

Mapping of the Encoded Archival Description DTD Element Set to the CIDOC CRM

Maria Theodoridou,
Martin Doerr

Institute of Computer Science, Foundation for Research and Technology – Hellas
Science and Technology Park of Crete
P.O. Box 1385, GR 711 10, Heraklion, Crete, Greece
maria@ics.forth.gr

Technical Report 289, ICS-FORTH, June 2001

Abstract: The CIDOC CRM is the first ontology designed to mediate contents in the area of material cultural heritage and beyond, and has been accepted by ISO TC46 as work item for an international standard. The EAD Document Type Definition (DTD) is a standard for encoding archival finding aids using the Standard Generalized Markup Language (SGML). Archival finding aids are detailed guides to primary source material which provide fuller information than that normally contained within cataloging records. As a potential international standard, the EAD DTD is maintained in the Network Development and MARC Standards Office of the Library of Congress in partnership with the Society of American Archivists.

This report describes the semantic mapping of the current EAD DTD Version 1.0 Element Set to the CIDOC CRM and its latest extension. This work represents a proof of concept for the functionality the CIDOC CRM is designed for.

Keywords: Ontology, Archival Finding Aid, Data Structure Mapping, Encoded Archival Description, CIDOC CRM

1 Introduction

The CIDOC Conceptual Reference Model (see [CRM1999], [Doerr99]), in the following only referred to as «CRM», is outcome of an effort of the Documentation Standards Group of the CIDOC Committee (see «<http://www.cidoc.icom.org>», «<http://cidoc.ics.forth.gr>») of ICOM, the International Council of Museums beginning in 1996. It is a domain ontology formulated in the form of an object-oriented semantic model aiming to solve the problem of semantic interoperability between museum data of various kinds and their relations to archive and library material.

After the model has found a stable form in 1998, and was accepted by CIDOC on the CIDOC Conference in Melbourne in this year, the Group decided that the model has

to undergo a series of practice test in order to verify its utility and completeness in a so-called «Correlation Test Project». Particular importance was seen in the demonstration that and how the CRM is able the mediate between current relevant standards of the domain, that are regarded relevant to the scope set for the CRM. This is done via sets of mapping rules, that define how to transform descriptions given in the structures of one standard into an equivalent description in the CRM with the same intended meaning.

This process has produced a set of extensions and improvements over the version 2.2 of the CRM, which latter has been accepted by ISO TC46, SC4 in September 2000, (see <http://www.niso.org/sc4ballt.html>). Results from the Correlation Test Project are intended to support the discussion and finalizing processes foreseen by ISO. So far, CIDOC members have been creating mappings to the AMICO [<http://www.amico.org/docs/dataspec.html>] data model, the Dublin Core Elements [DC],[Doerr2000], EAD [EAD] and the SPECTRUM data fields. In a meeting between representatives of the CIDOC Documentation Standards Group and other stakeholders in the domain in Agios Pavlos, Crete, in June 2000, experiences from these mappings have been discussed and a set of extensions for the CRM have been proposed, which are now incorporated in the CRM version 3.0 [CRM2001], still to be approved by the responsible working group.

This report presents the mapping from the EAD DTD Version 1.0 Element Set [EAD98] to the CRM carried out at ICS-FORTH, using CRM version 3.0. The EAD Document Type Definition (DTD) is a standard for encoding archival finding aids using the Standard Generalized Markup Language (SGML). Archival finding aids are detailed guides to primary source material which provide fuller information than that normally contained within cataloging records. As a potential international standard, the EAD DTD is maintained in the Network Development and MARC Standards Office of the Library of Congress in partnership with the Society of American Archivists. This mapping can be regarded as one of the necessary proofs of concept for the functionality the CRM is designed for.

2 Formalism

The EAD DTD is a set of rules for designating the intellectual and physical parts of archival finding aids so that information contained therein may be searched, retrieved, displayed, and exchanged in a predictable platform-independent manner.

The EAD DTD accomplishes three principal tasks [EAD98]:

- breaks down the content of finding aids into data fields or categories of information called "elements."
- identifies and defines any attributes that the DTD has associated with those elements. Attributes are characteristics or properties that further refine the element designation.
- defines the relationship between elements, specifying where the elements may be used and describing how they may be modified with attributes.

The purpose of the EAD DTD is to define the structure according to which instances will be created. However, the EAD DTD contains only a few required elements; the rest are optional. Thus, the amount of markup selected may vary from one repository to another, from "full tagging" to "less tagging", depending on intellectual and financial considerations. EAD instances can be described as "self-describing", a term used to describe semi-structured data where there is no separate description of the type or structure of data [Abiteboul].

Information is arranged hierarchically (sequence, nesting), descending from some root element, as if all subsequent information were part of the previous. There are three types of elements:

- a "wrapper element" indicates an element that cannot contain text directly; a second, nested element must be opened first. Wrapper elements designate sets of elements that comprise a distinctive section of a finding aid.
- a "generic element" refers to elements common to many kinds of documents that contain information that does not specifically relate to the structural parts of a finding aid.
- a "formatting element" indicates elements that can be used to invoke special character or text presentation, such as block quotes, chronologies, and emphasis.

On the other hand, CRM is a formal ontology in the form of a semantic model. A semantic model,

- declares each attribute value as an a priori independent entity in the universe of discourse and connects such values symmetrically by directed links, arcs, properties .
- The entity instances acquire identity independent from the composition of description elements in some data structure.
- The structure of a formal ontology is supposed to reflect the organization of the universe of discourse as we perceive it, rather than that of the document we create (e.g., if not given graphically, the presentation in a document will have to serialize a multiply connected graph in some way). It has an «ontological commitment» to the «underlying conceptualization» of its creators [Guarino98].

Our aim in the mapping of EAD DTD to CRM is not the description of the DTD structure in terms of the CRM but rather to create a semantic equivalent of the information about the world described in an EAD DTD instance. The wish for flexibility in markup has motivated the use of "floating" tags, i.e. tags that identify a category within a text string without any kind of relation, for example, a person without a role (the mixture of free text with embedded tags is also called a "mixed element"). Their use poses severe limits to the mapping that can be achieved. A CRM instance is a connected semantic graph, i.e. the connection between nodes is done by relations meaningful in the described world. If a date appears, let's say, in a SCOPECONTENT element, it cannot easily (automatically) be related to another meaningful node. This is not a problem of the CRM – no other semantic model could capture that case better – but a problem of the EAD itself. The TYPE attribute allows at least for recovering the kind of event associated.

The least specific relations in the CRM are about Stuff and Actors *participating* in an event, and intellectual *references* a Conceptual Object makes to anything. These relations already support powerful fundamental inferences, and are not difficult to be

maintained. We would recommend in any case, that a tagging scheme should allow automatically for identifying these relations. Anyhow, all “floating tags” in the EAD can properly be mapped to CRM entities, and at least be related to the finding aids themselves – at precisely the same level of precision the EAD tagging provides – even though a human reader may understand much more information that could be properly encoded in the CRM.

The EAD information, at the most basic level, consists of two segments [EAD98]:

1. a segment that provides information about the finding aid itself (its title, compiler, compilation date, etc.);
2. a segment that provides information about a body of archival materials (a collection, a record group, a fonds, or a series).

The EAD DTD splits the first segment into two second-level elements known as EAD Header <eadheader> and Front Matter <frontmatter>. The second segment, consisting of information about the archival materials, is contained within the third second-level element named Archival Description <archdesc>. All three of these second-level elements are contained within the first-level element named Encoded Archival Description <ead>, which wraps around the entire document. So, EAD is made up of three discrete parts [RLG]:

- 1 Information describing the finding aid itself contained in <eadheader>
- 2 Prefatory matter incorporating information useful for the display or publication of the finding aid, contained in <frontmatter>
- 3 The actual description of the archival materials and associated administrative and contextual information contained in <archdesc>

In the following, CRM entities are referred to by the unique id and name, eventually followed by a comment in parenthesis, like:

E7 Activity (of type «publication»)

CRM links are referred by the unique id of the **applicable** entity in the given context, followed by the unique id of the entity from which the link is **inherited**, the name of the link and the name of the referred entity, like:

E22 (E7) *was used for*: Activity

We use as name of the link only the part associated with the direction applicable for the given context. E.g. we present the link «E11 Modification.has produced (was produced by)» as

E22 (E11) *was produced by*: Modification

if we want to read it from the product rather than from the event.

We have chosen a mapping format, where for each EAD element (tag) we declare the correspondence of the **value** to a CRM entity, like:

titleproper = E35 Title (of type «proper»)

For a transfer from EAD to CRM, an identifier uniquely defined by the circumstances has to be created in order to be used by the respective CRM entity instance. In many cases, the contents of the mapped tag can be used to create a respective unique identifier e.g. the contents of the «**titleproper**» tag are used to create a respective unique identifier of an E35 entity instance. Attributes of the tag may also be used to derive additional CRM entities linked to the E35 entity instance:

titleproper = E35 Title (of type «proper»)
has note: String (of type «PUBSTATUS»)

where the value of attribute PUBSTATUS will be used to create an instance of the E62 String entity. The E35 instance is also assigned the type «proper» from a thesaurus.

The contents of a tag may also be used to derive additional CRM entities, if they can be parsed, e.g. <**titleproper pubstatus=«pub»**>**The Edgar Holden Papers**, <**date**>1978-1993</**date**></**titleproper**> may be used to find a date.

In the sequence we declare the correspondence of nested tags to CRM links, like:

ead = E31 Document
ead eadheader = E31 *is composed of*: Document

This means for a transfer from EAD to CRM that a property instance of type «is composed of» is created, that connects the instance of E31 «Finding AidX» representing our example finding aid to the above created instance of E31 «Ead HeaderX».

If the relation expressed by an EAD tag corresponds to a path with intermediate entities in the CRM, we mark it as (**JOIN**), like:

author = E65 (E5) *carried out by*: Actor – E39 (E1) *is identified by*: Actor
Appellation
where E65 Creation, E39 Actor, E72 Actor Appellation (JOIN)

This means, that for a transfer from the EAD author tag to CRM, an **intermediate** instance of E39 has to be created, with an identifier uniquely defined by the circumstances, like «Finding AidX-author1». In the sequence, it is inserted appropriately between the node «CreationX» and the Actor Appellation instance, e.g. «David Mathisen». This intermediate node may be bare of other attributes. Joins may be double or triple, requiring multiple intermediate nodes.

The «has note» link is extensively used in the mapping from EAD to CRM, for any information about a tag not formally described by an entity or property in the CRM, as e.g. paragraph, edition statement etc. It can also be used to capture the contents of a tag as a string which, with an appropriate parser, can be further analyzed into appropriate CRM entities as e.g. <**AUTHOR**>**Prepared by David Mathisen and**

Margaret H. McAleer with the assistance of Paul Colton*</AUTHOR>*. For fine-grain distinction of the semantic of such comments, the «has note» link has a type attribute.

The basic mapping mechanism is as follows: In the transformation from a semi-structured document to CRM, first a set of unique entity identifiers is created. These allow later to merge the compiled knowledge, and to retrieve connections not visible in single documents. Eventually, the same event may be registered multiply, if not enough evidence for its identity is found. Heuristics may be run to deal with the merging of such cases. Then the entity instances are connected by property instances, and the equivalent CRM instance is ready. This in turn can be transformed into many other forms of records without further ambiguities.

3 Mapping of the Encoded Archival Description <ead> Tag

An encoded finding aid instance as a whole is mapped to a CRM Document:

ead = E31 Document

which is composed of two E31 Documents representing the EAD Header and the Front Matter respectively and an E22 Man-Made Object that represents the Archival Description itself.

eadheader = E31 Document , E33 Linguistic Object
frontmatter = E31 Document
archdesc= E22 Man-Made Object

The relations among ead as a whole and its three distinct parts can be seen in Figure 1.

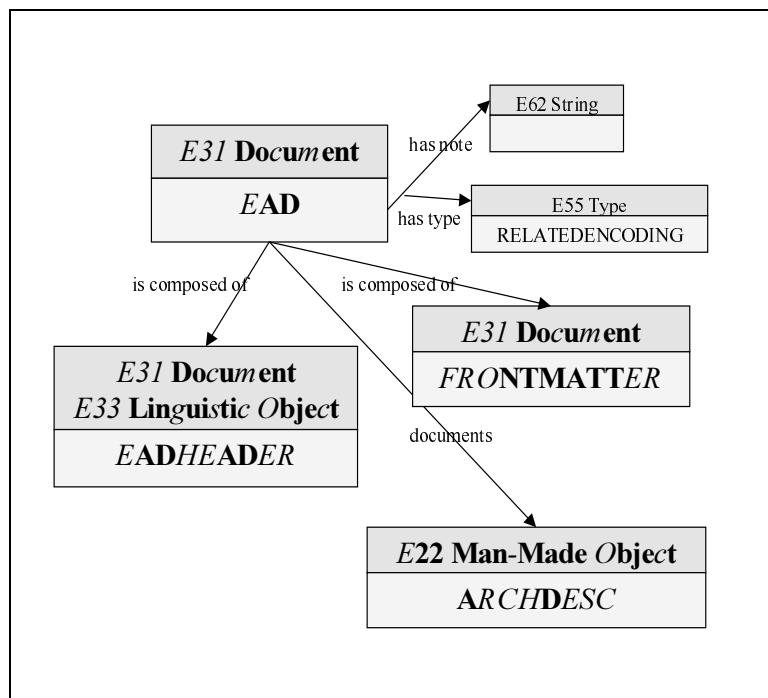


Figure 1: Mapping EAD to the CIDOC CRM

4 Generic elements

The EAD DTD allows for the nesting of elements to capture more detailed and specific description as desired. While there are certain elements that ought to appear in any finding aid, various intellectual and economic factors influence the depth and detail of analysis employed. Taking this into consideration, the encoding scheme is designed with a minimum of required elements, but allows for progressively more detailed and specific levels of description as desired. The Research Libraries Group (RLG) has produced the Recommended Application Guidelines for EAD [RLG98] which suggests a basic degree of uniformity in the creation of a valid EAD document. In our mapping from EAD to CRM we took into account these guidelines and we focused to this set of core data elements.

In this section we present the mapping of generic EAD elements to appropriate CRM entities. and in the following section we will present the mapping of required wrapper EAD elements to CRM entities.

4.1 DATE

Date is a generic EAD DTD element that contains a month, day, or year in any format. It is used to identify any dates that merit encoding, except for the creation dates of the described materials, which are instead tagged with the <unitdate> element.

date = E7 (E2) *has time-span*: Time-Span – E52 (E1) *is identified by*: Time-Appellation

where E7 Activity, E52 Time-Span, E49 Time-Appellation (JOIN)
and E7 Activity *has type*: TYPE
E52 Time-Span *has type*: CERTAINTY

attribute NORMAL maps to E52 Time-Span *begins at*: Time Primitive
ends at: Time Primitive

attribute NORMAL can also be mapped as an attribute of the link **has time span**

We can approach the above in a different way too. Instead of using The Time-Appellation we can use the attribute **begins at qualify** to put the value of date.

date = E7 (E2) *has time span*: Time-Span – E52 *begins at qualify*: String
where E7 Activity, E52 Time-Span (JOIN)
and E7 Activity *has type*: TYPE
E52 Time-Span *has type*: CERTAINTY

attribute NORMAL maps to E52 Time-Span *begins at*: Time Primitive
ends at: Time Primitive

The TYPE attribute is used to supply more specific designation, e.g. “life”, “flourish”, “depiction”, “publication”, “acquisition”. Whenever applicable, the E7 Activity can be substituted with a more specific subclass. For example if TYPE is “acquisition” then instead of E7 Activity we use the more specific class E8 Acquisition

4.2 UNITDATE

Unitdate describes the creation date of the described material. As such it can be mapped according to the above to the specific CRM class E12 Production.

unitdate = E12 (E2) *has time span*: Time-Span – E52 (E1) *is identified by*: Time-Appellation

where E12 Production, E52 Time-Span, E49 Time-Appellation (JOIN)
and E52 Time-Span *has type*: TYPE, CERTAINTY

attribute NORMAL maps to E52 Time-Span *begins at*: Time Primitive
ends at: Time Primitive

attribute NORMAL can also be mapped as an attribute of the link **has time span**

We can approach the above in a different way too. Instead of using The Time-Appellation we can use the attribute **begins at qualify** to put the value of unitdate.

unitdate = E12 (E2) *has time span*: Time-Span – E52 *begins at qualify*: String
where E12 Production, E52 Time-Span (JOIN)
E52 Time-Span *has type*: TYPE, CERTAINTY

attribute NORMAL maps to E52 Time-Span *begins at*: Time Primitive
ends at: Time Primitive

4.3 AUTHOR

Author is an element for names of institutions or individuals responsible for compiling the intellectual content of the finding aid. As such, it is mapped to an activity Creation that has participants:

author = E65 (E5) *had participants*: Actor – E39 (E1) *is identified by*: Actor Appellation

where E65 Creation, E39 Actor, E72 Actor Appellation (JOIN)
and attribute ID maps to E72 Actor Appellation

4.4 PUBLISHER

Publisher is the name of the party responsible for issuing or distributing the encoded finding aid. It is mapped to CRM in a similar to the author element way. Since there is no specific activity for publication in the CRM model, we use the *has type* link to associate the type "Publication" with the Activity.

publisher = E7 (E5) *carried out by*: Actor – E39 (E1) *is identified by*: Actor Appellation

where E7 Activity, E39 Actor, E72 Actor Appellation (JOIN)
and E7 Activity *has type*: Type[Publication]
and attribute ID maps to E72 Actor Appellation

4.5 ADDRESS

Address is a generic element for information about the place where someone or something is located and may be reached. The address element may be directly mapped to the CRM Address node.

address = E45 Address

has note: String

addressline = E62 String

4.6 CONTAINER and PHYSICAL LOCATION

Physical location identifies the place where the described material are stored, such as the name or number of the building, room, stack, shelf, or other tangible area.

physloc = E53 Place

Container is used to identify the cartons, boxes, reels, folders and other storage devices used to hold the described materials.

container = E53 Place

is identified by: Place Appellation (ID)

forms part of: Place (PARENT and physloc if it is available)

has type: Type (TYPE)

5 Information describing the finding aid

EAD Header provides descriptive information about the finding aid document and is mapped to a CRM Document. CRM allows multiple instantiation so since EAD Header may also contain information regarding the language, we map it to the CRM Linguistic Object too.

EAD.eadheader = E31 Document, E33 Linguistic Object

EAD → EAD.eadheader = E31 *is composed of*: Document

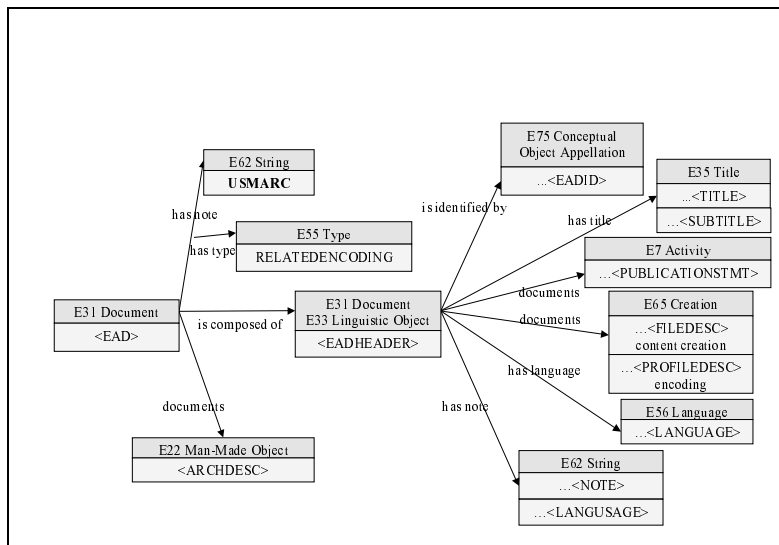


Figure 2: EADHEADER

<eadid> is a required subelement of <eadheader> that designates a unique identifier for a particular EAD finding aid document. Information about the naming scheme is given through the attributes SOURCE, SYSTEMID, TYPE. In the CRM mapping, the values of attributes are assigned through the **has type** link of the Conceptual Object Appellation node that maps the <eadid>:

eadid = E73 Conceptual Object Appellation

has type: TYPE, SYSTEMID, SOURCE

EAD.eadheader.eadid = E31 *is identified by*: Conceptual Object Appellation

<titleproper> is a required element within the <titlestmt> subelement of <filedesc>, part of <eadheader> while <subtitle> is a secondary or subsidiary name of the encoded finding aid that is subordinate to the main name encoded in <titleproper>. In the mapping to CRM we use the **has type** link to distinguish between the two titles:

titleproper = E35 Title

has type: proper

subtitle = E35 Title

has type: subtitle

EAD.eadheader.filedesc.titlestmt.titleproper = E31 *has title* : Title

EAD.eadheader.filedesc.titlestmt.subtitle = E31 *has title* : Title

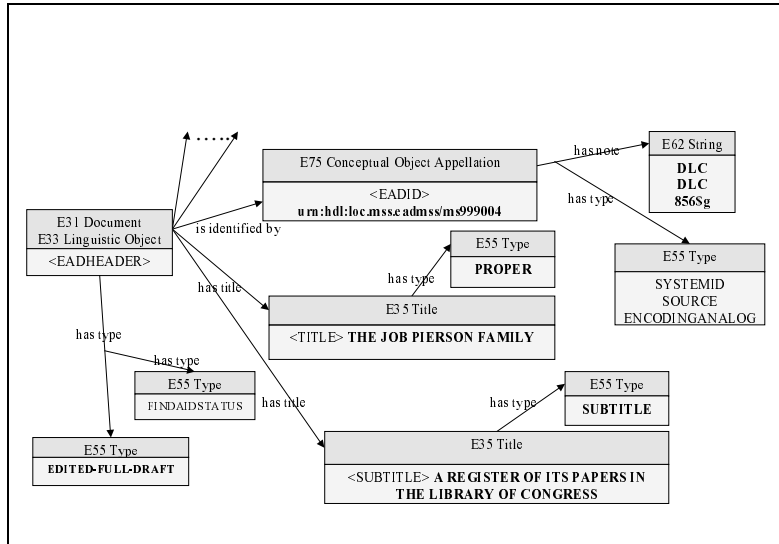


Figure 3: EADHEADER, EADID, TITLE, SUBTITLE

EAD.eadheader.filedesc.titlestmt.titleproper.date = E31 *documents*: Activity (see section 3.2 Date)

EAD.eadheader.filedesc.titlestmt.author = E31 *documents*: Creation (see section 3.3 Author)

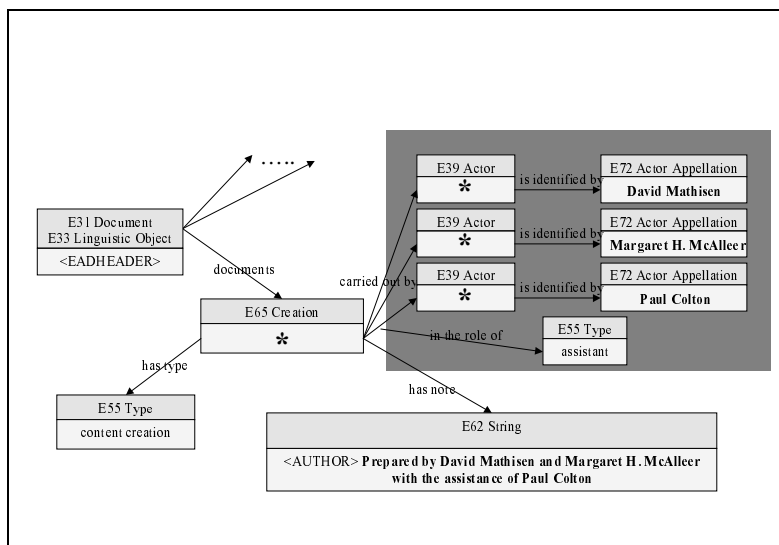


Figure 4: EADHEADER, AUTHOR

The publicationstmt may contain just text, laid out in paragraphs in which case it is mapped to a plain string

EAD.eadheader.filedesc.publicationstmt = E31 **has note**: String
(has type: Type[publication])

or it may include the publisher, address, date and num elements

publicationstmt = E7 Activity
 EAD.eadheader.filedesc.publicationstmt = E31 **documents**: Activity

publicationstmt.publisher = E7 **carried out by**: Actor
 publicationstmt.date = E7 **has time span**: Time Span
 publicationstmt.address = E7 **carried out by**: Actor – E39 **has contact points**: Address
 address = E45 (E39) **provides access to**: Actor – E39 (E7) **performed**: Activity
 publicationstmt.num = E41 Appellation

The Activity with type “publication” should be the same instance in all the above mappings.

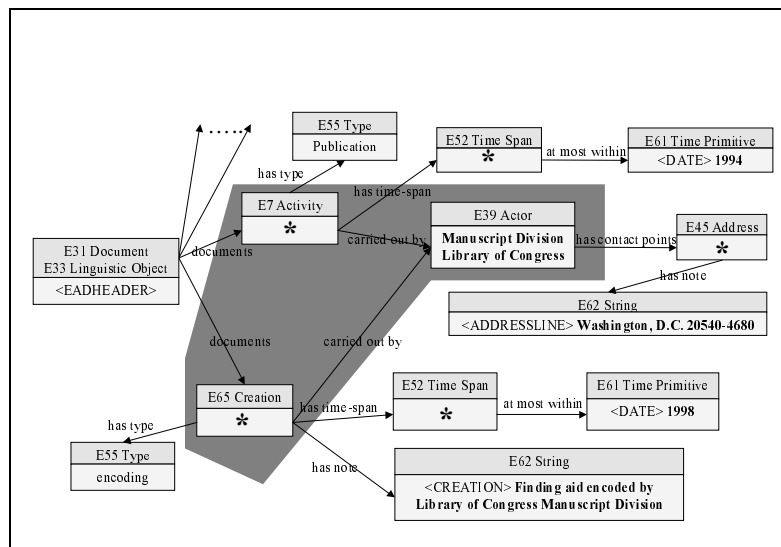


Figure 5: EADHEADER, PUBLICATIONSTMT

The notestmt and editionstmt elements are optional elements with no strict way of giving the information. We believe that the information they contain is useful as display information only and as such it is mapped to the **has note** attribute of the EAD.eadheader document.

EAD.eadheader.filedesc.notestmt.note = E31 **has note**: String
(has type: Type[note])

EAD.eadheader.filedesc.editionstmt = E31 **has note**: String

(*has type*: Type[edition])

EAD.eadheader.filedesc.seriesstmt = E31 (E76) *forms part of*: Document
(*has type*: Type[series])

EAD.eadheader.filedesc.seriesstmt.titleproper = E31 *has title* : Title

EAD.eadheader.filedesc.seriesstmt.num = E31 *is identified by*: Conceptual
Object Appellation

EAD.eadheader.filedesc.seriesstmt.p = E31 *has note*: String

language = E56 Language

EAD.eadheader = E33 Linguistic Object (multiple instantiation)

EAD.eadheader.profiledesc.langusage = E33 *has note*: String
(*has type*: Type[language])

EAD.eadheader.profiledesc.langusage.language = E33 *has language*:
Language

EAD.eadheader.profiledesc.creation = E31 *documents*: Creation
has note: String

EAD.eadheader.profiledesc.creation.date = E65 (E2) *has time-span*: Time-
Span – E52 (E1) *is identified by*: Time-Appellation

EAD.eadheader.revisiondesc = E31 *documents*: Activity
(*has type*: Type[revision])

EAD.eadheader.revisiondesc.change.item = E31 *documents*: Activity – E7
has note: String

EAD.eadheader.revisiondesc.change.date = E7 (E2) *has time-span*: Time-
Span – E52 (E1) *is identified by*: Time-Appellation

EAD.eadheader.revisiondesc.list = E31 *documents*: Activity – E7 *has note*:
String

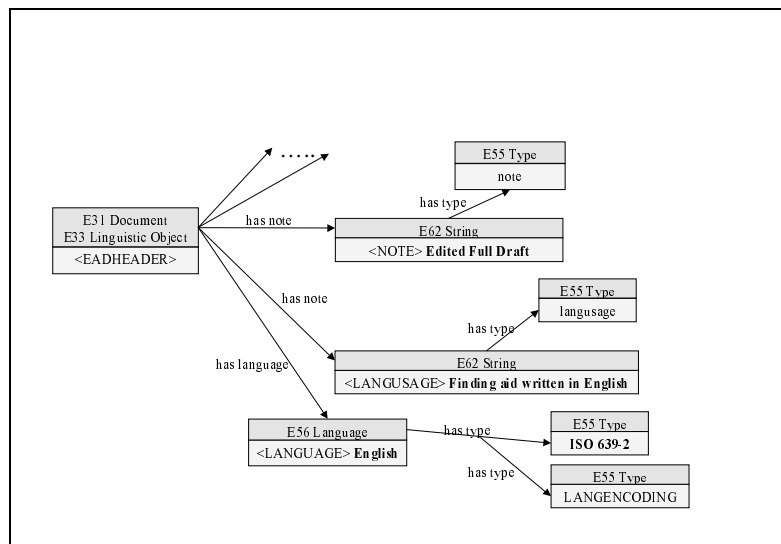


Figure 6: EADHEADER, LANGUAGE

6 Information necessary for display

EAD.frontmatter = E31 Document

has note: String

(*has type*: Type[frontmatter])

where E62 String contains all the frontmatter

EAD → EAD.frontmatter = E31 *is composed of*: Document

7 Description of the archival materials and associated administrative and contextual information

EAD.archdesc = E22 Man-Made Object (ARCHDESC)

EAD → EAD.archdesc = E31 *documents*: Man-MadeObject

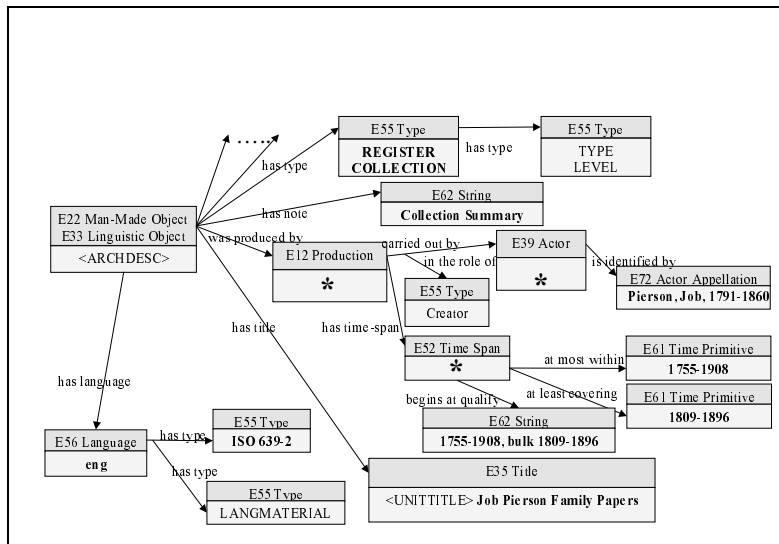


Figure 7: ARCHDESC

7.1 Descriptive Identification <did>

Descriptive Identification (did) is a required wrapper element that bundles other elements identifying core information about the described materials in either the Archival Description <archdesc> or a Component <c>. The first <did> in <archdesc> is the high level <did> and describes the collection as a whole. In the CRM mapping we refer to an instance of <did> as DID and to the high level <did> instance as DID1

did.head = E62 String

did.unittitle = E35 Title

Unitdate describes the creation date of the described material . In order to map unitdate to CRM we introduce a Production activity that has a Time-Span

did.unitdate = E12 (E2) **has time span**: Time-Span – E52 (E49) **is identified by**:
Time-Appellation
where E12 Production, E52 Time-Span, E49 Time-Appellation (JOIN)
and E52 Time-Span has type: TYPE, CERTAINTY
attribute NORMAL maps to E52 Time-Span **at most within**: Time Primitive

attribute NORMAL can also be mapped as an attribute of the link **has time span**

We can approach the above in a different way too. Instead of using The Time-Appellation we can use the attribute **begins at qualify** to put the value of unitdate.

did.unitdate = E12 (E2) **has time span**: Time-Span – E52 **begins at qualify**: String
where E12 Production, E52 Time-Span (JOIN)
E52 Time-Span has type: TYPE, CERTAINTY
attribute NORMAL maps to E52 Time-Span **at most within**: Time Primitive

did.unitid = E42 Object Identifier
where E42 Object Identifier has type: TYPE, LABEL
COUNTRYCODE, REPOSITORYCODE ???????

did.origination = E11 (E7) **carried out by**: Actor (in the role of: LABEL)
where E11 Modification, E7 Activity

Physical description bundles information about the appearance or construction of the described materials. Some of these aspects of appearance can be mapped directly to CRM (dimensions, technique, marks). Some, as color, style, do not have a straight forward mapping. The Attribute assignment and Type classes could be used to create all the appropriate types of appearance. However, we estimate that this information is not to be used for retrieval and we choose to map all this information in the has note attribute of the Physical Feature class.

did.physdesc = E26 **has note**: String
where E26 Physical Feature

Repository is the institution or agency responsible for providing the intellectual access to the materials being described. In the EAD Tag Library, in the <repository> tag, it is suggested to use <physloc> to identify the custodian and <repository> to designate the intellectual caretaker when it is clear that the physical custodian does not provide intellectual access. However, in the <physloc> description, the tag is described as the place where the described materials are stored. In my understanding, they use the place as a way to identify an institution. In the mapping to CRM, I choose to keep the two tags conceptually distinct, and map repository to Actor and physloc to Place.

did.repository = E39 Actor
did.physloc = E53 Place

did.abstract = E62 String
did.note = E62 String

No natural language interpretation

Syntactic analysis

What level of analysis ???

ARCHDESC → did1.head = E22 **has note**: String

ARCHDESC → did1.unittitle = E22 **has title**: Title

ARCHDESC → did1.unitdate = E22 (E11) **was produced by**: - E12 (E2) **has time**

span: Time-Span – E52 (E49) **is identified by**: Time-Appellation

where E12 Production, E52 Time-Span, E49 Time-Appellation (JOIN)

and E52 Time-Span has type: TYPE, CERTAINTY

attribute NORMAL maps to E52 Time-Span **at most within**: Time Primitive

attribute NORMAL can also be mapped as an attribute of the link **has time span**

We can approach the above in a different way too. Instead of using The Time-Appellation we can use the attribute **begins at qualify** to put the value of unitdate.

ARCHDESC → did1.unitdate = E22 (E11) **was produced by**: - E12 (E2) **has time**

span: Time-Span – E52 **begins at qualify**: String

where E12 Production, E52 Time-Span (JOIN)

E52 Time-Span has type: TYPE, CERTAINTY

attribute NORMAL maps to E52 Time-Span **at most within**: Time Primitive

ARCHDESC → did1.unitid = E22 **is identified by**: Object Identifier

where E42 Object Identifier has type: TYPE, LABEL

COUNTRYCODE, REPOSITORYCODE ???????

ARCHDESC → did1.origination =

E22 (E11) **was produced by** – E11 (E7) **carried out by**: Actor (in the role of: LABEL)

where E11 Modification, E7 Activity (JOIN)

ARCHDESC → did1.physdesc = E22 **has note**: String

has type: Type[extent]

note + parts

ARCHDESC → did1.repository =

E22 (E5) **has current keeper**: Actor (**in the role of**: intellectual caretaker)

ARCHDESC → did1.physloc = E22 **has current location**: Place

ARCHDESC → did1.abstract = E22 **has note**: String

has type: Type[abstract]

ARCHDESC → did1.note = E22 **has note**: String

has type: Type[note]

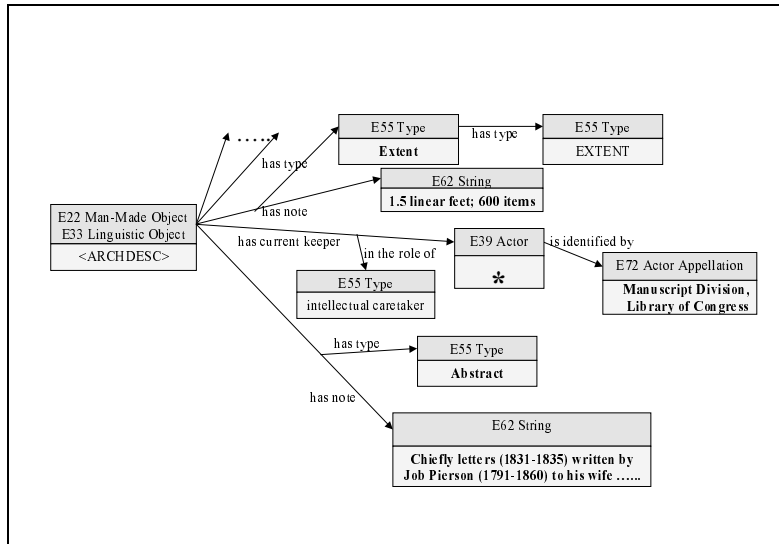


Figure 8: ARCHDESC

7.2 Controlled Access Headings

ARCHDESC → controlaccess = E22 **has note:** String
has type: Type[Search Terms]
has type: Type[Names/Subjects/Occupation/...]

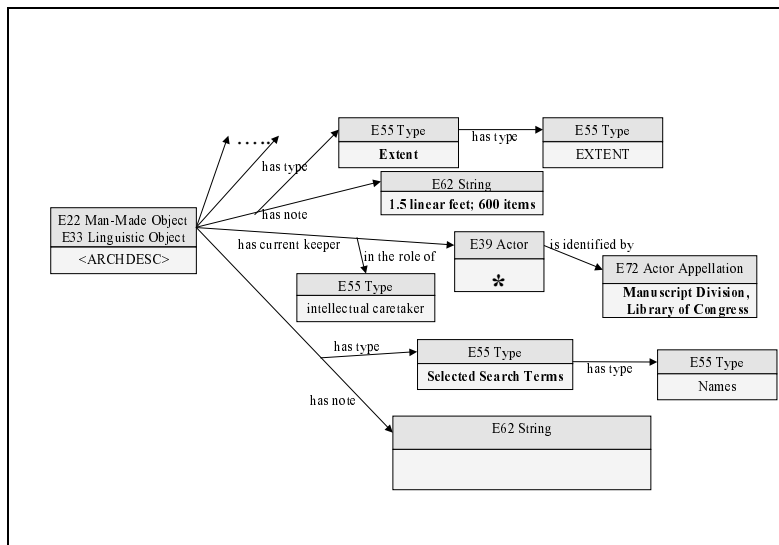


Figure 9: ARCHDESC

7.3 Description of Subordinate Components

Components, both numbered (01-12) and unnumbered provide information about the content, context and extent of a subordinate body of materials. Each component element identifies an intellectually logical section of the described materials. The physical filing separations between components do not always coincide with the intellectual separations.

9 References

[ARCHON] ARCHON - A Multimedia System for Archival, Annotation and Retrieval of Historical Documents. Project site:

<http://www.ics.forth.gr/proj/isst/Projects/archon.html>

[CRM2001] Nick Crofts, Ifigenia Dionissiadou, Martin Doerr, Matthew Stiff (editors), "Definition of the CIDOC object-oriented Conceptual Reference Model ", March 2001 (version 3.0). See locations:

http://cidoc.ics.forth.gr/docs/crm_version_3.0.rtf

[CRM1999] "Definition of the CIDOC object-oriented Conceptual Reference Model", produced by the ICOM/CIDOC Documentation Standards Group, editors: Nick Crofts, Ifigenia Dionissiadou, Martin Doerr, Matthew Stiff, September 1999, version 2.1 http://cidoc.ics.forth.gr/docs/CRM_version_2_1.rtf

[DC] The "Dublin Core Metadata Initiative", project site: <http://www.dublincore.org>

[Doerr99] Martin Doerr and Nicholas Crofts "Electronic Esperanto: The Role of the Object Oriented CIDOC Reference Model", Proc. of the ICHIM'99, Washington, DC, September 22-26, 1999.

[Doerr2000] Martin Doerr, "Mapping of the Dublin Core Metadata Element Set to the CIDOC CRM" , Technical Report, 274, ICS-FORTH, Heraklion, Crete, July 2000.

http://cidoc.ics.forth.gr/docs/dc_to_crm_mapping.rtf

[Guarino98] Guarino N., Formal Ontology and Information Systems. In N. Guarino (ed.), Formal Ontology in Information Systems. Proc. of the 1st International Conference, Trento, Italy, 6-8 June 1998. IOS Press

[EAD] Encoded Archival Description, Official Web Site <http://www.loc.gov/ead>

[EAD98] EAD Tag Library for Version 1.0, June 1998 <http://lcweb.loc.gov/ead/tglib>

[RLG] RLG Recommended Application Guidelines for EAD, 1998

<http://www.rlg.org/rlgead/guidelines.html>

[Theo1998] M. Theodoridou and M. Doerr, "Classifying Historical Documents" Proc. of the Applications of Information Technologies to Biblical Studies, OSI-HESP SUMMER SCHOOL, July 27-August 7, 1998, Sofia, Bulgaria.

<http://www.ics.forth.gr/proj/isst/Publications/paperlink/CHD/html/chd.w6.html>

Appendix A - An Example of an EAD Instance

```
<!DOCTYPE EAD PUBLIC "-//Society of American Archivists//DTD ead.dtd (Encoded Archival
Description (EAD) Version 1.0)//EN"

[

<!ENTITY lcseal SYSTEM "http://lcweb2.loc.gov/sgmlstd/panorama/lcseal.gif" NDATA GIF>

]>

<EAD RELATEDENCODING="USMARC">

<EADHEADER LANGENCODING="ISO 639-2" FINDAIDSTATUS="EDITED-FULL-DRAFT">

<EADID SYSTEMID="DLC" SOURCE="DLC"
ENCODINGANALOG="856$g">urn:hdl:loc.mss.eadms/ms999004</EADID>

<FILEDESC><TITESTMT>

<TITLEPROPER>THE JOB PIERSON FAMILY</TITLEPROPER>

<SUBTITLE>A REGISTER OF ITS PAPERS IN THE LIBRARY OF CONGRESS</SUBTITLE>

<AUTHOR>Prepared by David Mathisen and Margaret H. McAleer <LB>with the assistance of
Paul Colton</AUTHOR></TITESTMT>

<PUBLICATIONSTMT><DATE>1994</DATE><PUBLISHER>Manuscript Division<LB> Library of
Congress<EXTPTR

ENTITYREF="lcseal" SHOW="EMBED"
ACTUATE="AUTO"></PUBLISHER><ADDRESS><ADDRESSLINE>Washington, D.C. 20540-
4680</ADDRESSLINE></ADDRESS></PUBLICATIONSTMT>

<NOTESTMT><NOTE><P>Edited Full Draft</P></NOTE></NOTESTMT></FILEDESC>

<PROFILEDESC><CREATION>Finding aid encoded by Library of Congress<LB> Manuscript
Division,<DATE>1998</DATE></CREATION>

<LANGUSAGE>Finding aid written in
<LANGUAGE>English.</LANGUAGE></LANGUSAGE></PROFILEDESC></EADHEADER>

<ARCHDESC TYPE="REGISTER" LEVEL="COLLECTION" LANGMATERIAL="eng">

<DID><HEAD>Collection Summary</HEAD><ORIGINATION LABEL="Creator"><PERSNAME
ENCODINGANALOG="100">Pierson, Job, 1791-1860</PERSNAME></ORIGINATION>

<UNITTITLE LABEL="Title" ENCODINGANALOG="245">Job Pierson Family Papers

<UNITDATE LABEL="Dates" TYPE="INCLUSIVE" ENCODINGANALOG="260">1755-1908, bulk 1809-
1896</UNITDATE></UNITTITLE>

<PHYSDESC LABEL="Extent"><EXTENT ENCODINGANALOG="300">1.5 linear feet; 600
items</EXTENT></PHYSDESC>

<REPOSITORY LABEL="Repository" ENCODINGANALOG="852"><CORPNAME><SUBAREA>Manuscript
Division,</SUBAREA> Library of Congress</CORPNAME></REPOSITORY>

<ABSTRACT LABEL="Abstract" ENCODINGANALOG="520">Chiefly letters (1831-1835)
...</ABSTRACT></DID>
```

<CONTROLACCESS><HEAD>Selected Search Terms</HEAD>

<CONTROLACCESS><HEAD>Names:</HEAD>

<PERSNAME SOURCE="LCNAF" ENCODINGANALOG="600">Jackson, Andrew, 1767-1845</PERSNAME>

.
.
.

</CONTROLACCESS>

<CONTROLACCESS><HEAD>Subjects:</HEAD>

<SUBJECT SOURCE="LCSH" ENCODINGANALOG="650">Conversion</SUBJECT>

.
.
.

<GEOGNAME SOURCE="LCSH" ENCODINGANALOG="651">New York (State)--Politics and government</GEOGNAME>

<GEOGNAME SOURCE="LCSH" ENCODINGANALOG="651">United States--Politics and government--1829-1837</GEOGNAME>

<GEOGNAME SOURCE="LCSH" ENCODINGANALOG="651">Washington (D.C.)--Social life and customs--19th century</GEOGNAME>

</CONTROLACCESS>

<CONTROLACCESS><HEAD>Occupations:</HEAD>

<OCCUPATION SOURCE="OTHERSOURCE" OTHERSOURCE="itoamc" ENCODINGANALOG="656">Clergy</OCCUPATION>

.
.
.

</CONTROLACCESS>

</CONTROLACCESS>

<ADMININFO><HEAD>Administrative Information</HEAD>

<ACQINFO><P>The papers of the Job Pierson Family were purchased ... </P></ACQINFO>

<USERRESTRICT><P>The status of copyright ... </P></USERRESTRICT>

<PROCESSINFO><P>The papers acquired ... </P></PROCESSINFO>

<P>A description of the Pierson Papers ... </P>

</ADMININFO>

<BIOGHIST><HEAD>Biographical Notes</HEAD>

<BIOGHIST>

<HEAD>Job Pierson (1791-1860)</HEAD>

<CHRONLIST>

<CHRONITEM><DATE>1791, Sept. 23</DATE><EVENT>Born, <GEOGNAME>Bridgehampton,
N.Y.</GEOGNAME></EVENT></CHRONITEM>

<CHRONITEM><DATE>1811</DATE><EVENT>Graduated, <CORPNAME>Williams College,</CORPNAME>
<GEOGNAME>Williamstown, Mass.;</GEOGNAME> studied law in
<GEOGNAME>Salem</GEOGNAME> and <GEOGNAME>Schaghticoke,
N.Y.</GEOGNAME></EVENT></CHRONITEM>

<CHRONITEM><DATE>1815</DATE><EVENTGRP><EVENT>Admitted to bar; practiced law with
<PERSNAME>Herman Knickerbocker</PERSNAME> in <GEOGNAME>Rensselaer County,
N.Y.</GEOGNAME></EVENT>

<EVENT>Married <PERSNAME>Clarissa Taintor
Bulkeley</PERSNAME></EVENT></EVENTGRP></CHRONITEM>

.
.
.

</CHRONLIST></BIOGHIST>

<BIOGHIST>

<HEAD>Job Pierson (1824-1896)</HEAD>

<CHRONLIST>

<CHRONITEM><DATE>1824, Feb. 3</DATE><EVENT>Born, <GEOGNAME>Schaghticoke,
N.Y.</GEOGNAME></EVENT></CHRONITEM>

.
.
.

</CHRONLIST></BIOGHIST></BIOGHIST>

<SCOPECONTENT><HEAD>Scope and Content Note</HEAD>

<P>The Job Pierson Family Papers span the years 1755-1908, with the bulk of material
dating from 1809 to 1896. The collection consists

...

</SCOPECONTENT>

<DSC TYPE="IN-DEPTH"><HEAD>Container List</HEAD>

<C01><DID><CONTAINER TYPE="BOX">1</CONTAINER><UNITTITLE>Journal and diaries, 1843,
1849-1896 </UNITTITLE><PHYSDESC><EXTENT>(4 folders)</EXTENT></PHYSDESC></DID></C01>

<C01><DID><UNITTITLE>Correspondence</UNITTITLE></DID>

<C02><DID><UNITTITLE>May 1803-Apr. 1832 </UNITTITLE><PHYSDESC><EXTENT>(3 folders)
<REF

TARGET="overcor" SHOW="REPLACE" ACTUATE="USER">See also
Oversize</REF></EXTENT></PHYSDESC> </DID></C02>

<C02><DID><CONTAINER TYPE="BOX">2</CONTAINER><UNITTITLE>May 1832-Mar. 1835
</UNITTITLE><PHYSDESC><EXTENT>(8 folders) <REF

TARGET="overhouse" SHOW="REPLACE" ACTUATE="USER">See also
Oversize</REF></EXTENT></PHYSDESC></DID></C02>

<C02><DID><CONTAINER TYPE="BOX">3</CONTAINER><UNITTITLE>May 1835-Feb. 1883, n.d.
</UNITTITLE><PHYSDESC><EXTENT>(2 folders)</EXTENT></PHYSDESC></DID></C02></C01>

<C01><DID><UNITTITLE>Legal and financial papers</UNITTITLE></DID>

<C02><DID><UNITTITLE>Correspondence, 1810-1877, n.d.</UNITTITLE></DID></C02>

<C02><DID><UNITTITLE>Miscellany</UNITTITLE></DID>

<C03><DID><UNITTITLE>1755-1839</UNITTITLE></DID></C03>

<C03><DID><CONTAINER TYPE="BOX">4</CONTAINER><UNITTITLE>1840-1886, n.d.
</UNITTITLE><PHYSDESC><EXTENT>(2 folders)</EXTENT></PHYSDESC></DID></C03></C02>

<C02><DID><UNITTITLE>Wills, 1799-1885</UNITTITLE></DID></C02></C01>

<C01><DID><UNITTITLE>Miscellany, 1831-1889, n.d.</UNITTITLE></DID></C01>

<C01><DID><CONTAINER TYPE="BOX">5</CONTAINER><UNITTITLE>Scrapbook, ca. 1864-1908,
n.d.</UNITTITLE></DID></C01>

<C01><DID><CONTAINER TYPE="BOX">OV 1</CONTAINER><UNITTITLE>Oversize</UNITTITLE></DID>

<C02><DID><UNITTITLE ID="overcor">Correspondence, 1809-1832, n.d.</UNITTITLE></DID></C02>

<C02><DID><UNITTITLE ID="overhouse">House of Representatives seating chart,
n.d.</UNITTITLE></DID></C02></C01>

</DSC></ARCHDESC></EAD>