

XC MARCXML Normalization Service Documentation

Version 0.1

XC MARCXML Normalization Service Documentation

© eXtensible Catalog

University of Rochester

1325 Mt. Hope, Suite 102

PO Box 278960

Rochester, NY 14627-8960

<http://www.extensiblecatalog.org>

**Introduction**

The eXtensible Catalog’s MARCXML Normalization Service cleans up MARCXML records by fixing common problems and inconsistencies. The service also prepares MARCXML metadata for use in a faceted-browse interface by replacing coded data with vocabulary terms and parsing this data into locally-defined 9XX fields for easy access. It is run as a series of “Normalization Steps” which can be enabled or disabled using the Normalization Service’s configuration files. This document contains information on how to configure the Normalization Service and what action is taken by each Normalization Step. The description of each normalization step also contains a brief rationale for why the step is needed to prepare MARCXML metadata for use in a user interface that incorporates faceted browsing.

Currently, this service contains steps to normalize data in either MARCXML bibliographic or holdings records. The service looks at the Leader 06 in MARCXML record to determine the type of record (bibliographic or holdings) and then applies the steps appropriate to that type of record. MARCXML record types other than bibliographic and holdings are passed through the services with no changes. However, additional steps could be added easily to this service so that it would also normalize records from other types of MARC records, e.g. authority records, if desired.

**Configuration**

Before actually configuring the Normalization Service, the reader is expected to have read the sections of the [Metadata Services Toolkit User manual](http://code.google.com/p/xcmetadataservicestoolkit/downloads/list) that explain how to set up the MST to harvest records and pass them into Metadata Services.

The Normalization Service is configured by a properties file in the MST-instances/MetadataServicesToolkit/services/Normalization/serviceConfig directory. This properties file contains information on which Normalization Steps the Normalization Service should run and which ones it should skip. This file contains several lines of the format:

<normalization\_step\_name> = 1

Where <normalization\_step\_name> is the name of the step and ‘1’ indicates that the step is enabled (set this value to ‘0’ to disable a step and ‘1’ to enable it.)

Also it contains sections that define the properties used by step. The sections are defined like

#-----------------------------------------

<Section name>

#-----------------------------------------

The following section describes the actions taken by each Normalization Step, the step’s purpose, and any additional properties files it requires.

**Important Note:**

It is possible to change the configuration of steps in the Normalization Service by turning them on and off. However, adding new steps to the service and changing the definition of existing steps requires changes to the configuration file as well as to the Normalization JAR file.

W**henever a change is made to any of these, all data must be reprocessed through this service and through all succeeding MST services to enable the OAI-PMH processing of updated records to continue to be handled automatically.**

We advise libraries who wish to make changes to the Normalization Service (or to any other MST metadata service) to collect several changes and implement them all at the same time, thereby reducing the number of times that the data will need to be reprocessed.

**Description of MARCXML Bibliographic Normalization Steps**

Each normalization step will either modify the data in the original MARC tag, as indicated, or will create a 9XX field to contain the normalized data. If a 9XX field is created, the Normalization Service will supply a subfield 5 ($5) within the 9XX tag that contains the organization’s code from the MARC Code List for Organizations (<http://www.loc.gov/marc/organizations/orgshome.html>). The organization code is configured in the in the Normalization Services Configuration File. (It is located in the “Enabled Steps” section of the config file, under the step, “SupplyMARCOrgCode). The $5 will enable the newly-created 9XX field to be distinguished from other instances of the same MARC tag that might already be present in an institution’s MARC data.

The description of each step specifies the purpose for the step, the section name(if any) in the configuration file which contains the properties used by the step , and the default setting for the step, i.e. whether the step is “on” or “off” in the default “out of the box” version of the service. Services that are likely to be specific to a particular ILS or institution (such as those defined for the University of Rochester) are defaulted to “off”.

**Name: RemoveOCoLC003**

**Properties section name: none**

**Default setting: Off**

**Purpose:** If the filed 003 has a value of "OCoLC", delete it.

This step is designed to eliminate inaccurate 003 information found in legacy University of Rochester data.

**Name: DCMIType06**

**Properties section name: LEADER 06 TO DCMI TYPE Default setting: On**

**Purpose:** Converts the code in the bibliographic Leader 06 to the Dublin Core Type vocabulary, based loosely on the conversion in Appendix 2 in the Library of Congress’ MARC to DC mapping (<http://www.loc.gov/marc/marc2dc.html>). Using this vocabulary for formerly-MARC data may facilitate integrating MARC and DC records into the same discovery application.

This step adds a 931 field containing the DCMI Type for the record as determined by that record’s Leader 06 and 006 offset 00 values. The leader06ToDcmiType.properties file contains a mapping from these values to the DCMI Type that should be used. For example, the line “a = Text” tells the Normalization Service to add a DCMI Type of “Text” for each record whose Leader 06 or 006 offset 00 is ‘a’. The contents of this file may be modified to associate any value with any DCMI Type, however a single value may not be associated with more than one DCMI Type. If there is not a row for a given Leader 06 or 006 offset 00 value, a DCMI Type will not be added for records with that value and no 931 field is created

**Name: Leader06Vocab**

**Properties section name: LEADER 06 TO MARC VOCAB**

**Default setting: On**

**Purpose:** Replaces the MARC Leader 06 code with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 932 field containing the MARC Leader 06 Vocabulary Term for the record as determined by that record’s Leader 06 and 006 offset 00 values. The leader06ToMarcVocab.properties file contains a mapping from these values to the MARC Leader 06 Vocabulary that should be used. For example, the line “a = Language material” tells the Normalization Service to add a term of “Language Material” for each record whose Leader 06 or 006 offset 00 is ‘a’. The contents of this file may be modified to associate any value with any MARC Leader 06 Vocabulary Term, however a single value may not be associated with more than one term. If there is not a row for a given Leader 06 or 006 offset 00 value, a term will not be added for records with that value and no 932 field is created.

**Name: 007Vocab06**

**Properties section name: LEADER 06 TO FULL TYPE**

**Default setting: On**

**Purpose:** generates a term from the MARC 007/00 vocabulary based upon the MARC Leader 06 to ensure that this data element is populated and can be used as part of a facet definition even when the original MARC record does not contain an 007 field.

This step adds a 933 field containing the MARC 007 Vocabulary term for the record as determined by that record’s Leader 06 value. The leader06ToFullType.properties file contains a mapping from the Leader 06 values to the MARC 007 Vocabulary term that should be used. For example, the line “a = Text” tells the Normalization Service to add a term of “Text” for each record whose Leader 06 is ‘a’. The contents of this file may be modified to associate any Leader 06 value with any MARC 007 Vocabulary term however a single value may not be associated with more than one term. If there is not a row for a given Leader 06 value, a term will not be added for records with that Leader 06 and no 933 field is created.

**Name: ModeOfIssuance**

**Properties section name: LEADER 07 TO MODE OF ISSUANCE**

**Default setting: On**

**Purpose:** Replaces the MARC Leader 07 code with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 935 field containing the Mode of Issuance for the record as determined by that record’s Leader 07 value. The leader07ToModeOfIssuance.properties file contains a mapping from the Leader 07 values to the Mode of Issuance that should be used. For example, the line “c = Collection” tells the Normalization Service to add the Mode of Issuance “Collection” for each record whose Leader 07 is ‘c’. The contents of this file may be modified to associate any Leader 07 value with any Mode of Issuance, however a Leader 07 may not be associated with more than one Mode of Issuance. If there is not a row for a given Leader 07 value, a Mode of Issuance will not be added for records with that Leader 07 and no 935 field is created.

**Name: MoveMARCOrgCode**

**Properties section name: none**

**Default setting: On**

**Purpose:** Creates a valid 035 field that can then be mapped to the XC Schema property, “recordID” by the XC MARC Transformation Service.

This step creates a new 035 field from the 001 and 003 control field. If both the 001 and 003 fields exist, a new 035 will be added to the record which contains the prefix from the 003 and the control number from the 001. The MoveMARCOrgCode\_moveAll step further defines the behavior of this step. If MoveMARCOrgCode\_moveAll is set to 0, a new 035 element will only be created if the 003 matches the organization’s Organization Code. If this property is 1, a new 035 will be added whenever 003 and 001 exist. The default setting for the MoveMARCOrdCode\_moveALL is 0.

**Name: DCMIType007**

**Properties section name: FIELD 007 OFFSET 00 TO DCMI TYPE**

**Default setting: On**

**Purpose:** Converts the code in the 007/00 to the Dublin Core Type vocabulary when such a mapping is possible. Using this vocabulary for formerly-MARC data may facilitate integrating MARC and DC records into the same discovery application.

This step adds a 931 field containing the DCMI Type for the record as determined by that record’s 007 offset 00 values. The section ‘FIELD 007 OFFSET 00 TO DCMI TYPE’ in configuration file contains mapping from these values to the DCMI Type that should be used. For example, the line “a = Image” tells the Normalization Service to add a DCMI Type of “Image” for each record whose 007 offset 00 is ‘a’. The contents of this file may be modified to associate any value with any DCMI Type, however a single value may not be associated with more than one DCMI Type. If there is not a row for a given 007 offset 00 value, a DCMI Type will not be added for records with that value and no 931 field is created.

**Name: 007Vocab**

**Properties section name: FIELD 007 OFFSET 00 TO FULL TYPE**

**Default setting: On**

**Purpose:** Replaces the MARC 007/00 code with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 933 field containing the MARC 007 Vocabulary term for the record as determined by that record’s 007 offset 00. The section ‘FIELD 007 OFFSET 00 TO FULL TYPE’ in configuration file contains a mapping from the 007 offset 00 values to the MARC 007 Vocabulary term that should be used. For example, the line “a = Map” tells the Normalization Service to add a term of “Map” for each record whose 007 offset 00is ‘a’. The contents of this file may be modified to associate any 007 offset 00 with any MARC 007 Vocabulary term however a single value may not be associated with more than one term. If there is not a row for a given 007 offset 00 value, a term will not be added for records with that 007 offset 00and no 933 field is created.

**Name: 007SMDVocab**

**Properties section name: FIELD 007 OFFSET 00 TO SMD TYPE**

**Default setting: On**

**Purpose:** Replaces the MARC 007/01 code with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 934 field containing the SMD Type for the record as determined by that record’s 007 offset 00-01. The section ‘FIELD 007 OFFSET 00 TO SMD TYPE’ in configurationfile contains a mapping from the 007 offsets 00 and 01 to the SMD Type that should be used. For example, the line “ad = Atlas” tells the Normalization Service to add a term of “Atlas” for each record whose 007 offset 00 is ‘a’ and 007 offset 01 is ‘d’. The contents of this file may be modified to associate any 007 offset 00-01 value with any SMD Type however a single pair may not be associated with more than one term. If there is not a row for a given 007 offset 00-01 value, a term will not be added for records with that 007 offset 00-01 value and no 934 field is created.

**Name: FictionOrNonfiction**

**Properties section name: none**

**Default setting: On**

**Purpose:** Replaces the MARC code as identified below with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 937 field with a value of either “Fiction” or “Non-Fiction.” All records with a Leader 06 of ‘a’ and an 008 offset 33 of ‘1’ will receive a 937 of “Fiction,” and all others will receive a 937 of “Non-Fiction.”

**Name: 008DateRange**

**Properties section name: none**

**Default setting: On**

**Purpose:** Normalizes the dates in the 008 when a range of dates is present to make them more understandable to users.

If the 008 offset 06 is either ‘c’, ‘d’, or ‘k’, this step copies the 008 offset 07-10 followed by a hyphen followed by the 008 offset 11-14 into a 939 field. If the resulting field contains “9999” as one of the dates it is replaced with four blank spaces. If the 008 offset 06 contains any other value, no 939 field is created.

**Name: 008Thesis**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates a possible discrepancy in MARC data between the 008 and the presence of a note.

If the record does not have a 502 field and has a Leader 06 of ‘a’ and there is an ‘m’ somewhere in the 008 offset 24-27, this step adds a 502 field with a value of “Thesis.” If these conditions do not apply, no 502 field is created.

**Name: ISBNCleanup**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates extraneous data in the 020 that is not part of the ISBN itself, to facilitate using the ISBN as an identifier.

This step copies the contents of an 020 field into a 947 field up to the first left parenthesis or colon. If no 020 field is present, no 947 field is created.

**Name: LanguageSplit**

**Properties section name: none**

**Default setting: On**

**Purpose:** Separates each language code (excluding original language, if a translation; and language of accompanying materials) so that it can be manipulated individually.

This step creates a 941 field for each language code found in the original record. Language codes are taken from the 008 offset 35-37 and from the 041 $a and $d subfields. Language codes of “mul”, “N/A”, “und”, “ZXX” and “XXX” are ignored, and no 941 is created for them.

**Name: LanguageTerm**

**Properties section name: LANGUAGE CODE TO LANGUAGE**

**Default setting: On**

**Purpose:** Replaces the MARC language code with the corresponding term from the MARC 21 standard so that the term (rather than the code) can be displayed to users.

This step adds a 943 field containing the language for each language code from the 941 fields. The section ‘LANGUAGE CODE TO LANGUAGE’ in configuration file contains a mapping from the 941 values to the language that should be used. For example, the line “eng = English” tells the Normalization Service to add a language of “English” for each record with a 941 of “eng”. The contents of this file may be modified to associate any language code with any language, however a single language code may not be associated with more than one language. If there is not a row for a language code, a language will not be added for records with that language code and no 941 field is created.

**Name: 008Audience**

**Properties section name: FIELD 008 OFFSET 22 TO AUDIENCE**

**Default setting: On**

**Purpose:** Enables additional data from the MARC Leader related to the intended audience of the resource to be displayed in a form that is understandable to users.

If the record’s leader 06 is either ‘a’, ‘c’, ‘d’, ‘g’, ‘k’, ‘m’, ‘o’, or ‘r’ this step supplies a 945 with the target audience based on the record’s 008 offset 22. Section ‘FIELD 008 OFFSET 22 TO AUDIENCE‘ in configuration file contains a mapping from the 008 offset 22 to the audience that should be used. For example, the line “a = Preschool” tells the Normalization Service to add an audience of “Preschool” for each record whose 008 offset 22 is ‘a’. The contents of this file may be modified to associate any 008 offset 22 with any audience, however a single 008 offset 22 may not be associated with more than one audience. If there is not a row for a given 008 offset 22 value, an audience will not be added for records with that value and no 945 field is created.

**Name: SupplyMARCOrgCode**

**Properties section name: none**

**Default setting: On**

**Purpose:** Creates an 035 with the organization’s code if an 003 field is lacking in the incoming record. This step must be used when the XC OAI Toolkit is not used and the incoming MARCXML record does not contain an 003.

If the record has no 003 field, this step moves the 001 into a new 035 field and supplies the organization code as the prefix for this 035. If this step is enabled, the library should replace the University of Rochester’s organization code (NRU) with the library’s own organization’s code from the MARC Code List for Organizations (<http://www.loc.gov/marc/organizations/orgshome.html>) in the Normalization Services Configuration File. (The code is located in the “Enabled Steps” section of the config file, under the step, “SupplyMARCOrgCode).

**Name: Fix035**

**Properties section name: none**

**Default setting: On**

**Purpose:** Normalizes the various formats of OCLC numbers in 035 fields by addressing commonly-found discrepancies in MARCXML records.

This step corrects all malformed OCLC 035 fields, giving them the format “(OCoLC)<control\_number>”. The following malformed 035 formats are corrected:

* 035 $b ocm $a <control\_number>
* 035 $b ocn $a <control\_number>
* 035 $b ocl $a <control\_number>
* 035 $a (OCoLC)ocm<control\_number>
* 035 $a (OCoLC)ocn<control\_number>
* 035 $a (OCoLC)ocl<control\_number>
* 035 $9 ocm<control\_number>
* 035 $9 ocn<control\_number>
* 035 $9 ocl<control\_number>

**Name: 035LeadingZero**

**Properties section name: none**

**Default setting: On**

**Purpose:** Remove leading zeros in 035 numeric identifiers to facilitate matching.

If the numeric portion of an 035 field (after the prefix) begins with one or more zeros, remove the zeros.

**Name: Fix035Code9**

**Properties section name: none**

**Default setting: Off**

**Purpose:** This is to fix the OCLC microform records.

If the contents of 035 $9 begin with the letters "ocm", move the number to 035 $a after the prefix ‘(OCoLC)’ to match the format used in step FixO35.

**Name: Dedup035**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates possible redundancies between 035 fields.

If there are multiple 035 fields with the same value, this step deletes all but one of the duplicate 035s.

**Name: RoleAuthor**

**Properties section name: none**

**Default setting: On**

**Purpose:** Enables this data to be mapped to the RDA role, “author” instead of to the more generic “creator”. This more specific role may then be used to populate a facet value in a discovery application.

If a record contains Leader 06 value “a” and contains a 100, 110, or 111 field that does not contain a $4, this step adds a $4 subfield with a value of “aut” to the 100, 110, or 111 field. .

**Name: RoleComposer**

**Properties section name: none**

**Default setting: On**

**Purpose:** Enables this data to be mapped to the RDA role, “composer” instead of to the more generic “creator”. This more specific role may then be used to populate a facet value in a discovery application.

If a record contains Leader 06 value “c” and contains a 100, 110 or 111 field that does not contain a $r, this step adds a $4 subfield with a value of “cmp” to the 100, 110, or 111 field.

**Name: UniformTitle**

**Properties section name: none**

**Default setting: On**

**Purpose:** Creates a uniform title when one is not present in the record (perhaps because that uniform title would have been identical to the 245 $a). This uniform title will be used as the basis for work and expression titles in the XC Schema.

If the record does not contain a 130, 240 or 243 and does contain a 245, this step copies the 245 $a, $n, $p, $k, and $f subfields into a 240 which it creates. The new 240 will have the same indicators as the 245 it was created from.

**Name: NRUGenre**

**Properties section name: none**

**Default setting: Off**

**Purpose:** Standardizes the data in University of Rochester (NRU) records for local genre terms.

For each 655 field with a $2 of “local” and a $5 of “NRU,” this step changes the $2’s value to “NRUgenre” and deletes the $5 subfield.

**Name: TopicSplit**

**Properties section name: none**

**Default setting: On**

**Purpose:** Separates subject data that reflects topics from multiple 6XX fields into one or more fields that can be used as the basis of facets in a discovery application.

This step copies all subfields up to the first $v, $y, or $z subfield from the 600, 610, 611, 630, and 650 datafields into separate 965 fields, retaining the original subfield codes and indicators. It also copies all subfields up to the first $x subfield from any 6XX field except 656, 657, 658, and 662 into separate 965 fields.

**Name: ChronSplit**

**Properties section name: none**

**Default setting: On**

**Purpose:** Separates subject data that reflects chronological periods from multiple 6XX fields into one or more fields that can be used as the basis of a facet in a discovery application.

This step copies all subfields up to the first $v, $x, or $z subfield from the 648 data field into separate 963 fields, retaining the original subfield codes and indicators. It also copies all subfields up to the first $y subfield from any 6XX field except 656, 657, 658, and 662 into separate 963 fields.

**Name: GeogSplit**

**Properties section name: none**

**Default setting: On**

**Purpose:** Separates subject data that reflects geographical areas from multiple 6XX fields into one or more fields that can be used as the basis of a facet in a discovery application.

This step copies all subfields up to the first $v, $x, or $y subfield from the 651 datafields into separate 967 fields, retaining the original subfield codes and indicators. It also copies all subfields up to the first $z subfield from any 6XX field except 656, 657, 658, and 662into separate 967 fields.

**Name: GenreSplit**

**Properties section name: none**

**Default setting: On**

**Purpose:** Separates subject data that reflects genre from multiple 6XX fields into one or more fields that can be used as the basis of a facet in a discovery application.

This step copies all subfields up to the first $x, $y, or $z subfield from the 655 data field into separate 969 fields, retaining the original subfield codes and indicators. It also copies all subfields up to the first $v subfield from any 6XX field except 656, 657, 658, and 662 into separate 969 fields.

**Name: DedupDCMIType**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates redundant data.

If the Normalization Service created more than one 931 field with the same value, this step deletes all but one of the redundant fields.

**Name: Dedup007Vocab**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates redundant data.

If the Normalization Service created more than one 933 field with the same value, this step deletes all but one of the redundant fields.

**Name: BibLocationName**

**Properties section name: LOCATION CODE TO LOCATION**

**Default setting: Off**

**Purpose:** Replaces the location code with a location name so that it can be displayed in a form that is understandable to users.

This step replaces the location code from the 852 $b with the name of the location. The section ‘LOCATION CODE TO LOCATION‘ in the configuration file contains a mapping from location codes to the location they represent. For example, the line “RareRev = Preservation/Rare Books” tells the Normalization Service to use the location “Preservation/Rare Books” in place of the location code “RareRev”. To use this step, a library must replace the sample data in the configuration file with its own location codes and location names. A single code may not be associated with more than location. If there is not a row for a given location code, that code will not be replaced with a location. This step defaults to “off” because the XC Drupal Toolkit will also enable location codes to be replaced by longer location names.

**Name: IIILocationName**

**Properties section name: LOCATION CODE TO LOCATION**

**Default setting: Off**

**Purpose:** This is a placeholder step to enable libraries that use III to map their location data.

This step replaces the location code in 945 $l with name of location from the section ‘LOCATION CODE TO LOCATION’ in the configuration file

**Name: SeparateName**

**Properties section name: none**

**Default setting: On**

**Purpose:** Positions the “name” portion of a name/title heading in a separate field so that it can be matched against an authority file on its own. Currently the service step is defined only to handle name/title analytics, which will be mapped to separate records in the XC Schema, and not to handle all name/title headings.

For each 700, 710 or 711 with a 2nd indicator of 2, this step copies all data in the MARC field up to but not including the $t subfield into a 959 field. An $8 linking field is then added to the original field and the resulting 959 in order to associate the two.

**Name: Dedup9XX**

**Properties section name: none**

**Default setting: On**

**Purpose:** Eliminates redundant data.

If there are multiple 963, 965, 967, 969, or 959 fields with the same contents and the same 2nd indicator, this step removes all but one of the redundant fields. This step ignores $8 linking fields, subfield codes, 1st indicators, and trailing periods to determine whether or not two fields are duplicates. If a field with a $8 linking field is removed by this step, that linking field is added to the duplicate field which was not removed to ensure that all links are maintained correctly.

**Name: TitleArticle**

**Properties section name: none**

**Default setting: On**

**Purpose:** This step will create a 246 field without an initial article whenever a 245 exists with an initial article. The 246 will be mapped to alternative title, and can be used to facilitate title sorting in XC.

If the 245 1st indicator is 1 and the 245 2nd indicator is not zero, this step ignores the number of characters from the beginning of the field that the 2nd indicator specifies and copies the remainder of the 245 $anpfk to a new 246 field, keeping the same subfields ($anpf). The indicators for the new 246 field are set as 3 and 0. The first character that was mapped is then replaced with a capital letter, if it is lower-case. If this processing step has created a 246 with identical content to a previously-existing 246, the 246 is removed.

**Description of MARCXML Holdings Normalization Steps**

**Name: HoldingsLocationName**

**Properties section name: LOCATION CODE TO LOCATION**

**Default setting: Off**

**Purpose:** Supplies a location name for a location code so that it can be displayed in a form that is understandable to users.

This step gets the location name from the configuration file for the location code in 852 $b and inserts it in 852 $c. The section ‘LOCATION CODE TO LOCATION‘ in the configuration file contains sample mapping for location codes to the location they represent. These should be deleted and replaced with the Institution’s actual location code mappings if this step is used. For example, the line “RareRev = Preservation/Rare Books” tells the Normalization Service to use the location “Preservation/Rare Books” for the location code “RareRev”. A single code may not be associated with more than location. If there is not a row for a given location code, that code will not be replaced with a location.

**Name: LocationLimitName**

**Properties section name: LOCATION CODE TO LOCATION LIMIT NAME**

**Default setting: Off**

**Purpose:** Groups specific location codes together into a smaller number of general Location Limit Names to facilitate the creation of broad facet values for location.

Replaces the location code in 852 $b with the corresponding location limit name from the ‘LOCATION CODE TO LOCATION LIMIT NAME’ section of theconfiguration file. The default configuration file contains sample mappings for location codes to general location limit names. These should be deleted and replaced with the Institution’s actual location codes and mappings if this step is used. For example, the line “RareRev = Rare Books” tells the Normalization Service to use the location limit “Rare Books” for the location code “RareRev”. A single code may not be associated with more than one location limit. If there is not a row for a given location code, that code will not be replaced with a location limit.