

EML FOR MERE MORTALS: A GUIDE TO USING ECOLOGICAL METADATA LANGUAGE TO DOCUMENT ECOLOGICAL DATA

Who should read this?

This document is designed for ecologists, ecological information managers, and ecology students.

Assumptions about the reader:

Is computer literate.

Has an understanding of ecological terminology.

May or may not know anything about XML

Definitions and Formatting Conventions

XML

XML Schema

EML document: to say that a document is an EML document means that it is a valid XML document; that the document follows the structure defined in a series of xml schema documents. When EML (all caps) is used, it is used as an abbreviation for Ecological Metadata Language. When `<eml>` (lower case, inside `< >`) is used, it refers to the the top level element of an EML document.

EML: abbreviation for Ecological Metadata Language.

`<eml>` : shorthand notation for `<eml:eml>` `</eml:eml>`

NOTATION CONVENTIONS. Since EML documents are XML documents, they use XML tagging notation. Throughout this document, you will find references to EML elements. Unless otherwise stated, all of the examples will refer to documenting data objects. EML can also be used to document literature citations, research protocols, or software independent of any dataset. The following notations will be used when referring to an EML element: `..<xxxx>` or `...<xxxx>`. The `..` means that the element is a the third level of an EML document. In effect, `..` means `<eml>/<dataset>`. So `..<coverage>` in the actual xml document would look like

`</eml>`

A reference in the form of `...<xxxx>` means that the element in question is an arbitrary number of levels deep in the EML structure.

XMLtags are case sensitive. For example, the tags `<species>` and `<Species>` are not the same.

.Introduction

Ecological Metadata Language (EML) is a content standard for documenting ecological data.

This standard is implemented using XML Schema to define the structure of XML documents that conform to the standard. EML has been designed for two major uses. The first and most important use is to define a common structure that all ecologists can use to document ecological data so that other ecologists can correctly interpret the data.

The second purpose for EML is to provide a structure so that software applications can be developed. The

kinds of applications anticipated range from a basic tool that allows a user to perform very specific targeted searches for datasets to advanced applications that could perform automatic integration of datasets.

```
<eml>
  <dataset>
    <coverage></coverage>
  </dataset>
</eml>
```

A reference to ..<coverage></geographicCoverage> would be interpreted as:

```
<eml>
  <dataset>
    <coverage>
      <geographicCoverage></geographicCoverage>
    </coverage>
  </dataset>
```

A Taxonomy of EML

The two purposes of EML can be looked at as forming the two basic branches in the taxonomy of EML. As can be seen in Figure 1, EML has both a scientific description “kingdom” and a “kingdom.” The scientific description branch contains those modules that answer questions like:

Who did the research? <creator>

In general terms, what is the research about? <abstract>

What are some of the key concepts that refer to the data? <keywords>

Where was the research done? <coverage>/ <geographicCoverage>

What time periods are covered in the data? <coverage>/<temporalCoverage>

What species are represented in this research/data? <coverage>/<taxonomicCoverage>

What methods were used? <methods>/<methodStep>, <sampling>,<qualityControl>

Is the data being documented part of a larger project <project>

The data representation branch contains modules that are used to describe:

What kind of data entity is it? <entity>/<dataTable>,<spatialVector>, <spatialRaster>

How is the actual file structured? <physical>/<dataFormat>

How would I retrieve this file? <physical>/<distribution>

Who will I allow access to the data? <access>

Anatomy of an EML Document

While most EML users will probably be using a tool such as [Morpho](#) or [Xylographa](#) or [Xanthoria](#) which take care of the formatting chores, that is, assuring that all of the metadata documents produced are valid EML documents, it helps to know what an eml document looks like “under the skin”.

While there are some people who take pleasure in documenting (providng metadata for research data), most people are more interested in doing the research than documenting it.

Cast of Characters:

Pat Ecologist: Pat is an ecologist. S/he knows that providing metadata is important, but has never done so in any organized way. Pat is computer literate, but has never worked with XML, “It’s like HTML right?”

Eco Informatics: Eco is an information manager with a strong interest in designing advanced software tools to help Pat with the analysis and modeling of ecologically interesting data.

emlMaven: emlMaven is an expert in EML

Arc Gis: Arc is a GIS (geographic information system) specialist. Arc often works with both Pat and Eco Informatics.

NSF: a mythical federal agency that gives out money, but always has strings attached.

Knowledge Network for Biocomplexity(KNB): KNB is a mythical association of ecological organizations. It has high standards and is committed to making ecological data available to researchers around the world. For the purpose of this drama,, Eco Informatics, is a representaive of KNB.

,Pat: I have just completed a research project. NSF tells me that I have to provide EML compliant metadata tp document my research, but I want to get back to the field as soon as possible. What is the minimum I have to do to produce a valid EML document?

emlMaven: The document in Figure 2 shows the the smallest valid EML dcoument.

<h2 style="color: green;">The smallest valid EML Document</h2>	
All XML documents start with this .	Most XML documents use UTF-8 also known as unicode
<code><?xml version="1.0" encoding="UTF-8"?></code>	
<code><eml:eml</code> must be unique for the SYSTEM	xmlns is short for XML Namespace.
<code>packageId="eml.1.1" system="knb"</code> <code>xmlns:eml="eml://ecoinformatics.org/eml-2.0.0"</code> <code>xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"</code> <code>xmlns:ds="eml://ecoinformatics.org/dataset-2.0.0"</code> <code>xsi:schemaLocation="eml://ecoinformatics.org/eml-2.0.0 eml.xsd"></code>	
<code><dataset></code> <code> <title>Rodents of the Southwest</title></code> <code> <creator></code> <code> <individualName></code> <code> <surName>Rodriguez</surName></code> <code> </individualName></code> <code> </creator></code> <code> <contact></code> <code> <individualName></code> <code> <surName>Chang</surName></code> <code> </individualName></code> <code> </contact></code> <code></dataset></code> <code></eml:eml></code>	These fields are the minimum necessary to produce a valid EML dataset document: <code><title></code> <code><creator></code> <code><contact></code>

Eco Informatics: That doesn't tell me very much. How can I design any useful tools with just a title and a last name?

Pat: I know I said minimal , but I know that NSF won't be satisfied with that. Even from my own selfish perspective, people won't know which Ecologist did this study. Ok. So, let me rephrase my question. What would an EML document look like that contains enough information that another ecologist would be able to understand.

emlMaven: That is a longer story. To tell this story requires some definitions and some detailed explanations. Before telling the full story, it will help to look more closely at the structure of the minimal EML document.

Documenting a dataset

XML Key terms: xmlns, schemaLocation, URI, URL

xmlns: This is a reference to an XML namespace. A namespace can be thought of as a way to specify where the element definitions come from. Since anyone can define their own tags, the use of namespaces allow XML processors to identify the specific vocabulary being used. A namespace is uniquely identified using a Uniform Resource Identifier (URI). A URI should be globally unique. Thus far, there is no method of enforcing this uniqueness, that is, there is no central registry of URIs. Often a URI is expressed as a URL (Uniform Resource Locator).

The schemaLocation

For the curious or obsessive: the technical specifications for URI, URL and the even more abstract URN (Uniform Resource Name) can be found at:

EML Key terms: eml, dataset, packageID, system, creator, contact, title

eml: As mentioned earlier, XML uses a hierarchical approach to representing information. All documents written to conform to Ecological Metadata Language standard (valid EML documents) have a top level element called <eml>. That is, all EML documents start with <eml:eml> and end with </eml:eml>.

packageID and system: A packageID is a unique identifier for a given system. The term **system** refers to the catalog or storage system that the metadata document is being contributed to. Examples of systems are:

KNB: the Knowledge Network for Biocomplexity. Any metadata document from an LTER, OBFS, or California NRS site should be considered to be contributed to KNB.

ESA Data Archive: The ESA data archive repository could be considered to be a system for EML purposes.

<dataset>: The term dataset means different things to different people. In EML, the term dataset refers to one or more data entities. The most common data entity is a **<dataTable>**. A data table is something that looks like this:

OBSERVATION_DATE	OBSERVATION_LOCATION	SPECIES	SPECIES_COUNT
2002-10-29	FCE_001	ALLIGATOR	1
2002-10-29	FCE_002	ALLIGATOR	2
2002-10-29	FCE_003	ALLIGATOR	0

Data tables are produced by people using software applications like text or word processors, and often saved as text files (ascii), or spreadsheets (Excel), statistical packages (SAS, SYSTAT, SPSS) by database systems (MS Access, My SQL, Oracle, MS Sql Server), or by certain kinds of sensors (data loggers).

In addition to data tables people using database applications may also produce a **<view>** or a **<storedProcedure>**.

People using GIS (geographical information system) applications generate both **<spatialVector>**, also referred to as boundary or shape files and **<spatialRaster>**. A spatialRaster is a geo-referenced image usually produced

by a camera on a satellite or other remote sensing device.

The final kind of data entity is **<otherEntity>**. An **<otherEntity>** is a data entity that cannot be represented by any of the previously defined data entity structures. A non-geo-referenced photograph is an **<otherEntity>**, e.g. a photograph of two different types of butterflies.

A **<creator>** is a structure for representing the owner of the data. A **<creator>** can be either a person, an organization, or an organizational role. Examples of organizational roles are: Executive Director, Department Administrator, Information Manager. A **<contact>** is a structure for representing the person, organization, or organizational role to contact regarding the use of the data. Without getting too deep into the language of XML Schema, we can say that both elements have the same structure, that is, both are **“of type responsibleParty”**.

At this point, we can expand the minimal EML document to look at a few examples of creators and contacts.

```
>
<creator>
  <individualName>
    <salutation>Dr.</salutation>
    <givenName>Juan</givenName>
    <givenName>Carlos</givenName>
    <surName>Ecologist</surName>
  </individualName>
  <address>
    <deliveryPoint>Department of Marine Ecology</deliveryPoint>
    <deliveryPoint>University of the Oceans</deliveryPoint>
    <deliveryPoint>1514 San Ysidro</deliveryPoint>
    <city>Seaside</city>
    <administrativeArea>LA</administrativeArea>
    <postalCode>78890</postalCode>
    <country>USA</country>
  </address>
  <phone>709-345-8970 x 254</phone>
  <electronicMailAddress>pat.ecologist@oceans.edu</electronicMailAddress>
</creator>
```

```
<creator>
  <organizationName>USDA</organizationName>
  <address>
    <deliveryPoint>US Department of Agriculture</deliveryPoint>
    <deliveryPoint>Mailstop 3654 </deliveryPoint>
    <deliveryPoint>25 Federal Plaza</deliveryPoint>
    <city>Washington</city>
    <administrativeArea>DC</administrativeArea>
    <postalCode>20025</postalCode>
    <country>USA</country>
  </address>
  <phone>phonetype="voice" 709-345-8970 x 254</phone>
  <phone>phonetype="fax" 709-345-8962</phone>
  <electronicMailAddress>dataservices@usda.gov</electronicMailAddress>
  <onlineUrl>http://www.usda.gov/ecoinformatics/</onlineUrl>
</creator>
```

```
<contact>
  <positionName>Information Manager</positionName>
  <address>
    <deliveryPoint>Oceans LTER</deliveryPoint>
    <deliveryPoint>Department of Marine Ecology</deliveryPoint>
    <deliveryPoint>University of the Oceans</deliveryPoint>
    <deliveryPoint>1514 San Ysidro</deliveryPoint>
    <city>Seaside</city>
    <administrativeArea>LA</administrativeArea>
    <postalCode>78890</postalCode>
    <country>USA</country>
  </address>
  <phone>709-345-8970 x 254</phone>
  <electronicMailAddress>pat.ecologist@oceans.edu</electronicMailAddress>
```

`<onlineUrl>http://oceansiter.lternet.edu</onlineUrl>`

`</contact>`

Things to notice and guidelines:

The element names may seem unfamiliar. You might be wondering, why did the developers of EML use such strange names? These elements have been taken from the International Standards Organization (ISO), schema for representing people: iso-party.

A person can have more than one `<givenName>`, but only one `<surname>`. EML does not have a specific tag for middle names. Since `<givenName>` is optional, `<surName>` would be used for “Cher” or “Madonna”

An address can have more than one `<deliveryPoint>`. A `<deliveryPoint>` is that part of an address that precedes the CITY. The most common `<deliveryPoint>` is a street address, Other examples are company names, department names, post office box, mailstop. Because XML does not recognize “carriage returns”, the way to enter that part of an address that precedes the city is to use a separate `<deliveryPoint>` element for each item that you would want to appear on a separate line when displayed on a web page or printed report. An `<administrativeArea>` is the tag that would be used in the United States to represent a STATE.

A `<creator>` or `<contact>` can have more than one address. Each address must be contained in its own set of `<address></address>` tags. (see below)

```
<creator>
  <individualName>
    <givenName>Sam</givenName>
    <surName>Ecologist</surName>
  </individualName>

  <address>
    <deliveryPoint>25 Oceans Avenue</deliveryPoint>
    <city>Oceans</city>
    <administrativeArea>CA</administrativeArea>
    <postalCode>98025</postalCode>
  </address>

  <address>
    <deliveryPoint>Department of Ecological Sciences</deliveryPoint>
    <deliveryPoint>University of the Oceans</deliveryPoint>
    <city>Oceans</city>
    <administrativeArea>CA</administrativeArea>
    <postalCode>98025</postalCode>
  </address>
</creator>
```


