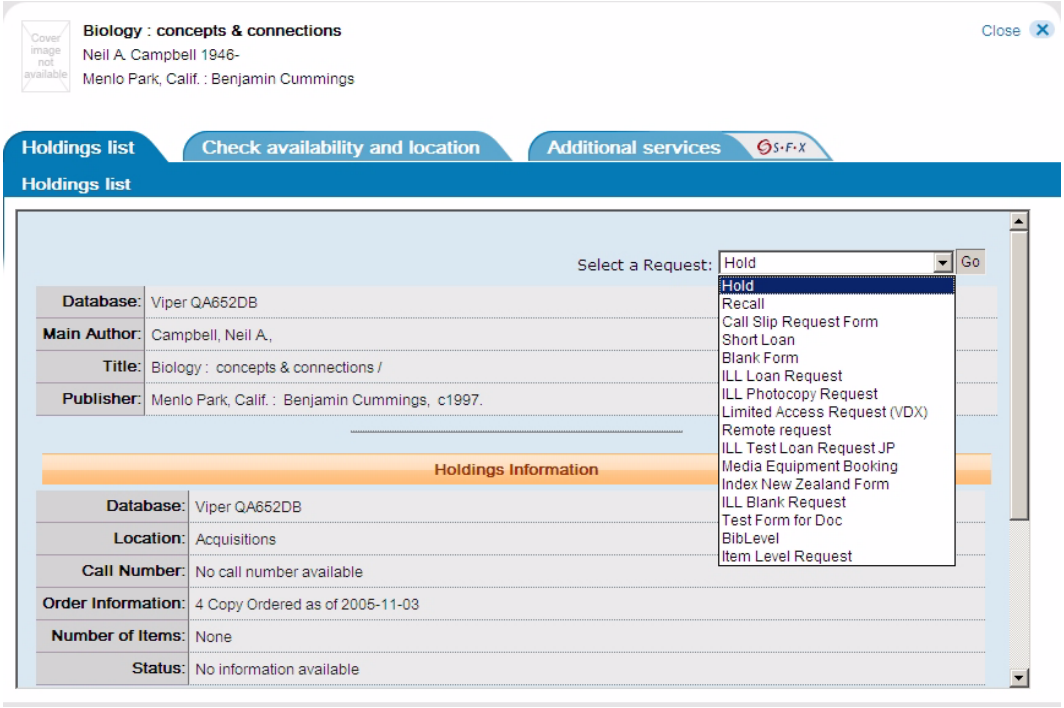

Figure 3-4. Select A Request options

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 successfully processed patron authentication.

Index

Numerics

949, 2-32
950, 2-32
951, 2-32

A

about this document, ix
At Bindery=, 2-30
audience
  of this document, ix
availability pipe, 2-3, 2-29, 2-38
AvailFilter=, 2-35

B

batch job
  Pprimoexp, 2-4, 2-38
BibRangeBegin, 2-4
BibRangeBegin=, 2-33
BibRangeEnd, 2-4
BibRangeEnd=, 2-33
BibsFromFile, 2-4
BibsFromFile=, 2-33

C

Call Slip Request=, 2-30
Cataloging Review=, 2-30
ChangedSince=, 2-34
Charged=, 2-30
checklist, 1-2
Circulation Review=, 2-30
Claims Returned=, 2-30
comments
  about this document, xii
configuration file
  PrimoExp-Availability.ini, 2-3, 2-5
  PrimoExp-Publishing.ini, 2-3, 2-5
  PrimoExp-Selective.ini, 2-4, 2-5
conventions used
  in this document, x

D

Damaged=, 2-30
data extraction, 2-1
Database=, 2-36
DatabaseHost=, 2-36
DatabasePort=, 2-36
DelBibFile=, 2-36
DelBibsDir=, 2-36
DelItemsDir=, 2-36
DelItemsFile=, 2-36
DelMFHDsDir=, 2-36
DelMFHDsFile=, 2-36
Discharged=, 2-30
document summary, x

E

[Enrichment Tags], 2-32
[Excluded Happening Locations], 2-31
[Excluded Operator Ids], 2-31

F

feedback, customer, xii
File=, 2-37

G

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

H

harvesting, 2-3
Heading Changes=, 2-34
Hold Request=, 2-30

I

idPrefix=, 2-37
In Process=, 2-30
In Transit Discharged=, 2-30
In Transit On Hold=, 2-30
In Transit=, 2-30
initial data load, 2-2
InsertAUTTag, 2-32
InsertAVATag, 2-32
InsertSUBTag, 2-32
[Institution Id], 2-31
intended audience
    of this document, ix
[Item Statuses], 2-29
[Item Statuses] parameter
    At Bindery=, 2-30
    Call Slip Request=, 2-30
    Cataloging Review=, 2-30
    Charged=, 2-30
    Circulation Review=, 2-30
    Claims Returned=, 2-30
    Damaged=, 2-30
    Discharged=, 2-30
    Hold Request=, 2-30
    In Process=, 2-30
    In Transit Discharged=, 2-30
    In Transit On Hold=, 2-30
    In Transit=, 2-30
    Lost--Library Applied=, 2-30
    Lost--System Applied=, 2-30
    Missing=, 2-30
    Not Charged=, 2-30
    On Hold=, 2-30
    Overdue=, 2-30
    Recall Request=, 2-30
    Remote Storage Request=, 2-30
    Renewed=, 2-30
    Scheduled=, 2-30
    Short Loan Request=, 2-30
    Withdrawn=, 2-30

L

Library Card, 3-4
[Location to Primo], 2-31
LogFile=, 2-35
LogFileDir=, 2-35
LogLevel=, 2-35
LogToStdOut=, 2-35
Lost--Library Applied=, 2-30
Lost--System Applied=, 2-30

M

Missing=, 2-30

N

Not Charged=, 2-30

O

On Hold=, 2-30
Overdue=, 2-30

P

patron authentication, 3-1
Patron Directory Services (PDS), 3-1
patron information, 3-4
patron requests, 3-4
PbundlePrimo, 2-2, 2-38
PDS, 3-1
photocopying
  documentation, xi
pipes, 2-3
PNX, 2-4
Pprimoexp, 2-4, 2-38
Primo database
  initial data load, 2-2
  ongoing updates, 2-3
Primo Library Card, 3-4
Primo Normalized XML (PNX), 2-4
[PrimoExp], 2-33
[PrimoExp] parameter
  AvailFilter=, 2-35
  BibRangeBegin=, 2-33
  BibRangeEnd=, 2-33
  BibsFromFile=, 2-33
  ChangedSince=, 2-34
  Database=, 2-36
  DatabaseHost=, 2-36
  DatabasePort=, 2-36
  DelBibFile=, 2-36
  DelBibsDir=, 2-36
  DelItemsDir=, 2-36
  DelItemsFile=, 2-36
  DelMFHDsDir=, 2-36
  DelMFHDsFile=, 2-36
  HeadingChanges=, 2-34
  LogFile=, 2-35
  LogFileDir=, 2-35
  LogLevel=, 2-35
  LogToStdOut=, 2-35
  UserPass=, 2-36
PrimoExp-Availability.ini, 2-3, 2-5
PrimoExp-Publishing.ini, 2-3, 2-5
PrimoExp-Selective.ini, 2-4, 2-5
publishing pipe, 2-3, 2-38
purpose
  of this document, ix

R

Recall Request=, 2-30
recsPerGroup=, 2-37
reissue
  reason for, x
Remote Storage Request=, 2-30
Renewed=, 2-30
reproduction, of documentation, xi

S

Scheduled=, 2-30
scheduling, 2-38
script
  PbundlePrimo, 2-2, 2-38
Short Loan Request=, 2-30
stanza
  [Enrichment Tags], 2-32
  [Excluded Happening Locations], 2-31
  [Excluded Operator Ids], 2-31
  [Institution Id], 2-31
  [Item Statuses], 2-29
  [Location to Primo], 2-31
  [PrimoExp], 2-33
  [XMLWriterProtocol], 2-37

T

tab_service.<institute>, 3-2
tar.gz files, 2-4

U

UserPass=, 2-36

V

Voyager institution ID, 2-31

W

Withdrawn=, 2-30

X

[XMLWriterProtocol], 2-37
[XMLWriterProtocol] parameter
    File=, 2-37
    idPrefix=, 2-37
    recsPerGroup=, 2-37
    XSL=, 2-37
XSL=, 2-37