Ten blog jest trzecim w serii poświęconej standardom metadanych używanym w archiwach.



EAD (Encoded Archival Description) jest standardem stworzonym specjalnie w celu zakodowania pomocy archiwalnych. Z tego powodu jest on pewnego rodzaju hybrydą. Z jednej strony stara się odzwierciedlić sposób, w jaki pracują archiwiści tworząc pomoce archiwalne, z drugiej stara się wprowadzić dyscyplinę i dokładność niezbędną do elektronicznej obróbki dokumentu. W wyniku mamy sporo dowolności w umiejscowieniu danych, co ułatwia pracę archiwiście a jednocześnie utrudnia wymianę danych. W nowej wersji EAD (EAD3), która jest w przygotowaniu od kilku lat, spodziewane jest zmniejszenie tych dowolności.

Reguły i zasady tworzenia pomocy archiwalnych zawarte są w osobnych dokumentach. Oprócz zasad międzynarodowych - ISAD(G) - są również zasady tworzone w różnych krajach, jak np. DACS w USA, które są podobne ale posiadają często subtelne różnice. EAD jest formą zapisu tych danych w postaci zrozumiałej przez człowieka ale także nadającej się do obróbki komputerowej. Jak wszystkie nowoczesne standardy metadanych, wyrażony jest w XML i składa się z serii etykiet, takich jak **<ead>**, które mieszczą się w innych, wraz z regułami ich umieszczania i regułami dotyczącymi ich zawartości.

<ead>

Podstawowym dokumentem w EAD jest jednostka zawierająca typowo jedną kolekcję (zespół archiwalny, fonds), zapisaną w postaci pliku zgodnego z zasadami XML. Dokument musi mieć następującą strukturę:

```
 <ead>
[zawartość]
</ead>
```

W dalszej części omówimy elementy które są tu oznakowane skrótowo jako [zawartość].

Standard EAD jest raczej prosty, choć zawiera wiele szczegółów. Dokument ead musi zawierać dwa elementy (z ich zawartością), <eadheader> i <archdesc>.

```
 <ead>
<eadheader>[zawartość]</eadheader>
<archdesc>[zawartość]</archdesc>
</ead>
```

Dla przypomnienia - elementy w XML oznakowane są ostrymi nawiasami **<** i **>** a zawartość elementu, np <ead>, to wszystko co mieści się między <ead> i </ead>, patrz “ Wstęp do standardów metadanych ”.

<eadheader>

Pole <eadheader> przeznaczone jest na informacje o pomocy archiwalnej - w odróżnieniu od samego zasobu - tytuł, autora i inne detale, a także informację o instytucji publikującej (tworzącej) ten zapis. Jest to nieco nadmiarowe jeśli cały dokument EAD jest tą pomocą archiwalną, ale pozwala np. na zapisanie danych autora - archiwisty w odróżnieniu od autora kolekcji archiwalnej. W polu tym jest też miejsce na unikalny identyfikator archiwum, w elemencie <eadid>, patrz artykuł " Unikalne identyfikatory w archiwach i bibliotekach ". Pole <eadheader> ma w intencji twórców tego standardu zawierać informacje pozwalające na wypełnienie danymi strony tytułowej albo okładki tomu pomocy archiwalnych; jeśli się to nie udało, można użyć dodatkowego pola o nazwie <frontmatter>.

<archdesc>

Pole <archdesc> jest przeznaczone na wszystkie informacje na temat opisywanego zasobu archiwalnego. Ead nie stawia ograniczeń co do zasobu, może to być zarówno cały zespół (fonds) jak i jego wyodrębniona część. Ponieważ jednak struktura EAD jest hierarchiczna, najlepiej jest zamykać w tym polu jeden zespól, w którym w sposób naturalny mieszczą się jego pod-elementy (serie, pod-serie, jednostki itp.). Element posiada atrybut “level” czyli poziom, a więc dla całej kolekcji pole to wygląda tak:

```
<archdesc level="fonds"> [zawartośc] </archdesc>
```

Pole <archdesc> jest obwolutą (wrapper) w której umieszczamy przeróżne informacje w osobnych pod-polach. Inne elementy które służą tylko jako obwoluty to <did> (descriptive identification), <dsc> (description of subordinate components> i <c> (component). Większość elementów i atrybutów jest jednak używanych do zapisania informacji o zasobie, t.j. posiada tekst a nie tylko inne elementy, np. dla zapisania streszczenia czy tytułu.

Co mieści się w polu <archdesc>? Pierwszym elementem który napotkamy to <did> czyli opisowa identyfikacja zasobu. W elemencie tym zawarte będą wybrane informacje o zasobie jako całości, przykładowo będzie tu miejsce na streszczenie, ale nie na dzieje twórcy zasobu. Najważniejsze pod-elementy w <did> to:

<unittitle> - tytuł tego zasobu
<unitid> - identyfikator zasobu (często zwany sygnaturą)
<unitdate> - daty zasobu - zakres dat.
<langmaterial> - informacja o języku lub językach użytych w tym zasobie.
<abstract> - streszczenie, opis zasobu.
<physdesc> - różne informacje dotyczące fizycznego opisu materiału, jego rozmiarów itp.

Inne, równie ważne elementy nie są umieszczane w obwolucie <did> a niejako “luzem”. Są to przykładowo:

<bioghist> - dzieje twórcy, biografia, historia twórcy materiału
<scopecontent> - szczegółowy opis zawartości, w formie narracji, z założenia szerszy od streszczenia
<index> - obwoluta zawierająca indeksy
<controlaccess> - obwoluta na hasła tematyczne itp.
<dao> (digital archival object) - obwoluta na obiekty cyfrowe

Wreszcie mamy tu miejsce na inwentarz zawartości zasobu. Ponieważ archiwum jest zorganizowane w sposób hierarchiczny, będziemy mieli pod-elementy - elementy składowe głównego zespołu, pod-pod-elementy - elementy składowe pod-elementów i tak dalej. Używamy tu dwóch obwolut:

<dcs> - obwoluta zawierająca spis i opisy elementów podrzędnych w <archdesc>.

<c> - obwoluta zawierająca opis jednego elementu podrzędnego (który może zawierać wewnątrz dalsze elementy podrzędne).

Dla przejrzystości w czytaniu (ale nie w obróbce maszynowej) możemy te poziomy również numerować, np. <c01>,<c02> itp.

Tak więc drugi element <ead> czyli <archdesc>, będzie wyglądał w skrócie tak:

- <archdesc>
- <did>
- <unittitle>...</unittitle>
- <unitid>...</unitid>
- <unitdate>...</unitdate>
- <langmaterial>...</langmaterial>
- <abstract>...</abstract>
- <physdesc>...</physdesc>
- ...
- </did>
- <bioghist>...</bioghist>
- <index>...</index>
- <controlaccess>...</controlaccess>
- <dao>...</dao>
- ...
- <dsc>...</dsc>
- </archdesc>
-

Wielokropki oznaczają miejsce na zawartość i/lub kolejne elementy. Zawartość ta ma często własna strukturę, z własnymi etykietami XML.

<dsc>

Element <dsc> jest obwolutą w której mieści się spis, wraz z zawartością, elementów podrzędnych zasobu opisanego w <archdesc>. Tak więc, jeśli najwyższy poziom opisuje zespół archiwalny (<archdesc level="fonds">), w elemencie <dsc> umieścimy wszystkie elementy podrzędne. Mogą to być pod-zespoły (subfonds), serie, jednostki (file) albo nawet pojedyncze dokumenty (item). Nie mieszamy jednak poziomów - jeśli podzieliliśmy cały zespół archiwalny na serie, to w <dsc> umieszczamy tylko serie.

Każdy element podrzędny (składnik) mieści się w swojej własnej obwolucie <c>. A więc przykładowo dla zespołu który ma 3 serie będziemy mieli

- <dsc>
- <c level="series">...</c>
- <c level="series">...</c>
- <c level="series">...</c>

- </dsc>

-

-

Opcjonalnie zamiast możemy w tym miejscu użyć <c01>.

<c>

Element <c> jest obwolutą zawierającą dane o części zasobu (elementu podrzędnego). Jak opisujemy element podrzędny? Mamy tu piękny przykład rekurencji - wewnątrz elementu <c> możemy zastosować wszystkie elementy, których użyliśmy w poziomie wyższym - wystarczy spojrzeć na <archdesc> aby zorientować się, jakie pola są dostępne. Jako minimum powinniśmy użyć etykiety tytułu zasobu <unittitle>, ale można tu umieścić wszystkie szczegóły tego zasobu. Jedyną różnicą w stosunku <archdesc> do jest to, że elementów następnego poziomu nie grupujemy już w <dcs> a wrzucamy je luzem.

-

- <c level=”series”>
- <did>

- <unittitle>...</unittitle>
- <unitid>...</unitid>
- <unitdate>...</unitdate>
- <langmaterial>...</langmaterial>
- ...

- </did>
- <bioghist>...</bioghist>
- <index>...</index>
- <controlaccess>...</controlaccess>
- <dao>...</dao>
- …
- <c level=”file”> …. </c>
- <c level=”file”> …. </c>
- <c level=”file”> …. </c>
- …

- </c>

Ponieważ wiele informacji o zasobie umieściliśmy już w elemencie wyższego poziomu, nie musimy ich tu powtarzać, ograniczając się do informacji specyficznej dla tego elementu.

Jak używać EAD?

Istnieją dwie szkoły używania EAD do opisów archiwalnych. Jednym sposobem jest praca bezpośrednio nad kodem zapisanym w XML. Wbrew pozorom nie jest to bardzo skomplikowane, gdyż jest wiele dobrych edytorów XML i można w nich łatwo dodawać, kopiować i modyfikować etykiety i atrybuty. Metoda ta daje największa elastyczność i możliwość dowolnego uformowania dokumentu EAD. Jest ona popularna szczególnie w instytucjach które zaczęły używać EAD zanim powstały dobre narzędzia. Drugą metodą jest użycie narzędzi bazodanowych. W tej chwili narzędzi takich istnieje już kilka i można już z dużą wygodą zapisywać dane archiwalne w programie dostosowanym do tego celu. Programy te są ciągle w rozbudowie, szerzej o nich w blogu “ Oprogramowanie dla archiwistów ”.

Standard EAD jest bardzo popularny i wiele liczących się archiwów stosuje go do zapisu danych. W Internecie można znaleźć wiele publikacji i poradników do których można i należy sięgnąć aby dowiedzieć się więcej o samym standardzie jak i o zasadach jego używania. Istnieje też polskie tłumaczenie standardu EAD, dostępne online.

Czytaj więcej

- Oficjalny standard EAD
- Polskie tłumaczenie standardu EAD - książka do pobrania w formacie pdf
- EADiva, biblioteka etykiet i atrybutów EAD
- EAD Best Practices at the Library of Congress

Marek Zieliński

Artykuł ukazał się 28 września 2013 w *Blogu archiwistów i bibliotekarzy Instytutu Piłsudskiego*

Może Cię też zainteresować

- Wstęp do standardów metadanych
- Standardy metadanych: Dublin Core
- Iza, Zosia i Atom odwiedzają archiwa