

Appendix: Preparing Tab-delimited Accession Files

Importing tab-delimited accessions data

The Toolkit will not import accession information straight from a database, such as Access or Filemaker Pro. To transfer accessions data to the Toolkit, a tab-delimited file must be exported from the database containing the information. This tab-delimited file can then be imported into the Toolkit. Before importing data into the Toolkit, the data must be mapped to the fields in the Toolkit. Some mapping tips are listed below.

Basic steps for importing tab-delimited accessions data:

1. Map the fields in the accessions database (or spreadsheet if that is where the data is stored) to the corresponding AT fields listed in the table below. If some data fields do not match AT fields, you may want to consider implementing a user-defined field. See Chapter 15 for more information on user-defined fields.
2. Format your data to meet the constraints listed below. This step may also be done after Step 3. The decision of when to adjust the data should be done after an evaluation of the requirements for data import (listed throughout the rest of this chapter) and a comparison of those requirements to how the data is currently stored in the database (or spreadsheet). In some instances, it may be more efficient to adjust the data while it is still in database form. In others, it may be quicker after the database has been exported to a tab-delimited file.
3. Export a tab-delimited, unformatted text document out of the accessions database (or spreadsheet).
4. If not done in Step 2, format the data according to the constraints listed below.
5. Import the tab-limited file according to the instructions listed in Chapter 6.

Constraints on legacy accessions data

There are constraints on the format of tab-delimited data to be imported that may require cleanup of your legacy data prior to import. Not satisfying these constraints will cause the accession import process to fail.

Formatting the file:

1. Export a tab delimited, unformatted text document out of your database (or spreadsheet). All tabs, line breaks, and hard returns within your data should be removed. This will help reduce the need for data cleanup later. It is optional, but may be helpful, to map the AT headings to your database fields, and then replace the previous headings with the AT

headings within the database prior to export. This will bypass the need to input the header line in Step 2.

2. The first line has to be a header line that contains information about what is in the column, generally the field name. If this was not exported into the tab-delimited file, it needs to be created. Names of the column headers must match the Import Field/Column Label entries listed in the table below; if your database application exported headers that match field names in your database, these names need to be converted to the entries listed in the table below. The column headers are case-sensitive.

The following shows a sample import file in tab-delimited format. Every space depicted between field values (where “Linear feet” is one field value and “Fales Family Papers” is another) is a single tab, and the first row is the field/column header. In this example, the accessionNumber for the first entry is 1957.029 and the title is Fales Family Papers:

```
resourceIdentifier accessionDate accessionNumber extentNumber extentType title
MSS 029 1957-01-01 1957.029 45 Linear feet Fales Family Papers MSS 002 1964-01-
01 1964.002 126 Linear feet The Elizabeth Robins Papers
```

3. If Excel is used to create a tab-delimited file, then the last column in the file needs to contain data for every single accession entry--none of the cells in the last column can be left blank. This helps guarantee the information will contain the correct number of tabs and thus be parsed correctly, otherwise the empty field may be parsed as a tab, and the data will import incorrectly. Excel will also add double quotes around any characters that can be interpreted as delimiters, such as commas. If these quotes are not stripped from the data, either manually or through find-and-replace mechanisms, they will be imported into the Toolkit.
4. Hard returns and other formatting within the cells of the table should be removed. Line breaks or hard returns in the data may be replaced by the pipe character (|) and those pipe characters will be converted to line breaks in the Toolkit if the target field has a free-text data type.
5. If you have a single field in the database that needs to get mapped into more than one field in the Toolkit, you will need to parse the data into two separate fields. For example, in many cases accessions data will combine extent into one field, such as “11 linear feet.” The Toolkit provides two separate fields for extent: **extentNumber** and **extentType**. Extent number contains the numerical value of the extent (as a real number) and extent measurement contains the extent measurement type of the accession, such as linear feet. Other fields that may need to be parsed include name fields, etc.
6. Boolean fields (cataloged, restrictionsApply, rights) may only contain the values true or false.
7. Integer fields (dateBegin, dateEnd) can only contain whole numbers.
8. Real number fields (extentNumber) can contain integers and decimals.

Field / Column Header Mapping

Preparing the Tab-delimited Accession file

The following table shows the allowed field names and the target fields for accession data to be imported into the Toolkit. Some fields have controlled vocabularies, and those are listed in the **Rule** column. The **Field Length** is the maximum number of characters allowable for each field. Importing more characters than allowed will result in data truncation.

Accession Fields / Column Headers				
<u>Import Field/Column Header</u>	<u>Where Stored /AT Target</u>	<u>Field Length</u>	<u>Data Type</u>	<u>Rule</u>
accessionNumber			text	parsed by delimiters into: accessionNumber, accessionNumber2 accessionNumber3 accessionNumber4
accessionNumber1	accessionNumber1	10	string	
accessionNumber2	accessionNumber2	10	string	
accessionNumber3	accessionNumber3	10	string	
accessionNumber4	accessionNumber4	10	string	
accessionDate	accessionDate		date	
description	description	no limit	text	
accessionDispositionNote	accessionDispositionNote	no limit	text	
accessRestrictions	accessRestrictions		boolean	
accessRestrictionsNote	accessRestrictionsNote	no limit	text	
useRestrictions	useRestrictions		boolean	
useRestrictionsNote	useRestrictionsNote	no limit	text	
generalAccessionNote	generalAccessionNote	no limit	text	
acknowledgementDate	acknowledgementDate		date	
acknowledgementSent	acknowledgementSent		boolean	
agreementReceivedDate	agreementReceivedDate		date	
agreementSentDate	agreementSentDate		date	
agreementSent	agreementSent		boolean	
agreementReceived	agreementReceived		boolean	
agreementProcessed	accessionProcessed		boolean	
accessionProcessedDate	accessionProcessedDate		date	
acquisitionType	acquisitionType	255	string	Deposit, Gift, Purchase, Transfer
dateExpression	dateExpression	255	string	
bulkDateBegin	bulkDateBegin		integer	
bulkDateEnd	bulkDateEnd		Integer	
dateBegin	dateBegin		Integer	
dateEnd	dateEnd		Integer	
cataloged	cataloged		boolean	
catalogedDate	catalogedDate		date	
catalogedNote	catalogedNote	no limit	string	
conditionNote	conditionNote	no limit	string	
extentNumber	extentNumber		real number	
extentType	extentType	255	string	Linear Feet, Cubic Feet
containerSummary	containerSummary	no limit	string	

inventory	inventory	no limit	text	
processingPlan	processingPlan	no limit	text	
processingPriority	processingPriority	255	string	
processors	processors	255	string	
processingStatus	processingStatus	255	string	
processingStartedDate	processingStartedDate		date	
resourceType	resourceType	255	string	Collection, Papers, Records
retentionRule	retentionRule	no limit	text	
title	title	no limit	text	
restrictionsApply	restrictionsApply		boolean	
rightsTransferred	rightsTransferred		boolean	
rightsTransferredDate	rightsTransferredDate		date	
rightsTransferrednote	rightsTransferredNote	no limit	text	
userDefinedDate1	userDefinedDate1		date	
userDefinedDate2	userDefinedDate2		date	
userDefinedBoolean1	userDefinedBoolean1		boolean	
userDefinedBoolean2	userDefinedBoolean2		boolean	
userDefinedInteger1	userDefinedInteger1		integer	
userDefinedInteger2	userDefinedInteger2		integer	
userDefinedReal1	userDefinedReal1		real number	
userDefinedReal2	userDefinedReal2		real number	
userDefinedString1	userDefinedString1	255	string	
userDefinedString2	userDefinedString2	255	string	
userDefinedString3	userDefinedString3	255	string	
userDefinedText1	userDefinedText1	no limit	text	
userDefinedText2	userDefinedText2	no limit	text	
userDefinedText3	userDefinedText3	no limit	text	
userDefinedText4	userDefinedText4	no limit	text	
resourceIdentifier	resourceIdentifier1	20	string	

Subject Fields / Column Headers

<u>Import Field/Column Header</u>	<u>Where Stored /AT Target</u>	<u>Field Length</u>	<u>Data Type</u>	<u>Rule</u>
subjectTerm	subjectTerm	150	string	
subjectTermType	subjectTermType	50	string	Types are: Function (657), Genre / Form (655), Geographic Name (651), Occupation (656), Topical Term (650), Uniform Title (630)
subjectSource	subjectSource	100	string	default "ingest"

Name Fields / Column Headers

<u>Import Field/Column Header</u>	<u>Where Stored /AT Target</u>	<u>Field Length</u>	<u>Data Type</u>	<u>Rule</u>
NameFunction	nameFunction	50	string	Creator, Source, Subject
Name_nameType	nameType	255	string	Person, Corporate Body, Family

Preparing the Tab-delimited Accession file

Name_sortName	sortName	255	string	
Name_qualifier	qualifier	255	string	
Name_number	number	255	string	
Name_Source	NameSource	50	string	
Name_descriptionType	descriptionType	255	string	Administrative History, Biography
Name_descriptionNote	descriptionNote	no limit	text	
Name_citation	citation	no limit	text	
Name_salutation	salutation	255	string	
Name_contactAddress1	contactAddress1	255	string	
Name_contactAddress2	contactAddress2	255	string	
Name_contactCity	contactCity	255	string	
Name_contactRegion	contactRegion	255	string	
Name_contactCountry	contactCountry	255	string	
Name_contactMailCode	contactMailCode	255	string	
Name_contactPhone	contactPhone	255	string	
Name_contactFax	contactFax	255	string	
Name_contactEmail	contactEmail	255	string	
Name_contactName	contactName	255	string	
Name_familyName	familyName	35	string	
Name_familyNamePrefix	familyNamePrefix	15	string	
Name_corporatePrimaryName	corporatePrimaryName	70	string	
Name_corporateSubordinate1	corporateSubordinate1	45	string	
Name_corporateSubordinate2	corporateSubordinate2	45	string	
Name_personalPrimaryName	personalPrimaryName	30	string	
Name_personalRestOfName	personalRestOfName	25	string	
Name_personalPrefix	personalPrefix	25	string	
Name_personalSuffix	personalSuffix	25	string	
Name_personalDates	personalDates	15	string	
Name_personalFullerForm	personalFullerForm	25	string	
Name_personalTitle	personalTitle	25	string	
Name_personalDirectOrder	personalDirectOrder		boolean	

Individual field constraints

accessionNumber

The Toolkit offers two ways to import **accessionNumber** data:

- a. If the import field is labeled **accessionNumber**, the number will be parsed so that an accession number, 2005-54.1 will be targeted to **accessionNumber1=2005**, **accessionNumber2=54**, **accessionNumber3=1**. Periods, forward slashes, and dashes are the three characters that will delimit the accession number fields.
- b. You may bypass this parsing by labeling the import field containing 2005-54.1 as **accessionNumber1**, but keep in mind that each of the four fields for the accession number is limited to ten characters.

Note: Accession numbers must be unique to import correctly.

resourceIdentifier

If the import field is **resourceIdentifier**, the ID will be mapped so that a resource identifier, MC 543 will be targeted to **resourceIdentifier1**=MC 543 and stored as part of the resource record, if the option to create resource data is selected at the beginning of the import process (see Chapter 6 for further clarification).

Date fields

dateBegin and dateEnd

These fields will accept only integer dates. A.D. dates after 1000 A.D. should be expressed with four digits, e.g., 1972, 1514, 2006, etc. B.C. dates should be expressed with a negative sign first, e.g., -999, -10000, -3.

dateExpression

Natural language date strings, such as ca. 1976-spring 1987, can be imported into the **dateExpression** field.

Note: The **dateExpression** field has a parser that will parse 1963-1987 into **dateBegin**=1963 and **dateEnd**=1987. If a **dateBegin** is provided without a corresponding **dateEnd** the record will not be imported, and if the **dateEnd** is less than the **dateBegin** (e.g., **dateBegin** = "1967" and **dateEnd** = "0"), the record will not be imported.

accessionDate, acknowledgementDate, agreementSent, agreementReceived

Upon import, the Toolkit requires that you declare the formal date structure for these transaction date fields. These transactional dates may be in the following formats: yyyy-m-d (example: 1963-09-27), yyyy/m/d (example: 1963/09/27), m-d-yyyy (example: 09-27-1963) or m/d/yyyy (example: 9/27/1963). Any data in this type of date field that is not properly formatted will cause an error.

Subject fields

The import of subjects requires the inclusion of both **subjectTerm** and **subjectTermType** fields.

subjectTermType

subjectTermType should be one of the following: (Function (657), Genre / Form (655), Geographic Name (651), Occupation (656), Topical Term (650), Uniform Title (630).

subjectTerm

Multiple subject terms may be imported by delimiting terms with a pipe character, for example: Boating. | United States History Civil War, 1861-1865. When multiple subjects are imported, they will all be given the same **subjectTermType** value, as only one **subjectTermType** may be used for a row of data. Therefore, only one type of subject may be imported for an accession record. Consequently, subject headings, especially multiple subject headings, imported with accession data will likely require remediation after import is complete. To overcome this limitation, you may

choose to import accession data without subject headings and create them directly in the AT. Subject headings created directly in the AT can be manually linked to appropriate accession records. Another option is to use the XML-based method of importing accession data, which allows for the ingest of multiple subjects and subject types. Repositories will need to determine which strategy is most efficient and cost-effective for them.

subjectSource

On import, subject terms are assigned the source value of “ingest” if no **subjectSource** is supplied. You may then more easily identify subject terms that require editing or remediation.

Name fields

Only one name per accession may be imported; therefore you may elect to import only a source name, only a creator name, or only the first name as a subject per each accession record. (Additional names can be added through the Toolkit interface, or by using the XML-based accession import method which does allow for the ingest of multiple names.)

Name column headings must be prefaced with “Name.” The only exceptions to this are the **nameFunction**, **nameRole**, and **nameForm** headings.

The import of names requires the inclusion of both **nameFunction** and **nameType**.

nameFunction

The function describes the relationship of the name to the accession. NameFunctions are limited to the following entries: source, creator, or subject.

nameType

The **nameType** must be: Person, Corporate Body or Family.

- a. If names exist in a single field in the legacy data, they may be imported into the primary name field for the particular name type, and parsed into individual fields by hand in the Toolkit. To load names into the correct type of primary name field, you must first separate each name into its appropriate target field, as shown below:

NameFunction	Name_corporatePrimaryName	Name_personalPrimaryName	Name_familyName	name_Type
creator		Smith, John, 1904-1990		Person
subject			Smith Family	Family
source	Acme Industrial Products			Corporate Body