



HAL
open science

ekdosis pour l'impatient

Estelle Debouy

► **To cite this version:**

| Estelle Debouy. ekdosis pour l'impatient. 3ème cycle. France. 2023, pp.11. hal-04059411

HAL Id: hal-04059411

<https://hal.science/hal-04059411v1>

Submitted on 5 Apr 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ekdosis pour l' impatient

<http://ekdosis.org>

Sommaire

1	Pour des éditions critiques numériques	2
2	Les bases du langage \LaTeX	3
3	Les préalables à la saisie de l'édition	6
4	Les principes d'encodage avec ekdosis	8

1 Pour des éditions critiques numériques

De nombreuses éditions numériques peuvent aujourd’hui être consultées en ligne : citons, pour prendre un exemple dans le domaine de l’Antiquité, le Perseus Project (<http://www.perseus.tufts.edu/>). Mais rares en revanche sont les éditions *critiques* numériques car, dans la plupart des cas, le texte est présenté sans préface où est étudiée l’histoire du texte, sans bibliographie et surtout sans appareil critique.

Les quelques projets d’éditions critiques numériques¹ reposent sur l’utilisation de l’encodage TEI `xml` qui permet de transformer le texte et son appareil en une base de données complète consultable par le lecteur en fonction de ses besoins. La TEI fournit en effet des éléments normalisés très précis pour la description des sources manuscrites et pour l’encodage de l’apparat critique. Dans ces projets, l’éditeur encode le texte en TEI puis le code `xml` est transformé en source \LaTeX pour produire un fichier au format PDF. Mais l’apparat critique généré à partir de ces données brutes se présentera comme un appareil de *données* et non comme un appareil *rédigé* à moins d’appliquer, lors de la transformation, des algorithmes spécifiques qui restent à écrire, à supposer que les éditeurs soient disposés à les utiliser. Or, il est important que l’éditeur puisse non seulement saisir les variantes du texte mais encore, s’il le souhaite, présenter au lecteur les arguments qui lui ont permis de prendre telle ou telle décision.

C’est ce que permet le logiciel *ekdosis* développé par Robert Alessi (CNRS UMR 8167)² qui a fait le choix d’une approche différente par rapport aux projets mentionnés plus haut : l’éditeur saisit tous les éléments constitutifs de l’édition dans un fichier source \LaTeX et demande ensuite une sortie aussi bien vers un fichier PDF que vers un fichier TEI `xml` qui pourra être l’objet de transformations. Cette démarche repose sur le constat que \LaTeX , contrairement à `xml`, est un langage naturellement conçu pour être utilisé dans la composition de documents complexes, tels que les éditions critiques où les notes de bas de page et autres types d’annotations peuvent être particulièrement abondantes. Voici ce qu’*ekdosis* permet de faire :

- Rédiger tout un appareil critique complexe : les textes édités peuvent être présentés avec plusieurs étages d’apparats, un *apparatus criticus* (pour les variantes), un *apparatus fontium* (pour les références au texte édité), un *apparatus testium* (pour

1. Cf. la *Digital Latin Library* (<https://digitallatin.org/>) ou encore le pôle Document numérique de l’université de Caen (https://www.unicaen.fr/recherche/mrsh/document_numerique/outils/apparat).

2. Il est disponible depuis le CTAN : <http://ctan.org/pkg/ekdosis>. Une documentation très détaillée a été rédigée par l’auteur et est disponible sur le site du CTAN : « *ekdosis – Typesetting TEI xml Compliant Critical Editions* », 2020, <https://mirrors.ctan.org/macros/luatex/latex/ekdosis/ekdosis.pdf>.

- les *testimonia*), ainsi que toutes sortes de notes complémentaires au texte lui-même.
- Produire un fichier TEI `xml` : *ekdosis*, à partir de la seule saisie en \LaTeX , peut générer deux fichiers à la fois, l'un au format PDF et l'autre au format TEI `xml`.
 - Présenter des éditions critiques multilingues avec alignement (ou non) du texte et de la traduction, voire des traductions³.

Il convient donc de présenter d'abord les bases du langage \LaTeX avant d'étudier la façon d'encoder une édition critique avec *ekdosis*.

2 Les bases du langage \LaTeX

Principe Le texte, saisi au kilomètre par l'auteur, est accompagné des instructions de structuration : cet ensemble s'appelle le texte source, ou plus simplement le source. Il est saisi dans un logiciel que l'on appelle éditeur de texte, et que l'on évitera de confondre avec le traitement de texte. Une fois le source composé — texte + commandes de structuration — on le compile à l'aide d'une version de \LaTeX conçue pour produire directement un fichier `.pdf`. \LaTeX retourne alors ce nouveau fichier qui ne peut plus être modifié directement. Par suite, si l'on veut corriger des erreurs telles que des fautes de frappe, on corrige le source, puis on le recompile. Que l'on se rassure : la compilation est une opération simple et rapide. Il est conseillé d'utiliser le moteur de compilation $\text{Lua}\LaTeX$ qui, outre qu'il reconnaît les caractères Unicode, combine deux langages de programmation et augmente considérablement les possibilités de \LaTeX .

Pour débiter, il est possible d'utiliser un éditeur et un compilateur en ligne : Overleaf (<https://www.overleaf.com>). À partir de la page d'accueil, il faut créer un nouveau projet en suivant le lien **Nouveau projet** puis choisir l'option **Projet vide** ou **Blank project**, et lui donner un nom, par exemple « Premiers pas ». Le nouveau projet est alors créé. Observons à présent notre premier projet tel qu'il se présente sur la figure 1 page suivante.

Nous y trouvons :

1. Un menu qui permet d'effectuer les réglages du document courant. Pensez à choisir tout de suite le bon moteur de compilation, $\text{Lua}\LaTeX$ à la place de $\text{pdf}\LaTeX$.
2. Un lien qui permet de quitter le document et de revenir à la page d'accueil d'Overleaf.
3. Une zone réservée à l'édition du document lui-même : c'est l'éditeur de texte intégré à Overleaf.

3. Mais *ekdosis* peut aussi être adapté aux modules de la TEI pour l'encodage de textes spécifiques tels que des transcriptions de discours (c'est la commande `EnvtoTEI` qui sera alors utilisée).

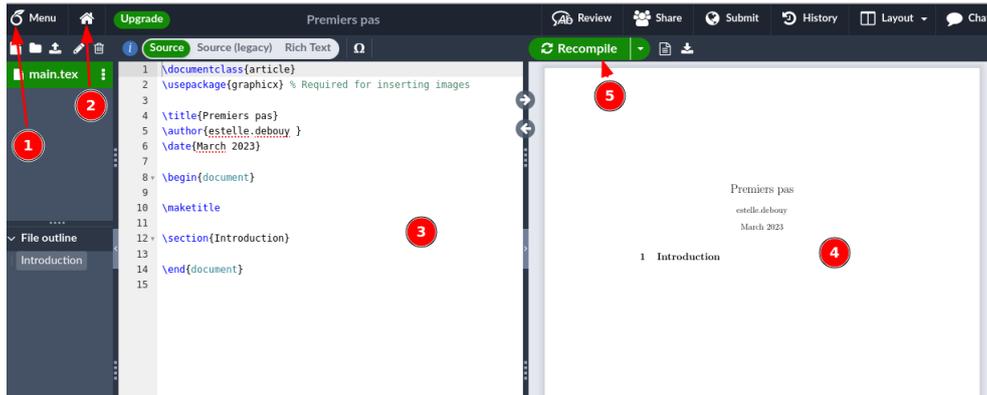


FIGURE 1 – Overleaf

4. Une zone qui permet de visualiser le document au format .pdf.
5. Un bouton qui permet de compiler le document.

Structure d'un document élémentaire Observons le document source suivant :

```

1 \documentclass{article}
2
3 % Préambule
4
5 \begin{document}
6
7 % Corps du document
8
9 Par les temps qui courent, il est plus que recommandé de
10 lire le \emph{Décaméron} de Bocacce.
11
12 \end{document}

```

— Lignes 1–4, le préambule. Tout fichier source \LaTeX commence par un préambule, à savoir : le code compris entre la première ligne qui est toujours un appel à une classe de document (ici `article`, mais on peut aussi choisir `book` par exemple) et l'environnement qui contient le document lui-même. Dans le préambule, on trouve des réglages ou des instructions générales concernant l'ensemble du document, telles que :

— la taille du papier, la taille et le choix des polices de caractères ;

- le choix des langues de travail ;
- les éléments du titre du document (titre, nom de l’auteur, date, etc.)
- les extensions que l’on souhaite charger en mémoire, que l’on appelle packages.
- Lignes 5–12, le corps du document. C’est la zone du document lui-même, c’est-à-dire ce qui est destiné à être imprimé. Le corps du document est toujours encadré par les deux déclarations que l’on trouve aux lignes 5 (`\begin{document}`) et 12 (`\end{document}`).

Observez la ligne 10 et remarquez la forme particulière des commandes \LaTeX à partir du fragment *Décaméron* : elles commencent toutes par le caractère `\`, appelé caractère d’échappement (angl. *escape character*). Toutes les commandes \LaTeX se présentent sous cette forme. Ces commandes sont des instructions que l’auteur donne à \LaTeX au cours de la saisie de son texte ; elles seront interprétées et exécutées durant la compilation. La syntaxe est la suivante : `\commande{argument}`

À savoir :

- (a) Le caractère d’échappement (`\`) qui fait passer \LaTeX en mode commande.
- (b) Le nom d’une commande, c’est-à-dire de l’ordre que \LaTeX doit exécuter.
- (c) Entre deux accolades qui le délimitent, l’argument, c’est-à-dire la zone de texte sur laquelle porte la commande donnée au point (b).

Notez que certaines commandes admettent des options qui seront placées entre crochets droits entre le nom de la commande et l’argument. Ex. : `\usepackage[french]{babel}` dit à \LaTeX de charger l’extension qui prend en charge les langues du monde tout en précisant que la langue principale de travail sera le français.

Dernière remarque : la zone de texte formée par les lignes situées entre les commandes de type `\begin{...}` et `\end{...}` se nomme **environnement**. Dans l’exemple ci-dessus, nous avons donc défini un environnement `document` qui est l’espace dans lequel est saisi le corps du texte à imprimer. Les environnements peuvent être emboîtés les uns dans les autres, comme des poupées russes. Mais il est formellement interdit de fermer un environnement donné s’il contient des environnements qui n’ont pas été eux-mêmes fermés. Il faut enfin observer que dans la commande *Décaméron* la paire d’accolades définit elle aussi un environnement. La différence est que l’on réserve plutôt les environnements définis par des paires d’accolades à quelques mots tandis que ceux qui sont définis par `\begin{...}` ... `\end{...}` portent plus volontiers sur des paragraphes continus.

Quelques commandes \LaTeX Vous trouverez dans le tableau [1 page suivante](#) les principales commandes pour commencer.

Commandes	Sens	Commentaire
<code>\usepackage</code>	Sert à charger les extensions appelées <i>packages</i> .	N'est autorisée que dans le préambule.
<code>\usepackage{fontspec}</code>	Change la police du document.	Faire suivre de la commande <code>\setmainfont{police}</code> .
<code>\usepackage{csquotes}</code>	Met en forme les citations.	<code>\blockquote{}</code> introduit la citation.
<code>\maketitle</code>	Produit le titre en représentant les données de <code>\title</code> et <code>\author</code> .	À insérer après <code>\begin{document}</code> .
<code>\section{}</code>	Pour les titres des parties.	Dans l'ordre on a : <code>\chapter</code> , <code>\section</code> et <code>\subsection</code> .
<code>\tableofcontents</code>	Produit la table des matières.	
<code>\emph{}</code>	Met en valeur le contenu de l'argument.	Produit de l'italique (sauf dans un contexte en italique).
<code>\footnote{}</code>	Introduit une note de bas de page.	
<code>\includegraphics{}</code>	Insère une image.	Le package <i>graphicx</i> doit être chargé dans le préambule.

TABLE 1 – Principales commandes

3 Les préalables à la saisie de l'édition

Prenons comme exemple cet extrait de l'édition imprimée de la *Seconde action contre Verrès* de Cicéron (II, IV, 25, éd. Peterson, Oxford University Press, 1916) :

56 O tempora, o mores! Nihil nimium vetus proferam. Sunt 25
vestrum aliquam multi qui L. Pisonem cognorint, huius
L. Pisonis, qui praetor fuit, patrem. Ei cum esset in Hi-

1 audistis. Atque hac tota de re audistis Archagathi p̄δ 6 nego-
tiatores RSHπ: negotiatoresque δ: del. Halm, Jacoby, Zielinski
15 scaphiis Gubelmius: scaphis RSHπ: scyphis δ includebat πδ:
concludebat RSH 16 tamen codd.: autem Madvig, interim

Eberh. 22 pallio rk, Lg. 42: pallio pullo RS: pallio pollio p
(purpureo, v, § 137) audiverit Zielinski 26 gnořt (= cognorint)
SD: unde gnorunt sive norunt Ψ: gonrit R: cognoverint π 27
Ei Graevius: et RS: is p̄δ

Apparat critique

Lemme donné par les mss S et D.

Variantes

Comment le reproduire avec *ekdosis*? Commençons par charger le package *ekdosis*

en ajoutant dans le préambule du source \LaTeX la ligne suivante qui produira à la fois le PDF et le TEI `xml` :

```
\usepackage[teiexport]{ekdosis}
```

Il convient ensuite de déclarer plusieurs éléments qui apparaîtront dans l'édition.

Les apparats Pour paramétrer deux étages de notes critiques, l'*apparatus testium* (ou *testimonia*) et l'apparat critique, il faut l'indiquer dans le préambule en utilisant la commande `DeclareApparatus` pour chaque appareil. Ils apparaîtront dans l'ordre où ils ont été déclarés :

```
\DeclareApparatus{testium}  
\DeclareApparatus{default}
```

Dans l'*apparatus testium*, l'éditeur indique les copies du texte et les passages parallèles qui se trouvent chez d'autres auteurs de l'Antiquité (tradition indirecte ou parallèle). Ces indications n'appartiennent pas à l'apparat critique et trouvent leur place à part, entre le texte et l'apparat lui-même.

Dans l'apparat critique, l'éditeur cite les leçons de tel ou tel témoin (« witness »), présente les conjectures ou les notes de tel ou tel éditeur moderne (« source »), et peut aussi prendre en compte ses propres conjectures ou celles d'un savant qui n'ont pas fait l'objet de publications (« scholar »). Il faut que chacun de ces éléments soit déclaré dans le préambule.

Les témoins C'est la commande `\DeclareWitness` qui permet de déclarer les témoins :

```
\DeclareWitness{H}{\emph{H}}{\emph{Harleianus} (Mus. Brit. 2719)}  
[origDate={s. \textsc{ix-x}}]
```

Comme on le voit, cette commande réclame trois arguments obligatoires placés entre accolades :

1. Un identifiant unique correspondant au témoin (il sera converti en `xml:id` dans le fichier TEI `xml`).
2. Le sigle tel qu'il doit apparaître dans l'apparat critique.
3. Une courte description destinée à la construction du *conspectus siglorum* qui apparaîtra au début de l'édition.

Il est ensuite possible d'ajouter des options entre crochets droits (comme la date dans notre exemple).

Les sources Pour déclarer les conjectures lues dans une édition moderne, on écrira :

```
\DeclareSource{Jacoby}{\emph{Jacoby}}
```

où le premier argument est l'étiquette utilisée dans le source .tex et le second argument ce qui apparaîtra dans l'apparat après compilation.

On notera que pour déclarer les corrections ou conjectures de l'éditeur lui-même (souvent introduites par *scripsi* ou *addidi*), on écrira par exemple `\DeclareScholar{ego}{ego}` où il est possible de préciser en option le nom de l'éditeur :

```
\DeclareScholar{ego}{ego}[  
  forename=Estelle,  
  surname=Debouy]
```

4 Les principes d'encodage avec ekdosis

L'édition et les variantes

L'encodage est très proche de la syntaxe TEI xml. Poursuivons avec l'extrait du discours de Cicéron cité plus haut :

```
\begin{ekdosis}  
O tempora, o mores! [...] Sunt vestrum aliquam multi qui L. Pisonem  
  \app{  
    \lem[wit={S, D}, alt=9noŕt, prewit=(= cognorint)]{cognorint}  
    \rdg[wit={ψ}, pre=\emph{unde}, prewit=\emph{sive} norunt]{gnorunt}  
    \rdg[wit={R}]{gonrit}  
    \rdg[wit={π}]{cognoverint}},  
huius L. Pisonis, qui praetor fuit, patrem.  
\end{ekdosis}
```

Comme on le voit, une fois l'environnement `ekdosis` ouvert, les différents éléments de l'apparat critique (ici les lemmes et les variantes) sont insérés grâce à la commande `\app`. On distingue alors la leçon retenue, saisie avec la commande `\lem`, des variantes rejetées par l'éditeur, saisies avec la commande `\rdg` (pour *reading*). Plusieurs éléments peuvent être introduits pour *rédigier* l'apparat critique : ici, c'est la commande `prewit` qui a été utilisée. Voici le fichier PDF produit :

1 O tempora, o mores! [...] Sunt vestrum aliquam multi qui L. Pisonem cognorint, huius L. Pisonis,

2 qui praetor fuit, patrem.

1 9nořt (= cognorint) S D] unde gnorunt sive norunt ψ gonrit R cognoverint π

Le fichier TEI xml produit automatiquement dans le même temps⁴ se trouve en annexe p. 10.

La traduction alignée

ekdosis permet aussi de présenter un texte et sa traduction en regard, sur la même page ou sur deux pages en vis-à-vis. Il est également possible d'aligner le texte et plusieurs traductions (par exemple si le texte a été conservé, au fil des siècles, dans plusieurs langues) ou encore d'aligner différentes recensions d'un même texte. Il faut alors utiliser l'environnement `alignment`⁵ :

```
\begin{alignment}[tcols=2, lcols=2]
  \begin{edition}
O tempora, o mores! [...] Sunt vestrum aliquam multi qui L. Pisonem
    \app{
      \lem[wit={S, D}, alt=9nořt, prewit=(= cognorint)]{cognorint}
      \rdg[wit={ $\psi$ }, pre=\emph{unde}, prewit=\emph{sive} norunt]{gnorunt}
      \rdg[wit={R}]{gonrit}
      \rdg[wit={ $\pi$ ]}{cognoverint}},
huius L. Pisonis, qui praetor fuit, patrem.
    \end{edition}

    \begin{translation}
Ô temps, ô mœurs! [...] Il y a un assez grand nombre d'entre vous qui ont connu
L. Pison, le père du dernier L. Pison qui fut préteur.
    \end{translation}
\end{alignment}
```

4. C'est ce que permet l'option `[teiexport]` ajoutée au package *ekdosis* chargé dans le préambule du source L^AT_EX.

5. On observe que plusieurs options ont été ajoutées : l'option `tcols` définit le nombre total de colonnes de texte à aligner ; et l'option `lcols` définit le nombre total de colonnes à imprimer sur la page de gauche. La valeur par défaut est 1 pour une présentation de l'édition et de la traduction sur deux pages en vis-à-vis. Dans notre exemple, la valeur choisie est 2 pour une présentation sur une même page.

Comme on le voit, deux environnements sont utilisés (*edition/translation*), le premier pour saisir l'édition et l'apparat critique (à gauche), et le second pour la traduction (à droite)⁶. Voici le PDF produit :

<p>1 O tempora, o mores! [...] Sunt vestrum ali-</p> <p>2 quam multi qui L. Pisonem cognorint, huius L.</p> <p>3 Pisonis, qui praetor fuit, patrem.</p> <hr style="width: 20%; margin-left: 0;"/> <p><small>2 ḡnořt (= cognorint) <i>S D</i>] unde gnorunt <i>sive</i> norunt ψ gon- rit <i>R</i> cognoverint π</small></p>	<p>Ô temps, ô mœurs! [...] Il y a un assez</p> <p>grand nombre d'entre vous qui ont connu</p> <p>L. Pison, le père du dernier L. Pison qui</p> <p>fut préteur.</p>
---	--

Vous trouverez en annexe p. 10 le fichier TEI xml correspondant où des xml ids ont été automatiquement attribués aux deux environnements définis dans le source .tex.

Cette présentation rapide a permis de montrer, à partir d'un exemple commenté, comment réaliser une édition critique numérique grâce à *ekdosis*. La saisie du source en LuaL^AT_EX offre à l'éditeur la possibilité de rédiger un apparat critique complexe et argumenté et de demander un export TEI xml qui lui permettra d'analyser et d'explorer les données ainsi structurées. Ce programme tient donc sa promesse de concilier les deux approches, critique et numérique, encore si rarement réunies aujourd'hui.

Annexes

Texte et apparat

```
<encodingDesc>
  <variantEncoding method="parallel-segmentation"
    location="internal" />
</encodingDesc>
</teiHeader>
<text>
  <body>
    <div xml:id="div-edition_1">
      <p>O tempora, o mores! [...] Sunt vestrum aliquam multi qui
        L. Pisonem
      <app>
```

6. Ce comportement par défaut peut être modifié grâce à l'option `texts=` qui permet de définir les noms donnés aux différentes colonnes à aligner.

```

        <lem wit="#S #D">cognorint</lem>
        <rdg wit="ψ">gnorunt</rdg>
        <rdg wit="#R">gonrit</rdg>
        <rdg wit="π">cognoverint</rdg>
    </app>, huius L. Pisonis, qui praetor fuit, patrem.</p>
</div>
</body>
</text>

```

Texte et traduction

```

<text>
  <body>
    <div xml:id="div-edition_1">
      <p>O tempora, o mores! [...] Sunt vestrum aliquam multi qui
      L. Pisonem
      <app>
        <lem wit="#S #D">cognorint</lem>
        <rdg wit="ψ">gnorunt</rdg>
        <rdg wit="#R">gonrit</rdg>
        <rdg wit="π">cognoverint</rdg>
      </app>, huius L. Pisonis, qui praetor fuit, patrem.</p>
    </div>
    <div xml:id="div-translation_1">
      <p>Ô temps, ô mœurs! [...] Il y en a un assez grand nombre
      d'entre vous qui ont connu L. Pison, le père du dernier L.
      Pison qui fut préteur.</p>
    </div>
  </body>
</text>

```