Latin American Plants Initiative XML Primer

May 14, 2007

Version 1.0

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Purpose

This Primer document is intended to provide basic information and explanations about the use of XML and the preparation of XML documents for the Latin American Plants Initiative.

1.2 Specimen Metadata

For the LAPI project, specimen metadata refers to the data recorded about a scanned specimen. This data is generally recorded in a database, but sometimes can be in a spreadsheet or text document. Examples of metadata are the barcode number, determined name, collector, locality, collection dates, etc.

For the LAPI project, metadata must be recorded for each digitally imaged type specimen. For each batch of image files submitted to Aluka, an XML file must be submitted containing metadata for each image and the file must comply with standards established for the project.

Additional information about the LAPI project, guidelines and data formats can be found at the Aluka website (http://aluka.ithaka.org/plants/lapidocuments.html/).

1.3 XML Schema

The format of the file to be used for exporting LAPI data is XML. A standard XML schema originally developed for the African Plants Initiative will be used for the Latin American Plants Initiative. The schema defines the required and optional elements of the XML and constrains the content and structure of the metadata being exported.

This document will review the standard XML schema and help to explain its sections.

1.4 Why Standardize?

Each LAPI partner may utilize its own method for the capture of the specimen metadata for each of the scanned specimens. However, these different metadata sources must be joined into a single repository at Aluka. To enable the disparate data sources to be joined together, they must be unified into a single, uniform database. By standardizing on a common XML schema, all partners have a pattern to follow to transform their original data into standard form that can be combined. However, conforming to a standard generally means that each partner will have some extra data transformation work to produce the standard XML metadata file. But, the effort to produce the standardized output will enable the combined repository to be created in way that enables each partner's data to be presented accurately and completely in the combined repository.

1.5 Terminology: Batches, Datasets and Units

Digitized images will be submitted to Aluka in groups called Batches. A Batch consists of a large number of TIFF files that have been created by the HerbScan device, one for each digitized specimen image. These files will be stored on an external hard drive that will be shipped out. The name of each image file will be the institution's Index Herbariorum code concatenated with barcode number (eg. AR1234567) with the .tif extension.

The term Dataset refers to the metadata for a Batch. A Dataset is an XML file which contains the metadata for all of the image files in the Batch. There will be one Dataset on each external hard drive when it is submitted. A one-to-one match is required between the specimen metadata records/UnitIDs in the Dataset and the specimen image files. Each metadata record/UnitID must match an image file, and each image file must have a metadata record.

Exception

Where multiple images are made for single specimen, such as a more detailed scan of a part of the sheet or multiple sheets for single specimen, there can be multiple files for the same UnitID, but the files must be named UnitID_a, UnitID_b, etc.

Note

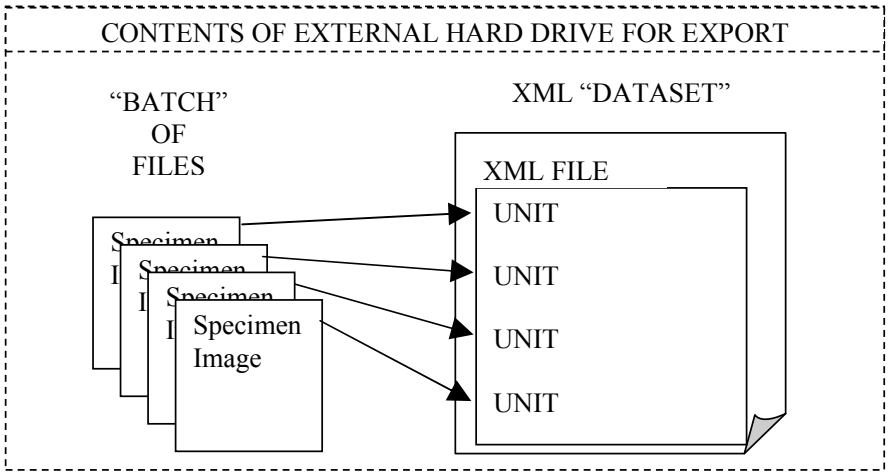
The term "dataset" is used generically to refer to the XML file. There is also an XML tag called DataSet that is part of the structure of the XML file.

The term Unit refers to a single specimen. All of the metadata for a single specimen image file is associated with that Unit. There is an XML tag called Unit that is part of the structure of the XML file.

Note

Do not send a separate XML file for each Unit or specimen. Instead, combine them together as Units within one Dataset.

The following diagram illustrates these concepts:



1.6 General XML Schema Information

All XML files must follow a standard structure to be readable by an XML viewer. The rules of this structure are defined in an "XML Schema" document. The standard schema for the LAPI project is AfricanTypesv2.xsd which is available at the Aluka website.

More information about XML and XML Schema can be found at:

XML Tutorial: http://www.w3schools.com/xml/default.asp
O'Reilly's XML Tutorials: http://www.xml.com
XML Schema Primer: http://www.w3.org/TR/xmlschema-0/

1.7 LAPI Metadata XML File Name

The XML file for the LAPI metadata must be named according to your institution herbariorum (http://sciweb.nybg.org/science2/IndexHerbariorum.asp), batch number and date using the following filename format:

institution_batch_YYYYMMDD.xml

Batches are numbered sequentially starting from 0 for the first test batch, and then 1, 2, 3 etc. for each batch sent to Aluka.

Example:

Institution: Missouri Botanical Garden (IH code: MO)

Batch number: 2 (the second batch ever sent to Aluka)
Date: 1 November 2006

Filename would be: **MO_2_20061101.xml**

1.8 XML Header (Mandatory)

Every XML file must have a header section. The header must be the first line of the file and contain the following:

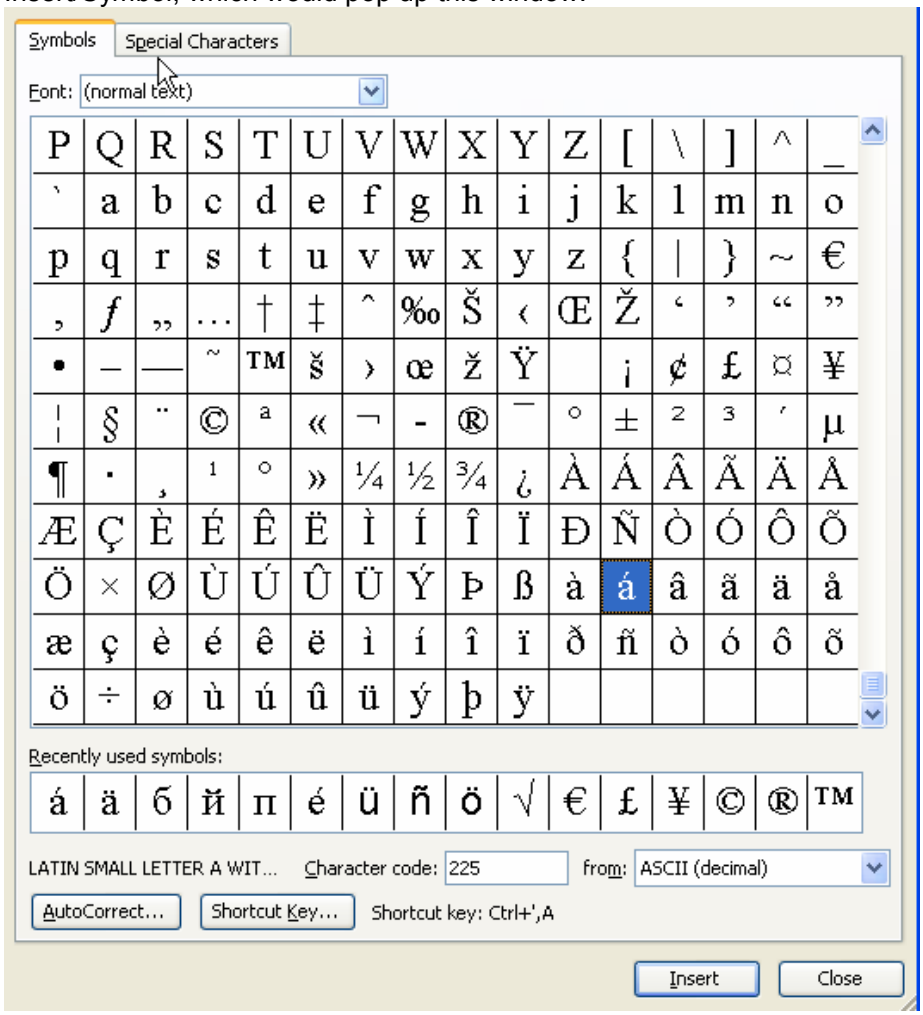
```
<?xml version="1.0" encoding="UTF-8" ?>
```

1.8.1 What is UTF-8?

UTF-8 refers to the underlying method to be used for the text characters included in the XML file. UTF-8 is one way of displaying Unicode; there are others, but UTF-8 is used for LAPI.

The main impact of specifying UTF-8 encoding for an XML file is that it actually *excludes* a very common form of data encoding used in Western countries, namely the Microsoft “symbols” which can be inserted in the Western versions of Microsoft Word, Excel and Access.

For example, to insert an “accented a” in Microsoft Word, you can use Insert/Symbol, which would pop up this window:



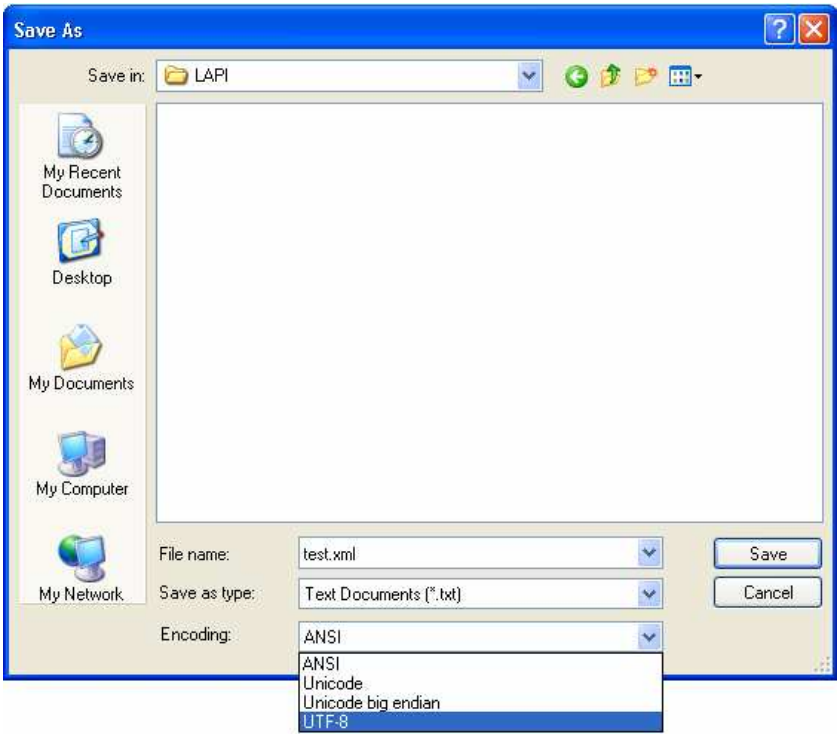
Then by selecting the “accented a” (highlighted in the figure), the following character would be inserted into the document: á The “character code” shown in the window at the bottom is 225 from ASCII (decimal) or in hexadecimal it is single-byte E1. This same code becomes two-byte hexadecimal 00E1 in Microsoft’s native Unicode format, UCS2 (also called UTF-16).

Note

XML that specifies “encoding =UTF-8” will cause the web browser to reject any ASCII or UTF-16 code between 129 and 255. For instance, Microsoft Internet Explorer will display this error: An invalid character was found in text content.

If any Windows extended characters are included in the metadata export, steps must be taken to ensure that the XML file is converted to UTF-8 encoding. This will not happen by default with Microsoft Office tools.

One simple method is to use Microsoft Notepad, Version 5, or later to open the exported XML file and then select Save As and choose UTF-8 as the Encoding at the bottom of the window as in this example:



1.9 General XML Formatting Rules

The following general XML formatting rules must be followed in constructing the LAPI XML metadata file. The XML document:

- must begin with the XML declaration (header)
- must have one unique root element ("DataSet" for LAPI)
- all start tags must match end-tags
- XML tags are case sensitive (unlike HTML!)
- all elements must be closed
- all elements must be properly nested
- all attribute values must be quoted
- XML entities must be used for reserved characters

1.10 XML Entities for Reserved Characters

The following text characters are reserved by the XML structure and cannot be included in any metadata values. The "XML Entitites" must be substituted for these characters before being included in the exported XML file.

	Reserved Character	XML Entities
Greater than	>	>
Less than	<	<
Ampersand	&	&
Quote	"	"
Apostrophe	'	'

Note

If these reserved characters are inadvertently included in the XML value data, the XML file will produce unpredictable results when viewed with a browser.

For Example:

<Locality>Koopmansfontein: Agricultural Research Station; Golden Rock. Pan. 28°11'49.7"S 24°06'17.9"E</Locality>

Should look like:

<Locality>Koopmansfontein: Agricultural Research Station; Golden Rock. Pan. 28°11'49.7"S 24°06'17.9"E</Locality>

1.11 Correct Spelling of Tags

All LAPI XML tags must be spelled correctly using the correct letter case.

Correct Tag	Incorrect Tags
DataSet	Dataset, dataset

StoredUnderName	Storedundername, storedundername
InstitutionCode	Institutioncode

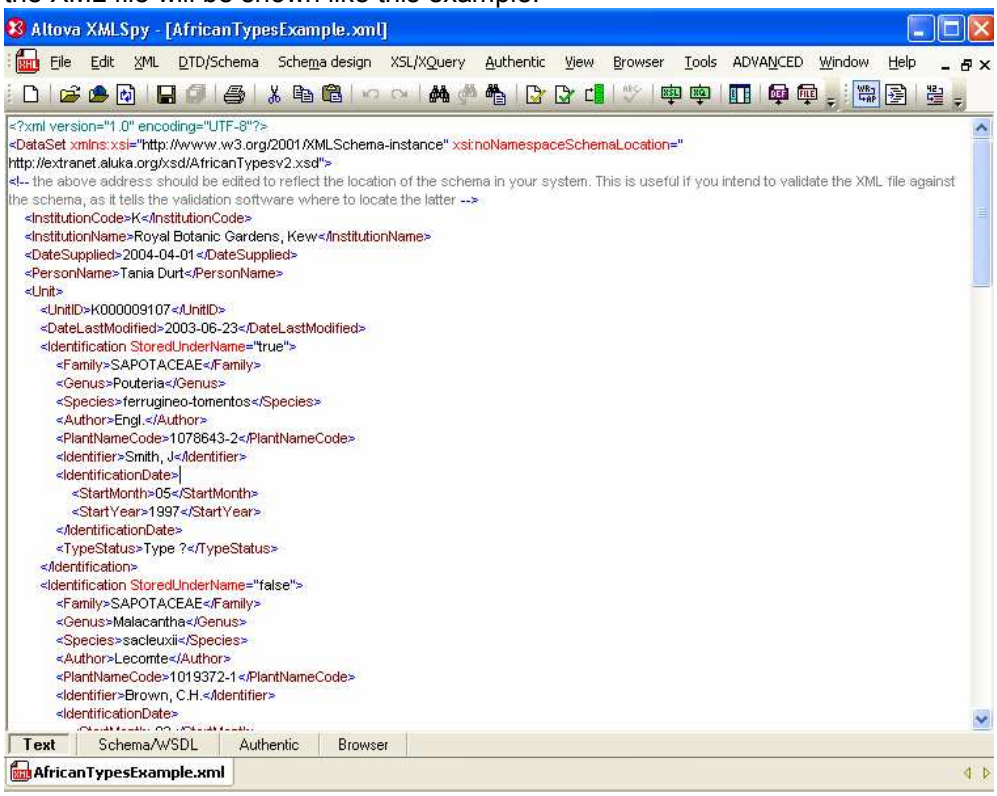
1.12 Validating the LAPI XML file

The LAPI Metadata XML file must be validated before sending to Aluka. Validated means that the file content and structure conform to the basic XML formatting rules as well as the LAPI standard schema.

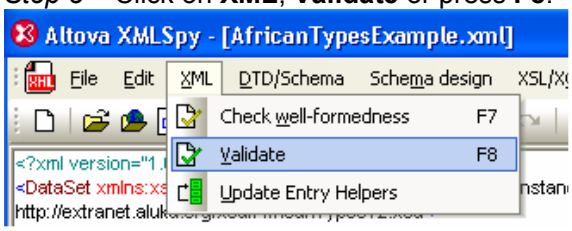
Using XML Spy to Validate an XML file
One easy way to validate an XML file is by using XML Spy. (XML Spy 2007 Home Edition is published by Altova, www.altova.com, and must be purchased after A 30-day free trial) The following provides an example of using XML Spy to validate.

Step 1 – Run **XML Spy**

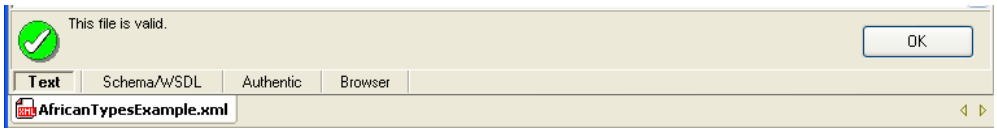
Step 2 – Click on **File, Open** and select the Metadata XML file. The contents of the XML file will be shown like this example:



Step 3 – Click on **XML**, **Validate** or press **F8**.



Step 4 – A message will appear at the bottom of the screen indicating whether the file is valid or not valid.



Here is an example of a not valid message, caused by a missing end tag. All such errors must be corrected.

The file must be valid before it can be sent to Aluka.

1.13 Other XML File Validation Tools/Websites

There are other websites or software tools that can be used to validate the XML Metadata file:

XML Cooktop – free download – Windows only - http://www.xmlcooktop.com/
XML Fox – free download – Windows only - http://xmlfox.com/download.htm
XML Buddy – free download – Eclipse-based - http://www.xmlbuddy.com/
Website - http://www.xmlvalidation.com
Website - http://www.validome.org/xml/

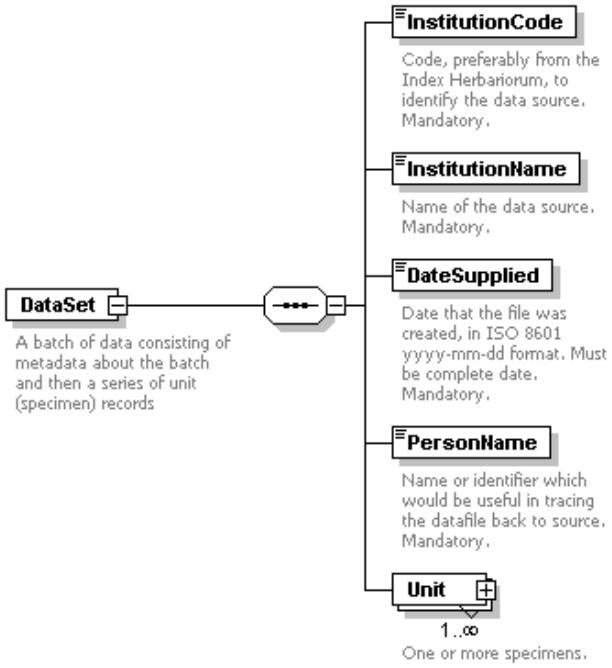
2. LAPI XML SCHEMA PRIMER

2.1 LAPI Schema Summary

The LAPI XML Schema consists of a “Dataset” tag with 5 main tags:

- InstitutionCode
- InstitutionName
- DateSupplied
- PersonName
- Unit (repeats, one for each specimen image file)

This is a diagram of the top-level schema structure:



2.2 DataSet Tag -(Mandatory)

The DataSet tag is required for the LAPI schema. Its form is:

```
<DataSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://extranet.aluka.org/xsd/AfricanTypesv2.xsd">
.
.
.
</DataSet>
```

This tag must be spelled “DataSet” and not “Dataset”, “dataset” or “dataSet”.

2.3 InstitutionCode, InstitutionName, DateSupplied and PersonName Tags (Mandatory)

Each of these four tags are mandatory.

1._ The **InstitutionCode** value must be from Index Herbariorum (http://sciweb.nybg.org/science2/IndexHerbariorum.asp) unless there is no IH code for the institution and then a code will be assigned for LAPI.
2._ The **InstitutionName** is the name of your institution.
3._ The **DateSupplied** is the date of the creation of the metadata file.
4._ The **PersonName** is a contact at the Instution for potential follow-up.

An example of the form of these tags is:

<InstitutionCode>**K**</InstitutionCode>
<InstitutionName>**Royal Botanic Gardens, Kew**</InstitutionName>
<DateSupplied>**2004-04-01**</DateSupplied>
<PersonNam e>**John Jones**</PersonName>

2.4 Unit Tags

There must be at least one Unit tag for each LAPI XML file. But an unlimited number of Unit tags can be included. Each Unit tag usually represents one image file in a Batch. The UnitID and the image filename name must match based on the barcode number.

Exception

Where multiple images are made for a single specimen, such as a more detailed additional scan of a part of the sheet or multiple sheets for a single specimen, there can be multiple image files for the same UnitID, but the files must be named UnitID_a.tif, UnitID_b.tif, etc.

Since there are many potential metadata values associated with a specimen image, the Unit tag has many sub tags available to be used. Some are required and some are optional.

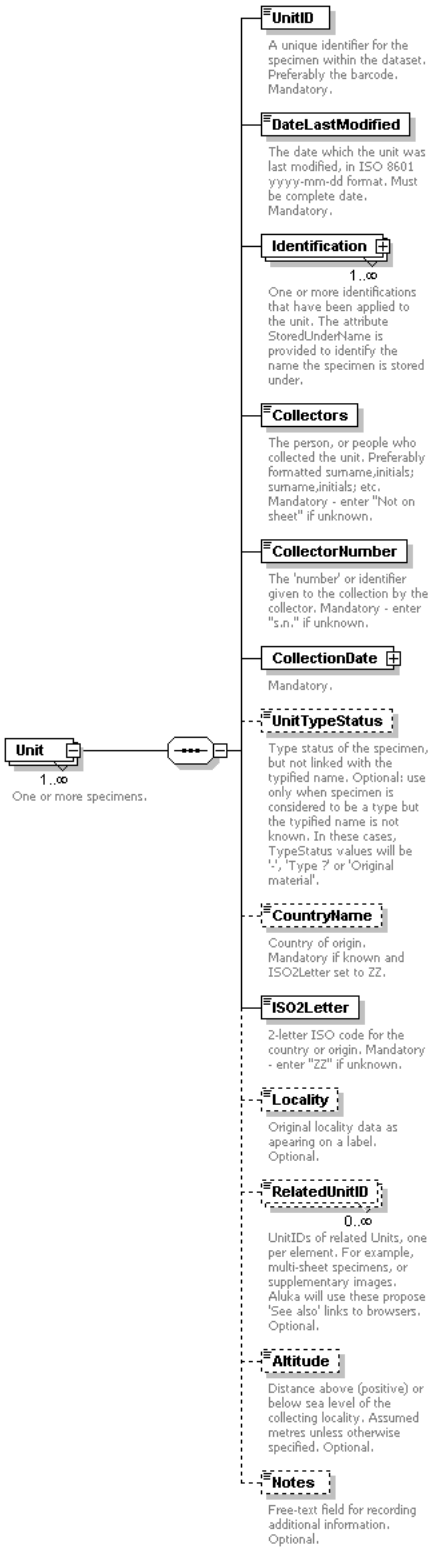
There are 7 required tags for each Unit:

- UnitID
- DateLastModified
- Identifications (at least one)
- Collectors
- CollectorNumber
- CollectionDate
- ISO2Letter

There are 6 optional tags for each Unit:

- UnitTypeStatus
- CountryName
- Locality
- RelatedUnitID
- Altitude
- Notes

Here is a diagram of the Unit tag and its subtags:



2.5 Explanation of Required Tags for Unit Tag

2.5.1 UnitID -(Mandatory)

This is a unique identifier for the Unit or specimen image. It is **required** to be the same as the InstitutionID concatenated with the barcode on the specimen and not preferable as stated in the schema description. The UnitID is also required to be the same as the name of the specimen image file for the Unit. One of the quality checks for data submission is to compare the UnitIDs in the XML dataset with the filenames on the hard drive for an exact match (except for the multi-image files with _a, _b extensions that have been noted above.)

Example:

<UnitID>K**1000081**</UnitID>
Or
<UnitID>US987634B</UnitID>

Invalid UnitIDs

<UnitID>12121212</UnitID>
<UnitID>34567ABC</UnitID>
<UnitID>L444555666</UnitID> if L is not the institution code

Note

Do not include any spaces in the **UnitID** value

If the barcode on the specimen does not include the institution's code as a prefix, the UnitID and the image filenames can be created on output from the institution's metadata by concatenating the institution code with the barcode.

2.5.2 DateLastModified -(Mandatory)

This date refers to the last time any part of the metadata for this Unit or specimen image was changed. This "last change date" is generally recorded in the primary database of the scanning institution. Note that the format is yyyy-mm-dd, including the dashes.

Example:

<DateLastModified>**2006-06-23**</DateLastModified>

2.5.3 Identification -(Mandatory)

Each Unit can have multiple Identification tags. The intent of this is to enable recording of multiple names associated with a specimen sheet rather than just one determined name. For instance, a single specimen could have multiple

determined names by different specialists, plus it could have different names for which it is a type, and it could have a name it is filed by.

StoredUnderName Attribute

The Identification tag has a mandatory attribute called StoredUnderName. This StoredUnderName attribute is either "true" or "false". **One and only one** Identification tag can have a "true" value for StoredUnderName, because no specimen should be stored or filed under more than one name.

In other words, the value of "true" for StoredUnderName should occur only once for any Identification tag within a single Unit. All other Identifications within a single Unit must have a value of "false" for StoredUnderName. This rule is checked as part of the quality control of submitted LAPI XML metadata files.

Note

Do not use "0" or "1" or "Yes" or "No" or "True" or "False" for the value of StoredUnderName. The only acceptable values are either "true" or "false". Only one true and as many false as needed can be used.

How is StoredUnderName intended to be used?

In some herbaria, specimens are placed inside folders or organized by one name, the "stored under" name, but may also have a different determined name or type name on the sheet. Using this attribute allows the name chosen by the institution as the stored or filed name to be distinguished from any others associated with the specimen.

What if we do not use "Stored Under" Names?

In many herbaria, no distinction is made between the determined name and the way it is stored or filed. And some herbarium databases do not record multiple names for a single specimen. In either of these cases, a single name would be recorded in the specimen database for the sheet.

If only one name is recorded in the database, then the StoredUnderName attribute must be "true" for that name, since that one name is serving the purpose of a stored or filed name.

Example:

<Identification StoredUnderName="**true**"> *Note: should occur only once for any Identification tag within a single Unit.*

.

</Identification>

Or

<Identification StoredUnderName="**false**"> *Note: All other Identification tags within a single Unit must have a value of "false"*
.
</Identification>

Identification <u>Required</u> Subtags

Each Identification has a further 7 **required** subtags:

1. **Family** – (Mandatory) Based on the scanning institution's own taxonomic decisions or as shown on the sheet. The Family must be entered in uppercase letters.

   Example:
   <Family>**ASCLEPIADACEAE**</Family>

2. **Genus** – (Mandatory) As recorded by the scanning institution. First letter should be uppercase.
   Example:
   <Genus>**Secamone**</Genus>

3. **Species** – (Mandatory) As recorded by the scanning institution. Should be all lowercase.

   Example:
   <Species>**grandiflora**</Species>

4. **Author** – (Mandatory) The author of the **species name** including basionym author and ex/in authors plus year of publication following standard format. Standard Author Abbreviations should be used based on Authors of Plant Names maintained by RBG Kew at http://www.ipni.org/ipni/authorsearchpage.do

   Example:
   <Author>**Klack.**</Author>

   Note
   If species author is missing or unknown, use "Not on sheet".

5. **Identifier** – (Mandatory) The name of the person recorded by the scanning institution who made the determination of this Identification.

   Note

Use “Not on sheet” if the identifying/determining person is not known.

Example:

<Identifier>**Not on sheet**</Identifier>

6. **IdentificationDate** – (Mandatory) The date recorded by the scanning institution for when the determination of this Identification was made. The date value is not entered directly under this tag. Rather the standard Date subtags must be used. See section 2.7 ***Date Subtags***.

   Example:

   <IdentificationDate>
   <StartDay>**27**</StartDay>
   <StartMonth>**01**</StartMonth>
   <StartYear>**1992**</CollectionDate>
   </IdentificationDate>

   Or where there is no date:

   <IdentificationDate>
   <Other Text>**Not on Sheet**</OtherText>
   </IdentificationDate>

7. **TypeStatus** – (Mandatory) The type status of this Identification. Use “-“ if the Identification is **not** a type name.

   Use ONLY one of the following values for type status:

“Holotype”	***“Epitype”***
“Isoepitype”	***“Lectotype”***
“Isolectotype”	***“Neotype”***
“Isoneotype”	***“Paratype”***
“Isoparatype”	***“Syntype”***
“Isosyntype”	***“Isotype”***
“Type”	***“Type?”***
“Original material”	*“-”*

   Please refer to the “Sorts Of Types” document on the Aluka website for guidance on type nomenclature usage.
   http://aluka.ithaka.org/plants/lapidocuments.html

   For example:

   - “Co-type” should be “Syntype”

- "Type material" should be "Type"
- "ISO" should be "Isotype"
- "TYPE" should be "Type"
- "TYPUS" should be "Type"

Note
If a specimen is known or thought to be a type specimen, but the name for which it is a type is unknown, then all Identifications will have TypeStatus of "-"..

Example:
<TypeStatus>**Type**</TypeStatus>

And each Identification has a further 6 **optional** tags:

1. GenusQualifier – (Optional) Qualifier expressing doubt about the genus epithet (eg. cf)

Example:
<GenusQualifier>**cf**</GenusQualifier>

2. SpeciesQualifier – (Optional) Qualifier expressing doubt about the species epithet (eg. cf)

Example
<SpeciesQualifier>**cf**</SpeciesQualifier>

3. Infra-specificRank – (Optional) Rank based on ICBN and as recorded by the scanning institution. Should be all lowercase.

Example:
<Infra-specificRank>**var.**</Infra-specificRank>

4. Infra-specificEpithet – (Optional) As recorded by the scanning institution. Should be all lowercase.

Example:
<Infra-specificEpithet>**alba**</Infra-specificEpithet>

5. Infra-specificAuthor – (Optional) Follow the same guidelines as for the Author subtag.

Example:
<Infra-specificAuthor>**Wild.**</Infra-specificAuthor>

6. PlantNameCode – (Optional) An optional code that is meaningful to the scanning institution for the name given for this Identification. Often

a tracking number. May be used to provide feedback from Aluka to the scanning institution.

Example:
<PlantNameCode>**123ABC**</PlantNameCode>

2.5.4 Collectors -(Mandatory)

This is just a text string listing the Collector or Collecting Team for this Unit. The preferred way of listing a Collecting Team is:

Surname1, Initials1; Surname2, Initials2; Surname3, Initials3

using a semi-colon to separate the individual collectors. The Senior Collector should be listed first. If no collector data is available, then **the value "Not on sheet" must be manually inserted** in the XML **This tag cannot be left blank**.

Example:
<Collectors>**Beentje, H.J.; Quansah, N.**</Collectors>

Or if there are no collectors recorded

<Collectors>**Not on Sheet**</Collectors>

2.5.5 CollectorNumber -(Mandatory)

This is generally the number assigned by the senior collector to the specimen. But, it can contain letters if needed. Where there is no collector's number for the Unit**, the value "s.n." must be inserted** in the XML. **This tag cannot be left blank.**

Example:
<CollectorNumber>**4559**</CollectorNumber>

Or

<CollectorNumber>**4559, 4560**</CollectorNumber>

Or if there is no collector number

<CollectorNumber>**s.n.**</CollectorNumber>

2.5.6 CollectionDate -(Mandatory)

The date recorded by the scanning institution for when the collection was made. The date value is not entered under this tag. Rather the standard Date subtags must be used. See section 2.7 ***Date Subtags***.

Example:

```
<CollectionDate>
        <StartDay>27</StartDay>
        <StartMonth>01</StartMonth>
        <StartYear>1992</CollectionDate>
</CollectionDate>
```

Or if there is no collection date

```
<CollectionDate>
        <OtherText>Not on Sheet</OtherText>
</CollectionDate>
```

2.5.7 ISO2Letter -(Mandatory)

This is the 2-letter ISO 3166-1 code for the country where the specimen was collected. The ISO 3166 master list is available at http://www.iso.org/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html

The following table lists all of the ISO 3166-1 Caribbean and Central and South American country codes:

COUNTRY NAME	ISO CODE	COUNTRY NAME	ISO CODE
ANTIGUA AND BARBUDA	AG	GUYANA	GY
ARGENTINA	AR	HAITI	HT
ARUBA	AW	HONDURAS	HN
BAHAMAS	BS	JAMAICA	JM
BARBADOS	BB	MARTINIQUE	MQ
BELIZE	BZ	MEXICO	MX
BOLIVIA	BO	NETHERLANDS ANTILLES	AN
BRAZIL	BR	NICARAGUA	NI
CAYMAN ISLANDS	KY	PANAMA	PA
CHILE	CL	PARAGUAY	PY
COCOS (KEELING) ISLANDS	CC	PERU	PE
COLOMBIA	CO	PUERTO RICO	PR
COSTA RICA	CR	SAINT KITTS AND NEVIS	KN
CUBA	CU	SAINT LUCIA	LC
DOMINICA	DM	SAINT VINCENT AND	VC

		THE GRENADINES	
DOMINICAN REPUBLIC	DO	SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	GS
ECUADOR	EC	TRINIDAD AND TOBAGO	TT
EL SALVADOR	SV	TURKS AND CAICOS ISLANDS	TC
FALKLAND ISLANDS (MALVINAS)	FK	URUGUAY	UY
FRENCH GUIANA	GF	VENEZUELA	VE
GRENADA	GD	VIRGIN ISLANDS, BRITISH	VG
GUADELOUPE	GP	VIRGIN ISLANDS, U.S.	VI
GUATEMALA	GT		

When the country is missing or unknown for a specimen, the 2-letter code "ZZ" must be inserted into the XML.

Note

Where the institution has not utilized ISO codes in its data system, a conversion will need to be made from the country coding system used by the institution to the ISO2 system before inclusion in the XML metadata file.

Example:

<ISO2Letter>**BR**</ISO2Letter>

2.6 Explanation of Optional Tags for Unit Tag

2.6.1 UnitTypeStatus –(*Optional)*

This tag is only used by RBG Kew. All other institutions will omit it.

Note

If a specimen is known or thought to be a type specimen, but the name for which it is a type is unknown, then for LAPI nothing is to be recorded in the XML metadata for UnitTypeStatus.

2.6.2 CountryName – (Optional)

This name is not needed if an ISO2Letter value has been provided. Only provide a value for this tag if there is no ISO2 code for the country and the value of "ZZ" has been inserted for ISO2Letter. This tag allows an unusual country name not recognized by ISO to be assigned to the Unit or specimen image.

Example:

<CountryName>**AnUnusualCountryName**</CountryName>

2.6.3 Locality – (Optional)

This is the literal string of text that was recorded for "locality" describing where the specimen was collected. The original data needs to be carefully examined before export to XML to replace any of the XML reserved characters - <, >, &, ', and ". Also, if locality data includes accented or extended ASCII characters refer to the UTF-8 information discussed earlier on section 1.7.1 What is UTF-8?

Example:

<Locality> **AMBANJA: Manongarivo Special Reserve, Bekolosi Mt.; in open montane forest.** </Locality>

2.6.4 RelatedUnitID – (Optional)

This is a multiple occurrence tag, so multiple UnitIDs for related Units can be included. This will be the UnitID (IH code concatenated with barcode) of another specimen. There is no attribute to describe or classify the nature of the relation to the other specimen, just the existence of a relation.

Example:

<RelatedUnitID> K0123456</RelatedUnitID>

2.6.5 Altitude – (Optional)

This is a number in meters of the altitude above sea level where the specimen was collected. Please enter 'meters' or 'feet' and not just the first character 'm' or 'f'.

Example:

<Altitude>**100 meters** </Altitude>
Or
<Altitude>**25.5 feet**</Altitude>

2.6.6 Notes – (Optional)

This can be any text and has no other constraint. Be cautious of the reserved characters and extended characters as with other text. Section 1.9 XML Entities for reserved characters.

Example:

<Notes>**This can be any text except reserved characters like &apos**
</Notes>

2.7 Date Subtags

For **CollectionDate** and **IdentificationDate** 7 subtags are used to specify the date value:

1. StartDay
2. StartMonth
3. StartYear
4. EndDay
5. EndMonth
6. EndYear
7. OtherText.

All of these subtags are optional. However, it is expected that if a Day is recorded, there must also be a Month and Year. And, if a Month is recorded, there must be a Year. A value for Year can be provided without Month or Day. And, there should not be an end date value if there is no start date value.

Note
At least one subtag of Date must have a value.

Note
If there are no date values on the sheet, then "Not on Sheet" must be inserted into OtherText.

2.8 Handling of Missing Data

For the LAPI project, specific data values have been chosen to be used when no data is available for required XML tags, rather than just leaving them empty. This enables standardization of the data for smoother integration into the Aluka repository.

Using "Not on Sheet"
The text string "Not on Sheet" is required to be used for the following required tags, if no data is available. This value is not requred to be recorded in the institution's database or regular specimen data formt. Each institution is free to record missing or blank data as it chooses. But, on output of the XML metadata, the recorded values must be converted to "Not on Sheet" in the XML file.

Use "Not on Sheet" for missing, empty or blank values for:

- Unit/Identification/Author
- Unit/Identification/Identifier
- Unit/Collectors
- Date/Other Text – if there is no Month, Day, or Year

Use “s.n.” for missing, empty or blank values for:

- Unit/CollectorNumber

Use “ZZ” for missing, empty or blank values for:

- Unit/ISO2Letter:

3. LAPI METADATA XML EXAMPLES

3.1 Example 1 – Standard Specimen

Kommentar [r1]: This image has the wrong color checker?

Image File Name(s) to Be Used for this Specimen
The barcode of this specimen is TAN000081, so the filename is:

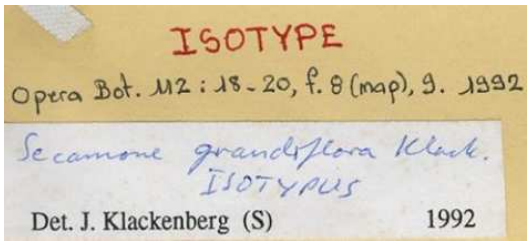
TAN000081.tif

Specimen Metadata Image Details

Observed LAPI Metadata

Barcode: **TAN000081**

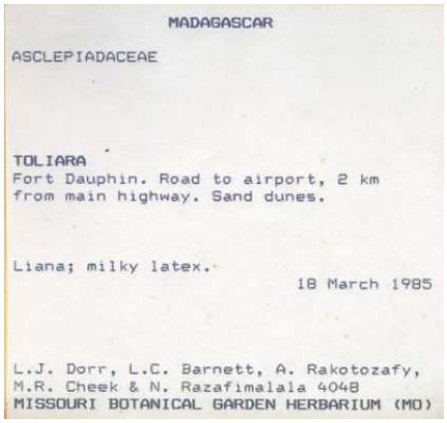
Note: Since this barcode includes the IH code TAN for the herbarium, it can be used for the UnitID and filename.



Observed LAPI Metadata

Identification
- Genus: **Secamone**
- Species: **grandiflora**
- Author: **Klack.**
- Type Status: **Isotype**

Identifier: **J. Klackenburg**
Identification Date: **1992**



Observed LAPI Metadata

Country Name: **Madagascar**
Identification
- Family: **ASCLEPIADACEAE**

Locality: **Fort Dauphin. Road to airport, 2 km from main highway. Sand dunes.**
Collection Date: **18 March 1985**
Collectors: **Dorr, L.J.; Barnett, L.C.; Rakotozafy, A.; Cheek, M.R.; Razafimalala, N.**
Collector Number: **4048**

Inferred LAPI Metadata

ISO2 Code: **MG** (The ISO code for Madagascar)
StoredUnderName = **"true"** (Since there is only one Identification of Secamone grandiflora Klack., it is required to be the stored under name.)

Other Required LAPI Metadata Not on the Sheet

Institution Code: **TAN** (also could be inferred from barcode)
Institution Name: **Herbier du Parc Botanique et Zoologique de Tsimbazaza** (derived from TAN institution code)
Date Supplied: **2006-06-23** (based on date of submission of this batch to Aluka)
Person Name: **Chris Freeland** (based on person submitting the batch to Aluka)
Plant Name Code: **2611054** (number assigned to the name Secamone grandiflora Klack.in the institution's database**)**
Date Last Modified: **2006-06-23** (based on date of submission of this particular specimen image)

SAMPLE SPECIMEN METADATA XML FOR THIS EXAMPLE
Here is a sample XML record for the specimen (TAN000081) shown above:

```
<?xml version="1.0" encoding="UTF-8" ?>
<DataSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://extranet.aluka.org/xsd/AfricanT
ypesv2.xsd">
<InstitutionCode>TAN</InstitutionCode>
<InstitutionName>Herbier du Parc Botanique et Zoologique de
Tsimbazaza</InstitutionName>
<DateSupplied>2006-06-23</DateSupplied>
<PersonName>Chris Freeland</PersonName>
<Unit>
        <UnitID>TAN000081</UnitID>
        <DateLastModified>2006-06-23</DateLastModified>
        <Identification StoredUnderName="true">
                <Family> ASCLEPIADACEAE</Family>
                <Genus>Secamone</Genus>
                <Species>grandiflora</Species>
                <Author>Klack.</Author>
                <PlantNameCode>2611054</PlantNameCode>
                <Identifier>J.Klackenberg</Identifier>
                <IdentificationDate>
                        <StartYear>1992</StartYear>
                </IdentificationDate>
                <TypeStatus>Isotype</TypeStatus>
        </Identification>
        <Collectors>L.J. Dorr, L.C. Barnett, A. Rakotozafy, M.R. Cheek
        &amp;amp;N. Razafimalala</Collectors>
        <CollectorNumber>4048</CollectorNumber>
```

```
        <CollectionDate>
            <StartDay>18</StartDay>
            <StartMonth>3</StartMonth>
            <StartYear>1951</StartYear>
        </CollectionDate>
        <CountryName>Madagascar</CountryName>
        <ISO2Letter>MG</ISO2Letter>
        <Locality>Fort Dauphin. Road to airport, 2 km from main highway,
        Sand dunes. </Locality>
</Unit>
<DataSet>
```

3.2 Example 2 – Multiple Specimens on One Sheet

There are three specimens mounted on this sheet. Each of these will be a Unit in the XML.

Image File Name(s) to Be Used for this Specimen:
Three copies of the same image file, each with a different name for each of the barcodes:

filename

Specimen Metadata Image Details

First Specimen
Image Details1

Observed LAPI Metadata

Second Specimen
Image Details2

Observed LAPI Metadata

Third Specimen
Image Details3

Observed LAPI Metadata
Xxx:
Xxx:
Xxx:

Inferred LAPI Metadata
Xxx:
Xxx:
Xxx:

Other Required LAPI Metadata Not on the Sheet
Xxx:
Xxx:
Xxx:

SAMPLE SPECIMEN METADATA XML FOR THIS EXAMPLE

3.3 Example 3 – Multiple Sheets for the Same Specimen

This specimen is mounted on three sheets. Two approaches may be used for the XML in this case. One Unit can be used for all three sheets. Or three Units can be included, each with relations to the others. Examples for both approaches are provided below.

BM0000123_a.tif

BM0000123.tif

BM0000123_b.tif

BM0000124.tif

BM0000123_c.tif

BM0000125.tif

Approach 1 – Treat the specimen as one Unit with 3 images

Image File Name(s) to Be Used for this Specimen:

BH0000123_a.tif
BH0000123_b.tif
BH0000123_c.tif

Specimen Metadata Image Details

Observed LAPI Metadata

Inferred LAPI Metadata

Other Required LAPI Metadata Not on the Sheet

SAMPLE SPECIMEN METADATA XML FOR THIS EXAMPLE

Approach 2 – Treat the specimen as three related Units with 3

Image File Name(s) to Be Used for this Specimen:

BH0000123.tif
BH0000124.tif
BH0000125.tif

Specimen Metadata Image Details

Observed LAPI Metadata

Inferred LAPI Metadata

Other Required LAPI Metadata Not on the Sheet

SAMPLE SPECIMEN METADATA XML FOR THIS EXAMPLE