Divers

- Insertion dans le teiHeader des noms retrouvés par Camille
- Insertion des mentions des responsabilités
    - **Quand on a "Camille", c'est Camille B, pas Camille V ? dans le doute j'ai laissé tel quel**
- Indentation des fichiers
- Je passe tous les identifiants de fichiers en minuscules (en prévision de l'upload sur Drupal, qui convertit les noms en fichiers en minuscules)
- Uniformisation des espaces autour de gap (avant et après), pb, lb (avant)
- **Il me semble qu'on a parfois des unclear à la place de gap : `<unclear>texte illisible</unclear>`, `<unclear>...</unclear>`, etc. : la requête XPath `//unclear` renvoie toutes les occurrences, après on peut rechercher remplacer par `<gap/>`**
- **Parfois, une note indique un ajout ou une suppression, mais on n'a pas le `<add>` correspondant. Exemple : t10010073-1967-01-oj.xml `<item><term ref="#carreslatins">Carrés latins</term> en littérature (<persName ref="#cb">Cl. Berge</persName>) <note n="12" resp="editor" xml:id="ftn12">Ajouté à la main</note></item>`**
- **Erreur sur le fac-similé dans t10010170-1975-10-cv.xml (je n'ai pas trouvé la bonne page dans le dossier)**

Notes

- Numérotation des notes en vue de l'affichage

Pour répondre à certaines questions d'Hélènes :

- J'enlève "[Note de l'éditeur]", redondant par rapport à l'attribut @resp="editor" : cette mention sera rajoutée automatiquement.
- T10010196-1977-05-CV.xml : `<p><note resp="editor">Page 7 du Pdf-coupon réponse, ajout manuscrit de la main de <persName ref="#GP">Perec</persName></note> : <note place="foot" resp="author" xml:id="ftn1" n="1">« Connais-tu cela ? I vitelli dei romani sono belli. C’est un certain <persName>Delahaye</persName> (<title>la Frontière et le texte</title>), qui le cite, d’après <persName>Umberto Eco</persName>. »</note>. </p>` On dira plutôt : `<note n="1" resp="editor" xml:id="ftn1">Page 7 du Pdf-coupon réponse, ajout manuscrit de la main de <persName ref="#gp">Perec</persName> : « Connais-tu cela ? I vitelli dei romani sono belli. C’est un certain <persName ref="#delahaye">Delahaye</persName> (<title ref="#lafrontiereetletete">la Frontière et le texte</title>), qui le cite, d’après <persName ref="#umbertoeco">Umberto`

`Eco</persName>. »</note>` (on ne peut pas représenter une note dans la note)

- T10010147-1973-12-CV.xml : on dirait plutôt `<note resp="editor">Ajout manuscrit de la main du secrétaire</note>` (le add concerne le texte édité, pas la note éditoriale)

Ajout des @ref sur les entités

- Pour les Oulipiens du fichiers oulipiens.xml : si le persName correspond à l'une des formes listés dans oulipiens.xml, l'identifiant correspond aux initiales de l'Oulipien
- Pour les autres formes : identifiant = forme en minuscules, sans accent, sans caractères spéciaux, sans ponctuation
    - On pourrait aussi ne prendre en compte que le dernier mot de la forme pour regrouper par exemple "Vallès" et "Jules Vallès" : ça évitera des erreurs, et ça en créera d'autres (par exemple "Mme Berge" sera assimilé à CB)
    - **Processus de correction :**
        - **Étape 1 : formes différentes à regrouper sous une même entité.** Je prépare un thésaurus des entités avec pour chacune l'identifiant et ses différentes formes. Si plusieurs formes appartiennent à une même entité mais ont des identifiants distincts, on les regroupe dans le thésaurus, et je réinsérerai les bons identifiants. Par exemple, pour l'instant, on a `<title ref="#centmillemilliards">Cent mille milliards</title>` et `<title ref="#n1mmdepoemes">100 M.M. de Poèmes</title>`, à regrouper sous une même entrée (*donc pour l'instant, ne pas corriger les identifiants dans les documents*). Mais il faut au préalable que le balisage soit le plus propre possible (Hélène, tu disais qu'il y avait encore du travail à faire de ce côté...)
        - **Étape 2 : formes similaires à lier à deux entités différentes.** Au besoin, corriger dans les fichiers s'il y a des cas ambigus (une même forme pour plusieurs entités) : par exemple si on a deux `<persName>Jacques</persName>`, dans un cas faisant référence à JD, et dans l'autre à JB. Mais je ne suis pas sûr que ce cas se produise souvent.

supplied, sic/corr, abbr/expan, orig/reg

Je rajoute abbr/expan, orig/reg et supplied au schéma. Pour tout ce qui suit, je ne suis pas sûr de toujours bien appliquer les recommandations de la TEI : à vérifier avec Lou !

supplied

Les crochets sont

- tantôt présents dans le texte d'origine (exemple : t10010234-1980-12-cr.xml : [mon œil de velours est moins côteleux que le ….MB.])

- tantôt des marques éditoriales (selon les consignes de retranscription), pour indiquer un texte ajouté, supprimé, erroné, abrégé, manquant

Pour tirer parti de l'encodage en TEI, j'ai remplacé automatiquement un certain nombre de paires de crochets par des balises `<supplied>` (j'ai visé les expressions sans espaces et sans chiffres pour ne pas prendre les expressions mathématiques et les expressions longs, qui sont souvent de la main de l'auteur) : http://www.tei-c.org/release/doc/tei-p5-doc/fr/html/ref-supplied.html. Supplied "permet d'encoder du texte restitué par l'auteur de la transcription ou par l'éditeur pour une raison quelconque, le plus souvent parce que le texte du document original ne peut être lu, par suite de dommages matériels ou de lacunes" Exemple : t10010332-1989-11-cr.xml : `W. <supplied>travail</supplied> de <term ref="#permutation">permutation</term>`

- **On peut aller plus loin et vérifier les crochets restants : rechercher avec expression régulière `\[.+?\]` ; au besoin, remplacer les [] par des balises add, del, supplied, abbr/expan, sic/corr, orig/reg**
- **Vérifier les occurrences de supplied : Xpath `//supplied`. Remplacer le cas échéant par add, del, abbr/expan, sic/corr, orig/reg**

abbr/expan

http://www.tei-c.org/release/doc/tei-p5-doc/fr/html/ref-abbr.html]. `<abbr>` "contient une abréviation quelconque", et peut fonctionner avec `<expan>` : `<choice> <abbr>abbréviation</abbr><expan>nom complet</expan></choice>`

- Je l'utilise pour les initiales des Oulipiens (à l'origine c'était le programme de transformation en HTML qui affichait une info-bulles pour les noms d'Oulipiens abrégés, mais c'est lourd et peu contrôlable, il me semble que l'information est mieux à sa place dans le fichier source) `<persName ref="#jb" role="expéditeur"> <choice><abbr>JB</abbr><expan>Jacques Bens</expan></choice> </persName>`
- Je l'utilise aussi dans un cas comme
    - t10010157-1973-05-cv.xml (le fichier où l'initiale des noms de famille des Oulipiens est supprimée) : `<persName ref="#ic"><choice> <abbr>ALVINO</abbr><expan>Italo Calvino</expan></choice> </persName>`
    - t10010180-1975-12-cr.xml : `<choice><abbr>Quenel</abbr> <expan>Queneleiev</expan></choice>`
    - t10010159-1974-02-cr.xml : `l'<choice><abbr>Ant</abbr> <expan>Anthologie</expan></choice>`

J'hésite parfois avec supplied, je m'en remets à Lou...

- **J'ai ajouté des balises autour des noms de personnes qui sont visiblement des initiales (pas plus de trois caractères, en majuscules). La plupart sont des noms d'Oulipiens, j'ai indiqué dans un élément expan le nom complet. Mais il reste**

des abbréviations qui ne sont pas explicitées (notamment dans le teiHeader, pour les invités) : indiquer le nom complet dans `<expan></expan>` ? Requête XPath : `//abbr[.=following-sibling::expan]/translate(.,' .-','')` (pour l'instant, dans ces cas, le contenu de expan est le même que celui de abbr). Soit vous corrigez directement les fichiers, soit vous m'envoyez un tableau d'équivalence (sans distinguer les différentes formes d'une même abbréviation : indiquez simplement la forme sans espace, sans tiret et sans point, "YM" et pas "Y.M.")

sic/corr

http://www.tei-c.org/release/doc/tei-p5-doc/fr/html/ref-sic.html "texte reproduit quoiqu'il soit apparemment incorrect ou inexact". Exemples :

- `<choice><sic>1985</sic><corr>1990</corr></choice>`
- `<term ref="#musiquealgorithmique">musique <choice> <sic>algorythmique</sic><corr>algorithmique</corr></choice> </term>`
- `<persName ref="#gp"><choice><sic><unclear>Peredur</unclear></sic> <corr>Perec</corr></choice></persName>`

Question : les `<sic>` pour lesquels aucun `<corr>` n'est proposé sont-ils destinés à recevoir une correction, ou bien ils s'agit de néologismes forgés sciemment par les Oulipiens ? Expression XPath : `//sic[not(parent::choice)]`. Exemple :

- Dans t10010038-1963-12-cr.xml : `<term ref="#isouvocalisme"> <sic>isouvocalisme</sic></term>`
- Dans t10010165-1974-08-cr.xml : `<sic>thémantique</sic>`

orig/reg

orig : "partie notée comme étant fidèle à l'original et non pas normalisée ou corrigée" http://www.tei-c.org/release/doc/tei-p5-doc/fr/html/ref-orig.html reg : "partie qui a été régularisée ou normalisée de façon quelconque" http://www.tei-c.org/release/doc/tei-p5-doc/fr/html/ref-reg.html Il me semble que ça s'applique bien à des cas comme :

- `<choice><orig>Mehico</orig><reg>Mexico</reg></choice>` (les deux orthographes sont attestées, mais on uniformise en Mexico)
- `<choice><orig>dico</orig><reg>dictionnaire</reg></choice>` (à moins qu'on considère ça comme une abbréviation ?)
- L'uniformisation Mobius/Mœbius/Möbius, il me semble que ça relève de orig/reg (dans la mesure où toutes ces variantes sont attestées) ?

Pour "Centre Pompidou", "Pompidou", "Beaubourg", on pourrait utiliser orig/reg (?), mais on peut aussi laisser comme ça, et regrouper ces différentes formes sous la même entité.

EAD

Nettoyage

- J'enlève les espaces et les tirets initiaux et finaux des item
- Je fusionne avec le précédent les item qui ne contiennent que le nombre de feuillets
    - **Mais il resterait des cas à revoir (notamment parce que je ne savais pas toujours si le nombre de feuillets s'appliquait au document juste avant ou juste après) : nettoyer avec la requête Xpath `//item`, classer par ordre alphabétique et regarder rapidement les cas qui semblent bizarres**
- Certains item font référence au précédents ("Annexe de *ce* compte-rendu"). Mais à mon avis on peut laisser comme ça : l'EAD et le corpus de fichiers TEI sont deux objets différents, il suffit qu'on puisse accéder à l'un depuis l'autre, mais je n'utiliserai pas les titres de l'EAD hors du contexte de l'EAD
- J'uniformise le contenu des scopecontent : je transforme en list/item les documents d'un dossier listés sous forme de paragraphes
- Pour les séries que nous ne publions pas, je ne sais pas trop ce que ça va donner : le design est fait pour des titres de documents courts, et il y en a de très long, avec parfois des niveaux de hiérarchisation supplémentataire (listes imbriquées ou plusieurs scopecontent quand un "dossier" contient plusieurs "chemises" contenant elles-mêmes plusieurs "documents")

Ajout des @id

- @corresp n'est pas valide en EAD, j'utilise @id plutôt que de me lancer dans l'ajout d'un nouvel espace de nom.
- Pour chaque document XML, je cherche l'item qui appartient au même dossier (numéro de scan) et qui comporte l'expression "Compte-rendu" (si le document est un CR), "Convocation" (si c'est un CV), "Ordre du jour" (si c'est un OJ). S'il y a plusieurs candidats, le @id est ajouté au premier.
    - **Éventuellement, vérifier que l'automate a relié tous les documents au bon item (mais c'est beaucoup de travail)**
- Les 70 documents qui n'ont pas trouvé d'item correspondant sont listés dans le tableau ead_files_without_record.xlsx
    - **Ajouter dans ead.xml les 70 documents à l'aide ead_files_without_record.xlsx (rechercher dans l'EAD le bon dossier à l'aide du numéro de scan (colonne gauche), puis il suffit de copier le contenu de la cellule de droite en tant qu'attribut du bon item)**
- Je peux automatiser plus en décidant qu'un item qui contient les mots "lettre" ou "circulaire" est considéré comme une convocation, mais ça laisse toujours une vingtaine de cas irrésolus, et ça risque de produire beaucoup d'erreurs.