

Service interministériel des Archives de France Bulletin sur les ressources archivistiques numériques, n° 46, juin 2013

Sommaire (cliquer sur les titres)

Version Alpha du schéma EAD, appel à commentaires (mars-mai 2013).....	1
1. Commentaires généraux.....	2
1.1. Définitions d'éléments et d'attributs.....	2
1.2. Nommage des éléments et des attributs.....	2
1.3. Modèle des relations.....	2
1.4. Multilinguisme.....	2
2. Remarques sur des éléments spécifiques.....	3
2.1. Contrôle de la description <control>.....	3
2.2. Description archivistique – Éléments hiérarchiques.....	3
2.3. Description archivistique – Intitulé.....	4
2.4. Description archivistique - Encodage des dates.....	5
2.5. Description archivistique – Points d'accès contrôlés.....	7
2.6. Description archivistique – Autres éléments et attributs.....	8
Cahier des charges type pour l'encodage des instruments de recherche.....	9

Version Alpha du schéma EAD, appel à commentaires (mars-mai 2013)

Le 1^{er} mars dernier, le sous-comité technique pour la Description archivistique encodée (TS-EAD) et l'équipe de développement (SDT) de la Société des archivistes américains (SAA) publiaient officiellement la version Alpha du schéma de l'EAD (<https://github.com/SAA-SDT/EAD-Revision>) et lançaient un appel à commentaires de deux mois auprès de la communauté professionnelle internationale (<http://www2.archivists.org/groups/technical-subcommittee-on-encoded-archival-description-ead/ead-revision-comments>).

Afin de tester cette version Alpha, les groupes d'experts « EAD » et « Données d'autorité » de l'Afnor ont organisé trois ateliers auxquels ont participé 34 professionnels des archives et des bibliothèques utilisateurs de l'EAD mais aussi des éditeurs de logiciels.

Une première synthèse des commentaires ainsi recueillis a été établie à l'issue de ces ateliers puis transmise à l'équipe de développement du Portail Archives Europe (APE ; <http://www.archivesportaleurope.net/fr/home>), en particulier aux responsables du groupe de travail chargé de la mise en œuvre des normes et de l'élaboration de bonnes pratiques. Pour le Portail Archives Europe, un profil EAD commun a été défini, utilisé à présent pour la présentation et l'interrogation des descriptions archivistiques provenant des services d'archives de toute l'Europe (actuellement, 15 pays fournissent des descriptions archivistiques pour le portail, mais le nombre de contributeurs devrait être le double à partir de 2014). Dans le cadre de ce projet, l'EAD est utilisée comme format d'échange entre les institutions participantes et le portail (les exports XML ou XML/EAD de bases de données locales ou encore les profils locaux XML/EAD sont convertis dans le format apeEAD défini et utilisé de manière centralisée), comme base de l'affichage HTML et des recherches de fonctionnalités, et comme format d'import pour une conversion ultérieure vers d'autres formats de données pour l'échange avec le portail inter-domaines Europeana sur les objets patrimoniaux culturels.

Les suggestions et recommandations des participants aux trois ateliers rejoignant celles de l'équipe du portail APE, une réponse commune a été préparée et adressée à la SAA. Ces commentaires sont en cours d'examen par la SAA, en vue de la publication, début août, d'une nouvelle version beta du schéma et d'une première ébauche de la Tag library. On en trouvera ci-après la teneur, mais **il est encore trop tôt pour savoir dans quelle mesure ils seront pris en compte par le TS-EAD et la SDT**. Dans tous les cas, le schéma de l'EAD dans sa version définitive devrait être rendu disponible à la fin de l'année 2013.

1. Commentaires généraux

1.1. Définitions d'éléments et d'attributs

=> [Sommaire](#)

- **Le schéma devrait être rendu suffisamment explicite en lui-même.**

Les noms utilisés pour les définitions comprennent des préfixes qui doivent être explicités. Ces noms viennent des anciennes déclarations d'entités de la DTD (e. : éléments ; m. : macro ; a. : attributs ; am. : attributs modifiables ; data. : modèle de données) mais ce n'est pas documenté et cela oblige à relire les commentaires de la DTD.

Par ailleurs, le TS-EAD a décidé depuis peu de ne pas déprécier l'élément Description des sous-composants <dsc> pour éviter des difficultés de migration. La SAA annoncera son intention de déprécier cet élément dans la prochaine version de l'EAD, laissant ainsi aux institutions un délai supplémentaire de cinq ans pour abandonner progressivement l'utilisation de cet élément (dans la Tag Library ?). Toutefois, le schéma devrait être explicite en lui-même et il faudrait que l'on puisse se reposer le moins possible sur une documentation externe.

- **La modularité du schéma est à revoir.**

La modularité du schéma est en fait celle de la DTD EAD 2002. Quelques définitions sont regroupées dans des divisions (Relations, Dates et Langues) mais cette structuration en divisions n'a pas été poursuivie plus avant. 58 éléments ont des définitions « enfouies » à l'intérieur de définitions d'éléments et sont donc non réutilisables de l'extérieur. Par exemple, il n'existe pas de définition isolée pour <archDesc> qui est défini dans le bloc <ead>.

1.2. Nommage des éléments et des attributs

=> [Sommaire](#)

- **Le nommage des éléments et des attributs doit être revu.**

Le nommage des éléments et des attributs n'est pas encore cohérent dans cette version Alpha. Certes, l'emploi des majuscules a été introduit, mais les noms des éléments continuent à comporter des acronymes, des abréviations, des mots tronqués. On a tantôt des noms complets (<unitDateStructured>) tantôt des abréviations (<archDesc>) et tous les noms d'éléments et d'attributs ne sont pas explicites. Par exemple, le nom de l'élément Description et identification <did> reste peu clair. Cela fait penser à l'élément <div> du HTML. On a besoin de se reporter au dictionnaire des balises pour savoir ce que <did> veut dire.

1.3. Modèle des relations

=> [Sommaire](#)

- **Le modèle des relations doit être entièrement revu (c'est un élément clé de la version Beta).**

Le modèle actuel des relations n'est pas abouti et hybride, avec d'un côté le maintien de <origination>, <repository>, <separatedMaterial>, <relatedMaterial> et des anciens éléments de <controlAccess>, et de l'autre la création de nouveaux éléments au sein de <relations>. Si on les maintient tels quels, il faudrait définir clairement et expliciter au moins dans la Tag Library quels éléments et attributs utiliser pour établir quel type de lien, de référence ou de relation. Il faudrait aussi définir quelles sont les différences entre les liens, les références et les relations.

1.4. Multilinguisme

=> [Sommaire](#)

- **Il faudrait permettre l'encodage d'informations multilingues au sein d'une même instance.**

Par rapport au contexte international du portail Archives Europe, l'intégration des attributs @xml:lang dans le schéma Alpha est vu comme un grand avantage. Toutefois, afin de pouvoir encoder des informations multilingues au sein de la même instance EAD, il serait souhaitable que les éléments pouvant contenir du texte soient rendus répétables. Cela semble être déjà le cas mais ce serait bien de faire une double vérification. De la même manière, il faudrait s'assurer que tous les éléments qui peuvent contenir du texte permettent l'utilisation de l'attribut @xml:lang ; Ce n'est par exemple pas le cas actuellement de l'élément <titleProper>.

2. Remarques sur des éléments spécifiques

2.1. Contrôle de la description <control>

=> [Sommaire](#)

- **Les données codées en anglais devraient être portées par des attributs et non pas par des éléments.**

Les valeurs de certains éléments (<maintenanceStatus>, <agentType>, etc.) sont en anglais et ne sont donc pas affichables en l'état pour des institutions françaises. De plus, l'élément <maintenanceStatus> est le seul élément qu'il n'est pas possible de qualifier avec un attribut @audience de valeur "internal".

- **L'attribut @localType de <addressLine> doit être rendu facultatif.**

L'attribut @localType de <addressLine> est obligatoire mais son contenu n'est pas normalisé et pas du tout contrôlé. Il faudrait rendre l'attribut @localType facultatif (comme tout attribut @localType à chaque fois qu'il apparaît).

- **Transférer les attributs des éléments <ead>, <eadheader> et <eadid> pour la migration de l'EAD 2002 vers la nouvelle EAD.**

Où les attributs de l'élément <ead> se retrouveront-ils dans cette nouvelle version ? Ce n'est pas évident. En particulier, la possibilité d'inclure l'attribut @schemaLocation d'un profil de l'EAD utilisé localement est essentiel, lorsque ce profil est utilisé aussi à des fins de validation, de conversion et d'échange de données comme par exemple dans le portail ArchivesEurope. Alors que les valeurs des attributs @ identifier et @mainagencycode de <eadid> pourraient être transférées par exemple dans <control><otherRecordId> et <control><maintenanceAgency> <agencyCode>, ce qui va advenir avec l'attribut @countrycode n'est pas clair.

En outre, il ne semble pas y avoir dans la nouvelle version de l'EAD d'élément ou d'attribut spécifique comprenant la valeur de l'attribut @url de l'élément <eadid> dans l'EAD 2002. Cet attribut est essentiel pour l'échange de données basées sur l'EAD, incluant de ce fait un lien vers les descriptions d'origine. Surtout quand il s'agit d'échanger des données avec l'objectif de créer des données ouvertes liées, où les URI sont des éléments clés. Il devrait être possible d'avoir cela à un endroit unique comme c'est le cas dans l'EAD 2002, pas seulement comme l'un des nombreux éléments <relationType="resourceRelation"> possibles.

2.2. Description archivistique – Éléments hiérarchiques

=> [Sommaire](#)

- **Les composants numérotés sont à supprimer.**

On aurait pu profiter de la révision de l'EAD pour ne conserver que des composants simples non numérotés. Le maintien des composants numérotés (supprimés par ailleurs du profil européen de l'EAD) va à l'encontre du principe de simplicité défini par le TS-EAD.

2.3. Description archivistique – Intitulé

=> [Sommaire](#)

- Il faudrait réintroduire `<lb/>`, `<emph>` et `<ref>` et les éléments d'indexation (avec l'attribut `@normal`) dans `<unitTitle>` et supprimer l'attribut `@normal` de `<unitTitle>`.

Le modèle de contenu de `<unitTitle>` est fortement appauvri par rapport à la DTD de 2002. Dans l'EAD 2002, `<unitTitle>` pouvait contenir du #PCDATA, ainsi que toute une série d'éléments (`abbr`, `archref`, `bibref`, `bibseries`, `corpname`, `date`, `edition`, `emph`, `expan`, `extptr`, `extref`, `famname`, `function`, `genreform`, `geogname`, `imprint`, `lb`, `linkgrp`, `name`, `num`, `occupation`, `persname`, `ptr`, `ref`, `subject`, `title`, `unitdate`). Dans la version Alpha du schéma, cet élément ne peut contenir que des informations textuelles et les éléments `<emph>`, `<lb/>`, `<ref>` et `<title>`.

La logique de suppression des contenus mixtes n'a prévalu que pour `<unitTitle>`. Dans le même temps, l'élément `<p>` de `<scopeContent>` ou de `<biogHist>` et en fait de tout élément où il apparaît (`relatedMaterial`, `separatedMaterial`, etc.) garde un contenu riche.

On ne peut plus faire de l'indexation au fil du texte, d'où des risques de perte d'information, de décontextualisation de l'indexation. Le schéma semble privilégier l'utilisation de `<controlaccess>`, différenciant ainsi le contenu et l'indexation. Alors qu'en France, les recommandations et guides de bonnes pratiques incitent à utiliser l'indexation au fil du texte, le schéma revient sur cette souplesse d'encodage. Certaines institutions (Archives nationales d'outre-mer) ou certains projets collaboratifs (portail Calames de l'Agence bibliographique de l'enseignement supérieur) ont construit leur système d'information sur l'indexation au fil du texte.

Pourquoi avoir rendu disponible l'attribut `@normal` pour qualifier `<unitTitle>` ce qui revient à confondre titre et intitulé ? Il faudrait plutôt rétablir `<title>` dans `<unitTitle>` et supprimer `@normal` de `<unitTitle>`.

Exemple (instance conforme à la DTD EAD 2002) :

```
<c id="ark--61561-ka455yswp" level="item">
  <did>
    <unitid type="cote d'archives">FR ANOM COL A 28 F° 45v</unitid>
    <unitid type="cote de communication">zone générique = FM, zone cote =
    A//28</unitid>
    <unittitle>Enregistrement au parlement de <geogname normal="Paris
    (France)">Paris</geogname> des pouvoirs respectifs des ducs de
    <persname normal="Bourbon-Condé, Louis Henri de"
    authfilenumber="22837">Bourbon</persname> et d'<persname
    normal="Bourbon, Philippe de"
    authfilenumber="22835">Orléans</persname> durant la minorité de
    <persname normal="Louis XV" authfilenumber="24097">Louis
    XV</persname>, ainsi que de la constitution du conseil de régence et
    des six autres conseils particuliers (n°71).</unittitle>
    <unitdate normal="1715-09-12">12 septembre 1715</unitdate>
  </did>
</c>
```

Exemple (les mêmes données converties en schéma Alpha) :

```
<c id="ark--61561-ka455yswp" level="item">
  <did>
    <unitId localType="cote d'archives">FR ANOM COL A 28 F° 45v</unitId>
    <unitId localType="cote de communication">zone générique = FM, zone
    cote =A//28</unitId>
    <unitTitle>Enregistrement au parlement de Paris des pouvoirs
    respectifs des ducs de Bourbon et d'Orléans durant la minorité de
```

```

Louis XV, ainsi que de la constitution du conseil de régence et des
six autres conseils particuliers (n°71).</unitTitle>
<unitDateStructured>
<dateSingle standardDate="1715-09-12">12 septembre 1715</dateSingle>
</unitDateStructured>
</did>
<controlAccess>
<geogNameEntry><part>Paris (France)</part></geogNameEntry>
<persNameEntry identifier="22837">
<part>Bourbon-Condé, Louis Henri de</part>
</persNameEntry>
<persNameEntry identifier="22835">
<part>Bourbon, Philippe de</part>
</persNameEntry>
<persNameEntry identifier="24097">
<part>Louis XV</part>
</persNameEntry>
</controlAccess>
</c>

```

2.4. Description archivistique - Encodage des dates

=> [Sommaire](#)

- **Le modèle de contenu des dates est à revoir et à homogénéiser.**

Le modèle de contenu des dates n'est pas homogène. Pourquoi ne pas avoir appliqué le modèle de l'EAC-CPF qui semble satisfaisant ?

- 1) Avec l'élément <date> de <p>, on est resté sur l'ancien modèle. Le modèle de contenu de cet élément doit donc être harmonisé avec celui des éléments <unitDate> et <unitDateStructured>.
- 2) Le schéma Alpha dispose apparemment d'un élément pour contenir du texte (<unitDate>) et d'un autre élément pour normaliser les dates (<unitDateStructured>). Or, c'est source de redondance et de confusion pour les utilisateurs puisque les éléments <unitDate> et <unitDateStructured> peuvent coexister dans une même instance EAD sans problème et que <unitDateStructured> peut même être laissé vide. Par ailleurs, l'élément <unitDateStructured> contient un élément <dateSingle> où là aussi on peut mettre directement du texte (comme par exemple « deuxième moitié du XIXe siècle ») et qui peut donc être utilisé de la même manière que l'élément <unitDate>. Quelle serait la différence entre :

```
<unitDate era="..." calendar="...">Deuxième moitié du XIXe
siècle</unitDate>
```

et

```
<unitDateStructured era="..." calendar="..."><dateSingle>Deuxième
moitié du XIXe siècle</dateSingle><unitDateStructured> ?
```

Quand utiliser l'un et l'autre de ces éléments ? Quelles pourraient être les recommandations à ce sujet ?

- 3) L'élément <unitDate>, qui sert à assurer la récupération des anciennes données, devrait de ce fait conserver l'ancien modèle de contenu pour permettre la récupération intégrale de ces anciennes données. Or, les attributs de <unitDate> sont les mêmes que dans la DTD EAD 2002 à l'exception de l'attribut @normal qui a été supprimé. Comment récupérer les dates normalisées au moyen de l'attribut @normal ?

Exemples d'encodage avec <unitdate> et l'attribut @normal dans l'EAD 2002 :

1.

```
<unitdate calendar="gregorian" era="ce"  
normal="1945-01-01/1949-12-31">1945 - 1949</unitdate>
```

2.

```
<unitdate calendar="gregorian" era="ce"  
normal="1945-01-01/1949-12-31">1945, 1949</unitdate>
```

3.

```
<unitdate calendar="gregorian" era="ce"  
normal="1945-01-01/1949-12-31">1945, 1948 - 1949</unitdate>
```

4.

```
<unitdate calendar="gregorian" era="ce"  
normal="1945-01-01/1949-12-31">1945 - 1946, 1949</unitdate>
```

Ces données seraient-elles ainsi converties dans le nouveau schéma EAD ?

1.a

```
<unitDateStructured calendar="gregorian" era="ce">  
<dateRange>  
<fromDate standardDate="1945-01-01">1945</fromDate>  
<toDate standardDate="1949-12-31">1949</unitdate>  
</dateRange>  
</unitDateStructured>
```

ou plutôt

1.b

```
<unitDateStructured calendar="gregorian" era="ce">  
<dateRange>  
<fromDate standardDate="1945">1945</fromDate>  
<toDate standardDate="1949">1949</unitdate>  
</dateRange>  
</unitDateStructured>
```

2.a

```
<unitDateStructured calendar="gregorian" era="ce">  
<dateSet>  
<dateSingle standardDate="1945-01-01">1945</dateSingle>  
<dateSingle standardDate="1949-12-31">1949</dateSingle>  
</dateSet>  
</unitDateStructured>
```

ou plutôt

2.b

```
<unitDateStructured calendar="gregorian" era="ce">  
<dateSet>  
<dateSingle standardDate="1945">1945</dateSingle>  
<dateSingle standardDate="1949">1949</dateSingle>  
</dateSet>  
</unitDateStructured>
```

3.a

```
<unitDateStructured calendar="gregorian" era="ce">  
<dateSet>  
<dateSingle standardDate="1945-01-01">1945</dateSingle>  
<dateRange>  
<fromDate standardDate="1948-01-01">1948</fromDate>
```

```
<toDate standardDate="1949-12-31">1949</unitdate>
</dateRange>
</dateSet>
</unitDateStructured>
```

ou plutôt

3.b

```
<unitDateStructured calendar="gregorian" era="ce">
<dateSet>
<dateSingle standardDate="1945">1945</dateSingle>
<dateRange>
<fromDate standardDate="1948">1948</fromDate>
<toDate standardDate="1949">1949</unitdate>
</dateRange>
</dateSet>
</unitDateStructured>
```

4.a

```
<unitDateStructured calendar="gregorian" era="ce">
<dateSet>
<dateRange>
<fromDate standardDate="1945-01-01">1945</fromDate>
<toDate standardDate="1946-12-31">1946</unitdate>
</dateRange>
<dateSingle standardDate="1949-12-31">1949</dateSingle>
</dateSet>
</unitDateStructured>
```

ou plutôt

4.b

```
<unitDateStructured calendar="gregorian" era="ce">
<dateSet>
<dateRange>
<fromDate standardDate="1945">1945</fromDate>
<toDate standardDate="1946">1946</unitdate>
</dateRange>
<dateSingle standardDate="1949">1949</dateSingle>
</dateSet>
</unitDateStructured>
```

- 4) Alors qu'on ne peut rentrer que du texte dans les éléments <unitDate> et <dateSingle>, seul <unitDate> peut contenir des éléments de mise en forme (<emph>, <lb/>) et <ref> et <title>. L'élément <dateSingle> devrait aussi contenir <emph> (au moins) de sorte que l'on puisse par exemple typer le « e » exposant dans dans « XIX^e siècle », le « bre » exposant de « dec^{bre} », etc.
- 5) L'utilisation des éléments <dateSingle>, <dateRange> et <dateSet> au sein de <unitDateStructured> doit être précisée afin que <dateSet> ne soit pas redondant au sein de <unitDateStructured>. L'élément <unitDateStructured> devrait ne pouvoir contenir qu'un <dateSingle>, qu'un <dateRange> ou qu'un <dateSet>. En n'utilisant que <dateSet>, il serait possible de combiner des dates simples et des intervalles de dates ou plusieurs intervalles de dates ou plusieurs dates simples.

2.5. Description archivistique– Points d'accès contrôlés

=> [Sommaire](#)

- **Le nommage et le modèle de contenu des éléments doivent être rendus cohérents.**

Actuellement, <corpNameEntry> est inclus dans <controlAccess> alors que <corpName> est contenu dans <origination>, avec des modèles de contenu différents. L'élément <part> n'est obligatoire que dans <corpNameEntry>. Les éléments <corpName> et <corpNameEntry> ont la même finalité. Pourquoi des noms et des modèles de contenu différents ? Cette complexité est inutile.

Même remarque pour :

- <corpName> au sein de <repository>,
- <persname>, <famname> et <name> au sein de <origination> d'une part, <persNameEntry>, <famNameEntry> et <genericNameEntry> dans <controlAccess>, d'autre part,
- <corpName>, <persName>, <famName>, etc. dans l'élément <indexEntry> et les éléments <corpNameEntry>, <persNameEntry>, <famNameEntry> dans <controlAccess>.

- **Le modèle de contenu des éléments d'indexation doit être révisé afin de pointer vers une autre ressource**

Le modèle actuel soulève plusieurs questions et problèmes :

- Le nom de l'attribut @identifier des éléments *Entry de <controlAccess> * est inapproprié (ce n'est pas l'identifiant de l'élément) et il n'est pas suffisant si l'on veut créer des liens vers plusieurs notices d'autorité. Un élément spécifique répétable serait préférable, ou bien un modèle de contenu avec une liste de anyURI doit être défini pour l'attribut @identifier
- Pourquoi des éléments comme <list> et <table> ont-ils été maintenus dans <controlAccess> ?

- **Une catégorie générique <place> doit être conservée pour les éléments d'indexation ou les éléments de <controlAccess>.**

Actuellement, le modèle de représentation des lieux n'est pas cohérent. On trouve :

- <placeEntry> au sein de <chronlist>.
- <geogNameEntry> au sein de <controlAccess> (avec un élément <part> comme composante – même modèle que les éléments de <controlAccess>).
- <geogName> comme élément d'indexation dans <scopeContent> et dans <biogHist> (comprenant <emph> et <lb> - même modèle que les éléments d'indexation des éléments <scopeContent>, sauf <title> et <archref>).

Quel est le rôle de chacun de ces éléments dans le schéma? Que signifie l'ajout de « Entry » très exactement ?

En outre, concernant les attributs de coordonnées géographiques (attributs @longitude, @latitude, @altitude de <placeEntry>), il est dommage de permettre aux utilisateurs de rentrer des données sur la longitude, la latitude et l'altitude et de les laisser mettre n'importe quoi. Étant donné que ces attributs sont séparés, des types doivent leur être assignés Les vocabulaires contrôlés utilisés doivent également être précisés.

2.6. Description archivistique – Autres éléments et attributs

=> [Sommaire](#)

- **L'élément <fileplan> doit être conservé.**

L'élément <fileplan> a été supprimé alors qu'il est actuellement utilisé dans apeEAD. Quelles sont les recommandations pour transférer le contenu des éléments <fileplan> existants lors des futures migrations de l'EAD 2002 vers le nouveau schéma EAD ?

- **L'attribut @xPointer doit être défini.**

Dans les attributs de lien, l'attribut @xPointer n'a pas de définition.

Cahier des charges type pour l'encodage des instruments de recherche

=> [Sommaire](#)

Les Archives de France viennent de publier sur leur site un cahier des charges type pour la numérisation et l'encodage des instruments de recherche (<http://www.archivesdefrance.culture.gouv.fr/gerer/classement/normes-outils/ead/>). Ce document a pour but de proposer aux services souhaitant s'engager dans la conversion rétrospective au format XML/EAD d'instruments de recherche, une assistance à la rédaction d'un cahier des charges pour l'encodage. Il s'agit d'un cahier des charges type. Seuls les aspects métier sont abordés ici.

Ce modèle de cahier des charges est diffusé en complément de l'appel à projets lancé par les Archives de France pour l'encodage des instruments de recherche, sur le modèle de celui du ministère de la Culture et de la Communication pour la numérisation.