



# Voyager® 7.0

## Technical User's Guide

December 2008

# **CONFIDENTIAL INFORMATION**

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

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

## **DISCLAIMER**

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

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

## **TRADEMARKS**

"Ex Libris," the Ex Libris bridge , Primo, Aleph, Alephino, Voyager, SFX, MetaLib, Verde, DigiTool, Preservation, URM, Voyager, ENCompass, Endeavor eZConnect, WebVoyage, Citation Server, LinkFinder and LinkFinder Plus, and other marks are trademarks or registered trademarks of Ex Libris Ltd. or its affiliates.

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

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

Oracle is a registered trademark of Oracle Corporation.

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

Microsoft, the Microsoft logo, MS, MS-DOS, Microsoft PowerPoint, Visual Basic, Visual C++, Win32, Microsoft Windows, the Windows logo, Microsoft Notepad, Microsoft Windows Explorer, Microsoft Internet Explorer, and Windows NT are registered trademarks and ActiveX is a trademark of the Microsoft Corporation in the United States and/or other countries.

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

Google is a registered trademark of Google, Inc.

Copyright Ex Libris Limited, 2008. All rights reserved.

Document released: December 2008

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

---

# Contents

---

## About This Document

• Purpose	xlv
• Intended Audience	xlv
• Reason for Reissue	xlv
• Document Summary	xlvi
• Conventions Used in This Document	xlviii
• Document Reproduction/Photocopying	xlix
• Comment on This Document	I
To Submit Comments by E-mail	I

---

## 1 Getting Started

• Introduction	1-1
• Prerequisite Skills and Knowledge	1-1
• Before You Begin	1-1

---

## 2 Overview of the Data Conversion Process

• Introduction	2-1
• Purpose of this Chapter	2-2
• Bibliographic Data	2-2
• Authority Data	2-2
• Patron Data	2-3
• Circulation Transactions	2-3
• Vendor Data	2-4

---

## 3 Server Activities in the Voyager System

• Introduction	3-1
----------------	-----

## Contents

• Purpose of This Chapter	3-1
• Services, Scripts, Service Dependencies and Rationale	3-2
Scripts	3-2
Service Dependencies	3-3
• Solaris and AIX Servers	3-4
Starting the Server	3-4
Rebooting or Shutting Down the Server	3-4
Stopping and Starting Oracle Services	3-6
Stopping and Starting Voyager	3-8
Stopping and Starting the Apache Web Server	3-10
Database Backup - Solaris Servers	3-11
Backup Process	3-12
Determining the Backup Device Driver	3-12
Identifying your Data Filesystem/Volume	3-14
Reviewing Cron Backup Results	3-16
• Windows Server 2003	3-16
Starting the Server	3-16
Rebooting or Shutting Down the Server	3-17
Stopping and Starting Oracle Services	3-18
Stopping and Starting Voyager	3-19
Stopping and Starting the Apache Web Server	3-20
Database Backup - Windows Server 2003	3-21
Automating the Backup	3-25
• Restoring from Backup	3-27
• Crons Set Up at Installation	3-27

---

## 4

### Voyager Client Installation and the Voyager.ini file

• Introduction	4-1
• Purpose of This Chapter	4-1
• Prerequisites	4-2
• FTP Clients from Server to PC	4-2
• Installing Voyager	4-5
• Installing Media Scheduling	4-13
• Modifying (Adding or Deleting Clients) an Installation	4-13
• Repairing an Installation of Voyager Clients	4-18
• Uninstalling the Voyager Clients	4-20
• The Voyager.ini File on the PC	4-23

## Contents

Client Module Stanzas	4-26
[GlobalLog] Stanza	4-27
[E-mail] Stanza	4-28
[MARC Posting] Stanza	4-28
[SearchURI] Stanza	4-29
Adding a Button to the Search Dialog Box	4-29
[Upgrade] Stanza	4-30
What Happens When New Software is Detected?	4-32
Circulation - Offline Option	4-41
Circulation Charge Timeout	4-42
Single Client Login	4-42
Encryption	4-43
Staff Client Re-Sort	4-44
Voyager-Aware Re-Sorts	4-45
Quick Sorts	4-46
MARC Record Posting	4-47
Example [MARC POSTing] Stanza	4-47
Selecting Send Record To	4-49
• Additional Files Installed with the Client Installation	4-50

---

## 5

### Patron Extract

• Introduction	5-1
• Purpose of this Chapter	5-1
• Overview of the Patron Extract Program	5-1
• Parameters	5-2
Running Patron Extract	5-4
• Output File Specification	5-4
• Additional Files	5-5
Audit File	5-5
Error File (Exception File)	5-6

---

## 6

### Patron Update

• Introduction	6-1
• Purpose of this Chapter	6-1
• Overview the Patron Update Program	6-1

## Contents

• Input File Specification	6-3
• Parameters	6-3
Running Patron Update	6-6
• Additional Files	6-7
Audit File	6-7
Error File (Exception File)	6-8

---

7

## Bursar Transfer System

• Introduction	7-1
• Purpose of this Chapter	7-1
• Overview of the Bursar Transfer System	7-2
• Steps of the Bursar Transfer System	7-3
• Supported One-Way Transfer Types	7-4
Transferring Total Patron Balances	7-4
Transferring Itemized Patron Fines/Fees	7-4
• Configuration File	7-5
• Parameters	7-6
Running Bursar Transfer	7-7
• Output File Specification	7-8
Bursar SIF Format	7-8
• Additional Files	7-9
Audit File	7-9
Error File	7-10
• Bursar Transfers in the Circulation Module	7-10

---

8

## Circulation Batch Jobs

• Introduction	8-1
• Purpose of This Chapter	8-2
• Update Shelving Status (Circjob 1)	8-3
• Archive and Expire Call Slip Requests (Circjob 8)	8-4
• Export OPAC Requests (Circjob 26)	8-4
• Archive Short Loans (Circjob 27)	8-5
• Purge Universal Borrowing (UB) Patron Stub Records (Circjob 29)	8-5
• Accrued Fines and Demerits (Circjob 30)	8-6

## Contents

• Patron Suspension (Circjob 31)	8-7
• Universal Borrowing (UB) Request Promotion (Circjob 32)	8-8
• Update Remote Circulation Cluster Cache (Circjob 33)	8-9
• Place Items on Active Course Reserve List (Circjob 34)	8-10
• Place Recalls and Holds for Items on Active Course Reserve List (Circjob 35)	8-12
• Take Items on Inactive Course Reserve List Off Reserve (Circjob 36)	8-14
• Forgive Demerits (Circjob 37)	8-16
• Retain Patron IDs (Circjob 38)	8-17
• Patron Purge (Circjob 39)	8-23
Patron Purge Files	8-23
Patron Purge Command Line (Circjob 39)	8-25
• Forgive Fines by Patron ID (Circjob 40)	8-27
Running circjob 40 from the command line	8-27
Error Logging for Batch Forgive Jobs	8-29
circjob.log	8-29
err.forgive.YYYYMMDD.HHMM	8-29
Audit Report for Batch Forgive Jobs	8-29
• Forgive Fines by Create Date (Circjob 41)	8-30
Running circjob 41 from the command line	8-30
Error Logging for Batch Forgive Jobs	8-32
circjob.log	8-32
err.forgive.YYYYMMDD.HHMM	8-32
Audit Report for Batch Forgive Jobs	8-33
• Forgive Fines by Patron Group and Expire Date (Circjob 42)	8-33
Running circjob 42 from the command line	8-33
Error Logging for Batch Forgive Jobs	8-35
circjob.log	8-35
err.forgive.YYYYMMDD.HHMM	8-36
Audit Report for Batch Forgive Jobs	8-36
• Synchronize Patron Counters for Universal Borrowing (Circjob 43)	8-37
Frequency of Use	8-37
How to Access	8-37
Server Access	8-38
Command Line Access	8-39
WebAdmin Access	8-40
Circjob Log	8-40

## Contents

Error Conditions	8-40
------------------	------

---

**9**

### Media Scheduling Batch Jobs

• Introduction	9-1
• Purpose of this Chapter	9-1
• Retain Patron IDs (Mediajob 5)	9-1

**10**

### Bulk Export of MARC Records

• Introduction	10-1
• Purpose of this Chapter	10-1
• Overview of Bulk Export	10-1
• Parameters	10-3
Running the Marlexport Program	10-9
• Output File Specification	10-10
• Additional Files	10-10
Audit File	10-10

**11**

### Prebulk Program

• Introduction	11-1
• Purpose of this Chapter	11-2
• Overview of Prebulk Processing	11-2
Using Prebulk to Check the Records	11-3
Using Prebulk to Strip Fields	11-3
Using Prebulk to Create an Interleaved File	11-4
Bulk Importing of the Interleaved File	
Generated by Prebulk	11-4
Holdings Created from the Interleaved File	11-5
• Input File	11-5
• Creating the Configuration File	11-5
Default Location of the Configuration File	11-6
Name of the Configuration File	11-7
Stanza Types	11-7
Using the Default Call Number	11-7

---

## Contents

Using a Call Number Stored in the Record	11-7
Overrides Stanza	11-8
CREATEMFHD	11-8
DEFAULTCALLNO	11-9
DEFAULTCALLIND	11-9
USE001FOR014	11-9
USE003FORLOC	11-9
Strip Stanza	11-9
Non-Standard 035 Field Elimination	11-10
MFHDTAG Stanza	11-11
CALLTYPES Stanza	11-12
LOCATIONS Stanza	11-13
MAPPING Stanza	11-15
008 Stanza	11-17
• Parameters	11-20
Running the Prebulk Program	11-21
• Output File Specification	11-22
• Additional Files	11-22
Log (Audit) File	11-22
Error File	11-22

---

**12****Bulk Import, Replace, and Merge of MARC Records**

• Introduction	12-1
• Purpose of this Chapter	12-2
• Overview of the Bulk Import, Replace, Merge MARC Records Program	12-2
The UTF-8 Character Set Encoding: Character Set Mapping, Record Leader, and Conversion Attempts	12-3
Character Set Mapping	12-4
Leader Byte 9 Value	12-4
Converting Records to Unicode	12-4
Disabling the Generation of the Keyword Index Files	12-6
• Parameters	12-7
Running Bulk Import	12-10
• Input File Specification	12-11
• Building the 035	12-11

## Contents

OCLC Control Number Expansion	12-12
• Creating Holdings and Item Records	12-12
Call Number Hierarchy	12-12
Barcode	12-13
Item Type	12-13
Mapping	12-13
• Additional Files	12-13
Log file	12-13
Delete, Discard, Replace, and Reject files	12-14
Error File	12-15
• Messages in Log and Error Files	12-15

---

## 13

### Global Heading Change Jobs

• Introduction	13-1
• Purpose of This Chapter	13-1
• Overview of GHC	13-1
• Parameters	13-2
Process Global Heading Change Queue Step 1 (Catjob 11)	13-4
Process Global Heading Change Queue Step 2 (Catjob 12)	13-5
Process Global Heading Change Fields Step 3 (Catjob 13)	13-6

---

## 14

### Storage Barcode Verify (Pstrgvfy) Program

• Introduction	14-1
• Purpose of This Chapter	14-2
• Changing Location Codes of Item Records or MFHDs	14-2
• Populating the Operator or Location Columns of a MFHD's History Tab	14-3
• Verifying Barcodes of Item Records or MFHDs	14-4
• Pstrgvfy Input File	14-4
• Pstrgvfy Parameters	14-5
• Pstrgvfy Command File	14-8
• Pstrgvfy Log File	14-9
• Sample Pstrgvfy Command Line	14-10

---

## Contents

---

**15****Popacjob**

• Introduction	15-1
• Purpose of This Chapter	15-1
OPAC_Search_Log Table	15-2
Bib_usage_log Table	15-3
Setting Up OPAC Search Logging	15-4
Setting Up Bibliographic Usage Logging	15-5
Setting up Selective Dissemination of Information (SDI)	15-6
• Accessing Logged Information	15-6
OPAC Search Log Export program	15-6
Interactive Method	15-7
Cron Method	15-8
SDI Searches Program	15-9
Enabling SDI	15-10
Configuring SDI Options	15-10
Customizing the Search Results URL Page	
Sent to Patrons by SDI	15-12
Interactive Method	15-14
Cron Method	15-15
OPAC Bib Usage Log Export Program	15-15
Interactive Method	15-15
Cron Method	15-17

---

**16****Acquisitions Batch Job - Fix Exchange Rates**

• Introduction	16-1
• Fix Exchange Rates	16-1
Purchase Orders Updated with Acqjob 5	16-2
Additional Considerations for Acqjob 5	16-3
• Running Acqjob 5	16-3
Running Acqjob 5 from the Command Line	16-4
Running Acqjob 5 Interactively	16-4
• Log File	16-6

## Contents

---

17

### Server-Side Configurations

• Introduction	17-1
• UB Barcode Lookup	17-1
Configuring the UB Barcode Lookup Feature	17-1
voyager.env	17-2
Creating the UB Barcode Lookup Configuration File	17-2
UB Barcode Lookup Configuration Schema File	17-8
Configuration File Validation	17-11
• Dynamic Noise Word Reduction	17-12
• Configuring the Z39.50 Server for UTF-8 Encoded Records	17-13

---

18

### Patron Record Standard Interface File

• Introduction	18-1
• Purpose of This Chapter	18-1
• File Specification	18-2
• Patron Record SIF Format	18-2
Base Segment	18-3
Address Segment	18-10
Variable Segment	18-12
End-of-Record Segment	18-13

---

19

### Charge Transaction Record Standard Interface File

• Introduction	19-1
• Purpose of This Chapter	19-1
• Input File Specification	19-1
• Charge Transaction Record SIF Format	19-2

---

20

### Item Delete Standard Interface File

• Introduction	20-1
----------------	------

---

## Contents

- Purpose of This Chapter 20-1
  - File Format 20-1
- 

**21****Vendor Record Standard Interface Format**

- Introduction 21-1
- Purpose of This Chapter 21-1
- Input File Specification 21-1
- Vendor SIF Format 21-2
  - Base Segment 21-3
  - Address Segment 21-4
  - Variable Segment 21-6
  - End-of-Record Segment 21-6

---

**22****Acquisitions Notices Standard Interface File**

- Introduction 22-1
- Purpose of This Chapter 22-1
- File Specification 22-2
- Acquisitions Notices SIF Format 22-3
  - Base Segment 22-4
  - Suffixes 22-5
    - Cancellation Notice Suffix (00) 22-6
    - Return Notice Suffix (01) 22-6
    - Order Claim Notice Suffix (02) 22-7
    - Serial Claim Notice Suffix (03) 22-7
    - Voucher/Check Request Suffix (04) 22-8
    - Cancel Serial Claim Notice Suffix (05) 22-8

---

**23****Acquisitions Reports Standard Interface File**

- Introduction 23-1
- Purpose of This Chapter 23-1
- File Specification 23-1
- Acquisitions Reports SIF Format 23-2

## Contents

Base Segment	23-4
Suffixes	23-4
Purchase Orders Report (00)	23-5
Open Orders Report (01)	23-8
Global Open Orders Report (02)	23-9
Fund Snapshot Report Suffix (03)	23-10
Open Orders Before Rollover Report (04)	23-12
Open Orders After Rollover Report (05)	23-12
Fund Snapshot Before Rollover Report (06)	23-12
Fund Snapshot After Rollover Report (07)	23-12
Fund Rollover Status Report Suffix (08)	23-12
Copy Rollover Status Report Suffix (09)	23-13

---

**24**

### Cataloging Reports Standard Interface File

• Introduction	24-1
• Purpose of This Chapter	24-1
• File Specification	24-1
• Cataloging Reports SIF Format	24-2
Base Segment	24-4
Suffixes	24-4
Unauthorized Subject Headings (00)	24-5
Unauthorized Name Headings (01)	24-5
Unauthorized Title Headings (02)	24-5
Unauthorized Name/Title Headings (03)	24-6
Unauthorized Subdivision Headings (04)	24-6
Duplicate Authority Records (05)	24-6
‘See’ References with Linked Bib Records (06)	24-7
‘See’ References Authorized in Another Authority Record (07)	24-7
‘See’ References without Corresponding Authority Record (08)	24-8
856 Link Failure Report Suffix (09)	24-8

---

**25**

### Circulation Notices Standard Interface File

• Introduction	25-1
----------------	------

---

## Contents

• Purpose of This Chapter	25-1
• File Specification	25-2
• Circulation Notices SIF Format	25-3
Base Segment	25-4
Suffixes	25-5
Cancellation Notice Suffix (00)	25-6
Item Available Notice Suffix (01)	25-6
Overdue Notice Suffix (02)	25-6
Recall Notice Suffix (03)	25-7
Recall-Overdue Notice Suffix (04)	25-7
Fine/Fee Notice Suffix (05)	25-7
Statement of Fines and Fees Suffix (06)	25-8
Courtesy (Due) Notice Suffix (07)	25-9

---

## 26

### Circulation Reports Standard Interface File

• Introduction	26-1
• Purpose of This Chapter	26-1
• File Specification	26-1
• Circulation Reports SIF Format	26-2
Base Segment	26-4
Suffixes	26-4
Reserved Items Active Report (00)	26-5
Reserved Items Expired Report (01)	26-5
Hold Shelf Expired Report (02)	26-5
Missing in Transit Report (03)	26-6
Circulation Statistics Report (04)	26-6
Circulation Item-Related Exceptions Report (05)	26-7
Circulation Patron-Related Exceptions Report (06)	26-7
Circulation Transaction-Related Exceptions Report (07)	26-7
Global Circulation Statistics Report (08)	26-8
Distribution Item Order List Report (09)	26-8

## Contents

---

**27**

### **Media Scheduling Notices Standard Interface File**

• Introduction	27-1
• Purpose of This Chapter	27-1
• File Specification	27-2
• Media Scheduling Notices SIF Format	27-3
Base Segment	27-3
Suffixes	27-4
Overdue Notice Suffix (00)	27-5

**28**

### **Media Scheduling Reports Standard Interface File**

• Introduction	28-1
• Purpose of This Chapter	28-1
• File Specification	28-1
• Media Scheduling Reports SIF Format	28-2
Base Segment	28-3
Suffixes	28-3
Media Equipment Inventory Report (00)	28-4
Booking Statistics Report (01)	28-4
Booking Exceptions Report (02)	28-5
Booking Charge Statistics Report (03)	28-5

**29**

### **Database Views**

• Introduction	29-1
• Purpose of This Chapter	29-2
• Views	29-2
Authblob_vw	29-2
Authhistory_vw	29-3
Authheading_vw	29-3
Authority1xx4xx_vw	29-4
Authority5xx1xx_vw	29-4
Authoritydupe_vw	29-5
Bib_vw	29-5
Bibblet_vw	29-7

---

## Contents

Bibhistory_vw	29-7
Bibloc_vw	29-8
Circcharges_vw	29-8
Circrenewal_vw	29-10
Fundledger_vw	29-12
Heading_vw	29-14
Issues_vw	29-15
Item_vw	29-16
LCclass_vw	29-18
Marccomputer_vw	29-19
Marcbook_vw	29-19
Marcmap_vw	29-20
Marcmusic_vw	29-20
Marcserial_vw	29-21
Marcvisual_vw	29-22
MFHDblob_vw	29-22
MFHDhistory_vw	29-23
NLMclass_vw	29-23
Recordcount_vw	29-24
Serials_vw	29-24
Sudocclass_vw	29-25
Vendorinvoice_vw	29-26
Vendororder_vw	29-28

---

**30****WebAdmin**

• Introduction	30-1
• The Purpose of this Chapter	30-2
• Setting Up WebAdmin	30-2
System Requirements	30-2
Starting the Server Daemon	30-3
Creating WebAdmin Users and Passwords	30-3
Adding the WebAdmin Authorization Section to the httpd.conf File	30-5
• Starting the Voyager Server Utilities	30-6
• Using WebAdmin	30-7
Voyager Server Utilities Page	30-7
Acquisitions Utilities	30-9
Cataloging Utilities	30-12

## Contents

Circulation Utilities	30-24
Media Scheduling Utilities	30-32
OPAC Reports Utilities	30-34
System Admin Utilities	30-36

---

**A****Data Dictionary**

A-1

**B****UseMARCON Configuration for Use with Voyager**

• What is UseMARCON?	B-1
In This Chapter	B-1
• Supported MARC Formats	B-2
• UseMARCON File Structure	B-2
UseMARCON Conversion Directories	B-2
UseMARCON Initialization (ini) File	B-3
Required Files	B-4
Sample UseMARCON Initialization File	B-5
• UseMARCON Log File	B-5
Fatal/Non-Fatal UseMARCON Errors	B-7
Fatal Errors	B-7
Non-Fatal Errors	B-8
Sample Fatal Error (END non OK) Message in UseMARCON Log File	B-8
Troubleshooting UseMARCON Error Conditions	B-8
UseMARCON Log File Example	B-9
• Bulk Import and UseMARCON	B-10
Overview	B-11
Bulk Import Example	B-11
Bulk Import Error Reporting	B-11
• MARC Export and UseMARCON	B-12
Overview	B-12
MARC Export Example	B-12
MARC Export Error Reporting	B-13
• Z39.50 Server and UseMARCON Interaction	B-13
Overview	B-13
The z3950svr.ini File	B-13

## Contents

Sample [usemarcon translations] Stanza in z3950svr.ini File (Lines 1-3)	B-15
UseMARCON/Z39.50 Error Reporting	B-15
• UseMARCON Error Numbers	B-16
General Error Numbering	B-16
Error Message Numbers	B-17

---

## C

<b>WebVoyáge Patron Authentication Adapter Feature</b>	C-1
• Overview	C-1
• Detailed WebVoyáge Patron Adapter Interaction	C-3
• WebVoyáge Configuration	C-6
Enabling and Disabling Webvoyáge Adapter Behavior	C-7
Setting the URL to the External Authentication System	C-8
Enabling and Disabling the WebVoyáge Patron Login Bypass	C-8
Setting the Text of the External Authentication Link Button	C-8
Format of the Redirect URL from the Authentication Adapter to WebVoyáge	C-8
• Example	C-9

---

## IN

## Index

IN-1

## Contents

## Tables

---

<b>3</b>	<b>Server Activities in the Voyager System</b>	
3-1.	Service Dependencies	3-3
3-2.	Filesystems on the server	3-15
 <b>4</b>	<b>Voyager Client Installation and the Voyager.ini file</b>	
4-1.	Module Stanza Parameters	4-26
4-2.	[GlobalLog] Stanza Parameters	4-27
4-3.	[E-mail] Stanza Parameters	4-28
4-4.	[SearchUri] Stanza Parameters	4-29
4-5.	Options for .bat file method	4-37
4-6.	Circulation ChargeTimeout Key Values	4-42
 <b>7</b>	<b>Bursar Transfer System</b>	
7-1.	Bursar SIF Format	7-8
7-2.	Audit Log SIF Format	7-10
 <b>8</b>	<b>Circulation Batch Jobs</b>	
8-1.	Parameters for Circjob 34	8-10
8-2.	Description of output SIF from Circjob 34	8-11
8-3.	Parameters for Circjob 35	8-12
8-4.	Description of Output SIF from Circjob 35	8-14
8-5.	Parameters for Circjob 36	8-15
8-6.	Description of output SIF from Circjob 36	8-15
8-7.	Parameters for Circjob 38	8-18
8-8.	Patron Purge Exceptions/Names	8-24
8-9.	Parameters for Circjob 39	8-25
8-10.	Circjob 43 Parameters	8-39

## Tables

---

<b>9</b>	<b>Media Scheduling Batch Jobs</b>	
	9-1. Parameter for Mediajob 5	9-3
<b>10</b>	<b>Bulk Export of MARC Records</b>	
	10-1. Record Type and Corresponding Parameter Value	10-3
	10-2. Character Mapping Codes	10-8
<b>12</b>	<b>Bulk Import, Replace, and Merge of MARC Records</b>	
	12-1. Bulk Import Conversion Situations	12-5
<b>15</b>	<b>Popacjob</b>	
	15-1. OPAC_search_log Table	15-2
	15-2. Bib_usage_log Table	15-3
	15-3. Components of the [SDI_Page] Stanza	15-10
<b>16</b>	<b>Acquisitions Batch Job - Fix Exchange Rates</b>	
	16-1. Acqjob 5 Batch Job Parameters	16-3
<b>17</b>	<b>Server-Side Configurations</b>	
	17-1. UB Barcode Lookup Configuration File Commands	17-5
	17-2. NOISEWORDFILTER and Keyword Searches	17-12

---

## Tables

---

<b>18</b>	<b>Patron Record Standard Interface File</b>	
18-1.	Base (Fixed) Segment of Record	18-4
18-2.	Address Segment of Record.	18-10
18-3.	Variable (Notes) segment of record	18-12
18-4.	End-of-record marker	18-13
<b>19</b>	<b>Charge Transaction Record Standard Interface File</b>	
19-1.	Charge Transaction Standard Interface File Format	19-3
<b>20</b>	<b>Item Delete Standard Interface File</b>	
20-1.	Item Delete SIF Format	20-2
<b>21</b>	<b>Vendor Record Standard Interface Format</b>	
21-1.	Base Segment of Record	21-3
21-2.	Address segment of record	21-4
21-3.	Variable (Notes) segment of record	21-6
21-4.	End-of-record marker	21-6
<b>22</b>	<b>Acquisitions Notices Standard Interface File</b>	
22-1.	Base Segment for Acquisitions Notices	22-4
22-2.	Cancellation Notice Suffix (00)	22-6
22-3.	Return Notice Suffix (01)	22-6
22-4.	Order Claim Notice Suffix (02)	22-7
22-5.	Serial Claim Notice Suffix (03)	22-7
22-6.	Voucher/Check Request Suffix (04)	22-8
22-7.	Cancel Serial Claim Notice Suffix (05)	22-8

## Tables

---

---

**23****Acquisitions Reports Standard Interface File**

23-1.	Base segment for acquisitions reports	23-4
23-2.	Purchase Orders Report Suffix (00)	23-5
23-3.	Open Orders Report Suffix (01)	23-8
23-4.	Global Open Orders Report Suffix (02)	23-9
23-5.	Fund Snapshot Report Suffix (03)	23-10
23-6.	Fund Rollover Status Report Suffix (08)	23-12
23-7.	Copy Rollover Status Report Suffix (09)	23-13

---

**24****Cataloging Reports Standard Interface File**

24-1.	Base segment for cataloging reports	24-4
24-2.	Unauthorized Subject Headings Report Suffix (00)	24-5
24-3.	Unauthorized Name Report Suffix (01)	24-5
24-4.	Unauthorized Title Headings Report Suffix (02)	24-5
24-5.	Unauthorized Name/Title Headings Report Suffix (03)	24-6
24-6.	Unauthorized Subdivision Headings Report Suffix (04)	24-6
24-7.	Duplicate Authority Records Report Suffix (05)	24-6
24-8.	'See' References with Linked Bib Records Report Suffix (06)	24-7
24-9.	'See' References Authorized in Another Authority Record Suffix (07)	24-7
24-10.	'See' References without Corresponding Authority Record Report Suffix (08)	24-8
24-11.	856 Link Failure Report Suffix (09)	24-8

---

**25****Circulation Notices Standard Interface File**

25-1.	Base segment for circulation notices	25-4
25-2.	Cancellation Notice Suffix (00)	25-6
25-3.	Item Available Notice Suffix (01)	25-6
25-4.	Overdue Notice Suffix (02)	25-6
25-5.	Recall Notice Suffix (03)	25-7
25-6.	Recall-Overdue Notice Suffix (04)	25-7

## Tables

---

25-7. Fine/Fee Notice Suffix (05)	25-7
25-8. Statement of Fines and Fees Suffix (06)	25-8
25-9. Courtesy (Due) Notice Suffix (07)	25-9

---

**26****Circulation Reports Standard Interface File**

26-1. Base segment for circulation reports	26-4
26-2. Reserved Items Active Report Suffix (00)	26-5
26-3. Reserved Items Expired Report Suffix (01)	26-5
26-4. Hold Shelf Expired Report Suffix (02)	26-5
26-5. Missing in Transit Report Suffix (03)	26-6
26-6. Circulation Statistics Report Suffix (04)	26-6
26-7. Circulation Item--related Exceptions Report Suffix (05)	26-7
26-8. Circulation Patron-related Exceptions Report Suffix (06)	26-7
26-9. Circulation Transaction-related Exceptions Report Suffix (07)	26-7
26-10. Global Circulation Statistics Report Suffix (08)	26-8
26-11. Distribution Item Order List Report Suffix (09)	26-8

---

**27****Media Scheduling Notices Standard Interface File**

27-1. Base Segment for Media Scheduling Notices	27-3
27-2. Overdue Notice Suffix (00)	27-5

---

**28****Media Scheduling Reports Standard Interface File**

28-1. Base Segment for Media Scheduling Reports	28-3
28-2. Media Equipment Inventory Report Suffix (00)	28-4
28-3. Booking Statistics Report Suffix (01)	28-4
28-4. Booking Exceptions Report Suffix (02)	28-5
28-5. Booking Charge Statistics Report Suffix (03)	28-5

**Tables**

---

**29****Database Views**

29-1. Authblob_vw	29-2
29-2. Authhistory_vw	29-3
29-3. Authheading_vw	29-3
29-4. Authority1xx4xx_vw	29-4
29-5. Authority5xx1xx_vw	29-4
29-6. Authoritydupe_vw	29-5
29-7. Bib_vw	29-5
29-8. Bibblob_vw	29-7
29-9. Bibhistory_vw	29-7
29-10. Bibloc_vw	29-8
29-11. Circcharges_vw	29-8
29-12. Circrenewal_vw	29-10
29-13. Fundledger_vw	29-12
29-14. Heading_vw	29-14
29-15. Issues_vw	29-15
29-16. Item_vw	29-16
29-17. LCclass_vw	29-18
29-18. Marccomputer_vw	29-19
29-19. Marcbook_vw	29-19
29-20. Marcmap_vw	29-20
29-21. Marcmusic_vw	29-20
29-22. Marcserial_vw	29-21
29-23. Marcvisual_vw	29-22
29-24. MFHDblob_vw	29-22
29-25. MFHDhistory_vw	29-23
29-26. NLMclass_vw	29-23
29-27. Recordcount_vw	29-24
29-28. Serials_vw	29-24
29-29. Sudoclass_vw	29-25
29-30. Vendorinvoice_vw	29-26
29-31. Vendororder_vw	29-28

---

**30****WebAdmin**

---

## Tables

---

### B

#### **UseMARCON Configuration for Use with Voyager**

B-1.	Required Fields	B-4
B-2.	Descriptions of Error Message Lines in UseMARCON Log File	B-9
B-3.	MARC Formats and Corresponding Z39.50 Format Codes	B-14
B-4.	General UseMARCON Error Message Guidelines	B-16
B-5.	UseMARCON Error Messages	B-17

---

### C

#### **WebVoyage Patron Authentication Adapter Feature**

C-1.	URL field Components	C-9
------	----------------------	-----

## Tables

## Figures

---

**3****Server Activities in the Voyager System**

3-1.	Example Output	3-14
3-2.	Standard Backup Script for Windows Server 2003s	3-24
3-3.	Backup Destination Field	3-25

---

**4****Voyager Client Installation and the Voyager.ini file**

4-1.	Run dialog box with server IP address where Voyager clients reside	4-3
4-2.	Example FTP session	4-5
4-3.	Voyager setup Welcome dialog box	4-6
4-4.	Voyager installation User Information dialog box	4-7
4-5.	Voyager installation User Information dialog box w/o administrator privileges	4-8
4-6.	Voyager installation Destination Folder dialog box	4-9
4-7.	Voyager installation Select an Installation Type dialog box	4-10
4-8.	Warning Message	4-11
4-9.	Installation in process	4-12
4-10.	Voyager Installation Successful Message	4-12
4-11.	Select Installation Type dialog box	4-14
4-12.	Voyager Setup with Selected Features	4-15
4-13.	Selecting System Administration client for deletion	4-16
4-14.	System Administration client identified for deletion	4-16
4-15.	Voyager ready to modify	4-17
4-16.	Voyager Programs Installation Successful	4-18

## Figures

4-17. Voyager Ready to Repair the Installation	4-19
4-18. Voyager Installation Successful	4-20
4-19. Control Panel with Add/Remove Programs Selected	4-21
4-20. Add/Remove Dialog Box with Voyager Selected	4-22
4-21. Confirmation of Removing Voyager	4-22
4-22. Sample voyager.ini file	4-23
4-23. Sample [SearchURI] Stanza	4-30
4-24. "Go to Google" Button Added to Search Dialog Box	4-30
4-25. [Upgrade] stanza example using a valid UNC resource path	4-31
4-26. [Upgrade] stanza example using an http resource	4-31
4-27. [Upgrade] stanza example using an ftp resource	4-31
4-28. [Upgrade] stanza example using a .bat file	4-32
4-29. Version available message	4-32
4-30. Installation Welcome dialog box	4-33
4-31. Security page (authentication)	4-34
4-32. File download dialog box	4-35
4-33. File download security warning dialog box	4-35
4-34. file download security warning dialog box	4-36
4-35. Example code for .bat method	4-37
4-36. Offline Circulation Message	4-41
4-37. Padlock Indicating That Encryption is Enabled	4-44
4-38. Search Results Page with Voyager-Aware and Quick Sort Drop-Down	4-45
4-39. Search Results with Voyager-Aware Drop-Down Menu	4-46
4-40. Search Results with Quick Sort Drop-Down Menu	4-47

## Figures

---

4-41. Sample [MARC POSTing] stanza (Classic WebVoyáge)	4-48
4-42. Sample [MARC POSTing] stanza (WebVoyáge)	4-48
4-43. Send Record To option	4-48
4-44. Send Record To button and drop-down as seen in the Acquisitions and Circulation modules	4-49

---

7

## Bursar Transfer System

7-1. Sample Configuration File	7-5
--------------------------------	-----

---

8

## Circulation Batch Jobs

8-1. Example of the Message in Circjob.log file after Running Circjob 38	8-19
8-2. Running Circulation Batch Job 38 Using the Command Line	8-20
8-3. Circjob.log file after Running Circjob 38	8-20
8-4. Available Circulation Batch Jobs	8-21
8-5. Providing the Circulation Batch Job Number	8-22
8-6. Example of Retaining Five Patron IDs	8-22
8-7. On Screen Display when Running Circulation Batch Job 38	8-23
8-8. Circjob.log File after Running Circjob 38	8-23
8-9. Purging Patron Records from the Database (Circjob 39)	8-27
8-10. Circjob 43 option on the server	8-38
8-11. Circjob 43 option from WebAdmin	8-40

## Figures

---

9

### Media Scheduling Batch Jobs

9-1.	Mediajob.log File after Running Mediajob 5	9-3
9-2.	Running Media Scheduling Batch Job 5 Using the Command Line	9-4
9-3.	Mediajob.log File after Running Mediajob 5	9-5
9-4.	Available Media Batch Jobs	9-5
9-5.	Providing the Media Scheduling Batch Job Number	9-6
9-6.	Example of Retaining Three Patron IDs	9-6
9-7.	On Screen Display when Running Media Scheduling Batch Job 5	9-7
9-8.	Mediajob.log file after Running Mediajob 5	9-7

10

### Bulk Export of MARC Records

10-1.	Example of Records where the Middle Dot Character Displays	10-9
-------	--	------

11

### Prebulk Program

11-1.	Sample LOCATIONS Stanza	11-6
11-2.	OVERRIDES Stanza Example	11-8
11-3.	The STRIP Stanza	11-9
11-4.	The MFHDTAG Stanza	11-12
11-5.	The CALLTYPES Stanza	11-12
11-6.	The LOCATIONS Stanza	11-13
11-7.	Sample LOCATIONS Stanza	11-15
11-8.	The MAPPING Stanza	11-16

## Figures

11-9. The 008 Stanza	11-17
11-10. Sample Configuration File	11-19

---

**15**

### Popacjob

15-1. Sample StatString Results	15-6
15-2. OPAC Job Options Menu	15-7
15-3. OPAC Search Log Export Job Date Range Prompt	15-8
15-4. The Default [SDI_Page] Stanza of the opac.ini File.	15-10
15-5. Default sdiemail.ini File	15-13
15-6. Sample SDI URL in email	15-13
15-7. OPAC Job Options Menu	15-14
15-8. OPAC Job Options Menu	15-16
15-9. OPAC Bib Usage Log Export Job Date Range Prompt	15-17

**16**

### Acquisitions Batch Job - Fix Exchange Rates

16-1. Screen Display for Acqjob 5 (Fix Exchange Rates)	16-5
16-2. Foreign Commitments Log File	16-7
16-3. Acqjob.log file	16-7

---

**17**

### Server-Side Configurations

17-1. Skeleton UB Configuration File	17-4
17-2. UB Configuration Schema File (Continued on next page)	17-8
17-3. UB Configuration Log File	17-11
17-4. Sample z3950svrUTF8.ini File Modifications	17-14
17-5. Sample Pz3950svr Script File Modifications	17-14

## Figures

---

**30**

### WebAdmin

30-1.	WedAdmin Log in Dialog Box	30-6
30-2.	Webadmin Main Page	30-8
30-3.	WebAdmin Acquisitions Reports and Notices Page	30-10
30-4.	Fix Exchange Rate Options Page	30-11
30-5.	Job Scheduler Page after Submitting Acqjob 5	30-12
30-6.	MARC Record Bulk Import Page	30-14
30-7.	Marc Record Bulk Export Page	30-18
30-8.	Drop-Down Menu Options on the Cataloging Reports and Processes Page	30-23
30-9.	Select Location and Operator to Run Catjob 13	30-23
30-10.	Circjob 40, WebAdmin	30-25
30-11.	Circjob 41, WebAdmin	30-26
30-12.	Circjob 42, WebAdmin	30-28
30-13.	Bursar Transfer Page	30-30
30-14.	Mediajob Reports and Processes Page	30-33
30-15.	Job Scheduler Page after Submitting the Job	30-34
30-16.	OPAC Reports and Processes Page	30-35

**B**

### UseMARCON Configuration for Use with Voyager

B-1.	Sample Contents of /m1/voyager/xxxdb/ini/usemarcon	B-3
B-2.	Sample Contents of UNI2US Conversion Profile	B-3
B-3.	Sample of the required files in the UseMARCON initialization file	B-5
B-4.	Sample log file	B-6
B-5.	Sample Fatal Error Message in a UseMARCON Log File	B-8
B-6.	Sample UseMARCON Log File Error Message Line	B-9
B-7.	Sample [usemarcon translations] Stanza	B-15

## Figures

---

C

### WebVoyáge Patron Authentication Adapter Feature

C-1. Patron Authentication Adapter Flow Diagram	C-3
C-2. webvoyage.properties example	C-7

## **Figures**

## Procedures

---

---

**3**

### **Server Activities in the Voyager System**

3-1. Rebooting or Shutting Down a Solaris Server	3-5
3-2. Rebooting or Shutting Down an AIX Server	3-6
3-3. Stopping Oracle Services on a Solaris or AIX Server	3-7
3-4. Starting Oracle Services on a Solaris and AIX Server	3-7
3-5. Stopping Voyager on a Solaris Server	3-8
3-6. Starting Voyager on a Solaris Server	3-8
3-7. Stopping Voyager on an AIX Server	3-9
3-8. Starting Voyager on an AIX Server	3-9
3-9. Stopping the Apache Web Server	3-10
3-10. Starting the Apache Web Server	3-10
3-11. Determining the Backup Device Driver	3-12
3-12. Identifying your Data Filesystem	3-15
3-13. Reviewing the Email Generated by the Cron Backup	3-16
3-14. Rebooting or Shutting Down	3-17
3-15. Stopping Oracle Services	3-18
3-16. Starting Oracle Services	3-19
3-17. Stopping Voyager	3-19
3-18. Starting Voyager	3-20
3-19. Stopping the Apache Web Server	3-20
3-20. Starting the Apache Web Server	3-21
3-21. Customizing the Backup Script	3-24
3-22. Automating the Backup	3-25

## Procedures

---

4

### Voyager Client Installation and the Voyager.ini file

4-1. Downloading Client Files Using FTP	4-3
4-2. Installing Voyager Clients	4-6
4-3. Adding or Deleting Voyager Clients	4-13
4-4. Repairing an Installation	4-18
4-5. Uninstalling Voyager Clients	4-21

---

8

### Circulation Batch Jobs

8-1. Running Circjob 38 from the Command Line	8-19
8-2. Running Circjob 38 Interactively	8-20
8-3. Purging Patron Records from the Database (Circjob 39)	8-26
8-4. Running circjob 40 interactively	8-28
8-5. Running circjob 41 interactively	8-31
8-6. Running circjob 42 interactively	8-34

---

9

### Media Scheduling Batch Jobs

9-1. Running Mediajob 5 from the Command Line	9-4
9-2. Running Mediajob 5 Interactively	9-5

---

17

### Server-Side Configurations

17-1. Creating the UB Barcode Lookup Configuration File	17-2
17-2. Configuring the Z39.50 Server to Send Both MARC-8 and UTF-8 Encoded Records	17-13

## Procedures

---

<b>30</b>	<b>WebAdmin</b>	
30-1.	Creating a New Login/Password for the First Time	30-3
30-2.	Adding an Additional Login/Password to the Existing File	30-4
30-3.	Deleting a Login/Password for an Existing User	30-4
30-4.	Modifying a Login/Password for an Existing User	30-4
30-5.	Adding the WebAdmin Authorization Section to the httpd.conf File	30-5
30-6.	Logging into WebAdmin	30-6
30-7.	Running Acqjob 5 Using WebAdmin	30-9
30-8.	Running Bulk Import Using WebAdmin	30-12
30-9.	Running MARC Record Bulk Export Using WebAdmin	30-17
30-10.	Running Global Headings Change Jobs Using WebAdmin	30-22
30-11.	Forgiving fines by patron ID (circjob 40)	30-24
30-12.	Forgiving fines by date created (circjob 41)	30-25
30-13.	Forgiving fines by patron group and expiration date (circjob 42)	30-27
30-14.	Running Bursar Transfer using WebAdmin	30-29
30-15.	Running Retain Patron IDs Media Scheduling, Mediajob 5	30-32
30-16.	Running OPAC Reports and Processes Using WebAdmin	30-35
30-17.	Accessing Log Files on the Server Using WebAdmin	30-36
30-18.	Accessing Report Files on the Server Using Webadmin	30-37

---

**B** **UseMARCON Configuration for Use with Voyager**

B-1.	Setting Up Initialization Paths in z3950srv.ini File	B-14
------	--	------

## Procedures

---

C

### WebVoyage Patron Authentication Adapter Feature

C-1. Patron Authentication Workflow Overview

C-2

---

## About This Document

---

### Purpose

---

This document provides information regarding basic server procedures and information on many server batch jobs used in Voyager®. Additionally, search logging in Voyager, WebAdmin, and the various input and output standard interface file (SIF) formats are covered.

### Intended Audience

---

This document is intended for Voyager customers, especially those responsible for working with the server where your Voyager database resides.

### Reason for Reissue

---

This user's guide incorporates and is being reissued for the following reasons:

- Important notation added to circjob 41 and circjob 42 instructions for -L location codes. See [Running circjob 41 from the command line](#) on page 8-30 and [Running circjob 42 from the command line](#) on page 8-33.
- Corrections to [MARC Record Posting](#) on page 4-47 and [Example \[MARC POSTing\] Stanza](#) on page 4-47.
- Changes regarding UseMARCON.

Previously, the path to usemarcon.errmsgs was set in the Pscript such as Pbulkimport. With this release, the path to usemarcon.errmsgs is set in the initialization .ini file in the [Default Files] stanza. See [Table B-1](#) on [page B-4](#) and [Figure B-3](#) on [page B-5](#) for information regarding the ErrorLogFile= changes.

The UseMARCON log file was also set in the Pscript, and multiple log files were created with unique names for MARCexport, Z3950svr, and so on and placed in the reportsdir directory. With this release, a single log file called usemarcon.log is created and stored, typically, in the sbin directory. See [UseMARCON Log File](#) on [page B-5](#) regarding the usemarcon.log for more information.

- New appendix. See [Appendix C, “WebVoyage Patron Authentication Adapter Feature](#) on page [C-1](#).

## Document Summary

---

This document consists of the following chapters:

- |            |  |
|------------|--|
| Chapter 1  | <a href="#">“Getting Started”</a> This chapter discusses some of the data loads in the data conversion process.                          |
| Chapter 2  | <a href="#">“Overview of the Data Conversion Process.”</a> This chapter discusses some of the data loads in the data conversion process. |
| Chapter 3  | <a href="#">“Server Activities in the Voyager System.”</a> This chapter provides the correct procedures for some server activities.      |
| Chapter 4  | <a href="#">“Voyager Client Installation and the Voyager.ini file.”</a> This chapter covers how to install the Voyager clients.          |
| Chapter 5  | <a href="#">“Patron Extract.”</a> This chapter provides information about the patron extract program.                                    |
| Chapter 6  | <a href="#">“Patron Update.”</a> This chapter provides information about the patron update program.                                      |
| Chapter 7  | <a href="#">“Bursar Transfer System.”</a> This chapter discusses the bursar transfer program.  |
| Chapter 8  | <a href="#">“Circulation Batch Jobs.”</a> This chapter provides information on Circulation batch jobs.                                   |
| Chapter 9  | <a href="#">“Media Scheduling Batch Jobs.”</a> This chapter provides information on Media Scheduling batch jobs.                         |
| Chapter 10 | <a href="#">“Bulk Export of MARC Records.”</a> This chapter provides information about the MARC extract program.                         |
| Chapter 11 | <a href="#">“Prebulk Program.”</a> This chapter discusses the Prebulk program.   |

- 
- Chapter 12     [“Bulk Import, Replace, and Merge of MARC Records.”](#) This chapter provides information on the bulk import program.
- Chapter 13     [“Global Heading Change Jobs.”](#) This chapter covers Global Headings Change batch jobs.
- Chapter 14     [“Storage Barcode Verify \(Pstrgvfy\) Program.”](#) This chapter provides information about the storage barcode verify program.
- Chapter 15     [“Popacjob.”](#) This chapter provides information on logging in Voyager.
- Chapter 16     [“Acquisitions Batch Job - Fix Exchange Rates.”](#) This discusses the Acquisitions batch job that updates commitments of Purchase Orders that use foreign currency.
- Chapter 17     [“WebAdmin.”](#) This chapter discusses the WebAdmin utility.
- Chapter 18     [“Server-Side Configurations.”](#) This chapter describes the set up information for features that require configuration changes on the server.
- Chapter 19     [“Patron Record Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 20     [“Charge Transaction Record Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 21     [“Item Delete Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 22     [“Vendor Record Standard Interface Format.”](#) This chapter provides information on this SIF.
- Chapter 23     [“Acquisitions Notices Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 24     [“Acquisitions Reports Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 25     [“Cataloging Reports Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 26     [“Circulation Notices Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 27     [“Circulation Reports Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 28     [“Media Scheduling Notices Standard Interface File.”](#) This chapter provides information on this SIF.
- Chapter 29     [“Media Scheduling Reports Standard Interface File.”](#) This chapter provides information on this SIF.

Chapter 30	" <a href="#">Database Views</a> ." This chapter provides information on the database views, simplified ways of retrieving data from the database.
Chapter 31	" <a href="#">Oracle Stored Functions</a> ." This chapter provides information on functions that may be used in SQL queries to return fields from MARC records and assist in miscellaneous queries for Circulation and Acquisitions.
Appendix A	" <a href="#">Data Dictionary</a> ." This chapter is the data dictionary.
Appendix B	" <a href="#">UseMARCON Configuration for Use with Voyager</a> ." This chapter discusses UseMARCON.
Appendix C	" <a href="#">WebVoyage Patron Authentication Adapter Feature</a> ." This chapter discusses patron authentication.
Index	The <a href="#">Index</a> is a detailed, alphabetical cross-reference of topics about which this document contains information.

## **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 *Curator Web Client 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](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@exlibrisgroup.com](mailto:docmanager@exlibrisgroup.com)

---



## **Getting Started**

---

# **1**

<b>Introduction</b>	<b>1-1</b>
<b>Prerequisite Skills and Knowledge</b>	<b>1-1</b>
<b>Before You Begin</b>	<b>1-1</b>



---

## Getting Started

# 1

---

---

### Introduction

---

This user's guide discusses a variety of server, and client related activities. It provides step-by-step instructions for completing some server maintenance tasks, client installation, and running batch jobs on the server.

---

### Prerequisite Skills and Knowledge

---

To use this document effectively, you need knowledge of the following:

- Basic Microsoft® Interface navigation
- Basic UNIX® commands and navigation

---

### Before You Begin

---

This user's guide is primarily intended for System Administrators. To accomplish most of the tasks described in this document you must have access to your Voyager server.



---

---

## **Overview of the Data Conversion Process**

---

# **2**

<b>Introduction</b>	<b>2-1</b>
<b>Purpose of this Chapter</b>	<b>2-2</b>
<b>Bibliographic Data</b>	<b>2-2</b>
<b>Authority Data</b>	<b>2-2</b>
<b>Patron Data</b>	<b>2-3</b>
<b>Circulation Transactions</b>	<b>2-3</b>
<b>Vendor Data</b>	<b>2-4</b>



---

## Overview of the Data Conversion Process

# 2

---

### Introduction

---

The data conversion process occurs in four main steps.

1. Data extraction. Bibliographic, authority, and holdings information is extracted from the current system. This step is completed by the site.
2. Data processing. The data is processed such that the holdings and item information is made into a loadable format.
3. Test load of your data. Your institution is provided with a test load of your site's data for your review. All of this information is overwritten when the production database is installed.
4. Production load of your data. This step loads the bibliographic, authority, patron, circulation transactions, and vendor data. There is an order in which the data must be loaded. Specifically, all bibliographic data (which includes holdings, item records, and authority records) must be loaded first, before the other data types of patron and vendor data, and circulation transactions.

In addition to the data loads discussed in the following sections, sites may also load fine/fee information, orders, serial publication patterns, as well as hold and recall information.

## Purpose of this Chapter

---

This chapter provides an overview of the following types of data loads.

- Bibliographic data
- Authority data
- Patron data
- Circulation Transactions
- Vendor data

## Bibliographic Data

---

Bibliographic (and authority) data must be loaded before any other data types. If necessary, your data will be converted to the UTF-8 encoding during the load process.

Before the bibliographic data is loaded, do not define or create anything in the System Administration module. Any information present will be completely overwritten. You will not be able to recover it.

After the bibliographic data has been loaded, the **Location Code** field is populated in System Administration. This value was defined by your institution's responses to the Data Migration Questionnaire.

To add the **Location Name**, **Spine Label Name**, and **OPAC Display Name**, go in to the System Administration module and select **System> Locations**.

In addition, the **Item Type Code** is also populated in System Administration. This value was also defined by your institution's responses to the Data Migration Questionnaire.

To add the **Item Type Name**, and the **Display Name**, go in to the System Administration module and select **System> Item Types**.

## Authority Data

---

Authority data should be loaded at the same time as the bibliographic data. If necessary, your data will be converted to the UTF-8 encoding during the load process.

Before the authority data is loaded, do not define or create anything in the System Administration module. Any information present will be completely overwritten. You will not be able to recover it.

There is no information that is automatically created in any module from the loaded authority data.

## Patron Data

---

Patron data must be loaded after all bibliographic data has been loaded.

Before the patron data is loaded, do not define or create any of the following:

- Patron Groups in System Administration
- Statistical Categories in System Administration
- Patron Records in Circulation

After the patron data has been loaded, the **Patron Group Code** field is populated in System Administration.

To add the **Patron Group Name**, and the **Display Name**, go in to the System Administration module and select **Circulation> Patron Groups**.

In addition, the **Patron** and **Item Statistical Category Code** fields are populated in System Administration.

To add the **Patron Category Name**, and the **Item Category Name**, go in to the System Administration module and select **System> Statistical Categories> Patron or Item tab**.

Patron Records are added to the system as well. The patron data is provided to Ex Libris in API format or in a delimited data file. For information regarding the specific file format see [Patron Update on page 6-1](#).

## Circulation Transactions

---

Circulation transaction data must be loaded after all bibliographic and patron data has been loaded.

Before the circulation transaction data is loaded, circulation policy definitions and circulation matrix definitions must be created in the System Administration module.

To create Circulation Policy Definitions for your institution, in System Administration, select **Circulation> Policy Definitions**. Be sure to define at least one circulation happening location during this process. See the *Voyager System Administration User's Guide*, [Circulation Locations \(Circulation Happening Locations\)](#) for more information. Transactions will not load if there is not a circulation happening location defined.

The transaction loader uses the defined circulation happening locations for every transaction it adds to the database.

There can be multiple circulation locations in any given policy group. Given this, when doing transaction loads, the site should:

1. Define Circulation policy groups so that there is at least one Circulation happening location for each group. This Circulation location should be the one that is correct for the transactions that are being loaded.
2. After the transactions have been loaded, the site can define additional Circulation locations for the Circulation policy groups.

Before the circulation transactions data is loaded, you must also create Circulation Policy Matrix Definitions. To do this in System Administration, select **Circulation> Policy Definitions> Matrix**. You must create these Circulation Definitions in order for Voyager to determine appropriate actions for specific patrons and item types. See the *Voyager System Administration User's Guide*, [Circulation Policy Matrix](#), for more information.



#### **IMPORTANT:**

*Before the circulation transaction data is loaded, do not circulate any items using Voyager.*

Also note that item and patron barcode numbers in the circulation transaction records must match the barcode numbers in the previously loaded item and patron records.

This load creates transactions in the circulation module.

---

## **Vendor Data**

---

Vendor data must be loaded after all bibliographic data has been loaded. Before the vendor data is loaded, do not create any vendor records in Acquisitions.

This vendor file load is an optional load. Doing the load, as opposed to manually entering the information in the System Administration module, depends on the number of vendors. The vendor file contains current address and contact information of vendors. It can include vendor types as well.

**NOTE:**

At least one vendor type must be set up in the System Administration module, if they were not included in the vendor file load.

After the vendor data has been loaded, add the appropriate vendor type definitions in the System Administration module. To define these values, select **System> Vendor Types**.

Also, you must define the Voyager base currency to match the default currency specified in the vendor data. To define a base currency, select, in the System Administration module, **System> Base Currency**. Only one currency is loaded, if your site uses more than one currency, add the appropriate additional currencies after the load using the Acquisitions module by selecting **Functions> Currency Maintenance**.

This load creates vendor address records in the Acquisitions module.



---

## **Server Activities in the Voyager System**

---

# **3**

<b>Introduction</b>	<b>3-1</b>
<b>Purpose of This Chapter</b>	<b>3-1</b>
<b>Services, Scripts, Service Dependencies and Rationale</b>	<b>3-2</b>
• Scripts	3-2
• Service Dependencies	3-3
<b>Solaris and AIX Servers</b>	<b>3-4</b>
• Starting the Server	3-4
• Rebooting or Shutting Down the Server	3-4
• Stopping and Starting Oracle Services	3-6
• Stopping and Starting Voyager	3-8
• Stopping and Starting the Apache Web Server	3-10
• Database Backup - Solaris Servers	3-11
Backup Process	3-12
Determining the Backup Device Driver	3-12
Identifying your Data Filesystem/Volume	3-14
Reviewing Cron Backup Results	3-16
<b>Windows Server 2003</b>	<b>3-16</b>
• Starting the Server	3-16
• Rebooting or Shutting Down the Server	3-17
• Stopping and Starting Oracle Services	3-18
• Stopping and Starting Voyager	3-19
• Stopping and Starting the Apache Web Server	3-20

• Database Backup - Windows Server 2003	3-21
Automating the Backup	3-25
<b>Restoring from Backup</b>	<b>3-27</b>
<b>Crons Set Up at Installation</b>	<b>3-27</b>

### **Introduction**

---

Voyager is client server software where the database resides on a server and the clients that allow access to the database reside on user's computers.

Voyager® runs on Solaris®, AIX®, or Windows Server® 2003.



#### **IMPORTANT:**

*It is assumed that the operator has basic knowledge of UNIX. If there are terms or procedures that are unfamiliar, call Ex Libris Customer Support.*

### **Purpose of This Chapter**

---

There are some basic server maintenance activities necessary for efficient functioning of Voyager.

This chapter discusses the following server activities:

- Services, scripts, dependencies and rationale
- Server procedures
  - Oracle Database Management Server (DBMS)
  - Voyager
  - Tomcat

- Apache™ web server procedures
- Backups

Due to different server platforms, this chapter is divided into a Solaris / AIX section and a Windows Server 2003 section. Within each section the procedures are discussed.

This information is specific for sites with a single server. Sites with multiple servers should contact Ex Libris Customer Support for the correct server procedures.

## **Services, Scripts, Service Dependencies and Rationale**

---

The Voyager system uses the following services:

- Oracle - this is the actual database, the “back-end”, the place where the data is stored.
- Voyager - this is the brains, the way to organize, retrieve, and add to the database, the “front-end”.
- Tomcat - this service enables specific activities to occur. Voyager may run multiple instances of Tomcat. For example, one instance of Tomcat runs Patron Self Registration (PSR) if enabled, and another instance runs Link Finder Plus (LFP)
- Apache - this is a web server, its job is to provide, “serve”, web pages. It runs WebVoyage.

### **Scripts**

---

There are several scripts provided by Ex Libris that automatically run various processes on the Voyager server to maintain efficient functioning.

The scripts are:

- dbora
- voyager
- httpd2

To run a start up script use the start parameter. To run a shut down script use the stop parameter.

For example:

- **dbora start** to start Oracle or **dbora stop** to shut down Oracle
- **voyager start** to start Voyager or **voyager stop** to shut down Voyager
- **httpd2 start** to start the Apache web server or **Apache stop** to shut down the Apache web server

There is also a backup script, see [Database Backup - Solaris Servers](#) on page 3-11. Additionally multiple Crons are provided to run some batch jobs, see [Crons Set Up at Installation](#) on page 3-27.

## Service Dependencies

Dependencies occur when running a particular script on the server effects multiple services.

There are service dependencies when running some of the shut down scripts.



### **IMPORTANT:**

*There are no dependencies among the start up scripts. Therefore, if starting the machine manually each script must be run.*

[Table 3-1](#) describes the service dependencies between the shut down scripts and the rationale for these dependencies.

**Table 3-1. Service Dependencies**

When Running Script...	This service shuts down...	and these dependent services shut down...	Rationale
dbora stop	Oracle	Voyager shuts down.  Tomcat instances shut down.	If Oracle is shut down, Voyager and Tomcat cannot work, therefore, they automatically shut down.  However, Apache is not dependent, and keeps running because it may be a web server for something other than Web-Voyage.  This allows running Oracle without running Voyager.

**Table 3-1. Service Dependencies**

<b>When Running Script...</b>	<b>This service shuts down...</b>	<b>and these dependent services shut down...</b>	<b>Rationale</b>
voyager stop	Voyager	Tomcat instance for Patron Self Registration (PSR) shuts down.	If Voyager is shut down, then the Tomcat instance used for Patron Self Registration is also shut down because there is no access to Voyager.  <b>NOTE:</b> Oracle, Apache, and the Tomcat instance used for LFP are not stopped.
httpd2 stop	Apache web server	No dependencies	Operators have the ability to shut down the web server without affecting Oracle, Voyager, and Tomcat.

## Solaris and AIX Servers

---

The following sections contain the server procedures for Sun Solaris® and IBM® AIX® servers.

### Starting the Server

---

There is no specific procedure for starting the server. By powering on the server, start up scripts (provided by Ex Libris) run, which start Oracle, Voyager, Tomcat and Apache.

### Rebooting or Shutting Down the Server

---

Ex Libris recommends rebooting your system at least once a week. Other occasions when you might need to shutdown your server are adding new equipment, moving the server hardware, or installing operating system patches.

The procedure for rebooting or shutting down a Solaris server is shown in [Procedure 3-1, Rebooting or Shutting Down a Solaris Server](#).



### Procedure 3-1. Rebooting or Shutting Down a Solaris Server

---

Use the following to reboot or shut down your Solaris server safely.

1. Log in as root.



**CAUTION:**

*Logging in as root gives the operator full access to the servers operating system and all of the software installed on the server. Use extreme caution when using this login.*

2. Two options are available:

- a. To reboot safely, that is, to shut down and restart the server, enter the following command:

**init 6**

- b. To remove power from the server safely, turning the server off, enter the following command:

**init 5**

Result: The system reboots or powers off.

**NOTE:**

Powering off in this manner automatically runs all shutdown scripts. Rebooting automatically runs all shutdown scripts, except power off. It then resets the machine, running all of the start-up scripts.

---

The procedure for rebooting or shutting down an AIX server is shown in [Procedure 3-2, Rebooting or Shutting Down an AIX Server](#).



### Procedure 3-2. Rebooting or Shutting Down an AIX Server

---

Use the following to reboot or shut down your AIX server safely.

1. Log in as root.

**⚠ CAUTION:**

*Logging in as root gives the operator full access to the servers operating system and all of the software installed on the server. Use extreme caution when using this login.*

2. Two options are available:

- a. To reboot safely, that is, to shut down and restart the server, enter the following command:

`shutdown -r now`

- b. To remove power from the server safely, turning the server off, enter the following command:

`shutdown -h now`

Result: The system reboots or powers off.

**NOTE:**

Powering off in this manner automatically runs all shutdown scripts. Rebooting automatically runs all shutdown scripts, except power off. It then resets the machine, running all of the start-up scripts.

---

### Stopping and Starting Oracle Services

---

Normally, you won't need to stop the Oracle services, however there will be some occasions, for example with a manual backup or to patch the operating system, where you may need to stop Oracle.

The procedure for stopping Oracle is shown in [Procedure 3-3, Stopping Oracle Services on a Solaris or AIX Server](#).



### **Procedure 3-3. Stopping Oracle Services on a Solaris or AIX Server**

---

Use the following to stop Oracle.

1. Log in as oracle.
2. Enter the following command to switch user to root:

```
su - root
```

3. For servers running Solaris 10, enter the following commands:

```
svcadm disable -t EISIdbora  
svcadm disable -t EISIolisten
```

For all other versions, enter the following command:

```
/etc/init.d/dbora stop
```

Result: Oracle and dependent services stop (see [Service Dependencies on page 3-3](#)).

---

The procedure for starting Oracle is shown in [Procedure 3-4, Starting Oracle Services on a Solaris and AIX Server](#).



### **Procedure 3-4. Starting Oracle Services on a Solaris and AIX Server**

---

Use the following to start Oracle.

1. Log in as oracle.
2. Enter the following command to switch user to root:

```
su - root
```

3. For servers running Solaris 10, enter the following command:

```
svcadm enable -r EISIdbora
```

For all other versions, enter the following command:

```
/etc/init.d/dbora start
```

Result: Oracle services start.

**NOTE:**

Only Oracle starts, there are no dependencies.

---

## **Stopping and Starting Voyager**

---

Ex Libris suggests stopping Voyager once a day to kill any stale Voyager processes. This happens automatically when the nightly backup occurs since the backup stops Voyager (presuming you are running the Ex Libris supplied backup script.)

The procedure for stopping Voyager is shown in [Procedure 3-5, Stopping Voyager on a Solaris Server](#).



### **Procedure 3-5. Stopping Voyager on a Solaris Server**

---

Use the following to stop Voyager.

1. Log in as root.
2. Enter the following command:

```
/etc/init.d/voyager stop
```

Result: Voyager and its dependent services stop (see [Service Dependencies on page 3-3](#)). For servers running Solaris 10, the system calls inetadm to disable Voyager ports.

---

The procedure for starting Voyager is shown in [Procedure 3-6, Starting Voyager on a Solaris Server](#).



### **Procedure 3-6. Starting Voyager on a Solaris Server**

---

Use the following to start Voyager.

1. Log in as root.
2. Enter the following command:

```
/etc/init.d/voyager start
```

Result: Voyager services start.

**NOTE:**

The Voyager service starts. For servers running Solaris 10, the system calls `inetadm` to enable Voyager ports.

---

The procedure for stopping Voyager is shown in [Procedure 3-7, Stopping Voyager on an AIX Server](#).



### Procedure 3-7. Stopping Voyager on an AIX Server

---

Use the following to stop Voyager.

1. Log in as root.
2. Enter the following command:

```
/etc/init.d/voyager stop
```

Result: Voyager stops, and its dependent services stop, see [Service Dependencies on page 3-3](#).

---

The procedure for starting Voyager is shown in [Procedure 3-8, Starting Voyager on an AIX Server](#).



### Procedure 3-8. Starting Voyager on an AIX Server

---

Use the following to start Voyager.

1. Log in as root.
2. Enter the following command:

```
/etc/init.d/voyager start
```

**NOTE:**

Only Voyager starts, there are no dependencies.

---

## **Stopping and Starting the Apache Web Server**

---

Ex Libris installs the Apache web server for the purpose of running WebVoyáge, the web-based OPAC.

The procedure for stopping the Apache web server is shown in [Procedure 3-9, Stopping the Apache Web Server](#).



### **Procedure 3-9. Stopping the Apache Web Server**

---

Use the following to stop the Apache web server.

1. Log in as oracle.
2. Enter the following command to switch user to root:

`su - root`

3. Enter the following command:

`/etc/init.d/httpd2 stop`

Result: The Apache web server stops.

---

The procedure for starting the Apache web server is shown in [Procedure 3-10, Starting the Apache Web Server](#).



### **Procedure 3-10. Starting the Apache Web Server**

---

Use the following to start the Apache web server.

1. Log in as oracle.
2. Enter the following command to switch user to root:

`su - root`

3. Enter the following command:

```
/etc/init.d/httpd2 start
```

Result: The Apache web server starts.

---

## **Database Backup - Solaris Servers**

---

Ex Libris requires sites to complete a backup of their data once each day. The administrator should perform the backup during off hours while the system is not operational (cold backup).

**NOTE:**

Backups that run while the system is still operational (hot backups) are generally not recommended because they require additional hardware and are more complex than a cold backup. Sites that want to run non-supported backups should contact Ex Libris Customer Support for approval.

Sites can run a backup of their database either manually by running the /m1/utility/dailybackup script or automatically by using the cron provided by Ex Libris.

At installation of your Voyager system, a cron is set up to perform the standard daily backup. This cron specifies the time that the backup should run and the appropriate commands to complete the backup. If you need assistance editing (or creating) a backup cron, contact Ex Libris Customer Support.

After the system runs the cron, the results are sent in an email to root. For more information on verifying the results of a cron, see [Reviewing Cron Backup Results](#) on [page 3-16](#).

Before you begin performing backups, verify that the following devices are ready:

- Backup device driver (see [Determining the Backup Device Driver](#) on [page 3-12](#))
- Data volume (see [Identifying your Data Filesystem/Volume](#) on [page 3-14](#))

**NOTE:**

AIX sites please contact Ex Libris customer support for assistance with backups. The discussion and procedures here are for the Solaris Operating System.

## Backup Process

The following activities are performed by the system during a backup:

- Shutdown Apache web server, Voyager, and Oracle instances.
- Start snapshots.
- Start Apache web server, Voyager and Oracle instances.
- Dump the data to tape.
- End snapshots.



### IMPORTANT:

*Your institution is responsible for safeguarding its own data. If database corruption occurs, restoring the data from a backup will insure your data integrity and decrease downtime. If you require Ex Libris to reload your data because of insufficient backups, you will be charged for that service. If you have any problem with backing up your system using ufsdump, we will be able to provide support. We cannot provide assistance for any other means of backup.*

## Determining the Backup Device Driver

You must determine your backup device driver.

Most tape devices can record at multiple densities, such as low, medium, and high. High density is preferable because it allows for the quick backup of data, as well as more data written to the tape.



### Procedure 3-11. Determining the Backup Device Driver

---

Use the following to determine the backup device.

1. Insert a tape of the appropriate size into your backup unit.
2. Log in as root.
3. Enter the following command:

```
ls /dev/rmt/*
```

Result: This lists all tape device drivers for which your server is configured. Your list will look similar to this:

```
/dev/rmt/0  
/dev/rmt/0b  
/dev/rmt/0bn  
/dev/rmt/0u  
/dev/rmt/0un
```

The digit (0, zero) is the driver number.

The character n represents the non-rewinding driver. This is used when backing up multiple filesystems sequentially.

The characters following the digit can be l, m, h, c, and u. They correspond to the density of the tape drive listed from lowest (l) to highest (u).

The character b represents Berkeley-style tape positioning and should not be used.

4. Write down the driver number with the highest density mode available. In the example above, 0u (zero u) is the driver number and highest density mode available.
5. You will need to determine the density at which your system will read/write tapes by testing the drivers.

Test the high density driver first by entering:

```
mt -f /dev/rmt/## status
```

Where the ## is the driver number/character combination for your highest density driver.

Therefore, following the example above, the command would be

```
mt -f /dev/rmt/0u status
```

Result: The system should respond by identifying the type of tape backup unit and if the unit is available ([Figure 3-1](#)). If available then that is the density to which your system can read/write tapes and the driver number/character combination to use in your ufsdump backup command (manual backups). See [Database Backup - Solaris Servers](#) on [page 3-11](#).

```
#mt -f /dev/rmt/0u status
Quantum DLT8000 tape drive:
    sense key(0x6)=Unit Attention residual=0 retries=0
    file no=0 block no=0
#
```

**Figure 3-1. Example Output**

If you get a message no tape loaded or drive offline, that means you cannot read/write tapes at that density. Try the same test with a medium density driver, and so on until you find a density where you can read/write.

6. Make a note of which device driver worked. You will need it if you run the usfdump backup command manually.

### Identifying your Data Filesystem/Volume

You will need to identify in which filesystem(s) your data resides so that you can backup the correct filesystem(s).

For Voyager 2006.2 and earlier installations, there will be an `/m1` filesystem (which is where Voyager resides) and an `/oracle` filesystem.

For Voyager 2006.5 and later installations, there will be an `/m1` filesystem (where Voyager resides), an `/oracle` filesystem, an `/oracle/oradata` filesystem, and possibly more depending on the configuration.

Earlier installations of Voyager may have the `/m1` filesystem only.



#### **IMPORTANT:**

*If you have more than one filesystem, such as `/m1` and `/oracle`, backups should be performed on each filesystem (such as `/m1` and `/oracle`). Otherwise, perform the backup on just the `/m1` filesystem.*



### Procedure 3-12. Identifying your Data Filesystem

Use the following to determine the filesystem structure and hence, which backup procedures your site should follow.

1. Type **df -k** and press **enter**.

Result: The output should look similar to the values in [Table 3-2](#). These values illustrate the filesystems on your server, along with the amount of space available on them.

**Table 3-2. Filesystems on the server**

Filesystem	kbytes	used	avail	capacity	Mounted on
/dev/dsk/c0t3d0s0	620710	267758	290882	48%	/
/proc	0	0	0	0%	/proc
fd	0	0	0	0%	/dev/fd
/dev/dsk/c0t3d0s7	96455	40190	46625	47%	/export/home
/dev/md/dsk/d0	16684646	9558112	5458074	64%	/m1
/dev/md/dsk/d1	17654241	12912680	5241561	70%	/oracle
swap	304044	8	304036	1%	/tmp

2. From the listing in [Table 3-2](#) one can see that this system has both an `/oracle` and `/m1` filesystem.

3. If `/m1` is not listed, enter the following command:

```
ls /
```

Result: You should now see `/m1` and possibly `/oracle` as well. If `/m1` is not present, please call Ex Libris Customer Support for assistance.

## Reviewing Cron Backup Results

After the system performs a scheduled backup, it sends an email containing the results of the backup to root. This email contains basic information and error messages that may occur during the backup.



### Procedure 3-13. Reviewing the Email Generated by the Cron Backup

---

Use the following to access the e-mail message produced when the croned backup runs.

1. Log in as root.
2. Enter the following command:  
`mailx`
3. Select the corresponding e-mail number for the backup output.
4. Review the backup output message.

**NOTE:**

If you see messages that include the words *abort* or *failure*, this *could* indicate a problem. Call Ex Libris Customer Support during regular business hours to discuss this with a technician.

5. Type **q** and press **enter** to exit the backup output message.
- 

## Windows Server 2003

---

The following sections are the procedures for Windows Server 2003.

### Starting the Server

---

There is no specific procedure for starting the server. When powering on the server, it automatically runs the start up scripts (provided by Ex Libris) and starts Oracle, Voyager, and Apache.

**NOTE:**

To perform the Voyager and Oracle tasks on a Windows Server 2003 the operator **must** log in as an Administrator, Voyager, or another ID that is a member of the Local Administrators group.

### **Rebooting or Shutting Down the Server**

---

The procedure for rebooting or shutting down a Windows Server 2003 is shown in [Procedure 3-14, Rebooting or Shutting Down](#).



#### **Procedure 3-14. Rebooting or Shutting Down**

---

Use the following to reboot or shut down a Windows Server 2003.

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.
2. Use the following to stop the Apache service:
  - a. open the **KornShell** Window
  - b. enter **httpd2 stop**

OR

stop the the Apache service from the **Services** Window.
3. Use the following to stop Voyager services:
  - a. open the **KornShell** Window
  - b. enter **voyager stop**
4. Use the following to stop the Oracle Instance and dismount the database LIBR:
  - a. open the **KornShell** Window
  - b. enter **dbora stop**
5. To
  - a. reboot, from the **Start** menu select **Shut Down**, then in the **Shut Down Windows** window, select **Restart**.
  - b. shutdown, from the **Start** menu select **Shut Down**, then in the **Shut Down Windows** window, select **Shut down**.

Result: The server reboots or shuts down.

---

## **Stopping and Starting Oracle Services**

---

The procedure for stopping Oracle is shown in [Procedure 3-15, Stopping Oracle Services](#).



### **Procedure 3-15. Stopping Oracle Services**

---

Use the following to stop Oracle services:

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.
2. Use the following to stop Oracle services:
  - a. open **KornShell** window
  - b. enter **dbora stop**

OR

Stop the following services from the **Services** Window:

ORA920\_HOMEAgent  
ORA920\_HOMEDataGatherer  
ORA920\_HOMETNSListener  
OracleServiceVGER

To do this, from the **Start> Settings> Control Panel> Administrative Tools> Services** window, select the service and then click the **stop** button in the toolbar.

Result: Oracle services stop.

---

The procedure for starting Oracle is shown in [Procedure 3-16, Starting Oracle Services](#).



### Procedure 3-16. Starting Oracle Services

---

Use the following to start Oracle services:

1. Log in as an Administrator, Voyager, any ID that is a member of the Local Administrators group.
2. Use the following to start Oracle services:
  - a. open the **KornShell** Window
  - b. enter **dbora start**

OR

Start the following services from the **Services** Windows:

ORA920\_HOMEAgent  
ORA920\_HOMEDataGatherer  
ORA920\_HOMETNSListener  
OracleServiceVGER

To do this, from the **Start> Settings> Control Panel> Administrative Tools> Services** window, select the service and then click the **start** button in the toolbar.

Result: Oracle services start.

---

### Stopping and Starting Voyager

---

The procedure for stopping Voyager is shown in [Procedure 3-17, Stopping Voyager](#).



### Procedure 3-17. Stopping Voyager

---

Use the following to stop Voyager:

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.

2. Use the following to stop Voyager services:

- a. open the **KornShell** Window
- b. enter **voyager stop**

Result: Voyager stops.

---

The procedure for starting Voyager is shown in [Procedure 3-18, Starting Voyager](#).



### **Procedure 3-18. Starting Voyager**

---

Use the following to start Voyager:

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.
2. Use the following to start Voyager services:
  - a. open the **KornShell** Window
  - b. enter **voyager start**

Result: Voyager starts.

---

## **Stopping and Starting the Apache Web Server**

---

The procedure for stopping the Apache web server is shown in [Procedure 3-19, Stopping the Apache Web Server](#).



### **Procedure 3-19. Stopping the Apache Web Server**

---

Use the following to stop the Apache web server:

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.
2. Use the following to stop Apache service:
  - a. open the **KornShell** Window
  - b. enter **httpd2 stop**

OR

stop the Apache service from the **Services** Window.

Result: The Apache web server stops.

---

The procedure for starting the Apache web server is shown in [Procedure 3-20, Starting the Apache Web Server](#).



#### Procedure 3-20. Starting the Apache Web Server

---

Use the following to start the Apache web server:

1. Log in as an Administrator, Voyager, or any ID that is a member of the Local Administrators group.
2. Use the following to start Apache service:
  - a. open the **KornShell** Window
  - b. enter `httpd2 start`

OR

start the Apache service from the **Services** Window.

Result: The Apache web server starts.

---

#### Database Backup - Windows Server 2003

---

Ex Libris requires sites to complete a backup of their data once each day.

Ex Libris provides a standard backup script located in  
`c:\etc\init.d\backup.ksh`.

The standard backup script provided by Ex Libris is shown in [Figure 3-2](#).

```

#!/bin/sh
#####
# Standard Windows Server 2003 Backup for Voyager 200x.X.X
#####
DB_TO_BACKUP="80"
BACKUP_LOCATION="d:/backup"
EXPORT_BACKUP="Y"
TAPE_BACKUP="Y"
TAPE_DEVICE="4mm DDS"
RESTART="Y"
#####
# End of Parameters
#####
mkdir d:/backup 2> NUL

        print "Starting Process for `date`" >
$BACKUP_LOCATION/backup.log
        print -n "Stoping Voyager Services...." >>
$BACKUP_LOCATION/backup.log
c:/etc/init.d/httpd stop
c:/etc/init.d/voyager stop
        print "OK" >> $BACKUP_LOCATION/backup.log
if [[ "$EXPORT_BACKUP" = "Y" ]]
then
        print -n "Beginning Export...." >> $BACKUP_LOCATION/
backup.log

        USER_PASS=`grep "export USERPASS=" d:/voyager/
$DB_TO_BACKUP/ini/voyager.env | awk -F/ '{print $2}'`'
        print "$DB_TO_BACKUP/$USER_PASS"
        exp $DB_TO_BACKUP/$USER_PASS FILE=$BACKUP_LOCATION/
temp.exp COMPRESS=N Owner=$DB_TO_BACKUP BUFFER=1024000
Log=d:/backup/Export.log
        EXP_RES="$?"
if [[ "$EXP_RES" = "0" ]]
then
        rm $BACKUP_LOCATION/${DB_TO_BACKUP}3_backup.exp
        mv $BACKUP_LOCATION/${DB_TO_BACKUP}2_backup.exp
$BACKUP_LOCATION/${DB_TO_BACKUP}3_backup.exp
        mv $BACKUP_LOCATION/${DB_TO_BACKUP}1_backup.exp
$BACKUP_LOCATION/${DB_TO_BACKUP}2_backup.exp
        mv $BACKUP_LOCATION/temp.exp $BACKUP_LOCATION/

```

```

${DB_TO_BACKUP}1_backup.exp
    print "OK" >> $BACKUP_LOCATION/backup.log
else
    print "FAILURE" >> $BACKUP_LOCATION/backup.log
fi
else
    print -n "No Export Requested...." >>
$BACKUP_LOCATION/backup.log
    print "OK" >> $BACKUP_LOCATION/backup.log
fi
if [[ "$TAPE_BACKUP" = "Y" ]]
then
    c:/etc/init.d/dbora stop
    print -n "Stoping Oracle Services...." >>
$BACKUP_LOCATION/backup.log
    print "OK" >> $BACKUP_LOCATION/backup.log
    print -n "Beginning Tape Backup...." >>
$BACKUP_LOCATION/backup.log
    ntbackup backup c:\\ d:\\ /D "%ComputerName%" /M
normal /F "d:\backup\backupkevin" /UM
    C_BACK_RES="$?"
    if [[ "$C_BACK_RES" = "0" ]]

    then
        print "OK" >> $BACKUP_LOCATION/backup.log
    else
        print "FAILURE (C & D Drive:$C_BACK_RES)" >>
$BACKUP_LOCATION/backup.log
    fi
else
    print -n "No Tape Backup Requested...." >>
$BACKUP_LOCATION/backup.log
    print "OK" >> $BACKUP_LOCATION/backup.log
fi
if [[ "$RESTART" != "Y" ]]
then
    print -n "Starting Voyager Services...." >>
$BACKUP_LOCATION/backup.log

```

```

c:/etc/init.d/voyager start
c:/etc/init.d/httpd start
print "OK" >> $BACKUP_LOCATION/backup.log
print -n "No Reboot Requested...." >>
$BACKUP_LOCATION/backup.log
print "OK" >> $BACKUP_LOCATION/backup.log

else
    print -n "Reboot Requested...." >> $BACKUP_LOCATION/
backup.log
    print "OK" >> $BACKUP_LOCATION/backup.log
    print -n "Attempting to Reboot..." >>
$BACKUP_LOCATION/backup.log
    echo y | c:\\winnt\\shutdown.exe /L /R /Y /C /T:05
    print "OK" >> $BACKUP_LOCATION/backup.log

fi

print -n "Done" >> $BACKUP_LOCATION/backup.log

```

**Figure 3-2. Standard Backup Script for Windows Server 2003s**

This script needs some customizing of the following variables found at the beginning of the script.

The procedure for customizing the backup script is shown in [Procedure 3-21, Customizing the Backup Script](#).



### **Procedure 3-21. Customizing the Backup Script**

To customize the script, set the variables accordingly. You can edit this script using Notepad®.

DB\_TO\_BACKUP=xxxdb  
where xxxdb is your production database name.

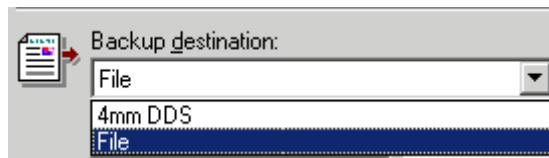
BACKUP\_LOCATION=d:/backup  
don't change this location.

EXPORT\_BACKUP=Y  
you must export the database.

TAPE\_BACKUP=Y  
if you are using the local tape device then Y otherwise N.

TAPE\_DEVICE=enter the name of the device

1. To find out the device name on your system, from the Windows **Start menu**> **Run** enter **ntbackup** in the at **Open** field.
2. At the **Backup - [Untitled]** page, click the **Backup** tab.
3. At the **Backup destination** field, press the down arrow and note the device name. [Figure 3-3](#) shows the device as 4mm DDS. The other device could be DLT.



**Figure 3-3. Backup Destination Field**

**RESTART=Y** Use Y if you are using the local tape device or want to reboot your server, otherwise use N, if you do not want to reboot your server.

### Automating the Backup

The procedure for automating the backup is shown in [Procedure 3-22, Automating the Backup](#).



### Procedure 3-22. Automating the Backup

Use the following to automate the backup.

1. Log in as Administrator, Voyager, or any ID that is a member of the Local Administrators group.
  2. Open **Windows Explorer**> **My Computer**> **Control panel**> **Scheduled tasks**.
  3. Double-click the **Add Scheduled Task** Icon.
- Result: The **Scheduled Task Wizard** windows opens.
4. Click the **Next** button.
  5. At the **Scheduled Task Wizard** window, from the **Click the program you want Windows to run** list

- a. select the **KornShell** application
  - b. click the **Next** button.
6. At the **Type a name for this task** field
    - a. enter **Daily Backup** (not case sensitive)
    - b. click the **Weekly** radio button
    - c. click the **Next** button.
  7. At the **Select the time and day you want this task to start** field
    - a. enter the start time at which you want the backup to begin running in the **Start time** field
    - b. in the **Every** field leave the default of 1
    - c. select AM or PM as appropriate
    - d. select the box corresponding to the day(s) you want the backup to run. Ex Libris suggests doing a nightly backup, not including weekends. Therefore you would select Monday, Tuesday, Wednesday, Thursday, and Friday.
    - e. click the **Next** button.
  8. At the **Enter user name**, **Enter the password**, and **Confirm password** fields, enter the user name (if not already populated), the password, and the password respectively, and click the **Next** button.

**NOTE:**

The default user name is the name of the operator currently logged in.

9. At the **Schedule Task Wizard** window, select the **Open advanced properties for this task when I click Finish** check box and click the **Finish** button.
10. At the **Daily Backup** window, in the **Run** field, at the end of the command line add **c:/etc/init.d/backup.ksh** to the end of the command line.

The full command line must be

**c:\WINNT\system32\cmd.exe /c %ROOTDIR%\bin\ncenv.cmd 6 sh && %ROOTDIR%\mksnt\sh.exe -L c:/etc/init.d/backup.ksh**

11. Click the **Settings** tab, at the **Stop the task if it runs for** field change the default of 72 to 4 hours and click the **OK** button.

Result: The **Set Account Information** window displays.

12. Type the password in the **password** and **confirm password** fields, click the **OK** button.

Result: In the **Schedule Task** window the daily backup is now listed. The backup is automated.

---

## **Restoring from Backup**

---

If you need to restore your data from a backup contact Ex Libris Customer Support. For contact information, see [http://www.exlibrisgroup.com/support\\_center.htm](http://www.exlibrisgroup.com/support_center.htm).

## **Crons Set Up at Installation**

---

In addition to the backup cron provided at installation of your Voyager system, additional crons are set up to perform the following batch jobs daily:

- Acquisitions batch job 1, Order Claim/Cancel processing
- Circulation batch job 12, All Daily jobs
- Circulation batch job 24, Transaction Exceptions

See the *Voyager Reporter User's Guide*, [Order Claim/Cancel Processing \(Acqjob 1\)](#), [All Daily Jobs \(Circjob 12\)](#), and [Transaction Exceptions \(Circjob 24\)](#) for additional information about these jobs.

**NOTE:**

For any cron to run, both Oracle and Voyager must be up and running.



---

## Voyager Client Installation and the Voyager.ini file

# 4

---

<b>Introduction</b>	<b>4-1</b>
<b>Purpose of This Chapter</b>	<b>4-1</b>
<b>Prerequisites</b>	<b>4-2</b>
<b>FTP Clients from Server to PC</b>	<b>4-2</b>
<b>Installing Voyager</b>	<b>4-5</b>
<b>Installing Media Scheduling</b>	<b>4-13</b>
<b>Modifying (Adding or Deleting Clients) an Installation</b>	<b>4-13</b>
<b>Repairing an Installation of Voyager Clients</b>	<b>4-18</b>
<b>Uninstalling the Voyager Clients</b>	<b>4-20</b>
<b>The Voyager.ini File on the PC</b>	<b>4-23</b>
• Client Module Stanzas	4-26
• [GlobalLog] Stanza	4-27
• [E-mail] Stanza	4-28
• [MARC Posting] Stanza	4-28
• [SearchURI] Stanza	4-29
Adding a Button to the Search Dialog Box	4-29
• [Upgrade] Stanza	4-30
What Happens When New Software is Detected?	4-32
Circulation - Offline Option	4-41
• Circulation Charge Timeout	4-42
• Single Client Login	4-42

• Encryption	4-43
• Staff Client Re-Sort	4-44
Voyager-Aware Re-Sorts	4-45
Quick Sorts	4-46
• MARC Record Posting	4-47
Example [MARC POSTing] Stanza	4-47
Selecting Send Record To	4-49
<b>Additional Files Installed with the Client Installation</b>	<b>4-50</b>

---

## Voyager Client Installation and the Voyager.ini file

# 4

---

### Introduction

---

Voyager is client server software where the database resides on a server and the clients that allow access to the database reside on a user's computer.

---

### Purpose of This Chapter

---

This section provides information and procedures regarding the following:

- Basic File Transfer Protocol (FTP) commands to access the Voyager client file from your Voyager server.
- Installing Voyager clients on a user's computer.
- Modifying an installation.
- Repairing an installation.
- The `voyager.ini` file to include the following:
  - Voyager Client Link.
  - Single Client Login.
  - Circulation Charge Time-out.
  - Encryption.
  - MARC record POSTing.
  - Search URI link.

## Prerequisites

---

For Voyager client prerequisites, see SupportWeb for the most updated list of supported PC operating system and hardware information located in the following path:

Customer Support > Technical Information > Other Resources > Supported Browers, PC Operating Systems & Hardware Requirements for Voyager®

or

[https://support.endinfosys.com/cust/voy/cust\\_supp/tech/clientplatform.html](https://support.endinfosys.com/cust/voy/cust_supp/tech/clientplatform.html)

## FTP Clients from Server to PC

---

There are several ways by which one can move/copy Voyager client files. One common method is to use FTP (File Transfer Protocol). It is a way to move files between a server and a PC. However, check with your system administrator to confirm the method your site uses to handle accessing client files and their installation.

The focus of this section is to describe the installation processing using the FTP method.

The Voyager client installation executable files are located in `/m1/voyager/clients`. The following files are stored in that directory:

- `VoyagerInstall.exe`
- `VoyagerInstall.msi` (Windows Installer Package version)
- `VoyagerMedia.exe`
- `VoyagerMedia.msi` (Windows Installer Package version)



### IMPORTANT:

*You must have the Voyager server user ID and password access to these files. If you do not have this information, see your server administrator.*

### NOTE:

For Windows 2000 servers, FTP is limited to the `/incoming` directory. Therefore, you need to move the clients or change the FTP access.

The procedure for downloading the client files using FTP is shown in [Procedure 4-1, Downloading Client Files Using FTP](#).



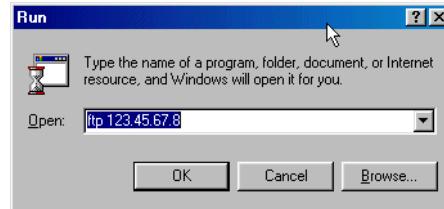
### Procedure 4-1. Downloading Client Files Using FTP

Use the following to FTP the client file from the server to the user's computer.

1. Click the **Start** button on the computer where you are installing the clients.
2. Select **Run** from the Window's **Start** menu.

Result: The **Run** dialog box displays and the cursor defaults to the **Open** field.

3. Enter *ftp xxx*, where the xxx is the IP address or the name of the server from which you are getting your client files. See [Figure 4-1](#).



**Figure 4-1.** Run dialog box with server IP address where Voyager clients reside

4. Click the **OK** button.

Result: Successful execution of FTP displays a "Connected to [IP address]" message and prompts you to enter the user ID.

5. Enter **voyager** and press **enter**.

Result: You are prompted to enter a password.

6. Enter **your voyager password** (site specific) and press **enter**.

Result: The "User voyager logged in" confirmation message and the ftp prompt, *ftp>* displays.

7. Navigate to the clients directory for the installation file by entering `cd /m1/voyager/clients`.
8. Enter `dir`.

Result: This displays/confirms the files that are stored in that directory. The following files display:

- `VoyagerInstall.exe`
- `VoyagerInstall.msi` (Windows Installer Package version)
- `VoyagerMedia.exe`
- `VoyagerMedia.msi` (Windows Installer Package version)

**NOTE:**

The VoyagerMedia installation files display if your institution has purchased this optional application.

9. At the prompt, enter `bin`.

Result: This sets the system to transfer the file in binary mode.

10. At the prompt, enter `hash`.

Result: This sets the system to display hash marks to indicate that the transfer is occurring.

11. At the prompt, enter “get” followed by the name of the installation file you want to download to the PC. For example, `get VoyagerInstall.msi`.

Result: This downloads the file to the user’s computer.

12. Enter `bye` and press `enter`.

Result: This ends your ftp session and you can proceed with you client installation. See [Figure 4-2](#).

**Figure 4-2.** Example FTP session

## Installing Voyager

There are two alternatives for installing Voyager client software. They are as follows:

- Traditional. See this section and [Procedure 4-2, Installing Voyager Clients](#), on page [4-6](#).
  - Automated. See [\[Upgrade\] Stanza](#) on page [4-30](#)

The core Voyager clients are installed by running one of the following files:

- VoyagerInstall.exe
  - VoyagerInstall.msi

When run, the installation file delivers the following:

- System files.
  - Directory structure.
  - Default files.
  - Voyager clients.

**NOTE:**

When using the `VoyagerInstall.exe` installation method, the user must have administrator privileges.

The procedure for installing the Voyager clients using the `VoyagerInstall.exe` installation file is shown in [Procedure 4-2, Installing Voyager Clients](#).



#### Procedure 4-2. Installing Voyager Clients

---

Use the following to install the Voyager clients on a user's computer using the `VoyagerInstall.exe` installation file.

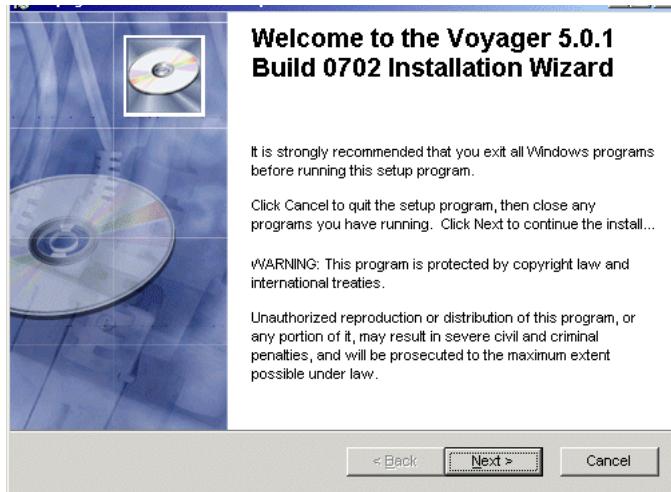
1. Click the **Start** button on the computer where you are installing the clients.

2. Select **Run** from the Window's **Start** menu.

Result: The **Run** dialog box displays and the cursor defaults to the **Open** field.

3. Enter `VoyagerInstall.exe`, and click **OK**.

Result: The Windows Installer extracts the necessary files and displays the setup **Welcome** dialog box. See [Figure 4-3](#).

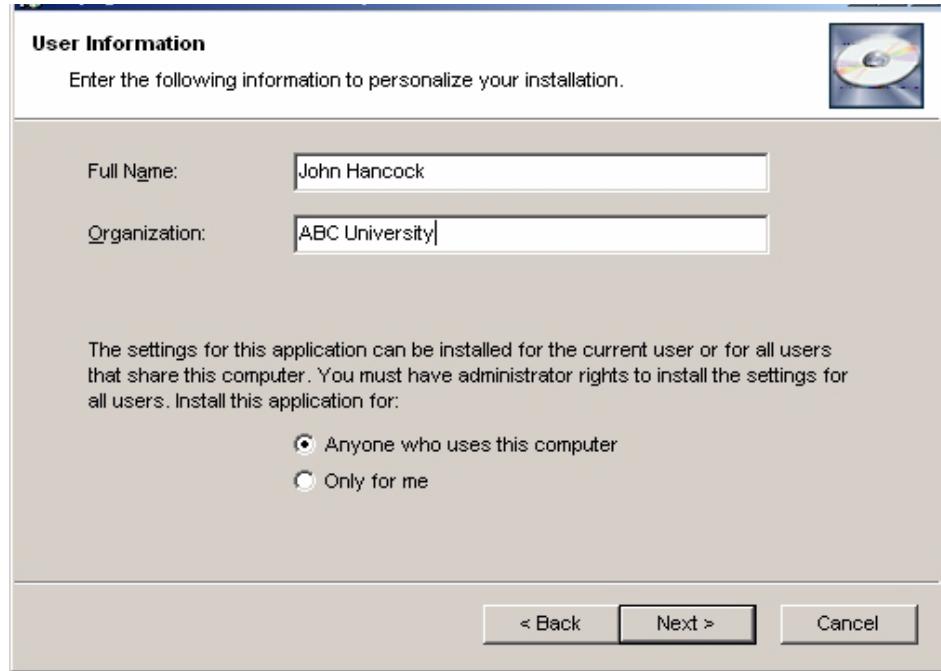


---

**Figure 4-3.** Voyager setup Welcome dialog box

4. Read the setup Welcome information, and click the **Next** button when you are ready.

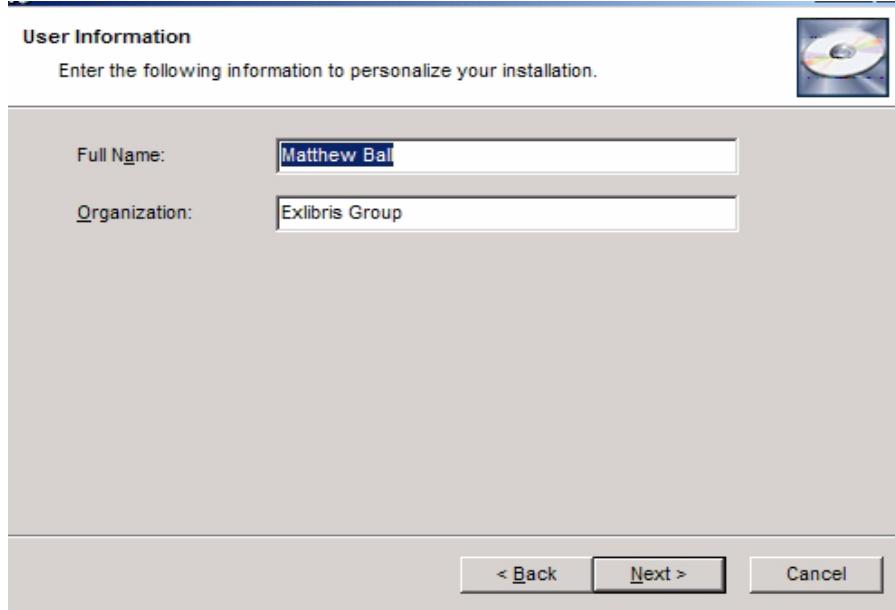
Result: The **User Information** dialog box opens. See [Figure 4-4](#).



**Figure 4-4.** Voyager installation User Information dialog box

**NOTE:**

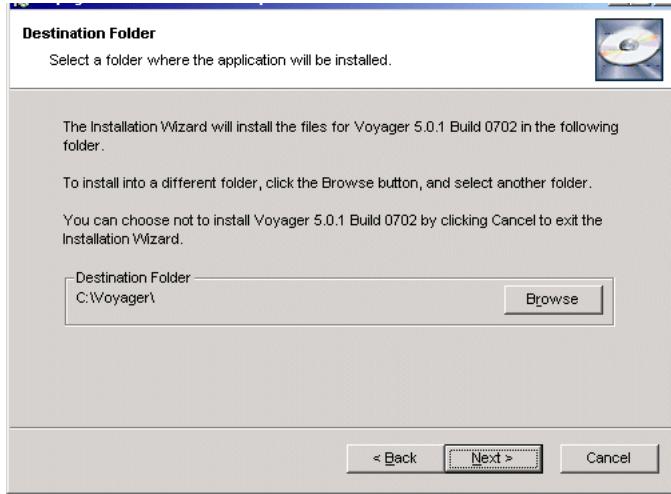
[Figure 4-4](#) illustrates the **User Information** dialog box presented to a user with administrator privileges. See [Figure 4-5](#) for an example of the **User Information** dialog box displayed for users without administrator privileges (using the `VoyagerInstall.msi` installation method). For more information regarding alternative installation methods, see the [\[Upgrade\] Stanza](#) on [page 4-30](#).



**Figure 4-5.** Voyager installation User Information dialog box w/o administrator privileges

5. Enter Full Name, Organization, and select one of the following:
  - Anyone who uses this computer
  - Only for me
6. Click **Next**.

Result: The **Destination Folder** dialog box opens. See [Figure 4-6](#).

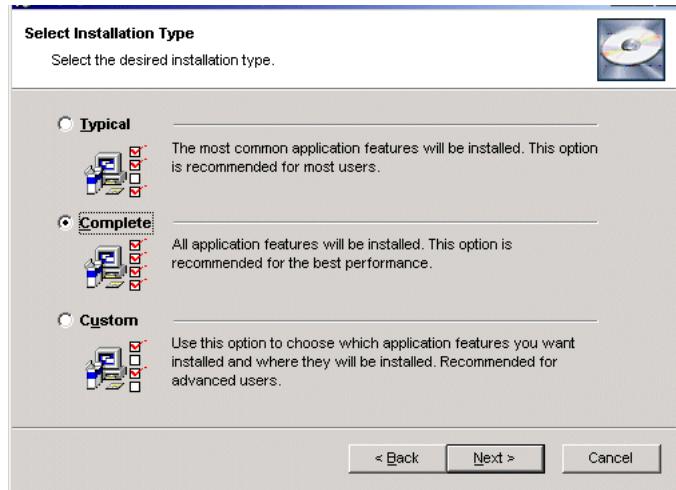


---

**Figure 4-6.** Voyager installation Destination Folder dialog box

7. Identify the Destination Folder. The default folder for installation is C:\\Voyager.
  - a. If this is the directory you want, click the **Next>** button.
  - b. If this is not the directory you want, click the **Browse** button and navigate to the correct directory and then click the **Next>** button.

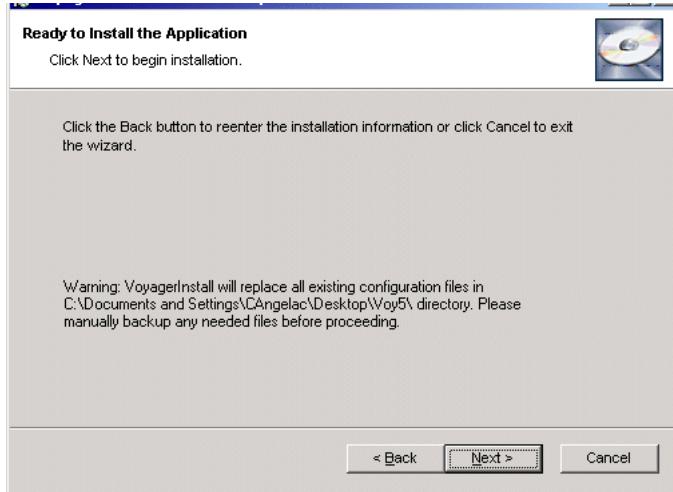
Result: The directory into which the clients are installed is defined and the **Select Installation Type** dialog box opens. See [Figure 4-7](#).



**Figure 4-7.** Voyager installation Select an Installation Type dialog box

8. Select the radio button corresponding to the type of installation you want to process, and click the **Next>** button.
  - **Typical** installs the Acquisitions, Circulation, and Cataloging clients.
  - **Complete** installs all the Voyager clients.
  - **Custom** allows the user to select specific clients. The Acquisitions, Circulation, and Cataloging clients are selected by default.

Result: The **Ready to install the Application** dialog box opens warning the user that files may be overwritten. See [Figure 4-8](#).



---

**Figure 4-8. Warning Message**

**OPTIONAL:**

9. *Back up any files that you may want to keep. For example, you may want to save any pre-existing configuration files like the voyager.ini file to a different directory for later reference.*

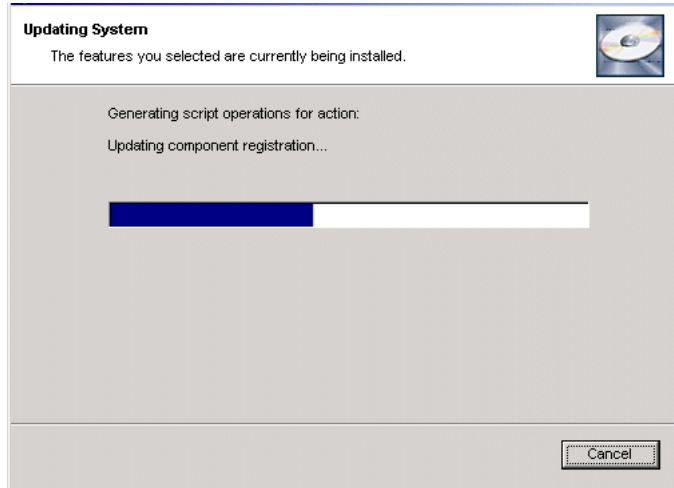


**IMPORTANT:**

*VoyagerInstall replaces all existing configuration files in the [C:\Voyager] directory.*

10. Click the **Next** button.

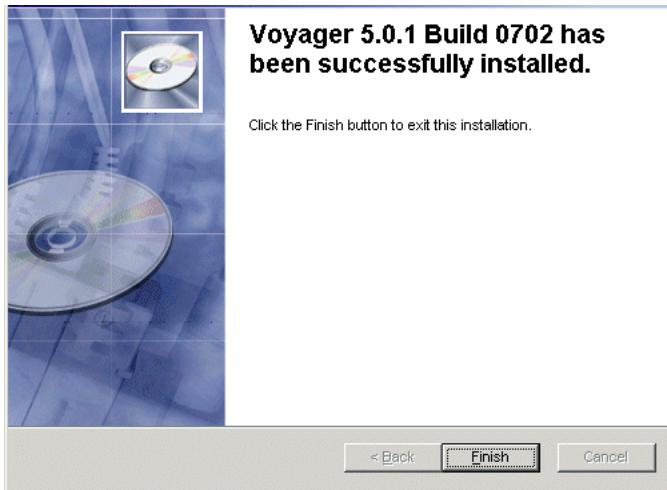
Result: The installation begins. See [Figure 4-9](#).



---

**Figure 4-9.** Installation in process

11. When the installation is complete, a dialog box displays. See [Figure 4-10](#).
- 



---

**Figure 4-10.** Voyager Installation Successful Message

12. Click the **Finish** button to exit the installation program.

Result: Voyager is installed on the user's PC.

13. Edit the `voyager.ini` file to include the appropriate server and port information. See [The Voyager.ini File on the PC](#) on [page 4-23](#) for more information.

Result: The voyager clients are installed and customized for your institution's use.

**NOTE:**

If your site uses Voyager's Media Scheduling application, run the `VoyagerMedia.exe` to install the Media clients. To do this, double-click on this executable and follow the directions on the screen.

---

## **Installing Media Scheduling**

---

If your site uses Voyager Media Scheduling, you need to run the `VoyagerMedia` executable file to place the Media Scheduling files on the user's PC.

The process is similar to the one described in [Procedure 4-2, Installing Voyager Clients](#). Simply substitute the `VoyagerMedia.exe` file name for `VoyagerInstall.exe` in Step 3.

## **Modifying (Adding or Deleting Clients) an Installation**

---

After installing the Voyager clients, users can modify their installation. They can add or delete Voyager modules.

For example, if you ran the typical installation of the Voyager clients that installs the Acquisitions, Cataloging, and Circulation clients, and you want to include the Callslip client, you can add it. Or, if you ran the complete installation, but do not need the System Administration module, you can delete it.

The procedure for modifying the Voyager installation is shown in [Procedure 4-3, Adding or Deleting Voyager Clients](#).



### **Procedure 4-3. Adding or Deleting Voyager Clients**

---

Use the following to add or delete Voyager clients on a computer that has Voyager clients currently installed.

1. Click the **Start** button.

2. Select **Run** from the Window's **Start** menu.

Result: The **Run** dialog box displays and the cursor defaults to the **Open** field.

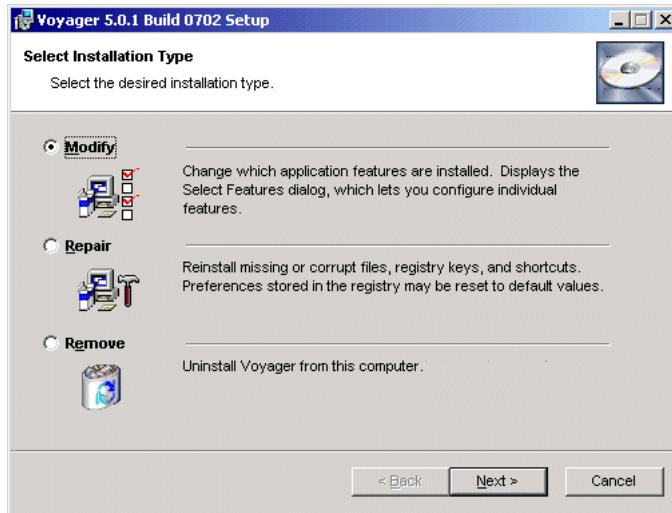
3. Enter **VoyagerInstall.exe**, and click **OK**.

Result: Files are extracted to the local computer and the **Select Installation Type** dialog box displays. See [Figure 4-11](#).

**NOTE:**

The **Modify** radio button is selected by default.

---

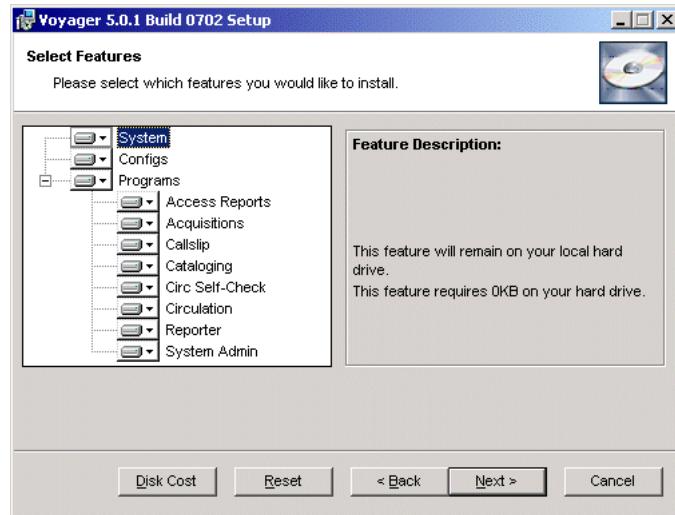


---

**Figure 4-11.** Select Installation Type dialog box

4. Click the **Next** button.

Result: The **Select Features** dialog box displays the features that are currently installed on the user's computer. See [Figure 4-12](#).



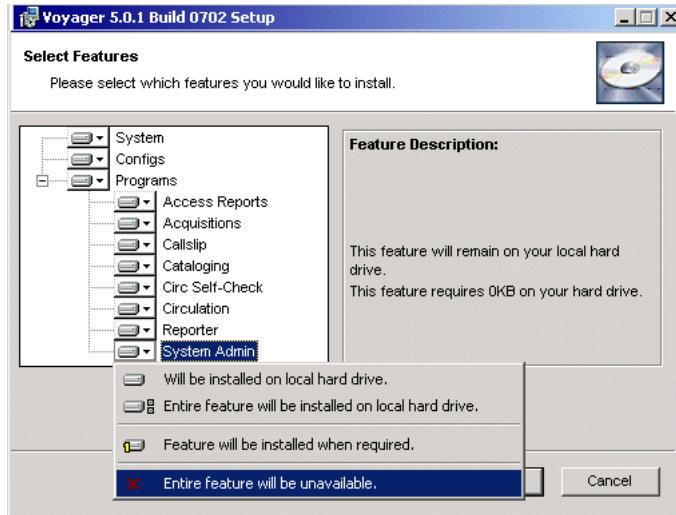
---

**Figure 4-12. Voyager Setup with Selected Features**

5. Select the clients you want to add or delete.

For this procedure, the example used is to delete the Voyager System Administration client. See [Figure 4-13](#).

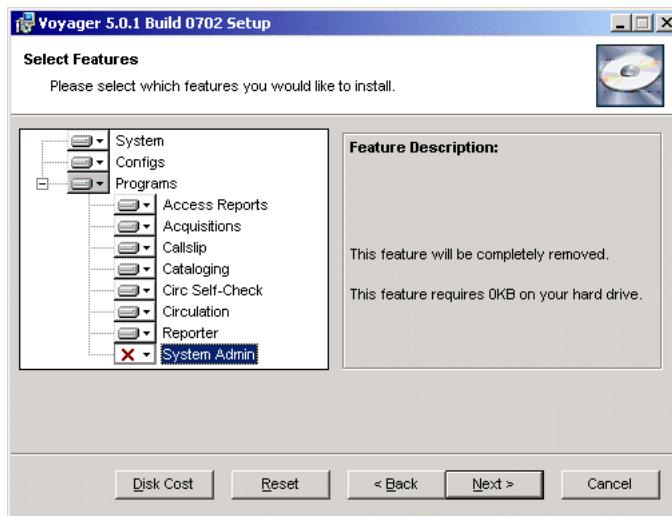
Result: The System Administration client is marked for deletion. See [Figure 4-14](#).



---

**Figure 4-13.** Selecting System Administration client for deletion

---



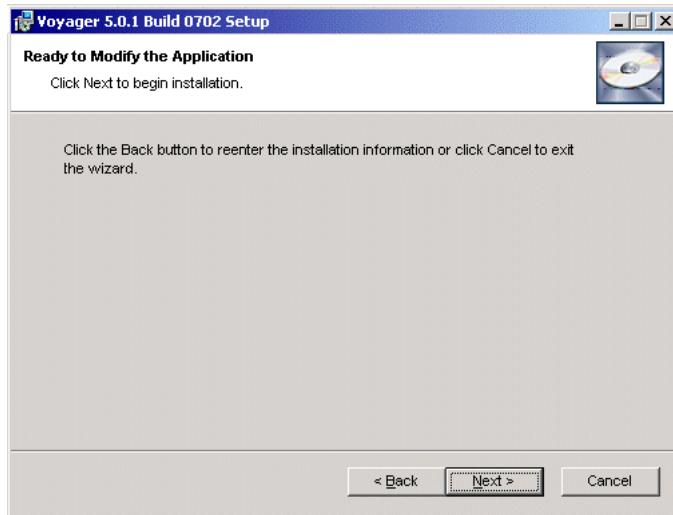
---

**Figure 4-14.** System Administration client identified for deletion

---

6. Click the **Next** button.

Result: The system is ready to modify the application. See [Figure 4-15](#).

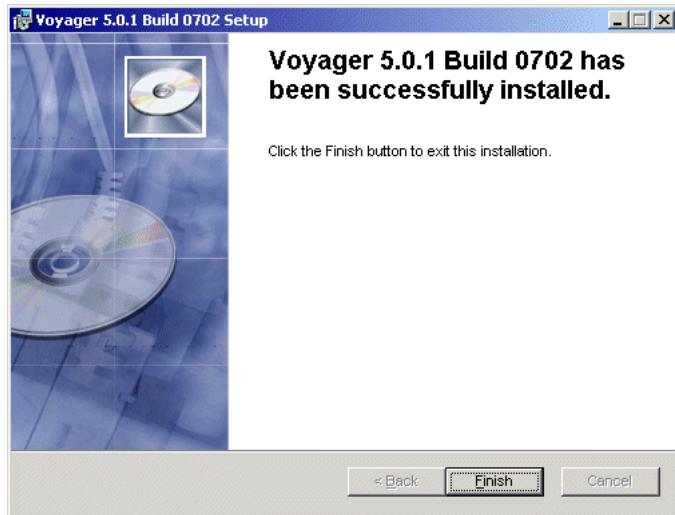


---

**Figure 4-15.** Voyager ready to modify

7. Click the **Next** button.

Result: The modification of the installation occurs and displays a confirmation message. See [Figure 4-16](#).



---

**Figure 4-16. Voyager Programs Installation Successful**

8. Click **Finish**.

Result: The installation is modified.

---

## Repairing an Installation of Voyager Clients

---

If instructed by Ex Libris's Customer Support department or you suspect that your installation was not complete, you can repair the installation.

The procedure for repairing a Voyager installation is shown in [Procedure 4-4, Repairing an Installation](#).



### Procedure 4-4. Repairing an Installation

---

Use the following to repair a Voyager client installation.

1. Click the **Start** button.

2. Select **Run** from the Window's **Start** menu.

Result: The **Run** dialog box displays and the cursor defaults to the **Open** field.

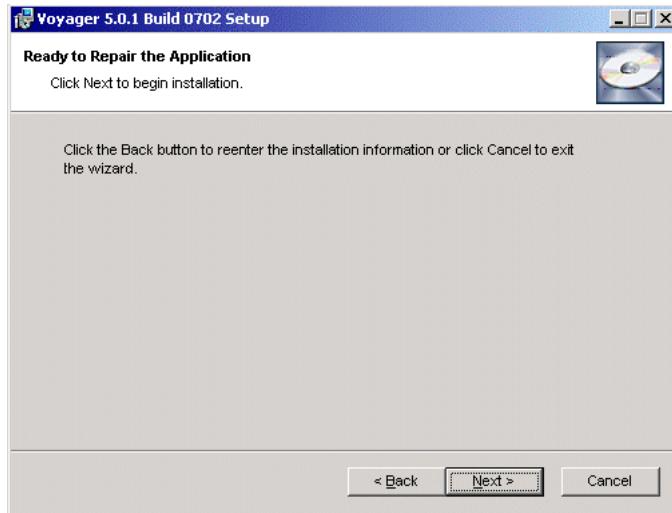
3. Enter *VoyagerInstall.exe*, and click **OK**.

Result: Files are extracted to the local computer and the **Select Installation Type** dialog box displays.

4. Click the **Repair** radio button and subsequent the **Next** button.

Result: The **Ready to Repair the Application** dialog box displays. See [Figure 4-17](#).

---

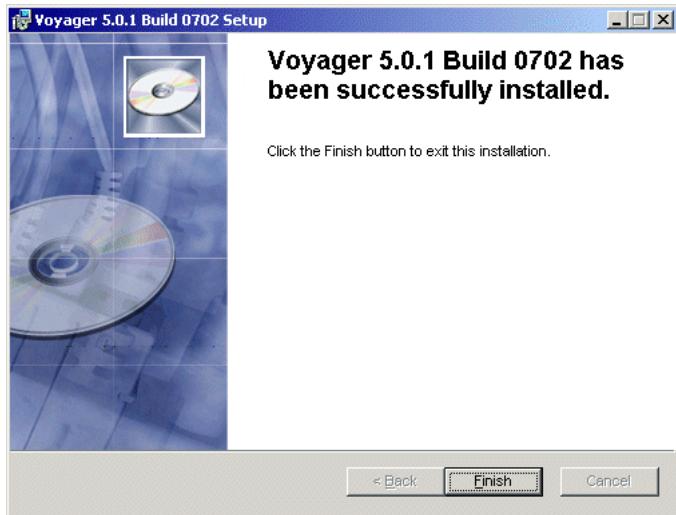


---

**Figure 4-17. Voyager Ready to Repair the Installation**

5. Click the **Next** button.

Result: The repair occurs and displays a confirmation message. See [Figure 4-18](#).



---

**Figure 4-18. Voyager Installation Successful**

6. Click **Finish**.

Result: The installation is repaired.

---

## **Uninstalling the Voyager Clients**

---

If needed, users should uninstall Voyager clients using Windows, not the installation program.

Before uninstalling the clients, be sure to save any customized files you may have such as tag table files, archived files, and access reports or queries.

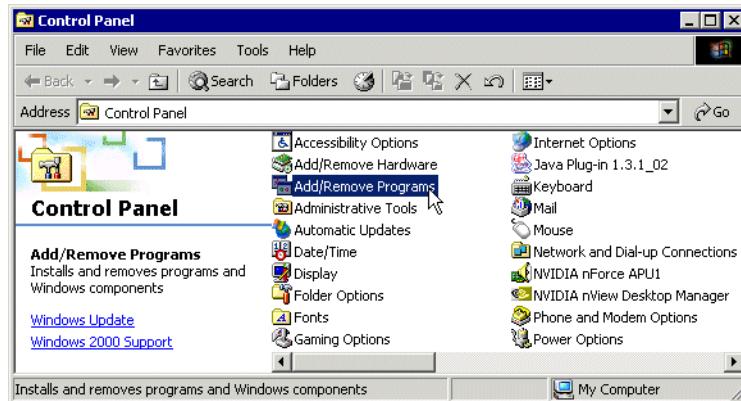
The procedure for uninstalling the Voyager clients is shown in [Procedure 4-5, Uninstalling Voyager Clients](#).



### Procedure 4-5. Uninstalling Voyager Clients

Use the following to uninstall the Voyager clients.

1. Access the Windows control panel and select **Add/Remove** programs. See [Figure 4-19](#).



**Figure 4-19. Control Panel with Add/Remove Programs Selected**

Result: The **Add/Remove** dialog box displays.

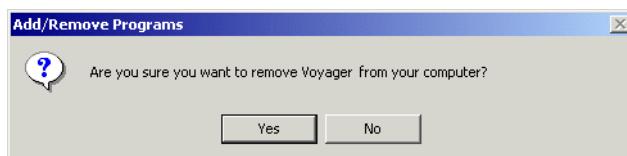
2. Select the Voyager program and click the **Remove** button. See [Figure 4-20](#).



**Figure 4-20.** Add/Remove Dialog Box with Voyager Selected

Result: The system asks for confirmation before removing the program. See [Figure 4-21](#).

---



**Figure 4-21.** Confirmation of Removing Voyager

3. Click **Yes** to remove Voyager.

Result: Voyager is removed and the **Add/Programs** dialog box updates.

4. Close the **Add/Remove** dialog box.

Result: Voyager is uninstalled.

---

## The Voyager.ini File on the PC

The `voyager.ini` file is an initialization file that contains important connection information that enables your Voyager clients to access the server.

During the installation process, the `voyager.ini` file is placed in the directory into which the clients are installed. This is typically the `c:\Voyager` directory.

**NOTE:**

Since the `voyager.ini` file is overwritten each time the `VoyagerInstall.exe` is executed, it must be edited to include the appropriate connection information and stanzas needed for the specific computer.

The `voyager.ini` file contains the following stanzas:

- [Cataloging]
- [Acquisitions]
- [Circulation]
- [Reports]
- [SysAdmin]
- [CallSlip]
- [Media Scheduling]
- [GlobalLog]
- [SearchURI]
- [MARC POSTing]
- [E-mail]
- [Upgrade]

[Figure 4-22](#) shows an example of a `voyager.ini` file.

```
[Cataloging]
Server=xxxx.xxxx.xxxx.xxxx
Port=7010
Timeout=60

[Acquisitions]
Server=xxxx.xxxx.xxxx.xxxx
```

**Figure 4-22.** Sample `voyager.ini` file

```
Port=7020
Timeout=60

[Circulation]
Server=xxx.xxx.xxx.xxx
Port=7030
Timeout=60
ChargeTimeout=60

[Reports]
Server=xxx.xxx.xxx.xxx
Port=7040
Timeout=60

[SysAdmin]
Server=xxx.xxx.xxx.xxx
Port=7050
Timeout=60

[CallSlip]
Server=xxx.xxx.xxx.xxx
Port=7080
Timeout=60

[MediaScheduling]
Server=xxx.xxx.xxx.xxx
Port=7085
Timeout=60

[GlobalLog]
SingleLogin=Y
Encrypt=N
ServerSortList=Y
ASCIISortList=Y
ASCIISortColumn=Y

[SearchURI]
```

---

**Figure 4-22.** Sample *voyager.ini* file (Continued)

```
#Name=Barn&esNoble
#URI=http://search.barnesandnoble.com
#Copy=Y
#SearchSyntax=/booksearch/results.asp?WRD=<searchtext>
#Name=WebVoyage
#URI=http://xxx.xxx.xxx.xxx/
#Copy=Y
#SearchSyntax=/vwebv/
    search?searchArg=<searchtext>&searchCode=TALL&limitTo=none&recCount
    =10&searchType=0
#Name=Amazon
#URI=http://www.amazon.com
#ECopy=Y
#SearchSyntax=/exec/obidos/external-search?keyword=<searchtext>&mode=blended
#Name=Worldcat
#URI=http://www.worldcat.org/
#Copy=Y
#SearchSyntax=search?&q=
#Name=Google
#URI=http://www.google.com/
#Copy=Y
#SearchSyntax=search?&q=

[MARC POSTing]
#Pre-7.0 WebVoyage
#WebVoyage="http://xxx.xxx.xxx.cgi-bin/Pbibredirect.cgi"
#Voyager 7.0 WebVoyage
WebVoyage="http://xxx.xxx.xxx.vwebv/holdingsInfo"

[E-mail]
Server=xxx.xxx.xxx.xxx
Port=25

# This stanza replaces the per-stanza NewVersion= in Voyager 7.0
[Upgrade]
Voyager=
Media=
```

Figure 4-22. Sample voyager.ini file (Continued)

## Client Module Stanzas

The client module stanzas set up the client/server connections for each module.

[Table 4-1](#) lists the parameters that can be specified for the following client module stanzas:

- [Acquisitions]
- [Call Slip]
- [Cataloging]
- [Circulation]
- [MediaScheduling]
- [Reports]
- [Sysadmin]

**Table 4-1. Module Stanza Parameters**

Parameter	Description
Server= <i>value</i>	This parameter contains the IP address of the Voyager server.
Port= <i>value</i>	This parameter corresponds to each module's designated port, as defined by the <code>/etc/services</code> file on the Voyager server.
Timeout= <i>value</i>	This parameter specifies the number of seconds that are allowed to elapse while attempting to connect the client to the Voyager server.  The default is 60 seconds.
ChargeTimeout= <i>value</i>	The parameter defines the charge timeout value for the [Circulation] stanza only.  The default is 60 seconds.  For more information, see <a href="#">Circulation Charge Timeout</a> on <a href="#">page 4-42</a> .

## [GlobalLog] Stanza

The [GlobalLog] stanza allows you to configure other options (such as single login, encryption, and staff client re-sort).

[Table 4-2](#) describes the parameters in the [GlobalLog] stanza.

**Table 4-2. [GlobalLog] Stanza Parameters**

Parameter	Description
SingleLogin= <i>value</i>	This parameter enables a single login screen for all modules so that a user does not have to login for each module.  Set the <i>value</i> to Y to enable or N to disable.  For more information, see <a href="#">Single Client Login on page 4-42</a> .
Encrypt= <i>value</i>	This parameter enables a secure connection between the Voyager server and the following modules: Acquisitions, CallSlip Daemon, Cataloging, Circulation, Reporter, System Administration, Media Scheduling, and Media System Administration.  Set the <i>value</i> to Y to enable or N to disable.  For more information, see <a href="#">Encryption on page 4-43</a> .
ServerSortList= <i>value</i>	This parameter enables the <b>Sort By</b> drop-down menu on search results screens from the Cataloging, Circulation, and Acquisitions modules. From this menu, the following items are made available: <ul style="list-style-type: none"> <li>• Title</li> <li>• Author</li> <li>• Publish Date</li> <li>• Publish Date Descending</li> <li>• Relevance (relevance searches only)</li> </ul> Default <i>value</i> is Y.  For more information on Voyager-aware re-sorts, see <a href="#">Staff Client Re-Sort on page 4-44</a> .

**Table 4-2. [GlobalLog] Stanza Parameters**

Parameter	Description
ASCIIList= value	This parameter enables the <b>Sort By</b> drop-down menu on search results screens from the Cataloging, Circulation, and Acquisitions modules. From this menu, Quicksort items, which are user-defined in the System Administration module, are made available.  Default value is N.  For more information on ASCII re-sorts, see <a href="#">Staff Client Re-Sort on page 4-44</a> .
ASCIIColumn= value	This parameter enables column headers on search results screens to be clicked to re-sort search results from the Cataloging, Circulation, and Acquisitions modules.  Default value is N.  For more information on ASCII re-sorts, see <a href="#">Staff Client Re-Sort on page 4-44</a> .

**[E-mail] Stanza**

The [E-mail] stanza specifies the email server and port connections. [Table 4-3](#) describes the parameters in the [E-mail] stanza.

**Table 4-3. [E-mail] Stanza Parameters**

Parameter	Description
Server=value	Specifies the IP address of the e-mail server.
Port=value	Specifies the port connection on the e-mail server.

This stanza must be configured to allow e-mailing notices produced in the Reporter module. See [Configuring the voyager.ini file for E-mail](#) in the *Voyager Reporter User's Guide* for information on this stanza and its configuration.

**[MARC Posting] Stanza**

The [MARC POSTing] stanza in the `voyager.ini` file contains a user-defined key-value pair. The key is the text that appears in the module, on the drop-down menu, listing where the user wants to send and display the MARC record. The value is the URI text to generate the record and an XML-formatted version of the record.

For more information on the [MARC POSTing] stanza, see [MARC Record Posting on page 4-47](#).

### [SearchURI] Stanza

The [SearchURI] stanza of the `voyager.ini` file configures and enables an additional button on the bibliographic **Search** dialog box to open a Uniform Resource Identifier (URI).

[Table 4-4](#) describes the parameters used in the [SearchURI] stanza.

**Table 4-4. [SearchUri] Stanza Parameters**

Parameter	Description
Name= <i>value</i>	This parameter provides a label for the additional button on the <b>Search</b> dialog box.
URI= <i>value</i>	This parameter contains the base of the URI.  <b>NOTE:</b> The base is the leftmost section of the address, for example, the web address portion.
Copy= <i>value</i>	This parameter is either 'y' for yes, or 'n' for no. If yes, the text in the <b>Search for</b> field is used as the search term(s).
SearchSyntax= <i>value</i>	This parameter provides the syntax (such as /search?&q=<search text>) for the search.  The system takes the base (URI) and combines it with the search syntax (SearchSyntax) to create the search.  <b>NOTE:</b> Users are responsible for the correct search syntax.

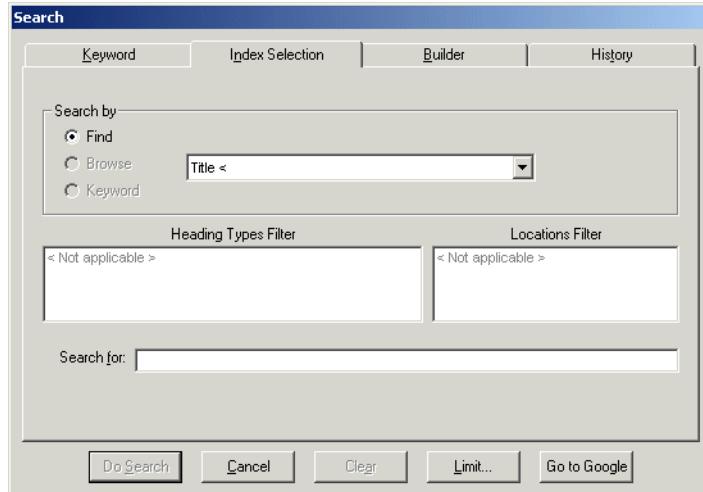
### Adding a Button to the Search Dialog Box

If you would like to add an additional button to the Search dialog box, configure the [SearchURI] stanza. A sample configuration is shown in [Figure 4-23](#).

```
[SearchURI]
Name=Go to Google
URI=http://www.google.com/
Copy=Y
SearchSyntax=/search?q=<search text>
```

**Figure 4-23.** Sample [SearchURI] Stanza

Using the sample configuration, [Figure 4-24](#) shows a “Go to Google” button on the **Search** dialog box. When this button is selected the Google™ website is accessed. The search terms from the **Search** dialog box are inserted into the Google search box and a Google search is executed using those terms.



**Figure 4-24.** “Go to Google” Button Added to Search Dialog Box

### **[Upgrade] Stanza**

The **[Upgrade]** stanza provides the facility for automating the download and installation of client software to a user's PC when a newer version is detected by the user's PC.

This functionality consolidates and replaces the **NewVersion=** parameter capability that was available with each client stanza.

The [Upgrade] stanza contains the following parameters:

- Voyager=
- Media=

See [Figure 4-22](#) on [page 4-23](#) for an illustration of the complete voyager.ini file that includes the [Upgrade] stanza.

The Voyager= parameter is used to initiate the download and installation of new Voyager client software, and the Media= parameter is used to initiate the download and installation of new Media Scheduling software (if this applies to your site).

Your system administrator needs to provide you with the content/format of the parameter option that is specific to your site for accessing the client installation from a server. This points to the location of the installation software. See the following for examples:

```
[Upgrade]
Voyager=\Voyager\Installs\VoyagerInstall.exe
Media=\Voyager\Installs\VoyagerMedia.exe
```

---

**Figure 4-25. [Upgrade] stanza example using a valid UNC resource path**

```
[Upgrade]
Voyager=http://10.102.100.91:37908/VoyagerInstall.exe
Media=http://10.102.100.91:37908/MediaInstall.exe
```

---

**Figure 4-26. [Upgrade] stanza example using an http resource**

```
[Upgrade]
Voyager=ftp://username:password@10.102.100.91/m1/voyager/clients/VoyagerIn-
stall.exe
Media=ftp://username:password@10.102.100.91/m1/voyager/clients/MediaIn-
stall.exe
```

---

**Figure 4-27. [Upgrade] stanza example using an ftp resource**

```
[Upgrade]
Voyager=\\Voyager\Installs\ABCUniversityVoyagerInstall.bat
Media=\\Voyager\Installs\ABCUniversityMediaInstall.bat
```

**Figure 4-28.** [Upgrade] stanza example using a .bat file

The Voyager= and Media= parameters may point to either the .exe or .msi type of installation. However, using the .msi method of installation has the benefit of not requiring administrator privileges to complete the installation process if the PC operating system is Windows XP.

**NOTE:**

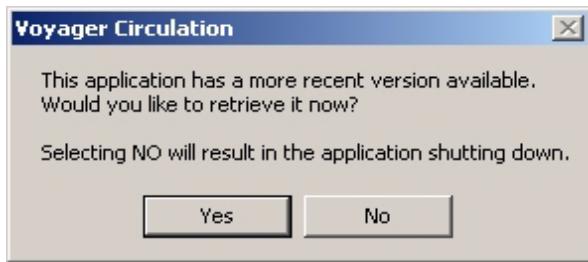
Due to the increased security model in Vista, administrator privileges are required for either method of installation, .exe or .msi.

The Voyager= and Media= parameters are considered inactive in the following situations:

- When not included in the stanza.
- When commented out.
- When the value is blank.

### What Happens When New Software is Detected?

When a Voyager client is started and the system determines that there is a newer version of the client available, the user receives a “version available” message. See [Figure 4-29](#).



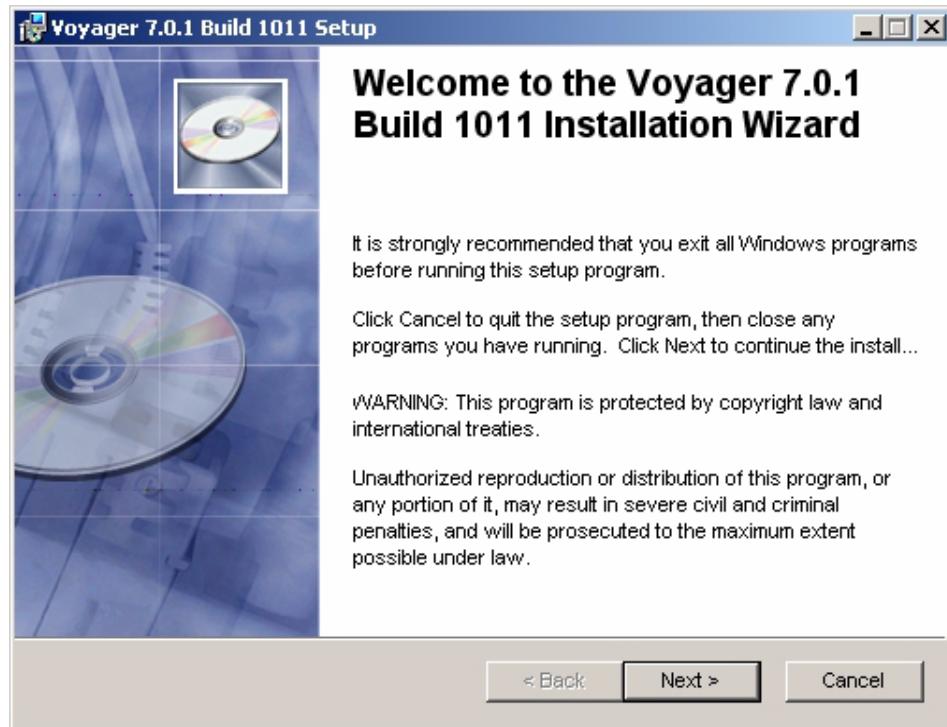
**Figure 4-29.** Version available message

When the user clicks the **Yes** button, the next step varies depending on the values stored in **Voyager=** and **Media=**.

- When the **Voyager=** value for the traditional method is used as in [Figure 4-25](#), see [Traditional Method](#) on [page 4-33](#).
- When the **Voyager=** value for the HTTP method is used as in [Figure 4-26](#), see [HTTP Method](#) on [page 4-34](#).
- When the **Voyager=** value for the FTP method is used as in [Figure 4-27](#), see [FTP Method](#) on [page 4-36](#).
- When the **Voyager=** value for the **.bat** method is used as in [Figure 4-28](#), see [.BAT File Method](#) on [page 4-36](#).

### Traditional Method

The system begins to process the installation executable from the server and displays the installation **Welcome** dialog box. See [Figure 4-30](#).



---

Figure 4-30. Installation Welcome dialog box

The remainder of the installation follows the steps identified in [Procedure 4-2, Installing Voyager Clients](#), on page [4-6](#). The Media Scheduling installation follows a similar procedure.

### HTTP Method

The system needs to validate your security. See [Figure 4-31](#).

---



---

**Figure 4-31. Security page (authentication)**

Enter the User ID, password, and click OK. The file download dialog box displays. See [Figure 4-32](#).

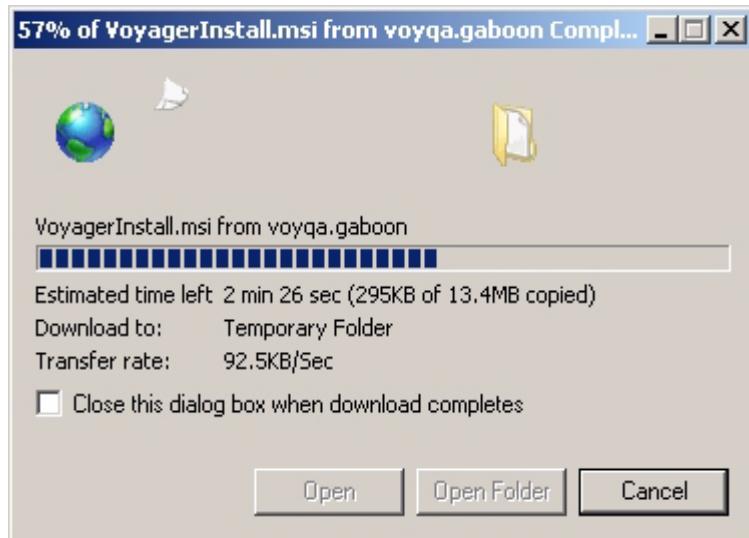


Figure 4-32. File download dialog box

Subsequently, the file download security warning dialog box displays. See [Figure 4-33](#).



Figure 4-33. File download security warning dialog box

When you click Run, the client software is downloaded for installation and the **Welcome** dialog box is displayed. See [Figure 4-30](#).

The remainder of the installation follows the steps identified in [Procedure 4-2, Installing Voyager Clients](#), on page [4-6](#). The Media Scheduling installation follows a similar procedure.

### FTP Method

The system displays the file download security warning dialog box. See [Figure 4-34](#).

---



---

**Figure 4-34.** File download security warning dialog box

When you click Run, the client software is downloaded for installation and the **Welcome** dialog box is displayed. See [Figure 4-30](#).

The remainder of the installation follows the steps identified in [Procedure 4-2, Installing Voyager Clients](#), on page [4-6](#). The Media Scheduling installation follows a similar procedure.

### .BAT File Method

The next steps relative to the .bat method are site specific. The .bat file provides for any number of installation options to be coded.

The following example illustrates the use of the .bat method:

```
VoyagerInstall.msi /qb+! /L*v "c:\testvoy\voy.log"  
INSTALLDIR="c:\testvoy" INSTALLLEVEL="4"
```

**Figure 4-35. Example code for .bat method**

The example in [Figure 4-35](#) does the following:

- This causes a progress dialog box (with no cancel button) and a completed dialog box to display.
- All messages are written to C:\testvoy\voy.log
- The software is installed in C:\testvoy
- A Complete (versus Custom or Typical) installation is performed.

See [Table 4-5](#) for a description of possible options using this approach.

**Table 4-5. Options for .bat file method**

Option	Parameters	Description
/x		Uninstalls a product.  Example: msieexec /x VoyagerInstall.msi

**Table 4-5. Options for .bat file method**

Option	Parameters	Description
/L	[i w e a r u c m o p v + !] Logfile	<p>Specifies path to the log file, and the flags indicate which information to log.</p> <p>i - Status messages. w - Non-fatal warnings. e - All error messages. a - Start up of actions. r - Action-specific records. u - User requests. c - Initial UI parameters. m - Out-of-memory or fatal exit information. o - Out-of-disk-space messages. p - Terminal properties. v - Verbose output. + - Append to existing file. ! - Flush each line to the log. * - Wildcard, log all information except for the v option. To include the v option, specify /L*v.</p> <p>The following example line of code logs all information except for the v option into c:\VoyagerLog.txt.</p> <p><b>Example:</b> VoyagerInstall.msi /L* c:\VoyagerLog.txt</p>

**Table 4-5. Options for .bat file method**

Option	Parameters	Description
/q	n b r f	<p>Sets user interface level.</p> <p>q , qn - No UI (User Interface).</p> <p>qb - Basic UI . Use qb! to hide the Cancel button.</p> <p>qr - Reduced UI with no modal dialog box displayed at the end of the installation.</p> <p>qf - Full UI and any authored Fatal Error, User Exit, or Exit modal dialog boxes at the end.</p> <p>qn+ - No UI except for a modal dialog box displayed at the end.</p> <p>qb+ - Basic UI with a modal dialog box displayed at the end. The modal box is not displayed if the user cancels the installation. Use qb!+ or qb!+ to hide the Cancel button.</p> <p>qb- - Basic UI with no modal dialog boxes. Please note that /qb-- is not a supported UI level. Use qb!- or qb!- to hide the Cancel button.</p> <p>The following example line of code installs the product with no user interface.</p> <p><b>Example:</b> VoyagerInstall.msi /qn</p> <p><b>NOTE:</b> The ! option is available with Windows Installer version 2.0 and works only with basic UI. It is not valid with full UI.</p>

**Table 4-5. Options for .bat file method**

Option	Parameters	Description
INSTALLDIR=	[directory]	<p>Set installation directory.</p> <p>The default installation directory is C:\Voyager.</p> <p>The following example line of code only installs the programs into the D:\Voyager\ directory.</p> <p><b>Example:</b> VoyagerInstall.msi INSTALLDIR="D:\Voyager"</p> <p><b>NOTE:</b> This option only works with VoyagerInstall and VoyagerMedia.</p>
EIS_APP=	[A B C D E F G H]	<p>Specifies applications and/or files to be installed from VoyagerInstall.msi. The default applications to be installed for a typical installation are ACDIJ.</p> <ul style="list-style-type: none"> <li>A - Acquisitions</li> <li>B - Callslip</li> <li>C - Cataloging</li> <li>D - Circulation</li> <li>E - Circ Self Check</li> <li>F - Reporter</li> <li>G - System Admin</li> <li>H - Access Reports</li> <li>I - System Files</li> <li>J - Config Files</li> </ul> <p>The following example line of code only installs Cataloging, Reporter, and System Administration.</p> <p><b>Example:</b> VoyagerInstall.msi EIS_APP="CFG"</p> <p><b>NOTE:</b> This option only works with VoyagerInstall.</p>

**Table 4-5. Options for .bat file method**

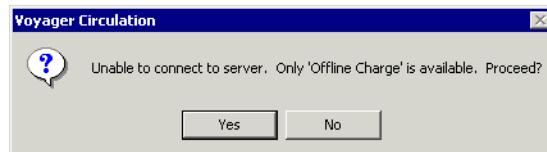
Option	Parameters	Description
INSTALLLEVEL=	[#]	<p>Set installation priority level. The default installation priority level is 3.</p> <p>3 - Typical Install</p> <p>4 - Complete Install</p> <p>The following example line of code installs all applications.</p> <p><b>Example:</b> VoyagerInstall.msi INSTALLLEVEL="4"</p>

**NOTE:**

The options `INSTALLLEVEL` and `EIS_APP` should not be used together.

**Circulation - Offline Option**

When the Circulation client determines there is a new version and the user clicks the **No** button when prompted to install the new software (see [Figure 4-29](#)), the system treats this as an unsuccessful attempt to connect to the Circulation module. When this happens, the Offline circulation message displays. See [Figure 4-36](#).

**Figure 4-36. Offline Circulation Message**

For more information regarding offline circulation, see the *Voyager Circulation User's Guide*.

## Circulation Charge Timeout

When charging items in Voyager's Circulation module, the system is configured such that the **Charge** workspace closes automatically. This occurs if, after entering the charging patron's information, a set amount of time passes with no activity occurring in the **Charge** workspace. This set amount of time is the charge time-out value.

The `ChargeTimeout=` key is in the `[circulation]` stanza of the `voyager.ini` file. The default charge time-out value is 60 seconds (see [Figure 4-22 on page 4-23](#)).

[Table 4-6](#) shows the possible time-out values for the `ChargeTimeout=` key and Voyager's response.

**Table 4-6. Circulation ChargeTimeout Key Values**

ChargeTimeout Value	System Response
Blank	No charge timeout, the timer will not count.
0	No charge timeout, the timer will not count.
1 - 4	The system will count 5 seconds.
> 4	The timer will count the number of seconds specified. There is no maximum value.  <b>NOTE:</b> The default charge timeout value is 60 seconds.

See [Configuring the voyager.ini file for Charge time-out](#) in the *Voyager Circulation User's Guide* for additional information on Circulation Charge Time-out.

## Single Client Login

The Single Client Login feature allows you, after logging in to one of the Voyager modules, to open subsequent staff applications (modules) without having to re-enter your user name or password. Single Client Login is configured on each personal computer, so you can enable it on some computers and not on others. To enable the Single Client Login feature, set the `SingleLogin` parameter in the `[GlobalLog]` stanza of the `voyager.ini` file to `y`. See [\[GlobalLog\] Stanza on page 4-27](#) for additional configuration information.

Since Single Client Login sends the username and password from module to module, the only visible change after enabling it is that there is no login screen.

The following are issues pertaining to Single Client Login.

- If multiple users are using multiple instances of the same module on the same computer, or if multiple users have access to a single computer, you may not want to enable Single Client Login.
- Single Client Login uses the most recent password entered.
- If a user logs in to one module and later attempts to log in to another module for which that user does not have privileges, the login dialog box is invoked for the second module. After logging in to that module, the second login is used for all subsequent modules opened.

This second login remains in effect until or unless all modules are closed, or another login failure occurs and a new user logs in.
- All normal security checks are performed when passwords and usernames are sent to modules using Single Client Login. For more information on Voyager security see the [Security](#) chapter in the *Voyager System Administration User's Guide*.

## Encryption

---

Encryption is available for secure communication between the Voyager server and the following clients: Acquisitions, CallSlip Daemon, Cataloging, Circulation, Reporter, System Administration, Media Scheduling, and Media System Administration.

Encryption is not available for communication between WebVoyage and the OPAC server, and the Prepackaged Access reports and Oracle. However, Secure Socket Layer (SSL) is an industry standard protocol for encrypting communications between web browsers and web servers. See the *Voyager WebVoyage User's Guide* for more information.

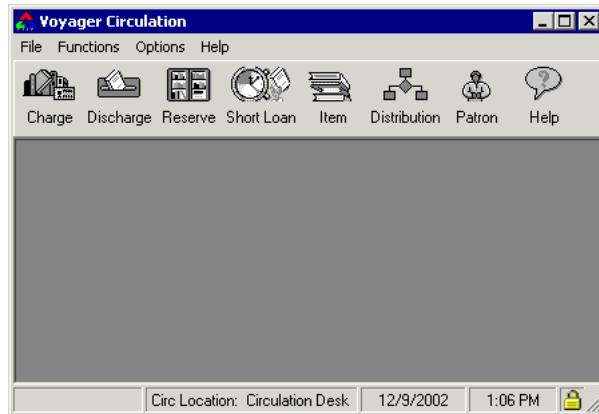


### IMPORTANT:

*To enable encryption, the Encrypt parameter in the [GlobalLog] stanza must be set to Y in both the voyager.ini file on the server and the voyager.ini file on the user's PC.*

The server-side `voyager.ini` file is located in `/m1/voyager/xxxdb/ini`.

When encryption is enabled, you'll notice a padlock icon in the status bar on the bottom of the module window ([Figure 4-37](#)).



**Figure 4-37. Padlock Indicating That Encryption is Enabled**

**NOTE:**

When enabling encryption in a Universal Borrowing environment, if one of the databases you are connecting to is using an earlier version of Voyager, encryption still occurs.

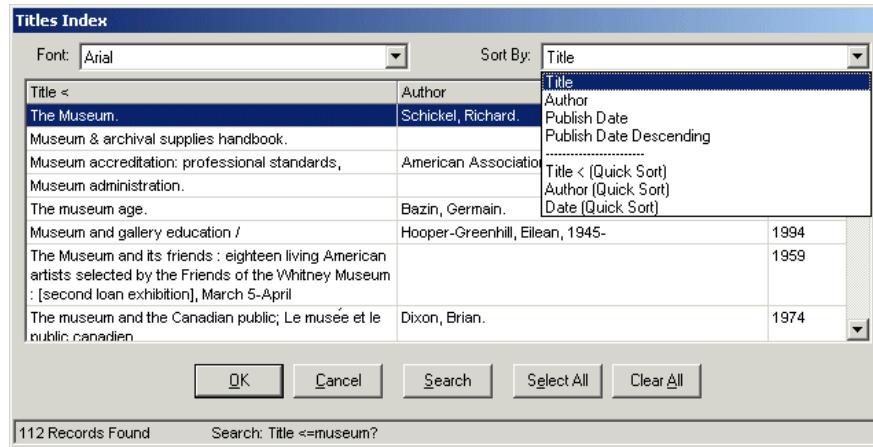
### Staff Client Re-Sort

This feature allows users to re-sort their search results from the Cataloging, Circulation, and Acquisitions modules. The search results may be re-sorted in the following methods:

- Voyager-aware re-sort via drop-down menu.
- Quick Sort via drop-down menu.
- Quick Sort via clickable column header.

The Staff Client Re-sort feature is enabled when the corresponding parameters are set in the [GlobalLog] stanza of the `voyager.ini` file. For more information, see [\[GlobalLog\] Stanza](#) on page 4-27.

Both Voyager-aware and Quick Sorts become available when a bibliographic search (such as a left-anchored title search) returns a list of titles on the results screen (see [Figure 4-38](#)).



**Figure 4-38. Search Results Page with Voyager-Aware and Quick Sort Drop-Down**

**NOTE:**

Re-sorts will *not* be available on search results screens for the following: searches that return a list of headings, browse searches, remote Z39.50 connections, remote connections to non-Voyager catalogs, and remote connections to multiple Voyager catalogs.

### Voyager-Aware Re-Sorts

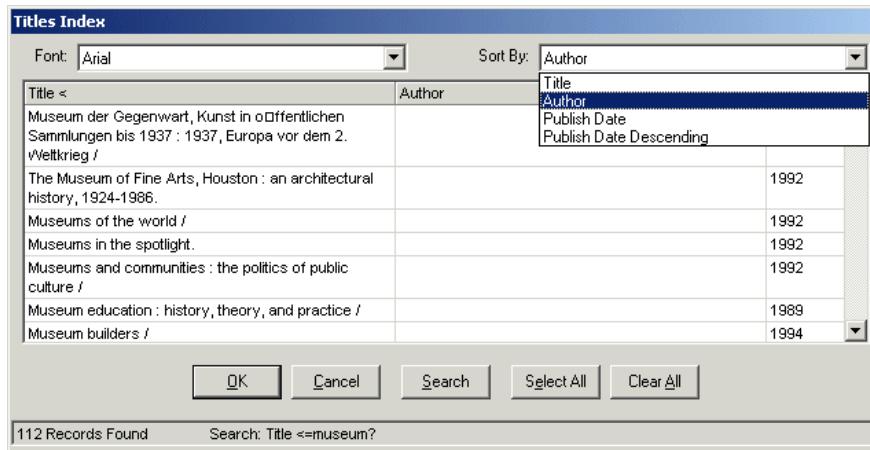
After a search is performed, the following menu items appear in the **Sort By** drop-down menu (see [Figure 4-39](#)) if the Voyager-Aware feature is enabled:

- Title
- Author
- Publish Date
- Publish Date Descending
- Relevance (appears after relevance searches only)

If the user selects any of these items, the search is performed again, sorting the results by the criteria indicated by the selected item. Unless indicated by the menu item, the results are sorted in ascending order, recognizing non-filing characters.

**NOTE:**

Since the system queries the database again, search results may be different if more than 10,000 titles were found in the initial search.



**Figure 4-39. Search Results with Voyager-Aware Drop-Down Menu**

To enable the Voyager-Aware drop-down menu, set the `ServerSortList` parameter in the `[GlobalLog]` stanza. For more information, see [\[GlobalLog\] Stanza](#) on page 4-27.

### Quick Sorts

Quick Sorts allow users to re-sort their search results using a sort criteria and to display the results alphanumerically ignoring non-filing characters.

To perform a quick sort, the user may select a sort criteria from the **Sort By** drop down menu (see [Figure 4-40](#)) or click a column header. The results can be viewed in descending order by selecting the same criteria a second time.

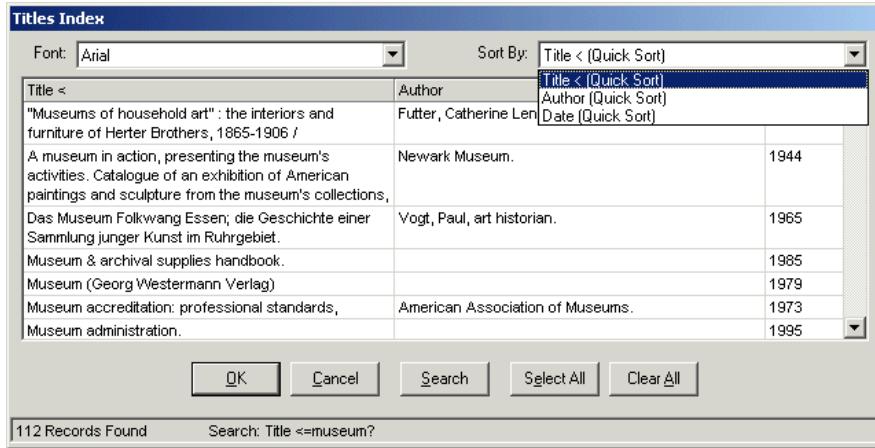
#### NOTE:

The **Sort By** drop-down menu items are defined in the System Administration module by assigning bib text fields to the search results of a search configuration under the **Search** menu. For more information, see the System Administration User's Guide.

The following parameters in the `[GlobalLog]` stanza are used to enable the Quick Sort features:

- The `ASCIISortList` parameter enables the ASCII drop-down menu.
- The `ASCIISortColumn` parameter enables clickable column headers.

For more information, see [\[GlobalLog\] Stanza](#) on page 4-27.



**Figure 4-40.** Search Results with Quick Sort Drop-Down Menu

### MARC Record Posting

This feature allows the operator to use an http POST request to send a MARC record to a web server for display. The POST request contains:

- the MARC record
- the record type: Bibliographic, Authority, or Holdings
- the record ID: BibID, AuthID, or MFHDID depending on the record type

For MARCPosting to work, the [MARC\_POSTing] stanza in the `voyager.ini` file must be configured; and in order to display the POST, the `Pbibredirect.cgi` (for Classic WebVoyage), `holdingsInfo` (for WebVoyage), and `PhttpLinkresolver.cgi` scripts are on the Voyager server to receive the records.

Sending a POST request can be done from any of the Voyager modules where a MARC record is visible.

#### Example [MARC POSTing] Stanza

For example, users could add a key for WebVoyage and the value would be the IP address to their WebVoyage. A sample [MARC\_POSTing] stanza is shown in [Figure 4-41](#) (Classic WebVoyage) and [Figure 4-42](#) (WebVoyage).

```
[MARC POSTing]
WebVoyage="http://<host>:<port>/cgi-bin/Pbibredirect.cgi"
```

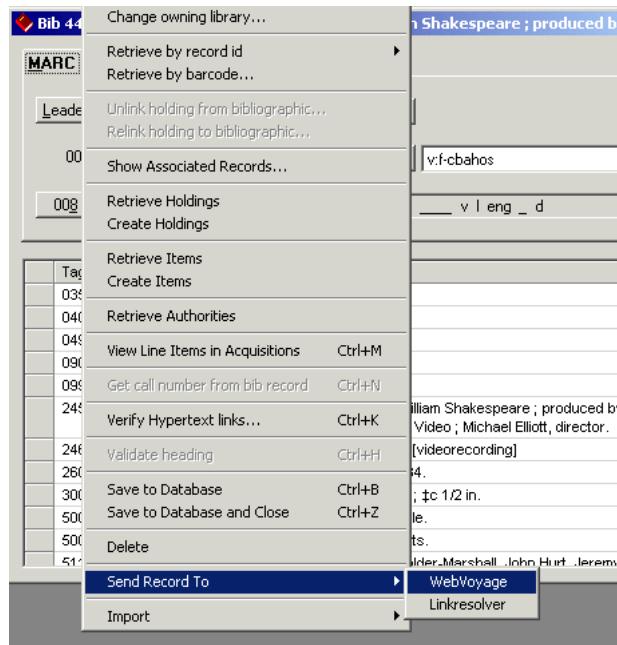
**Figure 4-41.** Sample [MARC POSTing] stanza (Classic WebVoyáge)

```
[MARC POSTing]
WebVoyage="http://<host>:<port>/vwebv/holdingsInfo"
```

**Figure 4-42.** Sample [MARC POSTing] stanza (WebVoyáge)

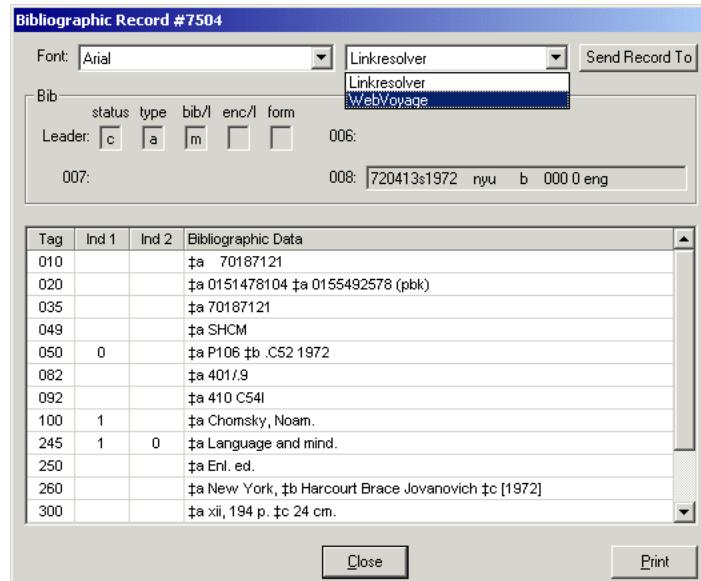
If the `voyager.ini` contains a [MARC POSTing] stanza, from an open record:

- In the Cataloging module, the **Record** menu contains a **Send Record To** option which allows the user to select from the key values in the [MARC POSTing] stanza (Figure 4-43).



**Figure 4-43.** Send Record To option

- In all other modules, a **Send Record To** button appears in the MARC record which allows the user to select from the key values in the [MARC POSTing] stanza ([Figure 4-44](#)).



**Figure 4-44.** Send Record To button and drop-down as seen in the Acquisitions and Circulation modules

**NOTE:**

If more than one key is provided, they are listed alphabetically in the drop-down menu.

### Selecting Send Record To

When the **Send Record To** option (or button) is selected, the system executes the following steps:

- Generates a temporary HTML file with the record, record type, and record ID.
- Passes the post to the browser, opens the form, and displays the record.

**NOTE:**

- The MARC data is encoded in UTF-8.
- Characters are HTML escaped as part of the request.

## Additional Files Installed with the Client Installation

---

In addition to the client executable and help files and the `voyager.ini` file, there are additional folders and files that are installed on the user's computer when installing Voyager. Typically, these files are installed in the `c:\Voyager` directory.

The following folders and files are installed in a complete installation:

- Access Reports folder
  - `Reports.mdb` file  
This is a Microsoft Access database containing prepackaged reports and queries. For more information about this file see [Prepackaged Access Reports Overview](#) in the *Voyager Reporter User's Guide*.
- Catalog folder
  - MARC Template folder
    - \* `EditBook.cfg`
    - \* `EditSerial.cfg`
    - \* `NewBook.cfg`
    - \* `NewSerial.cfg`
    - \* `Template.ini`
  - Tag Table folder
    - \* MARC21 folder
    - \* OCLC folder
    - \* Rlin folder
    - \* `Bmarcfix.cfg`
    - \* `Lang.cfg`
    - \* `Country.cfg`
  - Template folder

These files control templated input using Citation Server®. See the *Voyager Citation Server User's Guide* for more information.

- Tag Table folder
  - \* MARC21 folder
  - \* OCLC folder
  - \* Rlin folder
  - \* `Bmarcfix.cfg`
  - \* `Lang.cfg`
  - \* `Country.cfg`
- Template folder

The tag tables define various types of records, fields, and subfields for bibliographic and authority records. See Appendix A in the *Voyager Cataloging User's Guide* for more information.

- Template folder
  - The template folder contains default templates for bibliographic, holdings, and authority records. For more information about these templates see the *Voyager Cataloging User's Guide*.

```
* MARC21 Repertoire.cfg  
* Special Characters.cfg
```

For information about these files see the *Voyager Cataloging User's Guide*.

- Circulation folder
  - Offline folder
  - Circ.ini
  - Circbib.cfg
  - CircSC.ini

The Offline folder contains any charge files created when using offline charge. See [Offline Charge](#) in the *Voyager Circulation User's Guide*.

The Circ.ini file contains the templates for all the slips and statements that can be printed in the Circulation module. See [CIRC.INI](#) in the *Voyager Circulation User's Guide*.

The Circbib.cfg is used to configure the **Bib** tab of the **Add Bib/Item** dialog box when creating a bibliographic record in the Circulation module. See [Circbib.cfg file - Customizing the Bib Tab Fields](#) in the *Voyager Circulation User's Guide*.

The CircSC.ini file contains information to configure the Voyager Self-check module. See [CIRCSC.INI](#) in the *Voyager Circulation User's Guide*.

- Misc folder

- Acqbib.cfg

This file provides the template for brief bibliographic records created in the Acquisitions module. See the *Voyager Acquisitions User's Guide* for more information.

- Callslip.ini

For information about this file, see the *Voyager Call Slip Daemon User's Guide*.

- EDI\_msg.ini

- Limits.ini

This configures staff search limits. For more information, see the limits.ini section in any of the following user's guides: Acquisitions, Cataloging, or Circulation.

- Mediahelp.ini

- Spinelabel.cfg

This file enables the user to configure separate spine and piece label templates for bibliographic, holding, item and serial records. See [Appendix B, Printing Labels](#) of the *Voyager Circulation User's Guide* for more information.

- Reporter

- Notices

- Reports

This folder contains Microsoft Access files ( .mdb).

- System

This folder contains .dll files, fonts, and so on.

<b>Introduction</b>	<b>5-1</b>
<b>Purpose of this Chapter</b>	<b>5-1</b>
<b>Overview of the Patron Extract Program</b>	<b>5-1</b>
<b>Parameters</b>	<b>5-2</b>
• Running Patron Extract	5-4
<b>Output File Specification</b>	<b>5-4</b>
<b>Additional Files</b>	<b>5-5</b>
• Audit File	5-5
• Error File (Exception File)	5-6



---

## **Introduction**

---

The Patron Extract program allows you to retrieve patron records from the database and save that information in a file.

After running this program an output file of patron records is produced. It also creates an audit and error file.

---

## **Purpose of this Chapter**

---

This chapter provides

- Overview of the patron extract program
- Parameters that govern the program and running the program
- Output file specification
- Additional files created

---

## **Overview of the Patron Extract Program**

---

The patron extract program draws patron records from the database and stores them in a data file that you specify. To do this you must have read-access to the server where your database is stored.

The program extracts patron records to an output file, and creates an audit, and an error file. The default location for these files is /m1/voyager/xxxdb/rpt directory, unless otherwise specified by the operator.

The patron extract program retrieves a maximum of three barcodes per patron and pulls active barcodes first. Therefore, in cases of patron records with four or more barcodes only the first three active barcodes will be imported. Also, the extract program will retrieve only the first ten statistical codes per patron.

The extract program should be run from the /m1/voyager/xxxdb/sbin directory on your server. Running Pptrnextr executes the script which has been configured with your database name, username, and password. Therefore, you will not need to enter that information. If Pptrnextr is entered without any parameters, the program will query the user for the required information.

If you want to run a server activity using a telnet session, and you do not want to perform it interactively, you can use the following command line to make certain that the activity continues, even if the telnet session is lost,

```
nohup <activity with parms> &
```

where <activity with parms> is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

## Parameters

---

The following parameters govern the patron extract program.

**-d      Database name -- not required.**

Automatically specified by the Pptrnextr script. Specifying the database name with this parameter overrides the name in the script.

It is the name of the database that will be accessed. You must have read-access on the server to run the extract program.

**-u      Username and password -- not required.**

Automatically specified by the Pptrnextr script. Specifying the username and password with this parameter overrides the name and password in the script.

It is the username and password for access to the specified database. Enter it in this format: username/password.

**-p      Patron output filename -- not required.**

The name of the output file containing the patron information.

The default is `sif.pxtr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

If you specify *only* a filename it is placed in the `/m1/voyager/xxxdb/local` directory.

However, if you want the file placed in a different directory you must provide the complete path and filename. For example, you could enter

`-p /m1/voyager/xxxdb/log/patron.dat.`

**-a      Audit filename -- not required.**

The name of the audit file where the audit information will be saved.

The default is `log.pxtr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

If you specify *only* a filename it is placed in the `/m1/voyager/xxxdb/local` directory.

However, if you want the file placed in a different directory you must provide the complete path and a filename. For example, you could enter

`-a /m1/voyager/xxxdb/log/audit.ptn.`

**-e      Error filename -- not required.**

The name of the error file where any error records and messages will be saved. If there are no errors an empty file is created.

The default is `err.pxtr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

If you specify *only* a filename it is placed in the `/m1/voyager/xxxdb/local` directory.

However, if you want the file placed in a different directory provide the complete path and a filename. For example, you could enter

`-e /m1/voyager/xxxdb/log/error.ptn.`

**-m      Monitor records to process -- not required.**

The number of records the program will process before a message is displayed, indicating the program's progress.

The default is 100.

**-r      Number of records to process -- not required.**

The number of records that will be processed. To process all the records, type the number zero (0).

The default is 0 (All).

**-s      Extract stub and child patron records -- not required.**

If your database contains stub or child patron records, the -s parameter instructs the system to extract all patron records. If the -s parameter is not used, only parent patron records are extracted.

**-h      Help -- not required.**

Provides online help about the Pptrnextr function. This flag cannot be used with any other parameters.

## **Running Patron Extract**

---

To run the patron extract command, enter the Pptrnextr command from the /m1/voyager/xxxdb/sbin> directory. The command may be entered on one line with parameters as follows:

```
Pptrnextr -p patron.dat -e error.ptn -a audit.ptn -m  
10 -r 1
```

This command instructs the system to create a patron output file called patron.dat. Any errors will be written to the file named error.ptn and any audit information will be written to the file named audit.ptn. The program will display a message for every ten records processed and it will process only one input record.

## **Output File Specification**

---

The output file of the patron extract program uses the Patron Record Standard Interface File (SIF).

For information regarding the specific file format see [Patron Record SIF Format](#) on [page 18-2](#).

The default name is `sif.pxtr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

## Additional Files

---

In addition to the output file of patron records, the patron extract program creates two other files, an audit file and an error file.

### Audit File

---

The audit file contains audit information from running the program.

The default audit filename is `log.pxtr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

This file includes warning messages and information such as:

*GRAND TOTALS*

*Patron Records Received:* 216

*Patron Records Added:* 216

*Records Written To Exception File:* 0

*DETAIL TOTALS*

*Patron Barcode Records:* 258

*Patron Status Records:* 13

*Patron Address Records:* 390

*Patron Phone Records:* 200

*Patron Notes Records:* 52

*END OF REPORT*

## **Error File (Exception File)**

---

The error file contains error records and messages if there were errors when the program ran.

The default is `err.pptr.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

The error file contains error messages if there were errors when the program ran.

**NOTE:**

If there were no errors, a file is created, however it is empty.

<b>Introduction</b>	<b>6-1</b>
<b>Purpose of this Chapter</b>	<b>6-1</b>
<b>Overview the Patron Update Program</b>	<b>6-1</b>
<b>Input File Specification</b>	<b>6-3</b>
<b>Parameters</b>	<b>6-3</b>
• Running Patron Update	6-6
<b>Additional Files</b>	<b>6-7</b>
• Audit File	6-7
• Error File (Exception File)	6-8



### **Introduction**

---

The patron update program allows you to update patron records at any time.

To do this, sites must create an input file of patron records (usually from information provided by the Registrar's Office or other source).

After running this program, audit and error files are created.

### **Purpose of this Chapter**

---

This chapter provides

- Overview of the patron update program
- Input file specification
- Parameters that govern the program and running the program
- Additional files created

### **Overview the Patron Update Program**

---

The patron update program updates existing patron records and adds any new patrons included in the input file. To do this you must have write-access to the server where your database is stored.

The purpose of this program is to update existing patron personal information such as Name, Address, and the like. This program does not update or change Patron barcodes or Patron Groups because of the associated circulation history. However, operators may use the program to add new barcodes or groups. If using Patron Update to create a new barcode or group, be sure to consider expiring the original barcode/group.

The program updates patron records on the server based on the information in the input file. The program will look in the `/m1/voyager/xxxdb/local` directory for the input file if a complete path is not specified.

This program generates audit and error files. The default location for these files is the `/m1/voyager/xxxdb/rpt` directory. However, if a filename is provided by the operator, the default location for the audit and error files is the `/m1/voyager/xxxdb/local` directory.

The program will only input a maximum of three barcodes per patron.

The update program should be run from the `/m1/voyager/xxxdb/sbin` directory on your server. Running `Pptrnupd`t executes the script which has been configured with your database name, username, and password. You will not need to enter that information. If `Pptrnupd`t is entered without any parameters, the program will query the user for the required information.

**⚠️ IMPORTANT:**

*This batch program should only be run at times of low system use or in the evenings. Your staff may note slow response times if run outside of these suggested times.*

If you want to run a server activity using a telnet session, and you do not want to perform it interactively, you can use the following command line to make certain that the activity continues, even if the telnet session is lost,

`nohup <activity with parms> &`

where `<activity with parms>` is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

**NOTE:**

For sites that participate in Voyager's Universal Borrowing or sites that have multiple circulation clusters, the patron update batch job does not update or add any stub patron records. However, the audit file provides information on how many stub records were skipped.

## **Input File Specification**

---

The input file for the Patron update should be supplied in the same format as required in the Voyager patron load program. It is the Patron Record Standard Interface File (SIF).

For information regarding the specific file format see [Patron Record SIF Format on page 18-2](#).

The program will look in the `/m1/voyager/xxxdb/local` directory if a path is not specified with the filename.

## **Parameters**

---

The following parameters govern the patron update program.

**-p      Patron input filename -- required.**

The name of the file that contains the patron information. If you specify the filename but do not specify the full path to the file, the update program will look in the `/m1/voyager/xxxdb/local` directory. Therefore, if the input file is not in the `..../local` directory, you must include the complete path information to the file.

**-d      Database name -- not required.**

Automatically specified by the `Pptrnupd` script. Specifying the database name with this parameter overrides the name in the script.

It is the name of the database that will be updated. You must have write access on the server to run the update program.

**-u      Username and password -- not required.**

Automatically specified by the `Pptrnupd` script. Specifying the username and password with this parameter overrides the name and password in the script.

It is the username and password for access to the specified database. Enter it in this format: `username/password`

**-i      Index type -- not required.**

Indicates the match point for patron records. Enter the appropriate one-letter code of the index type immediately after the flag `-i`.

**S      SSAN**

This causes the program to update patron records based on matching the Social Security Number (SSAN) whether an Institution ID is present or not.

**I      Institution ID**

This causes the program to update patron records based on matching the Institution ID whether an SSAN is present or not.

**D      Program selected**

Based on the data contained in the first input record, this program chooses whether to use the SSAN or the Institution ID to determine whether to update an existing record or add the input item as a new record. It will check first for a SSAN. If one exists, it will use the SSAN to match records for update. If not, it will use the Institution ID to match records for update. If neither exists, the record will be rejected and a choice will be made based on the contents of the next record.

The default is D (Program selected).

**-a      Audit filename -- not required.**

The name of the file where any audit information is saved.

The default audit filename is `log.pupd.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

If you specify a filename *only*, the file is written to the `/m1/voyager/xxxdb/local` directory.

However, if you want the file placed in a different directory provide the complete path and a filename. For example, you could enter

`-a /m1/voyager/xxxdb/log/audit.dat.`

**-e      Error filename -- not required.**

The name of the file where any error messages are saved.

The default error filename is `err.pupd.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

If you specify a filename *only*, the file is written to the `/m1/voyager/xxxdb/local` directory.

However, if you want the file placed in a different directory provide the complete path and a filename. For example, you could enter

**-e /m1/voyager/xxxdb/log/error.dat.**

**-n Notes type -- not required.**

Indicates the type of note that will be created for any that are stored in the input file. Enter the appropriate one-letter code of the note type immediately after the flag -n.

**G General**

This will cause all notes records to be created with the type general.

**P Popup**

This will cause all notes records to be created with the type pop-up.

The default is G (General).

**-o Overlay previous notes -- not required.**

Determines how notes in the input file are handled.

**A Add new notes**

All notes in the input record are added to the database as additional note(s) for this patron.

**R Replace existing notes**

If notes exist in the input file, all existing notes for this patron will be removed from the database and the notes from the input record will be added to the database as the notes for this patron. If notes for this patron do not exist in the input file, existing notes will not be removed.

**I Ignore notes processing**

Does not process any notes (new or existing).

The default is A (Add new notes).

**-x Extended notes processing -- not required.**

Allows multiple notes to be included in the Notes segment of the input record.

These notes are then added to the database as separate notes for the specified patron.

In order for the -x parameter to be able to process multiple notes for a patron, each note must be preceded by a tab character in the input record, including the first note.

**-m      Monitor records to process -- not required.**

The number of records the program processes before a message displays, indicating the program's progress.

The default is 100.

**-r      Number of records to process -- not required.**

The number of records that are processed. To process all the records, enter the number 0.

The default is 0 (All).

**-h      Help -- not required.**

Provides online help about the Pptrnupd function. This flag cannot be used with any other parameters.

## Running Patron Update

---

Before you can run the patron update program you must first create the patron update input file, see [Patron Record SIF Format](#) on [page 18-2](#) in this user's guide.

Users can enter the Pptrnupd command without any parameters. This prompts the program to ask for the parameters interactively from the user.

At the /m1/voyager/xxxdb/sbin> enter **Pptrnupd**, the system prompts for input of any required parameters.

For example, if at /m1/voyager/xxxdb/sbin> you entered **Pptrnupd**, the system would respond with the following prompts: (possible responses in bold)

```
Enter Input Patron File Name:patron.dat
Enter Audit Report File Name:audit.dat
Enter Error Record File Name:error.dat
Enter Index Type: S=SSAN, I=Inst.ID, D=Default:D
Enter Default Notes Type -- G=General, P=Pop-up:G
Enter Notes Overlay Type -- A = Add new notes to old
ones, -- R = Replace old notes with new ones,-- I =
Ignore New Notes:I
Enter Extended Notes Processing: -- Y=Yes, N=No:n
Monitor progress after every ### records(enter ###):0
```

```
Maximum # of records to process, 0 = ALL (enter ###):0
```

If you do not want to enter the command interactively, an example of the Pptrnupd command with parameters might be entered (on one line) as follows:

```
Pptrnupd -p patron.dat -i s -e error.dat -a audit.dat  
-n p -m 1 -r 10
```

This command instructs the system to use the input file named patron.dat, match the patron records the SSAN, to create an error file named error.dat, to create an audit file named audit.dat, and to create pop up notes for any notes in the input file, to display a message for every one record processed and to process ten input records.

## **Additional Files**

---

The patron update program creates an audit (log) file and an error file.

### **Audit File**

---

The audit file contains audit information from running the program.

The default audit filename is log.pupd.YYYYMMDD.HHMM, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes. It is placed in the /m1/voyager/xxxdb/rpt directory.

This file includes warning messages and information such as:

*INVALID RECORD DATA!*

*Attempt to update LOST/STOLEN barcode  
Barcode: 2602*

*This barcode has NOT been updated for  
SSAN:*

*Record written to exception file.  
Patron ID: 3*

*GRAND TOTALS*

*Input Patron Records Received: 10*

*New Patrons Added: 0*

*Existing Patrons Updated: 10*

*Records Written To Exception File: 1*

*DETAIL TOTALS*

<i>New Patron Barcodes Added:</i>	0
<i>Existing Patron Barcodes Updated:</i>	0
<i>New Patron Statuses Added:</i>	0
<i>Existing Patron Statuses Updated:</i>	0
<i>New Patron Addresses Added:</i>	3
<i>Existing Patron Addresses Updated:</i>	10
<i>New Patron Phone records Added:</i>	10
<i>Existing Patron Phone Records Updated:</i>	0
<i>New Patron Notes records Added:</i>	0
<i>Existing Patron Notes Records Updated:</i>	0
<i>Existing Stub Records Skipped:</i>	0

*END OF REPORT*

### **Error File (Exception File)**

The error file contains the records where there were errors when the program ran.

The default error filename is `err.pupd.YYYYMMDD.HHMM`, where YYYY stands for the year, MM for the month, DD for the day, HH for the hour, and MM for the minutes and it is placed in the `/m1/voyager/xxxdb/rpt` directory.

**NOTE:**

If there were no errors, a file is created, however it is empty.

<b>Introduction</b>	<b>7-1</b>
<b>Purpose of this Chapter</b>	<b>7-1</b>
<b>Overview of the Bursar Transfer System</b>	<b>7-2</b>
<b>Steps of the Bursar Transfer System</b>	<b>7-3</b>
<b>Supported One-Way Transfer Types</b>	<b>7-4</b>
• Transferring Total Patron Balances	7-4
• Transferring Itemized Patron Fines/Fees	7-4
<b>Configuration File</b>	<b>7-5</b>
<b>Parameters</b>	<b>7-6</b>
• Running Bursar Transfer	7-7
<b>Output File Specification</b>	<b>7-8</b>
• Bursar SIF Format	7-8
<b>Additional Files</b>	<b>7-9</b>
• Audit File	7-9
• Error File	7-10
<b>Bursar Transfers in the Circulation Module</b>	<b>7-10</b>



### **Introduction**

---

The bursar transfer system allows users to export either the total patron balances, or itemize patron fines/fees information to another system such as a billing agency.

To do this sites must create a configuration file before running the bursar transfer program.

After running this program an output file of patron balances or itemized fines/fees is produced. It also creates audit and error files.

At the completion of the program, in the Voyager database the fine or fee will be considered paid.

**NOTE:**

This program does not extract accrued fines, demerit points, or accrued demerit points. Also, it does not extract fines or fees from stub or child patron records.

### **Purpose of this Chapter**

---

This chapter discusses

- Overview of the bursar transfer system
- General steps of the bursar transfer system

- Supported one-way transfers
- Configuration file creation
- Parameters and an example
- Output file specification
- Additional files
- Bursar transfers in the Circulation module

## **Overview of the Bursar Transfer System**

The bursar transfer allows two types of one-way transfers of patron data. You can transfer the total patron balance or transfer itemized patron fines/fees. To do this you must have read-access to the server where your database is stored .

If transferring the total patron balance, transactions are cumulated to calculate the total balance (debit or credit). Only patron information and the patron's current balance is transferred to the billing agency.

If transferring itemized fines/fees, if beyond a specific amount, individual fine/fee information records and patron information is transferred to the billing agency.

Sites must create a configuration file to govern the transfer of fines. This file allows the linkage between locations and patron groups. The bursar system transfers balances or specific fines/fees that are of the circulation locations designated in the configuration file *and* belonging to patrons that are ACTIVE members of patron groups designated in the configuration file. If a patron has fines but is not an active member of a patron group listed in the configuration file, the patron's fines will not be transferred.

The patron must have a Social Security Number or Institution ID to be transferred.

**NOTE:**

The system will not extract any fines or fees information from any stub (or child) patrons. The system looks for patron records in which the patron ID indicates it is a stub patron record with associated inter-cluster circulation transactions. if found, that record is skipped and its information not transferred.

The bursar program exports information to an output file, and creates an audit and an error file. The default location for these files is  
`/m1/voyager/xxxdb/rpt` directory.

**NOTE:**

For this batch job, the user cannot provide filenames for the output, audit or error files. They are always the default filenames of `sif.burs.yyyymmdd.hhmm`, `log.burs.yyyymmdd.hhmm`, and `err.burs.yyyymmdd.hhmm`.

The export program should be run from the `/m1/voyager/xxxdb/sbin` directory on your server. Running `Pbursar` executes the script which has been configured with your database name, username, and password. Therefore, you will not need to enter that information. If `Pbursar` is entered without any parameters, the program will query the user for the required information.

If you want to run a server activity using a telnet session, and you do not want to perform it interactively, you can use the following command line to make certain that the activity continues, even if the telnet session is lost,

`nohup <activity with parms> &`

where `<activity with parms>` is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

In the Voyager database, once the information is transferred, the fine or fee is considered paid. The fine or fee may be corrected (reduced or waived) after transfer to the billing agency by making a bursar refund debit/error credit. The bursar refund credit will be exported at the next running of the bursar program.

**IMPORTANT:**

*It is STRONGLY RECOMMENDED that this program be run when the library is closed or circulation desk activity is extremely slow or conflicts may occur (that is, fines being paid or posted against while the information is being transferred to the SIF).*

**NOTE:**

This server job can be run using WebAdmin, see [Bursar Transfer](#) on page 30-29.

---

## Steps of the Bursar Transfer System

---

The following steps need to be taken to successfully run the bursar transfer program.

1. Determine the type of transfer, total balances, or itemized fines/fees.

**NOTE:**

The program defaults to itemizing the fines/fees unless the operator includes the -m (Patron mode) parameter to transfer just patron total balances.

2. Create the configuration file. The `Pbursar` batch program transfers information only for the patron groups belonging to the Circulation locations specified in the configuration file.
3. Run the `Pbursar` batch program, using the required parameters and any optional parameters for your sites specific needs. Information on parameters follows.
4. When completed the batch program creates the output file, an audit file summarizing the job, and an error file.
5. Your library will have to create a batch program that will transfer the data in the SIF into your accounting system.
6. If you need to refund the money to patrons, you can use the Circulation module to post an amount to the patron(s) by selecting the **Fine/Fee** window and posting an amount for refund. If money is refunded, run the Bursar batch program again.

## **Supported One-Way Transfer Types**

---

You can transfer fines by patron group and location, as defined in the configuration file, see [Configuration File on page 7-5](#).

There are two types of supported transfers, transferring total patron balances, and transferring itemized patron fines/fees.

### **Transferring Total Patron Balances**

---

Only patron information and the patron's current balance is transferred to the billing agency. If the patron has a credit with the library, individual credits are transferred. The receiving system gets no information about specific charges.

### **Transferring Itemized Patron Fines/Fees**

---

Individual fine/fee information records and patron information is transferred to the billing agency. Specific information about the fines/fees are also transferred to the billing agency.

## Configuration File

---

You must create a configuration file in order for the bursar program to execute. The bursar system reads from this file to determine what information to transfer from the database. It is suggested that this file be created using vi or another server text editor.

The file has two stanzas, [CIRC LOCATIONS] and [PATRON GROUPS]. Each is followed by a list of patron group and location codes (as defined in **System Administration module> Circulation> Patron Groups** and **System Administration module> System> Locations**. See the *Voyager System Administration User's Guide*, [Patron Groups](#) and [Locations](#) sections, for more information.

After the [CIRC LOCATIONS] stanza, list the codes of the circulation locations for which you want to transfer fine and fee information.

After the [PATRON GROUPS] stanza, list the codes for the patron groups whose fine and fee information you want to transfer.

**NOTE:**

The bursar program will only transfer information pertaining to the specified patron groups at those specific locations. The file must contain both circulation location and patron group codes listed in order for information to be extracted. If you want to transfer information for all circulation locations or for all patron groups, enter ALL after the appropriate stanza.

[Figure 7-1](#) shows a sample configuration file.

---

```
[CIRC LOCATIONS]
ALL
[PATRON GROUPS]
UNDERGRAD
GRAD
```

---

**Figure 7-1. Sample Configuration File**

This file instructs the bursar program to transfer fine and fee information for the undergraduate and graduate patron groups at all circulation locations.

You can name the configuration file however you like. Likewise, you can place this configuration file wherever you like on your server. However, in order for the file to be used, you must specify the full path to the file and filename after the -c parameter when running the program.

## Parameters

---

The following parameters govern the bursar transfer program.

**-d      Database name - not required.**

Automatically specified by the Pbursar script. Specifying the database name with this parameter overrides the name in the script.

It is the name of the database that is accessed. You must have read-access on the server to run the extract program.

**-u      Username and password -- not required.**

Automatically specified by the Pbursar script. Specifying the username and password with this parameter overrides the name and password in the script.

It is the username and password for access to the specified database. Enter it in this format: *username/password*

**-c      Configuration file -- required.**

Specify the full path and filename of the bursar configuration file (for example, *-c /m1/voyager/xxxdb/sbin/bursar.cfg*).

A configuration file must exist in order for the bursar program to execute.

**-p      Patron mode -- not required.**

Transfers patron balance totals. The default is to transfer specific fine/fee information, not patron totals.

**-m      Minimum Fine/Fee Amount -- not required.**

Indicates the minimum amount that the fine/fee must be in order to be transferred (in whole numbers).

For example, *-m10* indicates transfer of fines greater than or equal to \$10.00.

If using Patron mode, it indicates the minimum a fine/fee must be in order to be added to the total.

**-i      Interval for Transfer -- not required.**

Indicates the number of days it must be after the fine create date after which the fine may be transferred.

If using Patron mode, it indicates the number of days it must be after the fine create date after which the fine will be added to the total.

**-o      Operator ID -- not required.**

Specifies the operator ID to be used with the bursar postings.

The default is BURSAR

**-t      Test Mode -- not required.**

In test mode, a SIF file is written, but no database updates are made.

**-q      Quiet Mode -- not required.**

Prevents Voyager from prompting for any missing parameters. No hash marks display to indicate the progress of the transfer.

**-v      Version Information -- not required.**

Provides version information about the current `Pbursar` program.

**-h      Help -- not required.**

Provides online help about the bursar transfer system. This flag cannot be used with any other parameters.

---

**Running Bursar Transfer**

Users can enter the `Pbursar` command without any parameters. This prompts the program to ask for the parameters interactively from the user.

At the `/m1/voyager/xxxdb/sbin>` enter `Pbursar`, the system prompts for input of any required parameters.

If you do not want to enter the command interactively, an example of the `Pbursar` command with parameters might be entered (on one line) as follows:

```
Pbursar -c/m1/voyager/xxxdb/sbin/bursar.cfg -p -m10 -oJane -i5
```

This command directs the system use a configuration file called `bursar.cfg`. To extract from the database patron balances, instead of fine/fee balances. Balances below \$10.00 are not be retrieved. The operator ID is Jane. The balance transferred does not include fines and fees less than five days old.

## Output File Specification

The output file of the bursar transfer program is the `Bursar SIF` file. The SIF file is named `sif.burs.yymmdd.hhmm`. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

### Bursar SIF Format

[Table 7-1](#) describes the format of the `Bursar SIF` file.

**Table 7-1. Bursar SIF Format**

	Length	Field	Comment
1	9	Patron Social Security Number (no dashes)	If the Social Security Number and Institution ID is missing, log patron to error file and process next.
10	30	Patron Institution ID	
40	10	Patron Home Location, Location Code	Leave blank if not found.
50	10	Patron Group Code	Use the first active patron group found for a patron that was selected as a run parameter. (A patron may have more than one active patron, we need to pick one that was specified for the run).
60	10	Fine/Fee Type Code	Patron total runs will use hard-coded value of PTRNTOTAL.
70	10	Fine/Fee Create Date	ccyy.mm.dd, patron total runs will use the run date.
80	10	Voyager Location Code	Location where the fine or fee was generated.

**Table 7-1. Bursar SIF Format**

	<b>Length</b>	<b>Field</b>	<b>Comment</b>
90	1	+/-	Credit (-) the patron's billing agency account or debit (+) the account. A plus sign means that the patron account was credited in Voyager. A minus indicates that the fine or fee was corrected after an initial transfer.
91	16	Net due for fine or fee or patron balance.	In Voyager base currency, use decimal separators and decimal places--no thousands separators.
107	1000	Description of the Fine/ Fee	This string will be right padded with blanks. For patron balance runs, this will always be blank.
1107		Record Length	

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## **Additional Files**

In addition to the output file of patron total balances or itemized fines/fees information, the bursar transfer program creates two other files, an audit file and an error file.

### **Audit File**

The audit file contains information about the bursar transfer.

It is named `log.bursar.yymmdd.hhmm`, where y is year, m is month, d is day, h is hour, and m is minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

The file includes the current date, time and summary transaction information.

The summary transaction information for patron balance transfers includes the number of patrons credited, total patron credits processed, the number of patron debited, total patron debits processed, total patrons processed, and net debits and credits.

The summary transaction information for individual fine/fee transfer includes the number of patrons credited, total patron credits processed, the number of patrons debited, total patron debits processed, total patrons processed, and net debits and credits, but is also broken apart by fine/fee type code.

[Table 7-2](#) provides an example of the audit file, the fee column displays the fine/fee types and each row contains a summary for each type.

**Table 7-2. Audit Log SIF Format**

<b>Fee</b>	<b>Credits</b>	<b>Patrons</b>	<b>Debits</b>	<b>Patrons</b>	<b>Net</b>	<b>Patrons</b>
F1	\$101.23	10	\$12.50	1	\$88.73	11
F2	\$50.01	5	\$0	0	\$50.01	5
Total	\$151.24	15	\$12.50	1	\$138.73	16

**NOTE:**

For this batch job, the user cannot provide a filename for the audit file.

---

### Error File

---

The error file stores the message regarding any error records that cannot be written to the SIF file.

The file is named `err.burs.yymmdd.hhmm`, where y is year, m is month, d is day, h is hour, and m is minutes. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

**NOTE:**

If there were no errors, a file is created, however it is empty.

---

## Bursar Transfers in the Circulation Module

---

Within the circulation module one can:

- view bursar transfer information
- give a bursar refund

Fines/fees that have been transferred can be viewed in the **Circulation module> Patron record> Fines/Fees History tab.**

The bursar refund credits the charge up to the amount paid by the bursar transfer and the balance due on that transaction becomes a negative amount. This negative amount is displayed as a current charge until the bursar program is run again and can debit this amount and transfer it to the bursar.

To refund an amount,

1. From a selected bursar transfer item on the **Fines/Fees History** tab, click the **Post** button.
2. Select a posting type of **Bursar Refund** and enter the amount.

This amount is considered a Bursar refund in the next run of the batch program. The next Circulation session, the amount displays as a Forgive.



---

## Circulation Batch Jobs

# 8

<b>Introduction</b>	<b>8-1</b>
<b>Purpose of This Chapter</b>	<b>8-2</b>
<b>Update Shelving Status (Circjob 1)</b>	<b>8-3</b>
<b>Archive and Expire Call Slip Requests (Circjob 8)</b>	<b>8-4</b>
<b>Export OPAC Requests (Circjob 26)</b>	<b>8-4</b>
<b>Archive Short Loans (Circjob 27)</b>	<b>8-5</b>
<b>Purge Universal Borrowing (UB) Patron Stub Records (Circjob 29)</b>	<b>8-5</b>
<b>Accrued Fines and Demerits (Circjob 30)</b>	<b>8-6</b>
<b>Patron Suspension (Circjob 31)</b>	<b>8-7</b>
<b>Universal Borrowing (UB) Request Promotion (Circjob 32)</b>	<b>8-8</b>
<b>Update Remote Circulation Cluster Cache (Circjob 33)</b>	<b>8-9</b>
<b>Place Items on Active Course Reserve List (Circjob 34)</b>	<b>8-10</b>
<b>Place Recalls and Holds for Items on Active Course Reserve List (Circjob 35)</b>	<b>8-12</b>
<b>Take Items on Inactive Course Reserve List Off Reserve (Circjob 36)</b>	<b>8-14</b>
<b>Forgive Demerits (Circjob 37)</b>	<b>8-16</b>
<b>Retain Patron IDs (Circjob 38)</b>	<b>8-17</b>

<b>Patron Purge (Circjob 39)</b>	<b>8-23</b>
• Patron Purge Files	8-23
• Patron Purge Command Line (Circjob 39)	8-25
<b>Forgive Fines by Patron ID (Circjob 40)</b>	<b>8-27</b>
• Running circjob 40 from the command line	8-27
• Error Logging for Batch Forgive Jobs	8-29
circjob.log	8-29
err.forgive.YYYYMMDD.HHMM	8-29
• Audit Report for Batch Forgive Jobs	8-29
<b>Forgive Fines by Create Date (Circjob 41)</b>	<b>8-30</b>
• Running circjob 41 from the command line	8-30
• Error Logging for Batch Forgive Jobs	8-32
circjob.log	8-32
err.forgive.YYYYMMDD.HHMM	8-32
• Audit Report for Batch Forgive Jobs	8-33
<b>Forgive Fines by Patron Group and Expire Date (Circjob 42)</b>	<b>8-33</b>
• Running circjob 42 from the command line	8-33
• Error Logging for Batch Forgive Jobs	8-35
circjob.log	8-35
err.forgive.YYYYMMDD.HHMM	8-36
• Audit Report for Batch Forgive Jobs	8-36
<b>Synchronize Patron Counters for Universal Borrowing (Circjob 43)</b>	<b>8-37</b>
• Frequency of Use	8-37
• How to Access	8-37
Server Access	8-38
Command Line Access	8-39
WebAdmin Access	8-40
• Circjob Log	8-40
• Error Conditions	8-40

### Introduction

---

Circulation batch processing is used in a variety of ways. This chapter describes maintenance-related batch processing. Separately, another subset of the circjobs can be run that generate data used as input files to Microsoft Access to generate standardized reports and notices. For more information regarding the Microsoft Access reporting options, see the *Voyager Reporter User's Guide*.

Circulation batch jobs can be run from the `/m1/voyager/xxxdb/sbin` directory on the server or by using [WebAdmin](#). For information regarding WebAdmin to run these circulation batch jobs, see [Using WebAdmin on page 30-7](#).

From the server, the circjobs can be executed from the command line or run as part of a cron job. For repetitive, maintenance-type batch jobs, you may find it more convenient to schedule the work and run the executables as part of a cron job.

Depending on the task being performed, some circjobs are prompted and require your dynamic input when executed.

After running these jobs, an entry in the `circjob.log` file is made.

## Purpose of This Chapter

---

This chapter explains circulation batch jobs that allow institutions to perform actions on multiple records in order to maximize the efficiency of both the user and the software.

For example, batch jobs 34, 35, and 36 can be used to place items on course reserve lists and change their status as follows:

- Active to Inactive.
- Available to On Hold or Recalled.
- Inactive to off the reserve list entirely.

The circulation-related batch jobs discussed in this chapter are as follows:

- Circjob 1: Update Shelving Status. See [Update Shelving Status \(Circjob 1\) on page 8-3](#).
- Circjob 8: Archive and Expire Call Slip Requests. See [Archive and Expire Call Slip Requests \(Circjob 8\) on page 8-4](#).
- Circjob 26: Export OPAC Requests. See [Export OPAC Requests \(Circjob 26\) on page 8-4](#).
- Circjob 27: Archive Short Loans. See [Archive Short Loans \(Circjob 27\) on page 8-5](#).
- Circjob 29: Purge UB Patron Stub Records. See [Purge Universal Borrowing \(UB\) Patron Stub Records \(Circjob 29\) on page 8-5](#).
- Circjob 30: Accrued Fines and Demerits. See [Accrued Fines and Demerits \(Circjob 30\) on page 8-6](#).
- Circjob 31: Patron Suspension. See [Patron Suspension \(Circjob 31\) on page 8-7](#).
- Circjob 32: Universal Borrowing Request Promotion. See [Universal Borrowing \(UB\) Request Promotion \(Circjob 32\) on page 8-8](#).
- Circjob 33: Update Remote Circulation Cluster Cache. See [Update Remote Circulation Cluster Cache \(Circjob 33\) on page 8-9](#).
- Circjob 34: Place Items on Active Course Reserve List. See [Place Items on Active Course Reserve List \(Circjob 34\) on page 8-10](#).
- Circjob 35: Place Recalls and Holds for Items on Active Course Reserve List. See [Place Recalls and Holds for Items on Active Course Reserve List \(Circjob 35\) on page 8-12](#).
- Circjob 36: Take Items on Inactive Course Reserve List Off Reserve. [Take Items on Inactive Course Reserve List Off Reserve \(Circjob 36\) on page 8-14](#).

- Circjob 37: Forgive Demerits. See [Forgive Demerits \(Circjob 37\) on page 8-16](#).
- Circjob 38: Retain Patron IDs. See [Retain Patron IDs \(Circjob 38\) on page 8-17](#).
- Circjob 39: Patron Purge. See [Patron Purge \(Circjob 39\) on page 8-23](#).
- Circjob 40: Forgive Fines by Patron ID. See [Forgive Fines by Patron ID \(Circjob 40\) on page 8-27](#).
- Circjob 41: Forgive Fines by Create Date. See [Forgive Fines by Create Date \(Circjob 41\) on page 8-30](#).
- Circjob 42: Forgive Fines by Patron Group and Expiration Date. See [Forgive Fines by Patron Group and Expire Date \(Circjob 42\) on page 8-33](#).
- Circjob 43: Synchronize Patron Counters for Universal Borrowing. See [Synchronize Patron Counters for Universal Borrowing \(Circjob 43\) on page 8-37](#).

## **Update Shelving Status (Circjob 1)**

This batch job updates the shelving status of items from discharged to not charged, depending on the interval specified in the Voyager System Administration module.

The frequency with which this job should be run depends on your site's shelving policies.

To run the job, at the `sbin>` prompt, enter

```
Pcircjob -j1
```

The `circjob.log` file should include the day, date, time entry, and possibly the following messages.

```
Thu Mar 21 13:18:13 2002 Update Shelving Status...
```

```
Thu Mar 21 13:18:14 2002 ...COMPLETED
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities on page 30-24](#).

## Archive and Expire Call Slip Requests (Circjob 8)

---

This batch job archives any requests with the status Processed, Not Found, or Expired and changes the status of unprocessed requests to Expired. An expire period of zero would indicate that the request should never expire.

It expires unprocessed requests after the interval specified in the Voyager System Administration module has elapsed. These requests will display in WebVoyage until the job archives them.

The frequency with which this job should be run depends on the interval your site

has set up in **System Administration> Call Slip Print Groups> Call Slip**

**Definitions.** See the *Voyager System Administration User's Guide*, for more information.

To run the job, at the sbin> prompt, enter

```
Pcircjob -j8
```

The circjob.log file should include the day, date, time entry, and possibly the following messages.

```
Thu Mar 21 13:18:13 2002 Archive and Expire Callslip  
Requests...
```

```
Thu Mar 21 13:18:14 2002 ...COMPLETED
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on [page 30-24](#).

## Export OPAC Requests (Circjob 26)

---

This batch job writes to a file on the server all OPAC requests that have been routed to the flat file (for example, item-level flat file requests and ARTEmail photocopy and loan requests). This job does not create any notes or notices. It simply writes the request information to a file on your server. The format in which these are written can be customized. For more information see the *Voyager System Administration User's Guide*, regarding request forms.

To run the job, at the sbin> prompt, enter

```
Pcircjob -j26
```

The circjob.log file should include the day, date, time entry, and possibly the following messages.

*Thu Mar 21 13:18:13 2002 Export OPAC Requests...*

*Thu Mar 21 13:18:14 2002 ...COMPLETED*

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

## **Archive Short Loans (Circjob 27)**

---

Updates the status of current short loan requests and archives expired requests in the Short Loans Search List. Short Loans allow your patrons to place a reservation for an item for a specific time period, and then pick it up and charge it out at the scheduled time. See the *Voyager Circulation User's Guide* for more information about short loans.

This batch job should be put into a cron so that it will run automatically.

To run the job, at the `sbin>` prompt, enter

`Pcircjob -j27`

The `circjob.log` file should include the day, date, time entry, and possibly the following messages.

*Thu Mar 21 13:18:13 2002 Archive Short Loans..*

*Thu Mar 21 13:18:14 2002 ...COMPLETED*

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

## **Purge Universal Borrowing (UB) Patron Stub Records (Circjob 29)**

---

This batch job purges any empty UB patron stub records that exist in the database.

The system looks for patron records in which the Patron ID indicates a UB patron, and then checks for the existence of fines and fees, holds, UB requests, and charged items. The `-f` parameter instructs the system to check for historical fines and fees.

If the system finds an empty stub patron record and you did not include the -f parameter, the stub record is deleted from the database. If the system finds an empty stub record and you did include the -f parameter, the system checks for historical fines and fees. If fines ever existed in the stub patron record, the system skips to the next stub patron record. If they never existed, the record is deleted.

Also, operators may use the -m parameter to retain stub records that were manually mapped.

To run the job without parameters, at the sbin> prompt, enter

```
Pcircjob -j29
```

To run the job with both parameters, at the sbin> prompt, enter

```
Pcircjob -j29 -f -m
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on [page 30-24](#).

See the *Voyager Circulation Clusters User's Guide* and the *Voyager Universal Borrowing User's Guide* for more information on Purge UB Patron Stubs (Circjob 29).

## **Accrued Fines and Demerits (Circjob 30)**

---

This circulation batch job calculates accrued fines or demerits.

If **Use Demerits** is not selected in **System Administration> Circulation> Miscellaneous**, then this job calculates the accrued fine. This is the fine that would be assessed if an overdue item were discharged at the time the job is run.

If **Use Demerits** is selected in **System Administration> Circulation> Miscellaneous**, then this job calculates the accrued demerits. This is the number of demerits that would be assessed if an overdue item were discharged at the time the job is run.

These calculations take into account the grace period that has been set up in the **System Administration> Circulation> Policy Definitions> Matrix> Settings** tab for the specific patron group/item type combination.

When the job runs, for each overdue item, it:

- deletes any previous accrued fine or demerits,
- calculates the fine or demerits for the item as if it were discharged immediately, for all patron records where fines/lost item fees apply,

- determines and applies the fine/fee type reason: accrued fine or accrued demerit
- report any accrued demerits or fines for stub patrons back to their home patron records.

**NOTE:**

There is no mailed notification of accrued fines or demerits. Patrons can check their accrued fines or demerits by asking at a Circulation desk, or by accessing their Patron Information in WebVoyage.

This job should be run daily and can be added to any current crons you may have set up.

To run the job, at the `sbin>` prompt, enter

```
Pcircjob -j30
```

The `circjob.log` file should include the day, date, time entry, and possibly the following messages.

```
Thu Mar 21 13:18:13 2002 Update Accrued  
Fines.Demerits...
```

```
Thu Mar 21 13:18:14 2002 ...COMPLETED
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

See the *Voyager Circulation User's Guide* for more information about demerits.

## Patron Suspension (Circjob 31)

---

This circulation batch job applies one or more suspensions to a patron's record if they have reached or exceeded the threshold number of demerits (**Max demerits**). The suspension period takes into account exception dates listed in the `exceptdates.cfg` file. This file should be created even if your site does not have any exception dates. See the Demerits appendix in the *Voyager Circulation User's Guide* for more information.

When the job runs, for each patron, it:

- sums the demerits and compares that sum to the threshold number (if the patron belongs to more than one patron group it will use the lowest threshold amount). If the threshold limit is reached or exceeded then,
- applies the suspension, taking into account any exception dates, and

- subtracts the value of the max demerits for the patron record, then
- checks for additional real demerits that reach or exceed the threshold amount. If so, another suspension will be applied with the begin date of that suspension following the end date of the prior suspension. This occurs until there are no longer enough real demerits to invoke a suspension
- if a stub patron is suspended, subtract the value of max demerits on the home patron record.

**NOTE:**

Suspension dates should be calculated from the system date (current date) or the patron suspension end date, whichever value is greater.

This job should be run daily and can be added to any current crons you may have set up.

To run the job, at the sbin> prompt, enter

```
Pcircjob -j31
```

The circjob.log file includes information and messages about how this job ran.

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on [page 30-24](#).

See the *Voyager Circulation User's Guide* for additional information regarding demerits.

---

## Universal Borrowing (UB) Request Promotion (Circjob 32)

---

If you have configured Universal Borrowing Request Promotion, Universal Borrowing requests (including patron UB eligible local callslips) which cannot be filled at one holding library (database) will automatically be sent to other holding libraries (databases) for fulfillment. This eliminates the need for manual promotion of requests by patrons. This circjob should be run on the patron's home library database.

Request Promotions are performed by a circjob (circjob 32). This circjob promotes requests automatically to other (specified) databases, adhering to all existing validation checks for borrowing requests. It also keeps track of the status of the requests it promotes at the different databases. The callslip status will reflect successful promotions.

Circjob 32 requires a configuration file, the `promoteXXX.cfg` file, where `XXX` is the call slip print group. Use the `-p` parameter to specify the `promoteXXX.cfg` file you want. See the *Voyager Universal Borrowing User's Guide* for more information on the contents of the `promoteXXX.cfg` file.

This job should be run daily and can be added to any current crons you may have set up.

**NOTE:**

The Universal Borrowing Request Promotions circjob takes some time, and is resource-intensive, so choose a suitable time to run it (suitable both to you and your fellow UB partners, whose databases you may be searching).

To run the job, at the `sbin>` prompt, enter

```
Pcircjob -j32 -ppromoteXXX.cfg
```

This is assuming that the `promoteXXX.cfg` file is in the `/m1/voyager/xxxdb/ini` directory on your server, if not add the full path.

The `circjob.log` file contains information on how the job ran, including error information. The `circjob.log` file is in the `/m1/voyager/xxxdb/rpt` directory. See the *Voyager Universal Borrowing User's Guide* for more information on the possible error messages, along with a description of each.

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

See the *Voyager Universal Borrowing User's Guide* for more information about this batch job.

## **Update Remote Circulation Cluster Cache (Circjob 33)**

---

For sites that participate in Voyager's Universal Borrowing this batch job will contact all remote UB databases and retrieve their Circulation Clusters and patron groups such that they are available for use in the Circulation module or WebVoyage.

Once contacted the tables in the local database are updated with information from the remote databases. The system then checks that no patron groups mappings are orphaned. That is, if a remote site had deleted a patron group the system updates that information in the local database.

**⚠️ IMPORTANT:**

*This job should be run after your upgrade to have the Circulation Cluster and Patron Group information available locally.*

The job can be run as often as the site wants. It can be set up on a cron. Lastly, it should be run any time that a remote database has altered its patron groups if you are aware of this change.

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on [page 30-24](#).

For additional information regarding Circjob 33, Update Remote Circulation Cluster Cache, see the *Voyager Circulation Clusters User's Guide*.

## Place Items on Active Course Reserve List (Circjob 34)

---

This batch job sets the On Reserve flag for items on active course reserve lists at one time. This job should be run as necessary.

The parameters available when running this job are described in [Table 8-1](#).

**NOTE:**

Required entries are indicated by an (R) in the Name field. If only one of the below values is present, the system uses the current date as the other. If neither is provided, the system uses the current date as the start date and an open-ended end date.

**Table 8-1. Parameters for Circjob 34**

Option	Name	Description
-s	Start date (R)	Beginning date used to find active reserve list.
-e	End date (R)	End date used to find active reserve list.

When the job runs, it:

- finds all reserve lists whose effective date is within the start and end dates provided
- finds all items on these lists where the On Reserve flag is No
- changes the On Reserve flag to Yes

- changes the temporary location of the item to the location associated with the reserve list
- changes the temporary item type of the item to the item type associated with the reserve list
- creates the output file.

With the exception of the output file, this is the same as the manual process done in the Circulation module.

**NOTE:**

If an item is on multiple reserve lists, it is placed on reserve using the temporary location and temporary item type of the first reserve list the job finds.

The output file is named `circjob.onreserve.yyyymmdd.hhmmss`, where yyyy is the year, mm is the month, dd is the day, hh is the hour, mm is the minute, and ss is the seconds that the file was created. This file is located in the `/m1/voyager/xxxdb/rpt` directory.

The format of the output file is described in [Table 8-2](#).

**NOTE:**

Required entries are indicated by an (R) in the Name field.

**Table 8-2. Description of output SIF from Circjob 34**

Name	Value	Description
On Reserve (R)	Y or N	Yes (Y) or No (N) is the On Reserve status of the item after it has been processed.
Item Id (R)	Numeric	The Voyager item id of the item in question.

To run the job, at the `sbin>` prompt, enter

```
Pcircjob -j34 -sYYYYMMDD -eYYYYMMDD
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

For information regarding Course Reserve, see the *Voyager Circulation User's Guide*.

## Place Recalls and Holds for Items on Active Course Reserve List (Circjob 35)

This batch job places Holds and Recalls for items on active course reserve lists at once. This job should be run as necessary.

The parameters available when running this job are described in [Table 8-3](#).

**NOTE:**

Required entries display an (R) in the Name field. If only one of the below start or end date values is present, the system uses the current date as the other. If neither is provided, the system uses the current date as the start date and an open-ended end date.

**Table 8-3. Parameters for Circjob 35**

Option	Name	Description
-s	Start date	<p>Beginning date used to find active reserve list.</p> <p><b>NOTE:</b> If only one of these is present, use the current date as the other. If neither are provided the system uses the current date as the start date and an open ended end date.</p>
-e	End date	<p>End date used to find active reserve list.</p> <p><b>NOTE:</b> If only one of these is present, use the current date as the other. If neither are provided the system uses the current date as the start date and the an open ended end date.</p>
-i	Operator id (R)	Operator id of person running the batch job.
-L	Location code (R)	This is the location code of the circulation desk that will be the create and pickup locations for the holds that the job creates.

**Table 8-3. Parameters for Circjob 35**

Option	Name	Description
-D	Due Date	This sets the due date for recalled items. If not provided, the system uses the recall rules defined in the System Administration module. See the <i>Voyager System Administration User's Guide</i> for more information.
-P	Print Location Code	This is the print location code used when generating recall notices. It must be a valid print location.  If specified, a recall notice is appended to the current <code>crcnotes.*.inp</code> file, where the * is the print location code of the print location given.  If not specified, notices are produced after running Circjob 3. See the <i>Voyager Reporter User's Guide</i> for more information about creating input files for circulation.

When the job runs it:

- finds all reserve lists whose effective date is within the start and end dates provided
- finds all items on these lists where the On Reserve flag is No
- if the item is charged to a patron, place an administrative recall request using the information specified in the command line, generate a recall notice
- if the item is not charged, place an administrative hold request using the information specified in the command line
- creates the output file.

**NOTE:**

If an item has pending holds or recall requests, cancel those requests. Also if the item is On Hold for a patron cancel that as well.

The output file is named `circjob.holdrecall.yyyymmdd.hhmmss`, where yyyy is the year, mm is the month, dd is the day, hh is the hour, mm is the minute, and ss is the seconds that the file was created. This file is located in the `/m1/voyager/xxxdb/rpt` directory.

The format of the output file is described in [Table 8-2](#)

**NOTE:**

Required entries display an (R) in the Name field.

**Table 8-4. Description of Output SIF from Circjob 35**

Name	Value	Description
Type of Request (R)	H, R, or N	Hold, Recall, or No request placed.
Item Id (R)	Numeric	The Voyager item id of the item in question.
Patron Id	Numeric	The Voyager patron id for the patron. Required to place request.
Hold/Recall Id	Numeric	The Voyager Hold/Recall id of the newly-placed request. Required to place request.

To run the job, at the sbin> prompt, enter

```
Pcircjob -j35 -sYYYYMMDD -eYYYYMMDD -icrcclerk  
-LRESV -DYYYYMMDD -PRESV
```

**NOTE:**

This job can be run using WebAdmin. See [Circulation Utilities](#) on [page 30-24](#).

For information regarding Course Reserve, see the *Voyager Circulation User's Guide*.

---

## Take Items on Inactive Course Reserve List Off Reserve (Circjob 36)

---

This batch job takes inactive course reserve lists off reserve. This job should be run as necessary.

The parameters available when running this job are described in [Table 8-5](#).

**NOTE:**

Required entries display an (R) in the Name field. If only one of the below values is present, the system uses the current date as the other. If neither is provided, the system uses the current date as the start date and an open-ended end date

**Table 8-5. Parameters for Circjob 36**

Option	Name	Description
-s	Start date	Beginning date used to find active reserve list.
-e	End date	End date used to find active reserve list.

When the job runs it:

- finds all reserve lists whose expire date is within the start and end dates provided
- finds all items on these lists where the On Reserve flag is Yes
- for each of these items
  - if the item is on another reserve list whose effective dates include the current date, leave it on reserve
  - if not set the On Reserve flag to No and remove the temporary location

The output file is named `circjob.offreserve.yyyymmdd.hhmmss`, where yyyy is the year, mm is the month, dd is the day, hh is the hour, mm is the minute, and ss is the seconds that the file was created. This file is located in the `/m1/voyager/xxxdb/rpt` directory.

The format of the output file is described in [Table 8-6](#).

**NOTE:**

Required entries display an (R) in the Name field

**Table 8-6. Description of output SIF from Circjob 36**

Name	Value	Description
On Reserve (R)	Y or N	Yes (Y) or No (N) is the On Reserve status of the item after it has been processed.
Item Id (R)	Numeric	The Voyager item id of the item in question.

To run the job, at the `sbin>` prompt, enter

```
Pcircjob -j36 -sYYYYMMDD -eYYYYMMDD
```

**NOTE:**

This job can be run using WebAdmin. See [Circulation Utilities](#) on [page 30-24](#).

For information regarding Course Reserve, see the *Voyager Circulation User's Guide*.

## **Forgive Demerits (Circjob 37)**

---

This batch job forgives an eligible patron's actual demerits. It is run on an entire database, therefore it includes all the Circulation Clusters within that database. Administrators provide the number of demerits, which may be a fraction, that they want to forgive for each eligible patron. Typically this job will be run at the end of semesters and over holidays.



**IMPORTANT:**

*This batch job does not forgive fines or fees.*

Patrons are eligible to have their demerits forgiven if they meet this criteria.

- have actual demerits
- the patron's number of total demerits, actual and accrued, does not exceed that patron group demerit threshold, that is the patron is not blocked. When multiple patron groups exist use the patron group with the lowest threshold.
- the patron is not suspended
- the patron has no outstanding fines or fees

**NOTE:**

There is no record of these demerits, and changes made by this batch job are permanent.

This job should be run after Pcircjob 30, Accrued Demerits, since it establishes eligibility to have demerits forgiven.

For sites participating in Universal Borrowing, this job may be run against databases in which demerit usage is not enabled. Also, reliable network connection must exist among Voyager databases so the home patron record may be updated. If there is a network interruption, the software will rollback changes in the local cluster for the patron currently being processed.

The -q parameter is used to define the number of demerits to be forgiven.

When the job runs it:

- connects to the remote databases
- finds all eligible patrons
- forgives the number of demerits provided.
- updates the patron record
- creates circjob log file

The `circjob.log` file includes the session beginning and end times, it lists any stub patrons whose demerits were not forgiven due to an inability to connect to their home database, and the total number of patron who had demerits forgiven.

To run the job instructing the system to forgive 25 demerits, at the `sbin>` prompt, enter

```
Pcircjob -j37 -q25
```

**NOTE:**

This job can be run using WebAdmin, see [Circulation Utilities](#) on page 30-24.

For additional information regarding demerits, see the *Voyager Circulation User's Guide*.

## Retain Patron IDs (Circjob 38)

This circulation batch job allows libraries to define the number of patrons (patron IDs) whose identification information is retained for item circulation history and item distribution history providing retention is configured in the System Administration module.

Configuring retention means that the following are selected on the **Systems - Miscellaneous** workspace:

- **Retain Patron Id for Circ History** check box, and
- **Retain Patron Id for Distribution History** check box.

There are three scenarios determining the number of patrons whose information is retained:

- If a **Retain Patron Id...** check box is selected and the job is not run, all the associated patron information is retained.
- If a **Retain Patron Id...** check box is selected and the job is run, the specified number of patron IDs are retained.
- If a **Retain Patron Id...** check box is not selected and the job is run, no patron information is retained.

When specifying the number of patrons to keep, the system retains the most recent patron IDs. Recency is determined by using the discharge date and the distribution date.

Essentially, for the last  $x$  number of patron IDs, the link between the item and the patron's information is retained. Those patron IDs not kept means that the link between the item and the patron's ID is broken and there is no way to retrieve that information.

For example, libraries that want to keep the circulation history of just the last few patrons who charged an item can define the number of patron IDs to retain at three. Therefore, the last three patrons who charged the item are kept. This information displays in the Circulation module to operators with appropriate authority to view patron information.



### **IMPORTANT:**

*To keep a consistent history, administrator's should request the same number of patron IDs to be retained each time the job is run.*

This job can be run as necessary. It can also be added to a cron.

The parameter available when running this job is described in [Table 8-7](#). An (R) designates a required parameter.

**Table 8-7. Parameters for Circjob 38**

Parameter	Name	Description
-T (R)	Number of IDs to retain	<p>Number of patrons whose identification information is retained for an item's circulation or distribution history.</p> <p>If zero, no patrons are retained.</p> <p><b>NOTE:</b> To retain all patron IDs, simply select the appropriate box on the <b>System - Miscellaneous</b> workspace in the System Administration module. There is no need to run this job.</p>

**NOTE:**

When an item is discharged, an entry is made in the CIRCULATION\_TRANSACTIONS\_ARCHIVE table and, if retention is configured, patron identification information is also added to this table. Similarly, when an item is distributed and, if retention is configured, patron identification information is added to the DISTRIBUTION\_TRANSACTION table.

When Circjob 38 runs:

- for each item in the CIRCULATION\_TRANSACTIONS\_ARCHIVE table, only the most recent x number of patron ID's (where x is the number provided with the -T parameter) retain the link between the item and the patron information. For all other IDs the link is broken.
- for each item in the DISTRIBUTION\_TRANSACTION table, only the most recent x number of patron ID's (where x is the number provided with the -T parameter) retain the link between the item and the patron information. For all other IDs the link is broken.

There is no output file for this job. However, an entry in the `circjob.log` file is made indicating that the job was run and the number of records updated ([Figure 8-1](#)). In the example in [Figure 8-1](#), there were 16 total records, 5 records are retained as the job parameter requests, and 11 records are updated.

```
Thu Apr  8 11:25:09 2004 RetainPatron...
Thu Apr  8 11:25:10 2004 11 Records updated in the table circ_trans_archive
Thu Apr  8 11:25:10 2004 0 Records updated in the table distribution_transaction
Thu Apr  8 11:25:10 2004 ...COMPLETED
```

**Figure 8-1. Example of the Message in Circjob.log file after Running Circjob 38**

Patron IDs that are retained for circulation history can be viewed in the Circulation module, if the operator has the authority to view Circulation history. See creating, editing, and deleting circulation profiles in the *Voyager System Administration User's Guide* and the *Voyager Circulation User's Guide* for additional information.

Patron IDs that are retained for distribution history are accessible in the DISTRIBUTION\_TRANSACTIONS table using SQL only. They are not visible in the circulation module.

This job can be run using a command line (see [Procedure 8-1, Running Circjob 38 from the Command Line](#)), interactively on the server (see [Procedure 8-2, Running Circjob 38 Interactively](#)), or in Webadmin (see [Circulation Utilities on page 30-24](#)).



### Procedure 8-1. Running Circjob 38 from the Command Line

Use the following to run Circjob 38 using a command line.

1. Log on to your Voyager server and navigate to the `/m1/voyager/xxxdb/sbin` directory.

2. At the `sbin>` prompt, enter `Pcircjob -j38 -T#` and press enter.

The # is the number of patron IDs you wish to retain (see [Figure 8-2](#)) In this example five IDs are retained.

---

```
/m1/voyager/xxxdb/sbin $ Pcircjob -j38 -T5
Initializing...
RetainPatron...
11 Records updated in the table circ_trans_archive
0 Records updated in the table distribution_transaction
...COMPLETED
/m1/voyager/xxxdb/sbin $
```

---

**Figure 8-2. Running Circulation Batch Job 38 Using the Command Line**

Result: The job runs retaining the specified number of patron's information. The system returns to the `sbin` prompt.

The information displayed when the job runs is also entered into the `circjob.log` file ([Figure 8-8](#)).

---

```
Thu Apr  8 11:25:09 2004 RetainPatron...
Thu Apr  8 11:25:10 2004 11 Records updated in the table circ_trans_archive
Thu Apr  8 11:25:10 2004 0 Records updated in the table distribution_transaction
Thu Apr  8 11:25:10 2004 ...COMPLETED
```

---

**Figure 8-3. Circjob.log file after Running Circjob 38**

---



---

### Procedure 8-2. Running Circjob 38 Interactively

---

Use the following to run Circjob 38 interactively.

1. Log on to your Voyager server and navigate to the `/m1/voyager/xxxdb/sbin` directory
2. At the `sbin>` prompt enter `Pcircjob`

Result: The screen displays the possible circulation batch jobs to run (see [Figure 8-4](#)).

```
/mi/voyager/>>>db/shin $ Pcircjob
Initializing...
1 -- Update Shelving Status
2 -- Overdue Notices
3 -- Recall Notices
4 -- Fine Fee Notices <new fines/fees only>
5 -- Hold Recall Available Notices
6 -- Hold Recall Cancelled Notices
7 -- Courtesy <Due> Notices
8 -- Archive and Expire Call Slip Requests

12 -- All Daily Jobs <1 - 8>
14 -- Fine Fee General Statements <all patrons with an outstanding balance>

19 -- Hold Shelf Expired Report <date range required>
20 -- Reserved Items Active Report <date range required>
21 -- Reserved Items Expired Report <date range required>
22 -- Missing in Transit Report
23 -- Transaction Statistics Report <date range required>
24 -- Transaction Exceptions Report
25 -- Global Transaction Statistics Report <date range required>

26 -- Export OPAC Requests
27 -- Archive Short Loans
28 -- Automatic Orders for Distribution Items

29 -- Purge UB Patron Stub Records

30 -- Accrued Fines/Demerits
31 -- Apply Suspensions

32 -- UB Request Promotion

33 -- Update Remote Circulation Cluster Cache

34 -- Place Items on Active Course Reserve List
35 -- Place Recalls and Holds for Items on Active Course Reserve List
36 -- Take Items on Inactive Course Reserve Lists Off Reserve

37 -- Forgive Patron Demerits

38 -- Retain Patron IDs <the number of patron IDs to retain required>
99 -- Quit

Process Job #?
```

Figure 8-4. Available Circulation Batch Jobs

3. At the Process Job #? prompt enter 38 (see [Figure 8-5](#)) and press enter.

```
/mi/voyager/xxxxdb/sbin $ Pcircjob  
Initializing...  
 1 -- Update Shelving Status  
 2 -- Overdue Notices  
 3 -- Recall Notices  
 4 -- Fine Fee Notices <new fines/fees only>  
 5 -- Hold Recall Available Notices  
 6 -- Hold Recall Cancelled Notices  
 7 -- Courtesy <Due> Notices  
 8 -- Archive and Expire Call Slip Requests  
  
12 -- All Daily Jobs <1 - 8>  
14 -- Fine Fee General Statements <all patrons with an outstanding balance>  
  
19 -- Hold Shelf Expired Report <date range required>  
20 -- Reserved Items Active Report <date range required>  
21 -- Reserved Items Expired Report <date range required>  
22 -- Missing in Transit Report  
23 -- Transaction Statistics Report <date range required>  
24 -- Transaction Exceptions Report  
25 -- Global Transaction Statistics Report <date range required>  
  
26 -- Export OPAC Requests  
27 -- Archive Short Loans  
28 -- Automatic Orders for Distribution Items  
  
29 -- Purge UB Patron Stub Records  
  
30 -- Accrued Fines/Demerits  
31 -- Apply Suspensions  
  
32 -- UB Request Promotion  
  
33 -- Update Remote Circulation Cluster Cache  
  
34 -- Place Items on Active Course Reserve List  
35 -- Place Recalls and Holds for Items on Active Course Reserve List  
36 -- Take Items on Inactive Course Reserve Lists Off Reserve  
  
37 -- Forgive Patron Demerits  
  
38 -- Retain Patron IDs <the number of patron IDs to retain required>  
99 -- Quit  
  
Process Job #? 38
```

---

Figure 8-5. Providing the Circulation Batch Job Number

4. At the Please enter the number of patron IDs to retain: prompt, enter the number of IDs you want to retain and press enter. In this example five IDs are retained (see [Figure 8-6](#)).
- 

```
Process Job #? 38  
Please enter the number of patron IDs to retain : 5
```

---

Figure 8-6. Example of Retaining Five Patron IDs

Result: The job runs retaining the specified number of patron's information. The system returns to the sbin prompt.

```
RetainPatron...
11 Records updated in the table circ_trans_archive
0 Records updated in the table distribution_transaction
...COMPLETED
```

**Figure 8-7. On Screen Display when Running Circulation Batch Job 38**

The information displayed on the screen when the job runs is also entered into the circjob.log file ([Figure 8-8](#)).

```
Thu Apr  8 11:25:09 2004 RetainPatron...
Thu Apr  8 11:25:10 2004 11 Records updated in the table circ_trans_archive
Thu Apr  8 11:25:10 2004 0 Records updated in the table distribution_transaction
Thu Apr  8 11:25:10 2004 ...COMPLETED
```

**Figure 8-8. Circjob.log File after Running Circjob 38**

## Patron Purge (Circjob 39)

This circulation job allows operators to delete patron records that are marked for deletion from the database. An expiration date and/or a purge date must be set in the patron's record to be considered for deletion (see the *Circulation User's Guide*, Patron Records for information on setting the dates).

Patron records that match the criteria specified on the command line are removed from the database as long as the patron records do not contain any of the exceptions listed in [Table 8-8](#).

### Patron Purge Files

When Circjob 39 is run, the system updates the circjob.log file with a report and creates many files under the /m1/voyager/xxxdb/rpt directory. These files include exception files, a deleted file, and a log file.

For each exception that is encountered while running this job, the system creates a SIF file named patron.name.yyyyymmddhhmmss where:

- *name* is the exception name from [Table 8-8](#)
- *yyyymmdd* is the date (year, month and day)

- *hhmmss* is the time (hours, minutes, seconds)

Each exception file contains the patron records that could not be deleted for that particular exception. For more information on the structure of the patron records, see [Patron Record SIF Format on page 18-2](#).

If any records are deleted during this job, the system creates a SIF file named `patron.purge.deleted.yyyymmddhhmmss` where `yyyymmddhhmmss` is the date in year (`yyyy`), month (`mm`), and day (`dd`). This file contains all patron records that were deleted during the running of this job.

Also, the system creates a file named `patron.purge.log.yyyymmddhhmmss` where `yyyymmddhhmmss` is the date in year (`yyyy`), month (`mm`), and day (`dd`). This file contains a list of eligible patron record IDs and their deletion status.

The following table lists the reasons for which a patron record *cannot* be purged from the database and the exception file name.

**Table 8-8. Patron Purge Exceptions/Names**

Name	Exception
sqlerror	SQL Error
itemscharged	current charged items
holdrecall	current hold/recall
finesfees	current fees
demerits	current demerits
exception	a circulation transaction exception has been logged against the patron
bookings	current bookings
callslip	current call slip requests
routinglist	patron is on routing list
shortloan	current short loans
proxy	patron is proxy for another patron
ubcharge	current UB charges
ubfine	current UB fines
ubrequest	current UB requests
ubdemerits	current UB demerits
stubexists	a stub record exists for the patron

**Table 8-8. Patron Purge Exceptions/Names**

Name	Exception
historicalfines	<p>historical fines or fees</p> <p><b>NOTE:</b> If global parameter, <b>Delete Patrons with Historical Fines</b>, is set to <b>Y</b>, patron records with historical fines/fees are eligible for deletion. For more information on setting global variables, see the <i>Voyager System Administration User's Guide</i>, Circulation Configuration.</p>

**Patron Purge Command Line (Circjob 39)**

To run Circjob 39, enter the following command line from the `sbin>` prompt:

```
Pcircjob -j39 -b -a YYYY-MM-DD -z
```

This job allows the operator to set the criteria for purging patrons from the database. Parameters may be entered on the command line to specify the following criteria:

- check purge date (default) or expiration date
- use system date (default) or specified date
- run in test mode (default) or deletion mode

The parameters available when running this job are described in [Table 8-9](#).

**Table 8-9. Parameters for Circjob 39**

Parameters	Description
<code>-j &lt;job #&gt;</code>	Specify <b>39</b> for the patron purge job (required).
<code>-a &lt;YYYY-MM-DD&gt;</code>	<p>Specify a date <code>YYYY-MM-DD</code> to override the system date.</p> <p><b>NOTE:</b> If this parameter is not specified, the system date is used.</p>

**Table 8-9. Parameters for Circjob 39**

Parameters	Description
-b	<p>Remove patron records by comparing the patron expiration date with the system date or a specified date.</p> <p><b>NOTE:</b> If this parameter is not specified, the patron records are removed on the basis of the patron purge date by default.</p>
-z	<p>Must be specified to remove patron records from the database.</p> <p><b>NOTE:</b> By default, this job runs in test mode. If this parameter is not specified, specification files are created, but the patron records are not removed from the database.</p>



### Procedure 8-3. Purging Patron Records from the Database (Circjob 39)

This procedure deletes all patron records that have a purge date that occurs before the system date.

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory.
2. At the prompt, type **Pcircjob -j39 -z** and then press enter.

Result: The job updates the **circjob.log** file and creates the following time-stamped files:

- patron purge exception files (a SIF file for each exception that occurs)
- a patron purge log file which lists the activity of the job
- a patron purge deleted file (a SIF file containing all deleted records)

The following information displays ([Figure 8-2](#)):

```
/m1/voyager/xxxdb/sbin $ Pcircjob -j 39 -z  
Initializing...  
Patron Purge...  
...COMPLETED
```

Figure 8-9. Purging Patron Records from the Database (Circjob 39)

## **Forgive Fines by Patron ID (Circjob 40)**

This batch job forgives fines and fees of patrons based on a supplied list of patron IDs.

This job is typically run when patron records need to be purged from the system and existing fines or fees on patron accounts are keeping them from being deleted.

After the patron ID list is run through the system, the system forgives all the fines and fees for the IDs on the list and generates an error log and audit report (see [Error Logging for Batch Forgive Jobs on page 8-35](#) and [Audit Report for Batch Forgive Jobs on page 8-36](#)). Patron IDs can then be purged from the system using Circjob 39.

Circjob 40 can be run from the server using [Procedure , Running circjob 40 from the command line](#) or [Procedure 8-4, Running circjob 40 interactively](#). It can also be run through the WebAdmin client. See [Procedure 30-11, Forgiving fines by patron ID \(circjob 40\), on page 30-24](#).



### **IMPORTANT:**

The file that you direct circjob 40 to process should include a list of carriage-return-delimited patron IDs.

### **Running circjob 40 from the command line**

To run Circjob 40, enter the following command line from the `sbin>` prompt:

```
Pcircjob -j40 -i operatorID -t -p forgive.txt
```

where

- **-i operatorID** = up to 10 alphanumeric characters (required)
- **-t** = run in test mode (optional), and
- **-p forgive.txt** = input file name (required) preceded by location/path if different from /m1/voyager/xxxdb/local.



#### Procedure 8-4. Running circjob 40 interactively

---

Use the following to run circjob 40 using prompts from the server.



##### **IMPORTANT:**

*Optional parameters such as running the job in test mode are not available when you use interactive prompts.*

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory.
2. At the sbin> prompt, enter **Pcircjob**  
Result: The system displays numbered options for all available circulation batch jobs.
3. At the **Process Job #?** prompt, enter **40**.
4. At the **Operator ID?** prompt, enter your operator ID. (This tags the job with your identification number in the system.)
5. At the **Enter Input Patron File Name:** prompt, enter the file name. If the file is not in the /m1/voyager/xxxdb/local directory, precede the file name with the directory path.

Result: The system runs the batch job and returns the log and audit report.

---

For information on running the log using the WebAdmin client, see [Procedure 30-11, Forgiving fines by patron ID \(circjob 40\)](#), on page [30-24](#).

## Error Logging for Batch Forgive Jobs

---

Errors for circjob 40 are logged with other jobs in the comprehensive circjob.log as well as in the err.forgive.YYYYMMDD.HHMM log. Both can be found in the /m1/voyager/xxxdb/rpt directory.

### **circjob.log**

The circjob.log records

- the start and end time of the processing and
- any errors that prevent the job from running accurately (such as missing parameters, the inability to locate an input file).

An example of a circjob.log entry when no errors are present is:

```
Mon Sep 17 09:42:29 2007 Forgive Fines/Fees...
Mon Sep 17 09:42:29 2007 ...COMPLETED
```

### **err.forgive.YYYYMMDD.HHMM**

The error report, err.forgive.YYYYMMDD.HHMM includes

- patron ID
- patron last name
- patron first name, and
- error description.

Errors include issues retrieving patron IDs or forgiving eligible fines.

## Audit Report for Batch Forgive Jobs

---

The audit report, log.forgive.YYYYMMDD.HHMM, is deposited in the /m1/voyager/xxxdb/rpt directory and includes

- patron ID, last name, first name, and amount of forgiven fine(s)
- totals for all patrons combined:
  - number of patrons with fines eligible to be forgiven
  - number of fines successfully forgiven
  - amount of fines successfully forgiven
  - number of patron records with at least one error.

An example of a log.forgive.YYYYMMDD.HHMM file:

Patron 97: Deer, Beverly forgiven 1 fees totalling 10.00

Patron 98: Fegner, Karen forgiven 1 fees totalling 20.00

Patron 99: Hobat, Matt forgiven 1 fees totalling 30.00

Patron 100: Julley, Eli forgiven 1 fees totalling 40.00

4 patrons had fines or fees eligible to be forgiven.

4 patrons had fines or fees that were forgiven.

Total amount forgiven: 100.00

0 patrons had at least one error.

## **Forgive Fines by Create Date (Circjob 41)**

This batch job forgives fines and fees of patrons based on the date the fines and fees were incurred. An operator can further limit the scope by location.

To run the job directly on the server, use [Procedure , Running circjob 41 from the command line](#) or [Procedure 8-5, Running circjob 41 interactively](#). To run the job using the WebAdmin client, see [Procedure 30-12, Forgiving fines by date created \(circjob 41\)](#), on page [30-25](#).

### **Running circjob 41 from the command line**

To run circjob 41, enter the following command line from the sbin> prompt, substituting your own variables for the text in italics. (Bracketed text is optional.)

```
Pcircjob -j41 -i operatorID -r -t -L main -s 1998-09-01 -e 2002-08-31
```

where

- **-i *operatorID*** = up to 10 alphanumeric characters (required)
- **-r** = a flag that determines whether fines are forgiven for (Universal Borrowing) stub records (optional).

Enter the **-r** to have stub record fines forgiven.

- **-t** = run in test mode (optional).
- **-L *main*** = forgive fines from these Circulation locations (optional).

The default is **All**. To specify multiple location codes, enter them separated by commas.

**IMPORTANT:**

The location code must be typed exactly the same (including uppercase/lowercase punctuation) as it is stored in Voyager System Administration. Voyager System Administration permits codes with the same spelling but different uppercase/lowercase usage to be stored concurrently. For example, law, Law, and LAW may all be valid location codes stored concurrently in Voyager System Administration.

- **-s** = fines created after start date (*yyyy-mm-dd*). Required.
- **-e** fines created before end date (*yyyy-mm-dd*). Required.

**Procedure 8-5. Running circjob 41 interactively**

Use the following to run circjob 41 using prompts from the server.

**IMPORTANT:**

Optional parameters such as forgiving stub record fines and limiting which Circulation locations to forgive are not available when you use interactive prompts.

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory.

2. At the sbin> prompt, enter **Pcircjob**.

Result: The system displays numbered options for all available circulation batch jobs.

3. At the Process Job #? prompt, enter **41**.

4. At the Operator ID? prompt, enter your operator ID. (This tags the job with your identification number in the system.)

5. At the Start Date? prompt, enter the start date for forgiving fines (required) in the following format: *yyyy-mm-dd*.

6. At the End Date? prompt, enter the end date for forgiving fines (required) in the following format: *yyyy-mm-dd*.

Result: The system runs the job and forgives all fines incurred within the specified dates for the specified groups.

---

Once the system forgives fines and fees incurred within the specified parameters, you can run batch jobs (such as job 39) that delete patron records from the system.

**NOTE:**

For information on running circjob 41 using the WebAdmin client, see [Procedure 30-12, Forgiving fines by date created \(circjob 41\)](#), on page [30-25](#).

## Error Logging for Batch Forgive Jobs

---

Errors for circjob 41 are logged with other jobs in the comprehensive `circjob.log` as well as in the `err.forgive.YYYYMMDD.HHMM` log. Both can be found in the `/m1/voyager/xxxdb/rpt` directory.

### `circjob.log`

The `circjob.log` records

- the start and end time of the processing and
- any errors that prevent the job from running accurately (such as missing parameters, the inability to locate an input file).

An example of a `circjob.log` entry when no errors are present is:

```
Mon Sep 17 09:42:29 2007 Forgive Fines/Fees...
Mon Sep 17 09:42:29 2007 ...COMPLETED
```

### `err.forgive.YYYYMMDD.HHMM`

The error report, `err.forgive.YYYYMMDD.HHMM` includes

- patron ID
- patron last name
- patron first name, and
- error description.

Errors include issues retrieving patron IDs or forgiving eligible fines.

## Audit Report for Batch Forgive Jobs

---

The audit report, `log.forgive.YYYYMMDD.HHMM`, is deposited in the `/m1/voyager/xxxdb/rpt` directory and includes

- patron ID, last name, first name, and amount of forgiven fine(s)
- totals for all patrons combined:
  - number of patrons with fines eligible to be forgiven
  - number of fines successfully forgiven
  - amount of fines successfully forgiven
  - number of patron records with at least one error.

An example of a `log.forgive.YYYYMMDD.HHMM` file:

```
Patron 97: Deer, Beverly forgiven 1 fees totalling 10.00
Patron 98: Fegner, Karen forgiven 1 fees totalling 20.00
Patron 99: Hobat, Matt forgiven 1 fees totalling 30.00
Patron 100: Julley, Eli forgiven 1 fees totalling 40.00
```

```
4 patrons had fines or fees eligible to be forgiven.
4 patrons had fines or fees that were forgiven.
Total amount forgiven: 100.00
0 patrons had at least one error.
```

## Forgive Fines by Patron Group and Expire Date (Circjob 42)

---

This batch job forgives fines and fees of patrons based on a combination of patron group and patron expiration date.

### Running circjob 42 from the command line

---

To run circjob 42, enter the following command line from the `sbin>` prompt, substituting your own variables for the text in italics. (Bracketed text is optional.)

```
Pcircjob -j42 -i operatorID -r -t -L main, media -s 1998-09-01 -e 2002-08-31 -g CAS01 CAS02
```

where

- `-i operatorID` = up to 10 alphanumeric characters (required).

- **-r** = a flag that determines whether fines are forgiven for stub records (optional).  
Enter the **-r** to have stub record fines forgiven.
- **-t** = run in test mode (optional).
- **-L** = forgive fines from these Circ locations (optional).  
The default is **All**. To specify multiple location codes, enter them separated by commas.

**IMPORTANT:**

*The location code must be typed exactly the same (including uppercase/lowercase punctuation) as it is stored in Voyager System Administration. Voyager System Administration permits codes with the same spelling but different uppercase/lowercase usage to be stored concurrently. For example, law, Law, and LAW may all be valid location codes stored concurrently in Voyager System Administration.*

- **-s** = patron record expiration date range start (**yyyy-mm-dd**). Required.
- **-e** patron record expiration date range end (**yyyy-mm-dd**). Required.
- **-g** = patron groups to include in batch write-off.

**Procedure 8-6. Running circjob 42 interactively**

---

Use the following to run circjob 42 using prompts from the server.

**IMPORTANT:**

*Optional parameters such as forgiving stub record fines and limiting which Circulation locations to forgive are not available when you use interactive prompts.*

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory.
2. At the **sbin>** prompt, enter **Pcircjob**.  
Result: The system displays numbered options for all available circulation batch jobs.
3. At the **Process Job #?** prompt, enter **42**.

4. At the `Operator ID?` prompt, enter your operator ID. (This tags the job with your identification number in the system.)
5. At the `Start Date?` prompt, enter the start date for the patron expiration date range for forgiving fines (required) in the following format: `yyyy-mm-dd`.
6. At the `End Date?` prompt, enter the end date for the expiration date range for forgiving fines (required) in the following format: `yyyy-mm-dd`.
7. At the `Forgive fines from these patron groups:` prompt, enter a comma-delimited list of patron group codes to forgive or enter **ALL**.

**Result:** The system runs the job and forgives all fines incurred within the specified dates for the specified groups.

Once the system forgives fines and fees incurred within the specified parameters, you can run batch jobs (such as job 39) that delete patron records from the system.

**NOTE:**

For information on running Circjob 42 using the WebAdmin client, see [Procedure 30-13, Forgiving fines by patron group and expiration date \(circjob 42\)](#), on page [30-27](#).

## Error Logging for Batch Forgive Jobs

Errors for circjob 42 are logged with other jobs in the comprehensive `circjob.log` as well as in the `err.forgive.YYYYMMDD.HHMM` log. Both can be found in the `/m1/voyager/xxxdb/rpt` directory.

### **`circjob.log`**

The `circjob.log` records

- the start and end time of the processing and
- any errors that prevent the job from running accurately (such as missing parameters, the inability to locate an input file).

An example of a `circjob.log` entry when no errors are present is:

```
Mon Sep 17 09:42:29 2007 Forgive Fines/Fees...
Mon Sep 17 09:42:29 2007 ...COMPLETED
```

#### **err.forgive.YYYYMMDD.HHMM**

The error report, err.forgive.YYYYMMDD.HHMM includes

- patron ID
- patron last name
- patron first name, and
- error description.

Errors include issues retrieving patron IDs or forgiving eligible fines.

#### **Audit Report for Batch Forgive Jobs**

---

The audit report, log.forgive.YYYYMMDD.HHMM, is deposited in the /m1/voyager/xxxdb/rpt directory and includes

- patron ID, last name, first name, and amount of forgiven fine(s)
- totals for all patrons combined:
  - number of patrons with fines eligible to be forgiven
  - number of fines successfully forgiven
  - amount of fines successfully forgiven
  - number of patron records with at least one error.

An example of a log.forgive.YYYYMMDD.HHMM file:

```
Patron 97: Deer, Beverly forgiven 1 fees totalling 10.00
Patron 98: Fegner, Karen forgiven 1 fees totalling 20.00
Patron 99: Hobat, Matt forgiven 1 fees totalling 30.00
Patron 100: Julley, Eli forgiven 1 fees totalling 40.00
```

4 patrons had fines or fees eligible to be forgiven.

4 patrons had fines or fees that were forgiven.

Total amount forgiven: 100.00

0 patrons had at least one error.

## Synchronize Patron Counters for Universal Borrowing (Circjob 43)

---

Circjob 43 (-j43) is designed to address patron transaction data discrepancies that occur in a Universal Borrowing (UB) environment. Running circjob 43 ensures that Universal Borrowing (UB) data in the home cluster is the most up-to-date by reading the currently available data from remote clusters. It cycles through all the UB-eligible, non-stub patron records in the cluster, connects to each remote cluster, and updates appropriate tables at the home cluster using data from the remote clusters.

In a Universal Borrowing environment you may notice that UB counters in the patron table do not match the real transaction activity throughout the environment. Local and remote tables are intended to be updated as transactions occur.

However, it is possible for transaction activity to occur when a remote location is unavailable that results in patron transaction counters becoming out of sync with reality. This may lead to patrons encountering some circulation block behavior. Your site's experience with these symptoms determines your use of circjob 43.

When run, circjob43 cycles through all the UB-eligible, non-stub patron records in the cluster and connects to each remote cluster; and the appropriate tables are updated at the home cluster using data from remote clusters.

### Frequency of Use

---

Circjob 43 may be run at any time and as frequently as your site decides it is appropriate.

The level of Universal Borrowing activity may be a determining factor.

Remote databases need to be available when synchronizing with those databases using circjob 43.

### How to Access

---

Circjob 43 may be run/accessed in the following manner:

- From the server menu list (see [Server Access](#) on [page 8-38](#)).
- In command line mode on the server (see [Command Line Access](#) on [page 8-39](#)).
- From WebAdmin (see [WebAdmin Access](#) on [page 8-40](#)).

## Server Access

To access the circjob 43 option (Synchronize Patron counters for Universal Borrowing) from the server menu list, run `Pcircjob` which is located in `/m1/voyager/xxxdb/sbin/` where `xxxdb` is your database name. This displays the list of available circulation batch jobs. See [Figure 8-10](#).

---

```
37 -- Forgive Patron Demerits  
38 -- Retain Patron IDs (the number of patron IDs to retain required)  
39 -- Patron Purge  
40 -- Forgive Fines/Fees by Patron ID (Patron ID file required)  
41 -- Forgive Fines/Fees by Date (date range required)  
42 -- Forgive Fines/Fees by Patron Group and Expiration Date  
43 -- Synchronize Patron Counters for Universal Borrowing  
; 99 -- Quit
```

Process Job #?

---

**Figure 8-10. Circjob 43 option on the server**

At the Process Job #? prompt, enter 43 and any additional parameters. See [Table 8-10](#) for a list of parameters.

**Table 8-10. Circjob 43 Parameters**

Parameter	Description
-t	<p>Test mode parameter.</p> <p>All changes to the database are rolled back after the reporting is complete for each patron. Nothing actually changes.</p> <p>The test mode report should be the same as the report from a normal run except for the possible discrepancy in the order in which patrons are processed.</p> <p><b>NOTE:</b> Test mode must be a selected parameter. Circjob 43 does not prompt for it.</p>
-C	<p>Cluster parameter. Specify -C followed by the circulation cluster code.</p> <p>Use this parameter to select a single cluster from a multiple-cluster environment to process synchronization.</p> <p>If you do not specify -C in a multiple-cluster environment, the circjob prompts for a cluster code.</p> <p>If the local database does not have circulation clusters enabled, no other parameters are required. The -C should not be specified. However if it is specified, you must supply the correct circulation cluster code for that cluster.</p>

**TIP:**

*A complete list of parameters for all circulation batch jobs can be displayed by running Pcircjob -h from the sbin directory.*

### Command Line Access

Circjob 43 can be run/entered in command line mode.

For example, enter the following at the /m1/voyager/xxxdb/sbin/ prompt (where xxxdb is your database name) to run circjob 43 in test mode:

```
Pcircjob -j43 -t
```

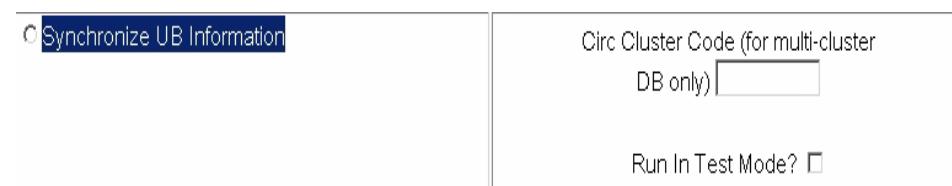
Or using the cluster code, see the following example:

```
Pcircjob -j43 -C C01DB
```

## WebAdmin Access

Circjob 43 can be accessed from WebAdmin.

Select **Reports and Notices** from the Circulation menu list in WebAdmin to display the circjob 43 option. See [Figure 8-11](#).



**Figure 8-11.** Circjob 43 option from WebAdmin

[Table 8-10](#) on [page 8-39](#) describes the cluster parameter and the test mode parameter.

For more information regarding WebAdmin, see Chapter [30](#) titled “[WebAdmin](#).”

## Circjob Log

The `log.ubsynch.YYYYMMDD.HHMM.` is the primary log file for circjob 43. It reports details on every stub analyzed.

Consistent with other circjob processing, general status reports and error messages are appended to the `circjob.log` located in `/m1/voyager/xxxdb/rpt/` where `xxxdb` is your database name.

## Error Conditions

The system registers an error condition in the following situations:

- Invalid Cluster Code option specified for `-c`.
- Parameter error preventing access to the local database.
- Local database has no clusters.
- Local database not enabled for Universal Borrowing.
- Failure to connect to a remote database.
- Failure to retrieve data from a remote library.
- Failure to update local data for a patron.





---



## **Media Scheduling Batch Jobs**

---

**9**

<b>Introduction</b>	<b>9-1</b>
<b>Purpose of this Chapter</b>	<b>9-1</b>
<b>Retain Patron IDs (Mediajob 5)</b>	<b>9-1</b>



---

### Introduction

---

The media scheduling batch jobs discussed here are jobs that do not get processed using the reporter module. They do not produce a report or notice, but are still important for media services.

**NOTE:**

Those media scheduling batch jobs that produce an input file (Mediajobs 1 - 4) and are consequently run through the reporter module to create a report or notice and are discussed in the *Voyager Reporter User's Guide*.

---

### Purpose of this Chapter

---

This chapter discusses the media scheduling batch job, Mediajob 5: Retain Patron IDs.

---

### Retain Patron IDs (Mediajob 5)

---

This media scheduling batch job allows libraries to define the number of patrons (patron IDs) whose identification information is retained (or not retained) for media booking history providing retention is configured in the System Administration module.

Configuring retention means that the **Retain Patron Id for Media Booking History** check box on the **Systems - Miscellaneous** work space is checked.

There are three scenarios determining the number of patrons whose information is retained:

- If the **Retain Patron Id for Media Booking History** check box is checked and the job is not run, all the associated patron information is retained.
- If the **Retain Patron Id for Media Booking History** check box is checked and the job is run, the specified number of patron's information is retained.
- If the **Retain Patron Id for Media Booking History** check box is not checked and the job is run, no patron information is retained.

It may be easier to think of this job as when media bookings are *not* retained. Since media bookings can contain multiple resources, each element of the booking is evaluated for retention. The link between the patron's information and the booking is broken only if every resource in the booking has been charged/scheduled more times than the number of IDs (parameter *-n*) provided in the job.

For example, a booking might include a videotape, a VCR, and a scheduled room to view the tape. Also in this example assume that the video tape has been charged out ten times, the VCR has been charged out five times, and the room has been scheduled two times.

- If this job is run asking to retain four IDs, the above booking is retained because the room has only been scheduled two times.
- If, however, this job is run asking to retain one ID, this booking is not retained because each element, the tape, VCR, and room have been charged more than one time.

**NOTE:**

To keep a consistent history, administrator's should request the same number of patron IDs to be retained each time the job is run.

This job can be run as necessary. It can also be added to a cron.

The parameter available when running this job is described in [Table 9-1](#). An (R) designates a required parameter.

**Table 9-1. Parameter for Mediajob 5**

Parameter	Name	Description
-n (R)	Number of IDs to retain	<p>Number of patrons whose identification information is retained for a media booking history.</p> <p>If zero, no patrons are retained.</p> <p><b>NOTE:</b> If you want to retain all patron IDs, simply check the <b>Retain Patron Id for Media Booking History</b> check box on the <b>System - Miscellaneous</b> work space in the System Administration module. There is no need to run this job.</p>

**NOTE:**

When a booking is archived, an entry is made in the MEDIA\_BOOKING\_ARCHIVE table, and if retention is configured, patron identification information is also added to this table.

When Mediajob 5 runs:

- for each item in the MEDIA\_BOOKING\_ARCHIVE table, those bookings where each element of the booking have been charged more times than the number provided with the -n parameter no longer retain the link to the patron information. For all other bookings the link to the patron information remains.

There is no output file for this job. However, an entry in the `mediajob.log` file is made indicating that the job was run and indicates the number of records updated ([Figure 9-1](#)).

---

```
Tue Nov 16 12:31:13 2004 Media RetainPatron...
Tue Nov 16 12:31:13 2004 1 Records Updated
Tue Nov 16 12:31:13 2004 ...COMPLETED
```

---

**Figure 9-1. Mediajob.log File after Running Mediajob 5**

Patron IDs that are retained for media booking history can be viewed in the Media Scheduling module, if the operator has the authority to view Media Booking history. See the *Voyager Media Scheduling System Administration User's Guide* and the *Voyager Media Scheduling User's Guide* for additional information.

This job can be run using a command line, see [Procedure 9-1, Running Mediajob 5 from the Command Line](#), interactively on the server, see [Procedure 9-2, Running Mediajob 5 Interactively](#), or in Webadmin see [Procedure 30-15, Running Retain Patron IDs Media Scheduling, Mediajob 5](#).



### Procedure 9-1. Running Mediajob 5 from the Command Line

---

Use the following to run Mediajob 5 using a command line.

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory
2. At the sbin> prompt, enter **Pmediajob -j5 -n#**

where the # is the number of patron IDs you wish to retain (see [Figure 9-2](#)) and press enter. In this example three IDs should be retained.

---

```
/m1/voyager/xxxdb/sbin $ Pmediajob -j5 -n3
```

```
Initializing...
```

```
Media RetainPatron...
number of retain patron: 3
1 Records Updated
...COMPLETED
```

---

**Figure 9-2. Running Media Scheduling Batch Job 5 Using the Command Line**

**Result:** The job runs retaining the specified number of patron's information. The system returns to the sbin prompt.

The information displayed on the screen when the job runs is also entered into the mediajob.log file ([Figure 9-3](#)).

```
Tue Nov 16 12:31:13 2004 Media RetainPatron...
Tue Nov 16 12:31:13 2004 1 Records Updated
Tue Nov 16 12:31:13 2004 ...COMPLETED
```

---

**Figure 9-3. Mediajob.log File after Running Mediajob 5**

---



---

**Procedure 9-2. Running Mediajob 5 Interactively**

---

Use the following to run Mediajob 5 interactively.

1. Log on to your Voyager server and navigate to the /m1/voyager/xxxdb/sbin directory
2. At the sbin> prompt enter **Pmediajob**

Result: The screen displays the possible media batch jobs to run (see [Figure 9-4](#)).

---

```
/m1/voyager/xxxdb/sbin $ Pmediajob
Initializing...
```

```
Media Scheduling Jobs
```

```
1 -- Overdue Notices
2 -- Inventory Reports
3 -- Booking Statistics
4 -- Charge Statistics
5 -- Retain Patron IDs
99 -- Quit
```

```
Process Job #?
```

---

**Figure 9-4. Available Media Batch Jobs**

---

3. At the Process Job # prompt enter 5 and press enter (see [Figure 9-5](#)).
- 

```
/m1/voyager/xxx/db/sbin $ Pmediajob
```

```
Initializing...
```

```
Media Scheduling Jobs
```

```
1 -- Overdue Notices  
2 -- Inventory Reports  
3 -- Booking Statistics  
4 -- Charge Statistics  
5 -- Retain Patron IDs  
99 -- Quit
```

```
Process Job #? 5
```

---

**Figure 9-5. Providing the Media Scheduling Batch Job Number**

4. At the Please enter the number of patron IDs to retain: prompt enter the number of IDs you want to retain and press enter. In this example three IDs are retained (see [Figure 9-6](#)).
- 

```
Process Job #? 5  
Please enter the number of patron IDs to retain : 3
```

---

**Figure 9-6. Example of Retaining Three Patron IDs**

Result: The job runs, retaining the specified number of patron's information ([Figure 9-7](#)). The system returns to the sbin prompt.

```
Media RetainPatron...
number of retain patron: 3
1 Records Updated
...COMPLETED
```

---

**Figure 9-7. On Screen Display when Running Media Scheduling Batch Job 5**

The information displayed on the screen when the job runs is also entered into the mediajob.log file ([Figure 9-8](#)).

---

```
Tue Nov 16 12:31:13 2004 Media RetainPatron...
Tue Nov 16 12:31:13 2004 1 Records Updated
Tue Nov 16 12:31:13 2004 ...COMPLETED
```

---

**Figure 9-8. Mediajob.log file after Running Mediajob 5**

---



<b>Introduction</b>	<b>10-1</b>
<b>Purpose of this Chapter</b>	<b>10-1</b>
<b>Overview of Bulk Export</b>	<b>10-1</b>
<b>Parameters</b>	<b>10-3</b>
• Running the Marcexport Program	10-9
<b>Output File Specification</b>	<b>10-10</b>
<b>Additional Files</b>	<b>10-10</b>
• Audit File	10-10



---

## **Introduction**

---

The bulk export of MARC records program (marcexport) allows the export of many MARC records at one time. You can specify the records you want to export using a variety of different criteria, for example, record create dates or ISBN.

The program produces an output (export) file of records and an audit file. It does not produce an error file, see the audit file for error messages.

---

## **Purpose of this Chapter**

---

This chapter provides

- Overview of the bulk export of MARC records program
- Parameters that govern the running of the marcexport program
- Output file specification
- Additional files created

---

## **Overview of Bulk Export**

---

The Bulk export of MARC records program allows you to export many MARC records at one time.

Generally exports are one type of record at a time, for example, bibliographic, authority, or holdings records, but you can also export bibliographic holdings (bib-MFHD) groups.

Since all MARC records in the Voyager database are in the UTF-8 character set, records exported are output in the Unicode character set unless otherwise stated. See the -a parameter in the parameters section of this chapter.

Also, sites may run the UseMARCON API which converts records from one MARC format to another. See [UseMARCON Configuration for Use with Voyager](#) on [page B-1](#).

You can specify the records you want to export using a variety of different criteria, for example, record create dates or ISBN; see the parameters section of this chapter.

Some parameters require the creation of an input file; see the parameters section of this chapter for more information.

The program exports records into an output (export) file, and creates an audit file. The default location for these files is the /m1/voyager/xxxdb/rpt directory, unless otherwise specified by the operator.

The marcexport program runs from the /m1/voyager/xxxdb/sbin directory on your server. Running Pmarcexport executes the script which has been configured with your database name, username and password. Therefore, you will not need to enter that information. If Pmarcexport is entered without any parameters, the program will prompt the user for the required information.

If you want to run a server activity using a telnet session, and you do not want to perform it interactively, you can use the following command line to make certain that the activity continues, even if the telnet session is lost,

*nohup <activity with parms> &*

where <activity with parms> is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

**NOTE:**

This server job can be run using WebAdmin, see [Bulk Export MARC Records](#) on [page 30-16](#).

## Parameters

---

The following parameters govern the marcexport program.

**-o USMARC output filename -- not required.**

The name of the file into which the exported records will be placed.

The default is `marc.exp.yyyymmdd.hhmm`, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. It is placed in the `/m1/voyager/xxxdb/rpt` directory.

**NOTE:**

If the output file exceeds  $2^{30}$  (1073741824) bytes, the system appends a 0, 1, 2, etc. in sequence to differentiate the output files (i.e. `marc.exp.yyyymmdd.hhmm.0`).

**-r Record type -- required.**

Specify a record type to be exported see [Table 10-1](#)

**Table 10-1. Record Type and Corresponding Parameter Value**

Code	Record Type	Code	Record Type
A	Authority records	H	Holdings
B	Bibliographic records	M	Main or Added Entry Authority records
E	Series Authority records	S	Subject Authority records
G	Bib-MFHD record groups		

**NOTE:**

For Bib-MFHD record groups:

- The code for bib-MFHD groups (G) results in writing to the export file the specified bib records followed by their associated MFHDs.
- The bibliographic record ID (001 field of the bibliographic record) is exported to the 004 field of the holdings record.
- Also, The following parameters are not supported when you select bib-MFHD groups:  
-I (Library), -O (OCLC only), and -X (exclude file).

**-n Create control number (001) from LCCN (010a) -- not required.**

Must use with the System control identifier (option -c).

This parameter takes the number located in the 010a subfield and creates an 001 field if one does not exist. The 010 LCCN is left intact. If there is no 010 field, the record is logged, an error message is output, and the record is not exported.

**-c <code>System control identifier -- not required.**

Must use with the Create control number (option -n).

This parameter creates an 003 field using the *code* specified on the command-line. Also, it creates an 035 field using the *code* and the bib id number.

**-s      Modifying Agency -- not required.**

Sets the 040d \$d (Modifying Agency) of each exported bibliographic and authority record. Type the code to be used immediately after the -s flag. The code may be a maximum of 15 characters.

**-m      Export mode -- required.**

Indicates the method used to determine which records to export. Enter only one of the following one-letter codes immediately after the flag -m.

<b>M</b>	<b>MARC ID input file</b>
<b>I</b>	<b>ISBN input file</b>
<b>R</b>	<b>MARC ID number range</b>
<b>C</b>	<b>Create dates only</b>
<b>U</b>	<b>Update dates only</b>
<b>B</b>	<b>Both create and update dates</b>
<b>K</b>	<b>OK to export file</b>
<b>S</b>	<b>Suppressed</b>

**-t      Export target -- required.**

Indicates which records will be exported. Enter the appropriate information immediately after the flag -t.

The syntax to enter date information on the command line after the -t parameter is:

**-t~~yyyy-mm-dd:yyyy-mm-dd~~**

If the export mode is:

**M        MARC ID input filename**

This exports any MARC records having MARC IDs (Voyager record IDs) given in the input file.

Specify the name of the input file that contains a list of MARC ID numbers to be exported.

To create the input file containing the MARC IDs use vi or another server side text editor. Each MARC ID in the file must be on a separate line. You can name the file whatever you want. Also, it can be located wherever you want. If the file is not in the same directory as Pmarcexport, that is the /sbin directory, include the full path to the file.

**I        ISBN input filename**

This exports any MARC records having ISBN numbers given in the input file.

Specify the name of the input file that contains a list of ISBN numbers of bibliographic records to be exported.

To create the input file containing the ISBN numbers use vi or another server side text editor. Each ISBN number in the file must be on a separate line. You can name the file whatever you want. Also, it can be located wherever you want. If the file is not in the same directory as Pmarcexport, that is the /sbin directory, include the full path to the file.

**R        MARC ID number range**

This exports any MARC records having MARC IDs (Voyager record IDs) within the range in the input file.

Specify a range of MARC record ID numbers to be exported. Must be either: ALL, for all records, or in the format NNNNN-NNNNN (where the N is replaced by the appropriate number). This includes records matching the upper and lower extent of the range.

**C        Create date range**

This exports any MARC records of the specified type that have a creation date within the specified range.

Specify a date range. The range must be in the format yyyy-mm-dd hh mi ss:yyyy-mm-dd hh mi ss (hh, mi, and ss, corresponding to hour, minute and second, are optional), or today -n (where n=number of days).

**U        Update date range**

This exports any MARC records of the specified type that have a update date within the specified range.

Specify a date range. The range must be in the format  
yyyy-mm-dd hh mi ss:yyyy-mm-dd hh mi ss (hh, mi, and ss, corresponding  
to hour, minute and second, are optional), or  
today -n (where n=number of days).

**B Both create and update date range**

This exports any MARC records of the specified type that have either a  
creation or update date within the specified range.

Specify a date range. The range must be in the format  
yyyy-mm-dd hh mi ss:yyyy-mm-dd hh mi ss (hh, mi, and ss, corresponding  
to hour, minute and second, are optional) or  
today -n (where n=number of days).

**K OK to export file**

This variable locates the records that are marked OK to export on a  
specified date or within a date range. The date is either set when the  
records are imported through Bulk Import, or when the **OK to export** check  
box is set manually in the Cataloging module.

Specify a date range. The range must be in the format  
yyyy-mm-dd hh mi ss:yyyy-mm-dd hh mi ss (hh, mi, and ss, corresponding  
to hour, minute and second, are optional) or  
today -n (where n=number of days).

All the records that are deemed OK to export in Cataloging module and  
that contain the date(s) you enter here are the records that are exported.

**S Suppressed records.**

This exports any MARC records that have been suppressed between the  
specified dates will be exported.

Specify a date range. The range must be in the format  
yyyy-mm-dd hh mi ss:yyyy-mm-dd hh mi ss (hh, mi, and ss, corresponding  
to hour, minute and second, are optional) or  
today -n (where n=number of days).

**-X Exclude -- not required.**

Used only with export modes C, U, or B. This option is not supported with bib-  
MFHD groups (-rG).

Excludes records from being exported based on certain cataloging locations. The  
cataloging location is location at which the record was created or most recently  
modified, rather than the MFHD location, from being exported. The system looks  
for the most recent based on update date or create date.

The -X parameter does not look at any cataloging location associated to any action besides 'CREATE' or 'UPDATE'. Therefore, it will not exclude based on a cataloging location associated with a 'REPLACE' action.

These cataloging locations must appear in the list contained in the user-created exclusion file.

The exclude file should include valid location codes as found in the System Administration module. Also, these location codes are case sensitive, that is if the location code is in upper case in the System Administration module, make sure that it is in upper case in the exclude file.

If the case does not match you may get errors such as Could not find location id for location XXXX, or Failure to retrieve id list of records to export.

For export mode U only the entry with the latest date will be considered.

To create the exclusion containing the cataloging locations that are not to be exported file use vi or another server side text editor. Each location code must be listed, on a separate line in the file. The location codes are found in the System Administration module. Also, location codes are case-sensitive (if the location is listed in System Administration as Main, you must enter Main, not MAIN or main).

You can name the file whatever you want. Also, it can be located wherever you want. If the file is not in the same directory as Pmarcexport, that is the /sbin directory, include the full path to the file.

An example might be **-x exclude.dat**, where exclude.dat is the name of the exclusion file that contains a list of cataloging happening location codes where records were created or updated that should not be exported.

**-i        Ignore Suppressed Bibliographic Records -- not required.**

Indicates that suppressed records should be prevented from being exported. This option overrides the export of suppressed records (-mS).

**-l        Library -- not required.**

Used only for bibliographic record type. This option is not supported with bib-MFHD groups (-rG).

Indicates that only records with the NUC code, as typed immediately after the flag -l, should be exported from the specified export mode (-m).

**-O Only OCLC records -- not required.**

Used only for bibliographic record type. This option is not supported with bib-MFHD groups (-rG).

Specifies that only OCLC records, as identified by the presence of ocm, **oc17**, **OCoLC**, or, (OCoLC) ocm in an 035 \$a or \$9, should be exported from the specified export mode (-m).

**-w MFHD 852\$a updating -- not required.**

When MFHDs are exported (including bib-MFHD groups), the string following the -w switch will replace the contents of any 852\$a in all exported MFHDs.

**-a Assign Character mapping -- not required.**

When this switch is used, -a<CODE>, the exported MARC records are converted from the default UTF-8 character encoding to the specified non-Unicode character set indicated by the CODE.

[Table 10-2](#) provides the supported character sets and their corresponding codes.

**Table 10-2. Character Mapping Codes**

Code	Record Type	Code	Record Type
V	Voyager Legacy	L	Latin-1
R	RLIN Legacy Encoding (this is RLIN's old character set mapping)	M	MARC21 MARC-8
O	OCLC		

For example, if you want to convert records from UTF-8 to the OCLC character set, you would use -aO on the command line. Not specifying any character mapping information (not using the -a switch) causes the program to export the records as they currently exist in UTF-8.

**⚠️ IMPORTANT:**

*When a UTF-8 character is not defined in the character set specified, the character is decomposed (assuming it is precomposed) and then a conversion is attempted, if there is no mapping for the decomposed characters the system substitutes an XML entity reference. The XML entity reference is in the format:*

*&#x 1234*

*where 1234 is the hexadecimal representation of the Unicode (UTF-8) character.*

*The exception to this is when the Latin-1 character set is specified. In this case the middle dot character displays (see Figure 10-1).*

09932cam 22002775a  
45000010006000000500170000600800410002301000170006403500260008103500220010703500100012903500100013  
904000170014904201001660500011001760660007001871000044001942450089002382600320032730000200035950  
00017003795201050039688000370050188007300538880004300611066599020030916143105.0010820s2000 sj  
||||||||||||||ara □ Da 2001303716 □ Da(CstRLIN) DCLN01-B4351 □ Da(DLC) 2001303716 □ Da559320  
Da585470 □ DaDLC-R□cDLC-RD □ DaIcode□000aMLCSN+□ □c(301 □ 06880-01DaR-ihq', cqMu·ammad `Abd  
All ah.□01006880-02Daab\_u D-a`ud, kayfa al-ay\_ah ghayr l-imak? /□cMu·ammad `Abd All ah al-R-i.□  
Da[Khartoum? :Ob:s.n.],□c2000. □ Da45 p. ;□c21 cm.□ DaUnicode test□ DaAbab\_u D-a`ud, `Abd al-  
'Az'iz, d. 1984; singers; Sudan; festscriften; history and criticism; memoirs.□1 □6100-  
01/(3/r□a..... □1006245-02/(3/r□a..... □c.....  
.....□100da..... □001126cam 2200301 a  
45000010006000000500170000600800410002301000180006403500270008203500140010903500110012303500100013  
40350010001440400018001540660007001721000052001792450092002312600640032330000210038750000540040850  
40054004626500037005166500030005536500032005838800050006158800068500065880007400750066600□2003090515  
1433.00930325s1989 ua b 000 0 ara d□ Da 89961881 □ Da(CstRLIN) NJPG93-B17040□  
09ADW1532TS0 □ Da796871D □ Da54915D □ DaNICcdNCICdNJP0 □c(301 Da `Abd al-Maj`id, Mu·ammad  
`Abd al-Nab`i.□06880-01□4Daal-`Adl f-i al-as`al-ib al-`Arab`iyah /□cMu·ammad `Abd al-Nab`i `Abd al-  
Maj`id.□06880-020 □ DaShubr`a, al-Q`ahirah :ObMa`ba`at al-Am anah,□c1989.06880-03D □a125 p. ;□c24  
cm.□ DaUnicode test - garbled in 2001.2 (bad 880 fields) □ DaIncludes bibliographical references  
(p. 113-123).□ DaArabic language□xWord formation.□ DaArabic language□xGrammar.□ DaArabic  
language□xSemantics.□ Da-(3eMeo YHO GdfHj YHO GdeLj0.- (B06100-01/(3/r□12Da-(3GdYod aj GdGSGdjH  
GdYQHfI /-(B0c-(3eMeo YHO GdfHj YHO GdeLj0.- (B06245-02/(3/r□ Da-(3THQG, GdbGgQI :-(B0b-(3eWHYI  
GdGeGfI,-(B0c-(39891.- (B06260-03/(3/r□

**Figure 10-1.** Example of Records where the Middle Dot Character Displays

**-v** Version information -- not required.

Provides version information about the current Pmarcexport program

**-h** Help -- not required.

Provides online help about the `Pmarcexport` program. This flag cannot be used with any other parameters.

**-q** Quiet -- not required.

Prevents Voyager from prompting for any missing parameters. No hash marks display to indicate the progress of the transfer.

## Running the Marcelexport Program

Users can enter the `Pmarcexport` command without any parameters. This prompts the program to ask for the parameters interactively from the user.

At the /m1/voyager/xxxdb/sbin> enter **Pmarcexport**, the system prompts for input of any required parameters.

If you do not want to enter the command interactively, an example of the **Pmarcexport** command with parameters might be entered (on one line) as follows:

**Pmarcexport -rB -mU -t1998-10-25:1998-10-27**

This command would export all updated bibliographic files within the specified date range of 10/25/98-10/27/98 to  
/m1/voyager/xxxdb/rpt/marc.exp.date.time.

## **Output File Specification**

---

The output file contains a file of MARC format records. The default filename is **marc.exp.yymmdd.hhmm**, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. It is placed in the **/m1/voyager/xxxdb/rpt** directory.

**NOTE:**

If the output file exceeds  $2^{30}$  (1073741824) bytes, the system appends a 0, 1, 2, etc. in sequence to differentiate the output files (i.e.  
**marc.exp.yyyymmdd.hhmm.0**).

## **Additional Files**

---

The marcexport program creates an audit file. It does not create an error file.

### **Audit File**

---

The audit file contains information about the marcexport, including error information.

It is named **log.exp.yyyymmdd.hhmm**, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. It is placed in the **/m1/voyager/xxxdb/rpt** directory.

For this batch job, the user cannot specify a filename for the audit file.

If the program cannot convert from UTF-8 to the specified character set the audit file contains a message such as:

Bib/mfhd/auth record: XXXX failed UTF8 translation

where XXXX is the record id.

---

<b>Introduction</b>	<b>11-1</b>
<b>Purpose of this Chapter</b>	<b>11-2</b>
<b>Overview of Prebulk Processing</b>	<b>11-2</b>
• Using Prebulk to Check the Records	11-3
• Using Prebulk to Strip Fields	11-3
• Using Prebulk to Create an Interleaved File	11-4
Bulk Importing of the Interleaved File Generated by Prebulk	11-4
Holdings Created from the Interleaved File	11-5
<b>Input File</b>	<b>11-5</b>
<b>Creating the Configuration File</b>	<b>11-5</b>
• Default Location of the Configuration File	11-6
• Name of the Configuration File	11-7
• Stanza Types	11-7
• Using the Default Call Number	11-7
• Using a Call Number Stored in the Record	11-7
• Overrides Stanza	11-8
CREATEMFHD	11-8
DEFAULTCALLNO	11-9
DEFAULTCALLIND	11-9
USE001FOR014	11-9
USE003FORLOC	11-9
• Strip Stanza	11-9

Non-Standard 035 Field Elimination	11-10
• MFHDTAG Stanza	11-11
• CALLTYPES Stanza	11-12
• LOCATIONS Stanza	11-13
• MAPPING Stanza	11-15
• 008 Stanza	11-17
<b>Parameters</b>	<b>11-20</b>
• Running the Prebulk Program	11-21
<b>Output File Specification</b>	<b>11-22</b>
<b>Additional Files</b>	<b>11-22</b>
• Log (Audit) File	11-22
• Error File	11-22

## Introduction

---

The Prebulk program

- checks your bibliographic records for errors before import. Prebulk can create a file of good bibliographic records that can be imported without errors. Bibliographic records containing errors are written to an error file and can be dealt with separately without disturbing the main Bulk Import process
- strips out one or many tags or subfields from bibliographic records. The resulting file can be either a file of only stripped bibliographic records or an interleaved bib/MFHD file (a file of stripped bibliographic records and MFHDs)
- creates an interleaved file of bibliographic records and MFHDs (holdings records) to be loaded into a Voyager database using Bulk Import. (Universal Catalog databases require a interleaved bib/MFHD file when loading records into the UC database.)

To do this sites must create an input file of records and a configuration file.

After running this program an output file of records is produced. It also creates a log (audit) file and an error file. The prebulk output file is UTF-8 encoded.

## Purpose of this Chapter

---

This chapter provides

- Overview of Prebulk processing
- Input file
- Configuration file
- Parameters and an example
- Output file specification
- Additional Files

## Overview of Prebulk Processing

---

The Prebulk program is designed to pre-process bibliographic records and create an output file that may be imported into a Voyager database.

The manner in which the file is processed can be customized. This is done by creating a configuration file that will govern the Prebulk process.

Therefore, sites must create a configuration file directing the processing, and an input file of records to be processed.

Prebulk can be used for three main purposes: to check records, strip tags from records and/or create an interleaved file of bibliographic and MFHD records. In the bib-MFHD file the newly-created holdings records are created based on the configuration file.

The Prebulk program looks to the `/m1/voyager/xxxdb/local` directory for the input file.

It creates an output file of processed records that are UTF-8 encoded. The default name is `out.prebulk.yyyymmdd.hhmm`, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. The file is placed in the `/m1/voyager/xxxdb/rpt` directory.

The prebulk program also creates error and log files. The default filenames are, `err.prebulk.yyyymmdd.hhmm`, and `log.prebulk.yyyymmdd.hhmm`, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. These files are placed in the `/m1/voyager/xxxdb.rpt` directory.

If you specify an output, audit or error filename only, they are placed in the `/m1/voyager/xxxdb/sbin` directory.

If you want any of the files to be read from or created in a different directory than the default, enter the full path and the filename.

The Prebulk program should be run from the `/m1/voyager/xxxdb/sbin` directory on your server.

**NOTE:**

If Prebulk is run more than once in one minute, it will write over the existing error and log files. If the default output filename is being used, it will also be written over. This means that if Prebulk is run more than once a minute, only the last set of files generated in that minute will be preserved. If you want to save all of the files created when running Prebulk more than once a minute, you must rename each set of files before running Prebulk again.

Once the files have been generated by Prebulk, they can be imported into the database using the Bulk Import program. When importing Prebulk-generated files into the database, the results of the Bulk Import process will depend on the type of file being imported: an interleaved file or not.

See [Bulk Import, Replace, and Merge of MARC Records](#) on page 12-1, and the *Voyager System Administration User's Guide*, [Bulk Import Rules](#), for more information on the Bulk Import Rules.

## Using Prebulk to Check the Records

---

Prebulk can be used to check records in your bibliographic import files and produce a new file with only the good bibliographic records.

To remove erroneous records, set **CREATEMFHD=NO** in the [OVERRIDES] stanza of the configuration file to create a bibliographic file. This file can then be imported with the bulk import feature to create one holdings record and one item per bibliographic record. For more information on the configuration file, see [Creating the Configuration File](#) on page 11-5.

**NOTE:**

This requires that you do not strip the fields containing barcodes, location, or other information necessary to the creation of holdings records and items. For more information on stripping fields, see [Using Prebulk to Strip Fields](#) on page 11-3.

## Using Prebulk to Strip Fields

---

Prebulk can strip fields out of the bibliographic records before you import them.

Stripping fields is done by listing the field to be stripped after the [Strip] stanza in the configuration file, see [Strip Stanza](#) on page 11-9 for more information.

If you use Prebulk only to strip fields from your records you can import the files into the database per normal Bulk Import procedures. Importing through Bulk Import, you can create one holdings record and one item per bibliographic record.

However, this requires that you not strip the fields containing barcodes, location, or other information necessary to the creation of holdings records and items. See the *Voyager System Administration User's Guide* for more information.

### **Using Prebulk to Create an Interleaved File**

---

The Prebulk program can process bibliographic records with holdings information stored in a particular field in the bibliographic record and create an interleaved file of bibliographic and MFHD records. Also, some tags and/or subfields can be stripped.

Prebulk can create multiple (duplicate) MFHDs when creating an interleaved bib/MFHD file. However, doing this type of processing also prevents the creation of any items in the database, which Bulk Import allows you to do without performing this type of processing.

When processing records, Prebulk looks in the bibliographic record for the field listed in the [MFHDTAG] stanza in the configuration file and, for every instance of that tag contained in the bibliographic record, Prebulk generates a new holdings record. The newly-created holdings records are created based on a configuration file that you can create.

### **Bulk Importing of the Interleaved File Generated by Prebulk**

When the records are being imported, the holdings records from the Prebulk file will load. When importing this type of file (and only when importing this type of file), Bulk Import will not automatically create any other holdings records or any item records. The bibliographic records are loaded according to the settings in the Bulk Import profile in the System Administration module. See [Bulk Import Rules](#) in the *Voyager System Administration User's Guide* for more information.

If the location of the MFHD(s) in the interleaved bib/MFHD file matches the location of the existing MFHD(s) in the database, Bulk Import will delete all existing MFHD(s) in the database with that matching location code. After the existing MFHD(s) are deleted, the system adds a new MFHD for each location listed in the interleaved bib/MFHD file.

If the location of the MFHD(s) in the interleaved bib/MFHD file matches the location of a MFHD in the database that has an item attached, then the MFHD in the database will *not* be deleted. The incoming MFHD will then be added to the database at that location.

**NOTE:**

This is the only situation where two MFHDs will be allowed at the same location.

If the interleaved bib/MFHD file contains more than one MFHD with the same location code, only one MFHD will be created in the database upon bulk import. This means that if you want to create more than one MFHD for each bib, each MFHD must have a unique location code. This is in order to prevent Universal Catalog databases from having more than one MFHD for each location.

### **Holdings Created from the Interleaved File**

When Prebulk creates the new holdings information for a bibliographic record, a standard Voyager MFHD is created. The holding records are created by reading the records stored in the input file and processing each record based on the settings in the Prebulk configuration file. Records processed are written to the output file, `out.prebulk.yyyymdd.hhmm`.

The leader is the standard MFHD default.

The **Record Type** in the MFHD leader will be set to serial item holdings (y) if the bibliographic record has s in offset 7 of its leader. Otherwise the **Record Type** of the MFHD leader will be set to single-part item holdings (x).

### **Input File**

---

Your site creates an input file of bibliographic records to be processed using the Prebulk program. The program by default looks for this file in the `/m1/voyager/xxxdb/local` directory, unless otherwise specified.

### **Creating the Configuration File**

---

You must create a configuration file that governs the Prebulk program execution. The Prebulk program reads from this file to determine how records are to be processed, that it what is to be done with each record.

The configuration file must be created using the server side text editor, `vi`. To separate pieces of data in the configuration file, Prebulk requires the tab stops that only `vi` can produce.

**IMPORTANT:**

*Configuration files created in any other editing programs, such as Notepad, and then FTP-ed to the server will cause Prebulk to fail. Use only vi to create your configuration files.*

The configuration file must be tab-delimited. A tab-delimited file means that for any entries in the file where multiple pieces of information must be specified on the same line, you must separate each piece of information with a tab. The Prebulk program looks for the tab (the tab character) as a separator, it will not recognize spaces as separators between pieces of information.

For example, your [ LOCATIONS ] stanza might look like [Figure 11-1](#)

---

[LOCATIONS]
MAIN -> MAORI-U ->099,090,050
REFERENCE ->MAORI-R ->949,099(1),090,050

---

**Figure 11-1. Sample LOCATIONS Stanza**

There are three pieces of information in the first line: MAIN, MAORI-U, and 099,090,050. Each of these is separated by a tab. (In this document, we will use -> to indicate a tab.) Your file should use the normal tab character (the tab key). The tab character is counted (treated as a regular character) when determining the length of strings.

For example, the last line in the previous example would appear in the file as:

***REFERENCE[tab]MAORI-R[tab]949,099(1),090,050***

**NOTE:**

When the # character appears in the first position on a line, that tells the Prebulk program that the line should be ignored. For example, when you want to enter a comment onto a line, preceding the comment with a # will cause Prebulk to not try and process that line as valid information.

---

**Default Location of the Configuration File**

---

The default location for the configuration file is the /m1/voyager/xxxdb/local directory.

## Name of the Configuration File

---

The name of the configuration file can be whatever you want. Prebulk has no default name for the configuration file. `Prebulk.cfg` is a suggested name, but not required or necessary.

## Stanza Types

---

There are several stanzas that can be included in the configuration file.

- Overrides
- Strip
- MFHDtag
- Calltypes
- Locations
- Mapping
- 008

## Using the Default Call Number

---

To force the Prebulk program to use the default call number for a particular record you must provide the `[LOCATIONS]` stanza with a reference to a field that contains no default call number.

To do this, you must specify the call number location in the third field of the `[LOCATIONS]` stanza, see [LOCATIONS Stanza on page 11-13](#). The location must be a field and override indicator combination, and must be a call number location that does not exist, that is, one that does not contain a call number. The subfield indicator is required in order to perform this action properly.

## Using a Call Number Stored in the Record

---

Use the following to force Prebulk to use a call number specified in the record. You must set this in the configuration file:

In the `[MFHDTAG]` stanza you must list the field and subfield where the call number is stored. The `[MFHDTAG]` stanza identifies the MARC tag which contains the data that is to be used to create the MFHD. See [MFHDTAG Stanza on page 11-11](#) for more information.

You must have the field and indicator listed in the third column of the [LOCATIONS] stanza. The location must be a field and override indicator combination. For example, if the call number was stored in the 949 field, the field and indicator would read 949(1). See [LOCATIONS Stanza](#) on [page 11-13](#) for more information.

The [CALLTYPES] stanza must be empty. This is because Prebulk will automatically look in the subfield 'a' of the fields listed in the [CALLTYPES] stanza. See [CALLTYPES Stanza](#) on [page 11-12](#) for more information.

In the [MAPPING] stanza, the field and subfield of the call number must be mapped to the appropriate field and subfield in the new MFHD. See [MAPPING Stanza](#) on [page 11-15](#) for more information.

## Overrides Stanza

---

[Figure 11-2](#) provides an example of the [OVERRIDES] stanza.

---

```
[OVERRIDES]
CREATEMFHD=yes
DEFAULTCALLNO=PZ3.L129
DEFAULTCALLIND=0
USE001FOR014=no
# uses 003 for location only, in addition
# another tag may be used for other mapping
# (or not) [MAPPING] stanza may be blank if
# 003 is used
USE003FORLOC=no
```

---

**Figure 11-2. OVERRIDES Stanza Example**

## CREATEMFHD

This variable indicates whether or not MFHDs (Holdings records) are to be created. Entering YES indicates that holdings records will be created. If YES is entered, the resulting file will be in interleaved bib/MFHD format. Entering NO indicates no holdings records are to be created and that the program is being run only to strip tags or check records. If NO is entered, a standard bibliographic file will be created. YES and NO are not case sensitive.

## DEFAULTCALLNO

If no call number is found in the [ MAPPING ] stanza (see [MAPPING Stanza](#) on [page 11-15](#) for more information), this call number is placed in the 852 \$h of the new MFHD. If no call number is found and this field is blank, no call number is created. The value can be blank or any string. The Default Call Number must be blank if the tag specified in the [ MFHDTAG ] stanza is listed for any of the locations in the [ Locations ] stanza.

## DEFAULTCALLIND

The value for indicator one of the 852 if the **DEFAULTCALLNO** is used. This should be a number from 0 through 8. Pipes and blanks are not acceptable.

## USE001FOR014

Entering Yes indicates that an 014 will be created with a \$a containing the value of the 001 tag. The indicators for the 014 are 1 and blank. Entering No indicates the 001 will not be used for the 014. YES and NO are not case sensitive.

## USE003FORLOC

Entering Yes indicates that the incoming 003 tag will be the location used in the 852 \$b. This is the default used if no location is specified in the field listed in the [ MFHDTAG ] stanza. Entering No means that it will not. YES and NO are not case sensitive.

## Strip Stanza

---

The [ STRIP ] stanza is a list of tags and/or subfields that you want to remove from the incoming bibliographic record (see [Figure 11-3](#)).

---

[STRIP]
# Consists of a tab-delimited list of tags
# and/or subfields
# to be stripped from the incoming bib record

---

**Figure 11-3.** The STRIP Stanza

```
# No subfields listed means the entire tag is
# stripped. 001,008,245 tags may not be
completely stripped.
650-> 04-> w
700-> X5-> asd
```

**Figure 11-3. The STRIP Stanza (Continued)**

- Listing a tag alone will strip the tag entirely. If tag and indicators are listed, then only tags with those indicators are removed. If tag, indicators, and subfields are listed, then all occurrences of the subfield within the matching tag and indicators are removed.
- X denotes a wild card and any indicator value will match.
- | indicates that the indicator is blank.
- If no data is left in the tag after stripping subfields, then the tag is removed.
- The 245 tag may not be removed entirely but subfields can be stripped.
- The 003 tag is not automatically stripped; it must be included in the [STRIP] stanza in order for it to be stripped.
- If no data is left in the 245 after stripping subfields, a log message exception is generated and the entire 245 is maintained.
- If you put the tag specified in the [MFHDTAG] stanza in the list to be stripped, Prebulk will build any holdings information you specify in the configuration file before stripping the tag.
- 001 and 008 tags may not be stripped.

### **Non-Standard 035 Field Elimination**

With the migration of bibliographic records from some library management systems into Voyager a non-standard 035a subfield may be left in the bibliographic record. The MARC standard dictates that the system control number field be prefixed by the organization of the original system control number. Therefore, a standard 035a would contain an alphabetic code for an organization and its system control number. For example, \$035a Ex Libris 12345. A non-standard 035a contains either only alphabetical data, or only numeric data.

With prebulk processing, administrators can specify that non-standard 035 fields be stripped from the record. This eliminates bulk import de-duplication problems.

Users can choose to strip out those 035 fields where the subfield a contains only letters or only numbers.

System Administrators use the prebulk configuration file's [ STRIP ] stanza to indicate what they want to strip.

**NOTE:**

As with any field you want to strip, indicators or wildcards must be included.

Examples of the entry in the [ STRIP ] stanzas are as follows:

To strip the subfield a of the 035 fields enter this line in the [ STRIP ] stanza:

035 XX a

To strip non-standard 035 fields, enter one of these lines in the [ STRIP ] stanza:

035 XX a\* where XX is the indicator wildcard, or

035 || a\* where | means the indicator is blank.

An \*(asterisk) after the 'a' is the function to strip the non-standard fields.

When stripping non-standard fields the steps of this process are:

1. The system administrator runs the prebulk program.
2. The program reads in the configuration file.
3. If the program identifies that a record has a non-standard 035a, that is only numbers or only letters (punctuation and spaces in the 035a are ignored), it strips that entire field from the record.

---

### MFHDTAG Stanza

---

The [ MFHDTAG ] Stanza identifies the MARC tag which contains the data that is to be used to create the MFHD, such as the holdings location, call number, barcode, operator and item type (see [Figure 11-4](#)).

```
[MFHDTAG]
# nothing below here is required if only using
# program to strip tags
# 'XXX' means no tag, use default location
949
```

**Figure 11-4.** The MFHDTAG Stanza

- Values must be from 010 through 999 or XXX, where XXX means that no tag contains holdings information and the default location should be used in the 852 \$b.
- Only one tag may be used as the MFHD tag.
- Each occurrence of the tag in a bibliographic record will create a separate MFHD record.

### **CALLTYPES Stanza**

---

The [CALLTYPES] stanza specifies what indicator is to be used when creating an 852 in the MFHD. The indicator may be different for each call number type listed. When the appropriate call number is determined by the [LOCATIONS] stanza its field number will be checked against this list and the call number will be placed in the 852 in the new MFHD with the subfield specified in this list (see [Figure 11-5](#)).

```
[CALLTYPES]
# assign these call number types to tags
# unless otherwise noted
099 -> 8
050 -> 0
086 -> 3
```

**Figure 11-5.** The CALLTYPES Stanza

- There must be at least one location listed.
- The first field can be 010-099.
- The second field can be 0-8 or c for CODOC numbers.

- The value in the second field is placed in the first indicator of the 852 field in the MFHD.
- Each row of the stanza have a tab separating the two fields.

**NOTE:**

Any fields listed in the [LOCATIONS] stanza that do not have the indicator specified immediately after must be listed in the [CALLTYPES] stanza. If fields appear without indicators and are not listed in the [CALLTYPES] stanza, the Prebulk program will fail.

**LOCATIONS Stanza**


---

The [LOCATIONS] stanza determines what location is to be used in the MFHD (see [Figure 11-6](#)).

---

[LOCATIONS]
# tab delimited by location. the first line is
# the default setting. no match means no call
# number and a blank indicator will be used
# unless override option.
# T means use 245 \$a in \$l with 7 indicator
#
#Incoming Loc New Loc Tag(Override)Hierarchy
MAIN -> MAORI-U -> 099,090,050
REFERENCE -> MAORI-R -> 949(1),099(1),090,050
PERIODICAL -> MAORI-P -> T
# entries not listed default to the first
# entry with no normalization

---

**Figure 11-6. The LOCATIONS Stanza**

- You must specify a list of locations that may be found in the bibliographic record. For each location, must specify a name to be given (that is, whether you want to rename the locations or not), and create a list of tags that specifies which tags are preferred over others. The name in the second field must be a location code (of not more than ten characters) as assigned in the System Administration module.

**NOTE:**

Entering an invalid location code will not cause problems within the Prebulk program, but it will cause problems when attempting to import the records with the Bulk import program. You must be sure that the location codes you enter are identical to the location codes listed in the System Administration module. See the *Voyager System Administration User's Guide*, [Locations](#) section, for more information.

If a record does not match any of the locations, the first row will be used as a default. Each row of the [LOCATIONS] stanza contains three different fields.

- The first field is an incoming location found in the field referenced in the [MFHDTAG] stanza. The location entered in this field must exactly match the location listed for the new MFHD (the location listed in the field in the [MFHDTAG] stanza). See [MFHDTAG Stanza](#) on [page 11-11](#) for more information.
- The second field is the location code to be placed in the 852 \$b. It may be the same as the incoming location name or a variation.
- The third field is the tag hierarchy to be used for that location to create a call number in the 852 \$h and \$i. For that location, you can specify which tags are to be used above others. For tags from 100 through 999, you must specify an indicator after the tag (for example, 949(2)). For tags from 010 through 099, an indicator is optional, but if it is not included, the tag must be listed in the [CALLTYPES] stanza as well.
- Each row of the stanza must have a tab separating each of the three fields.

For example, if the Archives department (which has a location code of Archives, and to be renamed Archv) puts the call number in the 099 and the rest of your locations (such as the main cataloging area, for which the location code is Cat, and not to be renamed) put the call number in the 050, you would want to create two locations in your list. The Archv location should have the 099 tag listed first so that if there is both an 050 and an 099 in the record, the 099 will be chosen over the 050. To handle all of the other locations, you could create a separate row for each one, but you could also just create one row for one of the locations and place it first so that if a record matches neither record, it will choose the default row (which is the first row).

Your [LOCATIONS] stanza might look like [Figure 11-7](#).

---

[LOCATIONS]
Cat -> Cat -> 050,099,090
Archives -> Archv -> 099,050,090

---

**Figure 11-7. Sample LOCATIONS Stanza**

**NOTE:**

The first indicator of the new 852 will be determined by the values in the [CALLTYPES] stanza. You can change this for a particular tag by entering a value in parentheses following the tag in the hierarchy.

If a T is found in the call number hierarchy, then no subfield h and I will be created. Instead a \$I (el) will be created and the first indicator will be 7. The first field of the call number tag will be used as the \$h and all remaining subfields are subfield i.

The first field can be any string. The second field can be any string up to ten characters. The third field can be a number from 010 through 999 or T. Values in parentheses can be 0-8. No normalization of strings is done other than stripping preceding and trailing blanks.

If incoming data does not match any field in the [LOCATIONS] stanza, the first entry is used as a default and an exception with 001 value is placed in the log file.

This means that if you want a location name to remain what it was originally, it must be listed in the [LOCATIONS] stanza and must have the same location specified in the first and second fields.

## MAPPING Stanza

---

The [MAPPING] stanza indicates the subfields in the tag (specified in the [MFHDTAG] stanza) which contain specific data, and where that data goes in the MFHD to be created. All of the subfields indicated in the list must be in the tag listed in the [MFHDTAG] stanza (see [Figure 11-8](#)).

[MAPPING]
# Incoming subfield and outgoing
# tag/subfield. Only one of each tag
# will be created if using XXX, then no
# mapping only 852 b.
# 852 b is required. Multiple subfields are in
# order listed except 852 k is before \$h and
# 852 m is after \$h and \$i.
# Multiple subfields are concatenated with
# blanks.
l -> 852b
vnk -> 866a
#{(If an 866 a is created, a \$80 is also
# created
p -> 852k
s -> 852m
c -> 852t
h -> 852h
#{(Illegal if no match in [LOCATIONS])
i -> 852i
#{(Also illegal if no match in [LOCATIONS])

**Figure 11-8. The MAPPING Stanza**

- The first field in each row is the subfield of the tag specified in the [MFHDTAG] stanza where a particular piece of information is stored. The second field is the field and subfield where that piece of information is to go in the new MFHD.
- One 852 and one 866 is created. The indicators of the 852 are the call number indicator and blank. All other tags have blank indicators. If an 866 tag is created, a preceding \$8 with the value of 0 is also created. Only one tag of each type will be created.

**NOTE:**

The value mapped to the 852 b will be used as the incoming location for the [LOCATIONS] stanza.

If the tag specified in the [MFHDTAG] stanza is not listed in the [LOCATIONS] stanza, the tags will be in numerical order. Subfields will be in the order found in the [MAPPING] stanza with the exception of the call number sequence in the 852 which will always have the following subfields in the order k, h, i, and then m.

If the tag specified in the [MFHDTAG] stanza is listed in the [LOCATIONS] stanza, the tags will be placed in the new MFHD in exactly the order that they appear in the [MAPPING] stanza. The tags will not be sorted.

If multiple occurrences of an incoming subfield appear in one field of the first row, the information will be concatenated with a single space between incoming subfields.

## 008 Stanza

---

The [008] stanza specifies how to create the 008 in the new MFHD (see [Figure 11-9](#)).

---

[008]
#use the 008 here based on the value of
#offsets 6 and 7 of the leader
# top is default, length must be correct
#(YYMMDD means date of run), spaces count
000-> 6-> am-> YMMDDetc
000-> 6-> as-> YMMDDetc
000-> 6-> gm-> 9812160p 8 4001aueng0000000
#overrides the information in the leader
007-> 0-> p-> YYMMDDetc

---

**Figure 11-9. The 008 Stanza**

- You can specify a location in a field in the bibliographic record and if it matches a string that you specify, you can designate a string that is to become the 008.
- Each row contains four fields. The first field is the fixed length field to be examined (000 indicates the leader). The second field is the offset to examine. The third field is the value to match. The final field is the 008 to be created if the value is matched.

**NOTE:**

The first entry will be the default if no value is matched. If multiple values are matched, the final matching entry in the map overrides all other matches.

Specifying YYMMDD in the first six entries in the fourth field indicate that the program run date is to be used to create this field.

**NOTE:**

The first field may be 000-009. The second field can be any 3 digits. The third field can be any string. The final field (the 008) must be 32 characters in length including alphanumeric characters, tabs, pipes, and spaces.

[Figure 11-10](#) provides a sample configuration file.

```
[OVERRIDES]
CREATEMFHD=YES
DEFAULTCALLNO=PS123.A2 1999
DEFAULTCALLIND=8
USE001FOR014=YES
USE003FORLOC=NO

[STRIP]
015-> ||-> a
020-> ||-> a
043-> ||-> a
082-> ||-> a
260-> ||-> abc
300-> ||-> a
400-> XX-> abt
440-> XX-> abnv
490-> 0|-> a
500-> XX-> a
504-> ||-> a
600-> 1|-> a
651-> |0-> axy
[MFHDTAG]
949
[CALLTYPES]
090-> 0
#050-> 0
099-> 0
[LOCATIONS]
#main--> main-> 050,090(1),099
#serials--> serials-> T
#Reference--> Reference-> 050,090(1),099
main-->main-> 092(1)
[MAPPING]
b-> 852b
i-> 852i
h-> 852h
[008]
#->->->0000000001111111122222222333
#->->->12345678901234567890123456789012
000-> 6-> am-> YYMMDD
```

Figure 11-10. Sample Configuration File

## Parameters

---

The following are the valid parameters for Prebulk.

**-i      Input file -- required.**

The name of the file where the bibliographic records to be processed are to be found. The default location for the input file is the /m1/voyager/xxxdb/local directory. Use the entire path to the file in the command line.

**-o      Output file -- not required.**

The name of the file where the bibliographic and holdings records are to be written after being processed. The output file created is called out.prebulk.YYYYMMDD.HHMM, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. It is placed in the /m1/voyager/xxxdb/rpt directory. If you specify a filename, but no path, the file is placed in the /m1/voyager/xxxdb/sbin directory. If the name is that of an existing file, Prebulk will not run unless you specify -f to overwrite the existing file.

**NOTE:**

The system displays an error message if you specify an output file that has the same name as the input file.

**-c      Configuration file -- required.**

The name of the file where the Prebulk configuration information is stored. There is no default name of the file. The default location for the configuration file is the /m1/voyager/xxxdb/local directory.

**-e      Error file -- not required.**

The name of the file where error information is to be written. Prebulk will write any records that cannot be processed to this file. The error file created is called err.prebulk.YYYYMMDD.HHMM, where y is the year, m is the month, d is the day, h is the hour, and m is the minute. It is placed in the /m1/voyager/xxxdb/rpt directory. If you specify a filename, but no path, the file is placed in the /m1/voyager/xxxdb/sbin directory. If the name is that of an existing file, Prebulk will not run unless you specify -f to overwrite the existing file.

**-p      Parameter file -- not required.**

The name of the file that contains a list of parameters commonly entered when running Prebulk. This allows you to create a file containing the parameters that you enter each time onto a single line. A sample parameter file might be:

```
-i Input.bib  
-o Output.bib  
-c prebulk.cfg
```

Therefore, instead of entering parameters repeatedly, you simply need to enter **-p** and the filename.

**-a Append tag for log file -- not required.**

Adds the log information to the end of the previously generated log file.

**-f Force overwrite of output files switch -- not required.**

Tells Prebulk to overwrite any existing output file of the name specified by the **-o** switch.

**-l Prebulk log file -- not required.**

The log file for processing of the record. Contains the number of bibliographic records processed (not including erroneous records) and the number of holdings records created. The log file created is called `log.prebulk.YYYYMMDD.HHMM`, where *y* is the year, *m* is the month, *d* is the day, *h* is the hour, and *m* is the minute. It is placed in the `/m1/voyager/xxxdb/rpt` directory. If you specify a filename, but no path, the file is placed in the `/m1/voyager/xxxdb/sbin` directory. If the name is that of an existing file, Prebulk will not run unless you specify **-a** to append the new information to the existing file.

## Running the Prebulk Program

---

Before you can run the prebulk program you must first create the input file and configuration file.

Users can enter the **Pprebulk** command without any parameters. This prompts the program to ask for the parameters interactively from the user.

At the `/m1/voyager/xxxdb/sbin>` enter **Pprebulk** and the required parameters after the command. All of the parameters must be on the same line.

For example:

```
Pprebulk -i /m1/voyager/xxxdb/local/input.bib -o  
output.bib -c prebulk.cfg
```

**NOTE:**

Use the full path to the input file with the `-i` parameter.

This command instructs the system to use the input file named `input.bib`, create an output file named `output.bib` and use the configuration file named `prebulk.cfg` to direct the processing of the records.

## **Output File Specification**

---

The output file is the file where the bibliographic and holdings records are to be written after being processed. It is in MARC format and it is UTF-8 encoded.

The default filename is `out.prebulk.YYYYMMDD.HHMM`, where y is the year, m is the month, d is the day, h is the hour and m is the minute, unless a name is specified with the `-o` parameter. The file is placed in the `/m1/voyager/xxxdb/rpt` directory. If you specify a filename, but no path, the file is placed in the `/m1/voyager/xxxdb/sbin` directory.

If the name is that of an existing file, Prebulk will not run unless you specify `-f` to overwrite the existing file.

## **Additional Files**

---

In addition to the output file of records, the Prebulk program creates an error and a log file.

### **Log (Audit) File**

---

The log file contains the number of bibliographic records processed (not including erroneous records) and the number of holdings records created.

The default filename is `log.prebulk.YYYYMMDD.HHMM`, where y is the year, m is the month, d is the day, h is the hour and m is the minute. It is placed in the `/m1/voyager/xxxdb/rpt` directory, unless specified by the `-l` parameter.

If the name is that of an existing file, Prebulk will not run unless you specify `-a` to append the new information to the existing file.

### **Error File**

---

The error file is where Prebulk will write any records that cannot be processed.

The default filename is `err.prebulk.YYYYMMDD.HHMM`, where y is the year, m is the month, d is the day, h is the hour and m is the minute. It is placed in the `/m1/voyager/xxxdb/rpt` directory, unless specified by the `-e` parameter.

If the name is that of an existing file, Prebulk will not run.

If there were no errors, a file is created, however it is empty.



---

## **Bulk Import, Replace, and Merge of MARC Records**

# **12**

---

<b>Introduction</b>	<b>12-1</b>
<b>Purpose of this Chapter</b>	<b>12-2</b>
<b>Overview of the Bulk Import, Replace, Merge MARC Records Program</b>	<b>12-2</b>
• The UTF-8 Character Set Encoding: Character Set Mapping, Record Leader, and Conversion Attempts	12-3
Character Set Mapping	12-4
Leader Byte 9 Value	12-4
Converting Records to Unicode	12-4
• Disabling the Generation of the Keyword Index Files	12-6
<b>Parameters</b>	<b>12-7</b>
• Running Bulk Import	12-10
<b>Input File Specification</b>	<b>12-11</b>
<b>Building the 035</b>	<b>12-11</b>
• OCLC Control Number Expansion	12-12
<b>Creating Holdings and Item Records</b>	<b>12-12</b>
• Call Number Hierarchy	12-12
• Barcode	12-13
• Item Type	12-13
• Mapping	12-13
<b>Additional Files</b>	<b>12-13</b>

• Log file	12-13
• Delete, Discard, Replace, and Reject files	12-14
• Error File	12-15

<b>Messages in Log and Error Files</b>	<b>12-15</b>
--	--------------

---

## Bulk Import, Replace, and Merge of MARC Records

# 12

---

### Introduction

---

The Bulk Import, Replace, and Merge (or Bulk Import) program

- imports, replaces, or merges many authority or bibliographic records (for example, USMARC, RLIN, and MARCIVE records) at one time
- allows for the automatic creation of MARC holdings records and Voyager item records based on an import profile established in the System Administration module
- can also load holdings records (MFHDs), item records, and purchase orders (POs), and
- indexes the records completely.

For example, you might want to load a file of bibliographic records that were supplied by a bibliographic utility.

To do this sites must have a file of records (in MARC format) to import. Sites may run the UseMARCON API which converts records from one MARC format to another. See [UseMARCON Configuration for Use with Voyager](#) on [page B-1](#).

Also sites must create a bulk import rule and a duplicate detection profile in the System Administration module to govern the import process.

After running the program the system creates an audit and an error log. Other files are created also depending on the Add or Replace duplicate handling option selected in the duplicate detection profile in the System Administration module.

## Purpose of this Chapter

---

This chapter discusses

- Overview of the Bulk Import program
- Input file specification
- Parameters that govern the program and running the program
- Building the 035
- Creation of Holdings and Item records
- Additional files

## Overview of the Bulk Import, Replace, Merge MARC Records Program

---

Bulk Import can be used to load the following types of information

- Bibliographic record load only
- MFHDs only
- MFHDs and POs
- MFHDs and Items
- MFHDs, Items and PO

Prior to running the Bulk Import program, the user must set up Bulk Import Rules and Duplicate detection profiles in the System Administration module (see [Bulk Import Rules](#) and [Bibliographic Duplicate Detection](#) in the *Voyager System Administration User's Guide* for more information), and have an input file of records to load.

The program will look in the /m1/voyager/xxxdb/sbin directory for the input file if a complete path is not specified.

This program generates audit and error files. The default location for these files is the /m1/voyager/xxxdb/rpt directory. The default filename for the log file is log.imp.YYYYMMDD.HHMM, where y is year, m is month, d is day, h is hour, and m is minute. The default filename for the error file is err.imp.YYYYMMDD.HHMM, where y is year, m is month, d is day, h is hour, and m is minute.

The additional files created are based on the type of bulk import being done. They are named discard.imp.yyyymmdd.hh.mm, delete.imp.yyyymmdd.hhmm, reject.imp.yyyymmdd.hhmm, and replace.imp.yyyymmdd.hhmm. These files are placed in the /m1/voyager/xxxdb/rpt directory.

If Bulk Import is run more than once in one minute, it will write over the existing delete, discard, log, reject, replace and error files.

The bulk import program runs from the `/m1/voyager/xxxdb/sbin` directory on your server. Running the `Pbulkimport` script sets the environment variables, tells the server the pieces it needs, executes the Bulk Import program, and writes to a log and an error file, as well as creating files of records based on the type of import.

For optimum importing performance, import 10,000 records (or less) at one time. If your record file is larger than 10,000 records, it should be broken into smaller sets of records (using the `-b` and `-e` parameters) and then imported one after the other.

**NOTE:**

When a new record (record status in the bibliographic leader is set to n) in the database is replaced or merged with another record, the record status of the record in the database will automatically be changed to corrected (c in the leader).

If you want to run a server activity using a telnet session, and you do not want to perform it interactively, you can use the following command line to make certain that the activity continues, even if the telnet session is lost,

```
nohup <activity with parms> &
```

where `<activity with parms>` is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

**NOTE:**

This server job can be run using WebAdmin, see [Bulk Import MARC Records](#) on [page 30-12](#).

---

## The UTF-8 Character Set Encoding: Character Set Mapping, Record Leader, and Conversion Attempts

Bibliographic, Authority, and Holdings records in your Voyager database are in the Unicode character set. Therefore, records being imported into the database must be in the Unicode character set, if they are not, the Bulk Import program attempts to convert them. Item records, which are not MARC records, are in the Latin-1 character set.

When Bulk Importing records into the Voyager database, the system uses the character set mapping defined in the Bulk Import Rule, the value of the leader byte 9, and the results of the conversion attempt to determine if a record is eligible for import.

Records error out, and consequently are not imported, because

- there is disagreement between the mapping character set rule and the value of the leader's byte 9
- the conversion to UTF-8 fails, or
- due to other reasons not related to the character set.

**NOTE:**

All MARC records in the Voyager database must be UTF-8 encoded. If the conversion from another encoding to the UTF-8 encoding fails, try using third party software to convert the records, or obtain UTF-8 encoded records for import.

## Character Set Mapping

Since the Bibliographic, Authority, and Holdings records in the Voyager database use UTF-8 encoding, the system attempts to convert records not at that level before importing them.

The Bulk Import rule specifies the expected character set of the incoming records. In the **System Administration> Cataloging> Bulk Import Rules> Rules tab> Expected Character Set Mapping of the Imported Records** field the character set is defined. See [Creating Bulk Import Rule](#) the *Voyager System Administration User's Guide* for more information.

## Leader Byte 9 Value

The Leader Byte 9 value specifies the character coding scheme used in the record. In the incoming record, if the value of the leader's byte 9 is an 'a', then the record is in the Unicode character set (UTF-8 encoded). If it is blank, or anything other than 'a', then the record is in a character set other than Unicode and requires converting.

## Converting Records to Unicode

The incoming records not already in the Unicode character set must be converted. The Bulk Import program attempts to convert these records one by one. The results of this conversion is either success, that is, a record is converted and then available for import, or failure, that is, the record is not converted and not able to be imported. A record fails if the program is unable to convert any part of it.



### IMPORTANT:

*The entire conversion process (and therefore the import process) aborts if the character set specified in the rule is not the same as the character set indicated in the leaders byte 9 in the first record. If a record is encountered later in the file where the leader byte 9 doesn't match the character set*

*specified in the bulk import rule, that record is sent to the error file and the job moves to process the next record.*

[Table 12-1](#) describes what occurs in the conversion process given a specific character set of incoming records, as specified in the Bulk Import Rule, and the Leader Byte 9 value.

**Table 12-1. Bulk Import Conversion Situations**

Character Set of Incoming Records from Bulk Import rule	Leader's Byte 9 Value	Conversion to UTF-8?	Processing Result
UTF-8	a	No conversion needed.	Proceed with import of the record.
UTF-8	Blank	No conversion attempted due to Character set and leader mismatch.	Error, see log.
OCLC	a	No conversion attempted due to Character set and leader mismatch.	Error, see log.
OCLC	Blank	Successful conversion.	Proceed with import of the record.
OCLC	Blank	Failed conversion.	Error, see log
RLIN Legacy	a	No conversion attempted due to Character set and leader mismatch.	Error, see log.
RLIN Legacy	Blank	Successful conversion.	Proceed with import of the record.
RLIN Legacy	Blank	Failed conversion.	Error, see log
Latin-1	a	No conversion attempted due to Character set and leader mismatch.	Error, see log.
Latin-1	Blank	Successful conversion.	Proceed with import of the record.
Latin-1	Blank	Failed conversion.	Error, see log
MARC-8	a	No conversion attempted due to Character set and leader mismatch.	Error, see log.

**Table 12-1. Bulk Import Conversion Situations**

<b>Character Set of Incoming Records from Bulk Import rule</b>	<b>Leader's Byte 9 Value</b>	<b>Conversion to UTF-8?</b>	<b>Processing Result</b>
MARC-8	Blank	Successful conversion.	Proceed with import of the record.
MARC-8	Blank	Failed conversion.	Error, see log
Voyager Legacy	a	No conversion attempted due to Character set and leader mismatch.	Error, see log.
Voyager Legacy	Blank	Successful conversion.	Proceed with import of the record.
Voyager Legacy	Blank	Failed conversion.	Error, see log

**NOTE:**

A warning may be provided when MARC21 characters which were formerly mapped to the Private Use Area (PUA) are encoded to Unicode. These warnings are informational and require no data clean up.

**Disabling the Generation of the Keyword Index Files**

When importing bibliographic records, Bulk Import simultaneously updates the Oracle tables as well as, the keyword index files. If importing a very large number of records, performance may be negatively affected. In order to improve the performance of the Bulk Import Utility, sites can run Bulk Import without the generation and maintenance of the keyword index files. Where the bibliographic records are added to the database, but not to the keyword index files. Holdings and Authority records are not indexed when importing.

The result of running Bulk Import with this option is that Keyword searching will not find these newly-imported records until a Keyword Index Regen is run.

If generation of the keyword index files is turned off for Bulk Import, it will not interfere with adding records using the Voyager modules. Therefore records created and edited using the staff modules are written to the keyword index and immediately accessible.

**IMPORTANT:**

*If you are certified at the 300 level (Certification course 300, provided by Ex*

*Libris) after completing the import you may run your own Keyword Index Regen. If, however, you are not certified, you must schedule a Keyword Index Regen with Ex Libris's Customer Support Department. **Schedule this at least a week before the import.***

Including the parameter and argument **-X NOKEY** in the command line disables the generation and maintenance of keyword indexes when running Bulk Import.

After running Bulk Import (it can be run more than once), a message displays on the screen at the end of the job stating that the keyword indexes are out of sync. Sites then have to perform a keyword index regen.



#### **CAUTION:**

*Because selecting this option impacts keyword searching and requires running keyword index regens after the import, Ex Libris does not recommend this. This should be used only when importing a great number of records at one time. If performance is affected, sites may want to decrease the number of records they are importing before using this option. Please consult with Ex Libris Customer Support.*

## **Parameters**

---

The following parameters govern the bulk import program.

**-f      Filename -- required.**

The filename containing the records you are importing. The default location of the file is the `/m1/voyager/xxxdb/sbin` directory. If the file is in a different directory, use the complete path.

**-i      Import code -- required.**

The Bulk Import Rule code. This is the code specified in the **Code** field, located on the **Rule Name** tab in the **Cataloging - Bulk Import Rules** section in the System Administration module. It instructs the system to use the Bulk Import Rule associated with the code specified and follow all of the rules defined therein.

This is also where you specify whether all loaded records should be suppressed from the OPAC and whether MARC holdings (MFHDs) and Voyager item records should automatically be created. MFHDs may be in different locations.

If the profile you select performs duplicate detection, note that if any single index listed in the duplication hierarchy in System Administration matches with more than 1000 records in the database, all duplicate detection will stop.

Only the first 100 records above the matching threshold will be returned to the client. There is no limit to the number of indexes that can be put in the hierarchy to check. However, this will hurt the accuracy of the matching being performed.

Also if the profile that you select has no indexes selected, the records will be added unconditionally to the database. See the *Voyager System Administration User's Guide* for more information on [Bulk Import Rules](#).

**-o      Operator name -- not required.**

The name of the operator importing the records. This information is recorded and used in Voyager to identify who last modified the record.

**-l      Location Code -- not required.**

The code for the cataloging happening location (as defined in System Administration module) that will be used in Voyager to identify the location from which the record was last modified.

**-b      Begin record -- not required.**

The first record in the file to be imported. For example, specifying the number 5 would instruct the program to begin importing from the fifth record in the file. This parameter is used with the -e parameter when importing fewer records than the entire file.

**-e      End record -- not required.**

The last record in the file to be imported. For example, specifying the number 10 would instruct the program to stop importing after the tenth imported record. This parameter is used with the -b parameter when importing fewer records than the entire file.

**-m      Load MFHDs -- not required. Must have interleaved file.**

Load MFHDs with bibliographic records from a single interleaved bib-MFHD file.

**-a      MFHD location code -- not required. Must have interleaved file.**

After the -a parameter enter a location code, for example **-aCIRC**. Then for all incoming MFHDs, the location code listed will be used in the MFHDs 852 field, subfield b. The location codes are defined in the Locations section of System Administration. The code used must match the code in System Administration exactly, that is it is case sensitive. Therefore in the example, if you use **-aCIRC**, then the MFHDs will have the location CIRC.

If used in conjunction with the -m parameter it will act as a match point, and then add a new MFHD with the location specified.

**-r      Delete MFHDs -- not required.**

Use this variable to delete specified MFHDs from your database.

The import file must be an interleaved file of bibliographic records and MFHDs. You cannot import a file of just MFHD records in an attempt to delete matching MFHDs from the database.

First, the incoming bibliographic records are matched with the bibliographic records on the database. For all matching bibliographic records, holdings records in the database with the same location code as the holdings records in the incoming file will be deleted. (The variable does not import records.)

**NOTE:**

You cannot delete records that are linked to an item record or purchase order.

**-x      Delete bibliographic records -- not required.**

This option is used only with Delete MFHDs (-r). To delete bibliographic records, both -x and -r should be entered as part of the same `Pbulkimport` command.

Use this variable to delete bibliographic records in your database that match the records in the data file.

The import file must be an interleaved file of bibliographic records and MFHDs.

This option does not import records. Records are only deleted from the database. The records in the file are matched with the records in the database.

This means that the location in the record in the data file must match the owning library of the bibliographic record in the database. If any of the matching bibliographic records in the database do not have any MFHDs attached to them (after having been deleted using the -r command), the bibliographic record in the database will be deleted.

If the import file contains only bibliographic records, it will delete those matching bibliographic records in the database if there is no linked MFHD and if the record is not linked to a purchase order.

This option is generally used by Universal Catalog databases to allow the local libraries to create files containing any records that they have suppressed or deleted from their database so that they can be removed from the Universal Catalog database as well. For more information see the *Voyager Universal Catalog User's Guide*.

**-k      OK to export -- not required.**

Use this variable to select the OK to export check box on the **System** tab of bibliographic, authority, and holdings records on view in the Cataloging module. The date on which each MARC record was last marked OK to export displays in the Cataloging module, **History** tab of the record on view in the Cataloging module.

**-X      NOKEY Disable keyword index generation and maintenance -- not required.**

This parameter and argument can be used when importing bibliographic records to disable the generation and maintenance of the keyword indexes.

**NOTE:**

This parameter is not available if running the job using WebAdmin.

**-h      Help -- not required.**

This parameter provides online help about the `Pbulkimport` function. This flag cannot be used with any other parameters.

**-p      Add Copy Number For New Item Records -- not required.**

This parameter specifies a single copy number to all newly created item records when it is used by itself. A number from 0 through 99999 (up to five places) may be specified consistent with the Cataloging client.

**NOTE:**

If `-p` is not used the copy number defaults to zero.

---

## Running Bulk Import

---

An example of a `Pbulkimport` command might be entered (on one line) as follows.

```
Pbulkimport -fmarc.rec -odeb -iADDCOND -b1 -e1200 -
1Main
```

This command instructs the system to import the records contained in the file named `marc.rec` numbered from 1 to 1200, following the ADDCOND bulk import rule as specified in the System Administration module, and give the operator name of deb and cataloging happening location of Main.

Also, if MFHDs and items are created the Call Number hierarchy set up in System Administration for the ADDCOND import/replace profile and the Item Type hierarchy will be used to create the appropriate Voyager item records for all MFHDs created.

## **Input File Specification**

---

The input file is a file of Bibliographic or Authority records, or an interleaved file of Bibliographic/Holdings record in MARC format.

## **Building the 035**

---

When you import bibliographic and authority records, Voyager creates an 035\$a field and adds it to the records. The 035\$a field identifies records from other systems. By creating a new field, Voyager preserves the source of the record. The system creates this field regardless of the method you use, bulk import or online import in the Cataloging module (see the *Voyager Cataloging* and the *Voyager Acquisitions User's Guides* for more information).

The following are the steps the system uses to create the 035\$a field when you import records.

1. The system first looks for the 003 field (source library's code name).
2. If there are no parentheses around the 003 field, the procedure adds parentheses.
3. The system looks for the 001 field (source's system ID number).
4. The system combines the 003 and 001 fields to create the 035\$a field, thus preserving the original source of the records.

For example,

Source 003 = OCoLC  
Source 001 = ocm 12345678  
035\$a field = (OCOlc)12345678

**NOTE:**

If the record has no 003 field, it creates the 035\$a field using only the 001 field. If the record has no 001 field, the system cannot create the 035\$a field.

## OCLC Control Number Expansion

---

To accomodate the growth of the Worldcat® database, the OCLC Control Number has been expanded to allow 9-digit control numbers. A new prefix of ocn has been added to the 001 field to indicate a 9-digit number.

For example,

Source 003 = OCoLC

Source 001 = ocn 123456789

035\$a field = (OCOlc)123456789

## Creating Holdings and Item Records

---

You can specify a call number, barcode, and item type hierarchy in System Administration that will be used with a respective Import/Replace Profile.

### Call Number Hierarchy

---

This function is only available if the **Create MFHDs/items** check box in the Bulk Import Rules is selected.

The Call Number hierarchy defines which values will be placed in the 852 \$h \$i of the MARC holdings record, if one is created.

This is accomplished by creating a call number hierarchy(ies) and assigning the hierarchy to a bulk import rule.

The hierarchy will look for the call number in the fields you have entered and then assign the call number to the 852 field of the holdings record. Voyager looks at the first profile and stops if it finds a match.

If Voyager does not find a match, it continues looking at the following profile. If Voyager finds that the tag appears more than once in the record (for example, multiple 086 tags), the last one listed in the record will be used.

The profiles are ordered by sequence (from cat\_control\_call no) in Ascending order.

The **Bulk Import Rule** must use the **Duplicate Profiles** selected in the System Administration module. Those duplicate profiles are set up from the **System Administration> Cataloging configuration** dialog box and monitors the creation of bibliographic and authority records. There is no limit to the number of duplicate

profiles that you can create. However, the profiles assigned to the import code will be used. The authority duplicate profile does not need to be chosen if the import code is used only for bib import and vice versa.

## **Barcode**

---

The Barcode rules look in the MARC bibliographic record for a barcode number that you have defined for the field and subfields in the bulk import rules. For example, a barcode rule might look in the 949 field subfield a. You can create a hierarchy of barcode rules that the system will check until it finds a match. See the *Voyager System Administration User's Guide* for more information.

## **Item Type**

---

You can identify the fields and subfields that should be checked for an item type and which Voyager item type that will be associated with it. See [Item Types](#) in the *Voyager System Administration User's Guide* for more information.

## **Mapping**

---

The mapping allows you to create a hierarchy to look for location and item information in the imported records and then map those item and location descriptions in your library. See the *Voyager System Administration User's Guide* for more information.

## **Additional Files**

---

The bulk import program creates an error and log file, as well as other files depending on the Add or Replace duplicate handling option selected in the duplicate detection profile in the System Administration module.

### **Log file**

---

After the `Pbulkimport` job is complete, an activity summary is written to a log file. This file contains a summary, which is broken down between Bibliographic and Authority records in the first group and Holdings records in the second group. The log file includes the number of:

```
records processed  
records added (to the database)  
records discarded (to the discard.imp file)
```

*records rejected* (written to the `reject.imp` file, does not apply to holdings records)

*records that caused errors* (written to the `err.imp` file)

*records replaced or merged* (written to the `replace.imp` file)

*records deleted* (written to the `delete.imp` file)

This log file also includes information about any records that fail conversion in the bulk import process. See [Messages in Log and Error Files](#) on [page 12-15](#).

The default filename is `log.imp.yyyymmdd.hhmm`, where `y` is the year, `m` is the month, `d` is the day, `h` is the hour, and `m` is the minute, and it is placed in the `/m1/voyager.***db/rpt` directory.

## Delete, Discard, Replace, and Reject files

---

The bulk import program creates the following files, depending on the Add or Replace option selected in the Import/Replace profile in the System Administration module.

### **NOTE:**

The files are named based on the time and date that `Pbulkimport` was run. Because the files are named with the minute and not the seconds, if bulk import is run more than once per minute, the files will be overwritten with the information from the latest run.

### **Delete**

An incoming record will be put in the delete file if it has a delete symbol in the record.

The filename is `delete.imp.ccyyymmdd.hhmm`, where `c` is the century, `y` is the year, `m` is the month, `d` is the day, `h` is the hour, and `m` is the minute, and it is placed in the `/m1/voyager.***db/rpt` directory.

### **Discard**

An incoming record will be put in this file if it could not be added, merged, or replaced because there are multiple records above the replace threshold or the warning threshold.

The filename is `discard.imp.ccyyymmdd.hhmm`, where `c` is the century, `y` is the year, `m` is the month, `d` is the day, `h` is the hour, and `m` is the minute, and it is placed in the `/m1/voyager.***db/rpt` directory.

### **Replace**

An existing database record will be put in this file if it is replaced by or merged with an incoming record.

The filename is `replace.imp.ccyyymmdd.hhmm`, where c is the century, y is the year, m is the month, d is the day, h is the hour, and m is the minute, and it is placed in the `/m1/voyager/xxxdb/rpt` directory.

### **Reject**

An incoming record will be put in this file if it could not be added, merged, or replaced because it has a lower quality rating than the existing record.

The filename is `reject.imp.ccyyymmdd.hhmm`, where c is the century, y is the year, m is the month, d is the day, h is the hour, and m is the minute, and it is placed in the `/m1/voyager/xxxdb/rpt` directory.

### **Error File**

---

When Bulk Import encounters records during import that cannot be processed, they are written to the error file.

The default filename is `err.imp.yyyymmdd.hhmm`, where y is the year, m is the month, d is the day, h is the hour, and m is the minute, and it is placed in the `/m1/voyager.ddd/rpt` directory.

This file includes any records that were incorrectly formatted MARC records found during the `Pbulkimport` job. These errors can include the messages listed below.

### **Messages in Log and Error Files**

---

There are many messages that may display in the log file after running bulk import.

#### **Authority record has a status of D, S, or X written to reject.imp file**

An authority record with this status cannot be imported.

### **DB Bib Record cancelled/deleted by Import Record!**

The Import/Replace profile dictated that an existing record in the database should be removed and replaced by the imported record, based on the Cancellation code selected. The record from the database is written to the delete file.

#### **Delete Failed - Items Attached**

A bibliographic record with attached items cannot be deleted.

#### **Delete Failed - MFHDs Attached**

A bibliographic record with attached holdings records cannot be deleted.

#### **Delete Failure**

If you receive this message, contact Ex Libris Customer Support.

### **Duplicate detected. Discarding dupe import record!**

The Import/Replace profile dictated that an imported record should never overwrite a matching record in the database. The imported record is written to the discard file.

#### **Duplicate detection failure!**

If you receive this message, contact Customer Support.

#### **Duplicates above replace threshold. Adding anyway**

The Import/Replace profile dictated that any duplicate records will be ignored and that the imported record will always be added to the database (Add-Unconditional).

#### **Duplicates above thresholds. Cannot resolve!**

The Import/Replace profile dictated that if more than one duplicate above the replace threshold is detected, the imported record should not be saved to the database. The record is sent to the discard file.

#### **Existing database record replaced by imported**

The Import/Replace profile dictated that an imported record will always overwrite a single matching record in the database (Replace) and the record is overwritten.

### **Failed to load converter for <character set>**

Mapping characters from the listed character set to UTF-8 is not working. If you receive this message, contact Ex Libris Customer Support.

### **No Matches found for Input Cancel Record. Import Record Discarded!**

The Import/Replace profile dictated that an existing record in the database should be found which matches the imported record and be then replaced, based on the Cancellation code selected. The imported record is written to the discard file.

### **No item option in use - No item records created**

No items will be created during this run of Bulk Import because Load Bib/MFHD is the option specified in the Bulk Import Rules.

### **Non-Bibliographic Record detected! Holdings record written to error file**

The record is not a bibliographic record type. The record is written to the error file. If you are trying to load or delete MFHDs, make sure the appropriate switch is included on the command line.

### **Record *nnn* discarded -- no matching records in the database**

Because a duplicate detection profile which had the discard incoming records option checked, record *nnn* was discarded. For more information on the discard incoming records option, see [Bulk Import Rules](#) and [Bibliographic Duplicate Detection](#) in the *Voyager System Administration User's Guide*.

### **Record does not match format for '<character set>' change your import rule**

The records do not match the input rule character set, if this is the first record then it will exit, otherwise it will just reject the record.

### **Record Retrieval Failure**

If you receive this message, contact Ex Libris Customer Support.

### **Record Parse Failure**

If you receive this message, contact Ex Libris Customer Support.

### **Unparseable record written to error file**

The Pbulkimport command was unable to determine what was being imported. The unrecognizable record is written to the delete file.

If the error involved the conversion, the error messages are similar to the output logs from the conversion process, see [Warning and Error Messages on page 4-6](#). The differences are that bulk import does not give the field index and loose translations are errors for bulk import.

<b>Introduction</b>	<b>13-1</b>
<b>Purpose of This Chapter</b>	<b>13-1</b>
<b>Overview of GHC</b>	<b>13-1</b>
<b>Parameters</b>	<b>13-2</b>
• Process Global Heading Change Queue Step 1 (Catjob 11)	13-4
• Process Global Heading Change Queue Step 2 (Catjob 12)	13-5
• Process Global Heading Change Fields Step 3 (Catjob 13)	13-6



---

## **Introduction**

---

Global Headings Change (GHC) is a way to update all the name, title, name/title, or subject fields of Bibliographic records, based on a change in an Authority record.

---

## **Purpose of This Chapter**

---

This chapter provides

- Overview of the Global Headings Change process
- Parameters used to govern the GHC jobs
- Step 1: Process Global Heading Change Queue- Catjob 11
- Step 2: Process Global Heading Change Queue- Catjob 12
- Step 3: Process Global Heading Change Fields- Catjob 13

---

## **Overview of GHC**

---

Global Headings Change (GHC) updates the name, title, name/title, or subject fields of Bibliographic records, based on a change in an Authority record.

When a change is made to an authority record the Global Change Queue (in the Cataloging module) displays the change as well as the associated records. This allows the user to decide which changes to implement.

GHC is basically a three step process.

- the bibliographic records that contain the old heading to be changed are found,
- a preview of the change is available,
- the change is actually made.

See the *Voyager Cataloging User's Guide* for information about Global Headings Change and Authority Control.

The GHC process consists of running a job on the server, then doing some activity in the cataloging module.

The catjobs are run from the `/m1/voyager/xxxdb/sbin` directory on your server.

After running each job, entries are made in the `cat.job.log` file, located in the `/m1/voyager/xxxdb/rpt` directory.

**▲ IMPORTANT:**

*Jobs 11, 12, and 13 MUST BE RUN IN THAT ORDER for a specific heading change to be complete.*

Any records that are to be processed through the GHC queue must not be manually changed between the steps of the batch jobs. If any changes are made to a record it will be removed from the queue.

**NOTE:**

These server jobs can be run using WebAdmin, see [Cataloging Utilities](#) on [page 30-12](#).

## Parameters

---

The following parameters govern the Global Headings Change batch jobs.

**-d      Database name -- not required.**

Automatically specified by the script

It is the name of the database that will be updated. You must have write access on the server to run the GHC program.

**-u      Username and password -- not required.**

Automatically specified by the script. Specifying the username and password with this parameter overrides the name and password in the script.

It is the username and password for access to the specified database. Enter it in this format: username/password

**-j      Job number -- required.**

Number of the job that you want to run.

**-L      Location code -- not required.**

Used for job 13. It is the cataloguing location code for the cataloguing happening location. This updates the record's location code in the **History** tab of the bibliographic record. If prompted for a code, press the **Enter** key if you do not want a location associated with the update.

**NOTE:**

The implication of not using a location when running Catjob 13 is:

- if this site contributes to a Universal Catalog (UC), *and*
- when importing into that UC you've set the Catjob parameter such that the system only looks at the last (update) location, *and*
- there are specified locations from which records should be excluded, that is not imported into the UC, *then* this record would be exported.

**-o      Operator -- not required.**

Used for job 13. This updates the record's update operator in the **Record History** dialog.

**-I      List job options -- not required.**

List of job options.

**-v      Version -- not required.**

Display version information.

**-h      Help -- not required.**

Display help and usage statement.

### **Process Global Heading Change Queue Step 1 (Catjob 11)**

---

This batch job begins the Global Headings Change process by finding all of the possible headings to be changed. The new heading will appear in the GHC queue (in the Cataloging module) as it would appear in all Bibliographic records with that heading, and it will show how many Bibliographic records are associated with the heading.

This job should be run from the `/m1/voyager/xxxdb/sbin` directory on your server. At the `/sbin>` enter

**Pcatjob -j11**

After running this job the operator should look at the GHC queue (in the Cataloging module), if you click the plus sign you will see the new heading.

Select the box marked **Process**.

It writes to the `catjob.log` file that is placed in the `/m1/voyager/xxxdb/rpt` directory.

The `catjob.log` should include the day, date, time entry, and the messages:

*Fri Dec 28 09:50:32 2001 Job execution begun.*

*Fri Dec 28 09:50:34 2001 Connection to Voyager Database successful...*

*Fri Dec 28 09:51:21 2001 process global heading change queue*

*Fri Dec 28 09:51:21 2001 Starting heading change queue processing...*

*Fri Dec 28 09:51:22 2001 Completed heading change queue processing.*

*Fri Dec 28 09:51:58 2001 Job execution complete, or*

*Fri Dec 28 09:51:28 2001 No heading queue entries to process at this time!*

Now you are ready to run Catjob 12.

## Process Global Heading Change Queue Step 2 (Catjob 12)

---

This batch job prepares the preview of all fields in the bibliographic and authority records to be changed.

This job should be run from the /m1/voyager/xxxdb/sbin directory on your server. At the /sbin> enter

**Pcatjob -j12**

After running the job the operator should look in the GHC queue (cataloging module) to see the proposed changes.

Click the **Preview** button to look at the bibliographic records to be changed. At this point you can delete any records you want from this list.

Select the **Process** box on the headings in which you would like the change to be reflected.

It writes to the catjob.log file that is placed in the /m1/voyager/xxxdb/rpt directory.

The catjob.log should include the day, date, time entry, and the messages:

```
Fri Dec 28 10:17:08 2001 Job execution begun.  
Fri Dec 28 10:17:09 2001 Connection to Voyager Database  
successful...  
Fri Dec 28 10:17:11 2001 process global heading changes  
Fri Dec 28 10:17:11 2001 Starting heading change  
processing...  
Fri Dec 28 10:17:12 2001 Total Bibs processed: 1  
Fri Dec 28 10:17:12 2001 Total Auths processed: 0  
Fri Dec 28 10:17:12 2001 Completed heading change  
processing.  
Fri Dec 28 10:17:15 2001 Job execution complete.
```

Now you are ready to run Catjob 13

### **Process Global Heading Change Fields Step 3 (Catjob 13)**

---

This batch job updates the bibliographic and authority records if the heading was marked for processing in the cataloging module's global headings change queue.

This job should be run from the /m1/voyager/xxxdb/sbin directory on your server. At the /sbin> enter

```
Pcatjob -j13 -L(location code) -o(Operator)
```

Neither the -L parameter, **Location code** nor the -O parameters are required. If used, the location and operator are logged in the **History** tab of the Bibliographic record.

**NOTE:**

If running the job from the menu, the system prompts for a Location and Operator code even though they are not required, operators should press the **Enter** key to indicate that no location or operator should be associated with the update.

**▲ IMPORTANT:**

*This step is very machine intensive. The time it takes to run is site-specific, and depends on the time of day and usage of the system. Run this job during slow use periods.*

After the job has run, go back into the Cataloging module and look at the GHC queue, notice that they are gone as the changes have been made. You can also search for a bibliographic record that was to change and verify that it has changed appropriately.

Any headings still listed in the queue have not been processed. This means that the system recognizes that a change to the authority record has happened but the changes have not been made to the corresponding bibliographic records.

It writes to the catjob.log file that is placed in the /m1/voyager/xxxdb/rpt directory.

The catjob.log should include the day, date, time entry, and the messages:

```
Fri Dec 28 10:20:48 2001 Job execution begun.
```

```
Fri Dec 28 10:20:49 2001 Connection to Voyager Database successful...
```

```
Fri Dec 28 10:20:51 2001 process global heading change fields
```

```
Fri Dec 28 10:20:56 2001 Starting heading change fields  
processing...  
Fri Dec 28 10:21:01 2001 Total records processed:  
3  
Fri Dec 28 10:21:01 2001 Total records left  
unprocessed: 1  
Fri Dec 28 10:21:01 2001 Completed heading change field  
processing.  
Fri Dec 28 10:21:08 2001 Job execution complete.
```



---

## **Storage Barcode Verify (Pstrgvfy) Program**

# **14**

---

<b>Introduction</b>	<b>14-1</b>
<b>Purpose of This Chapter</b>	<b>14-2</b>
<b>Changing Location Codes of Item Records or MFHDs</b>	<b>14-2</b>
<b>Populating the Operator or Location Columns of a MFHD's History Tab</b>	<b>14-3</b>
<b>Verifying Barcodes of Item Records or MFHDs</b>	<b>14-4</b>
<b>Pstrgvfy Input File</b>	<b>14-4</b>
<b>Pstrgvfy Parameters</b>	<b>14-5</b>
<b>Pstrgvfy Command File</b>	<b>14-8</b>
<b>Pstrgvfy Log File</b>	<b>14-9</b>
<b>Sample Pstrgvfy Command Line</b>	<b>14-10</b>



---

## Storage Barcode Verify (Pstrgvfy) Program

# 14

---

### Introduction

---

The Storage Barcode Verify (Pstrgvfy) program serves three purposes

- To change the location codes of item records or item records and MFHDs.
- To populate the Operator or Location columns of a MFHD's **History** tab in the Cataloging module (if the location codes of MFHDs are changed).
- To verify the barcodes of item records or item records and MFHDs.

When you run the change location command, the program also checks for duplicate barcodes. When you run the verify barcode command, the program checks for duplicate barcodes as well as verifies the location code and status.

Sites must create an input file consisting of the barcodes associated with records where you want to change the location code, populate the **History** tab, or verify the barcode.

#### NOTE:

Because of Pstrgvfy's ability to change location codes and verify barcodes, it is used to load and verify barcodes in the Automated Retrieval System (ARS). The Automated Retrieval System (ARS) is an extension module used to handle storage and accessibility to library materials that are warehoused at remote storage facilities. For detailed information, see the *Voyager Automated Retrieval System (ARS) User's Guide*.

## Purpose of This Chapter

---

This chapter discusses

- Changing location codes of item records or MFHDs
- Populating the operator or location columns of a MFHD's **History** tab
- Verifying barcodes of item records or MFHDs
- Input file
- Command line parameters
- Command file
- Log file
- Sample command line

## Changing Location Codes of Item Records or MFHDs

---

`Pstrgvfy` allows you to automatically change the location codes of item records or item records and MFHDs (852|b field).



### IMPORTANT:

*Although you can change the location codes of item records alone, you cannot change MFHD location codes alone. MFHD location codes must be changed with item record location codes simultaneously.*

The results of the location code change are reported in the log file (if a duplicate barcode is found, the log file will include information to that effect as well as the barcode duplicated).

You specify the item records or item records and MFHDs for which you want the location codes changed in an input file. This input file contains a list of barcodes that are associated with the item records or item records and MFHDs. The location code to which you want the item records or item records and MFHDs changed is specified at the `Pstrgvfy` command line (using the -m parameter).

For detailed information on locations and location codes, see the *Voyager System Administration User's Guide*.

Changing the location code of item records or item records and MFHDs using `Pstrgvfy` is particularly useful if you are moving many materials from one physical location to another. For example, if you are pulling many items from your

main stacks (with a location code of Main) and placing them in a special collection (with a location code of SpColl2), Pstrgvfy allows you to change the location of the associated item records or item records and MFHDs (852|b field) from Main to SpColl2 without having to manually wand in each item and change the location yourself. In this way, Pstrgvfy saves a significant amount of time and effort.

**NOTE:**

Pstrgvfy does not change the location for a line item on a purchase order. The line item location, which contains the serials history, populates the MFHD location when it is created. If Pstrgvfy is used to change the location of the MFHD/item record, then it will not match the line item location. When the line item location and the MFHD location are different, WebVoyage displays the holdings information based on the MFHD location. Therefore, if you change the MFHD/item record location using Pstrgvfy, the serials history will not display. You can change the line item location using the Acquisitions module. When the locations are the same, both the holdings information and the acquisitions serials history will display in WebVoyage. See the *Voyager Acquisitions User's Guide* for more information.

## **Populating the Operator or Location Columns of a MFHD's History Tab**

---

If you change the location codes of MFHDs using Pstrgvfy, you can opt to specify the operator name or Cataloging location code that will populate the Operator or Cataloging Location columns on the **History** tab of MFHDs in the Cataloging module. See the *Voyager Cataloging User's Guide* for details on the MFHD **History** tab. This is useful for record keeping purposes, as it allows you to keep track of which operators have made changes to MFHDs, and at what location.

**NOTE:**

This does not apply to item records, as item records do not have a **History** tab. In addition, it does not apply if you have verified barcodes related to MFHDs using Pstrgvfy.

You specify the operator name or Cataloging location code at the command line following the -o or -g parameters respectively. If you do not specify an operator name, the default (SYSTEM) will populate the Operator column. If you do not specify a Cataloging location code, the Cataloging Location column will be empty.

## Verifying Barcodes of Item Records or MFHDs

---

`Pstrgvfy` allows you to compare and verify barcodes related to item records, or item records and MFHDs, at specific locations. The verification process includes

- Checking for duplicate barcodes
- Ensuring that the related item records, or item records and MFHDs, have the correct location code
- Ensuring the barcodes do not have an inactive status

The results of the verification are reported in the `Pstrgvfy` log file (if a duplicate barcode is found or if the barcode is at the wrong location, the log file will include information to that effect as well as the barcode duplicated).

You specify which barcodes you want verified using an input file. The barcodes in the input file are compared with barcodes related to item records, or item records and MFHDs, with the location code you specify at the command line (via the `-m` parameter).

If you want to compare the barcodes in the input file with barcodes related to item records, or item records and MFHDs at multiple locations, you can create a command file (as opposed to specifying multiple `-m` parameters at the command line). This command file contains a list of `-m` parameters and location codes.

### **Pstrgvfy Input File**

---

You must create an input file to use `Pstrgvfy`. This input file is an ASCII flat file that you can create in any text editor, such as Microsoft® Notepad.

It consists of a list of barcodes that you wand in or enter by some other means. No other information should be included in the input file.

Depending on what you want `Pstrgvfy` to do (and of course, which command line parameters you use), `Pstrgvfy` will take the list of barcodes in the input file and either change the location code of the associated item records or item records and MFHDs, or verify the barcodes in the file against those associated with item records, or item records and MFHDs.

If you are changing location codes you can also command `Pstrgvfy` to populate the Operator or Location columns of the MFHD's **History** tab in the Cataloging module.

**⚠️ IMPORTANT:**

*Each barcode in the input file must be placed on its own line.*

By default, the input file will be taken from the /m1/voyager/xxxdb/rpt directory. You can place the input file in a different directory, in which case you need to specify the directory at the command line (with the -i parameter).

## Pstrgvfy Parameters

---

The following parameters govern the Pstrgvfy program.

**-d      Database name -- not required.**

Automatically specified by the Pstrgvfy script from the voyager.env file. It is the name of the database that will be accessed. You must have read-access on the server to run the extract program.

**-u      Username and password -- not required.**

Automatically specified by the Pstrgvfy script from the voyager.env file. It is the username and password for access to the specified database. Enter it in this format: username/password.

**-i      Input file -- required.**

If the input file is located in a directory other than the default (/m1/voyager/xxxdb/rpt), the -i is used to specify its path and filename with extension.

**-m      Location -- required.**

This is only valid with the -a or the -v.

When combined with the -a, -m is used to specify a single location code to which Pstrgvfy will set item records, or item records and MFHDs, related to the barcodes in the input file. Only one -m can be used with the -a. If you use more than one, Pstrgvfy will only apply the first one.

When used with the -v, -m will verify the barcodes listed in the input file against those associated with item records or MFHDs for the location you specify. You can specify multiple locations, in which case you should create a command file (as opposed to using multiple -m parameters at the command line).

**NOTE:**

Location codes are case sensitive and must appear exactly as they do in System Administration.



**IMPORTANT:**

*The -m parameter can be used with either the -a or the -v parameters but it cannot be used with both in the same command.*

**-a Alter/update mode flag -- either -a or -v is required.**

This is only valid with the -m; cannot be used with the -v.

When used with the -m, this stand-alone parameter (no qualifying data needed) is used to change the location codes of item records, or item records and MFHDs, related to the barcodes listed in the input file. You must also use the -b parameter if you want to change the location code of MFHDs. If you do not, only the location code of item records will be changed.

If a duplicate barcode is found, it will be counted as bad in the log file, and its location will not be changed or added.

**-b Update of MFHD 852b -- not required.**

This is only valid with the -a; cannot be used with the -v.

When used with the -a and the -m, this stand-alone parameter (no qualifying data needed) is used to change the location code of MFHDs (in addition to item records) that are related to barcodes in the input file. `Pstrgvy` will set the 852|b field of the MFHDs to the location code specified using the -m parameter. If the location code in the 852|b field matches the one specified via the -m, no change will be made. `Pstrgvy` will acknowledge that no change was made in the log file.

If you do not use this parameter, only the location codes of item records will be changed.

**-v Verify mode flag -- either -v or -a is required.**

This is only valid with the -m; cannot be used with the -a.

This stand-alone parameter (no qualifying data needed) is used to verify barcodes in the input file against barcodes associated with item records or item records and MFHDs with the location code you specify with the -m, (or with multiple location codes you specify in a command file). Using the -v, `Pstrgvy` will check for

duplicate barcodes, ensure that the associated item records or item records and MFHDs have the location code you specify, and ensure that the barcodes do not have an inactive status.

**-o      Operator ID -- not required.**

This is only valid with the -b.

The -o is used to specify an operator name that will populate the Operator column on the MFHD's **History** tab, (indicating which operator altered the MFHD). This does not apply to item records, as item records do not have a **History** tab. You define operator names in the System Administration module. See the *Voyager System Administration User's Guide* for more information.

If you do not specify an operator name with the -o, or if the operator name is invalid, the default (SYSTEM) will be used.

**-g      Cataloging Location -- not required.**

This is only valid with the -b.

With the -g you can specify a Cataloging location code (case sensitive) that will populate the Cataloging Location column on the MFHD's **History** tab in the Cataloging module. This does not apply to item records, as item records do not have a **History** tab.

The location specified with the -g must be a valid location code as defined in System Administration module. If it is invalid (for example if you misspell it at the command line), Pstrgvfy will stop. Pstrgvfy will acknowledge that the location code is invalid in the log file.

If the location is not a valid Cataloging happening location, Pstrgvfy will process the input file of barcodes but will not populate the Cataloging Location column on the **History** tab of the MFHD. Pstrgvfy will acknowledge that the location is not a valid Cataloging happening location in the log file. You define locations as Cataloging happening locations in the Cataloging Policy Groups portion of the System Administration module. See the *Voyager System Administration User's Guide*.

If you do not specify a Cataloging location with the -g, the Cataloging Location column of the MFHD will be empty.

**-c      Command filename -- not required.**

This is only used with the -v.

With the -c you specify the path and filename (with extension) of a command file to be referenced by `Pstrgvy`. This command file contains a list of -m parameters with location codes from which you want barcodes in the input file to be compared and verified.

The -c is only used if barcodes of item records or MFHDs are being verified (with the -v parameter); not added/changed.

**-l      Log filename -- not required.**

If you do not want the `Pstrgvy` log file written to the default directory (`/m1/voyager/xxxdb/rpt`), you can specify a path and filename (with extension) using the -l.

**-h      Help -- not required.**

The -h provides a listing and brief description of all the valid parameters for the `Pstrgvy` script.

## **Pstrgvy Command File**

---

If you want to compare and verify barcodes in the input file against those related to item records or item records and MFHDs at more than one location, you can create a `Pstrgvy` command file. That way, you do not have to use multiple -m parameters at the command line followed by multiple location codes. The command file will only be referenced by `Pstrgvy` if the -c parameter is used at the command line.

You can create the command file in a text editor such as Notepad. The command file has the .com file extension (`strgvy.com`), and by default, resides in the `/m1/voyager/xxxdb/local` directory. You can place the command file in a directory other than the default, in which case you must specify a full path name to the file (with extension) at the command line using the -c parameter. For example:

`-c/m1/voyager/xxxdb/temp/strgvy.com`

where `strgvy.com` is the name of the command file.

**NOTE:**

The command file should not reside in the `/m1/voyager/xxxdb/sbin` directory as that directory is used primarily for the Pscripts.

The command file contains a list of -m parameters, each followed by a single location code. Each -m and location code appears on its own, separate line exactly as it does in the System Administration module, as in the following example:

```
-m Main  
-m SpColl2  
-m SpColl3
```

## Pstrgvfy Log File

---

A log file called `log.strgvfy.date.time`, is automatically created by `Pstrgvfy`. It includes the following information:

- A count of all the barcodes changed or verified for the locations specified
- The date and time the run started and ended
- (Depending on whether or not you changed location codes or verified barcodes) information about problems `Pstrgvfy` encountered during its run (for example, invalid location codes, duplicate barcodes, barcodes at wrong locations.)

By default, `Pstrgvfy` places the log file in the `/m1/voyager/xxxdb/rpt` directory. You can specify another filename and path to which you want the log file written using the -l parameter at the command line.

### NOTE:

If you specify another filename and path using the -l command line parameter, the date and time will not be included as a part of the `Pstrgvfy` log filename.

An example of the contents of a `Pstrgvfy` log file is:

```
.Barcode Alter.Verify Processing: Start Time: Mon June  
20 08:58:00 20 00  
Error: Location Id Lookup/ <SpColl1>  
Error: Invalid location code./ <SpColl4>  
Error: Barcode at Wrong location./ <CIRC>  
Error: Barcode is duplicated./ <39550000123456>  
Error: Barcode at Wrong location./ <CIRC>  
Error: Barcode at Wrong location./ <CIRC>  
Error: Barcode at Wrong location./ <CIRC>  
Error: Barcode at Wrong location./ <CIRC>
```

```
Final counts: Read: 6/Valid: 0/Bad: 6/
Barcode Alter/Verify Processing: End Time: Mon June 20
08:58:01 2000
```

This sample log file pertains to a run that changed location codes.

## **Sample Pstrgvfy Command Line**

Pstrgvfy is run from the /m1/voyager/xxxdb/sbin directory on the Voyager server.

An example command line might be, at /m1/voyager/xxxdb/sbin> enter

```
Pstrgvfy -i/m1/voyager/testdb/local/
out.strg.2000405.1018 -mSpColl2 -a -b -gCatOp -oSpColl2
```

This instructs Pstrgvfy to reference an input file located in the /m1/voyager/testdb/local directory, called out.strg.2000405.1018, change the location code of both item records and MFHDs associated with barcodes in the input file to SpColl2, populate the operator name and Cataloging location code on the MFHD's **History** tab with CatOp and SpColl2 respectively.

---

<b>Introduction</b>	<b>15-1</b>
<b>Purpose of This Chapter</b>	<b>15-1</b>
• OPAC_Search_Log Table	15-2
• Bib_usage_log Table	15-3
• Setting Up OPAC Search Logging	15-4
• Setting Up Bibliographic Usage Logging	15-5
• Setting up Selective Dissemination of Information (SDI)	15-6
<b>Accessing Logged Information</b>	<b>15-6</b>
• OPAC Search Log Export program	15-6
Interactive Method	15-7
Cron Method	15-8
• SDI Searches Program	15-9
Enabling SDI	15-10
Configuring SDI Options	15-10
Customizing the Search Results URL Page Sent to Patrons by SDI	15-12
Interactive Method	15-14
Cron Method	15-15
• OPAC Bib Usage Log Export Program	15-15
Interactive Method	15-15
Cron Method	15-17



---

## Introduction

---

Included with Voyager is the Popacjob program, which is comprised of three parts.

- The first, OPAC Log Export (formerly *Popaclogexp*), pulls the specified information from the OPAC\_search\_log table in the database into a comma-delimited text file. The information in the comma-delimited text file is presented in [Table 15-1 on page 15-2](#).
- The second, SDI Searches, runs search queries stored by patrons at the appropriate intervals. SDI allows patrons to choose intervals for each saved search query to be run automatically (on the local database) and the results e-mailed to them in the form of a URL.
- The third, OPAC Bib Usage Log Export (formerly performed using the -b switch when running *Popaclogexp*), pulls the specified information from the bib\_usage\_table in the database into a comma-delimited text file. The information in the comma-delimited text file is presented in [Table 15-2 on page 15-3](#).

---

## Purpose of This Chapter

---

- The purpose of this chapter is to detail the 3 components of Popacjob, including configuration and running the jobs as crons.

## **OPAC\_Search\_Log Table**

For OPAC search logging, the following information is stored in the OPAC\_search\_log table in the database for every search (see [Table 15-1](#)).

**Table 15-1. OPAC\_search\_log Table**

FIELD	DESCRIPTION
Search_date	Search date and time.
Stat_string	The requesting client's 10-character ID number from the OPAC.ini file (see step 2 of "Setting Up OPAC Search Logging" on <a href="#">15-4</a> ).
Session_id	String generated by OPAC client when started or reset (yyyymmddhhmmss)
Search_type	Type of search performed (Author, Subject.).
Search_string	Query entered as the criteria for the search.
Limit_flag	Were limits in affect? Y/N
Limit_string	Details on limits.
Index_type	A=authority, B=browse, K=keyword or L=left-anchored.
Relevance	Were the results relevance ranked? Y/N
Hyperlink	Was the search the result of clicking a hyperlink? Y/N
Hits	The number of hits returned by the search. <b>Note:</b> Hit count will not apply to Heading Subject and Heading Call Number searches which are browse type searches of an entire index. For these searches, a "-1" will be recorded.
Search_tab	0-5 search tab number.
Client_type	W=Web OPAC A=ASCII OPAC Z=Z39.50
Client_ip	The IP (Internet Protocol) address of the requesting client, for example, 128.218.1.38.
dbkey	Unique identifier of the database which the user performed the search against.
Redirect Flag	Y= Search performed was redirected. N= Search performed was not redirected.

## Bib\_usage\_log Table

For bibliographic usage logging, the following information is stored in the bib\_usage\_log table in the database (see [Table 15-2](#)).

**Table 15-2. Bib\_usage\_log Table**

FIELD	DESCRIPTION
Client_type	C=Cataloging Z=Z39.50 W=Web OPAC A=ASCII OPAC
Use_date	The date and time on which the usage of the bibliographic record occurred.
Operator_id	The ID of the operator who instigated the transaction of the bibliographic record.  <b>Note:</b> This information will be recorded only for Cataloging clients, not for OPAC clients.
Location_id	The id of the location from which the transaction of the bibliographic record was done.  <b>Note:</b> This information will be recorded only for Cataloging clients, not for OPAC clients.
Session_id	String generated by OPAC client when started or reset (yyyymmddhhmmss).
Stat_string	The requesting client's ten-character ID number from the opac.ini file (see step 2 of "Setting Up OPAC Search Logging" on <a href="#">15-4</a> ).  <b>Note:</b> This information will be recorded only for OPAC clients, not for Cataloging clients.
Client_ip	The IP (Internet Protocol) address of the requesting client, for example, 128.218.1.38.  <b>Note:</b> This information will be recorded only for OPAC clients, not for Cataloging clients.
Bib_id	The id of the bibliographic record involved in the transaction.

**Table 15-2. Bib\_usage\_log Table**

FIELD	DESCRIPTION
Use_type	M=save, MARC format D=display S=save, text format E=e-mail P=print Z=Z39.50 retrieval  <b>Note:</b> S and E will be recorded only for OPAC clients, not for Cataloging clients, because a bibliographic record cannot be saved in text format or e-mailed from Cataloging. Also, P will not be recorded for Web OPAC because printing is done by the web browser.

The information stored in the above mentioned tables can be accessed in one of two ways: by querying the database using MS Access (through linking tables) or with Popacjobs 1 and 3 (see [Accessing Logged Information](#) on [page 15-6](#)). Any SQL software can query the database for the appropriate information.

**NOTE:**

Search logging takes up a lot of drive space. A single record is around 600 characters in length; saving 1,000 searches to the database requires 73 megabytes of hard-drive space. If you plan on keeping Search Logging activated for any length of time, you should plan on regularly extracting information and purging it from the database (by specifying option -p).

## **Setting Up OPAC Search Logging**

Use the following to set up OPAC search logging.

1. Turn the Search Logging feature on.

In the /ini directory on the server (/m1/voyager/xxxdb), the voyager.ini file contains a line reading

**LOGSEARCH=**

2. Place a **y** after the equals sign to turn search logging on; or an **n** after the equals sign to turn search logging off.

At the time of installation, search logging is turned off.

Optionally, you can set the identifier string for computers that will be running OPAC. This string is in the `opac.ini` file of each WebVoyage display directory (`vcit`, `zcit`, `z3950`, and so on) and its format is as follows:

**StatString=**

Any text after the equals sign will be recorded in the `OPAC_search_log` table or `bib_usage_log` table as the requesting OPAC clients's ID number. It may be a maximum of ten characters (including spaces). If left blank, the field will display as `WebOpac`.

The string may be unique for each WebVoyage display directory containing its own `opac.ini` file (`vcit`, `zcit`, `z3950`, etc...), or you can assign the same string to any number of display directories.

If OPAC is running on terminals from a server (WebVoyage), all terminals connecting to that server will return the same ID string.

In addition, regardless of whether the `StatString=` feature is in use, the IP address of each requesting OPAC client will be recorded in the `Client_ip` field of the `OPAC_search_log` table.

## **Setting Up Bibliographic Usage Logging**

To set up bibliographic usage logging, you must do the following:

Turn the bibliographic usage logging feature on.

In the `/ini` directory on the server (`/m1/voyager/xxxdb`), the `voyager.ini` file contains the following variable:

**LOGBIBUSAGE=**

Place a `Y` after the equals sign to turn bibliographic usage logging on; or an `N` after the equals sign to turn bibliographic usage logging off.

At the time of installation, bibliographic usage logging is turned off.

Optionally, you can set the identifier string for computers that will be running OPAC. This string is in the `opac.ini` file of each WebVoyage display directory (`vcit`, `zcit`, `z3950`, etc...) and its format is as follows:

**StatString=**

Any text after the equals sign will be recorded in the OPAC\_search\_log table or bib\_usage\_log table as the requesting OPAC clients's ID number. It may be a maximum of ten characters (including spaces). If left blank, the field will display as WebOpac.

The example in [Figure 15-1](#) shows information in the `opsrchlgexport.log` file from WebVoyage sessions on a database using SDirect as its StatString.

---

```
W,2002.04.03.11.20.45,,,20020403112025,SDirect,10.200.1.126,50887097,D,  
W,2002.04.03.11.20.46,,,20020403112025,SDirect,10.200.1.126,50887097,D,  
W,2002.04.03.11.20.52,,,20020403112025,SDirect,10.200.1.126,50465107,D,  
W,2002.04.03.11.20.52,,,20020403112025,SDirect,10.200.1.126,50465107,D,  
W,2002.04.03.11.20.57,,,20020403112025,SDirect,10.200.1.126,50465107,S,  
W,2002.04.03.11.20.57,,,20020403112025,SDirect,10.200.1.126,50465107,D,
```

**Figure 15-1. Sample StatString Results**

The string may be unique for each WebVoyage display directory containing its own `opac.ini` file (`vcit`, `zcit`, `z3950`, etc...), or you can assign the same string to any number of display directories.

If OPAC is running on terminals from a server (WebVoyage), all terminals connecting to that server will return the same ID string.

In addition, regardless of whether the `StatString=` feature is in use, the IP address of each requesting OPAC client will be recorded in the `Client_ip` field of the `bib_usage_log` table.

## **Setting up Selective Dissemination of Information (SDI)**

---

For information on enabling and configuring SDI, see [SDI Searches Program](#) on [page 15-9](#).

## **Accessing Logged Information**

---

### **OPAC Search Log Export program**

---

Included with Voyager is the `Popacjob` program, which is comprised of three parts. The first, OPAC Log Export (formerly `Popaclogexp`), pulls the specified information from the `OPAC_search_log` table in the database into a comma-delimited text file. The information in the comma-delimited text file is presented in [Table 15-1](#) on [page 15-2](#).

You can run the OPAC Search Log Export and Opac Bib Usage Log Export programs as separate jobs to access both search logging information and bibliographic usage information.

OPAC Search Log Export may be run interactively, or set to run automatically as a cron. Also, for information about running the job in WebAdmin see [Popacjob](#) on [page 30-34](#).

### Interactive Method

Use the following steps to run OPAC Search Log Export interactively:

Switch to the /sbin directory (/m1/voyager/xxxdb)

Type **Popac job**.

Press **Enter**.

A menu containing the following four options displays (see [Figure 15-2](#)):

1 -- Opac Search Log Export

2 -- SDI Searches

3 -- Opac Bib Usage Log Export

99 -- Quit

At the Process Job # prompt, type 1.

Press **Enter**.

---



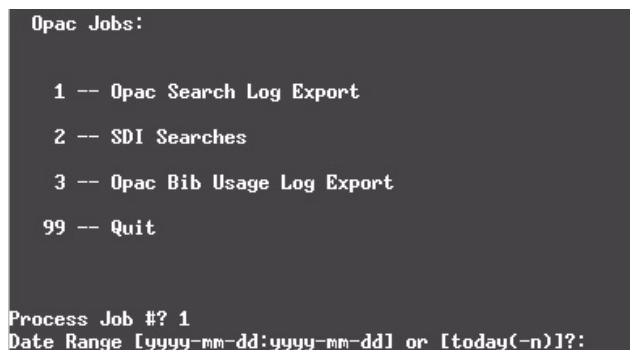
---

**Figure 15-2. OPAC Job Options Menu**

The program will prompt you for a date range (see [Figure 15-3](#)). Enter the desired date range and press **Enter**.

The range formats, *yyyy-mm-dd:yyyy-mm-dd* or *today-n* allow you to do one of the following:

- *yyyy-mm-dd:yyyy-mm-dd* format: Specify dates from which Voyager will extract entries from the applicable table in the database (for example, 1997-01-01:1998-03-15 will extract every entry from 1/1/97 through 3/15/98).
- *today-n* format: Specify the number of days before the current date for which Voyager will extract entries (for example, today-7 will extract entries for the current date and the 7 days previous). Excluding the *-n* switch will set the date to the current day.



**Figure 15-3. OPAC Search Log Export Job Date Range Prompt**

**NOTE:**

When running this job from the menu, the *-p* parameter, that is available when running the job as a cron or from the command line, is not available.

### Cron Method

You can run OPAC Search Log Export as a cron, using the following options:

- j      Specifies which Popacjob program to run.
  - 1 will run OPAC Search Log Export
  - 2 will run SDI Searches
  - 3 will run OPAC Bib Log Export

Set this to 1 to run OPAC Search Log Export.

-o<filename>Creates an output file with the specified filename. The default filename is opsrchlgexport.dat.

The range formats, -r<yyyy-mm-dd:yyyy-mm-dd> OR -r<today-n>allow you to do one of the following:

- -r<yyyy-mm-dd:yyyy-mm-dd> format: Specify dates from which Voyager will extract entries from the applicable table in the database (for example, -r1997-01-01:1998-03-15 will extract every entry from 1/1/97 through 3/15/98).
- -r<today-n> format: Specify the number of days before the current date for which Voyager will extract entries (for example, -rtoday-7 will extract entries for the current date and the 7 days previous). Excluding the -n switch will set the date to the current day.

-p      Purges the applicable table in the database of the specified entries after saving them to the export file. (Recommended)

-q      Quiet mode - will not prompt for any input.

-v      Provides version information.

-h      Provides help for the program.

After running the program:

Data extracted from the OPAC\_search\_log table is placed in the /rpt directory in the file opsrchlgexport.dat or in the file specified after the -o parameter. The log file for this operation is named opsrchlgexport.log or uses the filename specified after the -o parameter with the file extension of .log. A file in the /log directory called log.voyager gets updated with a record of the extraction operation.

## **SDI Searches Program**

---

Included with Voyager is the Popac job program, which is comprised of three parts. The second, SDI Searches, runs search queries stored by patrons at the appropriate intervals.

Saved Searches must be enabled to use SDI. SDI allows patrons to choose intervals for each saved search query to be run automatically (on the local database) and the results e-mailed to them in the form of a URL. SDI Searches may be run interactively, or set to run automatically as a cron.

Your library must configure SDI in WebVoyage (including the sdiemail.ini file) before running the SDI Searches program. For more information on configuring SDI options in WebVoyage, including the sdiemail.ini file, see the *Voyager WebVoyage User's Guide*.

## Enabling SDI

You enable/disable SDI in the SDIOption variable of the [SDI\_Page] stanza in the opac.ini file, by setting the variable to Y. The following is the default SDIOption variable:

```
SDIOption=N
```

The SDIOption= variable must be set to **Y** or **N**:

Setting this variable to **Y** enables SDI.

Setting this variable to **N** disables SDI.

## Configuring SDI Options

[Figure 15-4](#) contains an example of the [SDI\_Page] stanza. [Table 15-3](#) details the functions and descriptions of each variable in the stanza

```
[SDI_Page]
SDIOption=N
RunSearchEvery=Run Search Every:
GetNewHits=Y
NewHits>New Hits
None=None
Daily=Daily
Weekly=Weekly
Bi-Weekly=Bi-Weekly
Monthly=Monthly
```

**Figure 15-4.** The Default [SDI\_Page] Stanza of the opac.ini File.

**Table 15-3.** Components of the [SDI\_Page] Stanza

Variable Name	Description	Possible Values	Default Value
SDIOption=	Switches SDI on and off	Y or N	N
RunSearchEvery=	Heading of column listing options for SDI/saved search intervals	Free text	Run Search Every

**Table 15-3. Components of the [SDI\_Page] Stanza**

<b>Variable Name</b>	<b>Description</b>	<b>Possible Values</b>	<b>Default Value</b>
GetNewHits=	Switches the E-mail search results only if new search result information exists on and off.	Y or N  Y enables option to send results of query to patron only if new search result information exists.  N Disables option to send results of query to patron only if new search result information exists (results are sent by SDI whenever saved searches are performed).	Y
NewHits=	Heading of column containing the Y/N (yes/no) drop-down to select/deselect E-mail search results only if new search result information exists option. Only available if GetNewHits= above is set to Y.	Free text	New Hits?
None=	Text displayed on Saved Search Query screen when patron has no search queries saved	Free text	None
Daily=	Text in drop-down list box of the Run Search Every column corresponding to the E-mail results every day option.	Free text	Daily
Weekly=	Text in drop-down list box of the Run Search Every column corresponding to the E-mail results weekly option.	Free text	Weekly

**Table 15-3. Components of the [SDI\_Page] Stanza**

Variable Name	Description	Possible Values	Default Value
Bi-Weekly=	Text in drop-down list box of the Run Search Every column corresponding to the E-mail results bi-weekly option.	Free text	Bi-Weekly
Monthly=	Text in drop-down list box of the Run Search Every column corresponding to the E-mail results Monthly option.	Free text	Monthly

### Customizing the Search Results URL Page Sent to Patrons by SDI

You must customize the layout of the page(s) containing the URLs sent to patrons by SDI. This is done in the `sdiemail.ini` file. This file is in the `/m1/voyager/xxxdb/ini` directory.

#### Components of the `sdiemail.ini` File

This file is in the `/m1/voyager/xxxdb/ini` directory (see [Figure 15-5](#) for the default). Customize the sections of this file according to your preferences. [Figure 15-6](#) shows an e-mail generated by SDI.

Make sure that the `Address=` entry in the `[WebVoyage_Server]` stanza contains the correct address of the server webopac is running on. (For example: `Address=10.111.111.96:7008` or `Address=www.seekandfind.edu`).

Any fields in the `[Email]` stanza left blank will yield the default text.

#### NOTE:

The 80 character maximum length per line restriction includes the variable name and = sign.

The footer for this e-mail uses the `emailf.cfg` file (`/m1/xxxdb/etc/webvoyage/local`).

```
[Email]
Subject=Search Updates Results
Body1=The following URL contains the results of your automatic
      Search
Body2=Updates. Clicking the link will take you directly to the
Body3=Webvoyage titles index.
SearchString=Search for:
SearchType=Search Type:

[Webvoyage_Server]
Address=XXXX
```

Figure 15-5. Default sdiemail.ini File

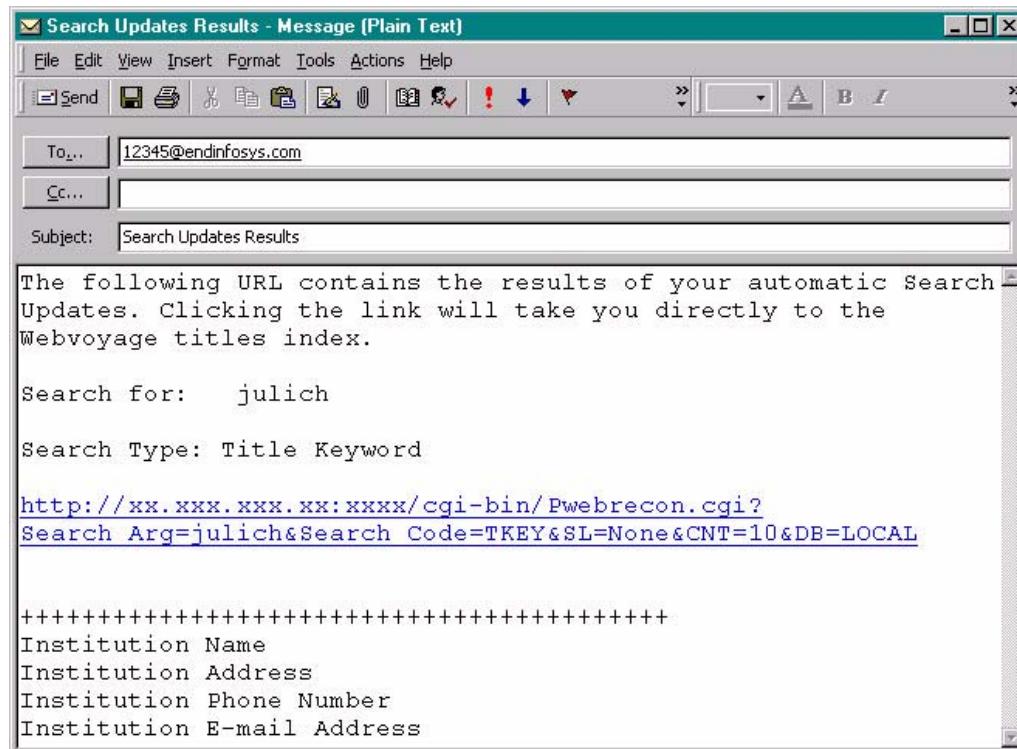


Figure 15-6. Sample SDI URL in email

## Interactive Method

Use the following steps to run SDI Searches interactively:

Switch to the /sbin directory (/m1/voyager/xxxdb)

Type **Popac job**.

Press **Enter**.

A menu containing the following four options displays (see [Figure 15-7](#)):

1 -- Opac Search Log Export

2 -- SDI Searches

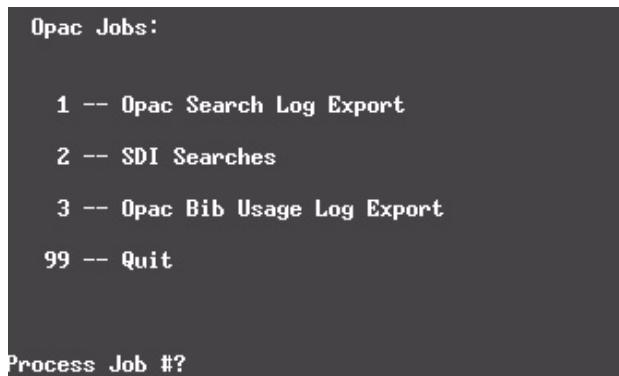
3 -- Opac Bib Usage Log Export

99 -- Quit

At the Process Job # prompt, type 2.

Press **Enter**.

---



---

**Figure 15-7. OPAC Job Options Menu**

SDI Searches will now determine which saved search queries to run, and email the results to patrons.

**NOTE:**

Make sure to set the cron to run at least as frequently as the shortest available interval in the New Hits Every drop-down box in the SDI options in WebVoyage. For example, if you give patrons the option of running SDI every day, make sure that the cron runs every day.

For more information on configuring SDI intervals for patron selection, see the Daily, Weekly, Bi-Weekly, and Monthly variables in the *Voyager WebVoyage User's Guide*.

### Cron Method

You can run SDI Searches as a cron, using the following options:

- j      Specifies which Popacjob program to run.
  - 1 will run OPAC Search Log Export
  - 2 will run SDI Searches
  - 3 will run OPAC Bib Usage Log Export

Set this to **2** to run SDI Searches.

After running the program, the log file (`opacjob.log`) can be found in the `/rpt` directory on your server (`/m1/voyager/xxxdb/rpt`).

### OPAC Bib Usage Log Export Program

---

Included with Voyager is the `Popacjob` program, which is comprised of three parts. The third, OPAC Bib Usage Log Export (formerly performed using the `-b` switch when running `Popaclogexp`), pulls the specified information from the `bib_usage_table` in the database into a comma-delimited text file. The information in the comma-delimited text file is presented in [Table 15-2](#) on [page 15-3](#).

You can run the OPAC Search Log Export and Opac Bib Usage Log Export programs as separate jobs to access both search logging information and bibliographic usage information.

OPAC Bib Usage Log Export may be run interactively, or set to run automatically as a cron.

### Interactive Method

Use the following steps to run OPAC Bib Usage Log Export interactively:

Switch to the `/sbin` directory (`/m1/voyager/xxxdb`)

Type `Popacjob`.

Press **Enter**.

A menu containing the following four options displays (see [Figure 15-8](#)):

- 1 -- Opac Search Log Export
- 2 -- SDI Searches

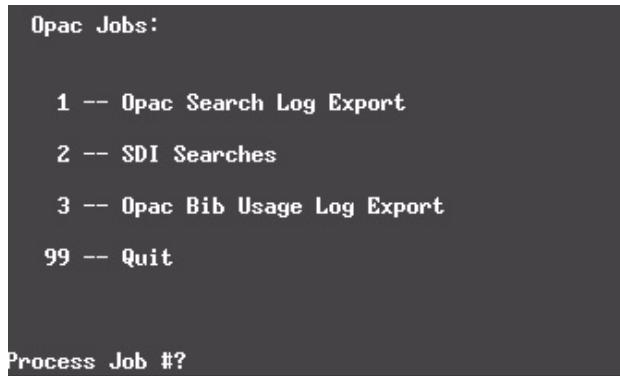
3 -- Opac Bib Usage Log Export

99 -- Quit

At the Process Job # prompt, type 3.

Press **Enter**.

---

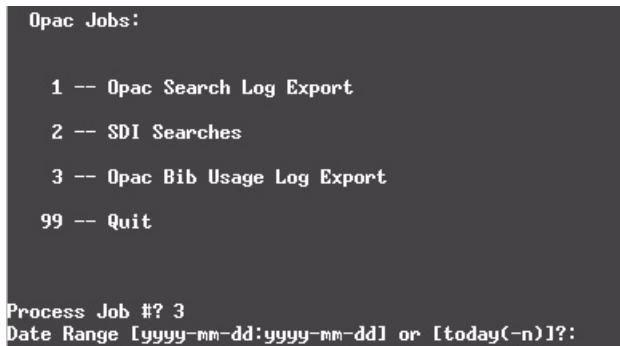


**Figure 15-8. OPAC Job Options Menu**

The program will prompt you for a date range (see [Figure 15-9](#)). Enter the desired date range and press **Enter**.

The range formats, *yyyy-mm-dd:yyyy-mm-dd* or *today-n* allow you to do one of the following:

- *yyyy-mm-dd:yyyy-mm-dd* format: Specify dates from which Voyager will extract entries from the applicable table in the database (for example, 1997-01-01:1998-03-15 will extract every entry from 1/1/97 through 3/15/98).
- *today-n* format: Specify the number of days before the current date for which Voyager will extract entries (for example, today-7 will extract entries for the current date and the 7 days previous). Excluding the *-n* switch will set the date to the current day.



---

Figure 15-9. OPAC Bib Usage Log Export Job Date Range Prompt

### Cron Method

You can run OPAC Log Export as a cron, using the following options:

- j      Specifies which Popacjob program to run.
  - 1 will run OPAC Search Log Export
  - 2 will run SDI Searches
  - 3 will run OPAC Bib Usage Log Export

Set this to 3 to run OPAC Bib Usage Log Export.

-o<filename>Creates an output file with the specified filename. The default filename is bibuselgexport.dat

The range formats, -r<yyyy-mm-dd:yyyy-mm-dd> OR -r<today-n>allow you to do one of the following:

- -r<yyyy-mm-dd:yyyy-mm-dd> format: Specify dates from which Voyager will extract entries from the applicable table in the database (for example, -r1997-01-01:1998-03-15 will extract every entry from 1/1/97 through 3/15/98).
- -r<today-n> format: Specify the number of days before the current date for which Voyager will extract entries (for example, -rtoday-7 will extract entries for the current date and the 7 days previous). Excluding the -n switch will set the date to the current day.

-p      Purges the applicable table in the database of the specified entries after saving them to the export file. (Recommended)

-q      Quiet mode - will not prompt for any input.

-v Provides version information.

-h Provides help for the program.

After running the program:

Data extracted from the bib\_use\_log table is placed in the /rpt directory in the file bibuselgexport.dat or in the file specified after the -o parameter. The log file for this operation is named bibuselgexport.log or uses the filename specified after the -o parameter with the file extension of .log. A file in the /log directory called log.voyager gets updated with a record of the extraction operation.

---

## **Acquisitions Batch Job - Fix Exchange Rates**

---

# **16**

<b>Introduction</b>	<b>16-1</b>
<b>Fix Exchange Rates</b>	<b>16-1</b>
• Purchase Orders Updated with Acqjob 5	16-2
• Additional Considerations for Acqjob 5	16-3
<b>Running Acqjob 5</b>	<b>16-3</b>
• Running Acqjob 5 from the Command Line	16-4
• Running Acqjob 5 Interactively	16-4
<b>Log File</b>	<b>16-6</b>



---

## Acquisitions Batch Job - Fix Exchange Rates

# 16

---

### Introduction

---

Acquisitions batch job 5, Fix Exchange Rates (formerly Pfixexchangerates script), updates the commitments (encumbrances) on selected Purchase Orders based on the currency conversion rates in the system at the time this job is run. Commitments are monies that are subtracted from the available fund total when an order is approved, but before it has been paid for. This job updates the commitments if the currency conversion rates have changed since the PO was approved.

The fund's commitments, total and pending, are updated as well.

Sites using foreign currencies, where conversion rates vary, may want to use this feature.



#### **IMPORTANT:**

*This job is not reversible. Once run, the commitments are adjusted permanently.*

### Fix Exchange Rates

---

The Fix Exchange Rates job updates commitments on Purchase Orders that use foreign currencies, providing a more accurate picture of commitments as conversion (exchange) rates fluctuate.

## **Purchase Orders Updated with Acqjob 5**

---

When the batch job is run, the system considers the following criteria when determining which POs are adjusted by this job:

- The Purchase Orders must be in a currency other than the base currency.
- The Purchase Orders must have a status of Received Partial or Approved/Sent.

For each PO that meets the above criteria, the system adjusts all outstanding commitments, omitting line-item copies that are covered by prepay or have already been invoiced.

A currency rate fluctuation may cause funds to be over committed. If this occurs, the details are reported in the log file. Also, this information is available in the various fund workspaces of the Acquisitions module.

**NOTE:**

Purchase Orders with a status of Pending, Received Complete, Complete, and Canceled are not considered for adjustment by this job. If a PO has a status of Pending, the system automatically adjusts currency rates when the PO is opened in the Acquisitions client.

## Additional Considerations for Acqjob 5

The user can limit the POs considered for adjustment by specifying any of the following criteria on the command line: Currency Codes, Ledger Names, Order Types, and Vendor Codes.

[Table 16-1](#) lists the optional parameters that the user can specify individually or in tandem with the Pacqjob -j5 batch job.

**Table 16-1. Acqjob 5 Batch Job Parameters**

Parameter	Description
-c<currency_codes>	This optional parameter limits the adjustments of POs to specific currency codes.  Specify a single currency code or a comma-separated list of currency codes in the following format:  -c "EUR, CND"
-m<fund_codes>	This optional parameter limits the adjustments of POs to specific ledgers.  Specify a single fund code or a comma-separated list of fund codes in the following format:  -m "FY05, Fiscal2004"
-p<vendor_codes>	This optional parameter limits the adjustments of POs to specific vendors.  Specify a single vendor code or a comma-separated list of vendor codes in the following format:  -p "EBSCO, B&TA"
-t<order_types>	This optional parameter limits the adjustments of POs to specific Order Types.  Specify a single order type or a comma-separated list of order types in the following format:  -t "Approval, FirmOrder"

## Running Acqjob 5

This job can be run in the following ways:

- Entering full command line on the server,
- Using the command line interactively on the server.

- Using WebAdmin, see [Acquisitions Utilities](#) on [page 30-9](#)
- Adding the job to a cron on the server.

## **Running Acqjob 5 from the Command Line**

---

To run this job from the command line, enter the command along with required and optional parameters from the `/m1/voyager/xxxdb/sbin>` directory. For example, the following command runs the adjustment for all Approved/Received Partial POs that use a foreign currency:

**Pacqjob -j5**

If the job is accepted, the screen displays a message, `FixExchangeRates` is running. When the job is finished, the `COMPLETED` message displays, followed by the command-line prompt.

## **Running Acqjob 5 Interactively**

---

To run this job interactively at the `/m1/voyager/xxxdb/sbin>` prompt, enter the following batch job command without parameters:

**Pacqjob**

After the system prompts you to select the batch job (see [Figure 16-1](#)), enter:

```
/m1/voyager/xxxdb/sbin $ Pacqjob
Initializing...
 1 -- Order Claim/Cancel Processing
 2 -- Open Order Report
 3 -- Fund Snapshot Report -- Global Distribution
 4 -- Rollover Status Report -- Global Distribution
 5 -- Fix Exchange Rate
 99 -- quit

Process Job #?           5
FixExchangeRate is running...
Processing begins...
(*) Finished processing 8 purchase orders.
(*) 1 successfully reconciled; 0 failed.
...COMPLETED
/m1/voyager/xxxdb/sbin $
```

---

**Figure 16-1. Screen Display for Acqjob 5 (Fix Exchange Rates)**

**NOTE:**

The optional parameters that are listed in [Table 16-1](#) on [page 16-3](#) cannot be used in interactive mode.

The system executes the job which has been configured with your database name, username, and password. The screen displays the number of POs considered for update, the number of PO updated, and the number of POs that failed. When completed, the system displays the COMPLETED message.

**NOTE:**

If you want to run a server activity in the background and you do not want to perform it interactively, enter

```
nohup <activity with parms> &
```

to make certain that the activity continues, even if the telnet session is lost, where <activity with parms> is the command followed by any required and optional parameters. This command will not guarantee that the activity will be completed if the server itself is not available.

## Log File

---

Each time the `Pacqjob -j5` batch job is run, it creates a log file named, `log.foreigncommitments.yyyyymmdd.hhmm`, where `yyyy` is year, `mm` is month, `dd` is day, `hh` is hour, and `mm` is minutes. It is stored in the `/m1/voyager/xxxdb/rpt` directory as a text file.

This log ([Figure 16-2](#)) can be examined to see changes in individual orders to monitor the fluctuation of currency rates. The fund records in the Acquisitions module can be used to see the current total commitments.

The log includes the following information.

- PO number of PO considered for update
- PO Header information for a PO that is updated
  - PO Number
  - Currency code
  - PO type
  - Status
  - Vendor name
- Line item information
  - Fund
  - Price in foreign currency
  - Conversion rate before adjustment
  - Commitment amount in the base currency before adjustment
  - Conversion Rate after adjustment
  - Commitment amount in the base currency after the adjustment.

**NOTE:**

If the new commitment causes the fund's total commitments to exceed its limit, this amount is annotated "OVERCOMMIT".

```
PO_ID# 994

000994 0000 000000 0 -- -----
000994 0000 000000 1 PO number: 994          PO total: GBP 10.00
000994 0000 000000 2   Type: Firm order      Status: Approved/Sent
000994 0000 000000 3   Vendor: Baker & Taylor| 

000994 0098 000605 4 * Fund: 04 Test Ledger/04 Test Fund
000994 0098 000605 5   Before @ 0.89000      $ 11.24

000994 0098 000605 4 * Fund: 04 Test Ledger/04 Test Fund
000994 0098 000605 6   After @ 0.75000       $ 13.33  *OVERCOMMIT*

PO_ID# 1000

PO_ID# 1001

PO_ID# 1002

PO_ID# 1004
```

---

**Figure 16-2. Foreign Commitments Log File**

The acqjob.log file lists the date and time the job began and ended (see [Figure 16-3](#)).

---

```
Thu Sep 29 15:18:33 2005 FixExchangeRate is running...
Thu Sep 29 15:18:33 2005 ...COMPLETED
```

---

**Figure 16-3. Acqjob.log file**



<b>Introduction</b>	<b>18-1</b>
<b>UB Barcode Lookup</b>	<b>18-1</b>
• Configuring the UB Barcode Lookup Feature	18-1
voyager.env	18-2
Creating the UB Barcode Lookup Configuration File	18-2
UB Barcode Lookup Configuration Schema File	18-8
• Configuration File Validation	18-11
<b>Dynamic Noise Word Reduction</b>	<b>18-12</b>
<b>Configuring the Z39.50 Server for UTF-8 Encoded Records</b>	<b>18-13</b>



### **Introduction**

---

This chapter describes the set up information for features that require configuration changes on the server.

### **UB Barcode Lookup**

---

The UB Barcode Lookup feature enables the Circulation client to automatically determine the Circulation cluster of patrons and items in Universal Borrowing (UB) or multi-cluster environments during charges, discharges, and patron searches - reducing the effort required by circulation clerks to determine the Circulation cluster of the patron or item.

Configuration files are used to enable the feature and to define the barcode patterns that the system uses to determine which patron/item barcodes are associated with a particular cluster or database within a UB or multi-cluster environment.

### **Configuring the UB Barcode Lookup Feature**

---

The following files are used to configure the UB Barcode Lookup feature:

- `voyager.env` file
- UB Barcode Lookup configuration file

- UB Barcode Lookup schema file

### voyager.env

The UB\_BARCODE\_CONFIG environment variable enables the auto-selection of local clusters for patron and item barcodes during charges, discharges, and patron searches. By default, the UB\_BARCODE\_CONFIG environment variable appears as follows in the /m1/voyager/xxxdb/ini/voyager.env file:

```
#export UB_BARCODE_CONFIG="$VOYAGER/$DATABASE/ini/  
ub_barcode_config.xml"
```

To enable auto-selection of barcodes, uncomment the line and modify the path and file name if different from the default. If the configuration file has not been created, the feature will not become active until the configuration file is created and the Circulation module is restarted.

### Creating the UB Barcode Lookup Configuration File

The UB Barcode Lookup Configuration file contains the cluster, database, and barcode patterns that the system uses to determine the local cluster of a patron or item barcode during charges, discharges, and patron searches from the Circulation client.

It is recommended that the configuration file be created and maintained by the consortium administrator. A copy of this file must be installed and enabled via the *voyager.env* file for each database in the consortium.

Before creating the configuration file, verify that all UB and cluster data is configured properly in the database tables (**Voyager\_Databases** and **Remote\_Circ\_Cluster\_Cache**) since they will be used to create a skeleton configuration file (see [Figure 17-1](#) for an example).



### Procedure 17-1. Creating the UB Barcode Lookup Configuration File

To create the UB Barcode Lookup Configuration file, perform the following steps:

1. Enter the following command from the /m1/voyager/xxxdb/sbin directory:

```
. /PgenUBBarcodeConfig
```

Result: If the `ub_barcode_config.xml` file already exists in the `/m1/voyager/xxxdb/ini` directory, you will be asked to create a new file. Otherwise, a skeleton configuration file is created (see [Figure 17-1](#)).

2. Open the file you created in Step [1](#).
3. Check to see if any database or cluster names contain ampersands. If any contain ampersands, change each ampersand to an “and” to prevent syntax errors.
4. For each `<Patron></Patron>` block and `<Item></Item>` block, enter a list of barcode patterns (regular expressions) that the system will use to distinguish the Circulation cluster of patron or item barcode. See [Table 17-1](#) on [page 17-5](#) for more information on the `<BarcodePattern>` field.

**NOTE:**

If the same pattern is used for more than one cluster, the system will return multiple matches while using this feature in the Circulation module.

---

[Figure 17-1](#) shows a sample configuration file that has been created using the `PgenBarcodeConfig` command. Note that the `<Patron></Patron>` and `<Item></Item>` blocks are empty and must be filled in manually. See [Table 17-1](#) on [page 17-5](#) for more information.

---

```
<?xml version="1.0" encoding="UTF-8"?>
<Consortium
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="schema/ub_barcode_config.xsd">
    <Flag>
        <TryLocalClusterFirst>true</TryLocalClusterFirst>
        <UseFirstMatchingPatronCluster>false</
    UseFirstMatchingPatronCluster>
        <UseFirstMatchingItemCluster>false</
    UseFirstMatchingItemCluster>
    </Flag>
    <Cluster>
        <DatabaseName>QAMK UB/Cluster Database</DatabaseName>
        <DatabaseKey>QA20050DB20020613131313</DatabaseKey>
        <ClusterName>QAMK Cluster 1</ClusterName>
        <ClusterId>1</ClusterId>
        <Patron>
        </Patron>
        <Item>
        </Item>
    </Cluster>
```

```
<Cluster>
  <DatabaseName>QAMK UB/Cluster Database</DatabaseName>
  <DatabaseKey>QA20050DB20020613131313</DatabaseKey>
  <ClusterName>QAMK Cluster 2</ClusterName>
  <ClusterId>2</ClusterId>
  <Patron>
    </Patron>
    <Item>
      </Item>
    </Cluster>
  </Consortium>
```

---

**Figure 17-1. Skeleton UB Configuration File**

[Table 17-1](#) describes the fields used in the UB Barcode Lookup Configuration file.

**Table 17-1. UB Barcode Lookup Configuration File Commands**

Tag Name	Description
<Flag>	<p>This field is used to define initial criteria that is considered by the system to determine the local cluster of a patron or item. It contains the following subfields:</p> <ul style="list-style-type: none"> <li>• &lt;TryLocalClusterFirst&gt;</li> <li>• &lt;UseFirstMatchingPatronCluster&gt;</li> <li>• &lt;UseFirstMatchingItemCluster&gt;</li> </ul> <p><b>XPath:</b> Consortium/Flag</p>
<TryLocalCluster-First>	<p>If set to <code>true</code>, this flag indicates that the system will search the local cluster for a match first. If no match is found, the system compares the patron or item barcode to the barcode patterns specified in the configuration file to determine the cluster. If a matching pattern is found, the system uses the cluster that contains the matching pattern; otherwise, the system prompts the user to select a cluster from a drop-down menu.</p> <p>If set to <code>false</code>, the system uses the &lt;UseFirstMatchingPatronCluster&gt; or &lt;UseFirstMatchingItemCluster&gt; flags to find the local cluster of the patron/item.</p> <p><b>NOTE:</b> This flag has precedence over the other flags.</p> <p><b>Default:</b> <code>true</code></p> <p><b>XPath:</b> Consortium/Flag/TryLocalClusterFirst</p>
<UseFirstMatching-PatronCluster>	<p>If set to <code>true</code>, this flag indicates that the system compares the patron barcode with the patron barcode patterns in the configuration file. If a match is found, the system tries the first matching cluster.</p> <p>If set to <code>false</code>, the system compares the patron barcode with the patron barcode patterns in the configuration file. If multiple matches are found, the system prompts the user to select a cluster from a drop-down menu.</p> <p><b>NOTE:</b> The &lt;TryLocalClusterFirst&gt; flag has precedence over this flag.</p> <p><b>Default:</b> <code>false</code></p> <p><b>XPath:</b> Consortium/Flag/UseFirstMatchingPatronCluster</p>

**Table 17-1. UB Barcode Lookup Configuration File Commands**

Tag Name	Description
<UseFirstMatching-ItemCluster>	<p>If set to <code>true</code>, this flag indicates that the system compares the item barcode with the item barcode patterns in the configuration file. If a match is found, the system tries the first matching cluster.</p> <p>If set to <code>false</code>, the system compares the item barcode with the item barcode patterns in the configuration file. If multiple matches are found, the system prompts the user to select a cluster from a drop-down menu.</p> <p><b>NOTE:</b> The <code>&lt;TryLocalClusterFirst&gt;</code> flag has precedence over this flag.</p> <p><b>Default:</b> <code>false</code></p> <p><b>XPath:</b> Consortium/Flag/UseFirstMatchingItemCluster</p>
<Cluster>	<p>This field is used to define the barcode patterns for each cluster in the UB or multi-cluster environment. It contains the following sub-fields:</p> <ul style="list-style-type: none"> <li>• &lt;DatabaseName&gt;</li> <li>• &lt;DatabaseKey&gt;</li> <li>• &lt;ClusterName&gt;</li> <li>• &lt;ClusterID&gt;</li> <li>• &lt;Patron&gt;</li> <li>• &lt;Item&gt;</li> </ul> <p><b>NOTE:</b> The system automatically populates the cluster field for each cluster defined in the UB or multi-cluster environment. The consortium administrator will need to add the barcode patterns manually.</p> <p><b>XPath:</b> Consortium/Cluster</p>
<DatabaseName>	<p>For each cluster, the system automatically populates the database name during creation of the configuration file.</p> <p><b>Data Source:</b> VOYAGER_DATABASES.db_name</p> <p><b>XPath:</b> Consortium/Cluster/DatabaseName</p>

**Table 17-1. UB Barcode Lookup Configuration File Commands**

<b>Tag Name</b>	<b>Description</b>
<DatabaseKey>	For each cluster, the system automatically populates the database key during creation of the configuration file.  <b>Data Source:</b> VOYAGER_DATABASES.db_key  <b>XPath:</b> Consortium/Cluster/DatabaseKey
<ClusterName>	For each cluster, the system automatically populates the cluster name during creation of the configuration file.  <b>Data Source:</b> REMOTE_CIRC_CLUSTER_CACHE.cluster_name  <b>XPath:</b> Consortium/Cluster/ClusterName
<ClusterID>	For each cluster, the system automatically populates the cluster ID during creation of the configuration file.  <b>Data Source:</b> REMOTE_CIRC_CLUSTER_CACHE.remote_circ_cluster_id  <b>XPath:</b> Consortium/Cluster/ClusterID
<Patron>	For each cluster, the system automatically populates an empty <Patron> </Patron> block, which is used to manually define the barcode patterns that the system uses to determine the local cluster of the patron barcode.  <b>XPath:</b> Consortium/Cluster/Patron
<BarcodePattern> (patron)	A list of regular expressions that represent the barcode patterns for each database/cluster.  <pre> &lt;Patron&gt;     &lt;BarcodePattern&gt;66.*&lt;/BarcodePattern&gt;     &lt;BarcodePattern&gt;10301.*&lt;/BarcodePattern&gt; &lt;/Patron&gt; </pre> <b>XPath:</b> Consortium/Cluster/Patron/BarcodePattern
<Item>	For each cluster, the system automatically populates an empty <Item> </Item> block, which is used to manually define the barcode patterns that the system uses to determine the local cluster of the item barcode.  <b>XPath:</b> Consortium/Cluster/Item
<BarcodePattern> (item)	A list of regular expressions that represent the barcode patterns for each database/cluster.  <pre> &lt;Item&gt;     &lt;BarcodePattern&gt;CL.*&lt;/BarcodePattern&gt; &lt;/Item&gt; </pre> <b>XPath:</b> Consortium/Cluster/Item/BarcodePattern

## UB Barcode Lookup Configuration Schema File

If an XML editor is used to create or modify the UB Barcode Lookup Configuration file, the text (as shown in [Figure 17-2](#)) can be copied and pasted into a file (schema/ub\_barcode\_config.xsd) on the server to validate the fields in the configuration file as it is being edited.

---

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
>
  <xs:element name="Consortium">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Flag">
          <xs:complexType>
            <xs:sequence>
              <xs:element
                name="TryLocalClusterFirst"
                type="xs:boolean"
                default="true"
              />
              <xs:element name="UseFirstMatchingPatronCluster"
                type="xs:boolean"
                default="false"
              />
              <xs:element
                name="UseFirstMatchingItemCluster"
                type="xs:boolean"
                default="false"
              />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      <xs:element name="Cluster" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
```

---

**Figure 17-2. UB Configuration Schema File (Continued on next page)**

```
<xs:element  
    name="DatabaseName"  
    type="xs:string"  
    minOccurs="0"  
>  
    <xs:annotation>  
        <xs:documentation>  
            Populated from  
            VOYAGER_DATABASES.db_name  
        </xs:documentation>  
    </xs:annotation>  
</xs:element>  
<xs:element  
    name="DatabaseKey"  
    type="xs:string"  
>  
    <xs:annotation>  
        <xs:documentation>  
            Populated from  
            VOYAGER_DATABASES.db_key  
        </xs:documentation>  
    </xs:annotation>  
</xs:element>  
<xs:element  
    name="ClusterName"  
    type="xs:string" minOccurs="0"  
>  
    <xs:annotation>  
        <xs:documentation>  
            Populated from  
            REMOTE_CIRC_CLUSTER_CACHE.remote_circ_  
                cluster_name  
            or CIRC_CLUSTER.circ_cluster_name  
        </xs:documentation>  
    </xs:annotation>  
</xs:element>  
<xs:element  
    name="ClusterId"  
    type="xs:string"  
>  
    <xs:annotation>  
        <xs:documentation>  
            Populated from
```

---

**Figure 17-2. UB Configuration Schema File (Continued on next page)**

```
        REMOTE_CIRC_CLUSTER_CACHE.remote_
            circ_cluster_id
        or CIRC_CLUSTER.circ_cluster_id
    </xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="Patron" minOccurs="0">
    <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs=
            "unbounded">
            <xs:element ref="BarcodePattern"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Item" minOccurs="0">
    <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs=
            "unbounded">
            <xs:element ref="BarcodePattern"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:annotation>
    <xs:documentation>
        POSIX compatible regular expression
    </xs:documentation>
</xs:annotation>
<xs:complexType>
    <xs:simpleContent>
        <xs:extension base="xs:normalizedString"/>
    </xs:simpleContent>
</xs:complexType>
</xs:element>
</xs:schema>
```

---

**Figure 17-2. UB Configuration Schema File**

## Configuration File Validation

---

The format and consistency of the configuration file (`ub_barcode_config.xml`) is validated each time the Circulation module starts up and attempts to read the barcode patterns from the UB Barcode Lookup configuration file.

If the configuration file contains format errors, an entry is placed in the `/m1/voyager/xxxdb/log/log.circsvr` file. For example, [Figure 17-3](#) shows a log entry which indicates that an invalid value, `flase`, has been entered for a field on Line 6 of the configuration file.

---

```
circsvr[4694] - ERROR - Thu Oct 27 14:16:28 2005
  - LoadBarcodeTables - BarcodePatternTable.cpp[454]
  - ReqCirculationDefaults
  - vSession::route_request
  - vSession::Run
SAXParseException: Expecting true or false; got flase
System ID: /m1/voyager/xxxdb/ini/
ub_barcode_config.xml
    Public ID:
        Line: 6
    Column: 39
```

---

**Figure 17-3. UB Configuration Log File**

## Dynamic Noise Word Reduction

Dynamic noise word reduction improves the performance of keyword searches, particularly in large databases. It eliminates common words from a user's query for the purposes of searching but retains them for the purposes of relevance ranking.

In the server-side `voyager.ini` file, set `NOISEWORDFILTER=Y` to turn on this feature. The default is `NOISEWORDFILTER=N`.

The following is a list of stop words used with this feature.

- AND
- OR
- NOT
- OF
- IN
- THE
- WITH
- TO
- FOR

If the Dynamic Noise Word Reduction feature is enabled, the behavior of keyword searching depends on the boolean operator used in the search. [Table 17-2](#) shows the behavior of boolean searches when this feature is enabled.

**Table 17-2. NOISEWORDFILTER and Keyword Searches**

String	Search Code	Enabled?	Translation
mice and men	GKEY	N	(GKEY mice) OR (GKEY and) OR (GKEY men)
mice and men	GKEY	Y	(GKEY mice) OR (GKEY men) The "and" is stripped out of the search.
mice and men	GKEY^	N	(GKEY mice) AND (GKEY and) AND (GKEY men)
mice and men	GKEY^	Y	NOISEWORDFILTER is ignored.

## Configuring the Z39.50 Server for UTF-8 Encoded Records

By default, the Voyager Z39.50 Server sends MARC-8 encoded records to remote clients that request records via the z39.50 protocol.

With the Unicode release, UTF-8 encoded records can be sent also by configuring the Z39.50 server to handle UTF-8 and MARC-8 encoded records on separate ports. Traditionally, port `xxx90` has been used for MARC-8 encoded records, where `xx` represents the first two digits of your Voyager port. Ex Libris suggests that you use port `xxx91` for UTF-8 encoded records.



### Procedure 17-2. Configuring the Z39.50 Server to Send Both MARC-8 and UTF-8 Encoded Records

Use the following procedure to configure the Voyager Z39.50 Server to send both MARC-8 and UTF-8 encoded records.

1. Open the `/etc/services` file, locate the Z39.50 port assignment, and insert the following assignment to support UTF-8 encoded messages:

```
xxxdb/Pz3950svr    xx91    # Z39.50 Server UTF8
```

#### NOTE:

If port `xx91` is already being used, use another port, such as `xxx95`.

2. Exit and save the changes to the `/etc/services` file.
3. Copy the `/m1/voyager/xxxdb/ini/z3950svr.ini` file and save it as `z3950svrUTF8.ini` under the same directory.
4. Open the `/m1/voyager/xxxdb/ini/z3950svrUTF8.ini` file and perform the following steps:
  - a. Change the port assignment of the `ZPort` parameter to the port used in Step 1 as follows:

```
ZPort=xxx91
```

- b. Modify the `Encoding` parameter of the `[Default]` stanza to use the UTF-8 format as follows:

```
Encoding=A
```

- c. Exit and save the changes to the z3950svrUTF8.ini file.

Result: The modifications to the z3950svrUTF8.ini file appear as shown in [Figure 17-4](#).

---

```
# What TCP port number for the server?
ZPort=14091
DBname=VOYAGER
.
.
.

# Encoding
# This is either UTF8 or it defaults to Marc
Encoding=A
```

**Figure 17-4. Sample z3950svrUTF8.ini File Modifications**

5. Open the /m1/voyager/xxxdb/sbin/Pz3950svr script file and perform the following steps:
  - a. Insert lines of code into the file (as indicated by the bold text in [Figure 17-5](#)) to run a separate process for the UTF-8 encoded messages.

**NOTE:**

Use the .ini file created in Step 3 and specify a different log file for UTF-8 encoded transactions.

- b. Exit and save the changes to the Pz3950svr script file.
- 

```
if [ $OS = 'Windows_NT' ]; then
  exec $VOYAGER_BIN/z3950svr -i $VOYAGER/$DATABASE/ini/z3950svr.ini \
  -c $VOYAGER/$DATABASE/ini/voyager.ini \
  > $VOYAGER/$DATABASE/log/z3950svr.nohup.out &
else
  /bin/nohup $VOYAGER_BIN/z3950svr -i $VOYAGER/$DATABASE/ini/z3950svr.ini \
  -c $VOYAGER/$DATABASE/ini/voyager.ini \
  > $VOYAGER/$DATABASE/log/z3950svr.nohup.out &
  /bin/nohup $VOYAGER_BIN/z3950svr -i $VOYAGER/$DATABASE/ini/z3950svrUTF8.ini \
  -c $VOYAGER/$DATABASE/ini/voyager.ini \
  > $VOYAGER/$DATABASE/log/z3950svrUTF8.nohup.out &
fi
```

**Figure 17-5. Sample Pz3950svr Script File Modifications**

6. Stop and start the web server. For more information on stopping and starting the web server, refer to [Stopping and Starting the Apache Web Server](#) on [page 3-10](#) and [Stopping and Starting the Apache Web Server](#) on [page 3-20](#).
  7. Stop and start Voyager server. For more information on stopping and starting the Voyager, refer to [Stopping and Starting Voyager](#) on [page 3-8](#) and [Stopping and Starting Voyager](#) on [page 3-19](#).
-



---

**Patron Record Standard Interface  
File**

**18**

---

<b>Introduction</b>	<b>19-1</b>
<b>Purpose of This Chapter</b>	<b>19-1</b>
<b>File Specification</b>	<b>19-2</b>
<b>Patron Record SIF Format</b>	<b>19-2</b>
• Base Segment	19-3
• Address Segment	19-10
• Variable Segment	19-12
• End-of-Record Segment	19-13



---

## **Introduction**

---

The Patron Record Standard Interface File (SIF) is the format of a file that is used in three ways in Voyager

- Input file for the Patron load done at migration
- Input file for the Patron Update program
- Output of the Patron Extract program.

---

## **Purpose of This Chapter**

---

This section discusses the layout of the Patron Record SIF.

- File specification
- File format fixed segment
- File format address segment
- File format variable segment
- File format end-of-record segment

## File Specification

---

The data required for the Patron database records should be supplied in the following manner.

Each patron data record should be one continuous string, terminated by a line-feed (\n in some formats, hex '0A' in others).

The fields must be the indicated length and must all be present and a specific order and format. The notes segment is an exception to the length requirement.

The data fields which are designated numeric must be right-justified and zero-filled on the left.

The data fields which are designated strings (not designated numeric) should be left-justified and blank-filled on the right but not hex-null-terminated.

If the export data is in a continuous stream, you must insert a hex null terminator between records. However, if the export data is broken into one line per record, the null terminator does not need to be inserted.

## Patron Record SIF Format

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Offset** Actual starting position of the item in the record relative to 1.

**Format** Format of the variable.

**s** String, left-justified/blank filled.

**n** Numeric, right-justified/zero-filled.

**d** Date in the format ccyy.mm.dd, where c is the century, y is the year, m is the month, and d is the day. The maximum high value that can be used is 2382.12.31. This field may be blank under certain conditions.

**b** Blanks.

**Required** Y for yes, the field must have valid data in it (not just blanks).

**NOTE:**

The Social Security Number (SSAN) or the Institution ID is required for all records.

The maximum record should consist of the following:

1 base segment length:  $1 \times 456 = 456$

9 address segments length:  $9 \times 429 = 3861$

1 notes segment length:  $1 \times 1000 = 1000$

1 end-of-record mark length:  $1 \times 1 \text{ or } 2 = 1 \text{ or } 2$

The total maximum record length is 5318 or 5319.

The minimum record should consist of the following:

1 base segment length:  $1 \times 456 = 456$

1 address segment length:  $1 \times 429 = 429$

1 notes segment length:  $1 \times 1 = 1$

1 end-of-record mark length:  $1 \times 1 \text{ or } 2 = 1 \text{ or } 2$

The total minimum record length is 887 or 888.

## **Base Segment**

---

[Table 18-1](#) on [page 18-4](#) defines the base (fixed) segment of all records.

Also, a patron record must have one set, but can have up to three sets of the following fields: patron barcode, patron group, barcode status, and modification date. However, if one of the above fields exists in a set that has data, the other three fields must also have data. The exception to this is the patron barcode, which can be blank.

If more than one patron barcode is present, the associated patron groups must be unique.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
01	patron id	1	n		10	Internal record number for Voyager. This field should be zero-filled. The patron ID will be determined by the SSAN or Institution ID (whichever is unique) and the patron record will be updated if either is found, or added if not.
02	patron bar-code id 1	11	n*		10	ID of first library card barcode.
03	patron bar-code 1	21	s		25	First library card barcode. Footnote 1
04	patron group 1	46	s		10	Valid patron group as defined by library in system administration. Footnote 2
05	barcode status 1	56	n*		1	Valid: 1 is active. 2 is lost. 3 is stolen. 4 is expired. 5 is other.
06	barcode modified date 1	57	d		10	If blank, load run-date will be used.
07	patron bar-code id 2	67	n*		10	ID of second library card barcode.
08	patron bar-code 2	77	s		25	Second library card barcode. Footnote 1
09	patron group 2	102	s		10	Valid patron group as defined by library in the System Administration module.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
10	barcode status 2	112	n		1	Valid: 1 is active. 2 is lost. 3 is stolen. 4 is expired. 5 is other.
11	barcode modified date 2	113	d		10	If blank, load run-date will be used.
12	patron barcode id 3	123	n*		10	ID of third library card barcode.
13	patron barcode 3	133	s		25	Third library card barcode. Footnote 1
14	patron group 3	158	s		10	Valid patron group as defined by library in the System Administration module.
15	barcode status 3	168	n*		1	Valid: 1 is active. 2 is lost. 3 is stolen. 4 is expired. 5 is other.
16	barcode modified date 3	169	d		10	If blank, load run-date will be used.
17	registration date	179	d		10	The date the record was added to an external system such as administration or human resources.
18	patron expiration date	189	d	y	10	Date patron record is to be considered no longer active by the Voyager system. Standard date format.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
19	patron purge date	199	d	y	10	Date patron record is to be deleted by the Voyager system. Standard date format.
20	voyager date	209	b		10	The run-date of the load will be used if this is a new patron.
21	voyager updated	219	b		10	The run-date of the load will be used.
22	circulation happening location code	229	s		10	This is the location code for any circulation happening location in the Circulation Policy Group to which the patron belongs. When entering patrons manually, it is the location code of the desk where the patron was added.  It must be a valid location code.
23	institution ID	239	s	Footnote 3	30	Any value. If the ID came from the registrar or from human/resources and if it is important to know the source, the indicator should be part of this field (a prefix of <b>reg</b> or <b>h/r</b> ).
24	ssn	269	s	Footnote 3	11	Patron's social security account number: with or without dashes.
25	statistical category 1	280	s		3	Valid statistical category code as defined by library in the System Administration module. If this is an update to an existing patron record and there is an entry in this field, all existing statistical category records will be removed from the database and replaced with those found in this record.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
26	statistical category 2	283	s		3	See statistical category 1.
27	statistical category 3	286	s		3	See statistical category 1.
28	statistical category 4	289	s		3	See statistical category 1.
29	statistical category 5	292	s		3	See statistical category 1.
30	statistical category 6	295	s		3	See statistical category 1.
31	statistical category 7	298	s		3	See statistical category 1.
32	statistical category 8	301	s		3	See statistical category 1.
33	statistical category 9	304	s		3	See statistical category 1.
34	statistical category 10	307	s		3	See statistical category 1.
35	name type	310	n	y	1	Valid data: 1 is personal, the name is an individual. 2 is institutional, the name may be a department or organization.
36	surname	311	s	y	30	Patron's last name or institution name. Include any suffix (Jr. or PH.D.).
37	first name	341	s		20	Patron's first name. Required for name type 1.
38	middle name	361	s		20	Patron's middle name or initial.
39	title	381	s		10	Patron's title, if any.
	transaction counters	--	--	--	--	The next 10 fields are transaction counters.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
40	historical charges	391	n		10	Total number of charges for patron's entire history. Maximum value is 2147483648.
41	claims returned count	401	n		5	Total number of claims returned for patron's entire history. Maximum value is 32767.
42	self-shelved count	406	n		5	Total number of items self-shelved for patron's entire history. Maximum value is 32767.
43	lost items count	411	n		5	Total number of lost items for patron's entire history. Maximum value is 32767.
44	late media returns	416	n		5	Total number of late media returns for patron's entire history. Maximum value is 32767.
45	historical bookings	421	n		5	Total number of historical bookings for patron's entire history. Maximum value is 32767.
46	canceled bookings	426	n		5	Total number of canceled bookings for patron's entire history. Maximum value is 32767.
47	unclaimed bookings	431	n		5	Total number of unclaimed bookings for patron's entire history. Maximum value is 32767.
48	historical callslips	436	n		5	Number of historical call slips for this patron.
49	historical distributions	441	n		5	Total number of historical distributions for this patron's entire history.
50	historical short loans	446	n		5	Total number of historical short loans for this patron's entire history.

**Table 18-1. Base (Fixed) Segment of Record**

Item #	Item Name	Offset	Format	Required	Length	Description
51	unclaimed short loans	451	n		5	Total number of unclaimed short loans for this patron's entire history.
52	address count	456	n	Y	1	Valid data: 1 through 9. There must be at least 1 address segment (type 1: permanent) following. There may be as many as 8 additional addresses (2 through 9) which may be either type 2: temporary, or 3: e-mail.

\* Numeric if present, blank (not zero filled) otherwise.

1 If any one of the following are defined: patron barcode, patron group, barcode status, and modification date, the remaining three fields must also be defined (except the patron barcode, which can be blank to allow for a blank barcode patron). If a barcode already exists, it will be updated with this information. If a barcode exists for this patron with the same patron group, the status on the existing barcode will be changed to 5 (other) and the new one will be added with the status 1 (active), except where the input barcode is blank and the patron group and barcode status match an existing record. If this is the case, no alterations are made.

2 If there is more than one patron barcode present, the associated patron groups must be unique. At least one patron group must be present.

3 For each patron, either the social security number or the institution ID is required and that number must also be unique to the current record. There can be no duplicates. The social security number will be used in update runs to identify the record. If it is absent, the institution ID will be used in its place. Only one of these two fields may be altered on an update as the other will be required to identify the record in the database.

## Address Segment

[Table 18-2](#) defines the address portion of the record. This segment is repeatable up to nine times consecutively. All address segments must follow the base segment. Note that for the address segment, these sequence numbers might vary, according to the number of the address records.

**Table 18-2. Address Segment of Record.**

Item #	Item Name	Offset	Format	Required	Length	Description
53	address id	457	n		10	Internal record number for Voyager.
54	address type	467	n	y	1	<p>Valid data:</p> <p>1 = permanent -- only one is permitted</p> <p>2 = temporary</p> <p>3 = e-mail</p> <p>Permanent address (type = 1) update: If existing permanent address has the protect_address flag = y, an error will be reported. If not, the address will be updated with the new data.</p> <p>Temporary/e-mail address update: Any existing temporary/e-mail addresses without address protection will be deleted from the database and replaced by the new type 2 and 3 addresses supplied. Type 2 and 3 addresses with address protection will not be deleted. All addresses added/updated will default to not protected.</p>

**Table 18-2. Address Segment of Record.**

Item #	Item Name	Offset	Format	Required	Length	Description
55	address status code	468	s	y	1	Valid data: n = normal h = hold mail  The next two items represent the beginning and ending effective dates for the address. For e-mail and temporary addresses this date will be rejected if it overlaps dates of another address of the same type.
56	address begin date	469	d	y	10	Beginning date.
57	address end date	479	d	y	10	Ending date. Beginning and Ending Dates can <i>not</i> be the same date.
58	address line 1	489	s	y	50	Street address, or e-mail address for address type 3 (cannot be blank).
59	address line 2	539	s		40	Second line of street address.
60	address line 3	579	s		40	Third line of street address.
61	address line 4	619	s		40	Fourth line of street address.
62	address line 5	659	s		40	Fifth line of street address.
63	city	699	s		40	City
64	state (province) code	739	s		7	Any 2-character state or 7-character province code.
65	zipcode/postal code	746	s		10	Zipcode in format 12345 or 12345-6789 or other postal code.
66	country	756	s		20	Country: any format is valid.

**Table 18-2. Address Segment of Record.**

Item #	Item Name	Offset	Format	Required	Length	Description
67	phone (primary)	776	s		25	Primary phone: any format is valid.
68	phone (mobile)	801	s		25	Mobile phone: any format is valid.
69	phone (fax)	826	s		25	Fax phone: any format is valid.
70	phone (other)	851	s		25	Other phone: any format is valid.
71	date added/updated	876	b		10	This will be set to the run-date of the load.

**Variable Segment**

[Table 18-3](#) defines the variable segment of the record. This segment must follow the last address segment. The segment is not required and may be between 0 (zero) and 1000 characters in length. A segment of zero characters (a non-existent segment) indicates that there are no notes for the record.

**NOTE:**

If the user is performing a Patron Update and extended notes processing is selected, then each note in the Notes segment, must be preceded by a tab character. This includes the first note.

**Table 18-3. Variable (Notes) segment of record**

Item #	Item Name	Offset	Format	Required	Length	Description
72	Notes	After Last Address	s		1000 Max.	Variable Portion of Record Patron notes. This field may contain any data. A field of zero characters in length indicates that there are no notes.

## End-of-Record Segment

[Table 18-4](#) defines the end-of-record marker. This segment must follow the notes segment.

**Table 18-4. End-of-record marker**

Item #	Item Name	Offset	Format	Required	Length	Description
73	End-of-Record Marker	Last	s	y	1 or 2	Valid data: Line-feed (\n or hex '0A'). A carriage-return/line-feed is also acceptable.



---

**Charge Transaction Record  
Standard Interface File**

---

**19**

<b>Introduction</b>	<b>20-1</b>
<b>Purpose of This Chapter</b>	<b>20-1</b>
<b>Input File Specification</b>	<b>20-1</b>
<b>Charge Transaction Record SIF Format</b>	<b>20-2</b>



### **Introduction**

---

The Charge Transaction Record Standard Interface File (SIF) is the format of the file that contains your site's charge transactions that are loaded during migration.

### **Purpose of This Chapter**

---

This section discusses the layout of the Charge Transaction Record SIF.

- Input File specification
- File format fixed segment

### **Input File Specification**

---

The data required for the Charge Transaction Records should be supplied in the following manner.

Each charge transaction record should be one continuous string, terminated by a line-feed (N).

The fields must be the indicated length and must all be present and in a specific order and format.

The data fields which are designated numeric must be right-justified and zero-filled on the left.

The data fields which are designated strings (not designated numeric) should be left-justified and blank-filled on the right but not hex-null-terminated.

The date format should be entered as CCYY.MM.DD and the Time format should be entered as HH:MM, where C is century, Y is year, M is month, D is day, H is hour, and S is second.

The data fields which are supplied (not blanks or zeroes) will be updated if they already exist or added if they do not.

## Charge Transaction Record SIF Format

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Type** Definition of the type of input.

**Offset** Actual starting position of the item in the record relative to 1.

**Format** Format of the variable.

**s** String, left-justified/blank filled.

**n** Numeric, right-justified/zero-filled.

**d** Date in the format ccyy.mm.dd, where c is the century, y is the year, m is the month, and d is the day. The maximum high value is 2382.12.31 which can be used no expiration. This field may be blank under certain conditions.

**t** Time in the format HH:MM.

**b** Blanks.

**Required** Y for yes, the field must have valid data in it (not just blanks).

[Table 19-1](#) defines the format required for all incoming records.

**Table 19-1. Charge Transaction Standard Interface File Format**

Item #	Item Name	Type	Offset	Format	Required	Length	Description
01	item barcode	varchar	1	s	Y	25	Barcode link to item record.
02	patron barcode	varchar	26	s	Y	25	Barcode link to the patron record.
03	date charged	date	51	d	Y	10	Date the item was checked out to a patron.
04	time charged	time	61	t	Y	5	Time the item was checked out to a patron.
05	date due	date	66	d	Y	10	The most current date the item is expected back.
06	time due	time	76	t	Y	5	The most current time the item is expected back.
07	date renewed	date	81	d	N	10	The most current date the item was renewed.
08	time renewed	time	91	t	N	5	The most current time the item was renewed.
09	renew count	numeric	96	n	N	5	The number of times the item has been renewed.
10	overdue notice count	numeric	101	n	N	3	The number of overdue notices sent for the item record.
11	overdue notice date	date	104	d	N	10	The last date an overdue notice was sent for the item record, if the overdue notice count is greater than zero.



<b>Introduction</b>	<b>21-1</b>
<b>Purpose of This Chapter</b>	<b>21-1</b>
<b>File Format</b>	<b>21-1</b>



---

## **Introduction**

---

The Item Delete Standard Interface File (SIF) is the format of the file to be produced by any staff client when an item is successfully deleted. Items deleted in any of the modules will append to this file.

---

## **Purpose of This Chapter**

---

This section discusses the layout of the Item Delete SIF.

- File Format

---

## **File Format**

---

All fields are comma delimited.

If a field is not present, it is represented by a comma for the position it would occupy.

The filename is located in the `/m1/voyager/xxxdb/rpt` directory and is named `delete.item`.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

[Table 20-1](#) describes the Item Delete SIF.

**Table 20-1. Item Delete SIF Format**

Position	Description	Notes
1	item barcode	Only the active barcode is placed in the SIF.
2	item id	System generated item record id.
3	mfhd id	System generated MFHD record id.
4	bib id	System generated bibliographic record id.
5	title	Title of the item.
6	create operator id	ID of the operator who created the record.
7	delete operator id	ID of the operator who deleted the record.
8	deletion date/time	Date and time item was deleted.
9	bib 010 \$a	LCCN.
10	item type id	Item type id.
11	item type code	Item type code.
12	media type id	Media type id.
13	media type code	Media type code.
14	location id	Location id.
15	location code	Location code.
16	enum	Enumeration of the item.
17	chron	Chronology of the item.
18	year	Year from the item record.
19	caption	Caption in the item record.
20	free text	Note from the item record.
21	spine label	Spine label.
22	copy number	Copy number for this item.
23	pieces	Pieces on the item record.
24	price	Price formatted according to the base currency.

---

## **Vendor Record Standard Interface Format**

---

# **21**

<b>Introduction</b>	<b>22-1</b>
<b>Purpose of This Chapter</b>	<b>22-1</b>
<b>Input File Specification</b>	<b>22-1</b>
<b>Vendor SIF Format</b>	<b>22-2</b>
• Base Segment	22-3
• Address Segment	22-4
• Variable Segment	22-6
• End-of-Record Segment	22-6



---

## **Introduction**

---

The Vendor Record Standard Interface File (SIF) is the format of the file of your site's vendor records loaded during migration.

---

## **Purpose of This Chapter**

---

This section discusses the layout for the Vendor Record SIF.

- Input file specification
- File format fixed segment
- File format address segment
- File format variable segment
- File format end-of-record segment

---

## **Input File Specification**

---

The data required for the Vendor database records should be supplied in the following manner.

Each vendor data record should be one continuous string, terminated by a line-feed (\N).

The fields must be the indicated length and must all be present and in a specific order and format. The notes segment is an exception to the length requirement.

The data fields which are designated numeric must be right-justified and zero-filled on the left.

The data fields which are designated strings (not designated numeric) should be left-justified and blank-filled on the right but not hex-null-terminated. The notes segment is an exception to the no-null termination requirement.

## **Vendor SIF Format**

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Offset** Actual starting position of the item in the record relative to 1.

**Format** Format of the variable.

**s** String, left-justified/blank filled.

**n** Numeric, right-justified/zero-filled.

**Required** Y for yes, the field must have valid data in it (not just blanks).

The maximum record should consist of the following:

1 base segment length:  $1 \times 147 = 147$

6 address segments length:  $6 \times 471 = 2826$

1 notes segment length:  $1 \times 2000 = 1900$

1 end-of record mark length:  $1 \times 1 = 1$

The total maximum record length is 4874.

The minimum record should consist of the following:

1 base segment length:  $1 \times 147 = 147$

1 address segment length:  $1 \times 471 = 471$

1 notes segment length:  $1 \times 1 = 1$

1 end-of record mark length:  $1 \times 1 = 1$

The total minimum record length is 620.

## Base Segment

[Table 21-1](#) defines the base segment of all incoming records.

**Table 21-1. Base Segment of Record**

Item #	Item	Offset	Format	Required	Length	Description
01	vendor type	1	s		2	Variable data based on user's set up of the vendor type table.
02	vendor code	3	s	Y	10	Variable data: user's discretion.
03	vendor name	13	s	Y	60	Vendor name.
04	vendor tax id	73	s		11	Vendor tax identification number.*
05	institution id	84	s		25	ID of ordering institution.
06	default currency	109	s		3	Variable data: user's discretion.
07	claim interval	112	n		5	Maximum value is 32767. This field is numeric (right-justified/zero-filled).
08	claim count	117	n		5	Maximum value is 32767. This field is numeric (right-justified/zero-filled).
09	cancel interval	122	n		5	Maximum value is 32767. This field is numeric (right-justified/zero-filled).
10	ship via	127	s		20	Variable data: user's discretion.

**Table 21-1. Base Segment of Record**

Item #	Item	Offset	Format	Required	Length	Description
11	address count	147	n	Y	1	Valid data: 1 through 6. There must be at least 1 address segment following. There can be as many as 6. This field is numeric (right-justified/zero-filled).
* Although the Federal Tax ID Number is officially defined as ten digits in length, this field has been defined to accept up to 11 digits by design, to account for the possibility of an alphabetic character at the end as per the IRS EIN documentation at <a href="http://ftp.fedworld.gov/pub/irs-pdf/p1635.pdf">http://ftp.fedworld.gov/pub/irs-pdf/p1635.pdf</a> . If the source info. to be loaded is only ten digits in length, this field may be padded with a space at the front of the number.						

**Address Segment**

[Table 21-2](#) defines the address portion of the incoming record. This segment is repeatable up to six times consecutively. All address segments must follow the base segment. Note that for the address segment, these sequence numbers might vary, according to the number of the address records.

**Table 21-2. Address segment of record**

Item #	Item	Offset	Format	Required	Length	Description
12	std. address number	148	s		8	Standard address number for the vendor.
13	type: order	156	s		1	Valid data: Y = Yes, N = No. Footnote 1
14	type: payment	157	s		1	Valid data: Y = Yes, N = No. Footnote 1
15	type: return	158	s		1	Valid data: Y = Yes, N = No. Footnote 1
16	type: claim	159	s		1	Valid data: Y = Yes, N = No. Footnote 1
17	type: email	160	s		1	Valid data: Y = Yes, N = No. Footnote 1

**Table 21-2. Address segment of record**

Item #	Item	Offset	Format	Required	Length	Description
18	type: other	161	s		1	Valid data: Y = Yes, N = No. Footnote 1
19	contact name	162	s		40	Name of the contact person at the vendor.
20	contact title	202	s		40	Title of the contact person at the vendor.
21	address line 1	242	s	desired	50	Address Line 1.
22	address line 2	292	s		40	Second line of street address.
23	address line 3	332	s		40	Third line of the street address.
24	address line 4	372	s		40	Fourth line of street address.
25	address line 5	412	s		40	Fifth line of street address.
26	city	452	s		30	Any value up to the maximum length.
27	state (province) code	482	s		7	Any 7-character state (province) code.
28	zipcode/postal code	489	s		10	Zipcode in format 12345 or 12345-6789
29	country	499	s		20	Any value up to the maximum length.
30 <sup>D</sup>	phone (primary)	519	s	desired	25	Any value up to the maximum length. At least one phone number (primary) should be present.
31	phone (mobile)	544	s		25	Any value up to the maximum length.
32	phone (fax)	569	s		25	Any value up to the maximum length.
33	phone (other)	594			25	Any value up to the maximum length.
1 at least one of these address types must be marked "y" (yes).						

## Variable Segment

---

[Table 21-3](#) defines the variable segment of the incoming record. This segment must follow the last address segment and is an exception to the length and no-null-termination requirements. The notes segment is the only portion which is variable in length. It must be present and it must be null-terminated but it may consist of only a null-terminator (\0 or hex '00').

**Table 21-3. Variable (Notes) segment of record**

Item #	Item Name	Offset	Format	Required	Length	Description
34	notes	619	s		1900 max	Vendor notes. This field may contain any data the user wants. This segment must be present but may be simply a null-terminator to indicate that there are no notes.

## End-of-Record Segment

---

[Table 21-4](#) defines the end-of-record marker. This segment must follow the notes segment.

**Table 21-4. End-of-record marker**

Item #	Item Name	Offset	Format	Required	Length	Description
35	End-of-Record Marker		s	Y	1	Valid Data: Line-feed (\N).

---

## **Acquisitions Notices Standard Interface File**

# **22**

---

<b>Introduction</b>	<b>23-1</b>
<b>Purpose of This Chapter</b>	<b>23-1</b>
<b>File Specification</b>	<b>23-2</b>
<b>Acquisitions Notices SIF Format</b>	<b>23-3</b>
• Base Segment	23-4
• Suffixes	23-5
Cancellation Notice Suffix (00)	23-6
Return Notice Suffix (01)	23-6
Order Claim Notice Suffix (02)	23-7
Serial Claim Notice Suffix (03)	23-7
Voucher/Check Request Suffix (04)	23-8
Cancel Serial Claim Notice Suffix (05)	23-8



### Introduction

---

The Acquisitions Notices Standard Interface File (SIF) is the file produced by activity at acquisitions locations and/or after running acquisitions batch jobs on the server.

Acquisitions activity and/or running acquisitions batch jobs produces a file containing the notice information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate notices. See [Creating Input Files for Acquisitions](#) in the *Voyager Reporter User's Guide* for more information.

### Purpose of This Chapter

---

This section discusses the layout for the Acquisitions Notices SIF.

- File specification
- File format base segment
- File format suffix segment

## File Specification

---

The name of the file produced is acqnotes.[print location code].inp, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the *Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called acqnotes.main.inp. When the file is generated, it is placed in the /m1/voyager/xxxdb/rpt directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the acquisitions notices SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of notice for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample acquisitions notice record might appear as follows:

```
00|97.2||2|EBSCO|John Adams|10400 Higgins Road|||||Des  
Plaines|IL|60018|||10/12/1999|Ex Libris Library of  
InformationAcquisitions|123 9th Street|||Des  
Plaines|IL|60000|||||56|08/04/1999||British journal of  
anaesthesia.|ISSN 0007-0912|2|2
```

The first 29 fields of the record are the base segment. Because this is a cancellation notice record, the suffix is made up of the last seven fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of notice. The notice numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## Acquisitions Notices SIF Format

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Required** If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description** Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Cancellation Notice = 36 fields.
- Type 01: Return Notice = 36 fields.
- Type 02: Order Claim Notice = 37 fields.
- Type 03: Serial Claim Notice = 38 fields.
- Type 04: Voucher/check request = 44 fields.
- Type 05: Cancel Serial Claim Notice = 38 fields.

## Base Segment

---

[Table 22-1](#) defines the base segment for all acquisitions notice type records.

**Table 22-1. Base Segment for Acquisitions Notices**

Item #	Required	Item Name	Length	Description
1	Y	notice id	2	This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice: 00 is a cancellation notice 01 is a return notice 02 is an order claim notice 03 is a serial claim notice 04 is a voucher/check request 05 is a cancel serial claims
2	Y	version number	30	Program version number to be used by client for version checking.
3		e-mail address	50	Vendor's e-mail address. If this field is not blank, the notice will be sent by e-mail. If it is blank, the notice will be printed for mailing.
4	Y	vendor id	10	Vendor's ID number.
5	Y	vendor name	60	Vendor's name.
6		attention	40	Attention to.
--->>		note		Addresses to be used for various notices are as follows: cancellation notice -- order address. return notice -- return address. order/serial claim notices -- claim address. voucher/check requests -- payment address.
7	Y	address line 1	50	Vendor's address line 1.
8		address line 2	40	Vendor's address line 2.
9		address line 3	40	Vendor's address line 3.
10		address line 4	40	Vendor's address line 4.

**Table 22-1. Base Segment for Acquisitions Notices**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
11		address line 5	40	Vendor's address line 5.
12		city	30	Vendor's city.
13		state/province	7	Vendor's state/province.
14		postal code	10	Vendor's postal code.
15		country	20	Vendor's country.
16		phone	25	Vendor's phone.
17		date	10	Current date.
18	Y	institution	50	Institution name.
19	Y	order site	25	Cancellation notices: order site location name.  Return notices: order site location name.  Order claim notices: order site location name.  Serial claim notices: ship to site.  Voucher/check request: bill to site.
20		address line 1	50	Order site address line 1.
21		address line 2	40	Order site address line 2.
22		address line 3	40	Order site address line 3.
23		city	30	Order site city.
24		state/province	7	Order site state/province.
25		postal code	10	Order site postal code.
26		country	20	Order site country.
27		phone	25	Order site phone.
28		fax	25	Order site fax.
29		e-mail address	50	Order site e-mail address.

**Suffixes**

[Table 22-2](#) through [Table 22-7](#) cover the various suffix segments of the acquisitions notices.

**Cancellation Notice Suffix (00)**

[Table 22-2](#) describes the cancellation notice suffix.

**Table 22-2. Cancellation Notice Suffix (00)**

Item #	Required	Item Name	Length	Description
30	Y	P.O. #	25	A single P.O. number.
31	Y	P.O. date	10	Purchase order date.
32		invoice number	25	Voyager invoice number for this line-item.
33	Y	title	255	Title/edition.
34		standard number	100	Standard number (use vendor title/number if available).
35	Y	line item number	10	Single line-item number.
36	Y	number of copies	10	Number of copies to cancel. <u>Last item in the record for</u> cancellation notices.

**Return Notice Suffix (01)**

[Table 22-3](#) describes the return notice suffix.

**Table 22-3. Return Notice Suffix (01)**

Item #	Required	Item Name	Length	Description
30	Y	P.O. #	25	A single P.O. number.
31	Y	P.O. date	10	Purchase order date.
32		invoice number	25	Voyager invoice number for this line-item.
33	Y	title	255	Title/edition.
34		standard number	100	Standard number (use vendor title/number if available).
35	Y	line item number	10	Single line-item number.
36	Y	number of copies	10	Number of copies returned. <u>Last item in the record for</u> return notices.

## Order Claim Notice Suffix (02)

[Table 22-4](#) describes the order claim notice suffix.

**Table 22-4. Order Claim Notice Suffix (02)**

Item #	Required	Item Name	Length	Description
30	Y	P.O. #	25	A single P.O. number.
31	Y	P.O. date	10	Purchase order date.
32	Y	title	255	Title/edition.
33		standard number	100	Standard number (use vendor title/number if available).
34	Y	line item number	10	Single line-item number.
35		claim type	70	Claim type.
36	Y	number of copies	10	Number of copies claimed.
37		note to vendor	255	Line-item note to vendor. <u>Last item in the record for</u> order claim notices.

## Serial Claim Notice Suffix (03)

[Table 22-5](#) describes the serial claim notice suffix.

**Table 22-5. Serial Claim Notice Suffix (03)**

Item #	Required	Item Name	Length	Description
30	Y	P.O. #	25	A single P.O. number.
31	Y	P.O. date	10	Purchase order date.
32	Y	title	255	Title.
33		issn	100	ISSN.
34	Y	line item number	10	Single line-item number.
35		claim type	70	Claim type.
36	Y	number of copies	10	Number of copies claimed.
37		note to vendor	255	Line-item note to vendor.
38		description	255	Issue information. <u>Last item in the record for</u> serial claim notices.

**Voucher/Check Request Suffix (04)**

[Table 22-6](#) describes the voucher/check request suffix.

**Table 22-6. Voucher/Check Request Suffix (04)**

Item #	Required	Item Name	Length	Description
30	Y	voucher id	25	Voyager voucher ID.
31	Y	currency	35	Transaction currency description.
32	Y	invoice amount	10	Voyager invoice amount.
33	Y	invoice number	25	Voyager invoice number.
34	Y	invoice date	10	Voyager invoice date.
35		vendor inst. id	25	Vendor institution ID.
36	Y	P.O. #	25	A single P.O. number.
37		P.O. date	10	Purchase order date.
38	Y	line item number	10	Single line-item number.
39	Y	line item title	255	Title for the item.
40	Y	line item amount	10	Line item amount (for this fund).
41	Y	fund total	10	Total amount for this fund which applies to this invoice.
42	Y	fund description	125	Fund description.
43	Y	ledger id	25	Ledger id used for sort key.
44	Y	fund id	25	Fund id used for sort key. <u>Last item in the record</u> for voucher/check requests.

**Cancel Serial Claim Notice Suffix (05)**

[Table 22-7](#) describes the cancel serial claim notice suffix.

**Table 22-7. Cancel Serial Claim Notice Suffix (05)**

Item #	Required	Item Name	Length	Description
30	Y	P.O. #	25	A single P.O. number.
31	Y	P.O. date	10	Purchase order date.
32	Y	title	255	Title.
33		issn	100	ISSN.

**Table 22-7. Cancel Serial Claim Notice Suffix (05)**

Item #	Required	Item Name	Length	Description
34	Y	line item number	10	Single line-item number.
35		claim type	70	Claim type.
36	Y	number of copies	10	Number of copies claimed.
37		note to vendor	255	Line-item note to vendor.
38		description	255	Issue information. <b><u>Last item in the record for</u></b> cancel serial claim notices.



---

## **Acquisitions Reports Standard Interface File**

# **23**

---

<b>Introduction</b>	<b>24-1</b>
<b>Purpose of This Chapter</b>	<b>24-1</b>
<b>File Specification</b>	<b>24-1</b>
<b>Acquisitions Reports SIF Format</b>	<b>24-2</b>
• Base Segment	24-4
• Suffixes	24-4
Purchase Orders Report (00)	24-5
Open Orders Report (01)	24-8
Global Open Orders Report (02)	24-9
Fund Snapshot Report Suffix (03)	24-10
Open Orders Before Rollover Report (04)	24-12
Open Orders After Rollover Report (05)	24-12
Fund Snapshot Before Rollover Report (06)	24-12
Fund Snapshot After Rollover Report (07)	24-12
Fund Rollover Status Report Suffix (08)	24-12
Copy Rollover Status Report Suffix (09)	24-13
	24-13



## Introduction

---

The Acquisitions Reports Standard Interface File (SIF) is a file produced by running acquisitions batch jobs on the server.

Running acquisitions batch jobs produces a file containing the report information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate reports. See [Creating Input Files for Acquisitions](#) in the *Voyager Reporter User's Guide* for more information.

## Purpose of This Chapter

---

This section discusses the layout for the Acquisitions Report SIF.

- File specification
- File format base segment
- File format suffix segment

## File Specification

---

The name of the file produced is acqrpts.[print location code].inp, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the

*Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called acqrpts.main.inp. When the file is generated, it is placed in the /m1/voyager/xxxdb/rpt directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the acquisitions reports SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of report for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample acquisitions report record might appear as follows:

```
06|97.2|08/16/1999|1999 (01/01/1999 - 12/31/1999)|Main  
Library FY99|1999 (01/01/1999 - 12/31/1999)|Main  
Acquisitions Group|American  
Literature|1|General|Literature|0|General|||N|  
10000.00|10000.00|9874.19|10000.00|0.00|0.00|125.81|0.0  
0|110%|102%
```

The first two fields of the record are the base segment. Because this is a Fund Snapshot Before Rollover report record, the suffix is made up of the last 20 fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of report. The report numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## **Acquisitions Reports SIF Format**

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, is only the field sequence numbers and are not to be included in the record.

**Item name** Definition of the item.

**Required** If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description** Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Purchase Order Report = 52 fields.
- Type 01: Open Orders Report = 22 fields.
- Type 02: Global Open Orders Report = 22 fields.
- Type 03: Fund Snapshot Report = 26 fields.
- Type 04: Open Orders Before Rollover = 22 fields.
- Type 05: Open Orders After Rollover = 22 fields.
- Type 06: Fund Snapshot Before Rollover = 22 fields.
- Type 07: Fund Snapshot After Rollover = 22 fields.
- Type 08: Fund Rollover Status = 9 fields.
- Type 09: Copy Rollover Status = 11 fields.

## Base Segment

---

[Table 23-1](#) defines the base segment for all acquisitions report type records.

**Table 23-1. Base segment for acquisitions reports**

Item #	Required	Item Name	Length	Description
1	Y	report id	2	<p>This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice:</p> <ul style="list-style-type: none"> <li>00 is a purchase order.</li> <li>01 is an open order report.</li> <li>02 is a global open orders report.</li> <li>03 is a fund snapshot report.</li> <li>04 is an open order rpt before roll-over.</li> <li>05 is an open order rpt after rollover.</li> <li>06 is a fund snapshot rpt before roll-over.</li> <li>07 is a fund snapshot rpt after roll-over.</li> <li>08 is a fund rollover status rpt.</li> <li>09 is a copy rollover status rpt.</li> </ul>
2	Y	version number	30	Program version number to be used by client for version checking.

## Suffixes

---

[Table 23-2](#) through [Table 23-7](#) cover the various suffix segments of the acquisitions reports.

## Purchase Orders Report (00)

[Table 23-2](#) describes the purchase order report suffix.

**Table 23-2. Purchase Orders Report Suffix (00)**

Item #	Required	Item Name	Length	Description
3	Y	order_site	50	Institution name.
4		rush_order	25	The word 'rush' if entire order is.
5	Y	p_o_number	25	Purchase order number.
6	Y	p_o_date	10	Date, preferred format: mm/dd/ccyy
7	Y	ordered_by	10	Name of ordering person.
8	Y	vendor_name	60	Vendor's name.
9		address line 1	50	Vendor's address line 1.
10		address line 2	50	Vendor's address line 2.
11		address line 3	50	Vendor's address line 3.
12		address line 4	50	Vendor's address line 4.
13		city	30	Vendor's city.
14		state/province	7	Vendor's state/province.
15		postal code	10	Vendor's postal code.
16		country	20	Vendor's country.
17		ship_via	20	Shipping method.
18		note to vendor	60	Order-level note to vendor.
19	Y	print_id	12	ID number used to keep individual copies of a purchase order separate.
20	Y	line_item_number	4	Line item number.
21		item_rush_word	18	The word rush in some language if this item (but not the entire order) is to be rushed.

**Table 23-2. Purchase Orders Report Suffix (00)**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
22	Y	description	255	A maximum of 255 characters will be printed due to limitations of MSAccess. This item is comprised of the following pieces:  Title and piece identifier.  The title will always be present. MARC 245 a s. It may be as long as 150 characters.  The piece identifier may be present. The sum of these two items must not exceed 255 characters, it is no longer necessary to calculate how many lines are needed for a line item as line spacing will be handled by the client program.
23		line item note	255	Line-item level note to the vendor.
24		publisher	255	Publisher identification.
25		vendor_title_number	100	Vendor title/number if it exists, otherwise use the print standard number for the line item preceded by its standard number type description. example:  ISSN 123456789 or PUB NO. 987654321. See the "vendor title" section following this table.
26	Y	quantity/unit	4	Number of units ordered.
27		unit_cost	12	Cost of 1 unit in format: 9.99
28		adjustments	12	Total of all adjustments to line item.
29		prepay_amount	12	Prepay amount of 1 unit in format: 9.99
30		net_total_cost	12	Extended cost in format: 9.99.  The extended cost is calculated by the following formula: ((UNIT COST * (NUMBER OF UNITS) +/- ADJUSTMENTS - PREPAY AMOUNT)).
31		sub_total	12	Sub_total of ordered items on this page: 9.99

**Table 23-2. Purchase Orders Report Suffix (00)**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
32		other_charges	12	Other charges applicable to this page: 9.99
33		order_total	12	Total of all items on this entire purchase order (all pages): 9.99
34	Y	ship_to_name	50	Ship to name.
35	Y	address line 1	50	Ship to address line 1.
36		address line 2	50	Ship to address line 2.
37		address line 3	50	Ship to address line 3.
38		address line 4	50	Ship to address line 4.
39		city	30	Ship to city.
40		state/province	7	Ship to state/province.
41		postal code	10	Ship to postal code.
42		country	20	Ship to country.
43	Y	bill_to_name	50	Bill to name.
44		address line 1	50	Bill to address line 1.
45		address line 2	50	Bill to address line 2.
46		address line 3	50	Bill to address line 3.
47		address line 4	50	Bill to address line 4.
49		city	30	Bill to city.
50		state/province	7	Bill to state/province.
51		postal code	10	Bill to postal code.
52		country	20	Bill to country.
--		record length	1983	Record length (not included as part of record).

**Vendor Title/Number Field**

The Vendor Title/Number will be provided if present. If not, print\_std\_num is specified in the line-item. If the latter is the case, use MARC fields and supply the label and the number as follows. If print\_std\_num is:

IS use: marc 022|a and supply: ISSN + number

IB use: marc 020|a and supply: ISBN + number

CO use: marc 030|a and supply: CODEN + number

GP use: marc 037|a and supply: STOCK NO. + number

PN use: marc 028|a and supply: PUB NO. + number

ST use: marc 027|a and supply: STRN + number

## Open Orders Report (01)

[Table 23-3](#) describes the open orders report suffix.

**Table 23-3. Open Orders Report Suffix (01)**

Item #	Required	Item Name	Length	Description
3	Y	report_date	10	Report run date.
4	Y	for_site	50	For institution name (or all).
5	Y	for_ledger	125	For ledger name (or all), fiscal-period (dates).
6	Y	order_site	50	Institution name.
7	Y	ledger_name	125	Ledger name, fiscal-period (dates).
8	Y	p_o_number	25	Purchase order number.
9	Y	p_o_id	50	Purchase order system ID number.
10	Y	vendor_name	60	Vendor's name.
11	Y	p_o_type	25	Purchase order type.
12	Y	line_item-number	4	Line item number.
13		description	255	This item is comprised of the following pieces: title and piece identifier.
14	Y	copy_number	4	Copy sequence number.
15	Y	location	25	Target location.
16		item_type	25	Line item type.
17	Y	copy_status	25	Copy status.
18		invoice_status	25	Invoice status.
19		item_amount	20	Line item amount [(\$987,654,321.1234567].
20	Y	fund_count	3	Number of fund segments to follow.
21	Y	ledger_id	15	Ledger_id: used for sorting fund segments.

**Table 23-3. Open Orders Report Suffix (01)**

Item #	Required	Item Name	Length	Description
22		fund_name	255	Fund name. If copy is allocated to more than 1 fund, etc., additional fund name(s), ledger(s), fiscal period(s), and percentages will be included here as additional (ledger_id/fund_name) segments.
note:		--->>	-->>	Ledger_id and fund_name fields (paired) constitute a repeatable segment. The fund_count field will have the number of these segments.
--		record length	1148	Record length (not included as part of record).

**Global Open Orders Report (02)**

[Table 23-4](#) describes the global open orders report suffix.

**Table 23-4. Global Open Orders Report Suffix (02)**

Item #	Required	Item Name	Length	Description
3	Y	report_date	10	Report run date.
4	Y	for_site	50	For institution name (or all).
5	Y	for_ledger	125	For ledger name (or all) , fiscal-period (dates).
6	Y	order_site	50	Institution name.
7	Y	ledger_name	125	Ledger name, fiscal-period (dates).
8	Y	p_o_number	25	Purchase order number.
9	Y	p_o_id	50	Purchase order system ID number.
10	Y	vendor_name	60	Vendor's name.
11	Y	p_o_type	25	Purchase order type.
12	Y	line_item-number	4	Line item number.
13		description	255	This item is comprised of the following pieces: title and piece identifier.
14	Y	copy_number	4	Copy sequence number.

**Table 23-4. Global Open Orders Report Suffix (02)**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
15	Y	location	25	Target location.
16		item_type	25	Line item type.
17	Y	copy_status	25	Copy status.
18		invoice_status	25	Invoice status.
19		item_amount	20	Line item amount [(\$)987,654,321.1234567].
20	Y	fund_count	3	Number of fund segments to follow.
21	Y	ledger_id	15	Ledger_id: used for sorting fund segments.
22		fund_name	255	Fund name. If copy is allocated to more than 1 fund, etc., additional fund name(s), ledger(s), fiscal period(s), and percentages will be included here as additional (ledger_id/fund_name) segments.
note:		--->>	-->>	Ledger_id and fund_name fields (paired) constitute a repeatable segment. The fund_count field will have the number of these segments.
--		record length	1148	Record length (not included as part of record).

**Fund Snapshot Report Suffix (03)**

[Table 23-5](#) describes the fund snapshot report suffix.

**Table 23-5. Fund Snapshot Report Suffix (03)**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
3	Y	report_date	10	Report run date.
4	Y	for_f_period	100	For fiscal period name (or all).
5	Y	ledger_name	40	Ledger name.
6	Y	fiscal_period	100	Fiscal period name.
7	Y	policy_group	40	Policy group name.
8	Y	fund_name	25	Fund name.

**Table 23-5. Fund Snapshot Report Suffix (03)**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
9	Y	fund_cat	2	Fund category: 00=summary, 01=allocated, 02=reporting.
10	Y	fund_type	25	Fund type name (descriptive name).
11		parent_fund_name	25	Fund name (parent).
12		parent_fund_cat	2	Fund category: 00=summary, 01=allocated, 02=reporting.
13		parent_fund_type	25	Fund type name (descriptive name).
14		begin_date	10	Fund begin date.
15		end_date	10	Fund end date.
16	Y	expend_only	1	Yes or no.
17		original_alloc	20	Original allocation [(\$987,654,321.1234567].
18		net_alloc	20	Net allocation [(\$987,654,321.1234567].
19		bal_avail	20	Balance available [(\$987,654,321.1234567].
20		bal_cash	20	Cash balance [(\$987,654,321.1234567].
21		pend_commit	20	Pending commitments [(\$987,654,321.1234567].
22		pend_expend	20	Pending expenditures [(\$987,654,321.1234567].
23		commits	20	Commitments [(\$987,654,321.1234567].
24		expends	20	Expenditures [(\$987,654,321.1234567].
25		over_commit	4	Over commitment percent (999%).
26		over_expend	4	Over expenditure percent (999%).
--		record length	525	Record length (not included as part of record).

**NOTE:**

There must be a separate Fund Snapshot Report Record for each fund to be reported. The record has been designed so that all funds may be reported in the same format. However, while all funds may be reported in the same record format, not all records will include data for all of the data items in the record. Be aware that even if a field is not appropriate for the type of fund you are reporting, it (the field) must still be present and represented by the pipe (|) delimiter.

**Open Orders Before Rollover Report (04)**

There must be one Open Orders Before Rollover record for each line-item which is to be displayed on the report. This report is currently the same layout as the standard Open Orders Report (report type 01).

**Open Orders After Rollover Report (05)**

There must be one Open Orders After Rollover record for each line-item which is to be displayed on the report. This report uses the same layout as the standard Open Orders Report (report type 01).

**Fund Snapshot Before Rollover Report (06)**

There must be a separate Fund Snapshot Report Record for each fund to be reported. The record has been designed so that all funds may be reported in the same format. This report uses the same layout as the standard Fund Snapshot Report (report type 03).

**Fund Snapshot After Rollover Report (07)**

There must be a separate Fund Snapshot Report Record for each fund to be reported. The record has been designed so that all funds may be reported in the same format. This report uses the same layout as the standard Fund Snapshot Report (report type 03).

**Fund Rollover Status Report Suffix (08)**

[Table 23-6](#) describes the fund rollover status report suffix.

**Table 23-6. Fund Rollover Status Report Suffix (08)**

Item #	Required	Item Name	Length	Description
3	Y	report_date	10	Report run date.
4	Y	rollover_Run_ID	50	Unique rollover run identifier.
5	Y	for_fiscal_period	100	For fiscal-period (dates).

**Table 23-6. Fund Rollover Status Report Suffix (08)**

Item #	Required	Item Name	Length	Description
6		ledger_name	125	Ledger name.
7		fund_name	25	Fund name.
8		rollover_status	255	Rollover status string.
9	Y	time_stamp	25	Rollover time.
--		record length	537	Record length (not included as part of record).

**Copy Rollover Status Report Suffix (09)**

[Table 23-7](#) describes the copy rollover status report suffix.

**Table 23-7. Copy Rollover Status Report Suffix (09)**

Item #	Required	Item Name	Length	Description
3	Y	report_date	10	Report run date.
4	Y	rollover_run_id	50	Unique rollover run identifier.
5	Y	p_o_number	25	Purchase order number.
6		item_number	4	Line item number.
7		description	255	Item title.
8		vendor title num- ber	100	Vendor's item identification.
9		copy_location	25	Copy location. To be replaced with copy sequence number when that becomes available.
10		rollover_status	255	Rollover status string.
11	Y	time_stamp	25	Rollover time.
--		record length	781	Record length (not included as part of record).



---

## Cataloging Reports Standard Interface File

# 24

---

<b>Introduction</b>	<b>25-1</b>
<b>Purpose of This Chapter</b>	<b>25-1</b>
<b>File Specification</b>	<b>25-1</b>
<b>Cataloging Reports SIF Format</b>	<b>25-2</b>
• Base Segment	25-4
• Suffixes	25-4
Unauthorized Subject Headings (00)	25-5
Unauthorized Name Headings (01)	25-5
Unauthorized Title Headings (02)	25-5
Unauthorized Name/Title Headings (03)	25-6
Unauthorized Subdivision Headings (04)	25-6
Duplicate Authority Records (05)	25-6
‘See’ References with Linked Bib Records (06)	25-7
‘See’ References Authorized in Another Authority Record (07)	25-7
‘See’ References without Corresponding Authority Record (08)	25-8
856 Link Failure Report Suffix (09)	25-8



---

## Cataloging Reports Standard Interface File

# 24

---

### Introduction

---

The Cataloging Reports Standard Interface File (SIF) is the file produced by running cataloging batch jobs on the server.

Running the cataloging batch jobs produces a file containing the report information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate the reports. See [Creating Input Files for Cataloging](#) in the *Voyager Reporter User's Guide* for more information.

---

### Purpose of This Chapter

---

This section discusses the layout for the Cataloging Reports SIF.

- File specification
- File format base segment
- File format suffix segment

---

### File Specification

---

The name of the file produced is catrprts.[print location code].inp, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the

*Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called catrprts.main.inp. When the file is generated, it is placed in the /m1/voyager/xxxdb/rpt directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the cataloging reports SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of report for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample cataloging report record might appear as follows:

```
09|97.2||198.80.36.91|http://198.80.36.91/inss/
books/ootw/ ootwhome.html|URL|996|Socket Write
Error|Bibliographic| 61218|09/16/1999
```

The first three fields of the record are the base segment. Because this is an 856 Link Failure Report report record, the suffix is made up of the last eight fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of report. The report numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## Cataloging Reports SIF Format

---

The following conventions are used in the listed tables.

<b>Item #</b>	Represents the relative position of the item in the record (sequence).
---------------	--

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name**      Definition of the item.

**Required**      If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description**      Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Unauthorized Subject Headings = 8 fields.
- Type 01: Unauthorized Name Headings = 8 fields.
- Type 02: Unauthorized Title Headings = 8 fields.
- Type 03: Unauthorized Name/Title Headings = 8 fields.
- Type 04: Unauthorized Subdivision Headings = 8 fields.
- Type 05: Duplicate Authority Records = 9 fields
- Type 06: 'See' Ref's with Linked Bib Records = 10 fields.
- Type 07: 'See' Ref's Authorized in Another Record = 11 fields.
- Type 08: 'See' Ref's without an Authority Record = 10 fields.
- Type 09: 856 Link Failure Report = 11 fields.

## Base Segment

---

[Table 24-1](#) defines the base segment for all cataloging report type records.

**Table 24-1. Base segment for cataloging reports**

Item #	Required	Item Name	Length	Description
1	Y	report id	2	<p>This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice:</p> <ul style="list-style-type: none"> <li>00 is an unauthorized subject heading.</li> <li>01 is an unauthorized name heading.</li> <li>02 is an unauthorized title heading.</li> <li>03 is an unauthorized name/title heading.</li> <li>04 is an unauthorized subdivision heading.</li> <li>05 is a duplicate authority record.</li> <li>06 is a 'see' references with linked bib records.</li> <li>07 is a 'see' references authorized in another authority record.</li> <li>08 is a 'see-also' reference without a corresponding authority record.</li> <li>09 is an 856 link failure report.</li> </ul>
2	Y	version number	30	Program version number to be used by client for version checking.
3		date[time]/selection (range)	50	Date (mm/dd/ccyy) with optional time (hh:mm) or a range of dates with optional times or a selected range of headings.

## Suffixes

---

[Table 24-2](#) through [Table 24-11](#) cover the various suffix segments of the cataloging reports.

### Unauthorized Subject Headings (00)

[Table 24-2](#) describes the unauthorized subject headings report suffix.

**Table 24-2. Unauthorized Subject Headings Report Suffix (00)**

Item #	Required	Item Name	Length	Description
4	Y	heading 1	255	First 255 characters of heading.
5		heading 2	255	Last 45 characters of heading.
6	Y	thesaurus	50	Thesaurus.
7	Y	date heading added	10	Date heading added.
8	Y	opac bib count	15	Opac bib count.

### Unauthorized Name Headings (01)

[Table 24-3](#) describes the unauthorized name headings report suffix.

**Table 24-3. Unauthorized Name Report Suffix (01)**

Item #	Required	Item Name	Length	Description
4	Y	heading 1	255	First 255 characters of heading.
5		heading 2	255	Last 45 characters of heading.
6	Y	name type	50	Name type.
7	Y	date heading added	10	Date heading added.
8	Y	opac bib count	15	Opac bib count.

### Unauthorized Title Headings (02)

[Table 24-4](#) describes the unauthorized title headings report suffix.

**Table 24-4. Unauthorized Title Headings Report Suffix (02)**

Item #	Required	Item Name	Length	Description
4	Y	heading 1	255	First 255 characters of heading.
5		heading 2	255	Last 45 characters of heading.
6	Y	title type	50	Title type.
7	Y	date heading added	10	Date heading added.
8	Y	opac bib count	15	Opac bib count.

**Unauthorized Name/Title Headings (03)**

[Table 24-5](#) describes the unauthorized name/title headings report suffix.

**Table 24-5. Unauthorized Name/Title Headings Report Suffix (03)**

Item #	Required	Item Name	Length	Description
4	Y	heading 1	255	First 255 characters of heading.
5		heading 2	255	Last 45 characters of heading.
6	Y	name/title type	50	Name/title type.
7	Y	date heading added	10	Date heading added.
8	Y	opac bib count	15	Opac bib count.

**Unauthorized Subdivision Headings (04)**

[Table 24-6](#) describes the unauthorized subdivision headings report suffix.

**Table 24-6. Unauthorized Subdivision Headings Report Suffix (04)**

Item #	Required	Item Name	Length	Description
4	Y	heading 1	255	First 255 characters of heading.
5		heading 2	255	Last 45 characters of heading.
6	Y	thesaurus	50	Thesaurus.
7	Y	date heading added	10	Date heading added.
8	Y	opac bib count	15	Opac bib count.

**Duplicate Authority Records (05)**

[Table 24-7](#) describes the duplicate authority records report suffix.

**Table 24-7. Duplicate Authority Records Report Suffix (05)**

Item #	Required	Item Name	Length	Description
4	Y	heading type	50	Heading type (name, title).
5	Y	heading 1	255	First 255 characters of heading.
6		heading 2	255	Last 45 characters of heading.
7	Y	normalized heading 1	255	First 255 characters of normalized heading.

**Table 24-7. Duplicate Authority Records Report Suffix (05)**

Item #	Required	Item Name	Length	Description
8		normalized heading 2	255	Last 45 characters of normalized heading.
9	Y	authority record id	100	Authority record ID.

**'See' References with Linked Bib Records (06)**

[Table 24-8](#) describes the see references with linked bib records report suffix.

**Table 24-8. 'See' References with Linked Bib Records Report Suffix (06)**

Item #	Required	Item Name	Length	Description
4	Y	heading type	50	Heading type (name, title).
5	Y	heading 1	255	First 255 characters of heading.
6		heading 2	255	Last 45 characters of heading.
7	Y	normalized heading 1	255	First 255 characters of normalized heading.
8		normalized heading 2	255	Last 45 characters of normalized heading.
9	Y	4xx authority record id	100	4xx authority record ID.
10	Y	opac bib count	15	Opac bib count.

**'See' References Authorized in Another Authority Record (07)**

[Table 24-9](#) describes the see references authorized in another authority record report suffix.

**Table 24-9. 'See' References Authorized in Another Authority Record Suffix (07)**

Item #	Required	Item Name	Length	Description
4	Y	heading type	50	Heading type (name, title).
5	Y	heading 1	255	First 255 characters of heading.
6		heading 2	255	Last 45 characters of heading.
7	Y	normalized heading 1	255	First 255 characters of normalized heading.
8		normalized heading 2	255	Last 45 characters of normalized heading.

**Table 24-9. ‘See’ References Authorized in Another Authority Record Suffix (07)**

Item #	Required	Item Name	Length	Description
9	Y	4xx authority record id	100	4xx authority record ID.
10	Y	1xx authority record id	100	1xx authority record ID.
11	Y	opac bib count	15	Opac bib count.

### ‘See’ References without Corresponding Authority Record (08)

[Table 24-10](#) describes the see references without corresponding authority record report suffix.

**Table 24-10. ‘See’ References without Corresponding Authority Record Report Suffix (08)**

Item #	Required	Item Name	Length	Description
4	Y	heading type	50	Heading type (name, title).
5	Y	heading 1	255	First 255 characters of heading.
6		heading 2	255	Last 45 characters of heading.
7	Y	normalized heading 1	255	First 255 characters of normalized heading.
8		normalized heading 2	255	Last 45 characters of normalized heading.
9	Y	5xx authority record id	100	5xx authority record ID.
10	Y	opac bib count	15	Opac bib count.

### 856 Link Failure Report Suffix (09)

[Table 24-11](#) describes the 856 link failure report suffix.

**Table 24-11. 856 Link Failure Report Suffix (09)**

Item #	Required	Item Name	Length	Description
4	Y	link_host	40	Link host.
5	Y	link	255	Link (URL).
6	Y	link type	3	Link type.
7	Y	error code	10	Error (status) code.
8	Y	error description	255	Error description.

**Table 24-11. 856 Link Failure Report Suffix (09)**

Item #	Required	Item Name	Length	Description
9	Y	record type	25	Link record type (bibliographic, holdings, or e-item).
10	Y	record id	25	Record ID.
11	Y	update date	10	Date last updated.



---

## **Circulation Notices Standard Interface File**

# **25**

---

<b>Introduction</b>	<b>26-1</b>
<b>Purpose of This Chapter</b>	<b>26-1</b>
<b>File Specification</b>	<b>26-2</b>
<b>Circulation Notices SIF Format</b>	<b>26-3</b>
• Base Segment	26-4
• Suffixes	26-5
Cancellation Notice Suffix (00)	26-6
Item Available Notice Suffix (01)	26-6
Overdue Notice Suffix (02)	26-6
Recall Notice Suffix (03)	26-7
Recall-Overdue Notice Suffix (04)	26-7
Fine/Fee Notice Suffix (05)	26-7
Statement of Fines and Fees Suffix (06)	26-8
Courtesy (Due) Notice Suffix (07)	26-9



---

## Circulation Notices Standard Interface File

---

# 25

### **Introduction**

---

The Circulations Notices Standard Interface File (SIF) is the file produced by activity at circulation locations and then running circulation batch jobs on the server.

Running the circulation batch jobs produces a file containing the notice information in a specific format. This file is then used as an input file ( .inp ) for the Reporter module to use to generate the notices. See [Creating Input Files for Circulation](#) in the *Voyager Reporter User's Guide* for more information.

### **Purpose of This Chapter**

---

This section dicusses the layout for the Circulation Notices SIF.

- File specification
- File format base segment
- File format suffix segment

## File Specification

---

The name of the file produced is `crcnotes.[print location code].inp`, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the *Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called `crcnotes.main.inp`. When the file is generated, it is placed in the `/m1/voyager/xxxdb/rpt` directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the circulation notices SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of notice for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample circulation notice record might appear as follows:

```
02|97.2||28|Smith|John ||8974 W.117th Street||||Kansas  
City|MO|76763||(290) 932-0371|10/15/1999|Ex Libris  
Library of Information>Main|2200 E. Devon|Suite  
382|Des Plaines|IL|60018|USA||Motor carriers' road  
atlas, 1998 : United States, Canada, Mexico / Rand  
McNally.|Rand McNally and Company.|12481632||10/13/  
1999|1||
```

The first 33 fields of the record are the base segment. Because this is an overdue notice record, the suffix is made up of the last five fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of notice. The notice numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## Circulation Notices SIF Format

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Required** If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description** Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Cancellation Notice = 33 fields.
- Type 01: Item Available Notice = 34 fields.
- Type 02: Overdue Notice = 38 fields.
- Type 03: Recall Notice = 37 fields.
- Type 04: Recall-Overdue Notice = 38 fields.
- Type 05: Fine/Fee Notice = 39 fields.
- Type 06: Statement of Fines/Fees = 39 fields.
- Type 07: Courtesy (due) Notice = 37 fields.

## Base Segment

---

[Table 25-1](#) defines the base segment for all circulation notice type records.

**Table 25-1. Base segment for circulation notices**

Item #	Required	Item Name	Length	Description
1	Y	notice id	2	This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice: 00 is a cancellation notice 01 is an item available notice 02 is an overdue notice 03 is a recall notice 04 is a recall overdue notice 05 is a fine/fee notice 06 is a statement of fines/fees 07 is a courtesy (due) notice
2	Y	version number	30	Program version number to be used by client for version checking.
3		e-mail address	50	Patron's e-mail address. If this field is not blank, the notice will be sent by e-mail. If it is blank, the notice will be printed for mailing.
4	Y	patron id	10	Patron's ID number.
5	Y	last name	30	Patron's last name.
6		first name	20	Patron's first name.
7		title	20	Patron's title.
8		address line 1	50	Patron's address line 1.
9		address line 2	40	Patron's address line 2.
10		address line 3	40	Patron's address line 3.
11		address line 4	40	Patron's address line 4.
12		address line 5	40	Patron's address line 5.
13		city	30	Patron's city.
14		state/province	7	Patron's state/province.
15		postal code	10	Patron's postal code.

**Table 25-1. Base segment for circulation notices**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
16		country	20	Patron's country.
17		phone	25	Patron's phone.
18	Y	date	10	Current date.
19	Y	institution	50	Institution name.
20		library	25	Library location name.
21		address line 1	50	Library's address line 1.
22		address line 2	40	Library's address line 2.
23		address line 3	40	Library's address line 3.
24		city	30	Library's city.
25		state/province	7	Library's state/province.
26		postal code	10	Library's postal code.
27		country	20	Library's country.
28		phone	25	Library's phone.
29		item title	255	Item title.
30		item author	255	Item author.
31		item id	25	Item barcode.
32		item call #	255	Item call number.
33		enum/chron	255	Enum/chron. This is the last common item and the <u>last item in the record</u> for circulation notices.

### Suffixes

---

[Table 25-2](#) through [Table 25-9](#) cover the various suffix segments of the circulation notices.

**Cancellation Notice Suffix (00)**

[Table 25-2](#) describes the cancellation notice suffix.

**Table 25-2. Cancellation Notice Suffix (00)**

Item #	Required	Item Name	Length	Description
--		note	--	All items are included in the common portion of record. There is nothing unique for cancellation notices.

**Item Available Notice Suffix (01)**

[Table 25-3](#) describes the item available notice suffix.

**Table 25-3. Item Available Notice Suffix (01)**

Item #	Required	Item Name	Length	Description
34		expiration date	10	Expiration date. This is the <u>last item in the record</u> for item available notices.

**Overdue Notice Suffix (02)**

[Table 25-4](#) describes the overdue notice suffix.

**Table 25-4. Overdue Notice Suffix (02)**

Item #	Required	Item Name	Length	Description
34	Y	due date	10	Due date.
35		sequence	2	Notice sequence number.
36		proxy patron last name	30	Proxy patron patron's last name.
37		proxy patron first name	20	Proxy patron patron's first name.
38		proxy patron title	20	Proxy patron's title. This is the <u>last item in the record</u> for overdue notices.

### Recall Notice Suffix (03)

[Table 25-5](#) describes the recall notice suffix.

**Table 25-5. Recall Notice Suffix (03)**

Item #	Required	Item Name	Length	Description
34	Y	due date	10	Due date.
35		proxy patron last name	30	Proxy patron patron's last name.
36		proxy patron first name	20	Proxy patron patron's first name.
37		proxy patron title	20	Proxy patron's title.

### Recall-Overdue Notice Suffix (04)

[Table 25-6](#) describes the recall-overdue notice suffix.

**Table 25-6. Recall-Overdue Notice Suffix (04)**

Item #	Required	Item Name	Length	Description
34	Y	due date	10	Due date.
35		sequence	2	Notice sequence number.
36		proxy patron last name	30	Proxy patron patron's last name.
37		proxy patron first name	20	Proxy patron patron's first name.
38		proxy patron title	20	Proxy patron's title. This is the <u>last item in the record</u> for recall-overdue notices.

### Fine/Fee Notice Suffix (05)

[Table 25-7](#) describes the fine/fee notice suffix.

**Table 25-7. Fine/Fee Notice Suffix (05)**

Item #	Required	Item Name	Length	Description
34	Y	fine/fee date	10	Date of this fine/fee.
35	Y	fine/fee description	25	This is the description of a fine/fee due.

**Table 25-7. Fine/Fee Notice Suffix (05)**

Item #	Required	Item Name	Length	Description
36		fine/fee amount	10	Total amount of fine/fee described above.
37		fine/fee balance	10	Net amount of fine/fee described above.
38		previously billed	10	Previously billed amounts total.
39		total fines/fees	10	Total of all fines/fees. This is the <u>last item in the record</u> for notice of fines and fees.

**Statement of Fines and Fees Suffix (06)**

[Table 25-8](#) describes the statement of fines and fees notice suffix.

**Table 25-8. Statement of Fines and Fees Suffix (06)**

Item #	Required	Item Name	Length	Description
34	Y	fine/fee date	10	Date of this fine/fee.
35	Y	fine/fee description	25	This is the description of a fine/fee due.
36		fine/fee amount	10	Total amount of fine/fee described above.
37		fine/fee balance	10	Net amount of fine/fee described above.
38		previously billed	10	Previously billed amounts total.
39		total fines/fees	10	Total of all fines/fees. This is the <u>last item in the record</u> for statement of fines and fees.

**Courtesy (Due) Notice Suffix (07)**

[Table 25-9](#) describes the courtesy (due) notice suffix.

**Table 25-9. Courtesy (Due) Notice Suffix (07)**

Item #	Required	Item Name	Length	Description
34	Y	due date	10	Due date.
35		proxy patron last name	30	Proxy patron's last name.
36		proxy patron first name	20	Proxy patron's first name.
37		proxy patron title	20	Proxy patron's title. This is the <u>last item in the record</u> for courtesy (due) notices.



---

## **Circulation Reports Standard Interface File**

# **26**

---

<b>Introduction</b>	<b>27-1</b>
<b>Purpose of This Chapter</b>	<b>27-1</b>
<b>File Specification</b>	<b>27-1</b>
<b>Circulation Reports SIF Format</b>	<b>27-2</b>
• Base Segment	27-4
• Suffixes	27-4
Reserved Items Active Report (00)	27-5
Reserved Items Expired Report (01)	27-5
Hold Shelf Expired Report (02)	27-5
Missing in Transit Report (03)	27-6
Circulation Statistics Report (04)	27-6
Circulation Item-Related Exceptions Report (05)	27-7
Circulation Patron-Related Exceptions Report (06)	27-7
Circulation Transaction-Related Exceptions Report (07)	27-7
Global Circulation Statistics Report (08)	27-8
Distribution Item Order List Report (09)	27-8



---

## Circulation Reports Standard Interface File

# 26

---

### Introduction

---

The Circulation Reports Standard Interface File (SIF) is the file produced by running circulation batch jobs on the server.

Running the circulation batch jobs produces a file containing the report information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate the reports. See [Creating Input Files for Circulation](#) in the *Voyager Reporter User's Guide* for more information.

---

### Purpose of This Chapter

---

This section discusses the layout for the Circulation Report SIF.

- File specification
- File format base segment
- File format suffix segment

---

### File Specification

---

The name of the file produced is `crcrprts.[print location code].inp`, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the

*Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called `crcrprts.main.inp`. When the file is generated, it is placed in the `/m1/voyager/xxxdb/rpt` directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the circulation reports SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of report for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample circulation report record might appear as follows:

```
03|97.2|09/16/1999|TL230.A523|America's light  
trucks.|39550000173749|main|07/30/1999
```

The first three fields of the record are the base segment. Because this is a Missing in Transit report record, the suffix is made up of the last six fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of report. The report numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## Circulation Reports SIF Format

---

The following conventions are used in the listed tables.

<b>Item #</b>	Represents the relative position of the item in the record (sequence).
---------------	--

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name**      Definition of the item.

**Required**      If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description**      Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Reserved Items Active Report = 9 fields.
- Type 01: Reserved Items Expired Report = 9 fields.
- Type 02: Hold Shelf Expired Report = 11 fields.
- Type 03: Missing in Transit Report = 8 fields.
- Type 04: Circ Transactions Statistics Rpt = 9 fields.
- Type 05: Circ Item-related Exceptions Rpt = 9 fields
- Type 06: Circ Patron-related Exceptions Rpt = 9 fields.
- Type 07: Circ Transactn-related Exceptions Rpt = 11 fields.
- Type 08: Global Circ Transactions Statistics = 9 fields.
- Type 09: Distribution Item Order Rpt = 22 fields.

## Base Segment

---

[Table 26-1](#) defines the base segment for all circulation report type records.

**Table 26-1. Base segment for circulation reports**

Item #	Required	Item Name	Length	Description
1	Y	report id	2	<p>This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice:</p> <ul style="list-style-type: none"> <li>00 is a reserved items active report.</li> <li>01 is a reserved items expired report.</li> <li>02 is a hold shelf expired report.</li> <li>03 is a missing in transit report.</li> <li>04 is a circ transaction statistics report.</li> <li>05 is a circ item-related override/exception.</li> <li>06 is a circ patron-related override/exception.</li> <li>07 is a circ transaction-related override/exception.</li> <li>08 is a global circ transaction statistics report.</li> <li>09 is a distribution item order list</li> </ul>
2	Y	version number	30	Program version number to be used by client for version checking.
3		date[time]/selection (range)	50	Date (mm/dd/ccyy) with optional time (hh:mm) or a range of dates with optional times. Only relevant to reserved items (reports 00 and 01) and circ transaction stats (reports 04 and 08).

## Suffixes

---

[Table 26-2](#) through [Table 26-11](#) cover the various suffix segments of the circulation reports.

**Reserved Items Active Report (00)**

[Table 26-2](#) describes the reserved items active report suffix.

**Table 26-2. Reserved Items Active Report Suffix (00)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Location code.
5		call number	144	Call number.
6	Y	title	100	Title.
7		item barcode	25	Item barcode.
8	Y	reserve list name	40	Reserve list name.
9	Y	effective date	10	Effective date.

**Reserved Items Expired Report (01)**

[Table 26-3](#) describes the reserved items expired report suffix.

**Table 26-3. Reserved Items Expired Report Suffix (01)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Location code.
5		call number	255	Call number.
6	Y	title	100	Title.
7		item barcode	25	Item barcode.
8	Y	reserve list name	40	Reserve list name.
9	Y	expiration date	10	Expiration date.

**Hold Shelf Expired Report (02)**

[Table 26-4](#) describes the hold shelf expired report suffix.

**Table 26-4. Hold Shelf Expired Report Suffix (02)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Location code.
5		call number	255	Call number.
6	Y	title	100	Title.

**Table 26-4. Hold Shelf Expired Report Suffix (02)**

Item #	Required	Item Name	Length	Description
7		item barcode	25	Item barcode.
8	Y	expiration date	10	Date reserve expires.
9	Y	patron's first name	20	Patron's first name (Patron.first_name).
10	Y	patron's middle name	20	Patron's middle name (Patron.middle_name).
11	Y	patron's last name	20	Patron's last name (Patron.last_name).

**Missing in Transit Report (03)**

[Table 26-5](#) describes the missing in transit report suffix.

**Table 26-5. Missing in Transit Report Suffix (03)**

Item #	Required	Item Name	Length	Description
4		source location	10	Location code where item is coming from.
5		call number	255	Call number.
6	Y	title	255	Title.
7		item barcode	25	Item barcode.
8		target location	10	Location code where item is going to.
9	Y	transit date	10	Transit date.

**Circulation Statistics Report (04)**

[Table 26-6](#) describes the circulation statistics report suffix.

**Table 26-6. Circulation Statistics Report Suffix (04)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Circulation location code.
5	Y	operator id	10	Operator ID.
6		charges	10	Number of charges.
7		discharges	10	Number of discharges.
8		renewals	10	Number of renewals.

**Table 26-6. Circulation Statistics Report Suffix (04)**

9		fines	12	Amount of fines collected.
---	--	-------	----	----------------------------

**Circulation Item-Related Exceptions Report (05)**

[Table 26-7](#) describes the circulation item-related exceptions report suffix.

**Table 26-7. Circulation Item--related Exceptions Report Suffix (05)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Home location code.
5		description	50	Exception description.
6	Y	title	255	Title.
7		item barcode	25	Item barcode.
8	Y	date	10	Exception date.
9	Y	operator id	10	Operator ID.

**Circulation Patron-Related Exceptions Report (06)**

[Table 26-8](#) describes the circulation patron-related exceptions report suffix.

**Table 26-8. Circulation Patron-related Exceptions Report Suffix (06)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Home location code.
5	Y	description	50	Exception description.
6	Y	patron name	50	Patron name.
7		patron id	25	Patron barcode.
8	Y	date	10	Exception date.
9	Y	operator id	10	Operator ID.

**Circulation Transaction-Related Exceptions Report (07)**

[Table 26-9](#) describes the circulation transaction-related exceptions report suffix.

**Table 26-9. Circulation Transaction-related Exceptions Report Suffix (07)**

Item #	Required	Item Name	Length	Description
--------	----------	-----------	--------	-------------

**Table 26-9. Circulation Transaction-related Exceptions Report Suffix (07)**

4	Y	location	10	Home location code.
5	Y	description	50	Exception description.
6	Y	patron name	50	Patron name.
Item #	Required	Item Name	Length	Description
7		patron id	25	Patron barcode.
8	Y	title	255	Title.
9		item barcode	25	Item barcode.
10	Y	date	10	Exception date.
11	Y	operator id	10	Operator ID.

**Global Circulation Statistics Report (08)**

[Table 26-10](#) describes the global circulation statistics report suffix.

**Table 26-10. Global Circulation Statistics Report Suffix (08)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Circulation location code.
5	Y	operator id	10	Operator ID.
6		charges	10	Number of charges.
7		discharges	10	Number of discharges.
8		renewals	10	Number of renewals.
9		fines	12	Amount of fines collected.

**NOTE:**

Report 08, Global Circ Transactions Statistics, is a duplicate of report 04 except that it includes all locations while all other reports only include the happening locations assigned to the particular Print Location which is issuing the report.

**Distribution Item Order List Report (09)**

[Table 26-11](#) describes the distribution order list report suffix.

**Table 26-11. Distribution Item Order List Report Suffix (09)**

Item #	Required	Item Name	Length	Description
4	Y	vendor code	25	Vendor code.

**Table 26-11. Distribution Item Order List Report Suffix (09)**

5	Y	vendor_name	60	Vendor's name.
6		address line 1	50	Vendor's address line 1.
7		address line 2	50	Vendor's address line 2.
Item #	Required	Item Name	Length	Description
8		address line 3	50	Vendor's address line 3.
9		address line 4	50	Vendor's address line 4.
10		city	30	Vendor's city.
11		state/province	7	Vendor's state/province.
12		postal code	10	Vendor's postal code.
13		country	20	Vendor's country.
14	Y	item title	255	Item title.
15		item author	255	Item author.
16	Y	item barcode	25	Item barcode.
17		item call #	255	Item call number.
18		enum/chron	255	Enum/chron.
19	Y	order date	10	Order date.
20	Y	quantity	10	Number of items to order.
21	Y	expected date	10	Expected date.
22	Y	operator	10	Operator ID. This is the last common item and the <u>last item in the record</u> for distribution item order list report.



---

## **Media Scheduling Notices Standard Interface File**

# **27**

---

<b>Introduction</b>	<b>28-1</b>
<b>Purpose of This Chapter</b>	<b>28-1</b>
<b>File Specification</b>	<b>28-2</b>
<b>Media Scheduling Notices SIF Format</b>	<b>28-3</b>
• Base Segment	28-3
• Suffixes	28-4
Overdue Notice Suffix (00)	28-5
	28-5



---

## **Media Scheduling Notices Standard Interface File**

---

# **27**

---

### **Introduction**

---

The Media Scheduling Notices Standard Interface File (SIF) is the file produced by activity at media scheduling locations and then running media scheduling batch jobs on the server.

Running the media scheduling batch job produces a file containing the notice information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate the notice. See [Creating Input Files for Media Scheduling](#) in the *Voyager Reporter User's Guide* for more information.

---

### **Purpose of This Chapter**

---

This section discusses the layout for the Media Scheduling Notices SIF.

- File specification
- File format base segment
- File format suffix segment

## File Specification

---

The name of the file produced is `mednotes.[print location code].inp`, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the *Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called `mednotes.main.inp`. When the file is generated, it is placed in the `/m1/voyager/xxxdb/rpt` directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the media scheduling notices SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of notice for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample media scheduling notice record might appear as follows:

```
00|97.2|mayer@endinfosys.com|8548742|Mayer|Henry||123  
8th Drive #843||||Chicago|IL|60000||(847) 296-2200|10/  
19/99| Institution of Laughing Person>Main  
Library||||||(847)296-2200 x 4621|Women  
jogging.|John K. Skrupnid|87943243|||||Random House  
Video/Media|||5t5698|09/02/99 19:54|09/02/99 22:00|9
```

The first 28 fields of the record are the base segment. Because this is a overdue notice record, the suffix is made up of the last 16 fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of notice. The notice numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

**NOTE:**

This SIF remains in the Latin-1 character set. If elements in the SIF were stored in the database using the Unicode character set, they are converted to Latin-1 before being exported to the SIF.

## Media Scheduling Notices SIF Format

---

The following conventions are used in the listed tables.

**Item #** Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name** Definition of the item.

**Required** If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description** Description of the item.

The total number of fields in the notice record is as follows.

- Type 00: Overdue Notice = 44 fields.

### Base Segment

---

[Table 27-1](#) defines the base segment for the media scheduling notice type records.

**Table 27-1. Base Segment for Media Scheduling Notices**

Item #	Required	Item Name	Length	Description
1	Y	notice id	2	This field must be 2 characters in length and contain the following code indicating the type of notice: 00 is an overdue notice.
2	Y	version number	30	Program version number to be used by client for version checking.
3		e-mail address	50	Patron's e-mail address. If this field is not blank, the notice will be sent by e-mail. If it is blank, the notice will be printed for mailing.
4	Y	patron id	10	Patron's id number.
5	Y	last name	30	Patron's last name.

**Table 27-1. Base Segment for Media Scheduling Notices**

<b>Item #</b>	<b>Required</b>	<b>Item Name</b>	<b>Length</b>	<b>Description</b>
6		first name	20	Patron's first name.
7		title	20	Patron's title.
8		address line 1	50	Patron's address line 1.
9		address line 2	40	Patron's address line 2.
10		address line 3	40	Patron's address line 3.
11		address line 4	40	Patron's address line 4.
12		address line 5	40	Patron's address line 5.
13		city	30	Patron's city.
14		state/province	7	Patron's state/province.
15		postal code	10	Patron's postal code.
16		country	20	Patron's country.
17		phone	25	Patron's phone.
18	Y	date	10	Current date.
19	Y	institution	50	Institution name.
20		library	25	Library location name.
21		address line 1	50	Library address line 1.
22		address line 2	40	Library address line 2.
23		address line 3	40	Library address line 3.
24		city	30	Library city.
25		state/province	7	Library state/province.
26		postal code	10	Library postal code.
27		country	20	Library country.
28		phone	25	Library phone.

**Suffixes**

[Table 27-2](#) covers the suffix segment of the media scheduling notice.

## Overdue Notice Suffix (00)

[Table 27-2](#) describes the overdue notice suffix.

**Table 27-2. Overdue Notice Suffix (00)**

Item #	Required	Item Name	Length	Description
29		item title	255	Item title.
30		item author	255	Item author.
31		item id	25	Item barcode.
32		item call #	255	Item call number.
33		enum/chron	255	Enum/chron.
34		equipment type	255	Equipment type.
35		equipment no	15	Unit/group ID.
36		equipment id	25	Equipment ID.
37		barcode	25	Unit/group barcode.
38		manufacturer	100	Equipment manufacturer.
39		model	100	Equipment model designation.
40		serial number	100	Equipment serial number.
41	Y	confirmation no.	10	Booking confirmation number.
42		booking start	25	Booking end date/time.
43		booking end	25	Booking end date/time.
44		sequence	2	Notice sequence number. This is the <u>last item in the record</u> for overdue notices.



---

## **Media Scheduling Reports Standard Interface File**

# **28**

---

<b>Introduction</b>	<b>29-1</b>
<b>Purpose of This Chapter</b>	<b>29-1</b>
<b>File Specification</b>	<b>29-1</b>
<b>Media Scheduling Reports SIF Format</b>	<b>29-2</b>
• Base Segment	29-3
• Suffixes	29-3
Media Equipment Inventory Report (00)	29-4
Booking Statistics Report (01)	29-4
Booking Exceptions Report (02)	29-5
Booking Charge Statistics Report (03)	29-5



---

### Introduction

---

The Media Scheduling Reports Standard Interface File (SIF) is the file produced by running media scheduling batch jobs on the server.

Running media scheduling batch jobs produce a file containing the report information in a specific format. This file is then used as an input file (.inp) for the Reporter module to use to generate the reports. See [Creating Input Files for Media Scheduling](#) in the *Voyager Reporter User's Guide* for more information.

---

### Purpose of This Chapter

---

This section discusses the layout for the Media Scheduling Report SIF.

- File specification
- File format base segment
- File format suffix segment

---

### File Specification

---

The name of the file produced is medrprts.[print location code].inp, where xxxx represents the appropriate printing location code as defined in "Print Locations" in the System Administration module. See [Print Locations](#) in the

*Voyager System Administration User's Guide* for more information. If the print location code is main, the file will be called `medrpts.main.inp`. When the file is generated, it is placed in the `/m1/voyager/xxxdb/rpt` directory.

Each record in the SIF is made up of two pieces: a base segment and a suffix.

The base segment is used in every record in the media scheduling reports SIF and is always at the beginning of the record. After the base segment, one of the suffixes will be attached. Depending on the type of report for which the record contains data, a different suffix will be attached. A record will only contain one base segment and one suffix.

For example, a sample media scheduling report record might appear as follows:

```
00|97.2||TEST|Television|media|101|Classrooms|MED101|7|
|2|TV002|122||0|Charged||||
```

The first three fields of the record are the base segment. Because this is a media equipment inventory report record, the suffix is made up of the last 18 fields. The type of suffix attached determines the record type.

The record type can be determined by examining the first field of the record. The two-digit number indicates the type of report. The report numbers are listed in the heading for each suffix type.

All fields must be present and pipe-delimited. The pipe for a given field must be present even if the field is empty.

Each record will be a line-feed-terminated string.

Maximum field lengths as imposed below must be adhered to.

## **Media Scheduling Reports SIF Format**

---

The following conventions are used in the listed tables.

**Item #**              Represents the relative position of the item in the record (sequence).

**NOTE:**

In all the following tables, the first column, Item #, consists of the field sequence numbers which are not to be included in the record.

**Item name**      Definition of the item.

**Required** If Y is entered, the field must have valid data in it and cannot be blank. If nothing is entered, the field is not required.

**Description** Description of the item.

The total number of fields varies with record type as follows.

- Type 00: Media Equipment Inventory Report = 21 fields.
- Type 01: Media Scheduling Statistics Report = 8 fields.
- Type 02: Media Scheduling Exceptions Report = 9 fields.
- Type 03: Media Scheduling Charge Statistics Report = 14 fields.

## Base Segment

[Table 28-1](#) defines the base segment for all media scheduling report type records.

**Table 28-1. Base Segment for Media Scheduling Reports**

Item #	Required	Item Name	Length	Description
1	Y	report id	2	This field must be 2 characters in length and contain 1 of the following codes indicating the type of notice: 00 is a media equipment inventory report. 01 is a booking statistics report. 02 is a booking exceptions report. 03 is a booking charge statistics report.
2	Y	version number	30	Program version number to be used by client for version checking.
3		date[time]/selection (range)	50	Date (mm/dd/ccyy) with optional time (hh:mm) or a range of dates with optional times.

## Suffixes

[Table 28-2](#) through [Table 28-5](#) cover the various suffix segments of the media scheduling reports.

## Media Equipment Inventory Report (00)

[Table 28-2](#) describes the media equipment inventory report suffix.

**Table 28-2. Media Equipment Inventory Report Suffix (00)**

Item #	Required	Item Name	Length	Description
4	Y	policy group	50	Policy group.
5	Y	equipment type	50	Equipment type.
6	Y	location name	25	Equipment location name.
7	Y	room number	15	Equipment room number.
8	Y	room type	50	Equipment room type code.
9		room name	100	Equipment room name.
10		group id	25	Equipment group ID.
11		group number	15	Equipment group number.
12	Y	equipment id	25	Equipment ID.
13	Y	equipment num-ber	15	Equipment number.
14		equipment bar-code	25	Equipment barcode.
15		acquisition date	10	Acquisition date.
16	Y	equipment value	12	Equipment value.
17	Y	equipment status	100	Equipment status.
18		manufacturer	100	Equipment manufacturer.
19		model	100	Model number/description.
20		equipment serial no.	100	Equipment serial number.
21		last maintenance date	10	Date of last maintenance.

## Booking Statistics Report (01)

[Table 28-3](#) describes the booking statistics report suffix.

**Table 28-3. Booking Statistics Report Suffix (01)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Media Scheduling location code.

**Table 28-3. Booking Statistics Report Suffix (01)**

5	Y	operator id	10	Operator ID.
6	Y	bookings made	10	Number of bookings made.
7	Y	bookings charged	10	Number of charges.
8	Y	bookings discharged	10	Number of discharges.
9	Y	bookings canceled	10	Number of cancellations.

**Booking Exceptions Report (02)**

[Table 28-4](#) describes the booking exceptions report suffix.

**Table 28-4. Booking Exceptions Report Suffix (02)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Home location code.
5	Y	description	25	Exception description.
6	Y	equipment type	255	Equipment type.
7		equipment bar-code	25	Equipment barcode.
8	Y	date	10	Exception date.
9	Y	operator id	10	Operator ID.

**Booking Charge Statistics Report (03)**

[Table 28-5](#) describes the booking charge statistics report suffix.

**Table 28-5. Booking Charge Statistics Report Suffix (03)**

Item #	Required	Item Name	Length	Description
4	Y	location	10	Media scheduling location code.
5	Y	operator id	10	Operator ID.
6	Y	staff delivery	10	Number of staff deliveries.
7	Y	delivery/return items	10	Number of items delivered.

**Table 28-5. Booking Charge Statistics Report Suffix (03)**

8		delivery/return equipment	10	Number of equipment items delivered.
9	Y	patron picked-up	10	Number of patron pick-ups.
10	Y	pickup items	10	Number of items picked-up.
11	Y	pickup equip	10	Number of equipment items picked-up.
12	Y	schedule room	10	Number of rooms scheduled.
13	Y	library items	10	Number of library items.
14	Y	library equip	10	Number of library equipment items.

<b>Introduction</b>	<b>30-1</b>
<b>Purpose of This Chapter</b>	<b>30-2</b>
<b>Views</b>	<b>30-2</b>
• Authblob_vw	30-2
• Authhistory_vw	30-3
• Authheading_vw	30-3
• Authority1xx4xx_vw	30-4
• Authority5xx1xx_vw	30-4
• Authoritydupe_vw	30-5
• Bib_vw	30-5
• Bibblob_vw	30-7
• Bibhistory_vw	30-7
• Bibloc_vw	30-8
• Circcharges_vw	30-8
• Circrenewal_vw	30-10
• Fundledger_vw	30-12
• Heading_vw	30-14
• Issues_vw	30-15
• Item_vw	30-16
• LCclass_vw	30-18
• Marccomputer_vw	30-19
• Marcbook_vw	30-19
• Marcmap_vw	30-20

• Marcmusic_vw	30-20
• Marcserial_vw	30-21
• Marcvisual_vw	30-22
• MFHDblob_vw	30-22
• MFHDhistory_vw	30-23
• NLMclass_vw	30-23
• Recordcount_vw	30-24
• Serials_vw	30-24
• Sudocclass_vw	30-25
• Vendorinvoice_vw	30-26
• Vendororder_vw	30-28

### Introduction

---

The Voyager database views provide you with a simplified way of retrieving data from the database. A view is like a table that, instead of holding new information, collects specific related data from a number of sources in the database.

A view, however, does not actually copy the information to a second location when it is created. Views are a part of SQL - each view is an SQL script. Each view contains an SQL query that draws the appropriate information from the database.

You can view (query) different sections of different tables at the same time. When you enter an SQL query on the view name, it finds the description of the view and returns with the table results.

Using SQL Plus, the following is an example of what you might enter to display the results of a *Authheading\_vw*.

Enter: *select \* from Authheading\_vw*

These commands will be different depending on what SQL editor you use.

## Purpose of This Chapter

This chapter includes Views tables.

## Views

The following information is described about each table:

**Field Name** The name of the field as defined by the view.

**Description and Use** General information.

**Report** Whether this would be useful in a report.

**Query Select** Whether this would be useful as a criterion for querying based on this field.

**Normalized Sort** Whether the data in this field can be used to sort in some meaningful way.

**Link to Field** Whether this field would be helpful in linking this view to other tables in the database. An asterisk (\*) indicates that this field may be linked to a table in another database.

### Authblob\_vw

This view provides access to the entire MARC authority record (see [Table 29-1](#)).

**Table 29-1. Authblob\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Auth_id	Authority ID number	Y	Y		Y
Marc_record	Whole MARC authority record in its native format, contained in a single row	Y	N	N	N/A

**NOTE:**

Normally you will run MARC record parsing functions to access a particular field in the record.

## **Authhistory\_vw**

---

This view, provides information on the create and last update history of authority records (see [Table 29-2](#)).

**Table 29-2. Authhistory\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Auth_id	Authority ID number	Y	Y		Y
Create_operator_id	ID of operator who created the record	Y	Y	Y	Y
Create_date	Date and time of record creation	Y	Y	Y	
Create_location_id	ID of happening location where record was created		Y		Y
Update_operator_id	ID of operator that updated the record	Y	Y	Y	Y
Update_date	Date and time of record update	Y	Y	Y	
Update_location_id	ID of happening location where record was updated		Y		Y

## **Authheading\_vw**

---

This view is used by the heading\_vw. It may be of limited use to the end-user (see [Table 29-3](#)).

**Table 29-3. Authheading\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Heading_id_pointer	The heading pointed to by the reference				Y
Heading_id_pointee	The heading pointed from by the reference				Y
Auth_id	Authority record that established this reference				Y

**Table 29-3. Authheading\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Reference_type	Cross reference type	Y	Y	Y	

**Authority1xx4xx\_vw**

This view can be used to make a report for authority maintenance purposes (see [Table 29-4](#)).

**Table 29-4. Authority1xx4xx\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Index_type	The type of the heading index. (name, subject)	Y	Y	Y	
Auth_id_1XX	The authorized heading authority record ID	Y	Y	Y	Y
Auth_id_4XX	The see from heading authority record ID	Y	Y	Y	Y
Staffbibs	Count of all bibliographic records using this heading	Y	Y	Y	
Opacbibs	Count of non-suppressed bibliographic records using this heading	Y	Y	Y	Y
Display_heading	The heading, including subfield markers	Y	Y	Y	

**Authority5xx1xx\_vw**

This view can be used to make a report for authority maintenance purposes (see [Table 29-5](#)).

**Table 29-5. Authority5xx1xx\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Index_type	The type of the heading index. (name, subject)	Y	Y	Y	
Auth_id_5XX	The see-also from heading authority record ID	Y	Y	Y	Y

**Table 29-5. Authority5xx1xx\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Display_heading	The heading, including subfield markers	Y	Y	Y	

**Authoritydupe\_vw**

This view can be used to make a report for authority maintenance purposes. These records may be complete duplicate records or different authority records which may be authorizing the same heading for use in the same heading-index type (see [Table 29-6](#)).

**Table 29-6. Authoritydupe\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Auth_id	The authority record ID	Y	Y	Y	Y
Display_heading	The heading	Y	Y	Y	

**Bib\_vw**

This is a very powerful view that links the bibliographic record to the holdings record if one exists. This view will have a row for every MFHD in the database and every bib record that does not have a MFHD. It can be linked to the bib\_text or bib\_id for additional bibliographic information to display in a report (see [Table 29-5](#)).

**Table 29-7. Bib\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Title	Title Statement (245 a and b)	Y			
Sort_title	Normalized title for sorting a report by title non-filing words removed, punctuation removed		Y	Y	
Bib_id	The bibliographic record ID	Y	Y	Y	Y

**Table 29-7. Bib\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Create_date	The bibliographic record create date	Y	Y	Y	
Create_opid	The bibliographic record create operator	Y	Y	Y	Y
Mfhd_id	Mfhd_id attached to bib record. This field will be blank if no MFHDs are attached to the bib record. If more than one MFHD is attached to a bib record, there will be one row for each MFHD in the table	Y	Y	Y	Y
Call_no	The call number for the MFHD	Y			
Call_no_type	The MFHD indicator of call number type- Dewey, LC	Y	Y	Y	Y
Normalized_call_no	The normalized call number		Y	Y	
Mfhd_location_id	The internal location id of the MFHD				Y
Mfhd_location_code	The MFHD location code	Y	Y	Y	
Mfhd_location_name	The MFHD location name	Y		Y	
Mfhd_create_date	The MFHD record create date	Y	Y	Y	
Mfhd_create_opid	The MFHD record create operator	Y	Y	Y	Y

### **Bibblob\_vw**

---

This view provides access to the entire MARC bibliographic record (see [Table 29-8](#)).

**Table 29-8. Bibblob\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bibliographic ID number	Y	Y		Y
Marc_record	Whole MARC bibliographic record in its native format, contained in a single row	Y	N	N	N/A

**NOTE:**

Normally you will run MARC record parsing functions to access a particular field in the record.

### **Bibhistory\_vw**

---

This view provides information on the create and last update history of bibliographic records (see [Table 29-9](#)).

**Table 29-9. Bibhistory\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bibliographic ID number	Y	Y		Y
Create_operator_id	ID of operator who created the record	Y	Y	Y	Y
Create_date	Date and time of record creation	Y	Y	Y	
Create_location_id	ID of happening location where record was created		Y		Y
Update_operator_id	ID of operator that updated the record	Y	Y	Y	Y
Update_date	Date and time of record update	Y	Y	Y	

**Table 29-9. Bibhistory\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Update_location_id	ID of happening location where record was updated		Y		Y

**Bibloc\_vw**

This view extracts the coded location from the bib 008 field (see [Table 29-10](#)).

**Table 29-10. Bibloc\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib ID number	Y	Y		Y
Marcloccode	Three-character MARC location code	Y	Y	Y	*

**Circcharges\_vw**

This report brings together circulation statistics from current charges and archive charges (see [Table 29-11](#))

**Table 29-11. Circcharges\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Patron_group_id	The internal number of the patron group types				Y
Patron_group_code	The patron group code	Y	Y	Y	
Patron_group_name	The patron group name	Y		Y	
Item_id	The internal ID number of the item	Y	Y	Y	Y

**Table 29-11. Circcharges\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhid_id	Mfhid_id attached to bib record. This field will be blank if no MFHDs are attached to the bib record. If more than one MFHD is attached to a bib record, there will be one row for each MFHD in the table	Y	Y	Y	Y
Bib_id	Bib ID number	Y	Y		Y
Perm_location_code	The permanent location code of the item being charged	Y	Y	Y	
Perm_location	The permanent location of the item	Y		Y	
Gov_location_code	The temporary location code of the item if it exists or else the permanent location	Y	Y	Y	
Gov_location	The temporary location code of the item if it exists, or else the permanent location code	Y		Y	Y
Perm_item_type_code	The permanent item type code		Y		
Perm_item_type	The permanent item type	Y	Y	Y	
Gov_item_type_code	The temporary item type if it exists or else the permanent item type code		Y		
Gov_item_type	The temporary item type if it exists or else the permanent item type	Y	Y	Y	

**Table 29-11.** Circcharges\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Charge_date	Date item was charged	Y	Y	Y	
Charge_date_time	Date and time item was charged	Y	Y	Y	
Charge_date_only	Date item was charged	Y	Y	Y	
Charge_oper_id	Operator ID of the person who made the charge	Y	Y	Y	
Charge_location	ID of circulation desk where item was charged out				Y
Charge_location_code	Location code of circulation desk		Y		
Charge_location_name	Location name of circulation desk	Y		Y	
Renewal_count	Item renewal count	Y	Y	Y	
Notice_count	Number of notices sent including over-due recall notices	Y	Y	Y	

**Circrenewal\_vw**

This view brings together circulation statistics from charged (archived) items that have been renewed (see [Table 29-12](#)).

**Table 29-12.** Circrenewal\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Patron_group_id	The internal number of the patron group types				Y
Patron_group_code	The patron group code	Y	Y	Y	
Patron_group_name	The patron group name	Y		Y	
Item_id	The internal ID number of the item	Y	Y	Y	Y

**Table 29-12. Circrenewal\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhd_id	Mfhd_id attached to bib record. This field will be blank if no MFHDs are attached to the bib record. If more than one MFHD is attached to a bib record, there will be one row for each MFHD in the table				
Bib_id	Bib ID number	Y	Y		Y
Perm_location_code	The permanent location code of the item being charged	Y	Y	Y	
Perm_location	The permanent location of the item	Y		Y	
Gov_location_code	The temporary location code of the item if it exists or else the permanent location code	Y	Y	Y	Y
Gov_location	The temporary location of the item if it exists or else the permanent location	Y		Y	Y
Perm_item_type_code	The permanent item type code		Y		
Perm_item_type	The permanent item type	Y	Y	Y	
Gov_item_type_code	The temporary item type code if it exists or else the permanent item type code		Y		
Gov_item_type	The temporary item type if it exists or else the permanent item type	Y	Y	Y	
Charge_date_time	Date and time item was charged	Y	Y	Y	

**Table 29-12. Circrenewal\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Charge_date_only	Date item was charged	Y	Y	Y	
Charge_oper_id	Operator ID of the person who made the charge	Y	Y	Y	
Charge_location	ID of circulation desk where item was charged out				Y
Charge_location_code	Location code of circulation desk		Y		
Charge_location_name	Location name of circulation desk	Y		Y	
Renewal_count	Item renewal count	Y	Y	Y	
Renew_date_only	Date item was renewed	Y	Y	Y	
Renew_oper_id	Operator ID of the person who renewed the item	Y	Y	Y	
Renew_location_code	ID of circulation desk where item was renewed				Y
Location_name	Location name of circulation desk where item was renewed	Y		Y	

**Fundledger\_vw**

This view summarizes funds (see [Table 29-13](#)).

**Table 29-13. Fundledger\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Fundline	The name of the fund and all funds that connect it back to the main fund			Y	

**Table 29-13. Fundledger\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Fiscal_period_id	The ID of the fiscal period that the ledger and fund is in		Y		Y
Fiscal_period_name	The name of the fiscal period, the ledger, and the fund	Y	Y	Y	
Fiscal_period_start	The start date of the fiscal period	Y	Y	Y	
Fiscal_period_end	The end date of the fiscal period	Y	Y	Y	
Ledger_id	The id of the ledger that the fund is in		Y		Y
Ledger_name	Ledger name	Y	Y	Y	Y
Normal_ledger_name	Normalized ledger name		Y	Y	
Policy_name	The name of the acquisitions policy associated with this ledger	Y	Y	Y	
Fund_type	Fund type as defined in System Administration	Y	Y	Y	
Fund_category	Fund category-effects the meaning of allocation, commitment and expenditure amounts	Y	Y	Y	
Fund_id	The ID of the fund		Y		Y
Fund_name	Fund name	Y	Y	Y	
Normal_fund_name	Normalized fund name		Y	Y	
Parent_fund_id	Will be 0 (zero) if this is in the first level of the ledger; otherwise, the ID number of the parent fund (the fund above)	Y	Y		Y
Parent_fund	Name of the parent fund-this will be blank if the parent fund is the ledger	Y	Y	Y	Y

**Table 29-13. Fundledger\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Institution_id	Institution assigned ID-optional	Y	Y	Y	*
Begin_date	Fund beginning date	Y	Y	Y	
End_date	Fund ending date	Y	Y	Y	
Original_allocation	Original fund allocation in base currency	Y	Y	Y	
Current_allocation	Original allocation +/- fund transfers and adjustments in base currency	Y	Y	Y	
Cash Balance	Cash balance of fund in base currency	Y	Y	Y	
Free_balance	Free balance of fund in base currency	Y	Y	Y	
Expenditures	Total expenditures in base currency	Y	Y	Y	
Commitments	Total commitments in base currency	Y	Y	Y	
Commit_pending	Pending commitments in base currency	Y	Y	Y	
Expend_pending	Pending expenditures in base currency	Y	Y	Y	

**Heading\_vw**

Pulls together the headings from the heading index and the associated authority records with bibliographic records that count (see [Table 29-14](#)).

**Table 29-14. Heading\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Auth_id	The authority record for this heading if it exists	Y	Y		Y
Heading_id	The heading's ID number		Y		Y

**Table 29-14. Heading\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Reference_type	The reference type of this heading if an authority record exists	Y	Y	Y	
Normal_heading	The normalized version of the heading		Y	Y	
Display_heading	The display version of the heading	Y			
Opacbibs	The count of records NOT suppressed from the OPAC using this heading	Y	Y	Y	
Create_date	The date that this heading was first added to the database, either by a bibliographic record or authority record	Y	Y	Y	
Index_type	The index of the heading-name, subject	Y	Y	Y	
Heading_type	The thesaurus or sub-index type. For example, corporate, personal, LCSH, MESH	Y	Y	Y	

**Issues\_vw**

The issues\_vw view combines predicted and unpredictable issues into a single table. It is used by serials\_vw to give a complete view of serials check-in (see [Table 29-15](#)).

**Table 29-15. Issues\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Issue_id	Issue id number-combined with the component_id to form a key				Y

**Table 29-15. Issues\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Component_id	Component ID number-combined with issue_id to form a key				Y
Enumchron	The enumeration/chronology field of the issue	Y		Y	
Expected_date	The expected receipt date of the issue	Y	Y	Y	
Receipt_date	The date the issue was received or NULL if it has not been received	Y	Y	Y	
Received	The number of copies received	Y			

**Item\_vw**

Pulls information from holding and item records to give a complete description of the item. The MFHD ID can be used to link this to retrieve bibliographic information (see [Table 29-16](#)).

**Table 29-16. Item\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhd_id	Item's parent MFHD	Y	Y		Y
Call_no	Call number for MFHD	Y			
Call_no_type	MFHD indicator of call number type-Dewey, LC	Y	Y	Y	Y
Normalized_call_no	Normalized call number		Y	Y	
Item_id	The internal ID number of the item	Y	Y		Y
Barcode	The active barcode of the item, if one is set	Y	Y	Y	*
Perm_location_code	The permanent location code of the item	Y	Y	Y	
Perm_location	The permanent location of the item	Y		Y	

**Table 29-16. Item\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Gov_location_code	The temporary location code of the item if it exists, or else the permanent location code	Y	Y	Y	
Gov_location	The temporary location code of the item if it exists, or else the permanent location code	Y		Y	Y
Perm_item_type_code	The permanent item type code		Y		
Perm_item_type	The permanent item type	Y	Y	Y	
Gov_item_type_code	The temporary item type code if it exists or else the permanent type code		Y		
Gov_item_type	The temporary item type if it exists, or else the permanent item type	Y	Y	Y	
Media_type_code	The media type code if the item has been assigned a media type	Y	Y	Y	
Media_type	The media type if this item has been assigned a media type	Y	Y	Y	
Enumeration		Y		Y	
Chronology		Y			
Year		Y		Y	
Caption		Y			
Historical_browses	The number of times this item has been browsed	Y	Y	Y	
Historical_charges	The number of times this item has been charged out	Y	Y	Y	
Historical_bookings	the number of times this item has been booked in Media Scheduling	Y	Y	Y	

**Table 29-16. Item\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Hold_placed	The number of holds placed against this item	Y	Y	Y	
Recalls_placed	The number of recalls placed against this item	Y	Y	Y	
Create_date	The date this item record was created	Y	Y	Y	
Create_opid	The operator who created this record	Y	Y	Y	Y

**LCclass\_vw**

Takes Library of Congress classification numbers and breaks them down into individual classes used to do reports based on call numbers, call number ranges (see [Table 29-17](#)).

**Table 29-17. LCclass\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhid_id	The MFHD ID	Y	Y		Y
Firstletter	The first letter of the class number	Y	Y	Y	*
Class	The first 1 to 3 characters in the normalized call number field	Y	Y	Y	*
Longclass	A 7-character string that includes the class letters and class number up to the decimal point. The number is right justified	Y	Y	Y	*
Classnumber	Numeric field with the class number, including decimal places	Y	Y + class		

### **Marccomputer\_vw**

This view gives information from the bib 008 for computer files (see [Table 29-18](#)).

**Table 29-18. Marccomputer\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Audience	Target audience 008/22	Y	Y	Y	*
Filetype	Type of computer file 008/26	Y	Y	Y	*
Governmentpub	Government publication 008/28	Y	Y	Y	*

### **Marcbook\_vw**

This view extracts key information from the bib 008 for books (see [Table 29-19](#)).

**Table 29-19. Marcbook\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Audience	Target audience 008/22	Y	Y	Y	*
Itemform	Form of item 008/22	Y	Y	Y	*
Entirenature	Nature of entire form 008/24	Y	Y	Y	*
Governmentpub	Government publication 008/28	Y	Y	Y	*

**Table 29-19. Marcbook\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Conferencepub	Conference publication 008/29	Y	Y	Y	*
Literaryform	Literary form 008/33	Y	Y	Y	*
Biography	Biography 008/34	Y	Y	Y	*

**Marcmap\_vw**

This view extracts key information from the bib 008 for maps (see [Table 29-20](#)).

**Table 29-20. Marcmap\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Projection	Projection 008/23	Y	Y	Y	*
Cartographictyp	Type of cartographic material 008/25	Y	Y	Y	*
Governmentpub	Government publication 008/28	Y	Y	Y	*
Indexed	Index 008/31	Y	Y	Y	*

**Marcmusic\_vw**

This view extracts key information from the bib 008 for music (see [Table 29-21](#)).

**Table 29-21. Marcmusic\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*

**Table 29-21. Marcmusic\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Compositionform	Form of composition 008/18-19	Y	Y	Y	*
Musicformat	Format of music 008/20	Y	Y	Y	*
Audience	Target audience 008/22	Y	Y	Y	*
Itemform	Form of item 008/23	Y	Y	Y	*

**Marcserial\_vw**

This view extracts key information from the bib 008 for serials (see [Table 29-22](#)).

**Table 29-22. Marcserial\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Frequency	MARC frequency code 008/18	Y	Y	Y	*
Regularity	MARC regularity code 008/19	Y	Y	Y	*
Type	MARC serial type 008/21	Y	Y	Y	*
Originalform	Form of original item 008/22	Y	Y	Y	*
Itemform	Form of item 008/23	Y	Y	Y	*
Entirenature	Nature of entire form 008/24	Y	Y	Y	*
Governmentpub	Government publication 008/28	Y	Y	Y	*
Conferencepub	Conference publication 008/29	Y	Y	Y	*

**Marcvisual\_vw**

This view extracts key information from the bib 008 for visual materials (see [Table 29-23](#)).

**Table 29-23. Marcvisual\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	Bib record ID	Y	Y		Y
Bibtype	The bibliographic record type from the leader/06	Y	Y	Y	*
Biblevel	The bibliographic level from the leader/07	Y	Y	Y	*
Runningtime	Running time 008/18-20	Y	Y		
Audience	Target audience 008/22	Y	Y	Y	*
Governmentpub	Government publication 008/28	Y	Y	Y	*
Visualtype	Type of visual material 008/33	Y	Y	Y	*
Technique	Technique 008/34	Y	Y	Y	*

**MFHDblob\_vw**

This view provides access to entire MARC MFHDs (holding records) (see [Table 29-24](#)).

**Table 29-24. MFHDblob\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhid_id	MFHD ID number	Y	Y		Y
Marc_record	Whole MARC MFHD record in its native format, contained in a single row	Y	N	N	N/A

**NOTE:**

Normally you will run MARC record parsing functions to access a particular field in the record.

### **MFHDhistory\_vw**

This view provides information on the create and last update history of MFHDs (see [Table 29-25](#)).

**Table 29-25. MFHDhistory\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhid_id	MFHD ID number	Y	Y		Y
Create_operator_id	ID of operator who created the record	Y	Y	Y	Y
Create_date	Date and time of record creation	Y	Y	Y	
Create_location_id	ID of happening location where record was created		Y		Y
Update_operator_id	ID of operator that updated the record	Y	Y	Y	Y
Update_date	Date and time of record update	Y	Y	Y	
Update_location_id	ID of happening location where record was updated		Y		Y

### **NLMclass\_vw**

This view takes National Library of Medicine classification numbers and breaks them down into individual classes used to do reports based on call numbers, call number ranges (see [Table 29-26](#)).

**Table 29-26. NLMclass\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhid_id	The MFHD ID	Y	Y		Y
Firstletter	The first letter of the class number	Y	Y	Y	*
Class	The first 1 to 3 characters in the normalized call number field	Y	Y	Y	*

**Table 29-26.** NLMclass\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Longclass	A 7-character string that includes the class letters and class number up to the decimal point. The number is right justified	Y	Y	Y	*
Classnumber	Numeric field with the class number, including decimal places	Y	Y + class		

**Recordcount\_vw**

This view counts various records in the database (see [Table 29-27](#)).

**Table 29-27.** Recordcount\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Recordtype	Record type name	Y	Y	Y	
Count	Count of records	Y			

**Serials\_vw**

This view lists all issues for serial or multi-part check-in components (see [Table 29-28](#).)

**Table 29-28.** Serials\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Bib_id	The bib ID that is linked to this serial	Y	Y		Y
Mfhd_id	The MFHD ID that is linked to this serial	Y	Y		Y
Component_id	The serial check-in component ID				Y
Component_name	The check-in component name	Y			

**Table 29-28. Serials\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Component_name_norm	The normalized version of the component name		Y	Y	
Predict	Whether the serial is predicted or not; will be Y or N	Y	Y		
Next_issue_id	The ID number of the next expected issue	Y	Y	Y	Y
Note	Note field	Y			
Issue_id	ID of issue		Y		Y
Enumchron	The enumeration/chronology field of the issue	Y		Y	
Expected_date	The expected receipt date of the issue	Y	Y	Y	
Receipt_date	The date the issue was received or NULL if the issue has not yet been received	Y	Y	Y	
Received	The number of copies received	Y			

**Sudoclass\_vw**

This view takes SuDoc (Superintendent of Documents) classification numbers and breaks them down into individual classes used to do reports based on agency, call number ranges (see [Table 29-29](#)).

**Table 29-29. Sudoclass\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Mfhd_id	The MFHD ID	Y	Y		Y
Class	The first character letters in the normalized SuDoc call number field	Y		Y	*

**Table 29-29. Sudoclass\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Longclass	The character string that includes letters and numbers up to the decimal point. The number is right justified	Y	Y	Y	*

**Vendorinvoice\_vw**

This view includes actual fund amounts invoiced by vendors (see [Table 29-30](#)).

**Table 29-30. Vendorinvoice\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Vendor_code	The vendor code	Y	Y	Y	
Vendor_name	The vendor name	Y	Y	Y	
Vendor_type	The vendor type assigned to the vendor	Y	Y	Y	
Institution_id	The institution ID of the vendor if one has been assigned to the vendor	Y	Y	Y	*
Bill_to_location_code	The billing location code on the invoice	Y	Y	Y	
Bill_to_location	The billing location on the invoice	Y		Y	
Invoice_date	The date of the invoice	Y	Y	Y	
Invoice_number	The invoice number	Y	Y	Y	*
Currency_code	The currency code the invoice ID is in. The amounts expressed in this view are already in the base currency.	Y	Y	Y	Y
Currency_name	Name of currency	Y		Y	

**Table 29-30. Vendorinvoice\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Invoice_id	The invoice ID				Y
Invoice_status	The status of the invoice in Voyager	Y	Y	Y	
Invoice_status_date	The date the above status was set	Y	Y	Y	
Voucher_number	The voucher number printed on the voucher/check request	Y	Y	Y	*
Expenditures	Expenditures of this invoice in base currency	Y	Y	Y	
Expend_pending	Pending expenditures of this invoice—pending invoices only	Y	Y	Y	
Policy_name	The name of the acquisitions policy associated with this ledger	Y	Y	Y	
Fiscal_period_name	The name of the fiscal period used to expend funds	Y	Y	Y	
Fiscal_period_start	The beginning date of the fiscal period	Y	Y	Y	
Fiscal_period_end	The ending date of the fiscal period	Y	Y	Y	
Ledger_name	The ledger name	Y	Y	Y	Y
Fund_name	The fund name	Y	Y	Y	
Institution_fund_id	The institution assigned ID—optional	Y	Y	Y	*

**Vendororder\_vw**

This view summarizes orders by purchase order line item and vendor (see [Table 29-31](#)).

**Table 29-31. Vendororder\_vw**

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Vendor_code	The vendor code	Y	Y	Y	
Vendor_name	The vendor name	Y	Y	Y	
Vendor_type	The vendor type assigned to the vendor	Y	Y	Y	
Institution_id	The institution ID of the vendor if one has been assigned to the vendor	Y	Y	Y	*
Po_number	The purchase order number	Y	Y	Y	
Po_type	The purchase order type	Y	Y	Y	
Order_location_code	The order location code	Y	Y	Y	
Order_location	The order location	Y	Y	Y	
Po_status	The purchase order status	Y	Y	Y	
Po_status_date	The date that the purchase order was sent	Y	Y	Y	
Currency_name	The name of the currency that was used on the purchase order	Y	Y	Y	
Total	The total amount of the purchase order in base currency	Y	Y		
Unit_price	The single unit price of the line item in base currency	Y	Y		
Quantity	The number of units ordered	Y	Y		
Line_price	The line item total in base currency (unit_price * quantity) +/- adjustments	Y	Y		
Mfhd_id	The MFHD id of the title ordered	Y	Y		Y

**Table 29-31.** Vendororder\_vw

Field Name	Description and Use	Report	Query Select	Normalized Sort	Link to Field
Po_line_status	The line item status	Y	Y	Y	
Invoice_status	The invoice status of a line	Y	Y	Y	
Line_status_date	The date of the last change to the line item status or line item invoice status	Y	Y	Y	



---

<b>Introduction</b>	<b>30-1</b>
<b>The Purpose of this Chapter</b>	<b>30-2</b>
<b>Setting Up WebAdmin</b>	<b>30-2</b>
• System Requirements	30-2
• Starting the Server Dæmon	30-3
• Creating WebAdmin Users and Passwords	30-3
• Adding the WebAdmin Authorization Section to the httpd.conf File	30-5
<b>Starting the Voyager Server Utilities</b>	<b>30-6</b>
<b>Using WebAdmin</b>	<b>30-7</b>
• Voyager Server Utilities Page	30-7
Acquisitions Utilities	30-9
Cataloging Utilities	30-12
Circulation Utilities	30-24
Media Scheduling Utilities	30-32
OPAC Reports Utilities	30-34
System Admin Utilities	30-36



---

## **Introduction**

---

Voyager's WebAdmin (Server Utilities) allows operators to run various server batch jobs by selecting from the options that appear on the opening page. Instead of starting a telnet session and entering the batch job commands at a command-line prompt, you can simply point your web browser at WebAdmin and click the job that you want to run. WebAdmin uses the existing batch jobs in order to do the processing, but provides a more user-friendly interface for operators. Using WebAdmin, you can run the following batch jobs:

- Acquisitions (`Pacqjob`)
- Cataloging (`Pcatjob`)
- Circulation (`Pcircjob`)
- Bursar Transfer (`Pbursar`)
- Bulk Import (`Pbulkimport`)
- MARC Export (`Pmarcexport`)
- OPAC Reports and Processes (`Popacjob`)
- Media Scheduling Utilities (`Pmediajob`)

You can also access the log and report files that are created by the batch jobs through WebAdmin.

Information on the Circulation, Cataloging, and Acquisitions batch jobs is found in this user's guide and the *Voyager Reporter User's Guide*, [Creating Input Files for Circulation](#), [Creating Input Files for Cataloging](#), and [Creating Input Files for Acquisitions](#).

See also [Bursar Transfer System](#) for additional information on the Bursar Transfer, [Bulk Import, Replace, and Merge of MARC Records](#) for additional information on the Bulk Import batch job, [Bulk Export of MARC Records](#) for additional information the MARC Export batch job, and [Popacjob](#) for additional information OPAC Search Logging is found in this user's guide.

## **The Purpose of this Chapter**

---

This chapter discusses

- Setting up WebAdmin
- Starting the Voyager Server Utilities
- Changing your WebAdmin password
- Using WebAdmin

## **Setting Up WebAdmin**

---

This section discusses how to set up WebAdmin.

### **System Requirements**

---

In order to run WebAdmin, you must have Perl installed on the server. Perl is commonly installed in the `/m1/shared/perl/version` directory.

You must also have the Perl DBI (Database Interface) installed as well as perl DBD:Oracle.

Perl and the DBI and DBD:Oracle components are included with the standard Voyager installation.

Each copy of WebAdmin works on only one database. If you want to run WebAdmin on multiple databases, you must install a separate copy of WebAdmin for each database you want to access. Each copy must have its own directory structure and set of files.

## Starting the Server Daemon

---

The WebAdmin daemon must be running in order for the jobs to be run.

The daemon should start when you start Voyager.

The script for the WebAdmin daemon is called `Pwebadmind` and is located in the `/m1/voyager/xxxdb/sbin` directory. You can use the `-f` parameter to specify the interval in minutes at which you want the daemon to run the batch jobs. For example, to run the daemon at five minute intervals, you would enter the following at the command line:

`Pwebadmind -f5`

You may want to add this line to the startup script for your server so that it automatically starts up when you start your server.

### NOTE:

The daemon should not be set to run more than every two to five minutes. Setting the daemon to run more often than that will diminish the performance of not just the daemon but also any batch jobs being run manually.

When starting up the daemon, you must be logged in as Voyager in order for the daemon to run.

## Creating WebAdmin Users and Passwords

---

The file that defines and encrypts users and passwords is named `htpasswd` and it is located in the `/m1/shared/apache2/bin` directory.

The login and password information will be stored in a text file named `xxxdb.users` which is located in the `/m1/shared/apache2/conf` directory. Here, `xxxdb` is the name of the database.



### Procedure 30-1. Creating a New Login/Password for the First Time

---

Use the following to create a new login/password for the first time:

1. Login as `root` and enter: `cd /m1/shared/apache2/bin`
2. At the prompt enter: `./htpasswd -c /m1/shared/apache2/conf/AuthorizedUsers/xxxdb.users webadmin` where `webadmin` is the user name.

3. Then, the system prompts you to enter that user's password two times.

Result: A new login and password is created.

---



#### **Procedure 30-2. Adding an Additional Login/Password to the Existing File**

---

Use the following to add an additional login/password to the existing file.

1. Login as **root** and enter: **cd /m1/shared/apache2/bin**
2. At the prompt enter: **./htpasswd /m1/shared/apache2/conf/AuthorizedUsers/xxxdb.users newname** where newname is the new user.
3. Then, the system prompts you to enter the password for newname two times.

Result: An additional login and password is created.

---



#### **Procedure 30-3. Deleting a Login/Password for an Existing User**

---

Use the following to delete a login/password for an existing user.

1. Login as **root** and enter: **cd /m1/shared/apache2/conf**
2. Open the **xxxdb.users** file using vi or another server text editor.
3. Delete the line for the specific user you want to delete then save the **xxxdb.users** file.

Result: The login and password are deleted.

---



#### **Procedure 30-4. Modifying a Login/Password for an Existing User**

---

Use the following to modify a login/password for an existing user.

1. Login as **root** and move to the **/bin** directory, enter:  
**cd /m1/shared/apache2/bin**

2. At the bin prompt, enter:

```
./htpasswd /m1/shared/apache2/conf/AuthorizedUsers/
xxxdb.users webadmin where webadmin is the user name.
```

3. Then, the system prompts you to enter the new password two times.

Result: The login and password are modified.

---

### **Adding the WebAdmin Authorization Section to the httpd.conf File**

Add the following section to the httpd.conf file to set up the authorization for the WebAdmin login page.

The httpd.conf file is located in the /m1/shared/apache2/conf/AuthorizedUsers directory.



#### **Procedure 30-5. Adding the WebAdmin Authorization Section to the httpd.conf File**

Use the following to add WebAdmin authorization to the httpd.conf file.

1. Login as root and navigate to the /m1/shared/apache2/conf/AuthorizedUsers directory and backup the httpd.conf file, at the prompt enter `cp httpd.conf httpd.$$.backup`

2. Then open the httpd.conf file using vi or another server text editor.

3. After the line in the file that reads

`DocumentRoot /m1/voyager/xxxdb/webvoyage/html` enter:

```
#####Begin WebAdmin Configuration Change#####
ScriptAlias /webadmin/cgi-bin/ /m1/voyager/xxxdb/webadmin/
cgi-bin/
```

```
Alias /webadmin /m1/voyager/xxxdb/webadmin
```

```
DirectoryIndex webvoy.htm index.htm index.html
```

```
<Directory /m1/voyager/xxxdb/webadmin >
```

```
AuthName "xxxdb security"
```

```
AuthType Basic  
  
AuthUserFile /m1/shared/apache2/conf/AuthorizedUsers/  
xxxdb.users  
  
require valid-user  
  
</Directory>  
  
#####End WebAdmin Configuration Change#####
```

Result: You are now ready to access WebAdmin.

## Starting the Voyager Server Utilities

---

To start WebAdmin, in your browser address box, enter the URL of the server where your database resides followed by /webadmin.

The login dialog opens (see [Figure 30-1](#)).

---



---

**Figure 30-1. WedAdmin Log in Dialog Box**



### Procedure 30-6. Logging into WebAdmin

---

Use the following to log into WebAdmin.

1. Enter your user name into the **User Name** field.

2. Enter your password into the **Password** field.

You can have your user name and password appear the next time you start WebAdmin by selecting the **Save this password in your password list** check box. If your user name and password currently appear automatically in the dialog, you can prevent them from appearing by unselecting this check box.

3. Click the **OK** button.

Result: The WebAdmin main page, Voyager Server Utilities, opens (see [Figure 30-2 on page 30-8](#)).

---

## Using WebAdmin

---

WebAdmin is entirely web-based. From this main page users select the batch jobs to run or files to examine. Each batch job or set of files has its own web page, accessed by clicking the link to that page.

### Voyager Server Utilities Page

The Voyager Server Utilities (main) page ([Figure 30-2](#)) contains all of the links to the various batch jobs and report and log files. This page comes from the file `index.html` on the server. You can edit this file like you would a standard html page.

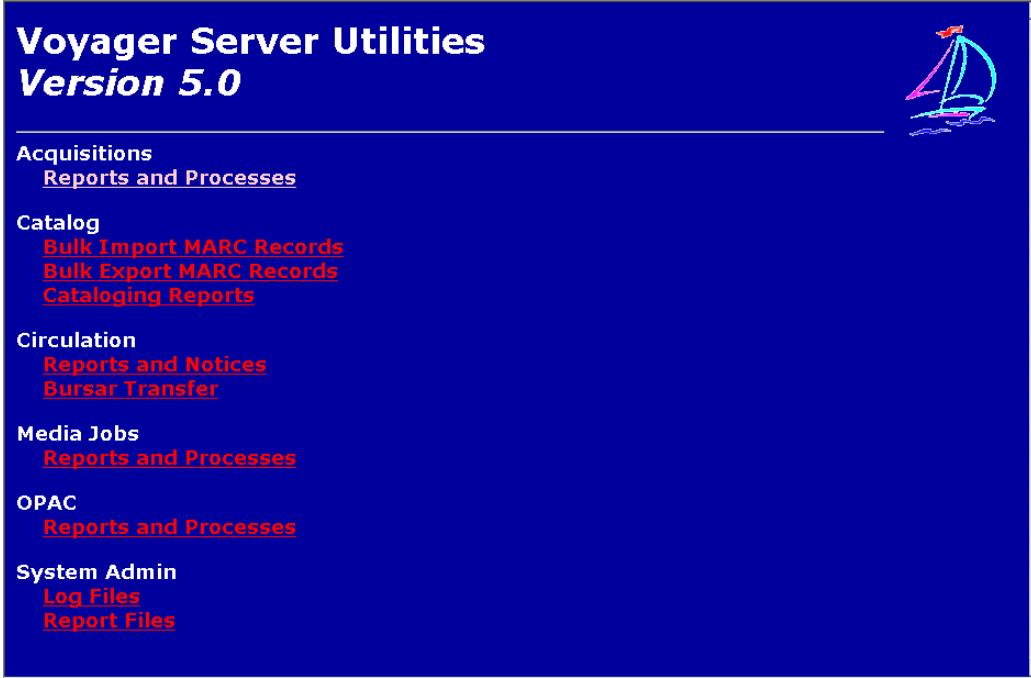


Figure 30-2. Webadmin Main Page

The activities available on the main page are broken down by module. You can click the links to display the following pages:

- Acquisitions: Reports and Processes: Acquisitions Batch Job page (Pacctjob)
- Catalog: Bulk Import MARC Records: Bulk Import Batch Job page (Pbulkimport)
- Catalog: Bulk Export MARC Records: MARC Export Batch Job page (Pmarcexport)
- Catalog: Cataloging Reports: Cataloging Batch Job page (Pcatjob)
- Circulation: Circ. Reports and Notices: Circulation Batch Job page (Pcircjob)
- Circulation: Bursar Transfer: Bursar Transfer Batch Job (Pbursar)
- Mediajobs: Reports and Processes
- OPAC: Reports and Processes: Search Logging Batch Jobs (Popacjob)
- System Admin: Log Files: Contents of the /log subdirectory
- System Admin: Report Files: Contents of the /rpt subdirectory

## Acquisitions Utilities

On the main page, the Acquisitions utility available is for acquisitions reports and notices. The Reports and Processes utility allows you to run all acqjobs.

See [The Acquisitions Batch Jobs](#) in the *Voyager Reporter User's Guide* for explanation of Acqjob 1-4 jobs and for instructions on running in WebAdmin.

See [Acquisitions Batch Job - Fix Exchange Rates](#) on [page 16-1](#) for information on Acqjob 5.

The procedure for running Acqjob 5 using Webadmin is shown in [Procedure 30-7, Running Acqjob 5 Using WebAdmin](#).



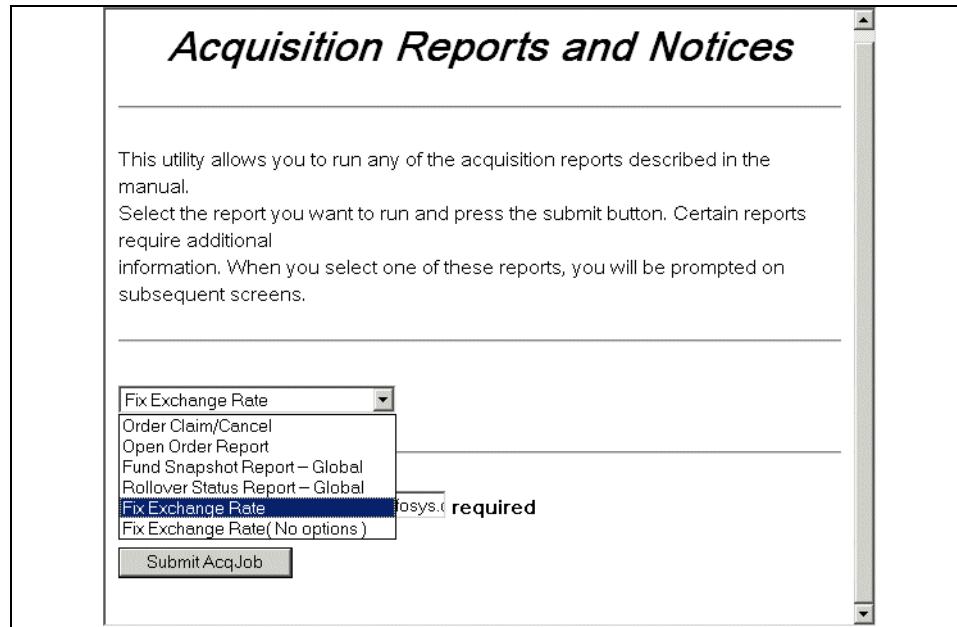
### Procedure 30-7. Running Acqjob 5 Using WebAdmin

---

Use the following to run Acqjob 5 (Fix Exchange Rates) with optional parameters.

1. From the WebAdmin main page ([Figure 30-2](#)), Acquisitions section, click the **Reports and Processes** link.

Result: The **Acquisitions Reports and Notices** page opens (see [Figure 30-3](#)).



**Figure 30-3. WebAdmin Acquisitions Reports and Notices Page**

2. From the drop-down menu, select **Fix Exchange Rates**.
3. Enter your e-mail address in the **E-Mail Address** field.

**NOTE:**

You must enter the entire email address (that is, user@hostname.com) in order for the e-mail to be sent.

4. Click the **Submit Acqjob** button.

Result: The Fix Exchange Rate Options page opens (see [Figure 30-4](#)).

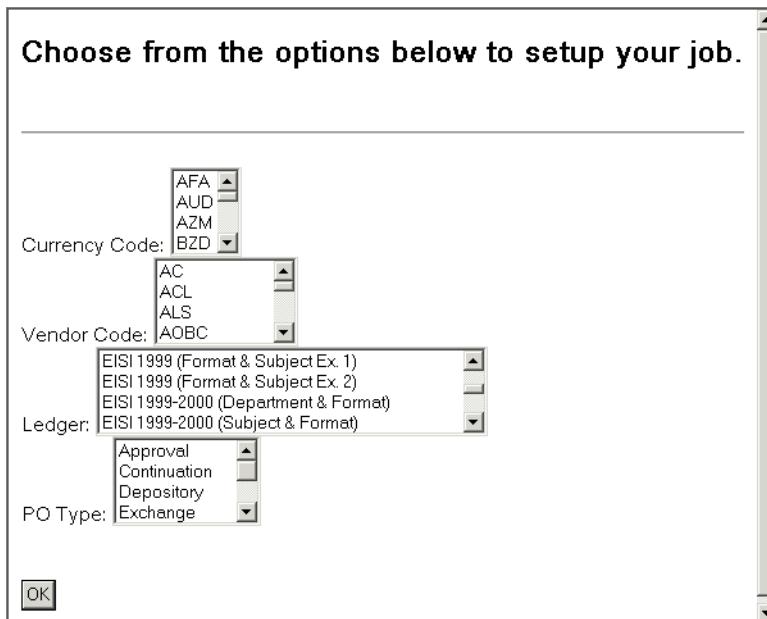


Figure 30-4. Fix Exchange Rate Options Page

5. As needed, select items from the following fields to limit the adjustment:
    - **Currency Code**
    - **Vendor Code**
    - **Ledger**
    - **PO Type**
  6. Select **OK** to run the job.
- Result: The job runs, the **Job Scheduler** page opens, and an e-mail is sent to the user (see [Figure 30-5](#)).

## Job Scheduler

[Return to Main Page](#)

---

**Email notification to:** m.phillips@endinfosys.com  
**job:** acqjob -j 5

---

---

**Figure 30-5. Job Scheduler Page after Submitting Acqjob 5**

---

### Cataloging Utilities

On the main page, the Cataloging utility is available for cataloging reports.

Users run Bulk Import MARC Records, Bulk Export MARC Records, and other Cataloging Reports.

See [The Cataloging batch jobs](#) in the *Voyager Reporter User's Guide* for explanations of the various cataloging jobs. See the *Voyager Reporter User's Guide* for instructions on how to run cataloging jobs one through ten.

#### Bulk Import MARC Records

The Bulk Import MARC Records utility allows to automatically import a large number of MARC records from a file.

All of the same options in the Bulk Import batch job are available when you import records this way except users may not use the -X NOKEY parameter and argument when running bulk import using WebAdmin.

See [Bulk Import, Replace, and Merge of MARC Records](#) on [page 12-1](#) for more information.



#### Procedure 30-8. Running Bulk Import Using WebAdmin

---

Use the following to run the Bulk Import Utility.

1. After logging in to WebAdmin, from the main page, Cataloging section, click the **Bulk Import MARC Records** link.

Result: The **MARC record Bulk Import** page opens (see [Figure 30-6](#)).

## MARC Record Bulk Import

Use the **Browse** button to select the MARC file on **your PC** to upload to the Voyager database. Sending large files (more than 1000 records) may take from one minute to several minutes. The file will not be sent until you press the **Submit** button at the bottom of the screen. When the file is sent, you will see notification of a successful upload.

Local Filename:

### Voyager Batch Upload Information

All of the fields in this part of the form come directly from the Voyager Conversion and Technical Manual, from the section Bulk Import and Replace of MARC records. Please consult the documentation for information about these fields.

Import Code:   required

Operator Name:   optional

Dedupe Location Code:   optional

Holdings Location Code:   optional

Begin Record:  ('#' or 'first')

End Record:  ('#' or 'last')

Import file has interleaved bib and holdings records ?:

Delete bib records matched in import file ?:

Delete holdings records matched in import file ?:

Mark imported records as OK for export ?:

Use the **Browse** button to select the USEMARCON file on **your PC**.  
USEMARCON Initialization Filename:

optional

Show/Approve MARC display before database load? :

Email Address:  required

---

Figure 30-6. MARC Record Bulk Import Page

2. In the following fields:

- a. Enter the filename that you want to import into your Voyager database in the **Local Filename** field. Or click the **Browse** button to select the file you want to import. This opens the **File Upload** dialog box. Select a file to load and click the **Open** button.
- b. In the **Import Code** field, select the import code of the Import/Replace profile that you want to apply to this import session. The Import Code comes from the **Rule Code** field on the Bulk Import Rules dialog in the System Administration module.
- c. In the **Operator Name** field, select the operator name that you want saved with the record in the database as having last modified the record.
- d. In the **Dedupe Location Code** field, select the code for the location at which the records were last modified.
- e. In the **Holdings Location Code** field, select the code for the location that you want saved with the record in the database as the location that the records come from.
- f. In the **Begin Record** field, enter the sequential number of the record which you want to begin importing. (For example, to start importing at the fifth record, enter 5.) Enter first to begin with the first record.
- g. In the **End Record** field, enter the sequential number of the record which you want to end importing. (For example, to end after record ten, enter 10). To end with the last record, enter last.
- h. If the import file contains both bibliographic and holdings records that are interleaved, select the **Import file has interleaved bib and holdings records?** check box.
  - i. If you want to delete any bibliographic records in your database that match the bibliographic records in the file, select the **Delete bib records matched in import file?** check box. Note that this will not import any records - it will only delete bibliographic records from the database.
  - j. If you want to delete any holdings records in your database that match the holdings records in the file, select the **Delete holdings records matched in import file?** check box. Note that this will not import any records - it will only delete holdings records from the database.
- k. If you want the **OK to export** check box (on the **System** tab of bibliographic, authority, and holdings records on view in the Cataloging module) to be turned on for all of the newly-imported records in the database, select the **Mark imported records as OK for export?** check box.

- i. For USEMARCON Initialization filename enter the full path. This points to the translation files needed to translate the records into MARC21 format before running Bulk Import as usual.
- m. Select the **Show/Approve MARC display before database load?** check box to view all of the bibliographic record that you are importing onscreen before performing the import. If the records are not satisfactory, you may cancel the process without importing the records.
- n. In the **E-mail Address** field, enter your email address. The utility will send an e-mail to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
- o. When you have verified that the settings on the page are all correct, click the **Submit Batch Import** button to begin the importing process.

Result: If you did not turn on the **Show/Approve** check box, the **Job Scheduler** page tells you when the import will begin.

If you did turn on the **Show/Approve** check box, you will be presented with a display of the records you want to import. To accept these records and begin the import, click the **Accept these records** button. Then the **Job Scheduler** page tells you when the import will begin. To reject these records, click the **Back** button in your browser until you reach the main page.

3. Click the **Back** button.

Result: Returns to the main menu.

---

### Bulk Export MARC Records

The Bulk Export MARC Records utility that allows to automatically export a large number of MARC records from a file. All of the same options in the MARC Export batch job are available when you import records this way. See [Bulk Export of MARC Records on page 10-1](#).



### Procedure 30-9. Running MARC Record Bulk Export Using WebAdmin

---

Use the following to run the MARC Record Bulk Export Utility.

1. After logging in to WebAdmin, from the main page, Cataloging section, click the **Bulk Export MARC Records** link.

Result: The **MARC record Bulk Export** page opens (see [Figure 30-7](#)).

**MARC Record Bulk Export**

---

This utility allows you to export several different types of MARC records (bib, holdings, authority, authority - main, authority - subject, authority - series, bib-holdings groups). Please complete the required information below and click on the submit button.

When the job is completed, you will receive notification via email that the job has ended as well as any additional information which was provided by the job.

Finally, check the "Voyager's Report Files" link for both log exp.[datetime] and marc.exp.[datetime] output files with the current date and time which will be generated from this run.

---

Record Type:	<input type="button" value="Bibliographic"/>
Export from Specific Library (Bib Export Only):	<input type="button"/>
Export OCLC Only (Bib Export Only):	<input type="checkbox"/>
System Control (003) Identifier:	<input type="text"/>
System Control (040\$0) Identifier:	<input type="text"/>
<b>Export Mode:</b>	
<input checked="" type="radio"/> MARC ID Input File <input type="button" value="Browse..."/> <input type="radio"/> ISBN Input File (Bib Export Only) <input type="button" value="Browse..."/>	
<input type="radio"/> Range of MARC IDs <input checked="" type="radio"/> All <input type="radio"/> ID Range (00000-00000) <input type="text"/> - <input type="text"/>	
<input type="radio"/> Date Range -- Create Dates <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- Update Dates <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- Create or Update Dates <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- OK to Export <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- Suppressed <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- Excluded <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<input type="radio"/> Date Range -- Excluded and Suppressed <input checked="" type="radio"/> Today - <input type="text"/> days <input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/> -	
<b>Select an Exclude Type</b>	
<input checked="" type="radio"/> Exclude from create location and last update location <input type="radio"/> Exclude from most recent location (create or update)	
<b>Select Exclude Location File</b> Select the Location file on your PC: <input type="text"/> <input type="button" value="Browse..."/>	
Create Control Number (001) from LCCN (010a) ? <input type="checkbox"/>	
Update 852\$a on Holdings Export with the following text: <input type="text"/>	
Ignore Suppressed Records? <input type="checkbox"/>	
Convert Exported Record to: <input type="button"/>	
Use the <b>Browse</b> button to select the USEMARCON file on your PC. USEMARCON Initialization Filename: <input type="text"/> <input type="button" value="Browse..."/> optional	
Email Address: <input type="text"/> required	
<input type="button" value="Submit Bulk Export"/>	

**Figure 30-7. Marc Record Bulk Export Page**

2. In the following fields:

- a. In the **Record Type** drop-down menu, select the type of record that you want to export.
- b. If you want to export only bibliographic records from a specific library, select the library from the **Export From Specific Library** drop-down menu.
- c. If you want to export only OCLC-created bibliographic records, select the **Export OCLC Only** check box.
- d. If you choose to create a control number (001) from the LCCN (101a), you are required to enter a System Control Identifier. Enter this identifier into the **System Control (003) Identifier** field. This code will be placed into the 003 of each exported bibliographic or authority MARC record. See Create Control Number (001) from LCCN (010a) (step #7).
- e. If you want to place a code for the Modifying Agency in the 040\$d of each exported record, enter the identifying code into the **System Control (040\$d) Identifier** field.
- f. From the table, select an **Export Mode**. Click one of the radio buttons in the Export Mode column to select one of the following options:
  1. **MARC ID Input.** (Bib Export Only.) Allows you to specify a file with a list of MARC ID numbers of the records you want to export. Enter the filename and path in the field or click the **Browse** button in the Export Target column to select the name of the list file.
  2. **ISBN Input File.** (Bib Export Only.) Allows you to specify a file with a list of ISBN numbers of the records you want to export. Enter the filename and path in the field or click the **Browse** button in the Export Target column to select the name of the list file.
  3. **Range of MARC IDs.** Select the All radio button in the Export Target column to designate that all records are to be exported; or, select the **ID Range** radio button in the Export Target column to specify a range of MARC IDs to be exported. Enter the beginning of the range into the first field and the ending of the range into the last field. Both the first and last records specified will be exported.
  4. **Date Range -- Create Dates.** Select the **Today** radio button in the Export Target column to specify that you want to export all records created between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records created between

two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.

5. **Date Range -- Update Dates.** Select the **Today** radio button in the Export Target column to specify that you want to export all records updated between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records updated between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.
6. **Date Range -- Create or Update Dates.** Select the **Today** radio button in the Export Target column to specify that you want to export all records created or updated between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records created or updated between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.
7. **Date Range -- OK to Export.** Select the **Today** radio button in the Export Target column to specify that you want to export all records that have had the OK to Export flag set or updated between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records that have had the OK to Export flag set or updated between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.
8. **Date Range -- Suppressed.** Select the **Today** radio button in the Export Target column to specify that you want to export all records suppressed between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records suppressed between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.

9. **Date Range -- Excluded.** Select the **Today** radio button in the Export Target column to specify that you want to export all records excluded between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records excluded between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.
  10. **Date Range -- Excluded and Suppressed.** Select the **Today** radio button in the Export Target column to specify that you want to export all records excluded and suppressed between the current day and a number of days previous; then enter the number of days previous in the days field. If you want to export all records excluded and suppressed between two dates, select the **Date Range** radio button in the Export Target column. Then enter the starting date into the first field and the ending date in the second field. Dates must be entered in the format YYYY-MM-DD.
- g. From the table, in the **Select an Exclude Type** section, click one of the radio buttons in the **Exclude Type** column:
    1. Exclude from create location or last update location.
    2. Exclude from most recent activity location
  - h. From the table, in the **Select Exclude Location File** section, enter or browse for the Location file on your PC.
  - i. **Create Control Number (001) from LCCN (010a).** If you want to change the LCCN number of the records being exported to a control number, select the check box. If you select this option, you may also specify a code to be placed into the 003 field in the System Control (003) Identifier.
  - j. **Update 852|a on Holdings Export with the following text.** If you want to enter any text into the 852a field of the exported record, enter the text into the field.
  - k. If you want to ignore suppressed records, that is not export them, select the **Ignore Suppressed Records?** check box.
  - l. If you want the records being exported to be converted to a specific cataloging character set (RLIN Legacy Encoding, OCLC, USMARC (MARC21 MARC8), select the character set from the **Convert Exported Record to** drop-down menu. The Unicode character set is the default.
  - m. If you are running USEMARCON, enter the name of the **USEMARCON initialization file.**

- n. In the **E-mail Address** field, enter your e-mail address. The Utility will send an email to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
- o. When you have verified that the settings on the page are all correct, click the **Submit Bulk Export** button to begin the export process. The **Job Scheduler** page will appear with information about when the export will begin.

Result: The Bulk Export runs.

3. Click the **Back** button.

Result: Returns to the main menu.

---

## Cataloging Reports

Users can run cataloging batch jobs using WebAdmin. Cataloging reports and the Global Headings Change (GHC) jobs.

See [Global Heading Change Jobs](#) on [page 13-1](#) for GHC information and the *Voyager Reporter User's Guide* for more information on the cataloging reports.



### IMPORTANT:

*The cataloging GHC jobs should be run in order and in concert with specific cataloging module activities.*



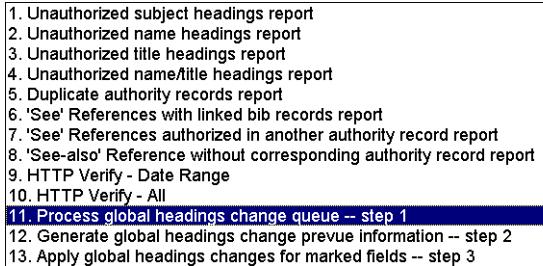
## Procedure 30-10. Running Global Headings Change Jobs Using WebAdmin

---

Use the following to run the GHC jobs.

1. After logging in to WebAdmin, from the main page, Cataloging section, click the **Cataloging reports** link.

The **Cataloging reports and processes** page opens.
2. Select from the drop-down menu the GHC job you wish to run, catjob 11, catjob 12, or catjob 13 (see [Figure 30-8](#)).



**Figure 30-8. Drop-DOWN MENU Options on the Cataloging Reports and Processes Page**

- a. If you select either catjob 11 or catjob 12
  1. In the **E-mail Address** field, enter your e-mail address. The Utility will send an email to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
  2. When you have verified that the settings on the page are all correct, click the **Submit** button to begin the job. The **Job Scheduler** page will appear.
- b. If you select catjob 13, the system prompts you to provide the optional variables of location and operator (see [Figure 30-9](#)).

---

**Make selections below to complete the setup of the job.**

---

Location Code:  optional

Operator Name:  optional

**Figure 30-9. Select Location and Operator to Run Catjob 13**

1. select the location code from the drop-down menu
2. select the operator name from the drop-down menu

3. In the **E-mail Address** field, enter your e-mail address. The Utility will send an email to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
4. When you have verified that the settings on the page are all correct, click the **Submit** button to begin the job. The **Job Scheduler** page opens.

Result: The catjob runs.

---

## Circulation Utilities

From the main Voyager WebAdmin page, the Circulation batch jobs are accessed from the following Circulation options:

- Reports and Notices.
- Bursar Transfer.

Additional information regarding Circulation batch jobs can be found in [Circulation Batch Jobs on page 8-1](#).

Voyager WebAdmin Circulation options provide batch processing that is used in a variety of ways. In particular, a subset of the circjobs that can be run from WebAdmin generate data that is used as input to Microsoft Access to generate standardized reports and notices. See the *Voyager Reporter User's Guide* for more information regarding these alternatives.

Forgiving fines and fees circjobs are described in the procedures in this section.



### **Procedure 30-11. Forgiving fines by patron ID (circjob 40)**

---

Use the following to forgive fines by Patron ID in the WebAdmin client.

#### **⚠️ IMPORTANT:**

Before starting this job, you should have a file listing patron IDs delimited by carriage-returns.

1. From Voyager Server Utilities (the main page of the WebAdmin client), click **Reports and Notices** under the **Circulation** heading.

- 
2. Scroll down the page until you see the **Forgive Fines by Patron ID** option in the left column of the table ([Figure 30-10](#)). Select the radio button beside it.
- 

The screenshot shows a web-based form titled "Circjob 40, WebAdmin". On the left, there is a table with two columns. The first column has a radio button labeled "Forgive Fines by Patron ID" which is circled in red. The second column contains three input fields: "Operator Id:" with a required indicator, "Input File:" with a required indicator, and a checkbox labeled "Run In Test Mode?".

---

**Figure 30-10. Circjob 40, WebAdmin**

3. In the corresponding right column, enter your Operator ID. (This will tag the job to your ID in the system.)
4. In the **Input File** field, enter the name of the input file. If the file resides somewhere besides /m1/voyager/xxxdb/local, include the path with the file name.
5. To run the file in test mode (and view the logs before running the actual batch job), select the check box beside **Run In Test Mode?**
6. Scroll down to the bottom of the page and enter your e-mail address in the required address field, then click the **Submit CircJob** button to run the job.

Result: Voyager runs the batch job, forgiving the fines and fees for those patrons whose IDs are listed in the file. If you ran the job in test mode, the database is not updated but Voyager produces an error log and an audit report. For information about these, see page [8-35](#).

---



---

#### **Procedure 30-12. Forgiving fines by date created (circjob 41)**

---

Use the following to forgive fines by date created in the WebAdmin client.

1. From Voyager Server Utilities (the main page of the WebAdmin client), click **Reports and Notices** under the **Circulation** heading.

- 
2. Scroll down the page until you see the **Forgive Fines by Fine Create Date** option in the left column of the table ([Figure 30-11](#)). Select the circle beside it.
- 

Forgive Fines by Fine Create Date

Operator Id:

Required

Forgive Stub Record Fines?

Forgive Fines from  
ONLY  
these Circulation  
Locations:

CIRC  
ELECTRONIC  
Offsite  
Res  
main  
media  
sc

Select All

Fine Create Date Range (YYYY-MM-DD):  
 -

Required

Run In Test Mode?

---

**Figure 30-11. Circjob 41, WebAdmin**

3. In the corresponding right column, enter your Operator ID. (This will tag the job to your ID in the system.)
4. Click the **Forgive Stub Record Fines?** check box to have fines for Universal Borrowing stub records forgiven.
5. To limit the Circulation locations from where you want the fines removed, select any of the locations in the **Circulation Locations** list box.

**NOTE:**

The list box is populated with locations where the fines and fees were first created.

6. Enter the **Fine Create Date Range (YYYY-MM-DD)**: to specify the incur dates for which you want to forgive.
7. To run the job in test mode and view logs, select the **Run In Test Mode?** check box.
8. Scroll down to the bottom of the page and enter your e-mail address in the required address field, then click the **Submit CircJob** button to run the job.

Result: Voyager runs the batch job, forgiving the fines and fees for the create date range with any limits you have set. If you ran the job in test mode, the database is not updated but Voyager produces an error log and an audit report. For information about these, see page [8-35](#).



#### **Procedure 30-13. Forgiving fines by patron group and expiration date (circjob 42)**

---

Use the following to forgive fees and fines using patron group and expiration date.

1. From Voyager Server Utilities (the main page of the WebAdmin client), click **Reports and Notices** under the **Circulation** heading.
2. Scroll down the page until you see the **Forgive Fines by Patron Group and Expiration Date** option in the left column of the table ([Figure 30-12](#)). Select the radio button beside it.

Forgive Fines by Patron Group and Expiration Date

Operator Id:  
  
Required

Forgive Stub Record Fines?

Forgive Fines from ONLY these Circulation Locations:

CIRC  
ELECTRONIC  
Offsite  
Res main

Select All

Expiration Date Range (YYYY-MM-DD):  
 -   
Required

Forgive Fines for ONLY these Patron Groups:

BINDERY  
CARREL  
GRAD  
RESTRICTED

Select All

Required

Run In Test Mode?

Figure 30-12. Circjob 42, WebAdmin

3. In the corresponding right column, enter your Operator ID. (This will tag the job to your ID in the system.)
4. Click the **Forgive Stub Record Fines?** check box to have fines for stub records forgiven.
5. To limit the Circulation Locations from where you want the fines removed, select any of the locations in the **Circulation Locations** list box.
6. Enter the **Expiration Date Range (YYYY-MM-DD)**: to specify the expiration date range for patron records whose fines you want to forgive.

7. To limit the patron groups whose fines are being forgiven, select one or more group names from the **Patron Groups** list box.
8. To run the job in test mode and view logs, select the **Run In Test Mode?** check box.
9. Scroll down to the bottom of the page and enter your e-mail address in the required address field, then click the **Submit CircJob** button to run the job.

Result: Voyager runs the batch job, forgiving the fines and fees for the patron group(s) and expiration date range specified. If you ran the job in test mode, the database is not updated but Voyager produces an error log and an audit report. For information about these, see page [8-35](#).

---

### Bursar Transfer

The Bursar Transfer utility allows you to transfer information from your Voyager database to another system, such as the bursar's accounting system. All of the options available to the Bursar Transfer batch job are available on the Bursar Transfer page. See [Bursar Transfer System](#) on [page 7-1](#).



#### Procedure 30-14. Running Bursar Transfer using WebAdmin

---

Use the following to run the Bursar Transfer Utility.

1. After logging in to WebAdmin, from the main page, Circulation section, click on the **Bursar Transfer** link.

Result: The **Bursar Transfer** page opens (see [Figure 30-13](#)).

## Bursar Transfer

This utility allows you to designate patron fines, fees, and refunds for transfer. Enter the required information below and click on the submit button. When the job has completed, you will receive notification via email.

Check the "Voyager's Report Files" link for the following files created by this program:  
Note [datetime] refers to the run time of the program.

sif.burs.[datetime] : Bursar transfer data file in SIF (standard information format)

Refer to Bursar Transfer documentation for more information.

log.burs.[datetime] : Audit file containing summary information for the transfer.

err.burs.[datetime] : Error file of problems logged during processing.

Transfer fines/fees from ONLY  
these circulation locations:

A scrollable list box containing the following items: CIRC, Library, Offsite, Res, SHLD, Testcirc, media. A vertical scrollbar is on the right side of the list box.

Select All

Transfer fines/fees for ONLY  
these patron groups:

A scrollable list box containing the following items: CC, FAC, GRAD, ILL, ST, UND. The item 'GRAD' is highlighted with a blue background. A vertical scrollbar is on the right side of the list box.

Select All

- Create transfer record for each fine/fee transferred  
 Create one summary transfer record per patron

Transfer fines/fees greater than or equal to: \$  .00

Transfer fines/fees  days after fine created

Figure 30-13. Bursar Transfer Page

2. In the following fields:

- a. From the **Transfer fines/fees from ONLY these circulation locations** list, select the locations for which you want to transfer information. Click the **Select All** button to select all of the locations in the list.
- b. From the **Transfer fines/fees for ONLY these patron groups** list, select the patron groups for which you want to transfer information. Click the **Select All** button to select all of the patron groups in the list.
- c. Click the **Create transfer record for each fine/fee transferred** radio button to transfer individual fine/fee records and patron information, as well as specific information about the fines and fees.

OR

Click the **Create one summary transfer record per patron** radio button to transfer only patron information and the patron's current balance.

- d. In the **Transfer fines/fees greater than or equal to:** field, enter the minimum amount that the fine or fee must be in order to be transferred (if **Create transfer record...** is selected)

OR

enter the minimum that a fine/fee must be in order to be added to the total (if **Create one...** is selected).

- e. In the **Transfer fines/fees [ ] days after fine created** field, enter the number of days that must pass after the fine create date before the fine can be transferred (if **Create transfer record...** is selected)

OR

enter the number of days that must pass after the fine create date before the fine can be added to the total (if **Create one...** is selected).

- f. Select an operator ID from the **Operator ID** drop down list that is to be used with the postings. Defaults to Bursar.
- g. In the **E-mail Address** field, enter your e-mail address. The Utility will send an email to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
- h. When you have verified that the settings on the page are all correct, click the **Submit Bursar Transfer** button

Result: The transfer process begins. The **Job Scheduler** page displays information about the export.

3. Click the **Back** button.

Result: Returns to the main menu.

---

## Media Scheduling Utilities

On the main page, the Media Jobs Reports and Notices allows users to run all of the Media Scheduling jobs using WebAdmin.

For explanations of the various jobs see [The Media Scheduling batch jobs](#) in the *Voyager Reporter User's Guide*, and [Retain Patron IDs \(Medajob 5\)](#) on [page 9-1](#) in this user's guide.



### Procedure 30-15. Running Retain Patron IDs Media Scheduling, Medajob 5

---

Use the following steps to run Medajob 5 from Webadmin.

1. After logging in to WebAdmin, click the **Reports and Processes** link from the **Media Scheduling** section of the main screen.

Result: The Media Scheduling Reports and Processes page opens.(see [Figure 30-14](#)).

## ***Mediajob Reports and Processes***

This utility allows you to run any of the Mediajob reports and processes described in the manual. Select the mediajob you want to run, select submenu for this job(if applicable), select date range (if applicable) and press the submit button.

Select Job to Run:	
<input type="radio"/> OverdueNotices <input type="radio"/> Inventory Reports <input type="radio"/> Booking Statistics <input type="radio"/> Charge Statistics <input checked="" type="radio"/> Retain Patron IDs	<div style="margin-bottom: 10px;"> <b>SUBMENU:</b> <ul style="list-style-type: none"> <li><input type="radio"/> Date Range For All Media Policies</li> <li><input type="radio"/> All Media Policies</li> <li><input type="radio"/> Specific Media Policy</li> </ul> <p>The Specific Media Policy option <b>requires</b> Media Policy Group name</p> <p><b>Please enter the name of Media Policy</b></p> <input type="text"/> <p>A Date Range (YYYY-MM-DD) <b>is required</b> only for the task Date Range For All Media Policies option</p> <input type="text"/> - <input type="text"/> </div> <div style="margin-bottom: 10px;"> <b>SUBMENU:</b> <ul style="list-style-type: none"> <li><input type="radio"/> All Booking Desks</li> <li><input type="radio"/> Global Booking Desk</li> <li><input type="radio"/> Specific Booking Desk</li> </ul> <p>The Specific Booking Desk option <b>requires</b> a Booking Desk location</p> <p><b>Please enter the name of the Booking Desk</b></p> <input type="text"/> <p>A Date Range (YYYY-MM-DD) <b>is required</b> for all tasks in this submenu</p> <input type="text"/> - <input type="text"/> </div> <div style="margin-bottom: 10px;"> <b>SUBMENU:</b> <ul style="list-style-type: none"> <li><input type="radio"/> All Booking Desks</li> <li><input type="radio"/> Global Booking Desk</li> <li><input type="radio"/> Specific Booking Desk</li> </ul> <p>The Specific Booking Desk option <b>requires</b> a Booking Desk location</p> <p><b>Please enter the name of the Booking Desk</b></p> <input type="text"/> <p>A Date Range (YYYY-MM-DD) <b>is required</b> for all tasks in this submenu</p> <input type="text"/> - <input type="text"/> </div> <div> <p>Number of patron IDs to retain : <input type="text" value="3"/></p> </div>
Email Address: <input type="text" value="matthew.phillips@endir"/> <b>required</b> <input type="button" value="Submit Media Job"/>	

**Figure 30-14. Mediajob Reports and Processes Page**

2. To run Mediajob 5, click the **Retain Patron IDs** radio button.
3. Enter the number of patron IDs in the **Number of patron IDs to retain** field.
4. Click the **Submit Media Job** button.

Result: The job runs. The **Job Scheduler** page displays information about the job (see [Figure 30-15](#)).

---

The screenshot shows a web page titled "Job Scheduler". At the top, there is a link "Return to Main Page". Below it, a message box contains the text "Email notification to: matthew.phillips@endinfosys.com" and "job: mediajob -j 5 -n 3".

**Figure 30-15. Job Scheduler Page after Submitting the Job**

5. Click the **Back** button on your browser to return to the **Mediajob Reports and Processes** page. Otherwise, click **Return to Main Page** to return to the Webadmin Home Page.
- 

## OPAC Reports Utilities

The OPAC Reports utility allows you to run the OPAC Search logging reports.

### Popacjob

The Popacjob program is comprised of three parts.

- The first, OPAC Log Export (formerly *Popaclogexp*), pulls the specified information from the OPAC\_search\_log table in the database into a comma-delimited text file.
- The second, SDI Searches, runs search queries stored by patrons at the appropriate intervals. SDI allows patrons to choose intervals for each saved search query to be run automatically (on the local database) and the results e-mailed to them in the form of a URL.
- The third, OPAC Bib Usage Log Export (formerly performed using the -b switch when running *Popaclogexp*), pulls the specified information from the bib\_usage\_table in the database into a comma-delimited text file.

See [Popacjob on page 15-1](#) for more information.



### Procedure 30-16. Running OPAC Reports and Processes Using WebAdmin

Use the following to run the OPAC Utility.

1. After logging in to WebAdmin, from the main page, OPAC section, click on the **Reports and Processes** link.

Result: The **OPAC Reports and Processes** page opens (see [Figure 30-16](#)).

## *OPAC Reports and Processes*

This utility allows you to run any of the OPAC reports and processes described in the Accessories manual. Select the report you want to run and press the submit button.

Select Job to Run:	
<input type="radio"/> OPAC Search Log Export	<input checked="" type="radio"/> Today - <input type="text"/> days
<input type="radio"/> Bib Usage Log Export	<input type="radio"/> Date Range (YYYY-MM-DD) <input type="text"/>
Purge Log after export ?: <input type="checkbox"/>	
<input type="radio"/> SDI Searches	
<input type="radio"/> Run Jobs 1 - 3	<input checked="" type="radio"/> Today - <input type="text"/> days

**Figure 30-16. OPAC Reports and Processes Page**

2. Complete the fields on this page, see [Popacjob](#) on [page 15-1](#) for more information.
3. In the **E-mail Address** field, enter your e-mail address. The Utility will send an email to that address when the job is done processing. You must enter the entire email address (user@hostname.com) in order for the e-mail to be sent.
4. When you have verified that the settings on the page are all correct, click the **Submit OPAC Job** button

Result: The transfer process begins. The **Job Scheduler** page displays information about the job.

5. Click the **Back** button.

Result: Returns to the main menu.

---

## System Admin Utilities

On the main page, the System Administrative utilities available are directory listings of the log files and of the report files.

### Log Files Utility

The Voyager Log Files page provides you with a list of your batch job log files. This page lists all of the files that exist in the log directory on your server. This directory is commonly the `/m1/voyager/xxxdb/log` directory. You can view any of the files in the directory.



#### Procedure 30-17. Accessing Log Files on the Server Using WebAdmin

---

Use the following to access the log files:

1. After logging in to WebAdmin, from the main page, System Admin section, click the **Log Files** link.

Result: The **Voyager Log Files** page opens.

2. Double-click the file that you want to view.

Result: The log file will display on the screen.

3. When you are done viewing the file, click the **Back** button twice.

Result: Returns to the log file list, then to the main menu.

---

### Report Files Utility

The Voyager Report Files page provides you with a list of your batch job report files. This page lists all of the files that exist in the report directory on your server. This directory is commonly the `/m1/voyager/xxxdb/rpt` directory. You can view any of the files in the directory.



---

**Procedure 30-18. Accessing Report Files on the Server Using Webadmin**

---

Use the following to access the log files.

1. After logging in to WebAdmin, from the main page, System Admin section, click the **Report Files** link.

Result: The **Voyager Report Files** page opens.

2. Double-click the file that you want to view.

Result: The file opens.

3. When you are done viewing the file, click the **Back** button twice.

Result: Returns to the report file list, then to the main menu.

---



---

## Data Dictionary

# A

---

### **ACCESS\_GROUP**

ACCESS\_GROUP\_ID: NUMBER  
ACCESS\_GROUP\_CODE: VARCHAR2(8)  
ACCESS\_GROUP\_NAME: VARCHAR2(40)  
ACCESS\_GROUP\_TYPE: CHAR(1)

### **ACCESS\_GROUP\_DATABASES**

ACCESS\_GROUP\_ID: NUMBER  
DB\_CODE: VARCHAR2(8)  
DB\_ID: NUMBER

### **ACCESS\_GROUP\_DOMAIN**

ACCESS\_GROUP\_DOMAIN\_ID: NUMBER  
ACCESS\_GROUP\_ID: NUMBER  
DOMAIN\_NAME: VARCHAR2(256)

### **ACCESS\_GROUP\_IP**

ACCESS\_GROUP\_IP\_ID: NUMBER  
ACCESS\_GROUP\_ID: NUMBER  
MIN\_IP\_ADDR: NUMBER  
MAX\_IP\_ADDR: NUMBER

### **ACCESS\_GROUP\_PATRON\_GROUP**

ACCESS\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER

### **ACCESS\_GROUP\_SORT\_GROUP**

ACCESS\_GROUP\_ID: NUMBER  
SORT\_GROUP\_ID: NUMBER

**ACCOUNT\_LOCATION**

ACCOUNT\_ID: NUMBER  
ACCOUNT\_LOCATION: NUMBER

**ACCOUNT\_NOTE**

ACCOUNT\_ID: NUMBER  
VENDOR\_ID: NUMBER  
NOTE: VARCHAR2(1900)

**ACQ\_LOCATIONS**

ACQ\_POLICY\_ID: NUMBER  
LOCATION\_ID: NUMBER  
ORDER\_LOC: CHAR(1)  
RECEIVE\_LOC: CHAR(1)  
DESTINATION\_LOC: CHAR(1)  
ORDER\_OPAC: CHAR(1)  
ORDER\_DEFAULT\_ITEM\_TYPE: NUMBER  
RECEIVE\_DEFAULT\_ITEM\_TYPE: NUMBER  
PRINT\_LOCATION: NUMBER

**ACQ\_OPERATOR**

OPERATOR\_ID: VARCHAR2(10)  
ACQ\_PROFILE\_ID: NUMBER

**ACQ\_POLICY\_GROUP**

ACQ\_POLICY\_ID: NUMBER  
ACQ\_POLICY\_NAME: VARCHAR2(40)  
DUP\_PROFILE\_ID: NUMBER

**ACQ\_PROFILE**

ACQ\_PROFILE\_ID: NUMBER  
ACQ\_PROFILE\_NAME: VARCHAR2(25)  
ORDER\_ADD\_UPDATE: CHAR(1)  
ORDER\_APPROVE: CHAR(1)  
OVERRIDE\_COMMIT: CHAR(1)  
OVERRIDE\_EXPEND: CHAR(1)  
RECEIVE: CHAR(1)  
SERIAL\_CHECKIN: CHAR(1)  
ORDER\_VIEW\_ONLY: CHAR(1)  
INVOICE\_ADD\_UPDATE: CHAR(1)  
INVOICE\_APPROVE: CHAR(1)  
INVOICE\_VIEW\_ONLY: CHAR(1)  
LEDGER\_ADD\_UPDATE: CHAR(1)  
CHANGE\_FUND\_ALLOC: CHAR(1)  
LEDGER\_VIEW\_ONLY: CHAR(1)  
FISCAL\_CLOSE: CHAR(1)  
VENDOR\_ADD\_UPDATE: CHAR(1)  
VENDOR\_VIEW\_ONLY: CHAR(1)  
MONO CLAIMS: CHAR(1)  
SERIAL CLAIMS: CHAR(1)

PATTERN\_ADD\_UPDATE: CHAR(1)  
PATTERN\_VIEW\_ONLY: CHAR(1)  
ORDER\_DELETE: CHAR(1)  
INVOICE\_DELETE: CHAR(1)  
LEDGER\_DELETE: CHAR(1)  
PATTERN\_DELETE: CHAR(1)  
VENDOR\_DELETE: CHAR(1)  
CURRENCY\_MAINTENANCE: CHAR(1)  
EDI\_INCOMING: CHAR(1)  
EDI\_OUTGOING: CHAR(1)  
HOLD\_IGNORE\_OWNERSHIP: CHAR(1)  
MODIFY\_EDI\_OUTGOING: CHAR(1)  
ITEM\_ADD\_UPDATE: CHAR(1)  
ITEM\_VIEW\_ONLY: CHAR(1)  
ITEM\_DELETE: CHAR(1)  
SERIALS\_VIEW\_ONLY: CHAR(1)  
PROBLEMS CLAIMS\_VIEW\_ONLY: CHAR(1)  
BIND\_VIEW\_VOL\_ISSUE: CHAR(1)  
BIND\_EDIT\_VOL: CHAR(1)  
BIND\_EDIT\_ISSUE: CHAR(1)  
BIND\_PRINT: CHAR(1)

**ACQ\_SECURITY\_LOCS**

ACQ\_PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER

**ACTION\_TYPE**

ACTION\_TYPE\_ID: NUMBER  
ACTION\_TYPE: VARCHAR2(20)

**ADDRESS\_TYPE**

ADDRESS\_TYPE: NUMBER  
ADDRESS\_DESC: VARCHAR2(25)

**ADJUST\_REASON**

REASON\_ID: NUMBER  
REASON\_TEXT: VARCHAR2(50)  
CHARGE\_OR\_CREDIT: CHAR(1)  
REASON\_EDI\_CODE: VARCHAR2(250)  
VENDOR\_ID: NUMBER

**AUTH\_DATA**

AUTH\_ID: NUMBER  
SEQNUM: NUMBER  
RECORD\_SEGMENT: VARCHAR2(990)

**AUTH\_HEADING**

HEADING\_ID\_POINTER: NUMBER  
HEADING\_ID\_POINTEE: NUMBER  
AUTH\_ID: NUMBER  
REFERENCE\_TYPE: CHAR(1)  
DISPLAY\_HEADING: VARCHAR2(330)  
SCOPE\_NOTE\_PRESENT: CHAR(1)

**AUTH\_HISTORY**

AUTH\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
ACTION\_DATE: DATE  
LOCATION\_ID: NUMBER  
ENCODING\_LEVEL: CHAR(1)  
ACTION\_TYPE\_ID: NUMBER

**AUTH\_INDEX**

AUTH\_ID: NUMBER  
INDEX\_CODE: CHAR(4)  
NORMAL\_HEADING: VARCHAR2(150)  
DISPLAY\_HEADING: VARCHAR2(150)

**AUTH\_MASTER**

AUTH\_ID: NUMBER  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE  
EXPORT\_OK: CHAR(1)  
EXPORT\_OK\_DATE: DATE  
EXPORT\_OK\_OPID: VARCHAR2(10)  
EXPORT\_OK\_LOCATION\_ID: NUMBER  
EXPORT\_DATE: DATE

**AUTH\_SUBDIVISION**

SUBDIV\_ID\_POINTER: NUMBER  
SUBDIV\_ID\_POINTEE: NUMBER  
REFERENCE\_TYPE: CHAR(1)  
AUTH\_ID: NUMBER  
DISPLAY\_SUBDIV: VARCHAR2(330)

**AUTHBLOB\_VW**

AUTH\_ID: NUMBER  
MARC\_RECORD: CLOB

**AUTHHEADING\_VW**

HEADING\_ID\_POINTER: NUMBER  
HEADING\_ID\_POINTEE: NUMBER  
AUTH\_ID: NUMBER  
REFERENCE\_TYPE: VARCHAR2(20)

**AUTHHISTORY\_VW**

AUTH\_ID: NUMBER  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
CREATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPERATOR\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE

UPDATE\_LOCATION\_ID: NUMBER

**AUTHORITY1XX4XX\_VW**

INDEX\_TYPE: VARCHAR2(10)  
AUTH\_ID\_1XX: NUMBER  
AUTH\_ID\_4XX: NUMBER  
STAFFBIBS: NUMBER  
OPACBIBS: NUMBER  
DISPLAY\_HEADING: VARCHAR2(330)

**AUTHORITY5XX1XX\_VW**

INDEX\_TYPE: VARCHAR2(10)  
AUTH\_ID\_5XX: NUMBER  
DISPLAY\_HEADING: VARCHAR2(300)

**AUTHORITYDUPE\_VW**

AUTH\_ID: NUMBER  
DISPLAY\_HEADING: VARCHAR2(330)

**AUTHORITYRECORDS\_VW**

AUTH\_ID: NUMBER  
INDEX\_TYPE: VARCHAR2(10)  
REFERENCE\_TYPE\_DESC: VARCHAR2(20)  
DISPLAY\_HEADING: VARCHAR2(300)  
NORMAL\_HEADING: VARCHAR2(300)

**BASE\_CURRENCY**

BASE\_COUNTRY\_NAME: VARCHAR2(25)  
BASE\_CURRENCY\_NAME: VARCHAR2(25)  
BASE\_CURRENCY\_CODE: VARCHAR2(3)  
BASE\_DECIMALS: NUMBER  
DECIMAL\_DELIMITER: CHAR(1)

**BIB\_DATA**

BIB\_ID: NUMBER  
SEQNUM: NUMBER  
RECORD\_SEGMENT: VARCHAR2(990)

**BIB\_FACET**

BIB\_ID: NUMBER  
INDEX\_CODE: CHAR(4)  
FACET1: VARCHAR2(20)  
FACET2: VARCHAR2(20)  
FACET3: VARCHAR2(20)

**BIB\_FORMAT\_DISPLAY**

BIB\_FORMAT: VARCHAR2(2)  
BIB\_FORMAT\_DISPLAY: VARCHAR2(20)

**BIB\_HEADING**

HEADING\_ID: NUMBER

BIB\_ID: NUMBER  
DISPLAY\_HEADING: VARCHAR2(330)  
SUPPRESS\_IN\_OPAC: CHAR(1)

**BIB\_HISTORY**

BIB\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
ACTION\_DATE: DATE  
LOCATION\_ID: NUMBER  
ENCODING\_LEVEL: CHAR(1)  
SUPPRESS\_IN\_OPAC: VARCHAR2(1)  
ACTION\_TYPE\_ID: NUMBER

**BIB\_INDEX**

BIB\_ID: NUMBER  
INDEX\_CODE: CHAR(4)  
NORMAL\_HEADING: VARCHAR2(150)  
DISPLAY\_HEADING: VARCHAR2(150)

**BIB\_ITEM**

BIB\_ID: NUMBER  
ITEM\_ID: NUMBER  
ADD\_DATE: DATE  
OPERATOR\_ID: VARCHAR2(10)

**BIB\_LOCATION**

BIB\_ID: NUMBER  
LOCATION\_ID: NUMBER

**BIB\_MASTER**

BIB\_ID: NUMBER  
LIBRARY\_ID: NUMBER  
SUPPRESS\_IN\_OPAC: CHAR(1)  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE  
EXPORT\_OK: CHAR(1)  
EXPORT\_OK\_DATE: DATE  
EXPORT\_OK\_OPID: VARCHAR2(10)  
EXPORT\_OK\_LOCATION\_ID: NUMBER  
EXPORT\_DATE: DATE

**BIB\_MEDIUM**

MEDIUM: CHAR(1)  
BIB\_ID: NUMBER

**BIB\_MFHD**

BIB\_ID: NUMBER  
MFHD\_ID: NUMBER

**BIB\_SUBDIVISION**

BIB\_ID: NUMBER  
SUBDIV\_ID: NUMBER  
DISPLAY\_SUBDIV: VARCHAR2(330)

**BIB\_TEXT**

BIB\_ID: NUMBER  
AUTHOR: VARCHAR2(255)  
TITLE: VARCHAR2(255)  
TITLE\_BRIEF: VARCHAR2(150)  
UNIFORM\_TITLE: VARCHAR2(255)  
EDITION: VARCHAR2(100)  
ISBN: VARCHAR2(50)  
ISSN: VARCHAR2(20)  
LCCN: VARCHAR2(20)  
NETWORK\_NUMBER: VARCHAR2(30)  
SERIES: VARCHAR2(255)  
CODEN: VARCHAR2(6)  
GPONUM: VARCHAR2(20)  
STDTECH: VARCHAR2(30)  
OTHER\_STD\_NUM: VARCHAR2(30)  
BEGIN\_PUB\_DATE: VARCHAR2(4)  
END\_PUB\_DATE: VARCHAR2(4)  
PUB\_DATES\_COMBINED: VARCHAR2(9)  
PUBLISHER\_DATE: VARCHAR2(25)  
PUB\_PLACE: VARCHAR2(100)  
PUBLISHER: VARCHAR2(150)  
PUBLISHER\_NUMBER: VARCHAR2(40)  
IMPRINT: VARCHAR2(200)  
LANGUAGE: VARCHAR2(3)  
BIB\_FORMAT: VARCHAR2(2)  
RECORD\_STATUS: VARCHAR2(1)  
ENCODING\_LEVEL: VARCHAR2(1)  
DESCRIP\_FORM: VARCHAR2(1)  
FIELD\_008: VARCHAR2(40)  
PLACE\_CODE: VARCHAR2(3)  
DATE\_TYPE\_STATUS: CHAR(1)  
MAP\_PROJECTION: CHAR(2)  
MAP\_MATH\_DATA: VARCHAR2(255)  
STOCK\_NUMBER: VARCHAR2(50)

**BIB\_TEXT\_DISPLAYFIELD**

BIB\_TEXT\_FIELD: VARCHAR2(30)  
DISPLAY\_NAME: VARCHAR2(40)

**BIB\_USAGE\_LOG**

CLIENT\_TYPE: CHAR(1)  
USE\_DATE: DATE  
OPERATOR\_ID: VARCHAR2(10)  
LOCATION\_ID: NUMBER  
SESSION\_ID: VARCHAR2(16)  
STAT\_STRING: VARCHAR2(15)  
CLIENT\_IP: VARCHAR2(15)  
BIB\_ID: NUMBER  
USE\_TYPE: CHAR(1)

**BIB\_VW**

TITLE: VARCHAR2(150)  
SORT\_TITLE: VARCHAR2(150)  
BIB\_ID: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OPERATOR: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
MFHD\_ID: NUMBER  
CALL\_NO: VARCHAR2(300)  
CALL\_NO\_TYPE: CHAR(1)  
NORMALIZED\_CALL\_NO: VARCHAR2(300)  
MFHD\_CREATE\_DATE: DATE  
MFHD\_CREATE\_OPERATOR: VARCHAR2(10)  
MFHD\_CREATE\_LOCATION\_ID: NUMBER  
MFHD\_LOCATION\_ID: NUMBER  
MFHD\_LOCATION\_CODE: VARCHAR2(10)  
MFHD\_LOCATION: VARCHAR2(25)

**BIBBLOB\_VW**

BIB\_ID: NUMBER  
MARC\_RECORD: CLOB

**BIBCOMPOSITEINDEX\_VW**

COMPOSITE\_SEARCHCODE: CHAR(4)  
COMPONENT\_SEARCHCODE: CHAR(4)

**BIBHISTORY\_VW**

BIB\_ID: NUMBER  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
CREATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPERATOR\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_LOCATION\_ID: NUMBER

**BIBLOC\_VW**

BIB\_ID: NUMBER  
MARCLOC\_CODE: VARCHAR2(3)

**BIBSORTING\_VW**

BIB\_ID: NUMBER  
DISPLAY\_TITLE: VARCHAR2(150)  
NORMAL\_TITLE: VARCHAR2(150)  
DISPLAY\_AUTHOR: VARCHAR2(150)  
NORMAL\_AUTHOR: VARCHAR2(150)  
PUB\_DATE: VARCHAR2(4)

**BINDERY\_COPY**

BINDERY\_COPY\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COPY\_ID: NUMBER

**BINDERY\_COPY\_DATA**

BINDERY\_DATA\_ID: NUMBER  
BINDERY\_COPY\_ID: NUMBER  
BINDERY\_DATA\_TYPE\_ID: NUMBER  
BINDERY\_DATA: VARCHAR2(1000)

**BINDERY\_COPY\_DATA\_TYPE**

BINDERY\_DATA\_TYPE\_ID: NUMBER  
BINDERY\_COPY\_DATA\_TYPE\_DESC: VARCHAR2(25)

**BINDERY\_VOLUME**

BINDERY\_VOLUME\_ID: NUMBER  
BINDERY\_COPY\_ID: NUMBER  
BIND\_ON\_DATE: DATE  
ITEM\_ID: NUMBER  
ITEM\_ENUM: VARCHAR2(80)  
CHRON: VARCHAR2(80)  
YEAR: VARCHAR2(20)  
CAPTION: VARCHAR2(256)  
FREETEXT: VARCHAR2(256)  
VOLUME\_NOTE: VARCHAR2(200)  
OTHER\_VOLUME\_DATA: VARCHAR2(200)

**BINDERY\_VOLUME\_ISSUES**

BINDERY\_VOLUME\_ID: NUMBER  
ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COPY\_ID: NUMBER  
SEQUENCE\_NUMBER: NUMBER

**BOOKING\_RESULT**

BOOKING\_RESULT\_ID: NUMBER  
BOOKING\_RESULT: VARCHAR2(20)

**BROWSE\_STATS**

STAT\_ID: NUMBER  
STAT\_SAMPLE: VARCHAR2(50)  
STAT\_TYPE: CHAR(1)  
SUB\_TYPE: CHAR(1)  
SUBSUB\_TYPE: NUMBER

**CACHE\_MAPS**

CACHE\_ID: NUMBER  
CODE: CHAR(4)  
ARG: VARCHAR2(60)  
DATE\_UPDATED: DATE  
BITS: NUMBER  
SEGSIZE: NUMBER

**CACHE\_SEGS**

CACHE\_ID: NUMBER  
SEQNUM: NUMBER  
RECORD\_SEGMENT: LONG RAW

## **CALENDAR**

CALENDAR\_ID: NUMBER  
CALENDAR\_BEGIN\_DATE: DATE  
CALENDAR\_END\_DATE: DATE  
FIXED\_DUE\_DATE: DATE  
CALENDAR\_DESC: VARCHAR2(25)  
SUNDAY\_OPEN: CHAR(1)  
SUNDAY\_OPENHOUR: NUMBER  
SUNDAY\_CLOSEHOUR: NUMBER  
MONDAY\_OPEN: CHAR(1)  
MONDAY\_OPENHOUR: NUMBER  
MONDAY\_CLOSEHOUR: NUMBER  
TUESDAY\_OPEN: CHAR(1)  
TUESDAY\_OPENHOUR: NUMBER  
TUESDAY\_CLOSEHOUR: NUMBER  
WEDNESDAY\_OPEN: CHAR(1)  
WEDNESDAY\_OPENHOUR: NUMBER  
WEDNESDAY\_CLOSEHOUR: NUMBER  
THURSDAY\_OPEN: CHAR(1)  
THURSDAY\_OPENHOUR: NUMBER  
THURSDAY\_CLOSEHOUR: NUMBER  
FRIDAY\_OPEN: CHAR(1)  
FRIDAY\_OPENHOUR: NUMBER  
FRIDAY\_CLOSEHOUR: NUMBER  
SATURDAY\_OPEN: CHAR(1)  
SATURDAY\_OPENHOUR: NUMBER  
SATURDAY\_CLOSEHOUR: NUMBER  
CIRC\_CLUSTER\_ID: NUMBER  
MONDAY\_HOURLY\_EFFECT: NUMBER  
MONDAY\_LOAN\_DUE: NUMBER  
TUESDAY\_HOURLY\_EFFECT: NUMBER  
TUESDAY\_LOAN\_DUE: NUMBER  
WEDNESDAY\_HOURLY\_EFFECT: NUMBER  
WEDNESDAY\_LOAN\_DUE: NUMBER  
THURSDAY\_HOURLY\_EFFECT: NUMBER  
THURSDAY\_LOAN\_DUE: NUMBER  
FRIDAY\_HOURLY\_EFFECT: NUMBER  
FRIDAY\_LOAN\_DUE: NUMBER  
SATURDAY\_HOURLY\_EFFECT: NUMBER  
SATURDAY\_LOAN\_DUE: NUMBER  
SUNDAY\_HOURLY\_EFFECT: NUMBER  
SUNDAY\_LOAN\_DUE: NUMBER

## **CALENDAR\_TERM\_DATE**

CALENDAR\_ID: NUMBER  
LEAD\_DAYS: NUMBER  
END\_OF\_TERM\_DATE: DATE

## **CALL\_NO\_HIERARCHY**

CALL\_NO\_HIERARCHY\_ID: NUMBER  
CALL\_NO\_TYPE: CHAR(1)  
CODE: VARCHAR2(8)

NAME: VARCHAR2(25)  
USE\_AS\_DEFAULT: CHAR(1)

**CALL\_NO\_TYPE**

CALL\_NO\_TYPE: CHAR(1)  
CALL\_NO\_DESC: VARCHAR2(25)  
INDEXRULES: VARCHAR2(300)  
CALL\_NO\_CODE: VARCHAR2(16)  
MAP\_CODE: CHAR(1)

**CALL\_SLIP**

CALL\_SLIP\_ID: NUMBER  
PRINT\_GROUP\_ID: NUMBER  
BIB\_ID: NUMBER  
ITEM\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
DATE\_REQUESTED: DATE  
DATE\_PROCESSED: DATE  
LOCATION\_ID: NUMBER  
STATUS: NUMBER  
STATUS\_DATE: DATE  
STATUS\_OPID: VARCHAR2(10)  
NO\_FILL\_REASON: NUMBER  
ITEM\_YEAR: VARCHAR2(20)  
ITEM\_ENUM: VARCHAR2(80)  
ITEM\_CHRON: VARCHAR2(80)  
NOTE: VARCHAR2(100)  
MFHD\_ID: NUMBER  
PICKUP\_LOCATION\_ID: NUMBER  
PICKUP\_DB\_ID: NUMBER  
PATRON\_DB\_ID: NUMBER  
NOT\_NEEDED\_AFTER: NUMBER  
REPLY\_NOTE: VARCHAR2(100)

**CALL\_SLIP\_ARCHIVE**

ARCHIVE\_ID: NUMBER  
PRINT\_GROUP\_ID: NUMBER  
BIB\_ID: NUMBER  
ITEM\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
DATE\_REQUESTED: DATE  
DATE\_PROCESSED: DATE  
LOCATION\_ID: NUMBER  
STATUS: NUMBER  
STATUS\_DATE: DATE  
STATUS\_OPID: VARCHAR2(10)  
NO\_FILL\_REASON: NUMBER  
ITEM\_YEAR: VARCHAR2(20)  
ITEM\_ENUM: VARCHAR2(80)  
ITEM\_CHRON: VARCHAR2(80)

NOTE: VARCHAR2(100)  
MFHD\_ID: NUMBER  
PICKUP\_LOCATION\_ID: NUMBER  
PICKUP\_DB\_ID: NUMBER  
PATRON\_DB\_ID: NUMBER  
NOT\_NEEDED\_AFTER: NUMBER  
REPLY\_NOTE: VARCHAR2(100)

**CALL\_SLIP\_GROUP\_LOCATION**

GROUP\_ID: NUMBER  
LOCATION\_ID: NUMBER  
RULE\_RANK: NUMBER  
CALL\_NO\_TYPE: CHAR(1)  
CALL\_NO\_MIN\_NORM: VARCHAR2(112)  
CALL\_NO\_MIN\_DISPLAY: VARCHAR2(144)  
CALL\_NO\_MAX\_NORM: VARCHAR2(112)  
CALL\_NO\_MAX\_DISPLAY: VARCHAR2(144)  
PERM\_LOCATION: NUMBER  
TEMP\_LOCATION: NUMBER  
ITEM\_TYPE\_ID: NUMBER  
TEMP\_ITEM\_TYPE\_ID: NUMBER  
YEAR\_MIN: VARCHAR2(20)  
YEAR\_MAX: VARCHAR2(20)

**CALL\_SLIP\_MSG**

MESSAGE\_ID: NUMBER  
MESSAGE\_CODE: VARCHAR2(10)  
MESSAGE\_NAME: VARCHAR2(25)  
SUSPENSION\_MESSAGE: CHAR(1)  
ACTIVE: CHAR(1)

**CALL\_SLIP\_PRINT\_GROUP**

GROUP\_ID: NUMBER  
GROUP\_CODE: VARCHAR2(10)  
GROUP\_NAME: VARCHAR2(25)  
DEFAULT\_GROUP: CHAR(1)  
PROCESS\_METHOD: CHAR(1)  
LOCATION\_ID: NUMBER  
ARCHIVE\_INTERVAL: CHAR(1)  
ARCHIVE\_PERIOD: NUMBER  
EXPIRE\_INTERVAL: CHAR(1)  
EXPIRE\_PERIOD: NUMBER  
PATRON\_INFO: CHAR(1)  
CAT REVIEW: CHAR(1)  
CIRC REVIEW: CHAR(1)  
DEFAULT\_ITEM\_TYPE\_ID: NUMBER

**CALL\_SLIP\_REASSIGN\_ARCHIVE**

ARCHIVE\_ID: NUMBER  
PRINT\_GROUP\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
REASSIGN\_DATE: DATE

**CALL\_SLIP\_REASSIGNMENT**

CALL\_SLIP\_ID: NUMBER  
PRINT\_GROUP\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
REASSIGN\_DATE: DATE

**CALL\_SLIP\_STATS**

CALL\_SLIP\_ID: NUMBER  
PATRON\_STAT\_ID: NUMBER

**CALL\_SLIP\_STATUS\_TYPE**

STATUS\_TYPE: NUMBER  
STATUS\_DESC: VARCHAR2(25)

**CAMBRIDGEDEPTCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(6)

**CAMBRIDGEMAINCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(6)

**CAMBRIDGEMEDICALCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(6)

**CAT\_CONTROL\_BARCODE**

IMPORT\_RULE\_ID: NUMBER  
INDICATOR1: CHAR(1)  
INDICATOR2: CHAR(1)  
FIELD: VARCHAR2(3)  
SUBFIELD: CHAR(1)  
SEQUENCE: NUMBER

**CAT\_CONTROL\_CALL\_NO**

CALL\_NO\_HIERARCHY\_ID: NUMBER  
INDICATOR1: CHAR(1)  
INDICATOR2: CHAR(1)  
FIELD: VARCHAR2(3)  
MAIN\_SUBFIELD: CHAR(1)  
CUTTER\_SUBFIELD: CHAR(1)  
SEQUENCE: NUMBER

**CAT\_CONTROL\_ITEM\_TYPE**

IMPORT\_RULE\_ID: NUMBER  
INDICATOR1: CHAR(1)  
INDICATOR2: CHAR(1)  
FIELD: VARCHAR2(3)  
SUBFIELD: CHAR(1)  
SEQUENCE: NUMBER  
FIXED\_START: NUMBER  
FIXED\_END: NUMBER

**CAT\_OPERATOR**

OPERATOR\_ID: VARCHAR2(10)  
CAT\_PROFILE\_ID: NUMBER

**CAT\_POLICY\_DUP**

CAT\_POLICY\_ID: NUMBER  
DUP\_PROFILE\_ID: NUMBER

**CAT\_POLICY\_GROUP**

CAT\_POLICY\_ID: NUMBER  
CAT\_POLICY\_NAME: VARCHAR2(40)  
NUC\_CODE: VARCHAR2(15)  
OPAC\_DISPLAY: CHAR(1)

**CAT\_POLICY\_HIERARCHY**

CAT\_POLICY\_ID: NUMBER  
CALL\_NO\_HIERARCHY\_ID: NUMBER

**CAT\_POLICY\_LOCS**

CAT\_GROUP\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CATALOGING\_LOCATION: CHAR(1)  
CIRC\_LOCATION: CHAR(1)  
ROUTING\_LOCATION: CHAR(1)  
CALL\_NO\_TYPE: CHAR(1)  
NUC\_CODE: VARCHAR2(15)  
DEFAULT\_ITEM\_TYPE: NUMBER

**CAT\_PROFILE**

CAT\_PROFILE\_ID: NUMBER  
CAT\_PROFILE\_NAME: VARCHAR2(25)  
BIB\_ADD: CHAR(1)  
BIB\_UPDATE: CHAR(1)  
BIB\_DELETE: CHAR(1)  
BIB\_VIEW\_ONLY: CHAR(1)  
HOLD\_ADD: CHAR(1)  
HOLD\_UPDATE: CHAR(1)  
HOLD\_DELETE: CHAR(1)  
HOLD\_VIEW\_ONLY: CHAR(1)  
ITEM\_ADD: CHAR(1)  
ITEM\_UPDATE: CHAR(1)  
ITEM\_DELETE: CHAR(1)  
ITEM\_VIEW\_ONLY: CHAR(1)  
AUTH\_ADD: CHAR(1)  
AUTH\_UPDATE: CHAR(1)  
AUTH\_DELETE: CHAR(1)  
AUTH\_VIEW\_ONLY: CHAR(1)  
MARCAUTH\_ADD\_UPDATE: CHAR(1)  
MARCAUTH\_VIEW\_ONLY: CHAR(1)  
MARCBIB\_ADD\_UPDATE: CHAR(1)  
MARCBIB\_VIEW\_ONLY: CHAR(1)  
MARCHOLD\_ADD\_UPDATE: CHAR(1)  
MARCHOLD\_VIEW\_ONLY: CHAR(1)  
GLOBAL\_REPLACE: CHAR(1)

CHANGE\_OWNERSHIP: CHAR(1)  
HOLD\_IGNORE\_OWNERSHIP: CHAR(1)  
AUTH\_EXPORT\_OK: CHAR(1)  
BIB\_EXPORT\_OK: CHAR(1)  
MFHD\_EXPORT\_OK: CHAR(1)  
USE\_TEMPLATE: CHAR(1)

**CAT\_SECURITY\_LOCS**

CAT\_PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER

**CHARACTER\_SET**

CHAR\_SET\_ID: NUMBER  
CHAR\_SET\_CODE: CHAR(1)  
CHAR\_SET\_NAME: VARCHAR2(30)

**CHRON**

CHRON\_TYPE\_ID: NUMBER  
CHRON\_SEQ: NUMBER  
CHRON\_VALUE: VARCHAR2(20)

**CHRON\_TYPE**

CHRON\_TYPE\_ID: NUMBER  
CHRON\_NAME: VARCHAR2(40)  
CHRON\_TYPE\_CODE: CHAR(2)

**CIRC\_ALERT\_CONDITIONS**

LOC\_ID: NUMBER  
ALERT\_ID: NUMBER  
ALERT\_DISPLAY: NUMBER  
ALERT\_USE\_PATRON\_NAME: CHAR(1)  
ALERT\_USE\_PATRON\_BARCODE: CHAR(1)  
ALERT\_USE\_PATRON\_PHONE: CHAR(1)

**CIRC\_ALERT\_TYPES**

ALERT\_TYPE: NUMBER  
ALERT\_TYPE\_DESC: VARCHAR2(100)

**CIRC\_ALERTS**

ALERT\_ID: NUMBER  
ALERT\_TYPE: NUMBER  
ALERT\_NAME: VARCHAR2(30)  
ALERT\_TEXT: VARCHAR2(100)

**CIRC\_BLOCK\_OVERRIDE**

CIRC\_PROFILE\_ID: NUMBER  
BLOCK\_ID: NUMBER

**CIRC\_BLOCKS**

BLOCK\_ID: NUMBER  
BLOCK\_TYPE: VARCHAR2(6)  
BLOCK\_NAME: VARCHAR2(30)  
BLOCK\_DISPLAY\_NAME: VARCHAR2(100)

**CIRC\_CLUSTER**

CIRC\_CLUSTER\_ID: NUMBER  
CIRC\_CLUSTER\_CODE: VARCHAR2(10)  
CIRC\_CLUSTER\_NAME: VARCHAR2(100)  
DEFAULT\_PICKUP\_LOCATION: NUMBER

**CIRC\_GROUP\_CALENDAR**

CIRC\_GROUP\_ID: NUMBER  
CALENDAR\_ID: NUMBER

**CIRC\_OPERATOR**

OPERATOR\_ID: VARCHAR2(10)  
CIRC\_PROFILE\_ID: NUMBER

**CIRC\_POLICY\_GROUP**

CIRC\_GROUP\_ID: NUMBER  
CIRC\_GROUP\_NAME: VARCHAR2(40)  
RENEW\_IF\_RECALL: CHAR(1)  
RENEW\_IF\_HOLD: CHAR(1)  
RENEW\_IF\_OVERDUE: CHAR(1)  
CLOSED\_DAYS\_FOR\_LOANS: CHAR(1)  
CLOSED\_DAYS\_FOR\_FINES: CHAR(1)  
LOST\_PROCESS\_FEE: CHAR(1)  
PROCESS\_FEE: NUMBER  
MAX\_FINE\_FEE\_FOR\_LOST: CHAR(1)  
EARLY\_PICKUP\_WINDOW: NUMBER  
UNCLAIMED\_INTERVAL: NUMBER  
EXTEND\_RECALL\_DUE\_DATE: CHAR(1)  
CIRC\_CLUSTER\_ID: NUMBER  
FIXED\_DUE\_TIME: CHAR(1)

**CIRC\_POLICY\_LOCS**

CIRC\_GROUP\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CIRC\_LOCATION: CHAR(1)  
COLLECT\_FINES: CHAR(1)  
COURTESY\_DISCHARGE: CHAR(1)  
SHELVING\_INTERVAL: CHAR(1)  
SHELVING\_PERIOD: NUMBER  
SUPPRESS\_FLY\_ITEMS: CHAR(1)  
DEFAULT\_ITEM\_TYPE: NUMBER  
DEFAULT\_LOCATION: NUMBER  
PRINT\_DATE\_DUES: CHAR(1)  
PRINT\_FINE\_RECEIPTS: CHAR(1)  
PRINT\_DISCHARGE\_RECEIPTS: CHAR(1)  
PRINT\_HOLD\_SLIPS: CHAR(1)  
PRINT\_ROUTING\_SLIPS: CHAR(1)  
TRANSIT\_PERIOD: NUMBER  
PRINT\_LOCATION: NUMBER  
AUTOMATED\_STORAGE: CHAR(1)  
OPAC\_CIRC\_DESK: CHAR(1)  
PICKUP\_LOCATION: CHAR(1)

HOLD\_LIFE: NUMBER  
RECALL\_LIFE: NUMBER  
PATRON\_AVAIL\_ITEMS\_ALERT: CHAR(1)  
PATRON\_FINE\_FEE\_ALERT: CHAR(1)

**CIRC\_POLICY\_MATRIX**

CIRC\_POLICY\_MATRIX\_ID: NUMBER  
CIRC\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
ITEM\_TYPE\_ID: NUMBER  
CHARGE\_RENEW: CHAR(1)  
PLACE\_RECALL: CHAR(1)  
PLACE\_HOLD: CHAR(1)  
LOAN\_INTERVAL: CHAR(1)  
LOAN\_PERIOD: NUMBER  
FINE\_INTERVAL: CHAR(1)  
FINE\_RATE: NUMBER  
MAX\_FINE: NUMBER  
GRACE\_PERIOD: NUMBER  
RENEWAL\_COUNT: NUMBER  
RENEWAL\_INTERVAL: CHAR(1)  
RENEWAL\_PERIOD: NUMBER  
RECALL\_FINE\_INTERVAL: CHAR(1)  
RECALL\_FINE\_RATE: NUMBER  
RECALL\_MIN\_LOAN: NUMBER  
RENEW\_FROM\_DUE\_DATE: CHAR(1)  
RECALL\_RETURN\_PERIOD: NUMBER  
PLACE\_CALL\_SLIP: CHAR(1)  
MAX\_RECALL\_FINE: NUMBER  
RECALL\_GRACE\_PERIOD: NUMBER  
COURTESY\_NOTICE\_INTERVAL: NUMBER  
COURTESY\_NOTICE\_MIN\_LOAN: NUMBER  
FIRST\_OVERDUE\_INTERVAL: NUMBER  
OTHER\_NOTICE\_COUNT: NUMBER  
OTHER\_NOTICE\_INTERVAL: NUMBER  
LOST\_NOTICE\_INTERVAL: NUMBER  
RECALL\_NOTICE\_INTERVAL: NUMBER  
RECALL\_NOTICE\_COUNT: NUMBER  
HOLD\_SHELF\_LIFE: NUMBER  
HOLD\_SHELF\_LIFE\_INTERVAL: CHAR(1)  
CHARGE\_LIMIT: NUMBER  
CHARGE\_LIMIT\_APPLY: CHAR(1)  
PLACE\_UB\_REQUEST: CHAR(1)

**CIRC\_PROFILE**

CIRC\_PROFILE\_ID: NUMBER  
CIRC\_PROFILE\_NAME: VARCHAR2(25)  
CHARGE\_RENEW: CHAR(1)  
DISCHARGE: CHAR(1)  
CHANGE\_DUE\_DATE: CHAR(1)  
CHANGE\_DISCHARGE\_DATE: CHAR(1)  
ITEM\_ADD\_UPDATE: CHAR(1)

ITEM\_STATUS: CHAR(1)  
ITEM\_DELETE: CHAR(1)  
ADD\_FINES: CHAR(1)  
PAY\_FINES: CHAR(1)  
FORGIVE\_FINES: CHAR(1)  
RECAHOLD\_ADD\_UPDATE: CHAR(1)  
RECAHOLD\_RESEQUENCE: CHAR(1)  
PATRON\_ADD\_UPDATE: CHAR(1)  
PATRON\_DELETE: CHAR(1)  
PATRON\_COUNTERS: CHAR(1)  
PATRON\_PROXY\_MAINTAIN: CHAR(1)  
PATRON\_VIEW\_ONLY: CHAR(1)  
PG\_RESTRICT\_VIEW: CHAR(1)  
PG\_RESTRICT\_MAINT: CHAR(1)  
PG\_RESTRICT\_CIRC: CHAR(1)  
RESERVE\_ADD\_UPDATE: CHAR(1)  
HOLD\_IGNORE\_OWNERSHIP: CHAR(1)  
DISTRIBUTION\_ITEM\_VIEW: CHAR(1)  
DISTRIBUTION\_ITEM\_DISTRIBUTE: CHAR(1)  
DISTRIBUTION\_ITEM\_CREATE: CHAR(1)  
DISTRIBUTION\_ITEM\_UPDATE: CHAR(1)  
DISTRIBUTION\_ITEM\_ORDER: CHAR(1)  
DISTRIBUTION\_ITEM\_RECEIVE: CHAR(1)  
DISTRIBUTION\_ITEM\_DELETE: CHAR(1)  
UPDATE\_PIN: CHAR(1)  
MFHD\_UPDATE: CHAR(1)  
PATRON\_MASK\_SSN: CHAR(1)  
EDIT\_STUB\_PATRON: CHAR(1)  
MANUALLY\_MAP\_PATRON: CHAR(1)  
VIEW\_PATRON\_CIRC\_HISTORY: CHAR(1)

### **CIRC\_SECURITY\_LOCS**

CIRC\_PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER

### **CIRC\_SECURITY\_PG**

CIRC\_PROFILE\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER

### **CIRC\_TRANS\_ARCHIVE**

CIRC\_TRANSACTION\_ID: NUMBER  
ITEM\_ID: NUMBER  
CIRC\_POLICY\_MATRIX\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
CHARGE\_DATE: DATE  
CHARGE\_LOCATION: NUMBER  
CHARGE\_TYPE: CHAR(1)  
CHARGE\_OPER\_ID: VARCHAR2(10)  
DUE\_DATE: DATE  
DISCHARGE\_DATE: DATE  
DISCHARGE\_LOCATION: NUMBER  
DISCHARGE\_TYPE: CHAR(1)

DISCHARGE\_OPER\_ID: VARCHAR2(10)  
RENEWAL\_COUNT: NUMBER  
RECALL\_DATE: DATE  
RECALL\_DUE\_DATE: DATE  
RECALL\_NOTICE\_COUNT: NUMBER  
RECALL\_NOTICE\_DATE: DATE  
OVERDUE\_NOTICE\_COUNT: NUMBER  
OVERDUE\_NOTICE\_DATE: DATE  
OVER\_RECALL\_NOTICE\_COUNT: NUMBER  
OVER\_RECALL\_NOTICE\_DATE: DATE  
PATRON\_ID: NUMBER  
PATRON\_ID\_PROXY: NUMBER  
COURTESY\_NOTICE\_DATE: DATE  
DB\_ID: NUMBER

**CIRC\_TRANS\_EXCEPT\_TYPE**

EXCEPTION\_TYPE: NUMBER  
EXCEPTION\_DESC: VARCHAR2(50)

**CIRC\_TRANS\_EXCEPTION**

CIRC\_TRANS\_EXCEPT\_ID: NUMBER  
ITEM\_ID: NUMBER  
ITEM\_LOCATION: NUMBER  
PATRON\_ID: NUMBER  
TRANS\_EXCEPT\_DATE: DATE  
TRANS\_EXCEPT\_LOCATION: NUMBER  
TRANS\_EXCEPT\_TYPE: NUMBER  
TRANS\_EXCEPT\_OPER\_ID: VARCHAR2(10)

**CIRC\_TRANSACTION\_STATS**

CIRC\_TRANSACTION\_ID: NUMBER  
PATRON\_STAT\_ID: NUMBER

**CIRC\_TRANSACTIONS**

CIRC\_TRANSACTION\_ID: NUMBER  
ITEM\_ID: NUMBER  
CIRC\_POLICY\_MATRIX\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_ID\_PROXY: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
CHARGE\_DATE: DATE  
CHARGE\_LOCATION: NUMBER  
CHARGE\_TYPE: CHAR(1)  
CHARGE\_OPER\_ID: VARCHAR2(10)  
CHARGE\_DUE\_DATE: DATE  
DISCHARGE\_DATE: DATE  
DISCHARGE\_LOCATION: NUMBER  
DISCHARGE\_TYPE: CHAR(1)  
DISCHARGE\_OPER\_ID: VARCHAR2(10)  
RENEWAL\_COUNT: NUMBER  
RECALL\_DATE: DATE  
RECALL\_DUE\_DATE: DATE  
CURRENT\_DUE\_DATE: DATE

RECALL\_NOTICE\_COUNT: NUMBER  
RECALL\_NOTICE\_DATE: DATE  
OVERDUE\_NOTICE\_COUNT: NUMBER  
OVERDUE\_NOTICE\_DATE: DATE  
OVER\_RECALL\_NOTICE\_COUNT: NUMBER  
OVER\_RECALL\_NOTICE\_DATE: DATE  
COURTESY\_NOTICE\_DATE: DATE  
DB\_ID: NUMBER

### **CIRCCHARGES\_VW**

PATRON\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_CODE: VARCHAR2(10)  
PATRON\_GROUP\_NAME: VARCHAR2(25)  
ITEM\_ID: NUMBER  
MFHD\_ID: NUMBER  
BIB\_ID: NUMBER  
PERM\_LOCATION\_CODE: VARCHAR2(10)  
PERM\_LOCATION: VARCHAR2(25)  
GOV\_LOCATION\_CODE: VARCHAR2(10)  
GOV\_LOCATION: VARCHAR2(25)  
PERM\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
PERM\_ITEM\_TYPE: VARCHAR2(25)  
GOV\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
GOV\_ITEM\_TYPE: VARCHAR2(25)  
CHARGE\_DATE\_TIME: DATE  
CHARGE\_DATE\_ONLY: DATE  
CHARGE\_OPER\_ID: VARCHAR2(10)  
CHARGE\_LOCATION: NUMBER  
CHARGE\_LOCATION\_CODE: VARCHAR2(10)  
CHARGE\_LOCATION\_NAME: VARCHAR2(25)  
RENEWAL\_COUNT: NUMBER  
NOTICE\_COUNT: NUMBER

### **CIRCRENEW\_VW**

PATRON\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_CODE: VARCHAR2(10)  
PATRON\_GROUP\_NAME: VARCHAR2(25)  
ITEM\_ID: NUMBER  
MFHD\_ID: NUMBER  
BIB\_ID: NUMBER  
PERM\_LOCATION\_CODE: VARCHAR2(10)  
PERM\_LOCATION: VARCHAR2(25)  
GOV\_LOCATION\_CODE: VARCHAR2(10)  
GOV\_LOCATION: VARCHAR2(25)  
PERM\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
PERM\_ITEM\_TYPE: VARCHAR2(25)  
GOV\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
GOV\_ITEM\_TYPE: VARCHAR2(25)  
CHARGE\_DATE\_TIME: DATE  
CHARGE\_DATE\_ONLY: DATE  
CHARGE\_OPER\_ID: VARCHAR2(10)  
CHARGE\_LOCATION: NUMBER

CHARGE\_LOCATION\_CODE: VARCHAR2(10)  
CHARGE\_LOCATION\_NAME: VARCHAR2(25)

RENEWAL\_COUNT: NUMBER

RENEW\_DATE\_TIME: DATE

RENEW\_DATE\_ONLY: DATE

RENEW\_OPER\_ID: VARCHAR2(10)

RENEW\_LOCATION\_CODE: VARCHAR2(10)  
LOCATION\_NAME: VARCHAR2(25)

### **CLAIM\_TYPES**

CLAIM\_TYPE: NUMBER

EDI\_CODE: VARCHAR2(11)

CLAIM\_TYPE\_DESC: VARCHAR2(70)

### **CLASS\_SECTION**

SECTION\_ID: NUMBER

SECTION\_NUMBER: VARCHAR2(10)

NORMAL\_SECTION\_NUMBER: VARCHAR2(10)

NUMBER\_OF\_STUDENTS: NUMBER

CIRC\_CLUSTER\_ID: NUMBER

### **COMPLEX\_COMP\_PATTERN**

CCP\_ID: NUMBER

CP\_ID: NUMBER

COMPONENT\_ID: NUMBER

START\_ISSUE\_ID: NUMBER

END\_ISSUE\_ID: NUMBER

END\_DATE: DATE

END\_CP\_ISSUE: NUMBER

### **COMPLEX\_PATTERN**

CP\_ID: NUMBER

PATTERN\_NAME: VARCHAR2(40)

PATTERN\_NAME\_NORM: VARCHAR2(40)

CREATE\_DATE: DATE

CREATE\_OPID: VARCHAR2(10)

CREATE\_LOCATION\_ID: NUMBER

UPDATE\_DATE: DATE

UPDATE\_OPID: VARCHAR2(10)

UPDATE\_LOCATION\_ID: NUMBER

### **COMPONENT**

COMPONENT\_ID: NUMBER

SUBSCRIPTION\_ID: NUMBER

COMPONENT\_NAME: VARCHAR2(100)

COMPONENT\_NAME\_NORM: VARCHAR2(100)

UNIT\_TITLE: NUMBER

CATEGORY: NUMBER

PREDICT: CHAR(1)

NEXT\_ISSUE\_ID: NUMBER

NOTE: VARCHAR2(256)

ITEM\_TYPE\_ID: NUMBER

CREATE\_ITEMS: CHAR(1)  
CLAIM\_INTERVAL: NUMBER

**COMPONENT\_ALTCRONDAY**

COMPONENT\_ID: NUMBER  
CHRON\_DAY: NUMBER  
TYPE\_OF\_DAY: CHAR(3)

**COMPONENT\_CHRONDAY**

COMPONENT\_ID: NUMBER  
CHRON\_DAY: NUMBER  
TYPE\_OF\_DAY: CHAR(3)

**COMPONENT\_ISSUE\_DAY**

COMPONENT\_ID: NUMBER  
EXPECTED\_DAY: NUMBER  
TYPE\_OF\_DAY: CHAR(3)

**COMPONENT\_ISSUES\_ROUTED**

ROUTING\_LIST\_ID: NUMBER  
ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER

**COMPONENT\_PATTERN**

COMPONENT\_ID: NUMBER  
PATTERN\_ID: NUMBER  
FREQUENCY\_CODE: CHAR(1)  
END\_DATE: DATE  
END\_ISSUE\_ID: NUMBER  
START\_ISSUE\_ID: NUMBER  
REGULARITY: CHAR(12)  
REGULARITY\_MARC: VARCHAR2(50)  
LVL1\_INC\_AT: NUMBER  
LVL2\_INC\_AT: NUMBER  
LVL3\_INC\_AT: NUMBER  
LVL4\_INC\_AT: NUMBER  
LVL5\_INC\_AT: NUMBER  
LVL6\_INC\_AT: NUMBER  
ALT\_LVL1\_INC\_AT: NUMBER  
ALT\_LVL2\_INC\_AT: NUMBER

**COMPONENT\_ROUTING**

ROUTING\_LIST\_ID: NUMBER  
COMPONENT\_ID: NUMBER

**CONVERSION\_RATE\_AUDIT**

CURRENCY\_ID: NUMBER  
AUDIT\_ID: NUMBER  
CONVERSION\_RATE: NUMBER  
RATE\_CREATE\_DATE\_TIME: DATE  
RATE\_CREATE\_OPERATOR\_ID: VARCHAR2(10)

**COURSE**

COURSE\_ID: NUMBER

COURSE\_NAME: VARCHAR2(40)  
NORMAL.Course\_NAME: VARCHAR2(40)  
COURSE\_NUMBER: VARCHAR2(10)  
NORMAL.Course\_NUMBER: VARCHAR2(10)  
BEGIN\_DATE: DATE  
END\_DATE: DATE  
CIRC\_CLUSTER\_ID: NUMBER

**CP\_CELL**

CP\_LEVEL\_ID: NUMBER  
CP\_ISSUE\_ID: NUMBER  
LEVEL\_INCREMENT: VARCHAR2(80)

**CP\_DOMAIN\_TYPE**

CP\_DOMAIN\_TYPE\_ID: NUMBER  
DOMAIN: CHAR(1)  
ENUM\_CHRON\_TYPE\_ID: NUMBER

**CP\_ISSUE**

CP\_ISSUE\_ID: NUMBER  
CP\_ID: NUMBER  
CP\_ISSUE: NUMBER  
EXPECTED\_DATE\_INC: NUMBER  
TIME\_UNIT\_CODE: CHAR(1)

**CP\_LEVEL**

CP\_LEVEL\_ID: NUMBER  
CP\_ID: NUMBER  
CP\_DOMAIN\_TYPE\_ID: NUMBER  
CP\_LEVEL: NUMBER  
PRINT\_ORDER: NUMBER  
REG\_OR\_ALT: CHAR(1)  
CAPTION: VARCHAR2(50)  
IS\_CONSTANT: CHAR(1)

**CURRENCY\_CONVERSION**

CURRENCY\_ID: NUMBER  
COUNTRY\_NAME: VARCHAR2(45)  
NORMAL\_COUNTRY\_NAME: VARCHAR2(45)  
CURRENCY\_NAME: VARCHAR2(35)  
NORMAL\_CURRENCY\_NAME: VARCHAR2(35)  
CURRENCY\_CODE: VARCHAR2(3)  
NORMAL\_CURRENCY\_CODE: VARCHAR2(3)  
CREATE\_DATE: DATE  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
CONVERSION\_RATE: NUMBER  
RATE\_CREATE\_DATE\_TIME: DATE  
RATE\_CREATE\_OPERATOR\_ID: VARCHAR2(10)  
DECIMALS: NUMBER  
DECIMAL\_DELIMITER: CHAR(1)

**DATABASE\_ADDRESS**

DB\_ID: NUMBER  
DB\_ADDR: VARCHAR2(100)  
DB\_PORT: NUMBER  
APPLICATION\_TYPE: VARCHAR2(20)

**DATABASE\_LICENSE**

LICENSE\_ID: NUMBER  
DB\_CODE: VARCHAR2(8)  
SESSION\_ID: NUMBER  
INIT\_DATE: DATE  
MODULE: VARCHAR2(20)

**DEPARTMENT**

DEPARTMENT\_ID: NUMBER  
DEPARTMENT\_NAME: VARCHAR2(40)  
NORMAL\_DEPT\_NAME: VARCHAR2(40)  
DEPARTMENT\_CODE: VARCHAR2(10)  
NORMAL\_DEPT\_CODE: VARCHAR2(10)  
CIRC\_CLUSTER\_ID: NUMBER

**DEWEYCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(3)  
LONGCLASS: VARCHAR2(300)

**DISTRIBUTION\_ITEM**

ITEM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ACTIVE: CHAR(1)  
ON\_HAND\_QUANTITY: NUMBER  
ORDER\_QUANTITY: NUMBER  
REORDER\_POINT: NUMBER  
REORDER\_AUTOMATIC: CHAR(1)  
HISTORICAL\_DISTRIBUTIONS: NUMBER  
CREATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
MODIFY\_DATE: DATE  
MODIFY\_LOCATION\_ID: NUMBER  
MODIFY\_OPID: VARCHAR2(10)

**DISTRIBUTION\_ORDER**

DISTRIBUTION\_ORDER\_ID: NUMBER  
ITEM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ORDER\_QUANTITY: NUMBER  
EXPECTED\_DATE: DATE  
NOT\_YET\_RECEIVED: NUMBER  
ORDER\_DATE: DATE  
ORDER\_LOCATION\_ID: NUMBER  
ORDER\_OPID: VARCHAR2(10)  
ORDER\_COMPLETE: CHAR(1)

**DISTRIBUTION\_RECEIPT**

DISTRIBUTION\_RECEIPT\_ID: NUMBER  
DISTRIBUTION\_ORDER\_ID: NUMBER  
RECEIPT\_QUANTITY: NUMBER  
RECEIPT\_DATE: DATE  
RECEIPT\_LOCATION\_ID: NUMBER  
RECEIPT\_OPID: VARCHAR2(10)

**DISTRIBUTION\_TRANSACTION**

DISTRIBUTION\_TRANSACTION\_ID: NUMBER  
ITEM\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
DISTRIBUTION\_DATE: DATE  
DISTRIBUTION\_LOCATION\_ID: NUMBER  
DISTRIBUTION\_OPID: VARCHAR2(10)

**DR\$HEADING\_CONTEXT\_IDX\$!**

TOKEN\_TEXT: VARCHAR2(64)  
TOKEN\_TYPE: NUMBER  
TOKEN\_FIRST: NUMBER  
TOKEN\_LAST: NUMBER  
TOKEN\_COUNT: NUMBER  
TOKEN\_INFO: BLOB

**DR\$HEADING\_CONTEXT\_IDX\$K**

DOCID: NUMBER  
TEXTKEY: ROWID

**DR\$HEADING\_CONTEXT\_IDX\$N**

NLT\_DOCID: NUMBER  
NLT\_MARK: CHAR(1)

**DR\$HEADING\_CONTEXT\_IDX\$R**

ROW\_NO: NUMBER  
DATA: BLOB

**DUP\_DETECTION\_PROFILE**

RECORD\_TYPE: CHAR(1)  
DUP\_PROFILE\_ID: NUMBER  
DUP\_PROFILE\_NAME: VARCHAR2(25)  
DUP\_PROFILE\_CODE: VARCHAR2(8)  
DUP\_HANDLING: CHAR(1)  
CANCELLATION: CHAR(1)  
DUP\_REPLACE: NUMBER  
DUP\_WARN: NUMBER  
DISPLAYFIELD1: VARCHAR2(30)  
DISPLAYFIELD2: VARCHAR2(30)  
DISPLAYFIELD3: VARCHAR2(30)  
SORTFIELD1: VARCHAR2(30)  
SORTFIELD2: VARCHAR2(30)  
SORTFIELD3: VARCHAR2(30)  
DISCARD\_UNMATCHED: CHAR(1)

### **DUP\_PROFILE\_FIELDS**

DUP\_PROFILE\_ID: NUMBER  
SEARCHCODE: CHAR(4)  
SEQNUM: NUMBER  
FIELD OVERRIDE: CHAR(3)  
SUBFIELD OVERRIDE: CHAR(10)  
WEIGHT: NUMBER  
INDICATOR\_1: VARCHAR2(1)  
INDICATOR\_2: VARCHAR2(1)

### **DUP\_PROFILE\_QUALITY**

DUP\_PROFILE\_ID: NUMBER  
SEQNUM: NUMBER  
NUC\_CODE: VARCHAR2(15)  
ENCODING\_LEVEL: CHAR(1)  
MODIFYING\_AGENCY: VARCHAR2(15)  
RECORD\_TYPE: VARCHAR2(2)

### **DUPE\_PROFILE\_MERGE**

DUP\_PROFILE\_ID: NUMBER  
MARC\_FIELD: CHAR(3)  
MARC\_IND1: CHAR(1)  
MARC\_IND2: CHAR(1)  
NUC5: VARCHAR2(15)

### **EDI\_CODE\_REF**

USAGE: NUMBER  
CODE: VARCHAR2(3)  
DESCR: VARCHAR2(70)

### **EDI\_CODE\_USAGES**

USAGE: NUMBER  
DATA\_ELEMENT: VARCHAR2(4)  
DESCR: VARCHAR2(70)

### **EDI\_CONNECTION\_PROFILE**

PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER  
VENDOR\_ID: NUMBER  
LIBRARY\_ENVELOPE\_ADDRESS: VARCHAR2(55)  
LIBRARY\_INSIDE\_ADDRESS: VARCHAR2(25)  
VENDOR\_ENVELOPE\_ADDRESS: VARCHAR2(55)  
VENDOR\_INSIDE\_ADDRESS: VARCHAR2(25)  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
USE\_VENDOR\_ACCOUNT: CHAR(1)  
USE\_PO: CHAR(1)  
USE\_SC: CHAR(1)  
USE\_MC: CHAR(1)  
USE\_IV: CHAR(1)  
USE\_SR: CHAR(1)

USE\_MR: CHAR(1)  
USE\_XM: CHAR(1)

**EDI\_CURSOR**

CURSOR\_ID: NUMBER  
FILE\_ID: NUMBER  
MSG\_ID: NUMBER  
FILE\_NAME: VARCHAR2(30)  
FILE\_POSITION: NUMBER  
MSG\_DELIMITERS: VARCHAR2(6)

**EDI\_EVENT\_TYPES**

EVENT\_TYPE: NUMBER  
EVENT\_DESC: VARCHAR2(25)

**EDI\_FILE**

FILE\_ID: NUMBER  
FILE\_NAME: VARCHAR2(30)  
FILE\_TYPE: CHAR(1)  
FILE\_STATUS: NUMBER  
FILE\_SIZE: NUMBER  
FILE\_UPDATE\_DATE: DATE  
CREATE\_DATE: DATE  
CREATE\_OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OP\_ID: VARCHAR2(10)  
TRANS\_COUNT: NUMBER  
GROUP\_COUNT: NUMBER  
MESSAGE\_COUNT: NUMBER

**EDI\_HISTORY**

EVENT\_ID: NUMBER  
FILE\_ID: NUMBER  
MSG\_ID: NUMBER  
EVENT\_TYPE: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OP\_ID: VARCHAR2(10)

**EDI\_MESSAGE**

MSG\_ID: NUMBER  
MSG\_DIRECTION: CHAR(1)  
MSG\_TYPE: VARCHAR2(6)  
MSG\_TYPE\_CODE: NUMBER  
MSG\_VERSION\_CODE: NUMBER  
MSG\_STATUS: NUMBER  
MSG\_NUMBER: VARCHAR2(35)  
MSG\_DATE: DATE  
LOAD\_OR\_APPEND\_DATE: DATE  
LINE\_ITEM\_COUNT: NUMBER  
TOTAL\_AMOUNT: NUMBER  
LOCATION\_ID: NUMBER

VENDOR\_ID: NUMBER  
SENDER\_CODE: VARCHAR2(55)  
RECEIVER\_CODE: VARCHAR2(55)  
DB\_REF\_ID: NUMBER  
FILE\_ID: NUMBER  
FILE\_START\_POS: NUMBER  
FILE\_END\_POS: NUMBER  
MATCHING\_PROFILE: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OP\_ID: VARCHAR2(10)  
CREATE\_LOC: NUMBER  
UPDATE\_DATE: DATE  
UPDATE\_OP\_ID: VARCHAR2(10)  
UPDATE\_LOC: NUMBER  
TRANS\_INDEX: NUMBER  
GROUP\_INDEX: NUMBER  
DOC\_MSG\_CODE: VARCHAR2(3)  
MSG\_DELIMITERS: VARCHAR2(6)  
DATA\_PRESENT: CHAR(1)

#### **EDI\_MISSING\_LINE\_ITEM**

EXCEPTION\_ID: NUMBER  
LINE\_ID: NUMBER  
PROBLEM\_CODE: NUMBER  
TITLE: VARCHAR2(100)  
STD\_NUMBER: VARCHAR2(40)  
PRINT\_STD\_NUM: CHAR(2)  
VENDOR\_TITLE\_NUM: VARCHAR2(40)  
VENDOR\_REF\_QUAL: VARCHAR2(3)  
VENDOR\_REF\_NUM: VARCHAR2(35)

#### **EDI\_NOTE**

EVENT\_ID: NUMBER  
NOTE\_CODE: NUMBER  
POSITION: NUMBER

#### **EDI\_SECTION**

SECTION\_ID: NUMBER  
MSG\_ID: NUMBER  
SECTION\_TYPE: VARCHAR2(3)  
SECTION\_ORDINAL: NUMBER

SEG\_COUNT: NUMBER  
SEGMENTS: LONG RAW

#### **EITEM**

EITEM\_ID: NUMBER  
MFHD\_ID: NUMBER  
ENUMERATION: VARCHAR2(80)  
CHRONOLOGY: VARCHAR2(80)  
YEAR: VARCHAR2(20)  
CAPTION: VARCHAR2(255)  
SEQUENCE: NUMBER

LINK: VARCHAR2(255)  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_LOCATION\_ID: NUMBER

**EITEM\_NOTE\_TYPE**

NOTE\_TYPE: NUMBER  
NOTE\_DESC: VARCHAR2(25)

**EITEM\_NOTES**

EITEM\_ID: NUMBER  
EITEM\_NOTE\_TYPE\_ID: NUMBER  
NOTE: VARCHAR2(2000)

**ELINK\_INDEX**

ELINK\_ID: NUMBER  
RECORD\_TYPE: CHAR(1)  
RECORD\_ID: NUMBER  
SEQNUM: NUMBER  
LINK\_TYPE: CHAR(3)  
LINK: VARCHAR2(1024)  
LINK\_TEXT: VARCHAR2(1024)  
LINK\_TEXT\_NORMAL: VARCHAR2(1024)  
LINK\_SUBTYPE: VARCHAR2(10)  
URL\_HOST: VARCHAR2(255)  
URL\_PORT: NUMBER  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
CHECK\_DATE: DATE  
CHECK\_STATUS: CHAR(1)  
PARSE\_STATUS: CHAR(1)

**ELINK\_RECORD\_TYPE**

RECORD\_TYPE\_ID: VARCHAR2(10)  
RECORD\_TYPE: VARCHAR2(25)

**ENUM\_CHRON\_TYPES\_VW**

CP\_DOMAIN\_TYPE\_ID: NUMBER  
DOMAIN: CHAR(1)  
DOMAIN\_DESC: VARCHAR2(13)  
CODE: VARCHAR2(2)  
NAME: VARCHAR2(40)  
ENUMERATION\_TYPE\_ID: NUMBER

**ENUM\_CHRON\_TYPES\_VW**

CHRON\_TYPE\_ID: NUMBER

**ENUMERATION\_TYPE**

ENUMERATION\_TYPE\_ID: NUMBER  
CODE: CHAR(2)

NAME: VARCHAR2(40)

### **EQUIPMENT**

EQUIP\_ID: NUMBER  
EQUIP\_NO: VARCHAR2(15)  
EQUIP\_NO\_NORMALIZED: VARCHAR2(15)  
GROUP\_EQUIP\_ID: NUMBER  
IS\_GROUP: CHAR(1)  
DATE\_PURCHASED: DATE  
VALUE\_PURCHASE: NUMBER  
VALUE\_REPLACEMENT: NUMBER  
PART\_SUPPLIER: VARCHAR2(100)  
PART\_SUPPLIER\_NORMALIZED: VARCHAR2(100)  
DEALER: VARCHAR2(100)  
DEALER\_NORMALIZED: VARCHAR2(100)  
MANUFACTURER: VARCHAR2(100)  
MANUFACTURER\_NORMALIZED: VARCHAR2(100)  
MODEL: VARCHAR2(100)  
MODEL\_NORMALIZED: VARCHAR2(100)  
SERIAL\_NO: VARCHAR2(100)  
SERIAL\_NO\_NORMALIZED: VARCHAR2(100)  
PART\_NO: VARCHAR2(100)  
PART\_NO\_NORMALIZED: VARCHAR2(100)  
EQUIP\_FORMAT: VARCHAR2(25)  
EQUIP\_FORMAT\_NORMALIZED: VARCHAR2(25)  
EQUIP\_TYPE\_ID: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
TEMP\_ROOM\_ID: NUMBER  
HISTORICAL\_BOOKINGS: NUMBER  
HISTORICAL\_MAINTENANCE: NUMBER  
LAST\_INVENTORIED: DATE  
NEXT\_MAINTENANCE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_OPID: VARCHAR2(10)  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_LOCATION\_ID: NUMBER

### **EQUIPMENT\_BARCODE**

EQUIP\_ID: NUMBER  
BARCODE\_NO: VARCHAR2(25)  
BARCODE\_NO\_NORMALIZED: VARCHAR2(25)  
EQUIP\_BARCODE\_STS\_ID: NUMBER  
STATUS\_DATE: DATE

### **EQUIPMENT\_BARCODE\_STATUS**

EQUIP\_BARCODE\_STS\_ID: NUMBER  
BARCODE\_STS: VARCHAR2(25)

### **EQUIPMENT\_MEDIA\_TYPE**

EQUIP\_TYPE\_ID: NUMBER  
MEDIA\_TYPE\_ID: NUMBER

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
PRIORITY: NUMBER

**EQUIPMENT\_NOTE\_TYPE**

EQUIP\_NOTE\_TYPE\_ID: NUMBER  
TYPE: VARCHAR2(15)

**EQUIPMENT\_NOTES**

EQUIP\_ID: NUMBER  
EQUIP\_NOTE\_TYPE\_ID: NUMBER  
NOTE: VARCHAR2(2000)  
OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE

**EQUIPMENT\_STATUS**

EQUIP\_ID: NUMBER  
EQUIP\_STS\_TYPE\_ID: NUMBER  
NOTE: VARCHAR2(100)  
UPDATE\_DATE: DATE  
OP\_ID: VARCHAR2(10)

**EQUIPMENT\_STATUS\_TYPE**

EQUIP\_STS\_TYPE\_ID: NUMBER  
STS\_TYPE: VARCHAR2(40)  
DISPLAY\_PRIORITY: NUMBER  
WARN\_ON\_BOOKING: CHAR(1)  
BLOCK\_BOOKING: CHAR(1)  
WARN\_ON\_CHARGE: CHAR(1)  
BLOCK\_CHARGE: CHAR(1)  
MESSAGE: VARCHAR2(50)  
DISCHARGE\_MESSAGE\_SHOW: CHAR(1)  
DISCHARGE\_MESSAGE: VARCHAR2(50)

**EQUIPMENT\_TYPE**

EQUIP\_TYPE\_ID: NUMBER  
TYPE\_CODE: VARCHAR2(10)  
TYPE: VARCHAR2(50)  
IS\_GROUP: CHAR(1)  
REPLACEMENT\_DEFAULT: NUMBER  
SETUP\_TIME: NUMBER  
CLEANUP\_TIME: NUMBER

**EXCEPTION\_CALENDAR**

CALENDAR\_ID: NUMBER  
EXCEPTION\_DATE: DATE  
EXCEPTION\_OPEN: CHAR(1)  
EXCEPTION\_OPENHOUR: NUMBER  
EXCEPTION\_CLOSEHOUR: NUMBER  
EXCEPTION\_HOURLY\_EFFECT: NUMBER  
EXCEPTION\_LOAN\_DUE: NUMBER

**EXCEPTION\_TYPES**

EXCEPTION\_TYPE: NUMBER

EXCEPTION\_TYPE\_DESC: VARCHAR2(20)

### **FIELDWEIGHTS**

FIELDCODE: CHAR(4)  
FIELDWEIGHT: NUMBER

### **FINE\_FEE**

FINE\_FEE\_ID: NUMBER  
PATRON\_ID: NUMBER  
ITEM\_ID: NUMBER  
CREATE\_DATE: DATE  
OPERATOR\_ID: VARCHAR2(10)  
FINE\_FEE\_TYPE: NUMBER  
FINE\_FEE\_LOCATION: NUMBER  
FINE\_FEE\_AMOUNT: NUMBER  
FINE\_FEE\_BALANCE: NUMBER  
FINE\_FEE\_NOTE: VARCHAR2(1000)  
ORIG\_CHARGE\_DATE: DATE  
DUE\_DATE: DATE  
FINE\_FEE\_NOTICE\_DATE: DATE  
DB\_ID: NUMBER

### **FINE\_FEE\_TRANS\_METHOD**

METHOD\_TYPE: NUMBER  
METHOD\_DESC: VARCHAR2(25)

### **FINE\_FEE\_TRANS\_TYPE**

TRANSACTION\_TYPE: NUMBER  
TRANSACTION\_DESC: VARCHAR2(25)

### **FINE\_FEE\_TRANSACTIONS**

FINE\_FEE\_TRANS\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
TRANS\_DATE: DATE  
TRANS\_AMOUNT: NUMBER  
TRANS\_TYPE: NUMBER  
TRANS\_METHOD: NUMBER  
TRANS\_LOCATION: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
TRANS\_NOTE: VARCHAR2(1000)

### **FINE\_FEE\_TYPE**

FINE\_FEE\_TYPE: NUMBER  
FINE\_FEE\_DESC: VARCHAR2(25)  
FINE\_FEE\_CODE: VARCHAR2(10)

### **FISCAL\_PERIOD**

FISCAL\_PERIOD\_ID: NUMBER  
FISCAL\_PERIOD\_NAME: VARCHAR2(25)  
START\_DATE: DATE  
END\_DATE: DATE

### **FREQUENCY**

FREQUENCY\_CODE: CHAR(1)  
FREQUENCY\_DESC: VARCHAR2(25)  
FREQ\_INCREMENT: NUMBER  
FREQ\_CALC\_TYPE: CHAR(1)

**FUND**

FUND\_ID: NUMBER  
LEDGER\_ID: NUMBER  
PARENT\_FUND: NUMBER  
FUND\_NAME: VARCHAR2(25)  
NORMAL\_FUND\_NAME: VARCHAR2(25)  
FUND\_CODE: VARCHAR2(10)  
NORMAL\_FUND\_CODE: VARCHAR2(10)  
CATEGORY: NUMBER  
FUND\_TYPE: NUMBER  
BEGIN\_DATE: DATE  
END\_DATE: DATE  
INSTITUTION\_FUND\_ID: VARCHAR2(50)  
EXPEND\_ONLY: CHAR(1)  
ORIGINAL\_ALLOCATION: NUMBER  
ALLOCATION\_INCREASE: NUMBER  
ALLOCATION\_DECREASE: NUMBER  
COMMIT\_PENDING: NUMBER  
COMMITMENTS: NUMBER  
EXPEND\_PENDING: NUMBER  
EXPENDITURES: NUMBER  
OVERCOMMIT: CHAR(1)  
OVERCOMMIT\_WARN: NUMBER  
OVERCOMMIT\_PERCENT: NUMBER  
COMMIT\_FREEZE: DATE  
UNDERCOMMIT\_PERCENT: NUMBER  
OVEREXPEND: CHAR(1)  
OVEREXPEND\_WARN: NUMBER  
OVEREXPEND\_PERCENT: NUMBER  
EXPEND\_FREEZE: DATE  
UNDEREXPEND\_PERCENT: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)

**FUND\_NOTE**

FUND\_ID: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_NOTE: VARCHAR2(1900)

**FUND\_PAYMENT**

PAYMENT\_ID: NUMBER  
SPLIT\_FUND\_SEQ: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_ID: NUMBER  
PERCENTAGE: NUMBER

AMOUNT: NUMBER

**FUND\_TRANSACTION**

FUND\_ID: NUMBER  
AUDIT\_ID: NUMBER  
LEDGER\_ID: NUMBER  
TRANS\_TYPE: NUMBER  
AMOUNT: NUMBER  
TRANS\_DATE: DATE  
OPERATOR\_ID: VARCHAR2(10)  
REFERENCE\_NO: VARCHAR2(25)  
STATISTICAL\_FUND: NUMBER  
NOTE: VARCHAR2(1900)

**FUND\_TYPE**

FUND\_TYPE\_ID: NUMBER  
FUND\_TYPE\_NAME: VARCHAR2(25)  
COMMIT\_WARNING: NUMBER  
EXPEND\_WARNING: NUMBER  
OVERCOMMIT\_LIMIT: NUMBER  
OVEREXPEND\_LIMIT: NUMBER  
UNDERCOMMIT: NUMBER  
UNDEREXPEND: NUMBER

**FUNDLEDGER\_VW**

FUNDLINE: VARCHAR2(255)  
FISCAL\_PERIOD\_ID: NUMBER  
FISCAL\_PERIOD\_NAME: VARCHAR2(25)  
FISCAL\_PERIOD\_START: DATE  
FISCAL\_PERIOD\_END: DATE  
LEDGER\_ID: NUMBER  
LEDGER\_NAME: VARCHAR2(40)  
NORMAL\_LEDGER\_NAME: VARCHAR2(40)  
POLICY\_NAME: VARCHAR2(40)  
FUND\_TYPE: VARCHAR2(25)  
FUND\_CATEGORY: VARCHAR2(9)  
FUND\_ID: NUMBER  
FUND\_NAME: VARCHAR2(25)  
NORMAL\_FUND\_NAME: VARCHAR2(25)  
PARENT\_FUND\_ID: NUMBER  
PARENT\_FUND: VARCHAR2(25)  
INSTITUTION\_FUND\_ID: VARCHAR2(50)  
BEGIN\_DATE: DATE  
END\_DATE: DATE  
ORIGINAL\_ALLOCATION: NUMBER  
CURRENT\_ALLOCATION: NUMBER  
CASH\_BALANCE: NUMBER  
FREE\_BALANCE: NUMBER  
EXPENDITURES: NUMBER  
COMMITMENTS: NUMBER  
COMMIT\_PENDING: NUMBER  
EXPEND\_PENDING: NUMBER

**GEO\_COORD\_TYPE**

COORD\_TYPE: NUMBER  
COORD\_NAME: VARCHAR2(25)

**GEO\_FORMAT\_TYPE**

FORMAT\_TYPE: NUMBER  
FORMAT\_NAME: VARCHAR2(30)  
COORD\_TYPE: NUMBER

**GEO\_SEARCH**

SEARCH\_TYPE: NUMBER  
SEARCH\_NAME: VARCHAR2(25)

**GEO\_UNITS**

UNIT\_TYPE: NUMBER  
UNIT\_NAME: VARCHAR2(25)

**GLOBAL\_PARM**

PARM: VARCHAR2(25)  
VALUE: VARCHAR2(50)

**HEADING**

HEADING\_ID: NUMBER  
NORMAL\_HEADING: VARCHAR2(300)  
DISPLAY\_HEADING: VARCHAR2(300)  
INDEX\_TYPE: CHAR(1)  
HEADING\_TYPE: CHAR(1)  
STAFFBIBS: NUMBER  
OPACBIBS: NUMBER  
STAFFREFS: VARCHAR2(5)  
OPACREFS: VARCHAR2(5)  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE

**HEADING\_CHANGE**

HEADING\_CHANGE\_ID: NUMBER  
HEADING\_QUEUE\_ID: NUMBER  
NEW\_HEADING: VARCHAR2(330)  
INDEX\_TYPE: CHAR(1)  
HEADING\_ID\_OLD: NUMBER  
HEADING\_ID\_NEW: NUMBER  
PROCESS\_FLAG: CHAR(1)  
CHANGE\_DATE: DATE

**HEADING\_CHANGE\_FIELDS**

HEADING\_CHANGE\_ID: NUMBER  
REC\_ID: NUMBER  
REC\_TYPE: CHAR(1)  
MARC\_TAG: CHAR(3)  
MARC\_IND1: CHAR(1)  
MARC\_IND2: CHAR(1)  
OLD\_FIELD: VARCHAR2(330)  
NEW\_FIELD: VARCHAR2(330)

CHANGE\_DATE: DATE

**HEADING\_CHANGE\_QUEUE**

HEADING\_QUEUE\_ID: NUMBER  
REC\_ID: NUMBER  
INDEX\_TYPE: CHAR(1)  
HEADING\_ID\_OLD: NUMBER  
HEADING\_ID\_NEW: NUMBER  
CHANGE\_DATE: DATE  
PROCESS\_FLAG: CHAR(1)  
REC\_TYPE: CHAR(1)

**HEADING\_SUBDIVISION**

HEADING\_ID: NUMBER  
SUBDIV\_ID: NUMBER

**HEADING\_TYPE**

INDEX\_TYPE: CHAR(1)  
HEADING\_TYPE: CHAR(1)  
HEADING\_CODE: VARCHAR2(20)  
HEADING\_TYPE\_DESC: VARCHAR2(50)  
STAFFSUPPRESS: CHAR(1)

**HEADING\_VW**

AUTH\_ID: NUMBER  
HEADING\_ID: NUMBER  
REFERENCE\_TYPE: VARCHAR2(20)  
NORMAL\_HEADING: VARCHAR2(300)  
DISPLAY\_HEADING: VARCHAR2(300)  
OPACBIBS: NUMBER  
CREATE\_DATE: DATE  
INDEX\_NAME: VARCHAR2(30)  
HEADING\_TYPE: VARCHAR2(50)

**HOLD\_RECALL**

HOLD\_RECALL\_ID: NUMBER  
PATRON\_ID: NUMBER  
HOLD\_RECALL\_TYPE: CHAR(1)  
PICKUP\_LOCATION: NUMBER  
EXPIRE\_DATE: DATE  
AVAILABLE\_NOTICE\_COUNT: NUMBER  
AVAILABLE\_NOTICE\_DATE: DATE  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
BIB\_ID: NUMBER  
REQUEST\_LEVEL: CHAR(1)  
REQUEST\_ITEM\_COUNT: NUMBER  
REQUEST\_GROUP\_ID: NUMBER  
PATRON\_COMMENT: VARCHAR2(100)  
PATRON\_GROUP\_ID: NUMBER  
CALL\_SLIP\_ID: NUMBER  
HOLDING\_DB\_ID: NUMBER

**HOLD\_RECALL\_ARCHIVE**

HOLD\_RECALL\_ID: NUMBER  
HOLD\_RECALL\_TYPE: CHAR(1)  
PICKUP\_LOCATION: NUMBER  
EXPIRE\_DATE: DATE  
AVAILABLE\_NOTICE\_COUNT: NUMBER  
AVAILABLE\_NOTICE\_DATE: DATE  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
BIB\_ID: NUMBER  
REQUEST\_LEVEL: CHAR(1)  
REQUEST\_ITEM\_COUNT: NUMBER  
REQUEST\_GROUP\_ID: NUMBER  
PATRON\_COMMENT: VARCHAR2(100)  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
CALL\_SLIP\_ID: NUMBER  
HOLDING\_DB\_ID: NUMBER

**HOLD\_RECALL\_ITEM\_ARCHIVE**

HOLD\_RECALL\_ID: NUMBER  
ITEM\_ID: NUMBER  
HOLD\_RECALL\_TYPE: CHAR(1)  
HOLD\_RECALL\_STATUS: NUMBER  
HOLD\_RECALL\_STATUS\_DATE: DATE

**HOLD\_RECALL\_ITEMS**

HOLD\_RECALL\_ID: NUMBER  
ITEM\_ID: NUMBER  
QUEUE\_POSITION: NUMBER  
HOLD\_RECALL\_TYPE: CHAR(1)  
HOLD\_RECALL\_STATUS: NUMBER  
HOLD\_RECALL\_STATUS\_DATE: DATE

**HOLD\_RECALL\_STATS**

HOLD\_RECALL\_ID: NUMBER  
PATRON\_STAT\_ID: NUMBER

**HOLD\_RECALL\_STATUS**

HR\_STATUS\_TYPE: NUMBER  
HR\_STATUS\_DESC: VARCHAR2(25)

**IMPORT\_RULE**

IMPORT\_RULE\_ID: NUMBER  
CODE: VARCHAR2(8)  
NAME: VARCHAR2(25)  
BIB\_DUP\_PROFILE\_ID: NUMBER  
AUTH\_DUP\_PROFILE\_ID: NUMBER  
CREATE\_MFHDS\_ITEMS: CHAR(1)

LIBRARY\_ID: NUMBER  
CAT REVIEW: CHAR(1)  
LOC\_FIELD: VARCHAR2(3)  
LOC\_SUBFIELD: CHAR(1)  
LOC\_IND1: CHAR(1)  
LOC\_IND2: CHAR(1)  
SUPPRESS\_IN\_OPAC: CHAR(1)  
IMPORT\_RULE\_PO\_ID: NUMBER  
ORDER\_CREATE: CHAR(1)  
CREATE\_MFHDS\_ONLY: CHAR(1)  
BIB\_TO\_MFHD: CHAR(1)  
BIB\_DUP\_EXIST: CHAR(1)  
CHAR\_SET\_ID: NUMBER  
IGNORE\_OPAC\_SUPPRESS: CHAR(1)

#### **IMPORT\_RULE\_BIBTOMFHD**

IMPORT\_RULE\_ID: NUMBER  
MFHD\_FIELD: VARCHAR2(3)

#### **IMPORT\_RULE\_PO**

IMPORT\_RULE\_PO\_ID: NUMBER  
LOCATION\_ID\_ORDER: NUMBER  
CURRENCY\_CODE: CHAR(3)  
VENDOR\_ID: NUMBER  
ACCOUNT\_ID: NUMBER  
ORDER\_TYPE: NUMBER  
FUND\_CODE: VARCHAR2(10)  
COPY\_FIELD: CHAR(3)  
COPY\_SUBFIELD: CHAR(1)  
COPY\_IND1: CHAR(1)  
COPY\_IND2: CHAR(1)  
COPY\_DEFAULT: NUMBER  
PRICE\_FIELD: CHAR(3)  
PRICE\_SUBFIELD: CHAR(1)  
PRICE\_IND1: CHAR(1)  
PRICE\_IND2: CHAR(1)  
PRICE\_DEFAULT: NUMBER  
FUND\_FIELD: CHAR(3)  
FUND\_SUBFIELD: CHAR(1)  
FUND\_IND1: CHAR(1)  
FUND\_IND2: CHAR(1)  
NOTES\_FIELD: CHAR(3)  
NOTES\_SUBFIELD: CHAR(1)  
NOTES\_IND1: CHAR(1)  
NOTES\_IND2: CHAR(1)  
INSTRUCTION\_FIELD: CHAR(3)  
INSTRUCTION\_SUBFIELD: CHAR(1)  
INSTRUCTION\_IND1: CHAR(1)  
INSTRUCTION\_IND2: CHAR(1)  
TITLE\_NO\_FIELD: CHAR(3)  
TITLE\_NO\_SUBFIELD: CHAR(1)  
TITLE\_IND1: CHAR(1)

TITLE\_IND2: CHAR(1)  
PIECE\_FIELD: CHAR(3)  
PIECE\_SUBFIELD: CHAR(1)  
PIECE\_IND1: CHAR(1)  
PIECE\_IND2: CHAR(1)  
LINE\_ITEM\_TYPE\_FIELD: CHAR(3)  
LINE\_ITEM\_TYPE\_SUBFIELD: CHAR(1)  
LINE\_ITEM\_TYPE\_IND1: CHAR(1)  
LINE\_ITEM\_TYPE\_IND2: CHAR(1)  
LINE\_ITEM\_TYPE\_DEFAULT: NUMBER

**INDEX\_TYPE**

INDEX\_TYPE: CHAR(1)  
INDEX\_NAME: VARCHAR2(30)

**INSTRUCTOR**

INSTRUCTOR\_ID: NUMBER  
LAST\_NAME: VARCHAR2(50)  
NORMAL\_LAST\_NAME: VARCHAR2(50)  
FIRST\_NAME: VARCHAR2(40)  
TITLE: VARCHAR2(10)  
CIRC\_CLUSTER\_ID: NUMBER

**INTERVAL\_TYPE**

INTERVAL\_TYPE: CHAR(1)  
INTERVAL\_DESC: VARCHAR2(25)

**INV\_LINE\_ITEM\_NOTES**

INV\_LINE\_ITEM\_ID: NUMBER  
INVOICE\_ID: NUMBER  
NOTE: VARCHAR2(1900)

**INVOICE**

INVOICE\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ACCOUNT\_ID: NUMBER  
INVOICE\_NUMBER: VARCHAR2(25)  
NORMAL\_INVOICE\_NUMBER: VARCHAR2(25)  
INVOICE\_STATUS: NUMBER  
INVOICE\_STATUS\_DATE: DATE

**INVOICE**

INVOICE\_DATE: DATE  
VOUCHER\_NUMBER: VARCHAR2(25)  
CURRENCY\_CODE: VARCHAR2(3)  
CONVERSION\_RATE: NUMBER  
INVOICE\_TOTAL: NUMBER  
BILL\_LOCATION: NUMBER  
INVOICE\_QUANTITY: NUMBER  
LINE\_ITEM\_COUNT: NUMBER  
LINE\_ITEM\_SUBTOTAL: NUMBER  
ADJUSTMENTS\_SUBTOTAL: NUMBER  
TOTAL: NUMBER

INVOICE\_CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
INVOICE\_UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_LOCATION\_ID: NUMBER  
EDI\_REF: NUMBER  
EXPEND\_DATE: DATE  
CHECK\_NUMBER: VARCHAR2(40)  
NORMAL\_CHECK\_NUMBER: VARCHAR2(40)

### **INVOICE\_FUNDS**

INVOICE\_ID: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_ID: NUMBER  
COMMIT\_PENDING: NUMBER  
COMMITMENTS: NUMBER  
EXPEND\_PENDING: NUMBER  
EXPENDITURES: NUMBER

### **INVOICE\_LINE\_ITEM**

INV\_LINE\_ITEM\_ID: NUMBER  
INVOICE\_ID: NUMBER  
LINE\_ITEM\_ID: NUMBER  
UNIT\_PRICE: NUMBER  
LINE\_PRICE: NUMBER  
QUANTITY: NUMBER  
PREPAY\_AMOUNT: NUMBER  
PIECE\_IDENTIFIER: VARCHAR2(50)  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
EDI\_REF: NUMBER

### **INVOICE\_LINE\_ITEM\_FUNDS**

COPY\_ID: NUMBER  
INV\_LINE\_ITEM\_ID: NUMBER  
SPLIT\_FUND\_SEQ: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_ID: NUMBER  
PERCENTAGE: NUMBER  
AMOUNT: NUMBER

### **INVOICE\_NOTE**

INVOICE\_ID: NUMBER  
NOTE: VARCHAR2(1900)

### **INVOICE\_STATUS**

INVOICE\_STATUS: NUMBER  
INVOICE\_STATUS\_DESC: VARCHAR2(25)

### **ISSUES RECEIVED**

ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COPY\_ID: NUMBER  
LOCATION\_ID: NUMBER  
RECEIPT\_DATE: DATE  
OPAC\_SUPPRESSED: NUMBER  
NOTE: VARCHAR2(256)  
COLLAPSED: CHAR(1)  
ITEM\_ID: NUMBER

**ISSUES\_VW**

ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
ENUMCHRON: VARCHAR2(256)  
EXPECTED\_DATE: DATE  
RECEIPT\_DATE: DATE  
RECEIVED: NUMBER

**ITEM**

ITEM\_ID: NUMBER  
PERM\_LOCATION: NUMBER  
TEMP\_LOCATION: NUMBER  
ITEM\_TYPE\_ID: NUMBER  
TEMP\_ITEM\_TYPE\_ID: NUMBER  
COPY\_NUMBER: NUMBER  
ON\_RESERVE: CHAR(1)  
RESERVE\_CHARGES: NUMBER  
PIECES: NUMBER  
PRICE: NUMBER  
SPINE\_LABEL: VARCHAR2(25)  
HISTORICAL\_CHARGES: NUMBER  
HISTORICAL\_BROWSSES: NUMBER  
RECALLS\_PLACED: NUMBER  
HOLDS\_PLACED: NUMBER  
CREATE\_DATE: DATE  
MODIFY\_DATE: DATE  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
MODIFY\_LOCATION\_ID: NUMBER  
ITEM\_SEQUENCE\_NUMBER: NUMBER  
HISTORICAL\_BOOKINGS: NUMBER  
MEDIA\_TYPE\_ID: NUMBER  
SHORT\_LOAN\_CHARGES: NUMBER

**ITEM\_BARCODE**

ITEM\_ID: NUMBER  
ITEM\_BARCODE: VARCHAR2(30)  
BARCODE\_STATUS: NUMBER  
BARCODE\_STATUS\_DATE: DATE

**ITEM\_BARCODE\_STATUS**

BARCODE\_STATUS\_TYPE: NUMBER

BARCODE\_STATUS\_DESC: VARCHAR2(25)

**ITEM\_NOTE**

ITEM\_ID: NUMBER

ITEM\_NOTE: VARCHAR2(1000)

**ITEM\_STAT\_CODE**

ITEM\_STAT\_ID: NUMBER

ITEM\_STAT\_CODE: VARCHAR2(3)

ITEM\_STAT\_CODE\_DESC: VARCHAR2(25)

**ITEM\_STATS**

ITEM\_ID: NUMBER

ITEM\_STAT\_ID: NUMBER

DATE\_APPLIED: DATE

**ITEM\_STATUS**

ITEM\_ID: NUMBER

ITEM\_STATUS: NUMBER

ITEM\_STATUS\_DATE: DATE

**ITEM\_STATUS\_TYPE**

ITEM\_STATUS\_TYPE: NUMBER

ITEM\_STATUS\_DESC: VARCHAR2(25)

**ITEM\_TYPE**

ITEM\_TYPE\_ID: NUMBER

ITEM\_TYPE\_CODE: VARCHAR2(10)

ITEM\_TYPE\_NAME: VARCHAR2(25)

ITEM\_TYPE\_DISPLAY: VARCHAR2(40)

**ITEM\_TYPE\_MAPPING**

IMPORT\_RULE\_ID: NUMBER

MARC\_ITEM\_TYPE: VARCHAR2(50)

ITEM\_TYPE\_ID: NUMBER

MARC\_LOCATION: VARCHAR2(50)

LOCATION\_ID: NUMBER

CALL\_NO\_HIERARCHY\_ID: NUMBER

**ITEM\_TYPE\_POLICY**

CIRC\_GROUP\_ID: NUMBER

ITEM\_TYPE\_ID: NUMBER

REPLACE\_COST: NUMBER

ORDER\_QUANTITY\_NUMBER: NUMBER

REORDER\_POINT: NUMBER

SHORT\_LOAN: CHAR(1)

**ITEM\_VW**

MFHD\_ID: NUMBER

CALL\_NO: VARCHAR2(300)

CALL\_NO\_TYPE: CHAR(1)

NORMALIZED\_CALL\_NO: VARCHAR2(300)

ITEM\_ID: NUMBER

BARCODE: VARCHAR2(30)

PERM\_LOCATION\_CODE: VARCHAR2(10)  
PERM\_LOCATION: VARCHAR2(25)  
GOV\_LOCATION\_CODE: VARCHAR2(10)  
GOV\_LOCATION: VARCHAR2(25)  
PERM\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
PERM\_ITEM\_TYPE: VARCHAR2(25)  
GOV\_ITEM\_TYPE\_CODE: VARCHAR2(10)  
GOV\_ITEM\_TYPE: VARCHAR2(25)  
MEDIA\_TYPE\_CODE: VARCHAR2(10)  
MEDIA\_TYPE: VARCHAR2(50)  
ENUMERATION: VARCHAR2(80)  
CHRONOLOGY: VARCHAR2(80)  
YEAR: VARCHAR2(20)  
CAPTION: VARCHAR2(256)  
HISTORICAL\_BROWSES: NUMBER  
HISTORICAL\_CHARGES: NUMBER  
HISTORICAL\_BOOKINGS: NUMBER  
HOLDS\_PLACED: NUMBER  
RECALLS\_PLACED: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)

**LCCLASS\_VW**

MFHD\_ID: NUMBER  
FIRSTLETTER: VARCHAR2(1)  
CLASS: VARCHAR2(300)  
LONGCLASS: VARCHAR2(7)  
CLASSNUMBER: NUMBER

**LEDGER**

LEDGER\_ID: NUMBER  
FISCAL\_YEAR\_ID: NUMBER  
ACQ\_POLICY\_ID: NUMBER  
LEDGER\_NAME: VARCHAR2(40)  
NORMAL\_LEDGER\_NAME: VARCHAR2(40)  
OVERCOMMIT: CHAR(1)  
OVERCOMMIT\_WARN: NUMBER  
OVERCOMMIT\_PERCENT: NUMBER  
COMMIT\_FREEZE: DATE  
UNDERCOMMIT\_PERCENT: NUMBER  
OVEREXPEND: CHAR(1)  
OVEREXPEND\_WARN: NUMBER  
OVEREXPEND\_PERCENT: NUMBER  
EXPEND\_FREEZE: DATE  
UNDEREXPEND\_PERCENT: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
RULE\_ID: NUMBER  
NEW\_LEDGER\_NAME: VARCHAR2(40)  
NORMAL\_NEW\_LEDGER\_NAME: VARCHAR2(40)

NEW\_LEDGER\_ID: NUMBER

**LEDGER\_LOCATIONS**

LEDGER\_ID: NUMBER

LOCATION\_ID: NUMBER

**LEDGER\_NOTE**

LEDGER\_ID: NUMBER

NOTE: VARCHAR2(1900)

**LIBRARY**

LIBRARY\_ID: NUMBER

LIBRARY\_NAME: VARCHAR2(50)

LIBRARY\_DISPLAY\_NAME: VARCHAR2(80)

NUC\_CODE: VARCHAR2(15)

**LIBRARY\_ADDRESS\_DEFAULT**

LIBRARY\_NAME: VARCHAR2(50)

CONTACT\_NAME: VARCHAR2(50)

EMAIL: VARCHAR2(50)

ADDRESS\_LINE1: VARCHAR2(50)

ADDRESS\_LINE2: VARCHAR2(50)

ADDRESS\_LINE3: VARCHAR2(50)

ADDRESS\_LINE4: VARCHAR2(50)

ADDRESS\_LINE5: VARCHAR2(50)

CITY: VARCHAR2(30)

STATE\_PROVINCE: VARCHAR2(7)

ZIP\_POSTAL: VARCHAR2(10)

COUNTRY: VARCHAR2(20)

SAN: VARCHAR2(10)

**LINE\_ITEM**

LINE\_ITEM\_ID: NUMBER

PO\_ID: NUMBER

BIB\_ID: NUMBER

LINE\_ITEM\_TYPE: NUMBER

LINE\_ITEM\_NUMBER: NUMBER

PIECE\_IDENTIFIER: VARCHAR2(50)

UNIT\_PRICE: NUMBER

LINE\_PRICE: NUMBER

PRINT\_STD\_NUM: CHAR(2)

QUANTITY: NUMBER

PREPAY\_AMOUNT: NUMBER

RUSH: CHAR(1)

CLAIM\_INTERVAL: NUMBER

CANCEL\_INTERVAL: NUMBER

DONOR: VARCHAR2(50)

REQUESTOR: VARCHAR2(50)

VENDOR\_TITLE\_NUM: VARCHAR2(25)

VENDOR\_REF\_QUAL: VARCHAR2(3)

VENDOR\_REF\_NUM: VARCHAR2(35)

CREATE\_DATE: DATE

CREATE\_OPID: VARCHAR2(10)

UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
EDI\_REF: NUMBER

**LINE\_ITEM\_COPY**

LINE\_ITEM\_ID: NUMBER  
LOCATION\_ID: NUMBER  
SHIP\_TO\_LOCATION: NUMBER  
COPY\_COUNT: NUMBER  
USE\_LEDGER: NUMBER  
USE\_FUND: NUMBER

**LINE\_ITEM\_COPY\_HISTORY**

COPY\_ID: NUMBER  
AUDIT\_ID: NUMBER  
LINE\_ITEM\_STATUS: NUMBER  
STATUS\_DATE: DATE  
INV\_LINE\_ITEM\_ID: NUMBER

**LINE\_ITEM\_COPY\_STATUS**

LINE\_ITEM\_ID: NUMBER  
LOCATION\_ID: NUMBER  
COPY\_ID: NUMBER  
MFHD\_ID: NUMBER  
LINE\_ITEM\_STATUS: NUMBER  
INVOICE\_ITEM\_STATUS: NUMBER  
STATUS\_DATE: DATE  
ITEM\_ID: NUMBER

**LINE\_ITEM\_FUNDS**

COPY\_ID: NUMBER  
SPLIT\_FUND\_SEQ: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_ID: NUMBER  
PERCENTAGE: NUMBER  
PREPAY\_PERCENTAGE: NUMBER  
AMOUNT: NUMBER  
PREPAY: NUMBER

**LINE\_ITEM\_NOTES**

LINE\_ITEM\_ID: NUMBER  
PO\_ID: NUMBER  
PRINT\_NOTE: VARCHAR2(60)  
NOTE: VARCHAR2(1900)

**LINE\_ITEM\_STATUS**

LINE\_ITEM\_STATUS: NUMBER  
LINE\_ITEM\_STATUS\_DESC: VARCHAR2(25)

**LINE\_ITEM\_TYPE**

LINE\_ITEM\_TYPE: NUMBER  
LINE\_ITEM\_TYPE\_DESC: VARCHAR2(25)

**LOADLINK**

ORIGINALID: VARCHAR2(25)  
LIBID: NUMBER  
BIBID: NUMBER  
ITEMTYPE: CHAR(2)

### **LOCATION**

LOCATION\_ID: NUMBER  
LOCATION\_CODE: VARCHAR2(10)  
LOCATION\_NAME: VARCHAR2(25)  
LOCATION\_DISPLAY\_NAME: VARCHAR2(60)  
LOCATION\_SPINE\_LABEL: VARCHAR2(25)  
LOCATION\_OPAC: CHAR(1)  
SUPPRESS\_IN\_OPAC: CHAR(1)  
MFHD\_COUNT: NUMBER  
LIBRARY\_ID: NUMBER

### **LOCATION\_ADDRESS**

ADDRESS\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CONTACT\_NAME: VARCHAR2(50)  
EMAIL: VARCHAR2(50)  
ADDRESS\_LINE1: VARCHAR2(50)  
ADDRESS\_LINE2: VARCHAR2(50)  
ADDRESS\_LINE3: VARCHAR2(50)  
ADDRESS\_LINE4: VARCHAR2(50)  
ADDRESS\_LINE5: VARCHAR2(50)  
CITY: VARCHAR2(30)  
STATE\_PROVINCE: VARCHAR2(7)  
ZIP\_POSTAL: VARCHAR2(10)  
COUNTRY: VARCHAR2(20)  
SAN: VARCHAR2(10)  
SHIP\_TO\_ADDRESS: CHAR(1)  
BILL\_TO\_ADDRESS: CHAR(1)  
STREET\_ADDRESS: CHAR(1)  
CAMPUS\_ADDRESS: CHAR(1)  
CIRC\_DESK\_ADDRESS: CHAR(1)  
OTHER\_ADDRESS: CHAR(1)

### **LOCATION\_LIMIT**

LOCATION\_LIMIT\_ID: NUMBER  
LIMIT\_NAME: VARCHAR2(60)  
LIMIT\_CODE: VARCHAR2(10)  
SUPPRESS\_IN\_OPAC: CHAR(1)

### **LOCATION\_LIMIT\_LOCS**

LOCATION\_LIMIT\_ID: NUMBER  
LOCATION\_ID: NUMBER

### **LOCATION\_PHONE**

PHONE\_ID: NUMBER  
ADDRESS\_ID: NUMBER  
PHONE\_TYPE: NUMBER

PHONE\_NUMBER: VARCHAR2(25)

#### **MAINTENANCE**

MAINT\_ID: NUMBER  
EQUIP\_ID: NUMBER  
DATE\_IN: DATE  
DATE\_OUT: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_OPID: VARCHAR2(10)  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_LOCATION\_ID: NUMBER

#### **MAINTENANCE\_DETAIL**

MAINT\_DTL\_ID: NUMBER  
MAINT\_ID: NUMBER  
MAINT\_TYPE\_ID: NUMBER  
DETAIL\_COMMENT: VARCHAR2(100)

#### **MAINTENANCE\_NOTE**

MAINT\_ID: NUMBER  
NOTE: VARCHAR2(2000)  
OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE

#### **MAINTENANCE\_QUEUE**

RELEASE\_PROCESSED: VARCHAR2(30)  
MAINTENANCE\_CODE: CHAR(1)  
ENQUEUE\_DATE: DATE  
PROCESS\_DATE: DATE  
CAUSATION\_COMMENT: VARCHAR2(2000)

#### **MAINTENANCE\_TYPE**

MAINT\_TYPE\_ID: NUMBER  
TYPE\_CODE: VARCHAR2(10)  
TYPE: VARCHAR2(50)

#### **MAP\_INDEX**

BIB\_ID: NUMBER  
MAP\_INDEX\_ID: NUMBER  
WEST\_LONGITUDE\_DISPLAY: VARCHAR2(12)  
WEST\_LONGITUDE\_NORMAL: NUMBER  
EAST\_LONGITUDE\_DISPLAY: VARCHAR2(12)  
EAST\_LONGITUDE\_NORMAL: NUMBER  
NORTH\_LATITUDE\_DISPLAY: VARCHAR2(12)  
NORTH\_LATITUDE\_NORMAL: NUMBER  
SOUTH\_LATITUDE\_DISPLAY: VARCHAR2(12)  
SOUTH\_LATITUDE\_NORMAL: NUMBER

#### **MAP\_INDEX\_G\_RING**

MAP\_INDEX\_ID: NUMBER

SEQNUM: NUMBER  
G\_RING\_LATITUDE: VARCHAR2(12)  
G\_RING\_LATITUDE\_NORMAL: NUMBER  
G\_RING\_LONGITUDE: VARCHAR2(12)  
G\_RING\_LONGITUDE\_NORMAL: NUMBER

**MAP\_INDEX\_SCALE**

MAP\_INDEX\_ID: NUMBER  
MAP\_SCALE: NUMBER  
SCALE\_TYPE: CHAR(1)

**MARCBOOK\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
AUDIENCE: VARCHAR2(1)  
ITEMFORM: VARCHAR2(1)  
GOVERNMENTPUB: VARCHAR2(1)  
CONFERENCEPUB: VARCHAR2(1)  
LITERARYFORM: VARCHAR2(1)  
BIOGRAPHY: VARCHAR2(1)

**MARCCOMPUTER\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
AUDIENCE: VARCHAR2(1)  
FILETYPE: VARCHAR2(1)  
GOVERNMENTPUB: VARCHAR2(1)

**MARCMAP\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
PROJECTION: VARCHAR2(2)  
CARTOGRAPHICTYPE: VARCHAR2(1)  
GOVERNMENTPUB: VARCHAR2(1)  
INDEXED: VARCHAR2(1)

**MARCMUSIC\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
COMPOSITIONFORM: VARCHAR2(2)  
MUSICFORMAT: VARCHAR2(1)  
AUDIENCE: VARCHAR2(1)  
ITEMFORM: VARCHAR2(1)

**MARCSERIAL\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
FREQUENCY: VARCHAR2(1)

REGULARITY: VARCHAR2(1)  
TYPE: VARCHAR2(1)  
ORIGINALFORM: VARCHAR2(1)  
ITEMFORM: VARCHAR2(1)  
ENTIRENATURE: VARCHAR2(1)  
GOVERNMENTPUB: VARCHAR2(1)  
CONFERENCEPUB: VARCHAR2(1)

**MARCVISUAL\_VW**

BIB\_ID: NUMBER  
BIBTYPE: VARCHAR2(1)  
BIBLEVEL: VARCHAR2(1)  
RUNNINGTIME: VARCHAR2(3)  
AUDIENCE: VARCHAR2(1)  
GOVERNMENTPUB: VARCHAR2(1)  
VISUALTYPE: VARCHAR2(1)  
TECHNIQUE: VARCHAR2(1)

**MARK\_REASON**

MARK\_REASON\_ID: NUMBER  
CLAIM\_TYPE: NUMBER  
MARK\_REASON\_NAME: VARCHAR2(25)

**MARKED\_ISSUE**

MARKED\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
ISSUE\_ID: NUMBER  
SUBSCRIPTION\_ID: NUMBER  
MARK\_REASON: NUMBER  
LOCATION\_ID: NUMBER  
COPY\_ID: NUMBER  
OP\_ID: VARCHAR2(10)  
MARK\_DATE: DATE  
MARKED\_COMMENT: VARCHAR2(250)

**MARKED\_LINE\_ITEM**

MARKED\_ID: NUMBER  
LINE\_ITEM\_ID: NUMBER  
LOCATION\_ID: NUMBER  
COPY\_ID: NUMBER  
MARK\_REASON: NUMBER  
OP\_ID: VARCHAR2(10)  
MARK\_DATE: DATE  
MARKED\_COMMENT: VARCHAR2(250)

**MASTER\_OPERATOR**

OPERATOR\_ID: VARCHAR2(10)  
MASTER\_PROFILE\_ID: NUMBER

**MASTER\_PROFILE**

MASTER\_PROFILE\_ID: NUMBER  
MASTER\_PROFILE\_NAME: VARCHAR2(25)  
SECURITY: CHAR(1)

SYSTEM\_DEFINITIONS: CHAR(1)  
CURRENCY\_TABLES: CHAR(1)  
ACQ\_POLICIES: CHAR(1)  
CAT\_POLICIES: CHAR(1)  
CIRC\_POLICIES: CHAR(1)  
MEDIA\_POLICIES: CHAR(1)  
CLUSTER\_CREATE: CHAR(1)  
CLUSTER\_EDIT: CHAR(1)  
CLUSTER\_DELETE: CHAR(1)  
CLUSTER\_VIEW: CHAR(1)  
PATRON\_GROUP\_EDIT: CHAR(1)

**MASTER\_SECURITY\_LOCS**

MASTER\_PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER

**MEDIA\_BOOKING\_EXCEPTION**

MEDIA\_BOOKING\_EXCEPTION\_ID: NUMBER  
PATRON\_ID: NUMBER  
OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
LOCATION\_ID: NUMBER  
ITEM\_ID: NUMBER  
ITEM\_STATUS\_TYPE: NUMBER  
EQUIP\_ID: NUMBER  
EQUIP\_STS\_TYPE\_ID: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
ROOM\_STS\_TYPE\_ID: NUMBER  
ACTION: CHAR(1)

**MEDIA\_BOOKING\_EXCEPTION\_TYPE**

MEDIA\_BOOKING\_EXCEPTION\_ID: NUMBER  
MEDIA\_BOOKING\_EXCEPTION: VARCHAR2(25)

**MEDIA\_BOOKING\_TYPE**

MEDIA\_BOOKING\_TYPE\_ID: NUMBER  
TYPE: VARCHAR2(20)

**MEDIA\_OPERATOR**

PATRON\_ID: NUMBER  
STATUS: CHAR(1)  
UPDATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE

**MEDIA\_POLICY\_DELIVERY\_CALENDAR**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
CALENDAR\_ID: NUMBER

**MEDIA\_POLICY\_EQUIP\_SETTINGS**

SETTINGS\_ID: NUMBER  
SETTINGS\_NAME: VARCHAR2(40)  
CAN\_DELIVER: CHAR(1)  
CAN\_PICKUP: CHAR(1)  
BOOKING\_INTERVAL: CHAR(1)  
BOOKING\_PERIOD\_MAX: NUMBER  
BOOKING\_RENEW: NUMBER  
BOOKING\_RENEW\_COUNT: NUMBER  
FINE\_INTERVAL: CHAR(1)  
FINE\_RATE\_PICKUP: NUMBER  
FINE\_RATE\_DELIVERY: NUMBER  
FINE\_MAX: NUMBER  
FINE\_GRACE\_PERIOD: NUMBER  
USAGE\_FEE: NUMBER  
USAGE\_RATE: NUMBER  
USAGE\_RATE\_INTERVAL: CHAR(1)  
USAGE\_RATE\_PERIOD: NUMBER

**MEDIA\_POLICY\_EQUIPMENT\_MATRIX**

MATRIX\_ID: NUMBER  
MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
EQUIP\_TYPE\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
SETTINGS\_ID: NUMBER

**MEDIA\_POLICY\_EQUIPMENT\_TYPE**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
EQUIP\_TYPE\_ID: NUMBER  
REPLACEMENT\_DEFAULT: NUMBER  
SETUP\_TIME: NUMBER  
CLEANUP\_TIME: NUMBER  
REQUEST\_EQUIP\_USING\_OPAC: CHAR(1)

**MEDIA\_POLICY\_GROUP**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
SCHEDULE\_POLICY: VARCHAR2(40)  
WARNING\_INTERVAL: NUMBER  
BLOCK\_INTERVAL: NUMBER  
DELIVERY\_COUNT\_CLOSED: CHAR(1)  
DELIVERY\_COUNT\_CLOSED\_FEES: CHAR(1)  
PICKUP\_COUNT\_CLOSED: CHAR(1)  
PICKUP\_COUNT\_CLOSED\_FEES: CHAR(1)  
CANCEL\_UNCLAIMED\_BOOKING: NUMBER  
OVERDUE\_CONFLICT\_LIST\_INTERVAL: NUMBER  
OVERDUE\_FIRST\_INTERVAL: NUMBER  
OVERDUE\_NOTICE\_INTERVAL: NUMBER  
OVERDUE\_NOTICE\_COUNT: NUMBER  
OVERDUE\_LOST\_INTERVAL: NUMBER  
OVERDUE\_LOST\_FEE: CHAR(1)  
OVERDUE\_LOST\_FEE\_AMT: NUMBER  
OVERDUE\_LOST\_MAX\_FINE: CHAR(1)

OVERDUE\_RENEW: CHAR(1)

### **MEDIA\_POLICY\_ITEM\_MATRIX**

MATRIX\_ID: NUMBER  
MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
MEDIA\_TYPE\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
SETTINGS\_ID: NUMBER

### **MEDIA\_POLICY\_ITEM\_SETTINGS**

SETTINGS\_ID: NUMBER  
SETTINGS\_NAME: VARCHAR2(40)  
CAN\_PICKUP: CHAR(1)  
CAN\_DELIVER: CHAR(1)  
BOOKING\_INTERVAL: CHAR(1)  
BOOKING\_PERIOD\_MAX: NUMBER  
BOOKING\_RENEW: NUMBER  
BOOKING\_RENEW\_COUNT: NUMBER  
FINE\_INTERVAL: CHAR(1)  
FINE\_RATE\_PICKUP: NUMBER  
FINE\_RATE\_DELIVERY: NUMBER  
FINE\_MAX: NUMBER  
FINE\_GRACE\_PERIOD: NUMBER  
USAGE\_FEE: NUMBER  
USAGE\_RATE: NUMBER  
USAGE\_RATE\_INTERVAL: CHAR(1)  
USAGE\_RATE\_PERIOD: NUMBER  
RECALL\_FOR\_BOOKING: CHAR(1)

### **MEDIA\_POLICY\_LOCATION**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
LOCATION\_ID: NUMBER  
BOOKING: CHAR(1)  
PICKUP: CHAR(1)  
PICKUP\_SLIP\_PRINT: CHAR(1)  
DELIVERY: CHAR(1)  
DELIVERY\_SLIP\_PRINT: CHAR(1)  
COLLECT\_FINES: CHAR(1)  
COURTESY\_DISCHARGE\_ITEM: CHAR(1)  
COURTESY\_DISCHARGE\_EQUIPMENT: CHAR(1)  
COURTESY\_DISCHARGE\_ROOM\_KEY: CHAR(1)  
ITEM\_SHELVING\_INTERVAL: CHAR(1)  
ITEM\_SHELVING\_PERIOD: NUMBER  
ITEM\_TRANSIT\_PERIOD: NUMBER  
FLY\_ITEM\_SUPPRESS: CHAR(1)  
FLY\_ITEM\_TYPE: NUMBER  
FLY\_ITEM\_LOCATION: NUMBER  
PRINT\_LOCATION\_ID: NUMBER  
PRINT\_CONFIRMATION: CHAR(1)  
DELIVERY\_TIME: NUMBER  
RETURN\_TIME: NUMBER  
EQUIP\_RESTRICTED: CHAR(1)

**MEDIA\_POLICY\_MEDIA\_TYPE**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
MEDIA\_TYPE\_ID: NUMBER  
REPLACEMENT\_DEFAULT: NUMBER  
REQUEST\_ITEM\_USING\_OPAC: CHAR(1)

**MEDIA\_POLICY\_PATRON\_GROUP**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
OVERLAPPING\_BOOKINGS: CHAR(1)  
FINES\_APPLY: CHAR(1)  
FEES\_APPLY: CHAR(1)  
OUTSTANDING\_BALANCE\_LIMIT: CHAR(1)  
OUTSTANDING\_BALANCE\_MAX: NUMBER  
BOOKING\_LIMIT: CHAR(1)  
BOOKING\_MAX: NUMBER  
ITEM\_BOOKING\_LIMIT: CHAR(1)  
ITEM\_BOOKING\_MAX: NUMBER  
EQUIP\_BOOKING\_LIMIT: CHAR(1)  
EQUIP\_BOOKING\_MAX: NUMBER  
LATE\_RETURN\_LIMIT: CHAR(1)  
LATE\_RETURN\_MAX: NUMBER  
CANCELLED\_BOOKING\_LIMIT: CHAR(1)  
CANCELLED\_BOOKING\_MAX: NUMBER  
UNCLAIMED\_BOOKING\_LIMIT: CHAR(1)  
UNCLAIMED\_BOOKING\_MAX: NUMBER  
EARLY\_PICKUP: NUMBER  
EARLY\_PICKUP\_INTERVAL: CHAR(1)  
OVERDUE\_NOTICE\_APPLY: CHAR(1)  
REQUEST\_PATRON\_USING\_OPAC: CHAR(1)

**MEDIA\_POLICY\_PICKUP\_CALENDAR**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
CALENDAR\_ID: NUMBER

**MEDIA\_POLICY\_ROOM\_CALENDAR**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
CALENDAR\_ID: NUMBER

**MEDIA\_POLICY\_ROOM\_MATRIX**

MATRIX\_ID: NUMBER  
MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
MEDIA\_ROOM\_TYPE\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
SETTINGS\_ID: NUMBER

**MEDIA\_POLICY\_ROOM\_SETTINGS**

SETTINGS\_ID: NUMBER  
SETTINGS\_NAME: VARCHAR2(40)  
CAN\_BOOK: CHAR(1)  
BOOKING\_INTERVAL: CHAR(1)  
BOOKING\_PERIOD\_MAX: NUMBER  
USAGE\_FEE: NUMBER

USAGE\_RATE: NUMBER  
USAGE\_RATE\_INTERVAL: CHAR(1)  
USAGE\_RATE\_PERIOD: NUMBER

#### **MEDIA\_POLICY\_ROOM\_TYPE**

MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
MEDIA\_ROOM\_TYPE\_ID: NUMBER

ROOM\_SCHEDULED: CHAR(1)  
CAN\_DELIVER: CHAR(1)

#### **MEDIA\_ROOM**

MEDIA\_ROOM\_ID: NUMBER  
ROOM\_NO: VARCHAR2(15)  
ROOM\_NO\_NORMALIZED: VARCHAR2(15)  
ROOM\_NAME: VARCHAR2(40)  
ROOM\_NAME\_NORMALIZED: VARCHAR2(40)  
LOCATION\_ID: NUMBER  
MEDIA\_ROOM\_TYPE\_ID: NUMBER  
STORAGE: VARCHAR2(1)  
CAPACITY: NUMBER  
HISTORICAL\_BOOKINGS: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_OPID: VARCHAR2(10)  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_LOCATION\_ID: NUMBER

#### **MEDIA\_ROOM\_DETAIL\_TYPE**

MEDIA\_ROOM\_DTL\_TYPE\_ID: NUMBER  
TYPE\_CODE: VARCHAR2(10)  
TYPE: VARCHAR2(50)  
REPEATABLE: CHAR(1)

#### **MEDIA\_ROOM\_DETAILS**

MEDIA\_ROOM\_DETAILS\_ID: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
MEDIA\_ROOM\_DTL\_TYPE\_ID: NUMBER  
ROOM\_DTL: VARCHAR2(100)

#### **MEDIA\_ROOM\_KEY**

MEDIA\_ROOM\_KEY\_ID: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
KEY\_NO: VARCHAR2(15)

#### **MEDIA\_ROOM\_NOTE\_TYPE**

MEDIA\_ROOM\_NOTE\_TYPE\_ID: NUMBER  
TYPE: VARCHAR2(15)

#### **MEDIA\_ROOM\_NOTES**

MEDIA\_ROOM\_ID: NUMBER  
MEDIA\_ROOM\_NOTE\_TYPE\_ID: NUMBER

NOTE: VARCHAR2(2000)  
OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE

**MEDIA\_ROOM\_STATUS**

MEDIA\_ROOM\_ID: NUMBER  
MEDIA\_ROOM\_STS\_TYPE\_ID: NUMBER  
NOTE: VARCHAR2(100)  
UPDATE\_DATE: DATE  
OP\_ID: VARCHAR2(10)

**MEDIA\_ROOM\_STATUS\_TYPE**

MEDIA\_ROOM\_STS\_TYPE\_ID: NUMBER  
STS\_TYPE: VARCHAR2(40)  
DISPLAY\_PRIORITY: NUMBER  
WARN\_ON\_BOOKING: CHAR(1)  
BLOCK\_BOOKING: CHAR(1)  
MESSAGE: VARCHAR2(50)

**MEDIA\_ROOM\_TYPE**

MEDIA\_ROOM\_TYPE\_ID: NUMBER  
TYPE\_CODE: VARCHAR2(10)  
TYPE: VARCHAR2(50)  
EQUIP\_STORAGE: CHAR(1)

**MEDIA\_SCHEDULE**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_ID\_PICKED\_UP: NUMBER  
CONFIRM\_NO: VARCHAR2(77)  
CONFIRM\_DATE: DATE  
MEDIA\_BOOKING\_TYPE\_ID: NUMBER  
ADMIN\_BOOKING: CHAR(1)  
STAGING\_LOCATION\_ID: NUMBER  
BOOKING\_SETUP\_DATE: DATE  
BOOKING\_SETUP: NUMBER  
BOOKING\_START: DATE  
BOOKING\_END: DATE  
BOOKING\_CLEANUP\_DATE: DATE  
BOOKING\_CLEANUP: NUMBER  
SCHED\_COMMENT: VARCHAR2(1000)  
OPERATOR\_DELIVERY: NUMBER  
OPERATOR\_PICKUP: NUMBER  
WIZARD: CHAR(1)  
CREATE\_LOCATION\_ID: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_DATE: DATE  
UPDATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE

**MEDIA\_SCHEDULE\_ARCHIVE**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_POLICY\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
PATRON\_ID: NUMBER  
CONFIRM\_NO: VARCHAR2(77)  
CONFIRM\_DATE: DATE  
PATRON\_ID\_PICKED\_UP: NUMBER  
STAGING\_LOCATION\_ID: NUMBER  
MEDIA\_BOOKING\_TYPE\_ID: NUMBER  
ADMIN\_BOOKING: CHAR(1)  
BOOKING\_SETUP: NUMBER  
BOOKING\_START: DATE  
BOOKING\_END: DATE  
BOOKING\_CLEANUP: NUMBER  
SCHED\_COMMENT: VARCHAR2(1000)  
BOOKING\_RESULT: NUMBER  
OPERATOR\_DELIVERY: NUMBER  
OPERATOR\_PICKUP: NUMBER  
WIZARD: CHAR(1)  
CREATE\_LOCATION\_ID: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_DATE: DATE  
UPDATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE

**MEDIA\_SCHEDULE\_EQUIP\_ARCHIVE**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
COUNT\_ID: NUMBER  
EQUIP\_TYPE\_ID: NUMBER  
EQUIP\_ID: NUMBER  
FULFILL\_ITEM: CHAR(1)  
IN\_ROOM: CHAR(1)  
UPDATE\_DATE: DATE  
LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_EQUIPMENT**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
COUNT\_ID: NUMBER  
EQUIP\_TYPE\_ID: NUMBER  
EQUIP\_ID: NUMBER  
FULFILL\_ITEM: CHAR(1)  
IN\_ROOM: CHAR(1)  
UPDATE\_DATE: DATE  
LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_FINE**

MEDIA\_SCHEDULE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER

**MEDIA\_SCHEDULE\_FINE\_ARCHIVE**

MEDIA\_SCHEDULE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER

**MEDIA\_SCHEDULE\_ITEM**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
COUNT\_ID: NUMBER  
MEDIA\_TYPE\_ID: NUMBER  
BIB\_ID: NUMBER  
MFHD\_ID: NUMBER  
ITEM\_ID: NUMBER  
UPDATE\_DATE: DATE  
LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_ITEM\_ARCHIVE**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
COUNT\_ID: NUMBER  
MEDIA\_TYPE\_ID: NUMBER  
BIB\_ID: NUMBER  
MFHD\_ID: NUMBER  
ITEM\_ID: NUMBER  
UPDATE\_DATE: DATE  
LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_ROOM**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER  
COUNT\_ID: NUMBER  
MEDIA\_ROOM\_TYPE\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CAPACITY\_NEEDED: NUMBER  
CAPACITY\_OPERATOR: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
ROOM\_KEY\_ID: NUMBER  
UPDATE\_DATE: DATE  
TRANS\_LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_ROOM\_ARCHIVE**

MEDIA\_SCHEDULE\_ID: NUMBER  
MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
FINE\_FEE\_ID: NUMBER

COUNT\_ID: NUMBER  
MEDIA\_ROOM\_TYPE\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CAPACITY\_NEEDED: NUMBER  
CAPACITY\_OPERATOR: NUMBER  
MEDIA\_ROOM\_ID: NUMBER  
ROOM\_KEY\_ID: NUMBER  
UPDATE\_DATE: DATE  
TRANS\_LOCATION\_ID: NUMBER  
OP\_ID: VARCHAR2(10)

**MEDIA\_SCHEDULE\_TRANS\_TYPE**

MEDIA\_SCHEDULE\_TRANS\_TYPE\_ID: NUMBER  
TYPE: VARCHAR2(15)

**MEDIA\_SECURITY\_LOCATION**

MEDIA\_PROFILE\_ID: NUMBER  
LOCATION\_ID: NUMBER

**MEDIA\_SECURITY\_OPERATOR**

MEDIA\_PROFILE\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)

**MEDIA\_SECURITY\_PROFILE**

MEDIA\_PROFILE\_ID: NUMBER  
MEDIA\_PROFILE\_NAME: VARCHAR2(25)  
EQUIP\_ADD: CHAR(1)  
EQUIP\_DELETE: CHAR(1)  
EQUIP\_UPDATE: CHAR(1)  
EQUIP\_VIEW: CHAR(1)  
EQUIP\_MAINT\_ADD: CHAR(1)  
EQUIP\_MAINT\_DELETE: CHAR(1)  
EQUIP\_MAINT\_UPDATE: CHAR(1)  
EQUIP\_MAINT\_VIEW: CHAR(1)  
ROOM\_ADD: CHAR(1)  
ROOM\_DELETE: CHAR(1)  
ROOM\_UPDATE: CHAR(1)  
ROOM\_VIEW: CHAR(1)  
ITEM\_ADD: CHAR(1)  
ITEM\_DELETE: CHAR(1)  
ITEM\_UPDATE: CHAR(1)  
ITEM\_VIEW: CHAR(1)  
PATRON\_ADD: CHAR(1)  
PATRON\_DELETE: CHAR(1)  
PATRON\_UPDATE: CHAR(1)  
PATRON\_VIEW: CHAR(1)  
PATRON\_COUNTERS: CHAR(1)  
BOOKING\_ADD: CHAR(1)  
BOOKING\_UPDATE: CHAR(1)  
BOOKING\_CANCEL: CHAR(1)  
BOOKING\_VIEW: CHAR(1)  
BOOKING\_VIEW\_HISTORICAL: CHAR(1)  
BOOKING\_CHARGE: CHAR(1)

BOOKING\_DISCHARGE: CHAR(1)  
BOOKING\_RENEW: CHAR(1)  
OVERRIDE\_PATRON\_BLOCK: CHAR(1)  
OVERRIDE\_ITEM\_BLOCK: CHAR(1)  
OVERRIDE\_OTHER\_BLOCK: CHAR(1)  
CHANGE\_DISCHARGE\_DATE: CHAR(1)  
FEES\_ADD: CHAR(1)  
FEES\_PAY: CHAR(1)  
FEES\_ADJUST: CHAR(1)  
EQUIP\_BOOKED\_MOVE: CHAR(1)

**MEDIA\_TYPE**

MEDIA\_TYPE\_ID: NUMBER  
TYPE\_CODE: VARCHAR2(10)  
TYPE: VARCHAR2(50)

**MFHD\_DATA**

MFHD\_ID: NUMBER  
SEQNUM: NUMBER  
RECORD\_SEGMENT: VARCHAR2(300)

**MFHD\_HISTORY**

MFHD\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)  
ACTION\_DATE: DATE  
LOCATION\_ID: NUMBER  
ENCODING\_LEVEL: CHAR(1)  
SUPPRESS\_IN\_OPAC: VARCHAR2(1)  
ACTION\_TYPE\_ID: NUMBER

**MFHD\_ITEM**

MFHD\_ID: NUMBER  
ITEM\_ID: NUMBER  
ITEM\_ENUM: VARCHAR2(80)  
CHRON: VARCHAR2(80)  
YEAR: VARCHAR2(20)  
CAPTION: VARCHAR2(256)  
FREETEXT: VARCHAR2(256)

**MFHD\_MASTER**

MFHD\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CALL\_NO\_TYPE: CHAR(1)  
NORMALIZED\_CALL\_NO: VARCHAR2(300)  
DISPLAY\_CALL\_NO: VARCHAR2(300)  
SUPPRESS\_IN\_OPAC: CHAR(1)  
SOURCE\_MODULE: CHAR(1)  
RECORD\_STATUS: CHAR(1)  
RECORD\_TYPE: CHAR(1)  
ENCODING\_LEVEL: CHAR(1)  
FIELD\_007: VARCHAR2(23)  
FIELD\_008: VARCHAR2(32)  
CREATE\_DATE: DATE

UPDATE\_DATE: DATE  
EXPORT\_OK: CHAR(1)  
EXPORT\_OK\_DATE: DATE  
EXPORT\_OK\_OPID: VARCHAR2(10)  
EXPORT\_OK\_LOCATION\_ID: NUMBER  
EXPORT\_DATE: DATE

### **MFHDBLOB\_VW**

MFHD\_ID: NUMBER  
MARC\_RECORD: CLOB

### **MFHDHISTORY\_VW**

MFHD\_ID: NUMBER  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
CREATE\_DATE: DATE  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_OPERATOR\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_LOCATION\_ID: NUMBER

### **MISCELLANEOUS**

PATRON\_ID\_RETAIN: CHAR(1)  
MEDIA\_PATRON\_ID\_RETAIN: CHAR(1)  
DISTRIBUTION\_PATRON\_ID\_RETAIN: CHAR(1)  
BIBREADONLY: CHAR(1)  
MFHDREADONLY: CHAR(1)  
AUTHREADONLY: CHAR(1)  
AUTO\_RETRIEVE\_SYSTEM: CHAR(1)  
UNIQUE\_ID\_OFFSET: NUMBER  
UNIQUE\_ID\_FIELD: VARCHAR2(6)  
CALL\_SLIP\_ITEM\_REQUIRED: CHAR(1)  
OPAC\_ITEM\_SORT: CHAR(1)  
CUSTOM\_1: CHAR(1)  
ON\_SHELF\_HOLD: CHAR(1)  
USE\_DEFAULT\_POLICY: CHAR(1)  
DATABASELANGUAGE: VARCHAR2(30)  
UBPAGING: CHAR(1)

### **MONO\_CLAIM**

CLAIM\_THREAD: NUMBER  
COPY\_ID: NUMBER  
CLAIM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
CLAIM\_TYPE: NUMBER  
CLAIM\_DATE: DATE  
CLAIM\_COUNT: NUMBER  
OVERRIDE CLAIM\_DATE: DATE  
CLAIM\_STATUS: NUMBER  
OP\_ID: VARCHAR2(10)  
NOTE: VARCHAR2(256)  
EDI\_REF: NUMBER

### **MONO\_CLAIM\_ARCHIVE**

CLAIM\_THREAD: NUMBER  
COPY\_ID: NUMBER  
CLAIM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
CLAIM\_TYPE: NUMBER  
CLAIM\_DATE: DATE  
CLAIM\_COUNT: NUMBER  
OVERRIDE\_CLAIM\_DATE: DATE  
CLAIM\_STATUS: NUMBER  
OP\_ID: VARCHAR2(10)  
NOTE: VARCHAR2(256)  
EDI\_REF: NUMBER  
ARCHIVE\_DATE: DATE

**MONO\_SUPPLIER\_REPORT**

AUDIT\_ID: NUMBER  
CLAIM\_ID: NUMBER  
REPORT\_DATE: DATE  
ACTION\_DATE: DATE  
ACTION\_QUANTITY: NUMBER  
REPORT\_TYPE: NUMBER  
EDI\_REF: NUMBER  
NOTE: VARCHAR2(512)

**MY\_OPAC\_DB**

PATRON\_ID: NUMBER  
DB\_ID: NUMBER

**MY\_OPAC\_PREFERENCES**

PATRON\_ID: NUMBER  
SEARCH\_PREFERENCES: VARCHAR2(50)

**NALCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(3)  
LONGCLASS: VARCHAR2(300)

**NLMCLASS\_VW**

MFHD\_ID: NUMBER  
FIRSTLETTER: VARCHAR2(1)  
CLASS: VARCHAR2(300)  
LONGCLASS: VARCHAR2(7)  
CLASSNUMBER: NUMBER

**NO\_FILL\_REASON**

REASON\_ID: NUMBER  
REASON\_CODE: VARCHAR2(10)  
REASON\_DESC: VARCHAR2(50)  
SUPPRESS: CHAR(1)

**NOTE\_TYPE**

NOTE\_TYPE: NUMBER  
NOTE\_DESC: VARCHAR2(25)

**OLDYALECLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(6)

**OPAC\_CHANGE\_TYPE**

OPAC\_CHANGE\_TYPE: NUMBER  
OPAC\_CHANGE\_DESC: VARCHAR2(25)

**OPAC\_CIRC\_SETTINGS**

PATRON\_PURGE\_PERIOD: NUMBER  
SELF\_REG\_DFLT\_PATRON\_GRP: NUMBER

**OPAC\_FORM**

FORM\_ID: NUMBER  
FORM\_CODE: VARCHAR2(10)  
FORM\_NAME: VARCHAR2(40)  
EMAIL: VARCHAR2(100)  
INSTRUCTIONS: VARCHAR2(1000)  
BLANK\_FORM: CHAR(1)  
SUPPRESS\_IN\_OPAC: CHAR(1)  
FORM\_TYPE: CHAR(1)  
OUTPUT\_TYPE: CHAR(1)  
LOGIN: VARCHAR2(15)  
PASSWORD: VARCHAR2(15)  
VOUCHER\_START: NUMBER  
VOUCHER\_END: NUMBER  
VOUCHER\_LAST\_USED: NUMBER  
VOUCHER\_PREFIX: VARCHAR2(4)

**OPAC\_FORM\_DATABASES**

FORM\_ID: NUMBER  
DB\_CODE: VARCHAR2(8)  
DB\_ID: NUMBER

**OPAC\_FORM\_FIELDS**

FORM\_ID: NUMBER  
FIELD\_LABEL: VARCHAR2(20)  
FIELD\_REQUIRED: CHAR(1)  
FIELD\_SEQUENCE: NUMBER  
CLIO\_TAG: VARCHAR2(20)  
MAPPING: VARCHAR2(10)  
OPAC\_CHANGE: NUMBER

**OPAC\_FORM\_PATRON\_GROUP**

FORM\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER

**OPAC\_FORM\_REQUEST\_FILE**

REQUEST\_ID: NUMBER  
FORM\_ID: NUMBER  
BIB\_ID: NUMBER  
MFHD\_ID: NUMBER

ITEM\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
DATE\_REQUESTED: DATE  
FREE\_TEXT1: VARCHAR2(100)  
FREE\_TEXT2: VARCHAR2(100)  
FREE\_TEXT3: VARCHAR2(100)  
FREE\_TEXT4: VARCHAR2(100)  
FREE\_TEXT5: VARCHAR2(100)  
FREE\_TEXT6: VARCHAR2(100)  
NOTE: VARCHAR2(100)  
EXPIRE\_DATE: DATE  
VOUCHER\_NUMBER: NUMBER  
EMAIL\_TEXT: VARCHAR2(2000)

**OPAC\_FORM\_TYPE**

FORM\_TYPE: CHAR(1)  
FORM\_TYPE\_DESC: VARCHAR2(25)

**OPAC\_MESSAGES**

TAB\_NUMBER: NUMBER  
LINE\_NUMBER: NUMBER  
OPAC\_LINE: VARCHAR2(70)

**OPAC\_SEARCH\_LOG**

SEARCH\_DATE: DATE  
STAT\_STRING: VARCHAR2(15)  
SESSION\_ID: VARCHAR2(16)  
SEARCH\_TYPE: VARCHAR2(25)  
SEARCH\_STRING: VARCHAR2(250)  
LIMIT\_FLAG: CHAR(1)  
LIMIT\_STRING: VARCHAR2(250)  
INDEX\_TYPE: CHAR(1)  
RELEVANCE: CHAR(1)  
HYPERLINK: CHAR(1)  
HITS: NUMBER  
SEARCH\_TAB: CHAR(1)  
CLIENT\_TYPE: CHAR(1)  
CLIENT\_IP: VARCHAR2(15)  
DBKEY: VARCHAR2(100)  
REDIRECT\_FLAG: CHAR(1)

**OPERATOR**

OPERATOR\_ID: VARCHAR2(10)  
FIRST\_NAME: VARCHAR2(25)  
MIDDLE\_INITIAL: VARCHAR2(1)  
LAST\_NAME: VARCHAR2(25)  
PASSWORD: VARCHAR2(12)  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
MODIFY\_DATE: DATE

MODIFY\_OPID: VARCHAR2(10)

### **ORDER\_TYPES**

ORDER\_TYPE: NUMBER  
ORDER\_TYPE\_DESC: VARCHAR2(25)

### **PATCH\_REGISTRY**

RELEASE\_PROCESSED: VARCHAR2(30)  
PATCH\_FILE: VARCHAR2(30)  
PATCH\_STATUS: VARCHAR2(30)  
PATCH\_OPID: VARCHAR2(30)  
PATCH\_DATE: DATE

### **PATRON**

PATRON\_ID: NUMBER  
NAME\_TYPE: NUMBER  
LAST\_NAME: VARCHAR2(30)  
NORMAL\_LAST\_NAME: VARCHAR2(30)  
FIRST\_NAME: VARCHAR2(20)  
NORMAL\_FIRST\_NAME: VARCHAR2(20)  
MIDDLE\_NAME: VARCHAR2(20)  
NORMAL\_MIDDLE\_NAME: VARCHAR2(20)  
TITLE: VARCHAR2(20)  
SSAN: VARCHAR2(11)  
INSTITUTION\_ID: VARCHAR2(30)  
NORMAL\_INSTITUTION\_ID: VARCHAR2(30)  
REGISTRATION\_DATE: DATE  
CREATE\_OPERATOR\_ID: VARCHAR2(10)  
HOME\_LOCATION: NUMBER  
CREATE\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)  
MODIFY\_LOCATION\_ID: NUMBER  
MODIFY\_DATE: DATE  
EXPIRE\_DATE: DATE  
PURGE\_DATE: DATE  
CURRENT\_CHARGES: NUMBER  
TOTAL\_FEES\_DUE: NUMBER  
NOTE\_COUNT: NUMBER  
CURRENT\_HOLD\_SHELF: NUMBER  
RECALLS\_PLACED: NUMBER  
HOLDS\_PLACED: NUMBER  
ITEMS\_RECALLED: NUMBER  
HISTORICAL\_CHARGES: NUMBER  
CLAIMS\_RETURN: NUMBER  
LOST\_ITEMS: NUMBER  
SELF\_SHELVED: NUMBER  
COUNTER\_RESET\_DATE: DATE  
COUNTER\_RESET\_OPER\_ID: VARCHAR2(10)  
CURRENT\_BOOKINGS: NUMBER  
LATE\_MEDIA RETURNS: NUMBER

HISTORICAL\_BOOKINGS: NUMBER  
CANCELLED\_BOOKINGS: NUMBER  
UNCLAIMED\_BOOKINGS: NUMBER  
MEDIA\_COUNTER\_RESET\_DATE: DATE  
MEDIA\_COUNTER\_RESET\_OPID: VARCHAR2(10)  
CURRENT\_CALL\_SLIPS: NUMBER  
HISTORICAL\_CALL\_SLIPS: NUMBER  
HISTORICAL\_DISTRIBUTIONS: NUMBER  
CURRENT\_SHORT\_LOANS: NUMBER  
HISTORICAL\_SHORT\_LOANS: NUMBER  
UNCLAIMED\_SHORT\_LOANS: NUMBER  
DB\_ID: NUMBER  
PATRON\_ID\_UB: NUMBER  
CURRENT\_CHARGES\_UB: NUMBER  
HISTORICAL\_CHARGES\_UB: NUMBER  
REQUESTS\_UB: NUMBER  
HISTORICAL\_REQUESTS\_UB: NUMBER  
CLAIMS\_RETURN\_UB: NUMBER  
LOST\_ITEMS\_UB: NUMBER  
TOTAL\_FEES\_DUE\_UB: NUMBER  
SELF\_SHELVED\_UB: NUMBER  
PATRON\_PIN: VARCHAR2(12)  
SUSPENSION\_DATE: DATE  
TOTAL\_DEMERITS: NUMBER  
TOTAL\_DEMERITS\_DUE\_UB: NUMBER

**PATRON\_ADDRESS**

ADDRESS\_ID: NUMBER  
PATRON\_ID: NUMBER  
ADDRESS\_TYPE: NUMBER  
ADDRESS\_STATUS: CHAR(1)  
PROTECT\_ADDRESS: CHAR(1)  
ADDRESS\_LINE1: VARCHAR2(50)  
ADDRESS\_LINE2: VARCHAR2(40)  
ADDRESS\_LINE3: VARCHAR2(40)  
ADDRESS\_LINE4: VARCHAR2(40)  
ADDRESS\_LINE5: VARCHAR2(40)  
CITY: VARCHAR2(30)  
STATE\_PROVINCE: VARCHAR2(7)  
ZIP\_POSTAL: VARCHAR2(10)  
COUNTRY: VARCHAR2(20)  
EFFECT\_DATE: DATE  
EXPIRE\_DATE: DATE  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**PATRON\_BARCODE**

PATRON\_BARCODE\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
PATRON\_BARCODE: VARCHAR2(25)  
BARCODE\_STATUS: NUMBER

BARCODE\_STATUS\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)  
HOME\_BARCODE\_ID: NUMBER  
HOME\_PATRON\_GROUP\_ID: NUMBER

#### **PATRON\_BARCODE\_STATUS**

BARCODE\_STATUS\_TYPE: NUMBER  
BARCODE\_STATUS\_DESC: VARCHAR2(25)

#### **PATRON\_GROUP**

PATRON\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_CODE: VARCHAR2(10)  
PATRON\_GROUP\_NAME: VARCHAR2(25)  
PATRON\_GROUP\_DISPLAY: VARCHAR2(40)  
DEMERITS\_APPLIES: CHAR(1)  
MAX\_DEMERITS: NUMBER  
SUSPENSION\_DAYS: NUMBER  
CIRC\_CLUSTER\_ID: NUMBER  
CHARGED\_STATUS\_DISPLAY: CHAR(1)  
CHARGE\_LIMIT: NUMBER  
CHARGE\_LIMIT\_APPLY: CHAR(1)

#### **PATRON\_GROUP\_ITEM\_TYPE**

PATRON\_GROUP\_ID: NUMBER  
ITEM\_TYPE\_ID: NUMBER  
CHARGE\_LIMIT: NUMBER

#### **PATRON\_GROUP\_POLICY**

CIRC\_GROUP\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
FEES\_APPLIES: CHAR(1)  
MAX\_OUTSTANDING\_BALANCE: NUMBER  
OVERDUE\_NOTICE\_APPLIES: CHAR(1)  
MIN\_BALANCE\_FOR\_NOTICE: NUMBER  
MAX\_ITEM\_LIMIT: CHAR(1)  
ITEM\_LIMIT: NUMBER  
MAX\_OVERDUE\_LIMIT: CHAR(1)  
OVERDUE\_LIMIT: NUMBER  
MAX\_OVERDUE\_RECALL\_LIMIT: CHAR(1)  
OVERDUE\_RECALL\_LIMIT: NUMBER  
MAX\_RECALL\_LIMIT: CHAR(1)  
RECALL\_LIMIT: NUMBER  
MAX\_SELF\_SHELF\_LIMIT: CHAR(1)  
SELF\_SHELF\_LIMIT: NUMBER  
MAX\_CLAIM\_RETURN\_LIMIT: CHAR(1)  
CLAIM\_RETURN\_LIMIT: NUMBER  
MAX\_LOST\_LIMIT: CHAR(1)  
LOST\_LIMIT: NUMBER  
PLACE\_HOLD\_OUTSIDE\_LIB: CHAR(1)  
PLACE\_HOLD\_INSIDE\_LIB: CHAR(1)  
PLACE\_RECALL\_OUTSIDE\_LIB: CHAR(1)  
PLACE\_RECALL\_INSIDE\_LIB: CHAR(1)  
PLACE\_INTERLIB\_LOAN\_REQ: CHAR(1)

PLACE\_PURCHASE\_REQ: CHAR(1)  
COURTESY\_NOTICE\_APPLIES: CHAR(1)  
CALL\_SLIP\_LIMIT: NUMBER  
MAX\_CALL\_SLIPS: CHAR(1)

**PATRON\_GROUP\_POLICY**

PLACE\_CALL\_SLIPS: CHAR(1)  
EMAIL\_COURTESY\_NOTICE: CHAR(1)  
EMAIL\_CANCELLATION\_NOTICE: CHAR(1)  
EMAIL\_ITEM\_AVAILABLE\_NOTICE: CHAR(1)  
EMAIL\_OVERDUE\_NOTICE: CHAR(1)  
EMAIL\_OVERDUE\_NOTICE\_OTHER: CHAR(1)  
EMAIL\_OVERDUE\_RECALL\_NOTICE: CHAR(1)  
EMAIL\_RECALL\_NOTICE: CHAR(1)  
PLACE\_SHORT\_LOAN\_IN\_LIB: CHAR(1)  
MAX\_TOTAL\_SHORT\_LOAN: CHAR(1)  
TOTAL\_SHORT\_LOAN: NUMBER  
MAX\_TITLE\_SHORT\_LOAN: CHAR(1)  
TITLE\_SHORT\_LOAN: NUMBER  
MAX\_DAY\_SHORT\_LOAN: CHAR(1)  
DAY\_SHORT\_LOAN: NUMBER

**PATRON\_NAME\_TYPE**

PATRON\_NAME\_TYPE: NUMBER  
PATRON\_NAME\_DESC: VARCHAR2(25)

**PATRON\_NOTES**

PATRON\_NOTE\_ID: NUMBER  
PATRON\_ID: NUMBER  
NOTE\_TYPE: NUMBER  
ADDRESS\_ID: NUMBER  
NOTE: VARCHAR2(1900)  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**PATRON\_PHONE**

PATRON\_PHONE\_ID: NUMBER  
ADDRESS\_ID: NUMBER  
PHONE\_TYPE: NUMBER  
PHONE\_NUMBER: VARCHAR2(25)  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**PATRON\_STAT\_CODE**

PATRON\_STAT\_ID: NUMBER  
PATRON\_STAT\_CODE: VARCHAR2(3)  
PATRON\_STAT\_DESC: VARCHAR2(25)

**PATRON\_STATS**

PATRON\_ID: NUMBER

PATRON\_STAT\_ID: NUMBER  
DATE\_APPLIED: DATE

## **PATTERN**

PATTERN\_ID: NUMBER  
PATTERN\_NAME: VARCHAR2(40)  
PATTERN\_NAME\_NORM: VARCHAR2(40)  
FREQUENCY\_CODE: CHAR(1)  
LVL1: VARCHAR2(20)  
LVL1\_SCHEME: CHAR(2)  
LVL2: VARCHAR2(20)  
LVL2\_MAX: NUMBER  
LVL2\_SCHEME: CHAR(2)  
LVL2\_NUM\_CONT: NUMBER  
LVL3: VARCHAR2(20)  
LVL3\_MAX: NUMBER  
LVL3\_SCHEME: CHAR(2)  
LVL3\_NUM\_CONT: NUMBER  
LVL4: VARCHAR2(20)  
LVL4\_MAX: NUMBER  
LVL4\_SCHEME: CHAR(2)  
LVL4\_NUM\_CONT: NUMBER  
LVL5: VARCHAR2(20)  
LVL5\_MAX: NUMBER  
LVL5\_SCHEME: CHAR(2)  
LVL5\_NUM\_CONT: NUMBER  
LVL6: VARCHAR2(20)  
LVL6\_MAX: NUMBER  
LVL6\_SCHEME: CHAR(2)  
LVL6\_NUM\_CONT: NUMBER  
ALT\_LVL1: VARCHAR2(20)  
ALT\_LVL1\_SCHEME: CHAR(2)  
ALT\_LVL2: VARCHAR2(20)  
ALT\_LVL2\_MAX: NUMBER  
ALT\_LVL2\_SCHEME: CHAR(2)  
ALT\_LVL2\_NUM\_CONT: NUMBER  
CHRON1: NUMBER  
CHRON2: NUMBER  
CHRON3: NUMBER  
CHRON4: NUMBER  
ALT\_CHRON1: NUMBER  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_LOCATION\_ID: NUMBER

## **PHONE\_TYPE**

PHONE\_TYPE: NUMBER

PHONE\_DESC: VARCHAR2(25)

### **PLAN\_TABLE**

STATEMENT\_ID: VARCHAR2(30)  
TIMESTAMP: DATE  
REMARKS: VARCHAR2(80)  
OPERATION: VARCHAR2(30)  
OPTIONS: VARCHAR2(255)  
OBJECT\_NODE: VARCHAR2(128)  
OBJECT\_OWNER: VARCHAR2(30)  
OBJECT\_NAME: VARCHAR2(30)  
OBJECT\_INSTANCE: NUMBER  
OBJECT\_TYPE: VARCHAR2(30)  
OPTIMIZER: VARCHAR2(255)  
SEARCH\_COLUMNS: NUMBER  
ID: NUMBER  
PARENT\_ID: NUMBER  
POSITION: NUMBER  
COST: NUMBER  
CARDINALITY: NUMBER  
BYTES: NUMBER  
OTHER\_TAG: VARCHAR2(255)  
PARTITION\_START: VARCHAR2(255)  
PARTITION\_STOP: VARCHAR2(255)  
PARTITION\_ID: NUMBER  
OTHER: LONG  
DISTRIBUTION: VARCHAR2(30)  
CPU\_COST: NUMBER  
IO\_COST: NUMBER  
TEMP\_SPACE: NUMBER  
ACCESS\_PREDICATES: VARCHAR2(4000)  
FILTER\_PREDICATES: VARCHAR2(4000)

### **PO\_FUNDS**

PO\_ID: NUMBER  
LEDGER\_ID: NUMBER  
FUND\_ID: NUMBER  
COMMIT\_PENDING: NUMBER  
COMMITMENTS: NUMBER  
EXPEND\_PENDING: NUMBER  
EXPENDITURES: NUMBER

### **PO\_NOTES**

PO\_ID: NUMBER  
PRINT\_NOTE: VARCHAR2(60)  
NOTE: VARCHAR2(1900)

### **PO\_STATUS**

PO\_STATUS: NUMBER  
PO\_STATUS\_DESC: VARCHAR2(25)

**PO\_TYPE**

PO\_TYPE: NUMBER  
PO\_TYPE\_DESC: VARCHAR2(25)

**PO\_TYPE\_RULES**

RULE\_ID: NUMBER  
PO\_TYPE\_ID: NUMBER  
SINGLE\_PART: CHAR(1)  
SP\_INCREASE: NUMBER  
SUBSCRIPTION: CHAR(1)  
SUB\_INCREASE: NUMBER  
MEMBERSHIP: CHAR(1)  
MEM\_INCREASE: NUMBER  
STANDING\_ORDER: CHAR(1)  
SO\_INCREASE: NUMBER  
BLANKET\_ORDER: CHAR(1)  
BO\_INCREASE: NUMBER  
MULTI\_PART: CHAR(1)  
MP\_INCREASE: NUMBER  
APPROVAL: CHAR(1)  
APL\_INCREASE: NUMBER

**PO\_VENDOR\_HISTORY**

AUDIT\_ID: NUMBER  
PO\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ACCOUNT\_ID: NUMBER  
REPLACE\_DATE: DATE  
REPLACE\_OPID: VARCHAR2(10)  
REPLACE\_LOCATION: NUMBER

**PRICE\_ADJUSTMENT**

OBJECT\_TYPE: CHAR(1)  
OBJECT\_ID: NUMBER  
SEQUENCE: NUMBER  
REASON\_ID: NUMBER  
METHOD: NUMBER  
ADJUST\_AMOUNT: NUMBER  
PAYMENT\_ID: NUMBER

**PRINT\_LOCATION**

PRINT\_LOCATION\_ID: NUMBER  
PRINT\_LOCATION\_CODE: VARCHAR2(10)  
PRINT\_LOCATION\_NAME: VARCHAR2(25)  
DEFAULT\_PRINTING: CHAR(1)  
CIRC\_GLOBAL\_PRINTING: CHAR(1)  
ACQ\_GLOBAL\_PRINTING: CHAR(1)  
CAT\_GLOBAL\_PRINTING: CHAR(1)  
MEDIA\_GLOBAL\_PRINTING: CHAR(1)

**PROXY\_PATRON**

PATRON\_BARCODE\_ID: NUMBER  
PATRON\_BARCODE\_ID\_PROXY: NUMBER

CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION: NUMBER  
EXPIRATION\_DATE: DATE

**PURCHASE\_ORDER**

PO\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ACCOUNT\_ID: NUMBER  
PO\_TYPE: NUMBER  
PO\_NUMBER: VARCHAR2(25)  
NORMAL\_PO\_NUMBER: VARCHAR2(25)  
ORDER\_LOCATION: NUMBER  
SHIP\_LOCATION: NUMBER  
BILL\_LOCATION: NUMBER  
CURRENCY\_CODE: VARCHAR2(3)  
CONVERSION\_RATE: NUMBER  
PO\_STATUS: NUMBER  
PO\_STATUS\_DATE: DATE  
PO\_CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
PO\_UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_LOCATION\_ID: NUMBER  
SHIP\_VIA: VARCHAR2(20)  
NOT\_NEEDED\_AFTER: DATE  
RUSH: CHAR(1)  
CLAIM\_INTERVAL: NUMBER  
CANCEL\_INTERVAL: NUMBER  
LINE\_ITEM\_COUNT: NUMBER  
LINE\_ITEM\_SUBTOTAL: NUMBER  
ADJUSTMENTS\_SUBTOTAL: NUMBER  
TOTAL: NUMBER  
EDI\_REF: NUMBER  
PO\_APPROVE\_DATE: DATE  
APPROVE\_OPID: VARCHAR2(10)  
APPROVE\_LOCATION\_ID: NUMBER  
PREPAY\_CONVERSION\_RATE: NUMBER

**RECORDCOUNT\_VW**

RecordType: VARCHAR2(21)  
Count: NUMBER

**REFERENCE\_TYPE**

REFERENCE\_TYPE: CHAR(1)  
REFERENCE\_TYPE\_DESC: VARCHAR2(20)  
DISPLAY\_CONSTANT: VARCHAR2(80)

**REMOTE\_CIRC\_CLUSTER\_CACHE**

DB\_ID: NUMBER  
REMOTE\_CIRC\_CLUSTER\_ID: NUMBER  
REMOTE\_CIRC\_CLUSTER\_CODE: VARCHAR2(10)

REMOTE\_CIRC\_CLUSTER\_NAME: VARCHAR2(100)  
UPDATE\_DATE: DATE

#### **REMOTE\_PATRON\_GROUP\_CACHE**

DB\_ID: NUMBER  
REMOTE\_CIRC\_CLUSTER\_ID: NUMBER  
REMOTE\_PATRON\_GROUP\_ID: NUMBER  
UPDATE\_DATE: DATE

#### **REMOTE\_STORAGE\_QUEUE**

QUEUE\_ID: NUMBER  
LOCATION\_ID: NUMBER  
MESSAGE\_TYPE: CHAR(4)  
ITEM\_BARCODE: VARCHAR2(30)  
ITEM\_ID: NUMBER  
PATRON\_ID: NUMBER  
SENT: CHAR(1)  
PICKUP\_LOCATION\_ID: NUMBER

#### **RENEW\_TRANS\_ARCHIVE**

CIRC\_TRANSACTION\_ID: NUMBER  
RENEW\_DATE: DATE  
RENEW\_DUE\_DATE: DATE  
RENEW\_LOCATION: NUMBER  
RENEW\_TYPE: CHAR(1)  
RENEW\_OPER\_ID: VARCHAR2(10)

#### **RENEW\_TRANSACTIONS**

CIRC\_TRANSACTION\_ID: NUMBER  
RENEW\_DATE: DATE  
RENEW\_DUE\_DATE: DATE  
RENEW\_LOCATION: NUMBER  
RENEW\_TYPE: CHAR(1)  
RENEW\_OPER\_ID: VARCHAR2(10)

#### **REPORT\_TYPES**

REPORT\_TYPE: NUMBER  
EDI\_CODE: VARCHAR2(11)  
REPORT\_TYPE\_DESC: VARCHAR2(70)

#### **REPORTING\_LEVEL**

REPORTING\_LEVEL\_ID: NUMBER  
REPORTING\_LEVEL\_NAME: VARCHAR2(50)

#### **REPORTING\_OPERATOR**

REPORTING\_PROFILE\_ID: NUMBER  
OPERATOR\_ID: VARCHAR2(10)

#### **REPORTING\_PROFILE**

REPORTING\_PROFILE\_ID: NUMBER  
REPORTING\_PROFILE\_NAME: VARCHAR2(25)  
REPORTING\_LEVEL\_ID: NUMBER

DATABASE\_MODEL: CHAR(1)  
SYSADMIN: CHAR(1)  
CATALOGING: CHAR(1)  
ACQUISITIONS\_SERIALS: CHAR(1)  
CIRCULATION\_CALL\_SLIP: CHAR(1)  
MEDIA\_SCHEDULING: CHAR(1)  
OPAC: CHAR(1)  
REMOTE\_STORAGE: CHAR(1)  
LOCAL\_UB: CHAR(1)

**REQUEST\_GROUP**

GROUP\_ID: NUMBER  
GROUP\_CODE: VARCHAR2(10)  
GROUP\_NAME: VARCHAR2(25)

**REQUEST\_GROUP\_LOCATION**

GROUP\_ID: NUMBER  
LOCATION\_ID: NUMBER

**REQUEST\_HISTORY**

CALL\_SLIP\_ID: NUMBER  
DB\_KEY: VARCHAR2(100)  
CIRC\_CLUSTER\_ID: NUMBER  
SEQUENCE: NUMBER  
PROMOTE\_DATE: DATE

**RESERVE\_ITEM\_HISTORY**

ITEM\_ID: NUMBER  
EFFECT\_DATE: DATE  
EXPIRE\_DATE: DATE  
RESERVE\_CHARGES: NUMBER

**RESERVE\_LIST**

RESERVE\_LIST\_ID: NUMBER  
RESERVE\_LOCATION: NUMBER  
RESERVE\_ITEM\_TYPE: NUMBER  
LIST\_TITLE: VARCHAR2(40)  
NORMAL\_LIST\_TITLE: VARCHAR2(40)  
EFFECT\_DATE: DATE  
EXPIRE\_DATE: DATE  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_LOCATION\_ID: NUMBER

**RESERVE\_LIST\_COURSES**

RESERVE\_LIST\_ID: NUMBER  
DEPARTMENT\_ID: NUMBER  
INSTRUCTOR\_ID: NUMBER  
COURSE\_ID: NUMBER  
SECTION\_ID: NUMBER

**RESERVE\_LIST\_EITEMS**

EITEM\_ID: NUMBER  
RESERVE\_LIST\_ID: NUMBER

**RESERVE\_LIST\_ITEMS**

RESERVE\_LIST\_ID: NUMBER  
ITEM\_ID: NUMBER

**ROLLOVER\_AUDIT**

AUDIT\_ID: NUMBER  
RUN\_ID: NUMBER  
RECORD\_ID: NUMBER  
PARENT\_ID: NUMBER  
RECORD\_TYPE: NUMBER  
RESULT\_CODE: NUMBER  
TIMESTAMP: DATE  
OTHER\_INFO: VARCHAR2(50)

**ROLLOVER\_RESULT\_CODES**

RESULT\_CODE: NUMBER  
DESCRIPTION: VARCHAR2(256)

**ROLLOVER\_RULES**

RULE\_ID: NUMBER  
RULE\_NAME: VARCHAR2(25)  
NORMAL\_RULE\_NAME: VARCHAR2(25)  
FISCAL\_PERIOD\_ID: NUMBER  
NEW\_FISCAL\_PERIOD\_ID: NUMBER  
INITIALIZE\_TYPE: CHAR(1)  
CREATE\_DATE: DATE  
CREATE\_OP\_ID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OP\_ID: VARCHAR2(10)  
ACTION\_INDICATOR: CHAR(1)

**ROUTING\_LIST**

ROUTING\_LIST\_ID: NUMBER  
NAME: VARCHAR2(45)  
NORMAL\_NAME: VARCHAR2(45)  
CREATE\_OPID: VARCHAR2(10)  
CREATE\_LOCATION\_ID: NUMBER  
CREATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)  
UPDATE\_LOCATION\_ID: NUMBER  
UPDATE\_DATE: DATE  
NOTE: VARCHAR2(256)  
PRINT\_NOTE: CHAR(1)

**ROUTING\_LIST\_MEMBERS**

ROUTING\_LIST\_ID: NUMBER  
MEMBER\_ID: NUMBER  
MEMBER\_TYPE: CHAR(1)  
RANK: NUMBER

ADD\_DATE: DATE

### **SAVED\_RECORDS\_RESULTS**

PATRON\_ID: NUMBER

DB\_ID: NUMBER

BIB\_ID: NUMBER

SAVE\_DATE: DATE

### **SAVED\_SEARCHES**

SAVED\_SEARCHES\_ID: NUMBER

PATRON\_ID: NUMBER

SEARCH\_DATE: DATE

SEARCH\_TYPE: VARCHAR2(250)

SEARCH\_STRING: VARCHAR2(700)

LIMIT\_FLAG: CHAR(1)

LIMIT\_STRING: VARCHAR2(250)

INDEX\_TYPE: CHAR(1)

RELEVANCE: CHAR(1)

SEARCH\_TAB: CHAR(1)

LAST\_EXECUTED: DATE

SDI\_NEW\_HITS: CHAR(1)

SDI\_INTERVAL\_ID: NUMBER

SEARCH\_PAGE: VARCHAR2(3000)

NUMBER\_HITS: NUMBER

### **SDI\_INTERVALS**

SDI\_INTERVAL\_ID: NUMBER

SDI\_INTERVAL\_CODE: VARCHAR2(10)

SDI\_INTERVAL\_DAYS: NUMBER

### **SEARCHFIELDS**

SEARCHCODE: CHAR(4)

FIELDCODE: CHAR(4)

### **SEARCHPARM**

SEARCHCODE: CHAR(4)

SEARCHNAME: VARCHAR2(40)

INDEXRULES: VARCHAR2(300)

OPACSUPPRESS: CHAR(1)

STAFFSUPPRESS: CHAR(1)

DISPLAYFIELD1: VARCHAR2(30)

DISPLAYFIELD2: VARCHAR2(30)

DISPLAYFIELD3: VARCHAR2(30)

SORTFIELD1: VARCHAR2(30)

SORTFIELD2: VARCHAR2(30)

SORTFIELD3: VARCHAR2(30)

OPACCOUNT: NUMBER

CIRCCOUNT: NUMBER

ACQCOUNT: NUMBER

CATCOUNT: NUMBER

ORDERING: NUMBER

Z3950\_USE\_ATTRIBUTE: NUMBER

MEDIACOUNT: NUMBER

**SERIAL\_CLAIM**

CLAIM\_THREAD: NUMBER  
ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COPY\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CLAIM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
CLAIM\_TYPE: NUMBER  
CLAIM\_DATE: DATE  
CLAIM\_COUNT: NUMBER  
OVERRIDE\_CLAIM\_DATE: DATE  
CLAIM\_STATUS: NUMBER  
OP\_ID: VARCHAR2(10)  
NOTE: VARCHAR2(256)  
EDI\_REF: NUMBER

**SERIAL\_CLAIM\_ARCHIVE**

CLAIM\_THREAD: NUMBER  
ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COPY\_ID: NUMBER  
LOCATION\_ID: NUMBER  
CLAIM\_ID: NUMBER  
VENDOR\_ID: NUMBER  
CLAIM\_TYPE: NUMBER  
CLAIM\_DATE: DATE  
CLAIM\_COUNT: NUMBER  
OVERRIDE\_CLAIM\_DATE: DATE  
CLAIM\_STATUS: NUMBER  
OP\_ID: VARCHAR2(10)  
NOTE: VARCHAR2(256)  
EDI\_REF: NUMBER  
ARCHIVE\_DATE: DATE

**SERIAL\_ISSUES**

ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
ENUMCHRON: VARCHAR2(256)  
LVL1: NUMBER  
LVL2: NUMBER  
LVL3: NUMBER  
LVL4: NUMBER  
LVL5: NUMBER  
LVL6: NUMBER  
ALT\_LVL1: NUMBER  
ALT\_LVL2: NUMBER  
CHRON1: NUMBER  
CHRON2: NUMBER  
CHRON3: NUMBER  
CHRON4: NUMBER  
ALT\_CHRON: NUMBER

EXPECTED\_DATE: DATE  
RECEIPT\_DATE: DATE  
RECEIVED: NUMBER

**SERIAL\_SUPPLIER\_REPORT**

AUDIT\_ID: NUMBER  
CLAIM\_ID: NUMBER  
REPORT\_DATE: DATE  
ACTION\_DATE: DATE  
ACTION\_QUANTITY: NUMBER  
REPORT\_TYPE: NUMBER  
EDI\_REF: NUMBER  
NOTE: VARCHAR2(512)

**SERIALS\_VW**

BIB\_ID: NUMBER  
MFHD\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
COMPONENT\_NAME: VARCHAR2(100)  
COMPONENT\_NAME\_NORM: VARCHAR2(100)  
PREDICT: CHAR(1)  
NEXT\_ISSUE\_ID: NUMBER  
NOTE: VARCHAR2(256)  
ISSUE\_ID: NUMBER  
ENUMCHRON: VARCHAR2(256)  
EXPECTED\_DATE: DATE  
RECEIPT\_DATE: DATE  
RECEIVED: NUMBER

**SHORT\_LOAN**

SHORT\_LOAN\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER  
SHORT\_LOAN\_STATUS\_ID: NUMBER  
STATUS\_DATE: DATE  
BIB\_ID: NUMBER  
MFHD\_ID: NUMBER  
ITEM\_ID: NUMBER  
START\_TIME: DATE  
END\_TIME: DATE  
PICKUP\_LOCATION: NUMBER  
NOTE: VARCHAR2(100)  
CREATE\_DATE: DATE  
CREATE\_LOCATION: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_LOCATION: NUMBER  
UPDATE\_OPID: VARCHAR2(10)

**SHORT\_LOAN\_ARCHIVE**

SHORT\_LOAN\_ID: NUMBER  
PATRON\_ID: NUMBER  
PATRON\_GROUP\_ID: NUMBER

SHORT\_LOAN\_STATUS\_ID: NUMBER  
STATUS\_DATE: DATE  
BIB\_ID: NUMBER  
MFHD\_ID: NUMBER  
ITEM\_ID: NUMBER  
START\_TIME: DATE  
END\_TIME: DATE  
PICKUP\_LOCATION: NUMBER  
NOTE: VARCHAR2(100)  
CREATE\_DATE: DATE  
CREATE\_LOCATION: NUMBER  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_LOCATION: NUMBER  
UPDATE\_OPID: VARCHAR2(10)

### **SHORT\_LOAN\_STATS**

SHORT\_LOAN\_ID: NUMBER  
PATRON\_STAT\_ID: NUMBER

### **SHORT\_LOAN\_STATUS**

SHORT\_LOAN\_STATUS\_ID: NUMBER  
SHORT\_LOAN\_STATUS\_DESC: VARCHAR2(40)

### **SIMUL\_MERGE\_PROFILE**

BIB\_FIELD1: VARCHAR2(30)  
BIB\_FIELD2: VARCHAR2(30)  
BIB\_FIELD3: VARCHAR2(30)  
CITATION\_FIELD1: VARCHAR2(30)  
CITATION\_FIELD2: VARCHAR2(30)  
CITATION\_FIELD3: VARCHAR2(30)  
PSEUDO\_RELEVANCE: CHAR(1)

### **SORT\_GROUP**

SORT\_GROUP\_ID: NUMBER  
SEQUENCE\_NUMBER: NUMBER  
SORT\_GROUP\_DEFAULT: CHAR(1)  
SORT\_GROUP\_NAME: VARCHAR2(40)  
SORT\_GROUP\_CODE: VARCHAR2(8)

### **SORT\_GROUP\_LOCATION**

SORT\_GROUP\_ID: NUMBER  
SEQUENCE\_NUMBER: NUMBER  
LOCATION\_ID: NUMBER

### **SUBDIVISION**

SUBDIV\_ID: NUMBER  
NORMAL\_SUBDIV: VARCHAR2(300)  
DISPLAY\_SUBDIV: VARCHAR2(300)  
SUBDIV\_TYPE: CHAR(1)  
HEADING\_TYPE: CHAR(1)  
AUTHORIZED: CHAR(1)  
CREATE\_DATE: DATE

UPDATE\_DATE: DATE

**SUBDIVISION\_TYPE**

SUBDIV\_TYPE: CHAR(1)  
SUBDIV\_TYPE\_DESC: VARCHAR2(50)

**SUBSCRIPTION**

SUBSCRIPTION\_ID: NUMBER  
LINE\_ITEM\_ID: NUMBER  
START\_DATE: DATE  
SUBSCRIPTION\_LENGTH: NUMBER  
LENGTH\_TYPE: NUMBER  
RENEWAL\_DATE: DATE  
AUTO\_RENEWAL: CHAR(1)  
SICI: VARCHAR2(45)  
NORMAL\_SICI: VARCHAR2(45)  
UPC: VARCHAR2(30)  
NORMAL\_UPC: VARCHAR2(30)  
NOTE: VARCHAR2(256)

**SUDOCCCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(300)  
LONGCLASS: VARCHAR2(12)

**SUPPRESS\_SYSTEM CLAIMS**

COMPONENT\_ID: NUMBER  
ISSUE\_ID: NUMBER

**UB\_CHARGE**

PATRON\_ID: NUMBER  
DB\_ID: NUMBER  
CIRC\_TRANSACTION\_ID: NUMBER  
DUE\_DATE: DATE  
DISCHARGE\_DATE: DATE  
RECALL\_DATE: DATE

**UB\_CHARGE\_ARCHIVE**

PATRON\_ID: NUMBER  
DB\_ID: NUMBER  
CIRC\_TRANSACTION\_ID: NUMBER  
DUE\_DATE: DATE  
DISCHARGE\_DATE: DATE  
RECALL\_DATE: DATE

**UB\_FINE\_FEE**

PATRON\_ID: NUMBER  
DB\_ID: NUMBER  
FINE\_FEE\_TOTAL: NUMBER  
UPDATE\_DATE: DATE  
DEMERITS\_TOTAL: NUMBER

**UB\_HOLD**

PICKUP\_DB\_ID: NUMBER

HOLD\_RECALL\_ID: NUMBER  
PATRON\_ID: NUMBER  
ITEM\_ID: NUMBER

#### **UB\_PATRON\_GROUP\_MAP**

PATRON\_GROUP\_ID: NUMBER  
DB\_ID: NUMBER  
PATRON\_GROUP\_ID\_MAPPED: NUMBER  
MANUAL\_MAP: CHAR(1)  
CIRC\_CLUSTER\_ID: NUMBER  
REMOTE\_CIRC\_CLUSTER\_ID: NUMBER

#### **UB\_PATRON\_RECORD**

PATRON\_ID: NUMBER  
PATRON\_STUB\_ID: NUMBER  
DB\_ID: NUMBER  
CIRC\_CLUSTER\_ID: NUMBER  
CREATE\_DATE: DATE  
UPDATE\_DATE: DATE

#### **UB\_PG\_HOME\_POLICY**

PATRON\_GROUP\_ID: NUMBER  
UB\_ELIGIBLE: CHAR(1)  
FEES\_APPLIES: CHAR(1)  
MAX\_OUTSTANDING\_BALANCE: NUMBER  
MAX\_ITEM\_LIMIT: CHAR(1)  
ITEM\_LIMIT: NUMBER  
MAX\_OVERDUE\_LIMIT: CHAR(1)  
OVERDUE\_LIMIT: NUMBER  
MAX\_OVERDUE\_RECALL\_LIMIT: CHAR(1)  
OVERDUE\_RECALL\_LIMIT: NUMBER  
MAX\_SELF\_SHELVING\_LIMIT: CHAR(1)  
SELF\_SHELVING\_LIMIT: NUMBER  
MAX\_CLAIM\_RETURN\_LIMIT: CHAR(1)  
CLAIM\_RETURN\_LIMIT: NUMBER  
MAX\_LOST\_LIMIT: CHAR(1)  
LOST\_LIMIT: NUMBER  
MAX\_UB\_REQUESTS: CHAR(1)  
UB\_REQUEST\_LIMIT: NUMBER  
MAX\_DEMERITS\_LIMIT: CHAR(1)  
DEMERITS\_LIMIT: NUMBER  
OUTSTANDING\_BALANCE\_LCLBLOCK: CHAR(1)  
ITEM\_LIMIT\_LCLBLOCK: CHAR(1)  
OVERDUE\_LIMIT\_LCLBLOCK: CHAR(1)  
OVERDUE\_RECALL\_LCLBLOCK: CHAR(1)  
SELF\_SHELVING\_LIMIT\_LCLBLOCK: CHAR(1)  
CLAIM\_RETURN\_LIMIT\_LCLBLOCK: CHAR(1)  
LOST\_LIMIT\_LCLBLOCK: CHAR(1)  
UB\_REQUESTS\_LIMIT\_LCLBLOCK: CHAR(1)  
DEMERITS\_LIMIT\_LCLBLOCK: CHAR(1)

#### **UB\_REQUEST**

PATRON\_ID: NUMBER

DB\_ID: NUMBER  
CALL\_SLIP\_ID: NUMBER  
DATE\_REQUESTED: DATE  
REQUEST\_STATUS: VARCHAR2(25)  
STATUS\_DATE: DATE  
NOT\_NEEDED\_AFTER: NUMBER  
HOLDING\_ITEM\_ID: NUMBER  
PICKUP\_DB\_ID: NUMBER

**UB\_REQUEST\_ARCHIVE**

PATRON\_ID: NUMBER  
DB\_ID: NUMBER  
CALL\_SLIP\_ID: NUMBER  
DATE\_REQUESTED: DATE  
REQUEST\_STATUS: VARCHAR2(25)  
STATUS\_DATE: DATE  
NOT\_NEEDED\_AFTER: NUMBER  
HOLDING\_ITEM\_ID: NUMBER  
PICKUP\_DB\_ID: NUMBER

**UB\_REQUEST\_STATUS**

STATUS\_TYPE\_UB: NUMBER  
STATUS\_DESC\_UB: VARCHAR2(25)

**UB\_ROUTING**

UB\_ROUTING\_ID: NUMBER  
DB\_ID\_FROM: NUMBER  
DB\_ID\_TO: NUMBER  
LOCATION\_ID\_TO: NUMBER  
DB\_ID\_PATRON: NUMBER  
PATRON\_ID\_UB: NUMBER  
DB\_ID\_ITEM: NUMBER  
ITEM\_ID\_UB: NUMBER  
SHIPPED\_DATE: DATE  
RECEIVED\_DATE: DATE  
DB\_ID\_RECEIVED: NUMBER

**UB\_ROUTING\_ARCHIVE**

UB\_ROUTING\_ID: NUMBER  
DB\_ID\_FROM: NUMBER  
DB\_ID\_TO: NUMBER  
LOCATION\_ID\_TO: NUMBER  
DB\_ID\_PATRON: NUMBER  
PATRON\_ID\_UB: NUMBER  
DB\_ID\_ITEM: NUMBER  
ITEM\_ID\_UB: NUMBER  
SHIPPED\_DATE: DATE  
RECEIVED\_DATE: DATE  
DB\_ID\_RECEIVED: NUMBER

**UDCCLASS\_VW**

MFHD\_ID: NUMBER  
CLASS: VARCHAR2(6)

## **UNPREDICTABLE\_ISSUES**

ISSUE\_ID: NUMBER  
COMPONENT\_ID: NUMBER  
ENUMCHRON: VARCHAR2(256)  
EXPECTED\_DATE: DATE  
RECEIPT\_DATE: DATE  
RECEIVED: NUMBER

## **VENDOR**

VENDOR\_ID: NUMBER  
VENDOR\_TYPE: CHAR(2)  
NORMAL\_VENDOR\_TYPE: CHAR(2)  
VENDOR\_CODE: VARCHAR2(10)  
NORMAL\_VENDOR\_CODE: VARCHAR2(10)  
VENDOR\_NAME: VARCHAR2(60)  
NORMAL\_VENDOR\_NAME: VARCHAR2(60)  
FEDERAL\_TAX\_ID: VARCHAR2(10)  
INSTITUTION\_ID: VARCHAR2(25)  
DEFAULT\_CURRENCY: VARCHAR2(3)  
CLAIM\_INTERVAL: NUMBER  
CLAIM\_COUNT: NUMBER  
CANCEL\_INTERVAL: NUMBER  
SHIP\_VIA: VARCHAR2(20)  
CREATE\_DATE: DATE  
CREATE\_OPID: VARCHAR2(10)  
UPDATE\_DATE: DATE  
UPDATE\_OPID: VARCHAR2(10)

## **VENDOR\_ACCOUNT**

ACCOUNT\_ID: NUMBER  
VENDOR\_ID: NUMBER  
ACCOUNT\_NUMBER: VARCHAR2(25)  
ACCOUNT\_NAME: VARCHAR2(25)  
DEFAULT\_PO\_TYPE: NUMBER  
DEPOSIT: CHAR(1)  
DEFAULT\_DISCOUNT: NUMBER  
ACCOUNT\_STATUS: NUMBER  
STATUS\_DATE: DATE

## **VENDOR\_ADDRESS**

ADDRESS\_ID: NUMBER  
VENDOR\_ID: NUMBER  
STD\_ADDRESS\_NUMBER: VARCHAR2(8)  
ORDER\_ADDRESS: CHAR(1)  
PAYMENT\_ADDRESS: CHAR(1)  
RETURN\_ADDRESS: CHAR(1)  
CLAIM\_ADDRESS: CHAR(1)  
EMAIL\_ADDRESS: CHAR(1)  
OTHER\_ADDRESS: CHAR(1)  
CONTACT\_NAME: VARCHAR2(40)  
CONTACT\_TITLE: VARCHAR2(40)  
ADDRESS\_LINE1: VARCHAR2(50)

ADDRESS\_LINE2: VARCHAR2(40)  
ADDRESS\_LINE3: VARCHAR2(40)  
ADDRESS\_LINE4: VARCHAR2(40)  
ADDRESS\_LINE5: VARCHAR2(40)  
CITY: VARCHAR2(30)  
STATE\_PROVINCE: VARCHAR2(7)  
ZIP\_POSTAL: VARCHAR2(10)  
COUNTRY: VARCHAR2(20)  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**VENDOR\_BANK\_INFO**

VENDOR\_ID: NUMBER  
ACCOUNT\_NUMBER: VARCHAR2(25)  
TRANSIT\_NUMBER: VARCHAR2(25)  
TAX\_ID\_NUMBER: VARCHAR2(11)  
BANK\_NAME: VARCHAR2(60)  
ADDRESS\_LINE1: VARCHAR2(50)  
ADDRESS\_LINE2: VARCHAR2(40)  
ADDRESS\_LINE3: VARCHAR2(40)  
ADDRESS\_LINE4: VARCHAR2(40)  
ADDRESS\_LINE5: VARCHAR2(40)  
CITY: VARCHAR2(30)  
STATE\_PROVINCE: VARCHAR2(7)  
ZIP\_POSTAL: VARCHAR2(10)  
COUNTRY: VARCHAR2(20)  
PHONE: VARCHAR2(25)  
FAX: VARCHAR2(25)  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**VENDOR\_NOTE**

VENDOR\_ID: NUMBER  
NOTE: VARCHAR2(1900)

**VENDOR\_PHONE**

ADDRESS\_ID: NUMBER  
PHONE\_TYPE: NUMBER  
PHONE\_NUMBER: VARCHAR2(25)  
MODIFY\_DATE: DATE  
MODIFY\_OPERATOR\_ID: VARCHAR2(10)

**VENDOR\_TYPE\_DEFAULTS**

ACQ\_POLICY\_ID: NUMBER  
VENDOR\_TYPE: CHAR(2)  
ORDER\_TYPE: NUMBER  
DISCOUNT: NUMBER  
CLAIM\_INTERVAL: NUMBER  
CLAIM\_COUNT: NUMBER  
CANCEL\_INTERVAL: NUMBER  
SHIP\_VIA: VARCHAR2(20)

**VENDOR\_TYPES**

VENDOR\_TYPE: CHAR(2)  
VENDOR\_TYPE\_DESC: CHAR(40)

### **VENDORINVOICE\_VW**

VENDOR\_CODE: VARCHAR2(10)  
VENDOR\_NAME: VARCHAR2(60)  
VENDOR\_TYPE: CHAR(40)  
INSTITUTION\_ID: VARCHAR2(25)  
BILL\_TO\_LOCATION\_CODE: VARCHAR2(10)  
BILL\_TO\_LOCATION: VARCHAR2(25)  
INVOICE\_DATE: DATE  
INVOICE\_NUMBER: VARCHAR2(25)  
INVOICE\_ID: NUMBER  
CURRENCY\_CODE: VARCHAR2(3)  
CURRENCY\_NAME: VARCHAR2(35)  
INVOICE\_STATUS: VARCHAR2(25)  
INVOICE\_STATUS\_DATE: DATE  
VOUCHER\_NUMBER: VARCHAR2(25)  
EXPENDITURES: NUMBER  
EXPEND\_PENDING: NUMBER  
POLICY\_NAME: VARCHAR2(40)  
FISCAL\_PERIOD\_NAME: VARCHAR2(25)  
FISCAL\_PERIOD\_START: DATE  
FISCAL\_PERIOD\_END: DATE  
LEDGER\_NAME: VARCHAR2(40)  
FUND\_NAME: VARCHAR2(25)  
INSTITUTION\_FUND\_ID: VARCHAR2(50)

### **VENDORORDER\_VW**

VENDOR\_CODE: VARCHAR2(10)  
VENDOR\_NAME: VARCHAR2(60)  
INSTITUTION\_ID: VARCHAR2(25)  
VENDOR\_TYPE: CHAR(40)  
PO\_NUMBER: VARCHAR2(25)  
PO\_TYPE: VARCHAR2(25)  
ORDER\_LOCATION\_CODE: VARCHAR2(10)  
ORDER\_LOCATION: VARCHAR2(25)  
PO\_STATUS: VARCHAR2(25)  
PO\_STATUS\_DATE: DATE  
CURRENCY\_NAME: VARCHAR2(35)  
TOTAL: NUMBER  
UNIT\_PRICE: NUMBER  
QUANTITY: NUMBER  
LINE\_PRICE: NUMBER  
MFHD\_ID: NUMBER  
PO\_LINE\_STATUS: VARCHAR2(25)  
INVOICE\_STATUS: VARCHAR2(25)  
LINE\_STATUS\_DATE: DATE

### **VERSIONS**

MODULE: VARCHAR2(20)  
VERSION: VARCHAR2(30)

SYNCPOINT: NUMBER

### **VOYAGER\_DATABASES**

DB\_CODE: VARCHAR2(10)  
DB\_NAME: VARCHAR2(100)  
DB\_DESC: VARCHAR2(200)  
DB\_PROTOCOL: CHAR(1)  
DB\_TYPE: CHAR(1)  
DB\_SUBTYPE: CHAR(1)  
DB\_WEIGHT: NUMBER  
DATABASE\_NAME: VARCHAR2(50)  
USERID: VARCHAR2(50)  
PASSWORD: VARCHAR2(50)  
MAXHITS: NUMBER  
SEARCHTIMEOUT: NUMBER  
RETRIEVALTIMEOUT: NUMBER  
IMPLEMENTOR: VARCHAR2(5)  
DUP\_PROFILE\_ID: NUMBER  
OPACSUPPRESS: CHAR(1)  
STAFFSUPPRESS: CHAR(1)  
DB\_PUBLIC: CHAR(1)  
MAX\_LICENSE: NUMBER  
STAFF\_POOL: NUMBER  
PUBLIC\_POOL: NUMBER  
ACTION: VARCHAR2(10)  
PUBLIC\_HIGHWATER: NUMBER  
STAFF\_HIGHWATER: NUMBER  
DB\_KEY: VARCHAR2(100)  
DB\_ID: NUMBER  
UB\_DB: CHAR(1)  
CONNECTTIMEOUT: NUMBER  
CHAR\_SET\_ID: NUMBER

### **WOPAC\_PID\_PATRON\_KEYS**

PID: VARCHAR2(80)  
PATRON\_KEY: VARCHAR2(30)

### **Z3950\_ATTRIBUTES**

DB\_CODE: VARCHAR2(8)  
SEARCHCODE: CHAR(4)  
ATTRIBUTES: VARCHAR2(40)  
ATTRIB\_DESC: VARCHAR2(50)  
RH\_TRUNCATION: CHAR(1)  
LH\_TRUNCATION: CHAR(1)  
BOOLEAN\_ENABLED: CHAR(1)  
DB\_ID: NUMBER



---

## UseMARCON Configuration for Use with Voyager

B

---

### What is UseMARCON?

---

The UseMARCON API converts records from one MARC format to another. It interfaces with Voyager by converting records to or from the MARC21 format (used by Voyager) so that incoming records in non-MARC21 format can be used by Voyager and Voyager records can be used by non-MARC21 format processing applications. UseMARCON is used in conjunction with Bulk Import, MARC Export, and Z39.50 servers.

---

### In This Chapter

---

This chapter contains information regarding the UseMARCON API's interaction with Voyager and the Voyager configuration required to make UseMARCON interact properly with Voyager. Throughout this document, the distinction between the processes performed by the UseMARCON API and those specific to Voyager and Z39.50 servers is maintained to facilitate troubleshooting problems.

Functions, files, and messages specific to UseMARCON's actual conversion of MARC records from one MARC format to another are referred to as UseMARCON functions, files, or messages. Functions, files, and messages specific to Bulk Import, MARC Export, and Voyager Z39.50 database interaction are referred to as Voyager functions, files, and messages.

For background information regarding UseMARCON, refer to the Koninklijke Bibliotheek (National Library of the Netherlands) and access the following link:

<http://www.kb.nl/hrd/bibinfra/usemarcon/usema-en.html>

For information regarding the UseMARCON converter and creating and editing system/rules files refer to The British Library website and access the following links:

<http://www.bl.uk/bibliographic/pdfs/guidetousemarcon.pdf>

or

[http://www.bl.uk/bibliographic/pdfs/usemarcon\\_manual.pdf](http://www.bl.uk/bibliographic/pdfs/usemarcon_manual.pdf)

## **Supported MARC Formats**

---

The MARC formats supported by Voyager's interface with UseMARCON are:

- Unimarc
- Intermarc
- Ukmarc
- Normarc
- Librismarc
- Danmarc
- Finmarc
- Canmarc
- Picamarc
- AMARC21
- Ibermarc

## **UseMARCON File Structure**

---

The following section details the file structure of UseMARCON as it pertains to Voyager functions.

## **UseMARCON Conversion Directories**

---

Each conversion consists of *one* direction of MARC format translation *only*. For example, MARC21 to FinMARC and FinMARC to MARC21 are two separate conversions. Therefore, they each require their own conversion directories and

files. These directories contain the files necessary for UseMARCON to make the conversions in that direction, as well as the initialization file to direct UseMARCON to the required files.

Create a `usemarcon` subdirectory in the `ini` directory (`/m1/voyager/xxxdb/ini`) and place the conversion directories in it. A sample would look like Figure B-1.

---

UNI2US	US2UNI	FIN2US	US2FIN
--------	--------	--------	--------

**Figure B-1. Sample Contents of /m1/voyager/xxxdb/ini/usemarcon**

For more information on the UseMARCON conversion directories, see the *UseMARCON Technical Manual*.

Figure B-2 shows sample contents of the UNI2US conversion profile in the sample `usemarcon` subdirectory in Figure B-1 (for the Unimarc to MARC21 format).

---

uni2.ii standard.trs uni2us.rul uni.mrc unius.mrc uni.chk us.chk
---

**Figure B-2. Sample Contents of UNI2US Conversion Profile**

Note that the `uni2.ini` (UseMARCON initialization) file and the six required files are in the sample `usemarcon` subdirectory as detailed in [Required Files on page B-4](#).

For more information on the UseMARCON file structures and the files that it contains, see “System Files” and “Rules Files” sections of the *UseMARCON Technical Manual*.

---

### UseMARCON Initialization (ini) File

---

UseMARCON uses an initialization file for each conversion which points UseMARCON to the files it needs to perform conversions (for that direction). For more information on the UseMARCON initialization file, see the “Initialization File (.ini) section of the “System Files” chapter of the *UseMARCON Technical Manual*.

Each UseMARCON initialization file resides in the conversion directory for its conversion. For more information on the conversion directories, see [UseMARCON Conversion Directories](#) on [page B-2](#).

## **Required Files**

UseMARCON requires certain files to be in each conversion directory. These files must be specified in the UseMARCON initialization file for each conversion.

A sample UseMARCON .ini file can be found in [Sample UseMARCON Initialization File](#) on [page B-5](#).

ErrorLogFile

**▲ IMPORTANT:**

*All of these required files must be readable, and the MARC output file must be writable, using the Voyager ID.*

**Table B-1. Required Fields**

Required File	Description/Function
Error Log File	The path to the error log, usemarcon.errmsgs, is set with the ErrorLogFile= parameter in the initialization file. See <a href="#">Figure B-3</a> .
Rule (.rul) file	Translation file used by UseMARCON to convert one MARC format to another.
Translation (.trs) file	Character mapping table for any special characters that need to be converted when translating from one MARC format to another.
Input format checking (.chk) table	These files act as validity checking devices to make certain that the records being converted contain information in the requisite fields and subfields, especially where there are required fields necessary to convert records.
Output format checking (.chk) table	<p><b>▲ IMPORTANT:</b></p> <p><i>Make certain that the field requirements of these format checking files are not overly restrictive, to avoid a large number of error messages with the conversions.</i></p>

**Table B-1. Required Fields**

Required File	Description/Function
Marc input (.mrc) file, and	UseMARCON requires two empty files, though they are not used in conversion.
Marc output (.mrc) file	

### Sample UseMARCON Initialization File

---

Figure B-3 is a sample UseMARCON initialization (.ini) file containing examples of the required files.

---

Line

```
[DEFAULT_FILES]
ErrorLogFile=/m1/voyager/xxxdb/ini/usemarcon.errmsgs
1      RuleFile=/m1/voyager/xxxdb/ini/usemarcon/uni2us/
          uni2us.rul
2      TranscodingCharacterTable=/m1/voyager/xxxdb/ini/
          usemarcon/uni2us/iso2us.trs
3      InputFormatCheckingTable=/m1/voyager/xxxdb/ini/
          usemarcon/uni2us/uni.chk
4      OutputFormatCheckingTable=/m1/voyager/xxxdb/ini/
          usemarcon/uni2us/us.chk
5      MarcInputFile=/m1/voyager/xxxdb/ini/usemarcon/uni2us/
          uni.mrc
6      MarcOutputFile=/m1/voyager/xxxdb/ini/usemarcon/
          uni2us/unius.mrc
```

**Figure B-3. Sample of the required files in the UseMARCON initialization file**

---

### UseMARCON Log File

---

The UseMARCON log file is called `usemarcon.log`. All Voyager applications that run UseMARCON write log information to this file.

The log file is stored in the directory from which the application is run. In most cases, that is the `/m1/voyager/xxxdb/sbin` directory since that is where the Pscript is run and where `xxxdb` is your database name.

See [Figure B-4](#) for an example log file.

Line#	
1	-----
2	Started at : Thu Aug 16 11:04:07 2001
3	
4	WARNING(2105)-Unexpected field found in input record : Notice '13819' : field '092'
5	ERROR (1300)-Completed translating MARC record. :
6	WARNING(2102)-Invalid first indicator found in input record : Notice '15654' : field '082' (ind ' ')
7	WARNING(2105)-Unexpected field found in input record : Notice '15654' : field '092'
8	ERROR (3001)-Character not transcoded (unable to find it in transco table) : Notice '15654' : field '300' ( Unknown character '%(\be) ) : table 'us2iso.trs'
9	ERROR (5100)-Rule analysis error : ERROR `Syntax error' in rule 041\$a   101\$a (no)   If (I1=0) Then Sto(0); To(3);
10	Redo;
11	Mem(0); From(4) To(6);
12	Redo; Mem(0); From(7) To(9)"
13	ERROR (1300)-Completed translating MARC record. :
14	WARNING(2105)-Unexpected field found in input record : Notice 'UNI0000079' : field '200'
15	WARNING(2105)-Unexpected field found in input record : Notice 'UNI0000079' : field '205'
16	WARNING(2102)-Invalid first indicator found in input record : Notice 'UNI0000079' : field '210' (ind ' ')
17	WARNING(2104)-Invalid or redundant subfield found in input record : Notice 'UNI0000079' : field '210' (subfield '\$c')

**Figure B-4. Sample log file**

Line#	
18	WARNING(2104)-Invalid or redundant subfield found in input record : Notice 'UNI0000079' : field '210' (subfield '\$c')
19	WARNING(2104)-Invalid or redundant subfield found in input record : Notice 'UNI0000079' : field '210' (subfield '\$d')
20	WARNING(2106)-Mandatory field expected in input record : Notice 'UNI0000079' : field '008'
21	WARNING(2106)-Mandatory field expected in input record : Notice 'UNI0000079' : field '245'
22	WARNING(5004)-Only one indicator has been found : I2 is missing in field 700
23	WARNING(7103)-Invalid second indicator found in output record : Notice 'UNI0000080' : field '700' (ind ' ')
24	WARNING(7106)-Mandatory field expected in output record : Notice 'UNI0000080' : field '100'
25	WARNING(7106)-Mandatory field expected in output record : Notice 'UNI0000080' : field '101'
26	WARNING(7106)-Mandatory field expected in output record : Notice 'UNI0000080' : field '200'
27	ERROR (1300)-Completed translating MARC record. :

---

Figure B-4. Sample log file (Continued)

---

### Fatal/Non-Fatal UseMARCON Errors

UseMARCON divides errors into fatal and non-fatal categories. When errors occur, it processes the records and writes the errors to a log.

#### Fatal Errors

Fatal errors occur when something prevents UseMARCON from converting a record. Typically, this occurs when UseMARCON cannot find one of the six required files for translation. For more information on the required files, see [Sample of the required files in the UseMARCON initialization file on page B-5](#).

In the event of a fatal error, UseMARCON writes an error message to the log file for that session for the function being performed during that session stating that UseMARCON could not complete the translation.

The error warning for fatal errors when converting MARC files with UseMARCON, is END non OK. For an example of a UseMARCON fatal error message, see the [Sample Fatal Error \(END non OK\) Message in UseMARCON Log File](#) on [page B-8](#).

### Non-Fatal Errors

Non-Fatal errors occur when UseMARCON encounters a problem that does not prevent it from converting the record that causes the error. An example of this might be a redundant MARC field in a record.

After the record is converted, an error message is created in the log file for that session. For more information on the UseMARCON log file, see [UseMARCON Log File](#) on [page B-5](#).

The two error classifications of Non-fatal errors as they display in the UseMARCON logs are ERROR and WARNING. For the purposes of Voyager's interaction with UseMARCON, the ERROR and WARNING messages mean the same thing.

### **Sample Fatal Error (END non OK) Message in UseMARCON Log File**

---

Figure B-5 is an example of a fatal error message in a UseMARCON log file.

---

-----  
Started at : Thu Aug 16 11:03:09 2001  
  
  
  
END non OK : No rulefile specified

---

**Figure B-5. Sample Fatal Error Message in a UseMARCON Log File**

### **Troubleshooting UseMARCON Error Conditions**

---

UseMARCON generates an error type, an error number, a brief description of the error, and the field in which the error occurred for each error, as a line in the UseMARCON log for each session. This information is what enables troubleshooting of UseMARCON conversion errors.

Use the UseMARCON error messages to determine what changes to make to the conversion files used for that session.

Patterns of recurring errors in the UseMARCON logs (for example, recurring Mandatory field expected in input record messages in the same field) may indicate necessary adjustment of the UseMARCON files (making a field that is considered required by the format checking file a non-required field), or discrepancies/inconsistencies in the records of the database (the field set as required in the format checking file does not exist in many of the records, or is in the wrong field).

### UseMARCON Log File Example

---

For example, the log file in [Figure B-4](#) on [page B-6](#) contains the following error message:

---

WARNING(2105)-Unexpected field found in input record : Notice '13819' : field '092'  
1            2            3            4            5

---

**Figure B-6. Sample UseMARCON Log File Error Message Line**

Table B-2 contains a description of the format of each error message line (the numbers in the left column correspond to the numbered sections of the sample error message in Figure B-6):

**Table B-2. Descriptions of Error Message Lines in UseMARCON Log File**

#	Name	Format	Description
1	Error type	One of three values: <ul style="list-style-type: none"><li>• ERROR</li><li>• WARNING</li><li>• End non OK</li></ul>	Severity of error: <ul style="list-style-type: none"><li>• ERROR and WARNING are both non-fatal.</li><li>• END non OK indicates a fatal error, preventing the conversion of records.</li></ul>

**Table B-2. Descriptions of Error Message Lines in UseMARCON Log File**

#	Name	Format	Description
2	Error number	(nnnn)	UseMARCON error code number. For error codes, see <a href="#">UseMARCON Error Numbers on page B-16</a> .
3	Error description	Text	Brief description of error.
4	UseMARCON record ID number	Notice nnnn	The nnnn variable is the record ID number (which UseMARCON generates for each file) of the record which caused the error.
5	Error field	Field 'nnn'(subfield '\$x')	The nnn indicates the field which caused the error. The x in the subfield error field indicator specifies the subfield which caused the error (if applicable).

**NOTE:**

Fatal errors do not include all of the record specific error information, because the error prevents it from converting any records (for a sample fatal error message in a UseMARCON log file, see [Sample Fatal Error \(END non OK\) Message in UseMARCON Log File on page B-8](#)).

For more information about the error logs generated for the individual Voyager processes (MARC Export, Bulk Import, Z39.50), see the individual Error Reporting discussions for each Voyager process.

## **Bulk Import and UseMARCON**

This section discusses how UseMARCON and the bulk import program interact.

## Overview

---

UseMARCON interacts with Bulk Import by translating records specified in the **bulkimport** command into MARC21 format before importing the records into your Voyager database. Using the command line option **-y** allows you to specify the location of a UseMARCON initialization file which points Bulk Import to the translation files it needs to translate the records into MARC21 format before it runs Bulk Import as usual.

To convert a batch of non-MARC21 format records to MARC21 format when using Bulk Import, add the **-y** switch to the **pbulkimport** command, followed by the path and filename of the UseMARCON initialization file you wish to use for the conversion. The initialization file points the Bulk Import command to the directory and files needed by UseMARCON to perform the conversion.

## Bulk Import Example

---

The following example shows a sample Bulk Import command line with a UseMARCON activation switch.

```
pbulkimport -f/m1/xxxdb/sbin/marcfile.txt -imarcive -y/m1/  
xxxdb/ini/usemarcon/UNI2US/uni2us.ini
```



### IMPORTANT:

*UseMARCON translations are for Bibliographic records and Authority records only. See [Bulk Import, Replace, and Merge of MARC Records](#) on page 12-1 for more information.*

## Bulk Import Error Reporting

---

When UseMARCON encounters errors during Bulk Import batch jobs, it generates error messages in the log file for that session.

Two possible error messages can occur in the Bulk Import log for each session.

- For *each* record not translated, a *record not translated* message is sent to the Bulk Import log. Check the UseMARCON log file for the session to determine whether the error causing the *record not translated* message prevented translation completely or just affected some records.
- If the UseMARCON initialization file is missing, in the wrong place, or not configured properly, an *ini path not valid* message is sent to the log file. This message indicates a fatal error (which prevents translation), since UseMARCON cannot convert records without a valid initialization file.

## MARC Export and UseMARCON

---

This section discusses how UseMARCON and the MARC export program interact.

### Overview

---

UseMARCON interacts with MARC Export by using translation rules to convert the exported MARC records from MARC21 to the specified MARC format.

Using the command line option **-y** allows you to specify the location of a UseMARCON initialization file which points to the translation files needed to translate the records into the specified MARC format.

To use UseMARCON to convert a batch of MARC21 format records to another MARC format when using MARC Export, add the **-y** switch to the **pmarcexport** command, followed by the path and filename of the UseMARCON initialization file you wish to use for the conversion. The initialization file directs MARC Export to the directory and files needed by UseMARCON to perform the conversion. The following example shows a sample MARC Export command line with a UseMARCON activation switch.

### MARC Export Example

---

```
pmarcexport -rA -mU -t1997-10-25:1998-10-27 -y/m1/xxxdb/ini/
usemarcon/US2UNI/us2uni.ini
```



#### IMPORTANT:

*UseMARCON translations only work on these record types:*

- B (Bibliographic records)
- A (Authority records)

Therefore the required **-r** switch in the MARC Export command line can only be followed by the preceding record types when using UseMARCON for translations with MARC export. For more information on the MARC Export record type switch, see [Bulk Export of MARC Records](#) on page 10-1.

## MARC Export Error Reporting

---

When UseMARCON encounters errors during MARC Export batch jobs, it generates error messages to the log file for that session.

Two possible error messages can occur in the MARC Export log for each session.

- For *each* record not translated, a *record not translated* message is sent to the log. Check the log file for the session to determine whether the error causing the *record not translated* message prevented translation completely or just affected some records.
- If the UseMARCON initialization file is missing, in the wrong place, or not configured properly, an *ini path not valid* message is sent to the log file. This message indicates a fatal error (which prevents translation), since UseMARCON cannot convert records without a valid initialization file.

## Z39.50 Server and UseMARCON Interaction

---

This section discusses how UseMARCON and the Z39.50 Server interact

### Overview

---

UseMARCON interacts with Z39.50 servers enabling remote Z39.50 clients which only request MARC records in formats other than MARC21, to search your Voyager databases.

UseMARCON converts the records to the MARC format requested by the Z39.50 client before sending those records.

When a Z39.50 client sends a request for non-MARC21 records, your Voyager Z39.50 server locates the MARC conversion profile (specified in the `z3950svr.ini` file) corresponding to the requested format (if one exists). This conversion profile points to the UseMARCON initialization file containing the required conversion files. UseMARCON then converts the requested records to the specified format and then sends them to the Z39.50 client in the requested format.

### The `z3950svr.ini` File

---

To use UseMARCON with your Z39.50 server(s), you must first specify the UseMARCON initialization file paths for each conversion in the `z3950svr.ini` file.



## Procedure B-1. Setting Up Initialization Paths in z3950srv.ini File

Use the following to specify the initialization paths.

1. Add a stanza to the `z3950srv.ini` file entitled `[usemarcon translations]` (if one does not already exist).
2. Add a line for each MARC format supported by your library in the `[usemarcon translations]` stanza in the `z3950srv.ini` file, using the following format:

`Marcformat=/path/usemarconinifile.name`

3. Enter the `Marcformat` portions of the lines as they are formatted in Table B-3 (the table includes the Z39.50 format codes of the formats in case you need them).

**Table B-3. MARC Formats and Corresponding Z39.50 Format Codes**

MARC Format	Z39.50 Format Code
Unimarc	1.2.840.10003.5.1
Intermarc	1.2.840.10003.5.2
Ukmarc	1.2.840.10003.5.11
Normarc	1.2.840.10003.5.12
Librismarc	1.2.840.10003.5.13
Danmarc	1.2.840.10003.5.14
Finmarc	1.2.840.10003.5.15
Canmarc	1.2.840.10003.5.17
Picamarc	1.2.840.10003.5.19
AMARC21	1.2.840.10003.5.20
Ibermarc	1.2.840.10003.5.21

4. Set the path to the initialization file that converts MARC21 to the specified non-MARC21 format (for Z39.50 server processes, the records are converted from your Voyager, MARC21 records to the format specified by the Z39.50 client).

Do this for *each* non-MARC21 format supported by your library.

**IMPORTANT:**

*Only one non-MARC21 format can be requested by the Z39.50 client per session. If you request another MARC format by way of the Z39.50 client in a session after choosing the first one, the server and client receive an error message.*

For a sample [usemarcon translations] stanza in the z3950svr.ini, see [B-15](#).

---

**Sample [usemarcon translations] Stanza in  
z3950svr.ini File (Lines 1-3)**

Figure B-7 shows a sample [usemarcon translations] stanza in the z3950svr.ini file containing UseMARCON initialization file paths for two MARC formats.

(Line #)

```
1      # Translations
2      Unimarc=/m1/voyager/xxxdb/ini/usemarcon/US2UNI/
          us2.ini
3      Finmarc=/m1/voyager/xxxdb/ini/usemarcon/US2FIN/
          us2.ini
4
5      # STOP! !
6      #
7      # That should be all you have to edit in this
          file. Make sure you have your
8      # sapi.ini file configured properly.
```

---

Figure B-7. Sample [usemarcon translations] Stanza

---

**UseMARCON/Z39.50 Error Reporting**

When UseMARCON encounters errors during Z39.50 server sessions, it generates error messages specific to UseMARCON translations in the UseMARCON log file (default= usemarcon.z39.date.time) for that session. It also sends error messages, detailing how the UseMARCON translation error affected the Z39.50 process, to the normal Z39.50 log (default= z3950svr.log).

The possible error messages in the Z39.50 log for each session are:

- For each record not translated, a record not translated message gets sent to the Z39.50 log. Check the UseMARCON log file for the session to determine whether the error causing the record not translated message prevented translation completely or just affected some records.
- If the UseMARCON initialization file is missing, in the wrong place, or not configured properly, an ini path not valid message is sent to the Z39.50 log file.
- If the client requests a MARC format which has no entry, a No USEMARCON ini entry for this OID: oid error message is generated.
- If the client requests more than one non-MARC21 format in the same session, an Only one type of USEMARCON conversion allowed per session message is generated.

A fatal error (an error which prevents UseMARCON from converting the record) sends a message stating that there was a translation error, to the UseMARCON log, and an error message stating that the record was not translated by UseMARCON to the normal Z39.50 error log (`z3950svr.log`, by default).



### **IMPORTANT:**

*In the case of a fatal error, the Z39.50 client displays a general “Failed to retrieve record” message. In the case of non-fatal errors, the client receives no messages detailing the errors in the conversion. Because of this, troubleshooting errors in the initial phases of setting up UseMARCON for use with Z39.50 servers is very important.*

## **UseMARCON Error Numbers**

---

The following sections detail the UseMARCON error messages.

### **General Error Numbering**

---

Errors are numbered 000-9999. The first digit describes the part of the process in which the error appears. Table B-4 shows the general error categories and corresponding error number ranges.

**Table B-4. General UseMARCON Error Message Guidelines**

Error #	Error
0xxx	The error is due to improper interface use.

**Table B-4. General UseMARCON Error Message Guidelines**

Error #	Error
1xxx	The error appears during reading of a MARC file, that is, a format problem.
2xxx	The error occurs during MARC checking in input (in comparison with the input checking table).
3xxx	The error occurs during character translation.
4xxx	The error occurs during coded data translation.
5xxx	The error occurs during rule analysis or conversion.
7xxx	The error occurs during checking of the MARC output.
8xxx	The error occurs during writing of the MARC file.
9xxx	Other internal errors.

**Error Message Numbers**

A complete list of error messages used by the USEMARCON software is shown in Table B-5.

**Table B-5. UseMARCON Error Messages (Page 1 of 7)**

Error #	Error
501	Invalid rule pattern to search
502	No more patterns found
503	Maximum errors to be encountered limit has been reached
504	The last record to be converted has been reached
505	Type or browse a MARC file to open before
506	Pattern not found
507	Unable to evaluate an empty rule
508	No defined box to search. Please point to the box to search in
509	No search is available on this selection. Please select another one
510	Comment is too long and will be truncated

**Table B-5. UseMARCON Error Messages (Page 2 of 7)**

Error #	Error
511	Line is too long and will be truncated
512	Unknown input format comment
513	Unknown output format comment
514	Please select RI boundaries
515	Please select No boundaries
516	Invalid RI boundaries
517	Invalid No boundaries
590	Specified file does not exist (please check path )
1001	Unable to reset a writing mode opened file
1002	Invalid SCW encountered when attempting to read a MARC notice
1003	Unable to go further in MARC file reading
1004	Invalid length encountered when attempting to read a MARC notice
1005	Unable to go further in MARC file writing
1006	Unable to flush the MARC output file
1007	Invalid MARC data location address
1009	Unable to set the content of the field
1101	Invalid MARC tag
1102	Invalid MARC indicators
1202	Unable to set the label
1501	MARC buffer allocation failure
1502	Error encountered when attempting to read the MARC file
2001	Invalid input format checking rule ( item expected )
2002	Invalid or absent tag in input format checking rule
2003	Format checking rule redundancy in input format checking file
2004	Invalid or absent first indicators list in input format checking rule
2005	Invalid or absent second indicators list in input format checking rule
2006	Invalid or absent subfield in input format checking rule
2101	Not repeatable but redundant field found in input record
2102	Invalid first indicator found in input record

**Table B-5. UseMARCON Error Messages (Page 3 of 7)**

Error #	Error
2103	Invalid second indicator found in input record
2104	Invalid or redundant subfield found in input record
2105	Unexpected field found in input record
2106	Mandatory field expected in input record
2107	Mandatory subfield expected in input record
2108	Field without any subfield found in input record
2501	TControlField allocation failure when attempting to load a new input format checking rule
2502	TCtrlSubfield allocation failure when attempting to load a new input format checking rule
2503	First indicators list allocation failure when attempting to load a new input format checking rule
2504	Second indicators list allocation failure when attempting to load a new input format checking rule
3000	Memory allocation error
3001	Character not transcoded (unable to find it in transco table)
4001	Coded data not loaded
5000	Memory allocation error
5001	The selected rule file does not exist
5002	Unable to load the invalid rule
5003	Unable to find the label in any CD
5004	Only one indicator has been found
5005	Content of indicator is too long (>1)
5100	Rule analysis error
5101	A CDOOut like TTT(no)... has an invalid subfield occurrence number (no, nso or nto)
5102	A CDOOut like ...SS(no) has an invalid field occurrence number (no, nso or nto)
5103	A CDOOut like TTT(nto)... has an invalid subfield occurrence number (no, nso or nto)
5104	A CDOOut like ...SS(nso) has an invalid field occurrence number (no,nso or nto)
5200	Expected CD tag

**Table B-5. UseMARCON Error Messages (Page 4 of 7)**

Error #	Error
5201	Invalid CD tag ( three characters are required )
5202	Invalid CD tag ( only numerics or letters are allowed )
5203	Invalid CD subfield ( only two characters required )
5204	Invalid CD subfield ( only '1' or '2' is expected behind a 'I' subfield )
5205	Invalid CD subfield ( only numerics or letters are allowed behind a '\$' subfield )
5206	Invalid CD subfield ( only I1, I2 or \$? are allowed )
5207	Invalid position settings
5208	Invalid CD tag occurrence number
5209	Invalid CD subfield occurrence number
5210	Misplaced rule : please insert this rule before the previous one
5211	Misplaced rule : please insert this rule after the next one
5212	Invalid character found in rule
5301	Invalid output occurrence number 'no'
5302	Invalid output tag occurrence number 'nto'
5303	Invalid output sub occurrence number 'nso'
5304	Invalid input occurrence number 'n'
5305	Invalid input tag occurrence number 'nt'
5306	Invalid input sub occurrence number 'ns'
5307	Unknown main input CD ( please type or load it before evaluating )
5308	Unknown old output CD ( please type it before evaluating )
5309	Unknown other input CD ( please type it before evaluating )
5501	TRule allocation failure when attempting to analyse the rule
5502	Unable to allocate space for setting text of analysed rule
5503	Unable to allocate space for setting comment of analysed rule
5504	TCD allocation failure when attempting to analyse the rule
5505	TCDLib allocation failure when attempting to deal with other input CDs
5506	Buffer allocation failure when attempting to split the rule
5507	TCD allocation failure when attempting to load CD from MARC record
7001	Invalid output format checking rule ( item expected )

**Table B-5. UseMARCON Error Messages (Page 5 of 7)**

Error #	Error
7002	Invalid or absent tag in output format checking rule
7003	Format checking rule redundancy in output format checking file
7004	Invalid or absent first indicators list in output format checking rule
7005	Invalid or absent second indicators list in output format checking rule
7006	Invalid or absent subfield in output format checking rule
7101	Redundant field (not repeatable) found in output record
7102	Invalid first indicator found in output record
7103	Invalid second indicator found in output record
7104	Invalid or redundant subfield found in output record
7105	Unexpected field found in output record
7106	Mandatory field expected in output record
7107	Mandatory subfield expected in output record
7108	Field without any subfield found in output record
7501	TControlField allocation failure when attempting to load a new output format checking rule
7502	TCtrlSubfield allocation failure when attempting to load a new output format checking rule
7503	First indicators list allocation failure when attempting to load a new output format checking rule
504	Second indicators list allocation failure when attempting to load a new output format checking rule
8001	Unable to delete the Error Log File
9001	TRuleFile allocation failure when attempting to load the Rule File
9011	TCheckFile allocation failure when attempting to load the Input Check File
9012	TCheckFile allocation failure when attempting to load the Output Check File
9013	TTTransFile allocation failure when attempting to load the Translation Character Table
9021	TMARCFfile allocation failure when attempting to open the Input MARC File
9022	TMARCFfile allocation failure when attempting to open the Output MARC File
9031	TMARCRRecord allocation failure when attempting to load the Input MARC File

**Table B-5. UseMARCON Error Messages (Page 6 of 7)**

Error #	Error
9032	TMARCRecord allocation failure when attempting to load the Output MARC File
9041	TMARCFIELD allocation failure when attempting to load the notice into memory fields
9101	TRuleDoc not created
9102	TMARCDoc not created
9103	TDummyDoc not created
9104	TTransDoc not created
9105	TCheckDoc not created
9201	TCD allocation failure when attempting to search for another CD
9202	Label is mandatory and has not been converted
9203	TMARCFIELD allocation failure when attempting to merge CDs into fields
9301	TCDLib allocation failure when attempting to split a field into CDs
9401	Find SLIST allocation failure when attempting to memorize precedent find/replace request
9402	Replace SLIST allocation failure when attempting to memorize precedent find/replace request
9403	TRule allocation failure when attempting to search/replace items
9404	TCD allocation failure when attempting to search/replace items
9501	Unable to open the ASCII mode file
9502	Unable to open the binary mode file
9503	Unable to delete the file
9504	Unable to get the next line of a binary file
9505	Invalid #include 'file' directive found in file
9506	Unable to read two first lines of an ASCII file
9601	Unable to open the MARC Input Window
9602	Unable to open the MARC Output Window
9603	Unable to open the Rule Edit Window
9604	Unable to open the Rule Eval. Window
9700	Two identical CDs found in a record !
9701	Unable to save the MARC edit configuration file

**Table B-5. UseMARCON Error Messages (Page 7 of 7)**

Error #	Error
9703	Invalid tag field to add to the list of tags without indicator
9704	Unable to add the selected tag to the list of tags without indicator
9705	Unable to remove the selected tag from the list of tags without indicator
9706	(No) is not filled
9800	Unable to open the help file usemarco.csc/hlp
9999	Unknown error



---

## **WebVoyáge Patron Authentication Adapter Feature**

---

C

---

### **Overview**

---

The WebVoyáge Patron Authentication Adapter feature makes WebVoyáge compatible with an external patron authentication program. This feature enables libraries to develop customized security solutions for WebVoyáge. To allow WebVoyáge to communicate with the external patron authentication programs, libraries must develop a patron authentication adapter.

Once the adapter is created, you can use the external authentication system in lieu of the normal WebVoyáge patron login page or give patrons the option of using either login page.

This document details the process that occurs between WebVoyáge, Voyager, and the patron authentication adapter. It also includes the relevant WebVoyáge configuration and behavior. Use this information to develop your patron authentication adapter suited to your third party authentication program.

The Voyager patron login functionality that accommodates the external authentication is designed to be used with any external authentication program. No particular type of authentication system is assumed, and the system is only discussed in terms of its interaction with Voyager and WebVoyáge.

Some benefits of the WebVoyáge interaction design are as follows:

- You can create the adapter in any programming language.
- For added security, patron information is not communicated through the network.

- If you have a pre-established patron ID value in your external database and are not using Institution IDs, you can populate the Institution ID fields of your Voyager Patron records with your external database patron ID's.
- WebVoyage gives control of the browser to the authentication adapter/system so it may collect patron credentials as needed.

**NOTE:**

Throughout this document, the term “WebVoyage Patron Authentication Adapter feature” refers to the WebVoyage functionality that allows WebVoyage to communicate with an external authentication program, via a customer-developed authentication adapter. The term “patron authentication adapter” is used to refer to the customer-developed adapter which provides the communication bridge between WebVoyage and the external authentication program.



---

**Procedure C-1. Patron Authentication Workflow Overview**

---

The following steps outline the patron authentication process between WebVoyage and the patron authentication adapter, at a very general level. For a more detailed description of the interaction between WebVoyage and the patron adapter, see [Detailed WebVoyage Patron Adapter Interaction](#) on [page C-3](#).

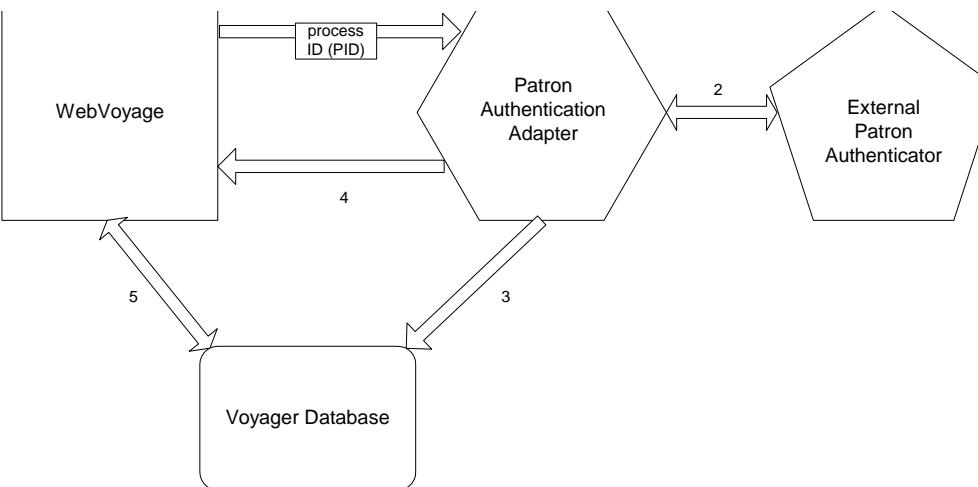
[Figure C-1](#) illustrates the interaction between WebVoyage and the patron authentication adapter, at an overview level.

When a patron performs a WebVoyage function requiring login:

1. WebVoyage is redirected to the external adapter.
2. The patron is authenticated using the third party authentication system.
3. The patron authentication adapter inserts information into the Voyager database.
4. The patron authentication adapter sends a URL to WebVoyage indicating the authentication status.

If the adapter returns a positive authentication status, WebVoyage uses the information placed into the Voyager database by the adapter, to locate the corresponding Voyager patron.

5. (Not shown in diagram) WebVoyage logs in that patron.

**Figure C-1. Patron Authentication Adapter Flow Diagram****NOTE:**

Voyager patron records must exist in the Voyager database before they can be retrieved using an external authentication program in conjunction with an adapter.

Patron records in the Voyager database must be uniquely identified by information in the Institution ID field. The information in the Institution ID field in the Voyager patron records in the Voyager database must also exist in the external patron authenticator (or database) and the adapter must be able to match this information in both patron record sources (the voyager patron records and external records).

## **Detailed WebVoyage Patron Adapter Interaction**

This section includes a more detailed description of the WebVoyage/adapter interaction process (once the adapter behavior is enabled in WebVoyage, and the adapter is created).

1. A WebVoyage user performs a WebVoyage function requiring log in.
2. The login page displays.

---

If	Then
Bypassing the Patron Login page	<p>a. WebVoyage sends a form to the URL specified in option.extAuthSystemURL= including the process ID (PID) of the WebVoyage session.</p> <p>The form information is submitted as a GET.</p> <p>b. The external authentication page opens.</p>
If	Then
Accessing the external authentication system via a button on the WebVoyage Patron Login page	<p>a. The WebVoyage Patron Login page opens, with a redirect button to the external authentication program.</p> <p>b. When you click the button, WebVoyage sends a form to the URL specified in the option.extAuthSystemURL= including the process ID (PID) of the WebVoyage session.</p>
3. WebVoyage gives control of the browser to the adapter for purposes of patron authentication.	
4. The patron is authenticated externally.	
If...	Then...
authentication fails	<p>a. The adapter includes a failure status in the URL redirect to WebVoyage in the following format:</p> <pre>http://&lt;host&gt;:&lt;port&gt;/vwebv/ externalLogin.do?&lt;redirect string&gt;&amp;authenticate=N</pre> <p>For more on this URL format, see <a href="#">Format of the Redirect URL from the Authentication Adapter to WebVoyage</a> on page C-8.</p> <p>b. WebVoyage provides the patron with another opportunity to log in, using whichever login page was originally provided (WebVoyage with the redirect button, or the external authentication page itself).</p> <p>c. Return to step 2 in this process until successfully authenticated.</p>

If...	Then...
authentication succeeds	continue to step <a href="#">5</a>
5. The adapter inserts a record into the WOPAC_PID_PATRON_KEYS table on the Voyager server that contains 2 fields:	<ul style="list-style-type: none"> <li>PID: Contains the encrypted WebVoyáge process ID obtained from the WebVoyáge redirect URL (step <a href="#">2</a> in this process) . When storing the redirect URL into the PID field of the WOPAC_PID_PATRON_KEYS table, the encryption and escape codes must be retained.</li> <li>PATRON_KEY: Contains a value unique to the patron, obtained from the external authentication. This value must correspond to the value in the Voyager patron's Institution ID information field, in the Voyager patron record (though the actual string in the Institution ID field need not actually be an institution ID per se).</li> </ul>
6. The adapter sends a URL back to the WebVoyáge session which does the following:	<ol style="list-style-type: none"> <li>Indicates to WebVoyáge an authentication success status.</li> <li>Returns control of the browser to WebVoyáge. This URL takes the following form: <code>http://&lt;host&gt;:&lt;port&gt;/vwebv/externalLogin.do?&lt;redirect string&gt;&amp;authenticate=Y</code></li> </ol>
7. WebVoyáge looks in the PATRON_KEY field of the Voyager database for the unique patron value added by the adapter.	
8. WebVoyáge looks for a Voyager patron containing the value of the PATRON_KEY (from step <a href="#">5</a> ) in its Institution ID field in the Voyager patron record.	

---

If...	Then...
WebVoyáge does not find a patron containing the PATRON_KEY information in his or her voyager patron record	<p>a. The adapter includes a failure status in the URL redirect to WebVoyáge in the following format:</p> <pre>http://&lt;host&gt;:&lt;port&gt;/vwebv/externalLogin.do?&lt;redirect string&gt;&amp;authenticate=N</pre> <p>For more on this URL format, see <a href="#">Format of the Redirect URL from the Authentication Adapter to WebVoyáge</a> on page C-8.</p> <p>b. WebVoyáge provides the patron with another opportunity to log in, using whichever login page was originally provided (WebVoyáge with the redirect button, or the external authentication page itself).</p> <p>c. Return to step 2 in this process until authentication success is attained and a patron match is found.</p>
WebVoyáge finds a patron containing the PATRON_KEY information in the voyager patron record	continue to step 9

9. WebVoyáge logs the corresponding patron in to the WebVoyáge session with the specified process ID.

---

## WebVoyáge Configuration

---

This section details the WebVoyáge configuration required to use the Patron Authentication Adapter feature.

The `webvoyage.properties` file controls the WebVoyáge behavior options for the Patron Authentication Adapter.

- [Enabling and Disabling Webvoyáge Adapter Behavior](#)
- [Enabling and Disabling the WebVoyáge Patron Login Bypass](#)

- [Setting the URL to the External Authentication System](#)
- [Setting the Text of the External Authentication Link Button](#) (used if *not* bypassing WebVoyage Patron Login page)

See [Figure C-2](#) for an example of the `webvoyage.properties` file settings.

```
#=====
# Should WebVoyage users be able to use an external authentication system when logging in?
# If Y, WebVoyage uses the external authentication system as configured below
# If N, WebVoyage displays the logon form as previously configured
#=====
option.extAuthSystemEnabled=Y
#=====
# URL to the external authentication system
#=====
option.extAuthSystemURL=
#=====
# Should WebVoyage bypass the logon form if using an external authentication system?
#=====
option.extAuthBypassLoginScreen=N
...
# use of the external authentication link is optional,
# this line will have no effect if option.extAuthSystemEnabled=N
#=====
page.logIn.extAuth.linkText=Go to External Patron Login System
```

---

**Figure C-2. `webvoyage.properties` example**

### **Enabling and Disabling Webvoyage Adapter Behavior**

The `option.extAuthSystemEnabled=` variable enables or disables the WebVoyage patron authentication adapter behavior.

- To enable the WebVoyage patron authentication adapter behavior, set the `option.extAuthSystemEnabled=` variable to `Y`.
- To disable the behavior, set the variable to `N`.

If the variable is set to `N`, WebVoyage only displays the standard WebVoyage Patron Login page for login functions.

## Setting the URL to the External Authentication System

Specify the URL to the external authentication program in the `option.extAuthSystemURL=` variable.

## Enabling and Disabling the WebVoyage Patron Login Bypass

The `option.extAuthBypassLoginScreen=` variable determines whether WebVoyage displays the normal WebVoyage Patron Login page or bypasses it and goes straight to the external authentication page.

- To bypass the WebVoyage Patron Login and display the external authentication system when performing WebVoyage functions requiring login, set the `option.extAuthBypassLoginScreen=` variable to `Y`.
- To display the WebVoyage Patron Login page, along with a button connecting to the external authentication system, set the `option.extAuthBypassLoginScreen=` variable to `N`.

If you set this variable to `N`, configure the `page.logIn.linkText` variable (see [Setting the Text of the External Authentication Link Button](#)).

## Setting the Text of the External Authentication Link Button

If you do not bypass the WebVoyage Patron Login page, a link displays on the login page that links to the external authentication system. The text of the link is set in the `page.logIn.linkText` variable.

## Format of the Redirect URL from the Authentication Adapter to WebVoyage

The format for the redirect URL that the Authentication Adapter must use to return control to WebVoyage combines fields in a conventional WebVoyage session URL with the authentication result field:

```
http://<host>:<port>/vwebv/externalLogin.do?<redirect  
string>&authenticate=<status>
```

[Table C-1](#) details the URL field components.

**Table C-1. URL field Components**

Component	Description
<host>	IP or domain name of the host where WebVoyage is running.
<port>	TCP port of the WebVoyage process.
<redirect string>	A string, such as <code>searchBasic</code> , consisting of form fields that the adapter must use to redirect back to the instance of WebVoyage that invoked it.
<status>	Adapter must set this field to: <ul style="list-style-type: none"> <li>• Y if patron authentication succeeded.</li> <li>• N if patron authentication failed or cancelled.</li> </ul>

When control is returned to WebVoyage, it will check the authenticate field in the redirect URL to determine the patron status. If authentication was successful, i.e. the authenticate field was set to Y, WebVoyage sends its process ID in a patron login request to the OPAC server. This causes the server to retrieve the unique patron key, corresponding to the encrypted WebVoyage process ID, from the WOPAC\_PID\_PATRON\_KEYS table in the Voyager database.

## Example

This section includes an example process, using a WebVoyage session which does not bypass the WebVoyage Patron Login page. The external authentication program in the example uses an LDAP database as its patron database, with the attribute of `empid` (employee ID) acting as the LDAP match-point to the Institution ID in the Voyager patron records in the Voyager database. The `empid` values from the LDAP directory have also been added to the Institution ID field of the corresponding patron records (to serve as match-points between the two databases).

1. A WebVoyage user performs a WebVoyage function requiring log in.
2. The login page displays including the redirect link.
3. Patron clicks the **Go to External Patron Login System** link.
4. The external authentication program displays a login page.

5. The adapter stores the encrypted process ID (PID) from the WebVoyage form submitted, retaining escape codes and encryption.
6. The user logs in to the external authentication program.
7. The external authentication program verifies the patron against the LDAP directory.

If authentication fails, the adapter returns the failure status in the URL `http://<host>:<port>/vwebv/externalLogin.do?<redirect string>&authenticate=N`. WebVoyage then returns to step [2](#).

8. If authentication succeeds, the patron authentication adapter retrieves the empid from the patron's information in the LDAP directory.
9. The adapter inserts the encrypted process ID (PID) stored in step [5](#) into the PID field of the WOPAC\_PID\_PATRON\_KEYS table in the Voyager database.
10. The adapter inserts the empid value into the PATRON\_KEY field in the WOPAC\_PID\_PATRON\_KEYS table in the Voyager database.
11. The adapter sends a URL to the web server from which it received the request, in the following format

```
http://<host>:<port>/vwebv/externalLogin.do?<redirect string>&authenticate=Y
```

12. When WebVoyage receives this URL, it retrieves the PATRON\_KEY from the Voyager database for that PID.
  13. WebVoyage searches for a voyager patron record that contains the PATRON\_KEY value in its Institution ID field.

If WebVoyage does not find a corresponding patron record, it returns to step [2](#).
  14. If it WebVoyage finds a corresponding patron record, it logs the Voyager patron into the WebVoyage session with the encrypted process ID in the PID field of the WOPAC\_PID\_PATRON\_KEYS table.
-

# Index

---

## Numerics

---

[008] stanza, [11-17](#)  
 035 field  
     stripping a non-standard 035 field, [11-10](#)  
 856 link failure report suffix segment  
     cataloging report, [24-8](#)

---

## A

About this document, [xlv](#)  
 Accept these records button, [30-16](#)  
 Access database file  
     Reports.mdb, [4-50](#)  
 accrued fines and demerits, [8-6](#)  
     running circulation batch job, [8-7](#)  
 Acqbib.cfg file, [4-51](#)  
 acqjob 5, [16-1](#)  
 acquisitions  
     batch job  
         fix exchange rates, [16-1](#)  
         running batch job 5, [16-3](#)  
 acquisitions notice  
     base segment, [22-4](#)  
     cancel serial claim notice suffix segment, [22-8](#)  
     cancellation notice suffix segment, [22-6](#)  
     order claim notice suffix segment, [22-7](#)  
     return notice suffix segment, [22-6](#)  
     serial claim notice suffix segment, [22-7](#)  
     standard interface file (SIF), [22-1](#)  
     suffix segment, [22-5](#)  
     voucher/check request suffix segment, [22-8](#)  
 acquisitions report  
     base segment, [23-4](#)  
     fund snapshot after rollover report suffix segment,  
         [23-12](#)  
     fund snapshot before rollover report suffix seg-  
         ment, [23-12](#)  
     fund snapshot report suffix segment, [23-10](#)  
     global open orders report suffix segment, [23-9](#)  
     open orders after rollover report suffix segment,

[23-12](#)

open orders before rollover report suffix segment,  
     [23-12](#)

open orders report suffix segment, [23-8](#)

purchase order report suffix segment, [23-5](#)

standard interface file (SIF), [23-1](#)

suffix segment, [23-4](#)

Acquisitions Reports and Notices page, [30-9](#)

acquisitions utilities

    WebAdmin, [30-9](#)

Add Bib/Item dialog box, [4-51](#)

adding

    clients after installation, [4-13](#)

adding a new login/password

    WebAdmin, [30-4](#)

AIX server, [3-4](#)

starting

    Apache, [3-10](#)

    Oracle services, [3-6](#)

    Voyager, [3-8](#)

stopping

    Apache, [3-10](#)

    Oracle services, [3-6](#)

    Voyager, [3-8](#)

archive and expire call slip requests, [8-4](#)

    running circulation batch job, [8-4](#)

archive short loans, [8-5](#)

    running circulation batch job, [8-5](#)

audience

    of this document, [xlv](#)

audit file

    bursar transfer program, [7-9](#)

    marcexport program, [10-10](#)

    patron extract program, [5-5](#)

    patron update program, [6-7](#)

Authblob\_vw, [29-2](#)

Authheading\_vw, [29-3](#)

Authhistory\_vw, [29-3](#)

authority data

    loading, [2-2](#)

Authority record has a status of D, S, or X written to  
     reject.imp file message, [12-15](#)

Authority records, [B-12](#)

Authority1xx4xx\_vw, [29-4](#)

Authority5xx1xx\_vw, [29-4](#)

Authoritydupe\_vw, [29-5](#)

automating backup

    Windows 2000 server, [3-25](#)

**B**

backup  
 cron, 3-11  
 determining device driver, 3-12  
 identifying data filesystem/volume, 3-14  
 restoring, 3-27  
 Windows 2000 server, 3-21  
 automating the backup, 3-25  
 customizing backup, 3-24

Backup tab, 3-25

barcode lookup, 17-1  
 configuration file, 17-2  
 schema file, 17-8

barcode rules  
 bulk import program, 12-13

base segment  
 acquisitions notice, 22-4  
 acquisitions report, 23-4  
 cataloging report, 24-4  
 circulation notice, 25-4  
 circulation report, 26-4  
 media scheduling notice, 27-3  
 media scheduling report, 28-3

batch forgive jobs  
 audit report, 8-29, 8-33, 8-36  
 error logging, 8-29, 8-32, 8-35  
 error report, 8-29, 8-32, 8-36

batch job  
 acquisitions  
 fix exchange rates, 16-1  
 cataloging  
 process GHC queue-step 1, 13-4  
 process GHC queue-step 2, 13-5  
 process GHC queue-step 3, 13-6

circulation  
 accrued fines and demerits, 8-6  
 archive and expire call slip requests, 8-4  
 archive short loans, 8-5  
 export OPAC requests, 8-4  
 forgive demerits, 8-16  
 forgive fines by create date, 8-30  
 forgive fines by patron group and expire date, 8-33  
 forgive fines by patron ID, 8-27  
 patron purge, 8-23  
 patron suspension, 8-7  
 place items on active course reserve list, 8-10  
 place recalls and holds for items on active

course reserve list, 8-12  
 purge UB patron stub record, 8-5, 8-8  
 retain patron IDs, 8-17  
 takes items on inactive course reserve lists off reserve, 8-14  
 update remote circulation cluster cache, 8-9  
 update shelving status, 8-3

media scheduling  
 retain patron IDs, 9-1  
 running acquisitions batch job 5, 16-3

Begin Record field, 30-15

Bib Usage Log Export  
 see OPAC Bib Usage Log Export program

Bib\_usage\_log table, 15-3

Bib\_vw, 29-5

Bibblob\_vw, 29-7

Bibhistory\_vw, 29-7

bibliographic data  
 loading, 2-2

bibliographic records, B-11, B-12

bibliographic usage logging  
 configuring, 15-5

Bibloc\_vw, 29-8

Bi-Weekly variable, 15-12

Bmarcfix.cfg file, 4-50

booking charge statistics report suffix segment  
 media scheduling report, 28-5

booking exceptions report suffix segment  
 media scheduling report, 28-5

booking statistics report suffix segment  
 media scheduling report, 28-4

building the 035, 12-11

bulk export of MARC records (marcexport) program, 10-1

bulk import, B-11  
 character set mapping, 12-4  
 converting records to Unicode, 12-4  
 leader byte 9 value, 12-4  
 UseMARCON, B-10  
 batch jobs, B-11  
 error reporting with, B-11  
 sample script with UseMARCON switch, B-11

bulk import program, 12-1  
 barcode rules, 12-13  
 building the 035, 12-11  
 call number hierarchy, 12-12  
 creating holdings and item records, 12-12  
 delete file, 12-14  
 disabling generation of keyword index files, 12-6  
 discard file, 12-14  
 error and log file messages, 12-15

---

error file, 12-15  
 input file, 12-11  
 item type, 12-13  
 log file, 12-13  
 mapping, 12-13  
 message  
     No item option in use - No item records created, 12-17  
     No Matches found for Input Cancel Record. Import Record Discarded!, 12-17  
     Non-Bibliographic Record detected! Holdings record written to error file, 12-17  
 messages  
     Authority record has a status of D, S, or X written to reject.imp file, 12-15  
     DB Bib Record cancelled/deleted by Import Record!, 12-16  
     Delete Failed - Items Attached, 12-16  
     Delete Failed - MFHDs Attached, 12-16  
     Delete Failure, 12-16  
     Duplicate detected. Discarding dupe import record!, 12-16  
     Duplicate detection failure!, 12-16  
     Duplicates above replace threshold. Adding anyway, 12-16  
     Duplicates above thresholds. Cannot resolve!, 12-16  
     Existing database record replaced by imported, 12-16  
     Failed to load converter, 12-17  
     Record does not match format for " change your import rule, 12-17  
     Record nnn discarded -- no matching records in the database, 12-17  
     Record Parse Failure, 12-17  
     Record Retrieval Failure, 12-17  
     Unparseable record written to error file, 12-17  
 parameters  
     -a MFHD location code, 12-8  
     -b begin record, 12-8  
     -d delete MFHDs, 12-9  
     -e end record, 12-8  
     -f filename, 12-7  
     -h help, 12-10  
     -i import code, 12-7  
     -k OK to export, 12-10  
     -l location, 12-8  
     -m load MFHDs, 12-8  
     -o operator name, 12-8  
     -x delete bibliographic records, 12-9  
     -X NOKEY, 12-7, 12-10, 30-12  
     reject file, 12-15  
     replace file, 12-15  
     running, 12-10  
     WebAdmin, 30-12  
 bulk importing  
     interleaved file generated by prebulk, 11-4  
 bulkimport command, B-11  
 bursar transfer program, 7-1  
     audit file, 7-9  
     bursar (SIF), 7-8  
     configuration file, 7-5  
     error file, 7-10  
     output file, 7-8  
 parameters  
     -c configuration file, 7-6  
     -d database name, 7-6  
     -h help, 7-7  
     -i interval for transfer, 7-7  
     -m minimum fine/fee amount, 7-6  
     -o operator ID, 7-7  
     -p patron mode, 7-6  
     -q quiet mode, 7-7  
     -t test mode, 7-7  
     -u username and password, 7-6  
     -v version, 7-7  
 refunding, 7-10  
 running, 7-7  
 stub patron records, 7-2  
 transferring itemized patron fines/fees, 7-4  
 transferring total patron balances, 7-4  
 viewing information, 7-10  
 WebAdmin, 30-29  
 button  
     Accept these records, 30-16  
     Post, 7-11  
     Preview, 13-5  
     Send Record To, 4-49  
     Submit Batch Import, 30-16  
     Submit Bulk Export, 30-22  
     Submit Bursar Transfer, 30-31  
     Submit OPAC Job, 30-35

---

## C

call number hierarchy  
     bulk import program, 12-12  
 Callslip.ini file, 4-51  
 [CALLTYPES] stanza, 11-12

cancel serial claim notice suffix segment  
     acquisitions notice, [22-8](#)

cancellation notice suffix segment  
     acquisitions notice, [22-6](#)  
     circulation notice, [25-6](#)

cataloguing batch job  
     process GHC fields-step 3, [13-6](#)  
     process GHC queue-step 1, [13-4](#)  
     process GHC queue-step 2, [13-5](#)

cataloguing report  
     856 link failure report suffix segment, [24-8](#)

base segment, [24-4](#)

duplicate authority records suffix segment, [24-6](#)  
     see references with linked bib records suffix segment, [24-7](#)

see refs authorized in other auth record suffix segment, [24-7](#)  
     see refs without corresponding auth record suffix segment, [24-8](#)

standard interface file (SIF), [24-1](#)

suffix segment, [24-4](#)

unauthorized name headings suffix segment, [24-5](#)

unauthorized name/title headings suffix segment, [24-6](#)

unauthorized subdivision headings suffix segment, [24-6](#)

unauthorized subject headings suffix segment, [24-5](#)  
     unauthorized title headings suffix segment, [24-5](#)

cataloguing utilities  
     WebAdmin, [30-12](#)

catjob 11  
     cataloguing batch job  
         process GHC queue-step 1, [13-4](#)

catjob 12  
     cataloguing batch job  
         process GHC queue-step 2, [13-5](#)

catjob 13  
     cataloguing batch job  
         process GHC queue-step 3, [13-6](#)

catjob.log file, [13-2](#)

changing location codes of item records or MFHDs, [14-2](#)

checking records with prebulk, [11-3](#)

[Circ Locations] stanza, [7-5](#)

Circ.ini file, [4-51](#)

Circbib.cfg file, [4-51](#)

Circcharges\_vw, [29-8](#)

circjob 1, [8-3](#)

circjob 26, [8-4](#)

circjob 27, [8-5](#)

circjob 29, [8-5, 8-8](#)

circjob 30, [8-6](#)

circjob 31, [8-7](#)

circjob 34, [8-10, 8-12](#)  
     output file, [8-11](#)

parameters  
     -e end date, [8-10, 8-12](#)  
     -s start date, [8-10](#)  
     -start date, [8-12](#)

circjob 35  
     output file, [8-13](#)

parameters  
     -D due date, [8-13](#)  
     -e end date, [8-12](#)  
     -i operator id, [8-12](#)  
     -L location code, [8-12](#)  
     -P print location, [8-13](#)  
     -s start date, [8-12](#)

circjob 36, [8-14](#)  
     output file, [8-15](#)

parameters  
     -e end date, [8-15](#)  
     -s start date, [8-15](#)

circjob 37, [8-16](#)  
     parameters  
         -q number of demerits to forgive, [8-16](#)

circjob 38, [8-17](#)  
     parameter  
         -T number of IDs to retain, [8-18](#)

running  
     command line, [8-19](#)  
     interactively, [8-20](#)

circjob 39, [8-23](#)

circjob 40, [8-27](#)

circjob 41, [8-30](#)

circjob 42, [8-33](#)

circjob 43, [8-3](#)

circjob 8, [8-4](#)

circjob.log, [8-1, 8-3, 8-4, 8-5, 8-7, 8-8, 8-9, 8-29, 8-32, 8-35](#)

Circrenewal\_vw, [29-10](#)

CircSC.ini file, [4-51](#)

circulation  
     batch job  
         accrued fines and demerits, [8-6](#)  
         archive and expire call slip requests, [8-4](#)  
         archive short loans, [8-5](#)  
         export OPAC requests, [8-4](#)  
         forgive demerits, [8-16](#)  
         forgive fines by create date, [8-30](#)

forgive fines by patron group and expire date, 8-33  
 forgive fines by patron ID, 8-27  
 patron suspension, 8-7  
 place items on active course reserve list, 8-10  
 place recalls and holds for items on active course reserve list, 8-12  
 purge UB patron stub record, 8-5, 8-8  
 retain patron IDs, 8-17  
 takes items on inactive course reserve lists off reserve, 8-14  
 update remote circulation cluster cache, 8-9  
 update shelving status, 8-3  
 patron purge, 8-23  
 circulation data (transactions)  
 loading, 2-3  
 circulation item exceptions report suffix segment  
 circulation report, 26-7  
 circulation notice  
 base segment, 25-4  
 cancellation notice suffix segment, 25-6  
 courtesy (due) notice suffix segment, 25-9  
 fine/fee notice suffix segment, 25-7  
 item available notice suffix segment, 25-6  
 overdue notice suffix segment, 25-6  
 recall notice suffix segment, 25-7  
 recall-overdue notice suffix segment, 25-7  
 standard interface file (SIF), 25-1  
 statement of fines and fees suffix segment, 25-8  
 suffix segment, 25-5  
 circulation patron exceptions report suffix segment  
 circulation report, 26-7  
 circulation report  
 base segment, 26-4  
 circulation item exceptions report suffix segment, 26-7  
 circulation patron exceptions report suffix segment, 26-7  
 circulation statistics report suffix segment, 26-6  
 circulation transaction exceptions rept suffix segment, 26-7  
 distribution item order list report suffix segment, 26-8  
 global circulation statistics report suffix segment, 26-7, 26-8  
 hold shelf expired report suffix segment, 26-5  
 missing in transit report suffix segment, 26-6  
 reserved items active report suffix segment, 26-5  
 reserved items expired report suffix segment, 26-5  
 standard interface file (SIF), 26-1  
 suffix segment, 26-4  
 circulation reports and notices  
 WebAdmin, 30-24  
 circulation statistics report suffix segment  
 circulation report, 26-6  
 circulation transaction exceptions rept suffix segment  
 circulation report, 26-7  
 clients  
 uninstalling, 4-20  
 command  
 bulkimport, B-11  
 MARC Export, B-12  
 nohup, 5-2, 6-2, 7-3, 10-2, 12-3  
 pbulkimport, 12-10, B-11  
 pbursar, 7-7  
 pmarcexport, 10-9, B-12  
 pprebulk, 11-21  
 pptrnextr, 5-4  
 pptrnupd, 6-6  
 pstrgvfy, 14-10  
 pwebadmin, 30-3  
 command file  
 pstrgvfy program, 14-8  
 comments  
 about this document, I  
 configuration file  
 Acqbib.cfg, 4-51  
 Bmarcfix.cfg, 4-50  
 bursar transfer program, 7-5  
 Circbib.cfg, 4-51  
 Country.cfg, 4-50  
 EditBook.cfg, 4-50  
 EditSerial.cfg, 4-50  
 emailf.cfg, 15-12  
 exceptdates.cfg, 8-7  
 Lang.cfg, 4-50  
 MARC21 Repertoire.cfg, 4-51  
 NewBook.cfg, 4-50  
 NewSerial.cfg, 4-50  
 prebulk program, 11-5  
 promote.cfg, 8-9  
 Special Characters.cfg, 4-51  
 Spinelabel.cfg, 4-51  
 configuring  
 bibliographic usage logging, 15-5  
 OPAC search logging, 15-4  
 conventions used  
 in this document, xlviii  
 converting  
 data, 2-1  
 Country.cfg file, 4-50

course reserve list  
  circjob 35, [8-12](#)  
  circjob 36, [8-14](#)  
  circjob34, [8-10](#)

courtesy (due) notice suffix segment  
  circulation notice, [25-9](#)

Create one summary transfer record per patron radio button, [30-31](#)

Create transfer record for each fine/fee transferred radio button, [30-31](#)

CREATEMFHD variable, [11-8](#)

creating  
  holdings and item records with bulk import, [12-12](#)  
  holdings from interleaved file with prebulk, [11-5](#)  
  interleaved bib-MFHD file with prebulk, [11-4](#)  
  login/password for WebAdmin, [30-3](#)

cron  
  backup, [3-11](#)  
  batch jobs set up at installation, [3-27](#)  
  running OPAC Bib Usage Log Export program as, [15-17](#)  
  running OPAC Search Log Export program as, [15-8](#)  
  running Selective Dissemination of Information as, [15-15](#)

customizing  
  back up for Windows 2000 server, [3-24](#)  
  search results page of Selective Dissemination of Information, [15-12](#)

---

**D**

Daily Backup window, [3-26](#)

Daily variable, [15-11](#)

data  
  converting, [2-1](#)

data filesystem, [3-14](#)

database view, [29-1](#)  
  Authblob\_vw, [29-2](#)  
  Authheading\_vw, [29-3](#)  
  Authhistory\_vw, [29-3](#)  
  Authority1xx4xx\_vw, [29-4](#)  
  Authority5xx1xx\_vw, [29-4](#)  
  Authoritydupe\_vw, [29-5](#)  
  Bib\_vw, [29-5](#)  
  Bibblob\_vw, [29-7](#)  
  Bibhistory\_vw, [29-7](#)  
  Bibloc\_vw, [29-8](#)

Circcharges\_vw, [29-8](#)  
Circrenewal\_vw, [29-10](#)  
Fundledger\_vw, [29-12](#)  
Heading\_vw, [29-14](#)  
Issues\_vw, [29-15](#)  
Item\_vw, [29-16](#)  
LCclass\_vw, [29-18](#)  
Marcbook\_vw, [29-19](#)  
Marccomputer\_vw, [29-19](#)  
Marcmaps\_vw, [29-20](#)  
Marcmusic\_vw, [29-20](#)  
Marcserial\_vw, [29-21](#)  
Marcvisual\_vw, [29-22](#)  
MFHDblob\_vw, [29-22](#)  
MFHDhistory\_vw, [29-23](#)  
NImclass\_vw, [29-23](#)  
Recordcount\_vw, [29-24](#)  
Serials\_vw, [29-24](#)  
Sudoclass\_vw, [29-25](#)  
Vendorinvoice\_vw, [29-26](#)  
Vendororder\_vw, [29-28](#)

date range, [15-7, 15-16](#)  
  format, [15-8, 15-16](#)

Date Range radio button, [30-20](#)

days after fine created, [30-31](#)

days after fine created field, [30-31](#)

DB Bib Record cancelled/deleted by Import Record! message, [12-16](#)

DEFAULTCALLIND variable, [11-9](#)

DEFAULTCALLNO variable, [11-9](#)

Delete Failed - Items Attached message, [12-16](#)

Delete Failed - MFHDs Attached message, [12-16](#)

Delete Failure message, [12-16](#)

delete file  
  bulk import program, [12-14](#)

delete.imp.cyyymmdd.hhmm file, [12-14](#)

deleting  
  clients after installation, [4-13](#)  
  login/password for WebAdmin, [30-4](#)

determining backup device driver, [3-12](#)

dialog box  
  Add Bib/Item, [4-51](#)  
  File Upload, [30-15](#)  
  Run, [4-3, 4-6, 4-14, 4-19](#)  
  Search, [4-29](#)

disabling generation of keyword index files, [12-6](#)

discard file  
  bulk import program, [12-14](#)

discard.imp.cyyymmdd.hhmm file, [12-14](#)

distribution item order list report suffix segment  
  circulation report, [26-8](#)

document summary, xlvi  
 duplicate authority records suffix segment cataloging report, 24-6  
 Duplicate detected. Discarding dupe import record! message, 12-16  
 Duplicate detection failure! message, 12-16  
 Duplicates above replace threshold. Adding anyway message, 12-16  
 Duplicates above thresholds. Cannot resolve! message, 12-16

Error numbers, B-10  
 error type, B-9  
 exceptdates.cfg file, 8-7  
 executable file VoyagerInstall.exe, 4-23  
 Existing database record replaced by imported message, 12-16  
 export OPAC requests, 8-4  
 , 8-4  
 external patron authentication, C-1

---

## E

EDI\_msg.ini file, 4-51  
 EditBook.cfg file, 4-50  
 EditSerial.cfg file, 4-50  
 E-Mail Address field, 30-10  
 E-mail Address field, 30-16, 30-22, 30-23, 30-24, 30-31, 30-35  
 [E-mail stanza, 4-28  
 [Email] stanza, 15-12  
 emailf.cfg file, 15-12  
 encryption, 4-43  
 END non OK message, B-8  
 End Record field, 30-15  
 err.burs.yymmdd.hhmm file, 7-10  
 err.forgive.YYYYMMDD.HHMM, 8-29, 8-32, 8-36  
 err.imp.yyyymmdd.hhmm file, 12-15  
 err.prebulk.YYYYMMDD.HHMM file, 11-23  
 err.pupd.YYYYMMDD.HHMM file, 6-8  
 err.pxtr.YYYYMMDD.HHMM file, 5-6  
 error  
     fatal, B-10, B-16  
 error and log file messages  
     bulk import program, 12-15  
 error classifications, B-8  
 error description, B-10  
 error field indicator, B-10  
 error file  
     bulk import program, 12-15  
     bursar transfer program, 7-10  
     patron extract program, 5-6  
     patron update program, 6-8  
     prebulk program, 11-22  
 error message  
     END non OK, B-8  
     WARNING, B-8  
 ERROR messages, B-8

---

## F

Failed to load converter message, 12-17  
 Failed to retrieve record message, B-16  
 fatal error, B-10, B-16  
 feedback, customer, I  
 field  
     Begin Record, 30-15  
     E-Mail Address, 30-10, 30-16  
     E-mail Address, 30-22, 30-23, 30-24, 30-31, 30-35  
     End Record, 30-15  
     Holdings Location Code, 30-15  
     Import Code, 30-15  
     Local Filename, 30-15  
     Location Code, 2-2, 30-15  
     Operator Name, 30-15  
     Password, 30-7  
     Patron Group Code, 2-3  
     Rule Code, 30-15  
     System Control (003) Identifier, 30-19  
     System Control (040\$d) Identifier, 30-19  
     Transfer fines/fees greater than or equal to, 30-31  
     User Name, 30-6  
 field indicators  
     error, B-10  
 file  
     Access database Reports.mdb, 4-50  
     configuration  
         Acqbib.cfg, 4-51  
         Bmarcfix.cfg, 4-50  
         bursar transfer program, 7-5  
         Circbib.cfg, 4-51  
         Country.cfg, 4-50  
         EditBook.cfg, 4-50

EditSerial.cfg, [4-50](#)  
 emailf.cfg, [15-12](#)  
 exceptdates.cfg, [8-7](#)  
 Lang.cfg, [4-50](#)  
 MARC21 Repertoire.cfg, [4-51](#)  
 NewBook.cfg, [4-50](#)  
 NewSerial.cfg, [4-50](#)  
 prebulk program, [11-5](#)  
 promote.cfg, [8-9](#)  
 Special Characters.cfg, [4-51](#)  
 Spinelabel.cfg, [4-51](#)  
 delete.imp.ccyymmdd.hhmm, [12-14](#)  
 discard.imp.ccyymmdd.hhmm, [12-14](#)  
 err.burs.yymmdd.hhmm, [7-10](#)  
 err.imp.yyyymmdd.hhmm, [12-15](#)  
 err.prebulk.YYYYMMDD.HHMM, [11-23](#)  
 err.pupd.YYYYMMDD.HHMM, [6-8](#)  
 err.pxtr.YYYYMMDD.HHMM, [5-6](#)  
 executable  
     VoyagerInstall.exe, [4-23](#)  
 httpd2.conf, [30-5](#)  
 index.html, [30-7](#)  
 initialization  
     Callslip.ini, [4-51](#)  
     Circ.ini, [4-51](#)  
     CircSC.ini, [4-51](#)  
     EDI\_msg.ini, [4-51](#)  
     Limits.ini, [4-51](#)  
     Mediahelp.ini, [4-51](#)  
     opac.ini, [15-5](#)  
     sdiemail.ini, [15-9, 15-12](#)  
     Template.ini, [4-50](#)  
     UseMARCON, [B-3](#)  
     voyager.ini, [4-13, 4-23, 4-43, 15-4](#)  
     Z39.50 server, [B-13, B-15](#)  
 log, [B-5](#)  
     catjob.log, [13-2](#)  
     circjob.log, [8-1](#)  
     mediajob.log, [9-3](#)  
     Z39.50, [B-15](#)  
 log.bursar.yymmdd.hhmm, [7-9](#)  
 log.exp.yyyymmdd.hhmm, [10-10](#)  
 log.imp.yyyymmdd.hhmm, [12-14](#)  
 log.prebulk.YYYYMMDD.HHMM, [11-22](#)  
 log.pupd.YYYYMMDD.HHMM, [6-7](#)  
 log.pxtr.YYYYMMDD.HHMM, [5-5](#)  
 log.strgfy.date.time, [14-9](#)  
 marc.exp.yymmdd.hhmm, [10-10](#)  
 out.prebulk.YYYYMMDD.HHMM, [11-22](#)  
 reject.imp.ccyymmdd.hhmm, [12-15](#)  
 replace.imp.ccyymmdd.hhmm, [12-15](#)  
 sif.burs.yymmdd.hhmm, [7-8](#)  
 standard interface  
     acquisitions notice, [22-1](#)  
     acquisitions report, [23-1](#)  
     cataloging report, [24-1](#)  
     charge transaction record, [19-1](#)  
     circulation notice, [25-1](#)  
     circulation report, [26-1](#)  
     item delete, [20-1](#)  
     media scheduling notice, [27-1](#)  
     media scheduling report, [28-1](#)  
     patron record, [18-1](#)  
     vendor record, [21-1](#)  
 VoyagerInstall.exe, [4-4, 4-5](#)  
 VoyagerInstall.msi, [4-4, 4-5](#)  
 VoyagerMedia.exe, [4-4, 4-13](#)  
 VoyagerMedia.msi, [4-4](#)  
 File Upload dialog box, [30-15](#)  
 files  
     log  
         UseMARCON Bulk Import log, [B-5](#)  
         UseMARCON MARC Export log, [B-5](#)  
         UseMARCON Z39.50 log, [B-5](#)  
     filesystem  
         data, [3-14](#)  
     fine/fee notice suffix segment  
         circulation notice, [25-7](#)  
     Fines/Fees History tab, [7-11](#)  
     fix exchange rates, [16-1](#)  
     forgive demerits, [8-16](#)  
         running circulation batch job, [8-17](#)  
     forgive fines  
         by date created, [30-25](#)  
         by patron group and expire date, [30-27](#)  
         by Patron ID, [30-24](#)  
         create date, [8-30](#)  
         patron group and expire date, [8-33](#)  
         patron ID, [8-27](#)  
     fund snapshot after rollover report suffix segment  
         acquisitions report, [23-12](#)  
     fund snapshot before rollover report suffix segment  
         acquisitions report, [23-12](#)  
     fund snapshot report suffix segment  
         acquisitions report, [23-10](#)  
 Fundledger\_vw, [29-12](#)

**G**

GetNewHits variable, 15-11  
 Getting Started, 1-1  
     prerequisite skills and knowledge, 1-1  
 GHC program  
     parameters  
         -d database name, 13-2  
         -h help, 13-4  
         -j job number, 13-3  
         -l list job options, 13-3  
         -L location code, 13-3, 13-6  
         -O operator, 13-6  
         -o operator name, 13-3  
         -u username and password, 13-3  
         -v version, 13-3  
     WebAdmin, 30-22  
 global circulation statistics report suffix segment  
     circulation report, 26-7, 26-8  
 Global Headings Change (GHC) program, 13-1  
 global open orders report suffix segment  
     acquisitions report, 23-9  
 [GlobalLog stanza, 4-27, 4-42]

**H**

Heading\_vw, 29-14  
 History tab, 12-10, 14-1, 14-3, 14-7  
 hold shelf expired report suffix segment  
     circulation report, 26-5  
 Holdings Location Code field, 30-15  
 httpd2.conf file, 30-5

**I**

ID number  
     record, B-10  
     UseMARCON record, B-10  
 ID Range radio button, 30-19  
 identifying data filesystem/volume, 3-14  
 Import Code field, 30-15  
 index.html file, 30-7  
 ini path not valid message, B-11, B-13, B-16

initialization file  
     Callslip.ini, 4-51  
     Circ.ini, 4-51  
     CircSC.ini, 4-51  
     EDI\_msg.ini, 4-51  
     Limits.ini, 4-51  
     Mediahelp.ini, 4-51  
     opac.ini, 15-5  
     sdiemail.ini, 15-9, 15-12  
     Template.ini, 4-50  
     UseMARCON, B-3  
     voyager.ini, 4-13, 4-23, 4-43, 15-4  
     Z39.50 server, B-13, B-15  
 input file  
     bulk import program, 12-11  
     patron update program, 6-3  
     prebulk program, 11-5  
     pstrgvfy program, 14-4  
 installation  
     modifying, 4-13  
 installation files  
     VoyagerInstall.exe, 4-2  
     VoyagerInstall.msi, 4-2  
     VoyagerMedia.exe, 4-2  
     VoyagerMedia.msi, 4-2  
 installation method  
     .bat file, 4-36  
     ftp, 4-36  
     http, 4-34  
     traditional, 4-33  
 intended audience  
     of this document, xlv  
 Issues\_vw, 29-15  
 item available notice suffix segment  
     circulation notice, 25-6  
 item type  
     bulk import program, 12-13  
 Item\_vw, 29-16

**J**

Job Scheduler page, 30-12  
 Job Scheduler screen, 30-16, 30-22, 30-23, 30-31, 30-34, 30-36

**K**

keyword index files  
     disabling generation during bulk import, [12-6](#)  
 KornShell window, [3-17](#)

**L**

Lang.cfg file, [4-50](#)  
 LCclass\_vw, [29-18](#)  
 Limits.ini file, [4-51](#)  
 loading  
     authority data, [2-2](#)  
     bibliographic data, [2-2](#)  
     circulation data (transactions), [2-3](#)  
     patron data, [2-3](#)  
     vendor data, [2-4](#)  
 Local Filename field, [30-15](#)  
 Location Code field, [2-2, 30-15](#)  
 [LOCATIONS] stanza, [11-13](#)  
 log (audit) file  
     prebulk program, [11-22](#)  
 Log Export  
     see OPAC Log Export program  
 log file, [B-5](#)  
     bulk import (UseMARCON), [B-5](#)  
     bulk import program, [12-13](#)  
     cataloging, [13-2](#)  
     circulation, [8-1](#)  
     MARC export (UseMARCON), [B-5](#)  
     media scheduling, [9-3](#)  
     Pfixexchangerates, [16-6](#)  
     pstrgvfy program, [14-9](#)  
     Z39.50, [B-15](#)  
     Z39.50 (UseMARCON), [B-5](#)  
 log file utility  
     WebAdmin, [30-36](#)  
 log.bursar.yymmdd.hhmm file, [7-9](#)  
 log.exp.yyyymmdd.hhmm file, [10-10](#)  
 log.forgive.YYYYMMDD.HHMM, [8-29, 8-33, 8-36](#)  
 log.imp.yyyymmdd.hhmm file, [12-14](#)  
 log.prebulk.YYYYMMDD.HHMM file, [11-22](#)  
 log.pupd.YYYYMMDD.HHMM file, [6-7](#)  
 log.ptrx.YYYYMMDD.HHMM file, [5-5](#)  
 log.strgvfy.date.time file, [14-9](#)  
 LOGBIBUSAGE= variable, [15-5](#)

logging in  
     using Single Client login, [4-42](#)  
 LOGSEARCH= variable, [15-4](#)

**M**

mapping  
     bulk import program, [12-13](#)  
 [MAPPING] stanza, [11-15](#)  
 MARC Export, [B-12-??](#)  
     sample script with UseMARCON switch, [B-12](#)  
     UseMARCON error reporting with, [B-13](#)  
 MARC export command, [B-12](#)  
 MARC Export error reporting, [B-13](#)  
 MARC Export record type switch, [B-12](#)  
 MARC Formats  
     supported by UseMARCON, [B-2](#)  
 [MARC POSTing] stanza, [4-28, 4-47](#)  
 MARC records, [B-12](#)  
 marc.exp.yymmdd.hhmm file, [10-10](#)  
 MARC21 format, [B-1, B-11](#)  
 MARC21 Repertoire.cfg file, [4-51](#)  
 Marcbook\_vw, [29-19](#)  
 Marccomputer\_vw, [29-19](#)  
 marcexport program, [10-1](#)  
     audit file, [10-10](#)  
     output file, [10-10](#)  
 parameters  
     -a assign character mapping, [10-8](#)  
         L Latin-1, [10-8](#)  
         M MARC21 MARC8, [10-8](#)  
         O OCLC, [10-8](#)  
         V Voyager Legacy, [10-8](#)  
     -c system control identifier, [10-4](#)  
     -h help, [10-9](#)  
     -l ignore suppressed bibliographic records, [10-7](#)  
     -L library, [10-7](#)  
     -m export mode, [10-4](#)  
         B both create and update dates, [10-4](#)  
         C create dates only, [10-4](#)  
         I ISBN input file, [10-4](#)  
         K OK to export file, [10-4](#)  
         M MARC ID input file, [10-4](#)  
         R MARC ID number range, [10-4](#)  
         S suppressed, [10-4](#)  
         U update dates only, [10-4](#)  
     -n create control number (001) from LCCN

- 9010a), [10-3](#)  
 -O only OCLC records, [10-8](#)  
 -o USMARC output filename, [10-3](#)  
 -q quiet, [10-9](#)  
 -r record type, [10-3](#)
  - A authority records, [10-3](#)
  - B bibliographic records, [10-3](#)
  - E series authority records, [10-3](#)
  - G Bib-MFHD record group, [10-3](#)
  - H holdings records, [10-3](#)
  - M main or added entry authority records, [10-3](#)
  - S subject authority records, [10-3](#)
 -s modifying agency, [10-4](#)  
 -t export target, [10-4](#)  
 -v version information, [10-9](#)  
 -w MFHD 852\$a updating, [10-8](#)  
 -X exclude, [10-6](#)  
 WebAdmin, [30-16](#)  
 Marcmaps\_vw, [29-20](#)  
 Marcmusic\_vw, [29-20](#)  
 Marcserial\_vw, [29-21](#)  
 Marcvisual\_vw, [29-22](#)  
 Matches found for Input Cancel Record. Import  
 Record Discarded! message, [12-17](#)  
 media equipment inventory report suffix segment  
 media scheduling report, [28-4](#)  
 media scheduling  
 batch job
  - retain patron IDs, [9-1](#)
 Media Scheduling batch jobs, [9-4](#)  
 media scheduling notice
  - base segment, [27-3](#)
  - overdue notice suffix segment, [27-5](#)
  - standard interface file (SIF), [27-1](#)
  - suffix segment, [27-4](#)
 media scheduling report
  - base segment, [28-3](#)
  - booking charge statistics report suffix segment, [28-5](#)
  - booking exceptions report suffix segment, [28-5](#)
  - booking statistics report suffix segment, [28-4](#)
  - media equipment inventory report suffix segment, [28-4](#)
  - standard interface file (SIF), [28-1](#)
  - suffix segment, [28-3](#)
 media scheduling reports and notices
  - WebAdmin, [30-32](#)
 Mediahelp.ini file, [4-51](#)  
 mediajob 5, [9-1](#)
  - parameter
 -n number of IDs to retain, [9-3](#)  
 running
  - command line, [9-4](#)
  - interactively, [9-5](#)
 mediajob.log file, [9-3](#)  
 message
  - END non OK, [B-8](#)
  - Failed to retrieve record, [B-16](#)
  - ini path not valid, [B-11, B-13, B-16](#)
  - Matches found for Input Cancel Record. Import  
 Record Discarded, [12-17](#)
  - No item option in use - No item records created, [12-17](#)
  - no USEMARCON ini entry for this OID, [B-16](#)
  - Non-Bib Record detected! Holdings record written  
 to error file, [12-17](#)
  - oid error, [B-16](#)
  - only one type of USEMARCON conversion allowed  
 per session, [B-16](#)
  - record not translated, [B-11, B-13, B-16](#)
  - WARNING, [B-8](#)
 Messages
  - record not translated, [B-11](#)
 MFHDblob\_vw, [29-22](#)  
 MFHDhistory\_vw, [29-23](#)  
 [MFHDTAG] stanza, [11-11](#)  
 middle dot character, [10-9](#)  
 missing in transit report suffix segment  
 circulation report, [26-6](#)  
 modifying a login/password  
 WebAdmin, [30-4](#)  
 modifying installation, [4-13](#)  
 Monthly variable, [15-12](#)
- 
- N**
- NewBook.cfg file, [4-50](#)  
 NewHits variable, [15-11](#)  
 NewSerial.cfg file, [4-50](#)  
 Nlmclass\_vw, [29-23](#)  
 nnnn variable, [B-10](#)  
 No item option in use - No item records created message, [12-17](#)  
 No USEMARCON ini entry for this OID message, [B-16](#)  
 nohup command, [5-2, 6-2, 7-3, 10-2, 12-3](#)  
 noisewordfilter, [17-12](#)  
 Non-Bibliographic Record detected! Holdings record

written to error file message, [12-17](#)  
None variable, [15-11](#)  
Non-MARC21 format, [B-1](#), [B-11](#)  
Non-MARC21 records, [B-13](#)  
non-standard 035 field  
stripping, [11-10](#)

---

## O

OCLC Control Number, [12-12](#)  
oid error message, [B-16](#)  
Only one type of USEMARCON conversion allowed  
per session message, [B-16](#)  
OPAC Bib Usage Log Export program, [15-1](#), [15-15](#),  
[30-34](#)  
    running as cron, [15-17](#)  
    running interactively, [15-15](#)  
OPAC Log Export program, [15-1](#), [30-34](#)  
OPAC reports  
    WebAdmin, [30-34](#)  
OPAC Search Log Export program, [15-6](#)  
    running as cron, [15-8](#)  
    running interactively, [15-7](#)  
OPAC search logging  
    configuring, [15-4](#)  
OPAC\_Search\_Log table, [15-2](#)  
opac.ini file, [15-5](#)  
open orders after rollover report suffix segment  
    acquisitions report, [23-12](#)  
open orders before rollover report suffix segment  
    acquisitions report, [23-12](#)  
open orders report suffix segment  
    acquisitions report, [23-8](#)  
Operator Name field, [30-15](#)  
order claim notice suffix segment  
    acquisitions notice, [22-7](#)  
out.prebulk.YYYYMMDD.HHMM file, [11-22](#)  
output file  
    bursar transfer program, [7-8](#)  
    circjob 34, [8-11](#)  
    circjob 35, [8-13](#)  
    circjob 36, [8-15](#)  
    marcexport program, [10-10](#)  
    patron extract program, [5-4](#)  
    prebulk program, [11-22](#)  
overdue notice suffix segment  
    circulation notice, [25-6](#)  
    media scheduling notice, [27-5](#)

[Overrides] stanza, [11-8](#)

---

## P

page  
    Acquisitions Reports and Notices, [30-9](#)  
    Job Scheduler, [30-12](#)  
    Voyager Server Utilities (main), [30-7](#)  
    webadmin main page, [30-9](#)  
parameter  
    circjob 38  
        -Tnumber of IDs to retain, [8-18](#)  
    mediajob 5  
        -n number of IDs tp retain, [9-3](#)  
parameters  
    bulk import program  
        -a MFHD location code, [12-8](#)  
        -b begin record, [12-8](#)  
        -d delete MFHDs, [12-9](#)  
        -e end record, [12-8](#)  
        -f filename, [12-7](#)  
        -h help, [12-10](#)  
        -i import code, [12-7](#)  
        -k OK to export, [12-10](#)  
        -l location, [12-8](#)  
        -m load MFHDs, [12-8](#)  
        -o operator name, [12-8](#)  
        -x delete bibliographic records, [12-9](#)  
        -X NOKEY, [12-7](#), [12-10](#), [30-12](#)  
    bursar transfer program  
        -c configuration file, [7-6](#)  
        -d database name, [7-6](#)  
        -h help, [7-7](#)  
        -i interval for transfer, [7-7](#)  
        -m minimum fine/fee amount, [7-6](#)  
    o operator ID, [7-7](#)  
    -p patron mode, [7-6](#)  
    -q quiet mode, [7-7](#)  
    -t test mode, [7-7](#)  
    -u username and password, [7-6](#)  
    -v version, [7-7](#)  
circjob 34  
    -e end date, [8-10](#), [8-12](#)  
    -s start date, [8-10](#), [8-12](#)  
circjob 35  
    -D due date, [8-13](#)  
    -e end date, [8-12](#)  
    -i operator id, [8-12](#)

---

-L location code, [8-12](#)  
 -P print location, [8-13](#)  
 -s start date, [8-12](#)  
**circjob** 36  
 -e end date, [8-15](#)  
 -s start date, [8-15](#)  
**circjob** 37  
 -q number of demerits to forgive, [8-16](#)  
**GHC program**  
 -d database name, [13-2](#)  
 -h help, [13-4](#)  
 -j job number, [13-3](#)  
 -l list job options, [13-3](#)  
 -L location code, [13-3, 13-6](#)  
 -O operator, [13-6](#)  
 -o operator name, [13-3](#)  
 -u username and password, [13-3](#)  
 -v version, [13-3](#)  
**marcexport program**  
 -a assign character mapping, [10-8](#)  
     L Latin-1, [10-8](#)  
     M MARC21 MARC8, [10-8](#)  
     O OCLC, [10-8](#)  
     Voyager Legacy, [10-8](#)  
 -c system control identifier, [10-4](#)  
 -h help, [10-9](#)  
 -l ignore suppressed bibliographic records, [10-7](#)  
 -L library, [10-7](#)  
 -m export mode, [10-4](#)  
     B both create and update dates, [10-4](#)  
     C create dates only, [10-4](#)  
     I ISBN input file, [10-4](#)  
     K OK to export file, [10-4](#)  
     M MARC ID input file, [10-4](#)  
     R MARC ID number range, [10-4](#)  
     S suppressed, [10-4](#)  
     U update dates only, [10-4](#)  
 -n create control number (001) from LCCN  
     9010a), [10-3](#)  
 -O only OCLC records, [10-8](#)  
 -o USMARC output filename, [10-3](#)  
 -q quiet, [10-9](#)  
 -r record type, [10-3](#)  
     A authority records, [10-3](#)  
     B bibliographic records, [10-3](#)  
     E series authority records, [10-3](#)  
     G Bib-MFHD record group, [10-3](#)  
     H holdings records, [10-3](#)  
     M main or added entry authority records,  
         [10-3](#)

S subject authority records, [10-3](#)  
 -s modifying agency, [10-4](#)  
 -t export target, [10-4](#)  
 -v version information, [10-9](#)  
 -w MFHD 852\$a updating, [10-8](#)  
 -X exclude, [10-6](#)  
**patron extract program**  
 -a audit filename, [5-3](#)  
 -d database name, [5-2](#)  
 -e error filename, [5-3](#)  
 -h help, [5-4](#)  
 -m monitor records to process, [5-4](#)  
 -p patron output filename, [5-3](#)  
 -r number of records to process, [5-4](#)  
 -u username and password, [5-2](#)  
**patron update program**  
 -a audit filename, [6-4](#)  
 -d database name, [6-3](#)  
 -e error filename, [6-4](#)  
 -h help, [6-6](#)  
 -i index type, [6-3](#)  
     D program selected, [6-4](#)  
     I institution ID, [6-4](#)  
     S ssan, [6-3](#)  
 -m monitor records to process, [6-6](#)  
 -n notes type, [6-5](#)  
     G general, [6-5](#)  
     P popup, [6-5](#)  
 -n number of records to process, [6-6](#)  
 -o overlay previous notes, [6-5](#)  
     A add new notes, [6-5](#)  
     I ignore notes processing, [6-5](#)  
     R replace existing notes, [6-5](#)  
 -p patron input filename, [6-3](#)  
 -u username and password, [6-3](#)  
 -x extended notes processing, [6-5](#)  
**prebulk program**  
 -a append tag for log file, [11-21](#)  
 -c configuration file, [11-20](#)  
 -e error file, [11-20](#)  
 -f force overwrite of output files switch, [11-21](#)  
 -i input file, [11-20](#)  
 -l prebulk log file, [11-21](#)  
 -o output file, [11-20](#)  
 -p parameter file, [11-20](#)  
**pstrgvfy program**  
 -a alter mode flag, [14-6](#)  
 -b update MFHD 852b, [14-6](#)  
 -c command filename help, [14-7](#)  
 -d database name, [14-5](#)  
 -g cataloging location, [14-7](#)

-h help, 14-8  
 -i input file, 14-5  
 -l log filename, 14-8  
 -m location, 14-5  
 -o operator ID, 14-7  
 -u username and password, 14-5  
 -v verify mode flag, 14-6  
 Password field, 30-7  
 patron  
     authentication, C-1  
 Patron Authentication Adaptor, C-1  
 patron data  
     loading, 2-3  
 patron extract program, 5-1  
     audit file, 5-5  
     error file, 5-6  
     output file, 5-4  
     parameters  
         -a audit filename, 5-3  
         -d database name, 5-2  
         -e error filename, 5-3  
         -h help, 5-4  
         -m monitor records to process, 5-4  
         -p patron output filename, 5-3  
         -r number of records to process, 5-4  
         -u username and password, 5-2  
     patron record standard interface file (SIF), 5-4  
     running, 5-4  
 Patron Group Code field, 2-3  
 [Patron Groups] stanza, 7-5  
 patron purge, 8-3, 8-23  
 patron suspension, 8-7  
     running circulation batch job, 8-8, 8-9  
 patron update program, 6-1  
     audit file, 6-7  
     error file, 6-8  
     input file, 6-3  
     parameters  
         -a audit filename, 6-4  
         -d database name, 6-3  
         -e error filename, 6-4  
         -h help, 6-6  
         -i index type, 6-3  
     D program selected, 6-4  
     I institution ID, 6-4  
     S ssan, 6-3  
         -m monitor records to process, 6-6  
         -n notes type, 6-5  
     G general, 6-5  
     P popup, 6-5  
         -n number of records to process, 6-6  
     -o overlay previous notes, 6-5  
         A add new notes, 6-5  
         I ignore notes processing, 6-5  
         R replace existing notes, 6-5  
     -p patron input filename, 6-3  
     -u username and password, 6-3  
     -x extended notes processing, 6-5  
     patron record standard interface file (SIF), 6-3  
     running, 6-6  
 Pbibredirect.cgi script, 4-47  
 pbulkimport command, 12-10, B-11  
 pbursar command, 7-7  
 Pcircjob, 8-3, 8-4, 8-5, 8-6, 8-7, 8-8, 8-9, 8-11, 8-14, 8-15, 8-17, 8-20, 8-25, 8-26, 8-27, 8-28, 8-30, 8-31, 8-33, 8-34, 8-39  
 Pfixexchangerates  
     log file, 16-6  
 photocopying  
     documentation, xlxx  
 Phttplinkresolver.cgi script, 4-47  
 place items on active course reserve list, 8-10  
     running circulation batch job, 8-11  
 place recalls and holds for items on active course  
     reserve list, 8-12  
 pmarcexport command, 10-9, B-12  
 Popacjob, 15-1  
 Popaclogexp  
     see OPAC Log Export program  
 populating operator or location on MFHD's history tab, 14-3  
 Post button, 7-11  
 POST request, 4-47  
 ppbulk command, 11-21  
 pptrnextr command, 5-4  
 pptrnupd command, 6-6  
 prebulk program, 11-1  
     checking records, 11-3  
 configuration file, 11-5  
     location, 11-6  
     name, 11-7  
     stanza types, 11-7  
 creating  
     holdings from interleaved file, 11-5  
     interleaved file, 11-4  
 creating interleaved bib-MFHD file, 11-4  
 error file, 11-22  
 importing interleaved file, 11-4  
 input file, 11-5  
 log (audit) file, 11-22  
 output file, 11-22  
 parameters

- a append tag for log file, [11-21](#)
  - c configuration file, [11-20](#)
  - e error file, [11-20](#)
  - f force overwrite of output files switch, [11-21](#)
  - i input file, [11-20](#)
  - l prebulk log file, [11-21](#)
  - o output file, [11-20](#)
  - p parameter file, [11-20](#)
  - running, [11-21](#)
  - striping fields, [11-3](#)
  - Preview button, [13-5](#)
  - Private Use Area, [12-6](#)
  - process
    - GHC fields-step 3 catjob 13, [13-6](#)
    - GHC queue-step 1
      - running cataloging batch job, [13-4](#)
    - GHC queue-step 1 catjob 11, [13-4](#)
    - GHC queue-step 2
      - running cataloging batch job, [13-5](#)
    - GHC queue-step 2 catjob 12, [13-5](#)
  - Process box, [13-5](#)
  - program
    - bulk export of MARC records (marcexport), [10-1](#)
    - bulk import, [12-1](#)
    - bursar transfer, [7-1](#)
      - stub patron records, [7-2](#)
    - Global Headings Change (GHC), [13-1](#)
    - marcexport, [10-1](#)
    - patron extract, [5-1](#)
    - patron update, [6-1](#)
    - prebulk, [11-1](#)
    - pstrgfy, [14-1](#)
    - storage barcode verify (pstrgfy), [14-1](#)
    - promote.cfg file, [8-9](#)
    - pstrgfy command, [14-10](#)
    - pstrgfy program, [14-1](#)
      - changing location codes of item records or MFHDs, [14-2](#)
      - command file, [14-8](#)
      - input file, [14-4](#)
      - log file, [14-9](#)
      - parameters
        - a alter mode flag, [14-6](#)
        - b update MFHD 852b, [14-6](#)
        - c command filename, [14-7](#)
        - d database name, [14-5](#)
        - g cataloging location, [14-7](#)
        - h help, [14-8](#)
        - i input file, [14-5](#)
        - l log filename, [14-8](#)
        - m location, [14-5](#)
  - o operator ID, [14-7](#)
  - u username and password, [14-5](#)
  - v verify mode flag, [14-6](#)
  - populating operator or location on MFHD's history tab, [14-3](#)
  - verifying barcodes of item records or MFHDs, [14-4](#)
  - PUA, [12-6](#)
  - purchase order report suffix segment
    - acquisitions report, [23-5](#)
  - purge UB patron stub record, [8-5, 8-8](#), [8-6](#)
  - purpose
    - of this document, [xlv](#)
  - pwebadmin command, [30-3](#)
  - Pz3950svr, [17-14](#)
- 
- ## R
- radio button
    - Create one summary transfer record per patron, [30-31](#)
    - Create transfer record for each fine/fee transferred, [30-31](#)
    - Date Range, [30-20](#)
    - ID Range, [30-19](#)
    - Today, [30-19](#)
  - rebooting
    - Windows 2000 server, [3-17](#)
  - recall notice suffix segment
    - circulation notice, [25-7](#)
  - recall-overdue notice suffix segment
    - circulation notice, [25-7](#)
  - record
    - bibliographic, [B-12](#)
  - Record does not match format for " change your import rule message, [12-17](#)
  - record ID number, [B-10](#)
  - Record nnn discarded -- no matching records in the database message, [12-17](#)
  - Record not translated message, [B-11](#)
  - record not translated message, [B-11, B-13, B-16](#)
  - Record Parse Failure message, [12-17](#)
  - Record Retrieval Failure message, [12-17](#)
  - Recordcount\_vw, [29-24](#)
  - refunding
    - bursar transfer program, [7-10](#)
  - reissue

- reason for, [xlv](#)  
 reject file  
   bulk import program, [12-15](#)  
 reject.imp.ccyyymmdd.hhmm file, [12-15](#)  
 repair, [4-18](#)  
 repairing  
   installation, [4-18](#)  
 replace file  
   bulk import program, [12-15](#)  
 replace.imp.ccyyymmdd.hhmm file, [12-15](#)  
 report file utility  
   WebAdmin, [30-36](#)  
 Reports.mdb file, [4-50](#)  
 reproduction, of documentation, [xlix](#)  
 reserved items active report suffix segment  
   circulation report, [26-5](#)  
 reserved items expired report suffix segment  
   circulation report, [26-5](#)  
 restoring  
   from backup, [3-27](#)  
 retain patron IDs, [8-17](#)  
   media batch job, [9-1](#)  
 return notice suffix segment  
   acquisitions notice, [22-6](#)  
 Rule Code field, [30-15](#)  
 Run dialog box, [4-3, 4-6, 4-14, 4-19](#)  
 running  
   bulk import program, [12-10](#)  
   bursar transfer program, [7-7](#)  
   cataloging batch job  
     process GHC field-step 3, [13-6](#)  
     process GHC queue-step 1, [13-4](#)  
     process GHC queue-step 2, [13-5](#)  
   circjob 38  
     command line, [8-19](#)  
     interactively, [8-20](#)  
   circulation batch job  
     accrued fines and demerits, [8-7](#)  
     archive and expire call slip requests, [8-4](#)  
     archive short loans, [8-5](#)  
     export OPAC requests, [8-4](#)  
     forgive demerits, [8-17](#)  
     patron suspension, [8-8, 8-9](#)  
     place items on active course reserve list, [8-11](#)  
     purge UB patron stub record, [8-6](#)  
     update shelving status, [8-3, 8-14, 8-15](#)  
 mediajob 5  
   command line, [9-4](#)  
   interactively, [9-5](#)  
 OPAC Bib Usage Log Export program as cron, [15-17](#)  
 OPAC Bib Usage Log Export program interactively, [15-15](#)  
 OPAC Search Log Export program as cron, [15-8](#)  
 OPAC Search Log Export program interactively, [15-7](#)  
 patron extract program, [5-4](#)  
 patron update program, [6-6](#)  
 prebulk program, [11-21](#)  
 Selective Dissemination of Information as cron, [15-15](#)  
 Selective Dissemination of Information interactively, [15-14](#)  
 running acqjob 5, [16-3](#)  
 running acquisitions batch jobs  
   webadmin, [30-9](#)  
 RunSearchEvery variable, [15-10](#)
- 
- ## S
- Schedule Task window, [3-27](#)  
 Scheduled Task Wizard window, [3-25](#)  
 screen  
   Job Scheduler, [30-16, 30-22, 30-23, 30-31, 30-34, 30-36](#)  
 SDI  
   see Selective Dissemination of Information  
 [SDI\_Page] stanza, [15-10](#)  
 sdiemail.ini file, [15-9](#)  
   components of, [15-12](#)  
 SDIOption= variable, [15-10](#)  
 Search dialog box  
   additional button, [4-29](#)  
 search dialog box  
   URI link, [4-29](#)  
 [SearchURI stanza, [4-29](#)  
 secure socket layer (SSL), [4-43](#)  
 see references with linked bib records suffix segment  
   cataloging report, [24-7](#)  
 see refs authorized in other auth record suffix segment  
   cataloging report, [24-7](#)  
 see refs without corresponding auth record suffix segment  
   cataloging report, [24-8](#)  
 Selective Dissemination of Information, [15-9](#)  
   configuring, [15-10](#)  
   enabling, [15-10](#)  
   running as cron, [15-15](#)  
   running interactively, [15-14](#)

search results page  
     customizing, [15-12](#)  
 Send Record To button, [4-49](#)  
 Send Record To option, [4-48](#)  
 serial claim notice suffix segment  
     acquisitions notice, [22-7](#)  
 Serials\_vw, [29-24](#)  
 server  
     AIX, [3-4](#)  
     Solaris, [3-4](#)  
     Windows 2000, [3-16](#)  
 Services window, [3-17](#)  
 Set Account Information window, [3-26](#)  
 setting up  
     WebAdmin, [30-2](#)  
 Settings tab, [3-26](#)  
 Shut Down Windows window, [3-17](#)  
 shutting down  
     Windows 2000 server, [3-17](#)  
 sif.burs.yymmdd.hhmm file, [7-8](#)  
 Single Client Login, [4-42](#)  
 Solaris server, [3-4](#)  
     starting  
         Apache, [3-10](#)  
         Oracle services, [3-6](#)  
         Voyager, [3-8](#)  
     stopping  
         Apache, [3-10](#)  
         Oracle services, [3-6](#)  
         Voyager, [3-8](#)  
 Special Characters.cfg file, [4-51](#)  
 Spinelabel.cfg file, [4-51](#)  
 standard interface file  
     acquisitions notice, [22-1](#)  
     base segment, [22-4](#)  
     cancel serial claim notice suffix segment, [22-8](#)  
     cancellation notice suffix segment, [22-6](#)  
     order claim notice suffix segment, [22-7](#)  
     return notice suffix segment, [22-6](#)  
     serial claim notice suffix segment, [22-7](#)  
     suffix segment, [22-5](#)  
     voucher/check request suffix segment, [22-8](#)  
 acquisitions report, [23-1](#)  
     base segment, [23-4](#)  
     fund snapshot after rollover report suffix segment, [23-12](#)  
     fund snapshot before rollover report suffix segment, [23-12](#)  
     fund snapshot report suffix segment, [23-10](#)  
     global open orders report suffix segment, [23-9](#)  
     open orders after rollover report suffix segment, [23-12](#)  
 open orders before rollover report suffix segment, [23-12](#)  
 open orders report suffix segment, [23-8](#)  
 purchase order report suffix segment, [23-5](#)  
 suffix segment, [23-4](#)  
 cataloging report, [24-1](#)  
     856 link failure report suffix segment, [24-8](#)  
     base segment, [24-4](#)  
     duplicate authority records suffix segment, [24-6](#)  
     see references with linked bib records suffix segment, [24-7](#)  
     see refs authorized in other auth record suffix segment, [24-7](#)  
     see refs without corresponding auth record suffix segment, [24-8](#)  
     suffix segment, [24-4](#)  
     unauthorized name headings suffix segment, [24-5](#)  
     unauthorized name/title headings suffix segment, [24-6](#)  
     unauthorized subdivision headings suffix segment, [24-6](#)  
     unauthorized subject headings suffix segment, [24-5](#)  
     unauthorized title headings suffix segment, [24-5](#)  
 circulation notice, [25-1](#)  
     base segment, [25-4](#)  
     cancellation notice suffix segment, [25-6](#)  
     courtesy (due) notice suffix segment, [25-9](#)  
     fine/fee notice suffix segment, [25-7](#)  
     item available notice suffix segment, [25-6](#)  
     overdue notice suffix segment, [25-6](#)  
     recall notice suffix segment, [25-7](#)  
     recall-overdue notice suffix segment, [25-7](#)  
     statement of fines and fees suffix segment, [25-8](#)  
     suffix segment, [25-5](#)  
 circulation report, [26-1](#)  
     base segment, [26-4](#)  
     circulation item exceptions report suffix segment, [26-7](#)  
     circulation patron exceptions report suffix segment, [26-7](#)  
     circulation statistics report suffix segment, [26-6](#)  
     circulation transaction exceptions rept suffix segment, [26-7](#)  
     distribution item order list report suffix segment,

- 26-8
- global circulation statistics report suffix segment, 26-7, 26-8
- hold shelf expired report suffix segment, 26-5
- missing in transit report suffix segment, 26-6
- reserved items active report suffix segment, 26-5
- reserved items expired report suffix segment, 26-5
- suffix segment, 26-4
- media scheduling notice, 27-1
- base segment, 27-3
- overdue notice suffix segment, 27-5
- suffix segment, 27-4
- media scheduling report, 28-1
- base segment, 28-3
- booking charge statistics report suffix segment, 28-5
- booking exceptions report suffix segment, 28-5
- booking statistics report suffix segment, 28-4
- media equipment inventory report suffix segment, 28-4
- suffix segment, 28-3
- standard interface file (SIF)
  - bursar transfer program, 7-8
  - charge transaction record, 19-1
  - item delete, 20-1
  - patron record, 5-4, 6-3, 18-1
  - vendor record, 21-1
- stanza
  - [008], 11-17
  - [CALLTYPES], 11-12
  - [Circ Locations], 7-5
  - [E-mail], 4-28
  - [Email], 15-12
  - [GlobalLog], 4-27, 4-42
  - [LOCATIONS], 11-13
  - [MAPPING], 11-15
  - [MARC POSTing], 4-28, 4-47
  - [MFHDTAG], 11-11
  - [Overrides], 11-8
  - [Patron Groups], 7-5
  - [SDI\_Page], 15-10
  - [SearchURI], 4-29
  - [Strip], 11-9
  - [Upgrade], 4-30
  - [Usemarcon translations], B-14
  - [Webvoyage\_Server], 15-12
- starting
  - Apache
- AIX server, 3-10
- Solaris server, 3-10
- Windows 2000 server, 3-20
- Oracle services
  - AIX server, 3-6
  - Solaris server, 3-6
  - Windows 2000 server, 3-18
- Voyager
  - AIX server, 3-8
  - Solaris server, 3-8
  - Windows 2000 server, 3-19
- WebAdmin, 30-6
- WebAdmin server daemon, 30-3
- statement of fines and fees suffix segment
  - circulation notice, 25-8
- StatString= variable, 15-5
- stopping
  - Apache
    - AIX server, 3-10
    - Solaris server, 3-10
    - Windows 2000 server, 3-20
  - Oracle services
    - AIX server, 3-6
    - Solaris server, 3-6
    - Windows 2000 server, 3-18
- Voyager
  - AIX server, 3-8
  - Solaris server, 3-8
  - Windows 2000 server, 3-19
- storage barcode verify (pstrvfy) program, 14-1
- [Strip] stanza, 11-9
- striping fields with prebulk, 11-3
- stripping
  - non standard 035 field, 11-10
- Submit Batch Import button, 30-16
- Submit Bulk Export button, 30-22
- Submit Bursar Transfer button, 30-31
- Submit OPAC Job button, 30-35
- Sudoclass\_vw, 29-25
- suffix segment
  - acquisitions notice, 22-5
  - acquisitions report, 23-4
  - cataloguing report, 24-4
  - circulation notice, 25-5
  - circulation report, 26-4
  - media scheduling notice, 27-4
  - media scheduling report, 28-3
- System Control (003) Identifier field, 30-19
- System Control (040\$) Identifier field, 30-19
- system requirements
  - WebAdmin, 30-2

System tab, [12-10](#), [30-15](#)

## T

### tab

- Backup, [3-25](#)
- Fines/Fees History, [7-11](#)
- History, [12-10](#), [14-1](#), [14-3](#), [14-7](#)
- Settings, [3-26](#)
- System, [12-10](#), [30-15](#)

takes items on inactive course reserve lists off reserve, [8-14](#)

Template.ini file, [4-50](#)

Today radio button, [30-19](#)

Transfer fines/fees greater than or equal to field, [30-31](#)

### transferring

- itemized patron fines/fees, [7-4](#)
- total patron balances, [7-4](#)

## U

### unauthorized name headings suffix segment

- cataloging report, [24-5](#)

### unauthorized name/title headings suffix segment

- cataloging report, [24-6](#)

### unauthorized subdivision headings suffix segment

- cataloging report, [24-6](#)

### unauthorized subject headings suffix segment

- cataloging report, [24-5](#)

### unauthorized title headings suffix segment

- cataloging report, [24-5](#)

Unicode, [12-6](#)

Unicode character set, [30-21](#)

uninstalling clients, [4-20](#)

Unparseable record written to error file message, [12-17](#)

update remote circulation cluster cache

- circjob 33, [8-9](#)

update shelving status, [8-3](#)

- running circulation batch job, [8-3](#), [8-14](#), [8-15](#)

[Upgrade] stanza, [4-30](#)

USE001FOR014 variable, [11-9](#)

USE003FORLOC variable, [11-9](#)

UseMARCON

- API, [10-2](#), [12-1](#), [B-1](#)
- bulk import, [B-10](#)
- conversion directories, [B-2](#)–[B-3](#)
- error reporting, [B-11](#)
  - END non OK message, [B-8](#)
  - MARC Export, [B-13](#)
  - Non-fatal errors, [B-8](#)
  - Z39.50, [B-15](#)
- file structure, [B-2](#)–[B-5](#)
- initialization file, [B-3](#)
- log file, [B-5](#)
- record ID number, [B-10](#)
- required files, [B-4](#)
  - Input format checking (.chk) file, [B-4](#)
  - MARC Input (.mrc) file, [B-5](#)
  - MARC Output (.mrc) file, [B-5](#)
  - Output format checking (.chk) file, [B-4](#)
  - Rule (.rul) file, [B-4](#)
  - Translation (.trs) file, [B-4](#)
- translations, [B-11](#)
- troubleshooting error conditions, [B-8](#)–??
- [Usemarcon translations] stanza, [B-14](#)
- usemarcon.bip.date.time file, [B-5](#)
- usemarcon.mxp.date.time file, [B-5](#)
- usemarcon.z39.date.time file, [B-5](#)
- User Name field, [30-6](#)
- using
- prebulk to check records, [11-3](#)
- prebulk to create interleaved file, [11-4](#)
- prebulk to strip fields, [11-3](#)
- WebAdmin, [30-7](#)

## V

### variable

- Bi-Weekly, [15-12](#)
- Daily, [15-11](#)
- GetNewHits, [15-11](#)
- LOGBIBUSAGE=, [15-5](#)
- LOGSEARCH=, [15-4](#)
- Monthly, [15-12](#)
- NewHits, [15-11](#)
- nnnn, [B-10](#)
- None, [15-11](#)
- RunSearchEvery, [15-10](#)
- SDIOption=, [15-10](#)
- StatString=, [15-5](#)
- Weekly, [15-11](#)

vendor data  
    loading, [2-4](#)  
Vendorinvoice\_vw, [29-26](#)  
Vendororder\_vw, [29-28](#)  
verifying barcodes of item records or MFHDs, [14-4](#)  
viewing  
    bursar transfer program information, [7-10](#)  
voucher/check request suffix segment  
    acquisitions notice, [22-8](#)  
Voyager Server Utilities (main) page, [30-7](#)  
voyager.env, [17-1](#)  
voyager.ini file, [4-13, 4-23, 15-4](#)  
    module stanzas, [4-26](#)  
voyager.ini file (server side)  
    noisewordfilter, [17-12](#)  
voyager.ini file (server-side), [4-43](#)  
VoyagerInstall.exe file, [4-4, 4-5, 4-23](#)  
VoyagerInstall.msi file, [4-4, 4-5](#)  
VoyagerMedia.exe file, [4-4, 4-13](#)  
VoyagerMedia.msi file, [4-4](#)

---

WebVoyage, [15-9](#)  
[Webvoyage\_Server] stanza, [15-12](#)  
Weekly variable, [15-11](#)  
window  
    Daily Backup, [3-26](#)  
    KornShell, [3-17](#)  
    Schedule Task, [3-27](#)  
    Scheduled Task Wizard, [3-25](#)  
    Services, [3-17](#)  
    Set Account Information window, [3-26](#)  
    Shut Down Windows, [3-17](#)  
Windows 2000 server, [3-16](#)  
    backing up, [3-21](#)  
    rebooting, [3-17](#)  
    shutting down, [3-17](#)  
    starting  
        Apache, [3-20](#)  
        Oracle services, [3-18](#)  
        Voyager, [3-19](#)  
    stopping  
        Apache, [3-20](#)  
        Oracle services, [3-18](#)  
        Voyager, [3-19](#)

## W

---

WARNING message, [B-8](#)  
WebAdmin, [30-1](#)  
    acquisitions utilities, [30-9](#)  
    adding a login/password, [30-4](#)  
    bulk import program, [30-12](#)  
    bursar transfer program, [30-29](#)  
    cataloging utilities, [30-12](#)  
    circulation reports and notices, [30-24](#)  
    creating a new login/password, [30-3](#)  
    deleting a login/password, [30-4](#)  
    GHC program, [30-22](#)  
    log file utility, [30-36](#)  
    marcexport program, [30-16](#)  
    media scheduling reports and notices, [30-32](#)  
    modifying a login/password, [30-4](#)  
    OPAC reports, [30-34](#)  
    report file utility, [30-36](#)  
    setting up, [30-2](#)  
    starting, [30-6](#)  
        sever daemon, [30-3](#)  
    system requirements, [30-2](#)  
    using, [30-7](#)  
webadmin  
    running acquisitions reports and notices, [30-9](#)  
webadmin main page, [30-9](#)

## X

---

XML entity reference, [10-8](#)

---

## Z

Z39.50, [B-13](#)  
Z39.50 clients, [B-13](#)  
Z39.50 Server, [B-13](#)  
    initialization file, [B-13](#)  
    log file, [B-15](#)  
    UTF-8 encoded records, [17-13](#)  
z3950svr.ini, [17-13](#)  
z3950svr.ini file, [B-13](#)  
z3950svr.log file, [B-15](#)