DESTINY™

# XML Transform Utility
for Destiny Patrons

Introduction

Follett Software Company's Destiny™ solution can import patron data in a comma-delimited format (Follett's Circulation Plus® patron file format) or an XML-based data format. There are several important differences to consider when choosing between these two file formats:

Differences in using Comma-delimited vs. XML patron file formats	
Standard comma-delimited file format	**XML file format**
Uploaded through Destiny web page	Uploaded through Destiny web page OR imported via a command-line utility on Destiny app server
Only one school per file	Up to entire district in one import file
Accessed by Site-level user (Back Office/Import Patrons)	Accessed by district-level user (Back Office/Update Patrons)
Add or Update patrons	Add, Update/Transfer, or Delete patrons
Can be created via Follett's "Patron Conversion Utility" from a variety of input file formats	XML must be provided by user. The Transform utility helps you convert comma- or tab-delimited data into XML

Some school districts find it simpler to use a comma- or tab-delimited file format. However, for districts wishing to create a batch-driven update of patron data into Destiny, the XML file format must be used. The Transform utility will help you convert library patron data from a comma- or tab-delimited format to an XML format suitable for batch-driven import to Destiny. This document provides an overview of the Transform process.

Note: Another name for comma-delimited data is "comma separated values" or CSV. For brevity this document will refer to comma or tab-delimited input data as CSV files.

Process Overview

Transform converts data from comma or tab-delimited format into Destiny XML in two basic steps. The first step converts the input CSV file into a "raw" XML format. Each line from the CSV file generates in a "<line>" element and each field in the line generates a "<field>" element in the resulting raw XML file. This process works the same regardless of the specific CSV file format—It is a generic conversion of CSV to XML.

The second step in the Transform process reads the "raw" XML file and produces a final Destiny-formatted XML file. This conversion is specific to the exact field contents and data format of the input data. The details by which this conversion occurs are embodied in a "style sheet". A style sheet is an XML file with the extension ".XSL" that contains formatting commands. This process is called XSL transformation (XSLT) and is an industry standard for converting XML from one form to another.

XSLT has many capabilities for converting XML data. Because XSLT is an industry standard, there are many third-party books and other resources available to help provide you the tools necessary to apply XSLT to your unique data needs.

An Example

Let’s walk through a simple end-to-end example using a small patron data file. While reading this walk-through, refer to the flowchart that follows. This entire process is depicted on that flowchart. Rundemo.bat will run the complete demo process.

Consider an input file named demopatrons.csv. This file contains two records, each with a barcode, district ID, last name, and first name:

demopatrons.csv

```
"BC1234", "ID1234", "Smith", "John"
"BC6789", "ID6789", "Jones", "Molly"
```

Step One

Step One will be to call Transform.bat, giving it the “CSVXML” command to tell it to create XML from a CSV file.

```
C:\FSC-Destiny\fsc\transform> Transform CSVXML demopatrons.csv demo_raw.xml
```

The “raw” XML file produced by this command looks like:

demo_raw.xml

```
<csvfile>
  <line>
    <field_0>BC1234</field_0>
    <field_1>ID1234</field_1>
    <field_2>Smith</field_2>
    <field_3>John</field_3>
  </line>
  <line>
    <field_0>BC6789</field_0>
    <field_1>ID6789</field_1>
    <field_2>Jones</field_2>
    <field_3>Molly</field_3>
  </line>
</csvfile>
```

Step Two will be to call Transform with the “XMLXML” command to tell it to create final XML from the raw XML file, according to the instructions in the XSL style sheet.

Step Two

```
C:\FSC-Destiny\fsc\transform> Transform XMLXML demo_raw.xml style demopatrons.xml
```

Important: Do NOT include “.xsl” in the XSL filename.

Note: You can run Transform.bat with no parameters to see all command-line options.

The example XSL style sheet tells Transform how to generate the output XML from the input raw XML. The example XSL style sheet looks like:

style.xsl

```
<xsl:template match="/csvfile">
<FSC-Destiny-PatronImport version=“2”>
<xsl:for-each select="line">
<Patron action="addmodify">
    <Match>
      <DistrictID><xsl:value-of select= "field_1"/></DistrictID>
    </Match>
    <Barcode><xsl:value-of select= "field_0"/></Barcode>
    <LastName><xsl:value-of select= "field_2"/></LastName>
    <FirstName><xsl:value-of select= "field_3"/></FirstName>
</Patron>
</xsl:for-each>
</FSC-Destiny-PatronImport>
</xsl:template>
```

Line 1 instructs Transform to look at (match) all data included inside the <csvfile> element (between the opening <csvfile> and the closing </csvfile> tags in the input file).

Line 2 does not begin with “xsl:”, therefore it is sent directly to the output file.

Line 3 For each <line> in the input file, repeat the instructions inside the <xsl:for-each> element.

Line 4 outputs <Patron action=”addmodify”> for each <line> in the input file.

Line 5 begins the Match section by sending <Match> to the output file.

Line 6 sends <DistrictID>, then the value of <field_1> in the input file, then </DistrictID>.

Line 7 ends the Match section by sending </Match> to the output file.

Line 8 sends <Barcode>, then the value of <field_0> in the input file, then </Barcode>.

Lines 9 and 10 do the same for the LastName and FirstName fields.

Line 11 sends </Patron> to the output file.

Line 12 signifies the end of the <xsl:for-each> instruction.

Line 13 sends </FSC-Destiny-PatronImport> to the output file.

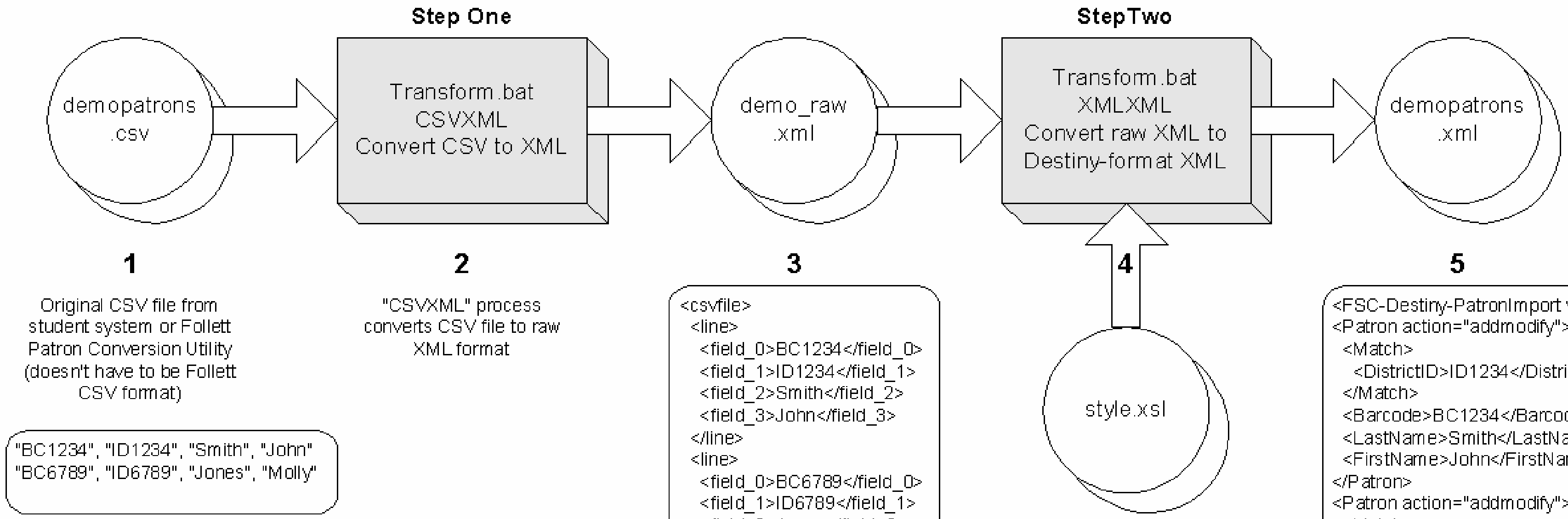
Line 14 signifies the end of the style sheet template.

The final output is an XML file that looks like:

demopatrons.xml

```
<FSC-Destiny-PatronImport>
<Patron action="addmodify">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
  <Barcode>BC1234</Barcode>
  <LastName>Smith</LastName>
  <FirstName>John</FirstName>
</Patron>
<Patron action="addmodify">
  <Match>
    <DistrictID>ID6789</DistrictID>
  </Match>
  <Barcode>BC6789</Barcode>
  <LastName>Jones</LastName>
  <FirstName>Molly</FirstName>
</Patron>
</FSC-Destiny-PatronImport>
```

Destiny Patron Data Conversion to XML


Step One
StepTwo
demopatrons .csv
Transform.bat CSVXML Convert CSV to XML
demo_raw .xml
Transform.bat XMLXML Convert raw XML to Destiny-format XML
demopatrons .xml
style.xsl
1
2
3
4
5

1

Original CSV file from student system or Follett Patron Conversion Utility (doesn't have to be Follett CSV format)

```
"BC1234", "ID1234", "Smith", "John"
"BC6789", "ID6789", "Jones", "Molly"
```

2

"CSVXML" process converts CSV file to raw XML format

3

```
<csvfile>
  <line>
    <field_0>BC1234</field_0>
    <field_1>ID1234</field_1>
    <field_2>Smith</field_2>
    <field_3>John</field_3>
  </line>
  <line>
    <field_0>BC6789</field_0>
    <field_1>ID6789</field_1>
    <field_2>Jones</field_2>
    <field_3>Molly</field_3>
  </line>
</csvfile>
```

4

"XMLXML" process uses XSL style sheet to convert raw to Destiny XML. This file is customized by each district.

FSCPatrons.xsl is provided as an example for converting from Follett's Circulation Plus format.

```
<xsl:template match="/csvfile">
<FSC-Destiny-PatronImport version="2">
  <xsl:for-each select="line">
    <Patron action="addmodify">
      <Match>
        <DistrictID><xsl:value-of select= "field_1"/></DistrictID>
      </Match>
    <Barcode><xsl:value-of select= "field_0"/></Barcode>
    <LastName><xsl:value-of select= "field_2"/></LastName>
    <FirstName><xsl:value-of select= "field_3"/></FirstName>
    </Patron>
  </xsl:for-each>
</FSC-Destiny-PatronImport>
</xsl:template>
```

5

```
<FSC-Destiny-PatronImport versio
<Patron action="addmodify">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
  <Barcode>BC1234</Barcode>
  <LastName>Smith</LastName>
  <FirstName>John</FirstName>
</Patron>
<Patron action="addmodify">
  <Match>
    <DistrictID>ID6789</DistrictID>
  </Match>
  <Barcode>BC6789</Barcode>
  <LastName>Jones</LastName>
  <FirstName>Molly</FirstName>
</Patron>
</FSC-Destiny-PatronImport>
```

Importing into Destiny

The final XML file can be imported into Destiny two ways. The first is by logging into Destiny as a district user with rights to edit patrons, in Back Office / Update Patrons.

The second is by using a command-line tool on the Destiny application server. This tool is named UpdatePatrons.bat, and has been placed in the \FSC-Destiny\fsc\bin directory. To run this batch file, pass it the name of the XML file to import:

```
C:\FSC-Destiny\fsc\bin> updatepatrons demopatrons.xml
```

Further XSLT Options

Of course, this simple example doesn't show every option available for converting XML. That is where the third-party books and tools can be your best resource. However, here are some tips for performing conversions common to Destiny users.

Test for field existing in input file

This XSL code only processes if the field is found in the input raw XML file.

```
<xsl:if test="field_5">
   ... (some code here)
</xsl:if>
```

Start a new line in the output file

This code sends a newline character to the output file.

```
<xsl:text>&#xA;</xsl:text>
```

Date conversion

This code converts a date from the standard Follett comma-delimited date format (YYYYMMDD) to the MM/DD/YYYY format required in Destiny XML.

```
<xsl:if test="string-length(field_5)=8">
<CardExpires>
  <xsl:value-of select="substring(field_5,5,2)"/>/<xsl:value-of
select="substring(field_5,7,2)"/>/<xsl:value-of select="substring(field_5,1,4)"/>
  </CardExpires>
</xsl:if>
```

Site name lookup

This code performs a lookup on cost center ID and converts it to a ShortSiteName.

```
<SiteShortName>
  <xsl:choose>
    <xsl:when test="field_14='0075'">Hawthorne</xsl:when>
    <xsl:when test="field_14='0275'">Washington</xsl:when>
    <xsl:when test="field_14='0344'">Watson</xsl:when>
    <xsl:when test="field_14='0976'">Lincoln</xsl:when>
    <xsl:otherwise>UNKNOWN</xsl:otherwise>
  </xsl:choose>
</SiteShortName>
```

Skip Follett-style header lines

This code just inside the for-each skips embedded Follett header lines in the input file.

```
<xsl:if test="field_1!='Patron Maintenance'">
```

Specifications for the Destiny Patron XML file

The Destiny Patron XML file must begin and end with these header/trailer lines:

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">

   Patron records go here...

</FSC-Destiny-PatronImport>
```

The patron records should all be included between the headers and the trailer. A patron record looks like this:

```
<Patron action="addmodify">
  <Match>
    <DistrictID>StudentID</DistrictID>
  </Match>
  ...
  Patron data fields go here...
  ...
</Patron>
```

Each patron record must include a <Match> section that includes either the <DistrictID> or both the <SiteShortName> and <Barcode> fields.

Following the <Match> section is a list of zero or more data fields for that particular patron (zero data fields if the action is “delete”). These data fields are used to set or overwrite existing values, not to match. Any or all fields in the patron record can be included here, including the fields used for matching. If a field is not included in this section, its existing value is preserved in the database. If a field is included in this section but has no content, the field is cleared in the database.

Example 1: Match on DistrictID. Update patron name and address.

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">

<Patron action="addmodify">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
  <LastName>Smith</LastName>
  <FirstName>John</FirstName>
  <AddressPrimaryLine1>1722 Ringwood Road</AddressPrimaryLine1>
  <AddressPrimaryCity>Chicago</AddressPrimaryCity>
  <AddressPrimaryState>IL</AddressPrimaryState>
  <AddressPrimaryZipCode>60611</AddressPrimaryZipCode>
</Patron>
...
<Patron action="addmodify">  (additional patron records)
</Patron>
...
</FSC-Destiny-PatronImport>
```

Example 2: Match on SiteShortName and Barcode. Update name and address; clear the email addresses.

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">

<Patron action="addmodify">
  <Match>
    <SiteShortName>Hill-Elem</SiteShortName>
    <Barcode>1234567</Barcode>
  </Match>
  <LastName>Smith</LastName>
  <FirstName>John</FirstName>
  <AddressPrimaryLine1>1722 Ringwood Road</AddressPrimaryLine1>
  <AddressPrimaryCity>Chicago</AddressPrimaryCity>
  <AddressPrimaryState>IL</AddressPrimaryState>
  <AddressPrimaryZipCode>60611</AddressPrimaryZipCode>
  <EmailPrimary></EmailPrimary>
  <EmailSecondary></EmailSecondary>
</Patron>
...
<Patron action="addmodify">  (additional patron records)
</Patron>
...
</FSC-Destiny-PatronImport>
```

Example 3: Match on DistrictID and transfer to a new school

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">

<Patron action="addmodify">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
  <SiteShortName>MyNewSchool</SiteShortName>
</Patron>
...
<Patron action="addmodify">  (additional patron records)
</Patron>
...
</FSC-Destiny-PatronImport>
```

Example 4: Match on DistrictID and delete a set of patron records

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">

<Patron action="delete">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
</Patron>
<Patron action="delete">
  <Match type="DistrictID">
    <DistrictID>ID7777</DistrictID>
  </Match>
</Patron>
</FSC-Destiny-PatronImport>
```

Fields in the Destiny Patron XML File

Patron Fields	Max Length	Comments
<Match> </Match>		Required section at start of patron record. Must contain either DistrictID or both SiteShortName and Barcode fields.
Barcode	14	Required after the Match section when adding a patron. Unique within a school. Value of “autoassign” causes Destiny to assign the next available barcode number to the patron. Cannot use “autoassign” inside a <Match></Match> section.
SiteShortName	10	[District only] Required after the Match section when adding a patron. Must match site short name entered in district configuration.
DistrictID	40	Optional. Unique within the district.
LastName	30	Required
FirstName	30	
MiddleName	30	
Gender	1	M, F, U – Defaults to U if not specified.
PatronType	30	If not specified, default will be used.
AccessLevel	30	If not specified, default will be used.
Status	1	A, I or R – Active, Inactive or Restricted
CardExpires	10	MM/DD/YYYY
UserDefined1	30	
UserDefined2	30	
UserDefined3	30	
UserDefined4	30	
UserDefined5	30	
GraduationYear	4	YYYY
BirthDate	10	MM/DD/YYYY
UserName	10	Optional. Unique within the district.
Password	20	
EmailPrimary	30	
EmailSecondary	30	
AddressPrimaryLine1	30	
AddressPrimaryLine2	30	
AddressPrimaryCity	30	
AddressPrimaryState	30	
AddressPrimaryZipCode	20	
AddressPrimaryPhoneNumberPrimary	30	
AddressPrimaryPhoneNumberSecondary	30	
AddressSecondaryLine1	30	
AddressSecondaryLine2	30	
AddressSecondaryCity	30	
AddressSecondaryState	30	
AddressSecondaryZipCode	20	
AddressSecondaryPhoneNumberPrimary	30	
AddressSecondaryPhoneNumberSecondary	30	
IsTeacher	5	true/false
GradeLevel	30	
Homeroom	90	
Nickname	30	
IsAcceptableUsePolicyOnFile	5	true/false

Example of a Destiny Patron XML File

```
<?xml version="1.0" encoding="UTF-8"?>
<FSC-Destiny-PatronImport version="2">
<Patron action="addmodify">
  <Match>
    <DistrictID>ID2312</DistrictID>
  </Match>
  <DistrictID>ID2312</DistrictID>
  <SiteShortName>Hawthorne</SiteShortName>
  <Barcode>Autoassign</Barcode>
  <LastName>Chwalisz</LastName>
  <FirstName>David</FirstName>
  <MiddleName>A</MiddleName>
  <Gender>M</Gender>
  <PatronType>Student</PatronType>
  <AccessLevel>Patron</AccessLevel>
  <Status>Active</Status>
  <CardExpires>12/31/2004</CardExpires>
  <UserDefined1>UD1</UserDefined1>
  <UserDefined2>UD2</UserDefined2>
  <UserDefined3>UD3</UserDefined3>
  <UserDefined4>UD4</UserDefined4>
  <UserDefined5>UD5</UserDefined5>
  <GraduationYear>2004</GraduationYear>
  <BirthDate>01/02/1974</BirthDate>
  <UserName>Freddy</UserName>
  <Password>myPassword</Password>
  <EmailPrimary>Joe.Student@anyschool.com</EmailPrimary>
  <EmailSecondary>Alternate.Student@anyschool.com</EmailSecondary>
  <AddressPrimaryLine1>123 Maple Ave</AddressPrimaryLine1>
  <AddressPrimaryLine2>Apt 200</AddressPrimaryLine2>
  <AddressPrimaryCity>Chicago</AddressPrimaryCity>
  <AddressPrimaryState>IL</AddressPrimaryState>
  <AddressPrimaryZipCode>60611</AddressPrimaryZipCode>
  <AddressPrimaryPhoneNumberPrimary>(312) 555-1234
    </AddressPrimaryPhoneNumberPrimary>
  <AddressPrimaryPhoneNumberSecondary>(312) 555-7777
    </AddressPrimaryPhoneNumberSecondary>
  <AddressSecondaryLine1>1st Line Again</AddressSecondaryLine1>
  <AddressSecondaryLine2>2nd Line Again</AddressSecondaryLine2>
  <AddressSecondaryCity>San Francisco</AddressSecondaryCity>
  <AddressSecondaryState>CA</AddressSecondaryState>
  <AddressSecondaryZipCode>90421</AddressSecondaryZipCode>
  <AddressSecondaryPhoneNumberPrimary>(815) 555-1212
    </AddressSecondaryPhoneNumberPrimary>
  <AddressSecondaryPhoneNumberSecondary>(815) 555-1111
    </AddressSecondaryPhoneNumberSecondary>
  <IsTeacher>false</IsTeacher>
  <GradeLevel>Senior</GradeLevel>
  <Homeroom>Mr. Smith- room 100</Homeroom>
  <Nickname>Dave</Nickname>
  <IsAcceptableUsePolicyOnFile>true</IsAcceptableUsePolicyOnFile>
</Patron>

<Patron action="delete">
  <Match>
    <DistrictID>ID1234</DistrictID>
  </Match>
</Patron>
</FSC-Destiny-PatronImport>
```