Archives and Manuscripts, 2015
Vol. 43, No. 2, 120–130, http://dx.doi.org/10.1080/01576895.2014.1000347

Transforming archival education

Jinfang Niu*

Jinfang Niu is an assistant professor at the School of Information, University of South Florida. She received her PhD from University of Michigan, Ann Arbor. Prior to that, she worked as a librarian at the Tsinghua University Library for three years. Her current research focuses on electronic records and digital curation.

This paper discusses limitations of the current archival description standard Encoded Archival Description (EAD) and proposes two solutions to overcome these limitations. One solution is to modify the current EAD schema based on the entity–relationship model defined in the Australian Series System. The other solution is to replace EAD with another standard, the Open Archives Initiative Object Reuse and Exchange (OAI-ORE), which can be used to produce more flexible archival descriptions in linked data format.

Keywords: archival description; Australian Series System; EAD; OAI-ORE

Introduction

Advancing technologies continually transform records and records creation context, and also give rise to new philosophies, methodologies, and tools that push forward the evolution of archival description. There have been many discussions about the need to adapt traditional archival description methodologies to the digital world. It is widely recognised today that archival descriptions should not be created *post hoc* after inactive records are transferred to archival institutions. Instead, they should start from the records creation stage and document the recordkeeping process.[1] The traditionally fixed aggregates 'fonds' or 'series' for managing paper records are not adequate for managing database records,[2] thus the units of archival description need to be redefined for new types of digital records. In addition, semantic web technology also calls for archival descriptions to be created in or converted into linked data format. Among all the ideas and technologies that have transformed archival description, the Australian Series System has played a vital role.

The Australian Series System was originally proposed by Peter Scott in 1964 as a solution to overcome the difficulties in applying the record group concept to Australian archival management practices.[3] It is very different from the European and American archival description approaches where fond or record group is the top level of description and a series has to be fixed within one fond/record group although recordkeeping reality requires otherwise. The Australian Series System is essentially a relational model that separates the description of records from the description of records creation context (provenance) and allows one series to be linked to multiple provenances. Over the years, the Australian Series System has been enriched greatly. Traditionally, provenance description in the Series System has focused on agents. Later, functions were also

*Email: Jinfang@usf.edu

identified and described as different kinds of records creation contexts. In the 1990s, the original Series System was developed into a complex entity–relationship model called SPIRT,[4] which defines three groups of entities, attributes of those entities, and relationships among those entities, that applies to both current records management and archival description context.

The SPIRT conceptual model, which is the current form of the Australian Series System, has had a significant impact on recordkeeping metadata and archival description. It has been used as the conceptual foundation for the international standard for recordkeeping metadata ISO 23081-2[5] and the Australian archival description rules *Describing Archives in Context: A Guide to Australasian Practice* (DAC).[6] The definition of the three groups of entities in the Series System has also caused the development of separate description standards for different groups of entities. For example, the General International Standard Archival Description (ISAD(G)) was created for the description of records, agents, and functions in one integrated standard. Later, two new standards, the International Standard Archival Authority Record for Corporate Bodies, Persons and Families (ISAAR (CPF)) and the International Standard for Describing Functions (ISDF), were created for the description of agents and functions, respectively. Encoded Archival Description (EAD) was created for the description of all three groups of entities defined in the SPIRT conceptual model. Later, the newer metadata standard, Encoded Archival Context (EAC), was created specifically for agent description.

Despite these remarkable impacts, in the author's opinion, the Series System could be more influential than it currently is. Although it has been used as the theoretical basis for DAC, it does not seem to have affected EAD, the most widely used metadata schema for archival description, nor does it seem to have influenced Describing Archives: A Content Standard (DACS),[7] the American archival description rules. To deepen the impact of the Series System and further transform archival description, this paper will discuss how EAD can be modified to be consistent with the entity–relationship model defined in the Australian Series System, and how the Open Archives Initiative Object Reuse and Exchange (OAI-ORE) can be used to produce more flexible archival description in linked data format.

Problems with EAD

Since its creation about two decades ago, EAD has played a vital role in publishing archival finding aids online. Despite its success, EAD has been criticised on various grounds. EAD was created to convert paper finding aids into online format, thus it imitates the content and layout of paper finding aids and employs a document-centric approach which is very different from the record-centric approach used by most other metadata schemas. An EAD finding aid contains not only metadata, but also formatting and structural information such as lists and paragraphs. This incompatibility with other metadata standards makes it difficult to convert EAD finding aids to other metadata formats and causes interoperability issues.[8] In addition, users, especially novice users, have trouble understanding the archival jargon used in EAD finding aids,[9] and become lost in the complex hierarchical structure of finding aids.[10] Yakel suggested solving this problem by educating users and employing virtual reference assistance in an online environment.[11] While this may work where users are using a search interface only for online archival finding aids, it does not sound realistic when archival resources are aggregated and mixed with various other kinds of web resources. Some researchers have also pointed out that the monolithic EAD files with a deep hierarchical structure make it difficult to directly access particular components without accessing the whole hierarchy first.[12]

Notwithstanding these criticisms and the shifting technology environment, the most recent revisions of EAD by the Technical Subcommittee for Encoded Archival Description and the Schema Development Team are mostly minor adjustments and do not address most of these problems.[13] The following section proposes more fundamental modifications of EAD from a document-centric tool to a metadata schema consistent with the entity–relationship model in the Australian Series System. In addition, the modified EAD schema attempts to address the criticisms mentioned above.

Transforming EAD

Make EAD a records-specific metadata schema

Although EAD was not created based on an entity–relationship model, it contains elements and attributes for describing all the three groups of entities defined in the Series System. For example, the elements Abstracts, Access Restriction, and Appraisal Information are all created to describe the records entities. The element Biography and History describes agents and the Function element describes the activities and processes that generated records. To be consistent with the entity–relationship model defined in the Series System, EAD can be made a metadata schema for records entities only by removing elements describing agents and functions. Descriptions of agents can be created using the EAC metadata schema and then linked with associated EAD records. Similarly, functions can be described using a metadata schema created based on ISDF and then linked to associated EAD records.

EAD also includes elements for entities not defined in the Series System, such as file plan, index, finding aids and bibliography of works created based on archival records. The <index> element can be removed because back-of-the-volume indexes are not necessary with the full-text search capability of electronic finding aids. Elements for each of the other entities can also be removed from EAD if there is a more suitable metadata schema for describing that kind of entity. For example, it might be more appropriate to use Dublin Core or Metadata Object Description Schema (MODS) to describe bibliographic works created based on archival records. Whether elements for these other entities are removed or retained, descriptions of each of the entities should be created in a separate metadata record and then linked with associated EAD records, instead of being included in the same EAD document. This would allow for clear identification of different entities, which makes the modified EAD schema consistent with an entity–relationship model.

The current EAD schema defines many elements that have similar meanings and makes the encoding unnecessarily complex and less interoperable with other metadata standards. Here are some examples:

- <unitdate> (creation date of the described materials) and <date> (any other dates);
- <titleproper> (title of the finding aid), <title> (title of a work listed in a finding aid) and <unittitle> (title of the described materials);
- <langmaterial> (language of the archival materials), <language> (language of the finding aid) and <langusage> (language represented in an encoded finding aid);

- <name> (unspecified name), <corpname> (corporate name), <famname> (family name), <geogname> (geographic name) and <persname> (personal name);
- <eadid> (identifier for the EAD finding aid) and <unitid> (identifier for a particular unit within the EAD finding aid).

These similar elements are defined in order to describe the same attributes of different entities. In the entity–relationship model, entities, which are objects of descriptions, are defined separately and explicitly, thus the same attributes (metadata elements) can be used to describe different entities. For example, the <id>, <title>, and <language> attributes can be assigned to both the record and finding aid entities. In other words, the similar elements listed above can be merged in the modified EAD schema.

Dissolve the hierarchical description of records into multiple separate and linked metadata records

When Peter Scott created the original Series System, he replaced record group with series as the primary level of description and allowed one series to be linked with multiple provenances. In this method, each series has its own separate description rather than being included in the description of a record group. Chris Hurley pointed out that this disaggregation of record group description can be applied to the series level and below.[14] This means that any unit, down to the level of each individual record in an archival hierarchy, can be described as a separate entity rather than being included in its parent entity. If this idea is applied to modify EAD, the multi-level deep hierarchical structure in EAD can be dissolved into multiple separate and linked metadata records. In other words, each node in the hierarchy can be described by one metadata record which is then linked to the metadata records for other nodes. This would allow each node in an archival hierarchy to be clearly identified as a separate entity and make archival description consistent with an entity–relationship model. It would also allow for more flexible linking than the hierarchical structure in current EAD finding aids. Any node (archival unit) can be linked with other nodes through any kind of relationship, such as hierarchical, poly-hierarchical, sequential and network relationships. In addition, each separate metadata record can be referenced, indexed, updated, or accessed directly. This addresses the criticism that the deep hierarchical structure of EAD finding aids makes it difficult to directly access particular components without accessing the whole hierarchy first.

In the current EAD schema, the <eadheader> element and its sub-elements are defined for describing the EAD finding aid. In other words, they are meta-metadata. After an EAD document is dissolved into multiple separate and linked metadata records, each individual metadata record can have its own meta-metadata to record who created that metadata and at what time. In other words, the <eadheader> section for a current EAD finding aid will also be dissolved into many separate meta-metadata records. In the case that several metadata records share the same metadata information, they can be linked to the same meta-metadata record.

The feasibility of using multiple separate and interlinked metadata records to represent multi-level structures has been researched or tested in the following studies or practices. Jinfang Niu analysed how multiple separate and linked MARC records can be used to represent multi-level whole-part structure.[15] The FRBRised cataloging software created by the VTLS company shows that multiple separate and linked MARC records can be used to represent the Work-Expression-Manifestation-Item (WEMI) structure defined in Functional Requirements for Bibliographic Records (FRBR).[16] Bountouri and Gergatsoulis proposed to create one separate MODS record for each node in the archival hierarchy.[17] The hierarchical structure will be represented through a series of nested <relatedItem> elements and its Type attribute with the value 'constituent' or 'host'. Prom and Habing created a solution in which one EAD file is dissolved into multiple Dublin Core (DC) metadata records, each for a node in the archival hierarchy.[18] To show the context, each DC record points to its parent and child node in the original EAD finding aids using the DC terms 'IsPartOf' and 'HasPart'.[19] In this approach, only the content of the EAD finding aids is converted into DC records. The structure and context relies on the accessibility and persistency of the original EAD finding aid. Ferro and Silvello proposed the Nested SeTs for Object hieRarchies (NESTOR) approach for representing the multi-level structure of archival finding aids.[20] Similar to the solution of Prom and Habing, this approach converts metadata content in EAD finding aids into multiple DC records. The difference is that it uses OAI sets to preserve the hierarchical structure. One OAI set is created for each internal node of the tree and the nesting of the sets, showing the hierarchical structure of the finding aids.

There are some potential drawbacks to using multiple linked metadata records to represent the archival hierarchy. Without the monolithic EAD documents that give users a sense of the entire archival collection and its internal structure, archival information retrieval software may need to automatically analyse the links among multiple metadata records and visually display the hierarchical structure on a user interface. With independent metadata records for each component in the archival hierarchy, the inheritance of metadata across levels is not as easily seen as in a monolithic EAD document. Therefore, some contextual metadata from the parent level(s) may need to be repeated on a child level, in case the user does not follow the links among the metadata records to view the relationship between the child and parent. As reported by Bountouri and Gergatsoulis,[21] in converting EAD finding aids into multiple linked MODS records, they inserted information not included in the original EAD description of the component to remedy this potential issue. For example, they inserted role terms in the creator element (such as 'Photographer') and the type of the resource ('still image'). On the other hand, the less visible metadata inheritance is sometimes preferable because metadata inheritance is sometimes not appropriate. As reported by Johnston,[22] index terms assigned for an archival collection may not apply to some particular components in the collection. This is also true for the inheritance of creator information because the creator of a collection may be different from the creator of an individual record in the collection.

Make relationships explicit

A prominent feature of an entity–relationship model is that relationships between entities are explicitly defined. EAD is able to support various kinds of relationships. However, these relationships are represented implicitly rather than through explicit linking or relational terms. For example, the hierarchical structure of an archival collection is represented through the nesting of the element <c>; the sequential order among sibling components in a hierarchical structure is represented through recording the components in a certain order; the relationships between the description of the finding aid (the <eadheader> element and its sub-elements) and the description of records (the <archdesc> element and its sub-elements), are all indicated by including them in the same EAD document. These implicit relationships rely on human interpretation to understand and are difficult to process automatically.

These relationships can be made explicit by using relational terms. For example, the hierarchical relationship between a series and an individual record it contains can be represented using a term 'has part'; the relationship between the description of the finding aid and the description of records can be indicated by the term 'describes'; and the relationship between records and the creator of the records can be indicated by the term 'created by'. This modification of EAD requires the creation of many relation terms. The explicit description of relationships will make it easier to convert EAD descriptions to linked data because Resource Description Framework (RDF) triples are constructed through relationships.

Make EAD a content-only standard

The entity–relationship model defines entities, attributes of entities, and relationships between entities. In other words, it only defines data content, and no formatting or displaying information is included. This is also what most other metadata standards, such as Dublin Core and MARC, usually do. The display of finding aids can be dealt with by external software or style sheets. To be compliant with the entity–relationship model and to be more interoperable with other metadata schemas, all the formatting elements and attributes in EAD can be removed, such as the <head> element for displaying headings, the <chronlist> element for displaying chronology lists, and the multiple elements for displaying tables. The Type attribute of the <dsc> (Description of Subordinate Components)[23] element allows for the recording of three different views of the components of an archival collection. This is very display oriented. The Type attribute can be removed and then all data recorded in the combined view retained.

Many wrapper elements in EAD do not have their own meanings and are only useful for grouping their sub-elements or for providing a heading to display the content of their sub-elements. Examples of these elements are:

- <did> (Descriptive Identification);[24]
- <archref> (Archival Reference);[25]
- <dsc> (Description of Subordinate Components);
- <filedesc> (File Description);[26]
- <profiledesc> (Profile Description);[27]
- <titlestmt> (Title Statement);[28]
- <publicationstmt> (Publication Statement);[29]
- <revisiondesc> (Revision Description).[30]

In fact, in the mapping of EAD to other metadata formats, these wrapper elements are usually not mapped because they have no meaning.[31] They can be removed and their sub-elements are used directly. The example below illustrates removed wrapper elements.

Example 1.[32]
~~<dsc type="combined">~~
<c01 level="series">
~~<did>~~
<unittitle>Activities, </unittitle>
<unitdate type="inclusive">1965-1971</unitdate>
<physdesc><extent>0.3 linear ft.</extent></physdesc>
~~</did>~~
~~</dsc>~~

The modified EAD schema described above is more consistent with the entity–relationship model defined in the Australian Series System and will be more interoperable with other metadata schemas. Since the recordkeeping metadata standard ISO 23081 is also based on the same entity–relationship model, the modified EAD schema would support a Records Continuum approach of archival description, whereby archival description can be inherited more smoothly from the recordkeeping stage.

Notwithstanding these benefits, the modified EAD still assumes that one single metadata schema is used for records description. In the following section, the use of OAI-ORE for archival description will be discussed. OAI-ORE will reform archival description even further by tossing away this assumption, and meanwhile produce archival descriptions consistent with the entity–relationship model in the Series System.

Using OAI-ORE for archival description

OAI-ORE is a linked data-based standard for describing aggregations of web resources (http://www.openarchives.org/ore/). It can be used to organise all kinds of resources, not necessarily archival materials. However, since it focuses on controlling aggregates, which is an important function of archival description, it has been proposed to replace EAD as the data model for archival description,[33] and has been used by the Europeana digital library as the mapping target for EAD finding aids.[34] Archival descriptions created using OAI-ORE are in linked data format, and thus do not need to be converted when they are published on the semantic web and become part of the linked open data cloud. This will avoid information loss that could happen during format conversion.

OAI-ORE defines a very small vocabulary that only includes four classes (Aggregation, AggregatedResource, Proxy, and ResourceMap) and eight relations (aggregates, isAggregatedBy, describes, isDescribedBy, lineage, proxyFor, proxyIn, and similarTo). These terms are primarily for describing the relationships between aggregations and aggregated resources. Thus, OAI-ORE only intends to describe resources, not agents and functions. Archival descriptions created using OAI-ORE can be linked with EAC records and descriptions of associated functions in the same way that EAD records are linked with descriptions of the other two groups of entities in the Series System.

In fact, OAI-ORE does not even define terms for describing attributes of resources, such as the title, creator, or subject content of resources. To describe the attributes of its classes, OAI-ORE recommends some external vocabularies, such as Dublin Core elements, Dublin Core terms, Friend of a Friend terms, RDF terms, and RDF schema terms. It also allows for the reuse of classes and relationships in these recommended vocabularies and other domain vocabularies. For example, in the implementation of OAI-ORE in the Europeana Data Model at the Europeana digital library, metadata elements from Simple Knowledge Organization Systems, Resource Description and Access

Group 2 elements are used.[35] This provides a large degree of flexibility and makes it interoperable with other resources descriptions.

In OAI-ORE, the descriptions of aggregations, aggregated resources, and the relationships between them are included in Resource Maps, which are similar to the <archdesc> (archival description) area in EAD. OAI-ORE also allows for the description of the Resource Map itself, for example, who published it and when it was most recently modified. This is metadata of metadata, similar to the Header section in an EAD finding aid. Different from EAD, however, the description of a Resource Map in OAI-ORE is separate from the Resource Map and explicitly linked to the Resource Map through the relation term 'describes/isDescribedBy'. This clearly separates the description of different entities and makes the relationship among the entities explicit.

Similar to EAD, OAI-ORE focuses on hierarchical relationships. In OAI-ORE, a multi-level structure is represented using recursive nesting of aggregations. This is similar to the recursive nesting of <c> elements in EAD. Different from EAD, which describes the nesting of all components in one monolithic document, the OAI-ORE description of each aggregation is separated into one Resource Map. The recursive nesting of aggregations is represented through external linking to other Resource Maps. This is similar to the separate and linked metadata records approach which was mentioned earlier in this paper. This would allow for direct access to each aggregation in the hierarchy. Using the term 'RDF: type', the types of OAI-ORE aggregations can be specified. Although EAD also can specify the type of components through its Level attribute, the types are usually archival aggregations such as fonds, record groups, series and so on. OAI-ORE allows more varieties of types by reusing any existing vocabularies. For example, it can reuse the DCMI Types Vocabulary at http://dublincore.org/documents/dcmi-type-vocabulary/. The type of an aggregation in OAI-ORE can be any archival aggregations, such as fonds, collections, series, or any other kind of aggregation that makes sense to general users, such as a learning object or photo album.

OAI-ORE defines only two levels of entities (aggregations and aggregated resources) but can represent an unlimited number of levels in a hierarchical description through recursive nesting. DAC defines series and item as two records entities. These are only two of many possible levels in an archival collection. In ISO 23081-2, six entities are defined for records: Item, Transaction, File, Series, Archive, and Archives. The definition of levels in OAI-ORE is more flexible than the rigid definitions in DAC and ISO 23081. In addition, compared with archival jargon such as series and transactions, aggregation and aggregated resources are easier to understand for general users. Thus, it might be better for the Australian Series System to adopt this definition.

Through the use of proxies, OAI-ORE can describe sequential relationships among aggregated resources.[36] OAI-ORE has also been used to describe the derivative relationships among multiple versions or multiple stages of development of scientific data and publications.[37] Since OAI-ORE allows for the reuse of relational terms from other vocabularies, potentially it can represent any relationship between entities, and these relationships are all explicitly recorded.

The capacity to accommodate terms from other existing vocabularies allows OAI-ORE to provide more specialised descriptions of archival records. As we know, archival records are identified based on their functions, independent of genres and formats. An archival collection contains records in various formats and genres, such as academic papers, emails, webpages, maps, audio, and visual materials. These different kinds of records are best described using specialised metadata schemas. OAI-ORE can reuse these specialised metadata formats designed for particular kinds of records and thus provide richer descriptions with more specificity. In contrast, EAD was invented to describe archival collections as aggregations. It defines a fixed and limited number of metadata elements that apply to all levels in an archival hierarchy and thus cannot describe the special features of individual records. In fact, many digital repositories and information systems have used EAD to describe the structure of collections only and use other metadata standards to describe individual items, such as those reported by Bountouri, et al.[38] and Zhang and Mauney.[39]

Conclusion

The current EAD schema can be modified based on the entity–relationship model defined in the Australian Series System. The modified EAD schema will be more interoperable with other metadata schemas and, easier to convert to linked data format, as well as, supporting a Records Continuum approach of archival description. Notwithstanding these benefits, using OAI-ORE for archival description seems a better way to transform archival description, because archival descriptions created using OAI-ORE are also compliant with the entity–relationship model in the Series System, even more interoperable with other metadata schemas through the reuse of elements in existing metadata schemas, and are in linked data format so that no conversion is needed when archival descriptions are published on the semantic web, and become part of the linked open data cloud.

Endnotes

1. Chris Hurley, 'The Hunting of the Snark: Searching for Digital Series', 2011, available at <*http://www.descriptionguy.com/images/WEBSITE/hunting-of-the-snark-search-for-digital-series.pdf*>, accessed 31 January 2014.
2. ibid.
3. PJ Scott, 'The Record Group Concept: A Case for Abandonment', *American Archivist*, vol. 29, no. 4, 1966, pp. 493–504.
4. Sue McKemmish, Glenda Acland, and Barbara Reed, 'Towards a Framework for Standardising Recordkeeping Metadata: The Australian recordkeeping metadata Schema', *Records Management Journal*, vol. 9, no. 3 1999, 173–98.
5. ISO 23081-2: Information and documentation – Records management processes – Metadata for records. Part 2: Conceptual and implementation issues, 2009, first edition.
6. Australian Society of Archivists Committee on Descriptive Standards, *Describing Archives in Context: A Guide to Australasian Practice*, Australian Society of Archivists, Sydney, 2008.
7. Society of American Archivists, *Describing Archives: A Content Standard (DACS), Second Edition, 2013*, available at <*http://files.archivists.org/pubs/DACS2E-2013.pdf*>, accessed 20 January 2015.
8. J Riley and K Shepherd, 'A Brave New World: Archivists and Shareable Descriptive Metadata', *American Archivist*, vol. 72, no. 1, 2009, pp. 91–112.
9. E Yakel, 'Encoded Archival Description: Are Finding Aids Boundary Spanners or Barriers for Users?' *Journal of Archival Organization*, vol. 2, nos 1–2, 2004, pp. 63–77.
10. CJ Prom, 'User Interactions with Electronic Finding Aids in a Controlled Setting', *American Archivist*, vol. 67, no. 2, 2004, pp. 234–68.
11. Yakel.
12. N Ferro and G Silvello, 'A Methodology for Sharing Archival Descriptive Metadata in a Distributed Environment', in B Christensen-Dalsgaard, D Castelli, L Ammitzb, B Jurik and J Lippincott (eds), *ECDL 2008*, Springer, Heidelberg, 2008, Lecture Notes in Computer Science, vol. 5173, pp. 268–79; G Silvello, 'Building a Distributed Digital Library System Enhancing the Role of Metadata', in *BCS-IRSG Symposium: Future Directions in Information*

Access, 2008, pp. 46–53. Available at <*http://irsg.bcs.org/proceedings/proceedings-fdia2007.pdf*>, accessed 15 January 2015.

13. Technical Subcommittee for Encoded Archival Description (TS-EAD) and the Schema Development Team (2013a), 'EAD Beta Schema Released – Comments Welcome', available at <*http://www2.archivists.org/groups/technical-subcommittee-on-encoded-archival-description-ead/ead-beta-schema-released-comments* ->, accessed 9 August 2013.
14. Hurley.
15. Jinfang Niu, 'Hierarchical Relationships in the Bibliographic Universe', *Cataloging and Classification Quarterly*, vol. 51, no. 5. 2013, pp. 473–90.
16. V Chachra and J Espley, 'Differentiating Libraries Through Enriched User Searching; FRBR as the Next Dimension in Meaningful Information Retrieval', 2004, American Library Association annual conference presentation, available at <*http://www.ala.org/alcts/sites/ala.org.alcts/files/content/events/pastala/annual/04/Chachra.pdf*>, accessed 10 May 2012.
17. L Bountouri and M Gergatsoulis, 'Interoperability Between Archival and Bibliographic Metadata: An EAD to MODS Crosswalk', *Journal of Library Metadata*, vol. 9, no. 1, 2009, pp. 98–133.
18. CJ Prom and TG Habing, 'Using the Open Archives Initiative Protocols with EAD', in G Marchionini and W Hersch (eds), *Proc. 2nd ACM/IEEE Joint Conference on Digital Libraries*, ACM Press, New York, 2002, pp. 171–80.
19. CJ Prom, 'Reengineering Archival Access Through the OAI Protocols', *Library Hi Tech*, vol. 21, no. 2, pp. 199–209.
20. N Ferro and G Silvello, 'The NESTOR Framework: How to Handle Hierarchical Data Structures', in M Agosti, J Borbinha, S Kapidakis, C Papatheodorou and G Tsakonas (eds), *ECDL 2009*, Springer, Heidelberg, 2009, Lecture Notes in Computer Science, vol. 5714, pp. 215–26.
21. Bountouri and Gergatsoulis.
22. P Johnston, 'The Data Transformation', 2011, available at <*http://blogs.sussex.ac.uk/salda/2011/05/16/the-data-transformation/*>, accessed 31 January 2014.
23. A wrapper element that bundles information about the hierarchical groupings of the materials being described.
24. A required wrapper element that bundles other elements identifying core information about the described materials in either archival description <archdesc> or a component <c>.
25. A reference element that provides a citation and/or an electronic link to separately described archival materials of special interest.
26. A required sub-element of the <eadheader> that bundles much of the bibliographic information about the finding aid, including its author, title, subtitle and sponsor (all in the <titlestmt>), as well as the edition, publisher, publishing series and related notes (encoded separately).
27. An optional sub-element of the <eadheader> that bundles information about the creation of the encoded version of the finding aid, including the name of the agent, place and date of encoding. The <profiledesc> element also designates the predominant and minor languages used in the finding aid.
28. A required wrapper element within the <filedesc> portion of <eadheader> that groups information about the name of an encoded finding aid and those responsible for its intellectual content.
29. A wrapper element within the <filedesc> portion of <eadheader> for information concerning the publication or distribution of the encoded finding aid, including the publisher's name and address, the date of publication and other relevant details.
30. An optional sub-element of the <eadheader> for information about changes or alterations that have been made to the encoded finding aid.
31. P Gaitanou, L Bountouri and M Gergatsoulis, 'Automatic Generation of Crosswalks Through CIDOC CRM', in *Metadata and Semantics Research*, Springer, Berlin and Heidelberg, 2012, pp. 264–75.
32. This example is adapted from the example on the webpage <*http://www.loc.gov/ead/tglib/elements/dsc.html*>, accessed 29 January 2014.
33. D Kaplan, A Sauer and E Wilczek, 'Archival description in OAI-ORE', *Journal of Digital Information*, vol. 12, no. 2, 2011, available at <*https://journals.tdl.org/jodi/index.php/jodi/article/view/1814/1769*>, accessed 15 January 2015.

34. Europeana Digital Library, 'Europeana Data Model Primer', 2013, available at <*http://pro.europeana.eu/documents/900548/770bdb58-c60e-4beb-a687-874639312ba5*>, accessed 15 January 2015.
35. Europeana.
36. C Lagoze, HV Sompel, ML Nelson, S Warner, R Sanderson and P Johnston, 'Object Re-Use & Exchange: A Resource-Centric Approach', *Arxiv preprint*, arXiv:0804.2273v1, 2008, available at <*http://arxiv.org/abs/0804.2273*>, accessed 15 January 2015.
37. A Pepe, M Mayernik, C Borgman and H Van de Sompel, 'From Artifacts to Aggregations: Modeling Scientific Life Cycles on the Semantic Web', *Journal of the American Society for Information Science and Technology*, vol. 61, no. 3, 2010, pp. 567–82.
38. L Bountouri, C Papatheodorou, V Soulikias and M Stratis, 'Metadata Interoperability in Public Sector Information', *Journal of Information Science*, vol. 35, no. 2, 2009, pp. 204–31.
39. J Zhang and D Mauney, 'When Archival Description Meets Digital Object Metadata: A Typological Study of Digital Archival Representation', *American Archivist*, vol. 76, no. 1, 2013, pp. 174–95.