

PETITE HISTOIRE DE LA FAQ L^AT_EX GUTENBERG

Résumé

L'objet de cet article est de retracer l'historique de la foire aux questions (FAQ)²⁸ L^AT_EX GUTenberg.

Origine

Les premières traces de questions (L^A)T_EX fréquemment posées, avec leurs réponses, se trouvent sur le forum *Usenet* anglophone `comp.text.tex`²⁹ sous la forme de *posts* intitulés *T_EX*, *L^AT_EX*, etc. : *Frequently Asked Questions with Answers [monthly]*, que Bobby Bodenheimer a commencé à publier en 1990 et qu'il a progressivement enrichis.

Les premières traces de FAQ (L^A)T_EX francophone remontent à 1992 sous la forme d'un article des *Cahiers* [1] qui était une traduction française par Philippe Louarn (aidé de Jacques André, Bernard Gaulle et Éric Picheral) du *post* de Bobby Bodenheimer en date du 2 mars 1992. Une deuxième édition de cet article, [2], traduite par les membres de l'époque du bureau de l'association GUTenberg, est parue en avril 1996 lorsque la maintenance de la FAQ de Bobby Bodenheimer a été prise en charge par le TUG britannique UK-TUG.

En 1996, Fabien Torre a produit et hébergé, sous la forme d'un site Web (quasi) statique, une version HTML de cette dernière FAQ, connue sous le nom de FAQ « GRAPPA »³⁰. Celle-ci est encore disponible à l'adresse <https://faqlatex.fabien-torre.fr/>.

Parallèlement à cela, Marie-Paule Kluth a publié la FAQ en 1999 chez Vuibert [3]. Cet ouvrage bénéficia d'une réédition l'année suivante [4].

Ensuite, et jusqu'en 2004, un projet de refonte complète de la FAQ a été mené par Benjamin Bayart avec l'aide de nombreux contributeurs (dont Yvon Henel, trésorier adjoint actuel de GUTenberg). L'ambition était notamment de fournir une version PDF de cette FAQ. Les défis techniques que cela impliquait, outre l'absence d'outils efficaces de travail collaboratif, semblent avoir eu raison de la survie du projet.

FAQ GUTenberg (2013–2020) : Dokuwiki

À partir de 2013, Jérémy Just (alors président de GUTenberg), avec l'aide de Maxime Chupin (secrétaire adjoint actuel de GUTenberg) et Manuel Pégourié-Gonnard, migre la FAQ de Benjamin Bayart sur système Dokuwiki : l'idée d'une version PDF est abandonnée au profit de celle d'un site Web dynamique, simplifiant notamment le travail collaboratif des contributeurs. Jérémy Just y ajoute les questions/réponses de la FAQ (L^A)T_EX anglophone et

28. Traduction libre habituelle de *Frequently Asked Questions* qui, contrairement à la traduction formelle « questions fréquemment posées », conserve l'acronyme « FAQ ».

29. Les archives jusqu'au 22 février 2024 en sont conservées à l'adresse <https://groups.google.com/g/comp.text.tex>.

30. « GRAPPA » pour le *Groupe de recherche en Apprentissage Automatique*, au sein de l'université de Lille.

commence progressivement à les traduire. Il est rejoint dans l'aventure par 7 personnes dont (chronologiquement) :

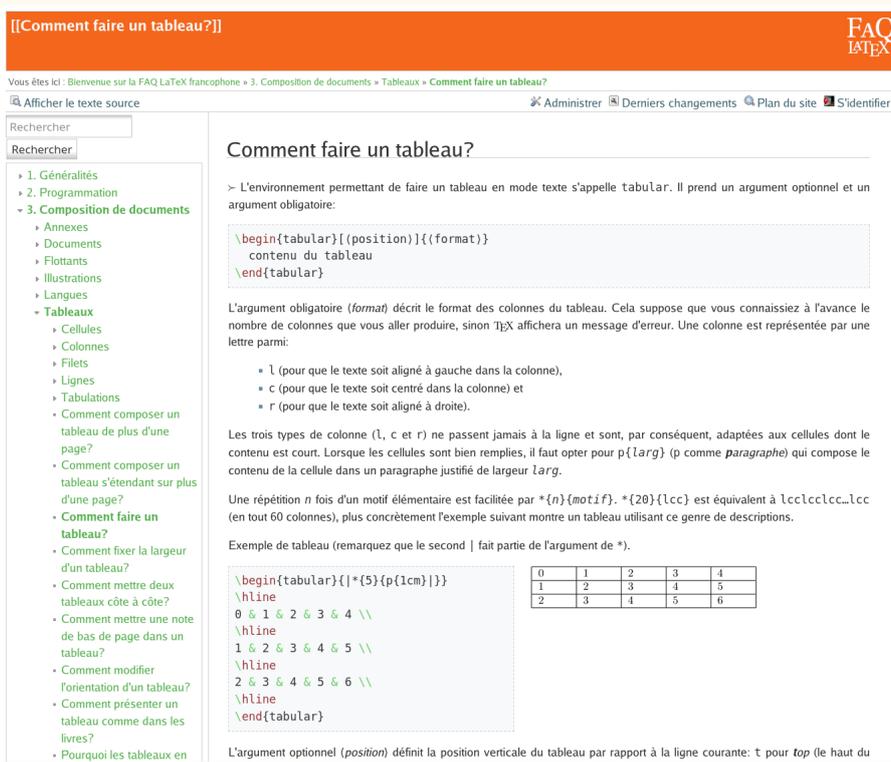
- Yvon Henel;
- moi-même, Denis Bitouzé, auteur du présent article³¹;
- Yannick Tanguy;
- Bastien Dumont.

Le travail accompli par Jérémy Just, remarquable, est à saluer. Tout d'abord techniquement, l'adaptation de Dokuwiki aux besoins d'une FAQ L^AT_EX n'a certainement pas été une mince affaire. Ensuite, particulièrement grâce à la contribution essentielle de Yannick Tanguy, un nombre incalculable de questions/réponses ont été améliorées, actualisées, traduites mais aussi créées³² et ce, de façon à la fois significative et pertinente.

Dès la parution de cette FAQ Dokuwiki, j'ai voulu y contribuer mais ai été rebuté par plusieurs aspects de ce site et du système sous-jacent :

- son design général tel qu'on peut le voir figure 1;

FIGURE 1 – Exemple de page de la FAQ sous Dokuwiki (vue ordinateur)



- son caractère non ou, désormais, piètrement réactif³³, tel qu'on peut le voir figure 2 page suivante;
- l'ergonomie du site. Par exemple la barre de recherche n'est accessible que si l'on se trouve en haut d'une page (cf. capture d'écran de droite de la figure 2 page suivante);

31. Comme expliqué ci-après, significativement seulement depuis 2020 (mais assez peu sur le contenu).

32. Il y a actuellement plus de 1200 questions/réponses!

33. Un site Web réactif (en anglais, *responsive web design*) permet aux pages Web de modifier leur disposition et leur apparence pour s'adapter à différentes largeurs d'écran, résolutions, etc. Cf. https://developer.mozilla.org/fr/docs/Learn/CSS/CSS_layout/Responsive_Design.

FIGURE 2 – Exemple de page de la FAQ sous Dokuwiki (vue téléphone portable)

The screenshot shows a mobile view of a Dokuwiki page titled "Comment faire un tableau?". The page layout includes a navigation menu on the left with items like "1. Géné", "2. Prog", "3. Com de docu", "A", "D", "F", "II", "L", and "T". Below the menu is a search bar with the text "Rechercher". The main content area features a heading "Comment faire un tableau?" and a paragraph explaining the LaTeX environment `tabular`. A code block shows the syntax: `\begin{tabular}[\langle position \rangle]{\langle format \rangle}`. Below the code block, it states: "L'argument obligatoire `\langle format \rangle` décrit le format des".

- l'éditeur en ligne Dokuwiki permettant de contribuer à la FAQ est très peu efficace et ergonomique, notamment par rapport aux éditeurs puissants (tels qu'Emacs) dont on peut disposer en local sur sa machine;
- la syntaxe Dokuwiki, très lourde;
- l'impossibilité de faire des modifications en masse (éventuellement scriptables), par exemple pour remplacer 180 occurrences dans 74 fichiers de :

Exemple 3

```
1 \documentclass{...}
2 ...
3 \usepackage[french]{babel}
```

par :

Exemple 4

```
1 \documentclass[french]{...}
2 ...
3 \usepackage{babel}
```

Comme Jérémy Just refusait d'envisager toute autre solution technique, je n'ai quasiment pas contribué à cette FAQ et m'en suis désintéressé. En novembre 2020, après mon élection au CA de GUTenberg, j'ai fait part de mon constat qui a été partagé par la plupart des administrateurs et, comme j'ai eu l'espoir d'être mieux entendu par Jérémy Just, j'ai commencé à prospecter d'autres solutions, avec comme prérequis que :

- les sources de la FAQ soient en syntaxe populaire, si possible... L^AT_EX!
- les sources soient stockables sur un dépôt Git³⁴ afin de, sinon faciliter, du moins rendre très efficace le travail collaboratif;
- ce dépôt Git soit public afin que chaque (potentiel) contributeur puisse le cloner pour pouvoir le modifier dans son environnement de travail habituel et, au besoin, en masse;
- un moteur de recherche soit *intégré*, notamment pour ne pas devoir recourir, comme c'est le cas du site Web (statique) de la FAQ L^AT_EX anglophone, au moteur de recherche Google (et devoir afficher que la recherche est... « *enhanced by Google* »).

FAQ GUTenberg (2020–2023) : Dokuwiki → Sphinx ?

En décembre 2020, j'ai été en mesure de proposer une alternative répondant à ces prérequis : Sphinx. En effet, ce générateur de documentation :

- offre de nombreux formats de sortie :
 - HTML (site Web statique);
 - PDF (via L^AT_EX)!
 - EPUB, XML, Texinfo, pages « *man* », *plain text*;
- propose comme langage de balisage pour ses sources :
 - REST³⁵, dans l'esprit voisin de L^AT_EX;
 - Markdown possiblement, qui est toutefois moins puissant que REST;
 - malheureusement *pas* L^AT_EX!
- pour sa sortie HTML offre (par défaut et gratuitement) :
 - un moteur de recherche *intégré*;
 - un design moderne et réactif;
- est plus qu'éprouvé, avec plus de 15 ans d'existence, et peut être enrichi grâce à une multitude d'extensions;
- est utilisé pour de très nombreuses documentations techniques, dont celle de Python;
- est écrit (et donc personnalisable) en Python.

Néanmoins, cette alternative présentait quelques inconvénients potentiels :

- Sphinx utilise comme langage de balisage :
 - définitivement pas L^AT_EX. Toutefois, on peut envisager une conversion depuis et vers L^AT_EX via pandoc;
 - REST dont la syntaxe est au moins aussi horrible et ésotérique que celle de Dokuwiki!

34. Système de gestion de versions décentralisé.

35. REST est le langage originel utilisé par Sphinx et celui qu'il propose avant tout autre.

- [Markdown](#) qui n’offre pas la finesse de $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ou `rest`. Toutefois, c’est un langage de balisage :

simple : léger, facile à lire et à écrire, comme dans les mails écrits en mode texte ;

populaire : de plus en plus massivement utilisé (et en tout cas bien plus que le langage de balisage de [Dokuwiki](#)) : fichiers `README.md` sur le [CTAN](#), *issues*³⁶ sur GitHub ou [GitLab](#), messages dans [Discourse](#)³⁷, etc.

- Contrairement à [Dokuwiki](#), [Sphinx](#) n’est pas un outil tout en un. Les fichiers sources de chaque question étant édités indépendamment les uns des autres, il n’y a à priori pas de facilités pour, dans le fichier de l’une d’elles, faire référence à une autre question, voire à une section d’une autre question. On verra ci-dessous que l’éditeur intégré de [GitLab](#) satisfait ce besoin.

Je me suis toutefois lancé dans l’aventure et, en décembre 2020, j’ai travaillé à un prototype, au moins pour une sortie `HTML`. Ce ne fut pas chose aisée du fait de la nécessaire conversion des sources [Dokuwiki](#) vers `rest`, effectuée par le truchement de [pandoc](#), de *custom writers* de ce dernier, de `sed`, de `Perl`, de scripts `bash`, etc. Cette migration fut aussi l’occasion de nombreuses améliorations tant ponctuelles en amont sur le site source [Dokuwiki](#) (mise en conformité et homogénéisation du code) qu’en masse en aval sur les fichiers `rest`.

Courant février 2021, ce prototype a été soumis au CA de Gutenberg et son accueil par ceux qui se sont exprimés a été, hors Jérémy Just, très favorable. Jérémy Just, quant à lui, n’a donné son avis qu’à une seule reprise : « C’est plutôt chouette ! Le look est plus moderne que celui qu’on a avec notre template [Dokuwiki](#) actuel (qui date de 2017) » ; sinon, silence radio : ni approbation, ni désapprobation. J’ai continué à améliorer progressivement ce prototype, par exemple en lui adjoignant un thème moderne, avec mode clair ou sombre, ou en lui greffant [Hypothes.is](#) qui est un service en ligne gratuit permettant d’annoter, commenter ou surligner n’importe quelle page Web.

Fin janvier 2021, le projet a été présenté aux adhérents lors d’un moment d’échange en ligne et les retours furent peu nombreux mais très favorables. De février 2021 à juin 2023, j’ai dû en arrêter le développement suite à divers points bloquants :

- j’avais l’ambition de pouvoir accompagner les exemples $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ minimaux du code (notamment du préambule) nécessaire pour qu’ils soient complets. Ainsi l’exemple montrant comment mettre en couleur du texte :

Exemple 5

```
1 \textcolor{red}{Important}
```

aurait été accompagné, par exemple au moyen d’onglets, du code compilable :

36. Rapports de bogues, demandes de fonctionnalités, etc.

37. Logiciel libre pour forum de discussions intégrant un système de liste de diffusion.

Exemple 6

```

1 \documentclass[french]{article}
2 \usepackage[T1]{fontenc}
3 \usepackage[utf8]{inputenc}
4 \usepackage{lmodern}
5 \usepackage[a4paper]{geometry}
6 \usepackage{xcolor}
7 \usepackage{babel}
8 \begin{document}
9 \textcolor{red}{Important}
10 \end{document}

```

Cela s'est révélé difficile à implémenter et plusieurs pistes explorées n'aboutissaient pas.

- Par ailleurs, la syntaxe `rest`, trop complexe, me paraissait rédhibitoire et à abandonner impérativement au profit de `Markdown`. Cela n'est toutefois pas directement possible et repose sur l'extension tierce `MyST`³⁸ de `Sphinx` qui, pendant plusieurs mois, a été incompatible avec les versions récentes de ce dernier qui offraient certaines fonctionnalités que je jugeais intéressantes, voire indispensables.
- Enfin, j'ai été accaparé par le développement de la classe `LATEX letgut` de la *Lettre GUTenberg*.

En juin 2023, ces points bloquants ont été levés :

- L'idée d'accompagner chaque exemple `LATEX` minimal non complet d'une version complète a été abandonnée au profit de blocs de code éditables et compilables³⁹ tels qu'on peut les voir en action sur les sites <https://www.learnlatex.org/fr/> ou `TEXnique`. En particulier, les fragments de code peuvent être complétés automatiquement de façon à devenir des documents minimaux complets !
- L'incompatibilité entre les versions de `MyST` et `Sphinx` a été levée suite à une mise à jour majeure du premier qui a eu en outre l'avantage de rendre les sources en `Markdown` aussi riches en fonctionnalités que celles en `rest` et, dans l'esprit, voisines de `LATEX`.

J'ai donc à nouveau travaillé d'arrache-pied au développement de ce projet (en faisant au passage de nombreux rapports de bugs et demandes de fonctionnalités pour de multiples projets : `pandoc`, `Sphinx`, `MyST`, `TeXLive.net`, etc.) Il m'a notamment fallu me plonger dans le code JavaScript de `TeXLive.net` afin de l'adapter aux particularités de la FAQ. Les résultats étaient encourageants : de plus en plus de fonctionnalités intéressantes étaient intégrées et le design du site s'améliorait continuellement.

Néanmoins, plusieurs questions se posaient :

38. `MyST` est un strict sur-ensemble de `CommonMark` dont un des développeurs principaux est John MacFarlane, l'auteur de `pandoc` ; `CommonMark` propose une spécification syntaxique qui règle le problème majeur que présente la syntaxe de `Markdown` de n'être ni standard, ni non ambiguë.

39. Cette fonctionnalité s'appuie sur le serveur `TeXLive.net` qui accepte des documents `LATEX` via une requête `HTTP POST` et renvoie un document `PDF` ou un fichier journal en cas d'erreur. Nous remercions vivement David Carlisle d'avoir développé et, avec Stefan Kottwitz, de fournir à la communauté cet outil extrêmement pratique !

1. celle du flot de travail collaboratif des contributeurs de la FAQ : celui-ci se devait d'être certes efficace, ce qui impliquait de passer par `Git`, mais également relativement simple, comparable à celui du site sous `Dokuwiki`. Or la prise en main de `Git` est quand même relativement complexe. Toutefois, des plateformes telles que `GitLab` peuvent rendre presque transparente la gestion des dépôts sous `Git`.
2. Celle de l'édition des fichiers sources de la FAQ : les contributeurs qui ne souhaitent pas cloner sur leur machine le dépôt `Git` de la FAQ doivent pouvoir éditer ces fichiers de façon simple et, si possible, ergonomique. `GitLab` offre depuis longtemps la possibilité, pour chaque fichier individuellement, d'effectuer des modifications via un éditeur en ligne de type `WYSIWYG` (prenant en charge la syntaxe `Markdown`) et de fournir un message de *commit*⁴⁰ indiquant la nature des modifications (comme c'était le cas sur le site `Dokuwiki`). Mais le fait de ne pouvoir modifier qu'un fichier à la fois finit par être peu ergonomique. Toutefois, `GitLab` intègre depuis quelques temps un `EDI` en ligne permettant d'éditer plusieurs fichiers à la fois. Cet environnement est basé sur (une version limitée de) l'éditeur de code `Visual Studio Code`, qui est de plus en plus populaire et donc potentiellement déjà utilisé par certains contributeurs ; il offre en outre de nombreuses et très puissantes fonctionnalités :
 - d'édition, notamment en ce qui concerne les fichiers `Markdown`⁴¹ ;
 - de recherche dans l'ensemble des fichiers du dépôt ;
 - d'intégration de `Git`, notamment de gestion des *commits* ;
 - etc.

L'intégration de cet `EDI` dans `GitLab` présente l'avantage qu'on peut y accéder directement depuis le site Web statique, simplement en cliquant sur une icône, ce qui simplifie énormément le flot de travail des contributeurs qui choisissent de ne pas travailler en local sur leur machine avec un clone du dépôt de la FAQ.

3. Celle de la prise en compte des modifications apportées par les contributeurs qui, sur le site sous `Dokuwiki`, était instantanée. Or `Sphinx` procède de façon similaire à `LATEX` : les sources, `rest` ou `Markdown`, doivent être *compilées* pour produire des fichiers « intéressants », notamment `HTML`⁴². Afin que les contributeurs puissent rapidement voir l'effet de leurs modifications, il était essentiel que ces compilations puissent être lancées automatiquement, sans intervention manuelle d'un des responsables de la FAQ. Or, justement, `GitLab` met à disposition un système `CI/CD` permettant cela : les pages `HTML` sont régénérées automatiquement à chaque fois que l'on enregistre des modifications sur le dépôt `Git` de la FAQ.
4. Celle de la mise en ligne du site Web statique ainsi obtenu : il était là aussi essentiel cela se fasse automatiquement, sans intervention manuelle d'un des responsables de la FAQ. Or, justement, `GitLab` met à

40. Dans notre cas, il s'agit en gros de l'enregistrement effectif de modifications apportées à un ou plusieurs fichiers sources de la FAQ.

41. Cf. par exemple <https://code.visualstudio.com/docs/languages/markdown>. On pourra y constater que cet `EDI` offre justement des facilités pour, dans le fichier d'une question, faire référence à une autre question, voire à une section d'une autre question (cf. un potentiel *inconvenient* signalé ci-dessus).

42. Mais aussi par exemple `PDF` via des fichiers `.tex`.

disposition la fonctionnalité « [Pages](#) » qui permet de publier des sites Web statiques directement à partir d'un dépôt dans [GitLab](#). Ainsi, les modifications apportées aux sources de la FAQ, immédiatement prises en compte grâce au CI/CD dès qu'elles sont « poussées » sur le dépôt, provoquent automatiquement une nouvelle compilation des pages HTML concernées et deviennent ainsi rapidement consultables par tout un chacun.

Au début de l'été 2023, les tests que j'ai menés avec ces fonctionnalités de [GitLab](#) montraient des résultats impeccables en terme d'ergonomie. En revanche, en termes de réactivité, j'ai fait face à un problème complètement bloquant ; en effet, la modification d'un unique fichier source [Markdown](#) provoquait :

localement sur ma machine la régénération quasi-instantanée de l'unique page HTML correspondante ;

en ligne sur l'instance [GitLab](#) la régénération des 1226 pages HTML correspondant aux 1226 fichiers sources que comptait alors la FAQ !

En plus d'être tout sauf écologique, cette consommation excessive de ressources durait à chaque fois au moins 15 minutes, ce qui signifiait que le contributeur devait, après chaque modification des fichiers sources « poussés » sur le dépôt [GitLab](#) de la FAQ, attendre tout ce temps pour en voir l'effet sur le site en ligne.

J'ai pensé à priori que ce détail prendrait une journée à être réglé ; j'ai déploré à posteriori que ce détail ait pris tout l'été à être réglé... Mais il a été réglé ! (Pour ceux que cela intéresse, mes péripéties à ce sujet sont détaillées en annexe, page 31.) Dès lors, la modification d'un unique fichier source [Markdown](#) ne provoquait plus que la compilation de l'unique page HTML correspondante et le délai pour que le site soit actualisé était de l'ordre de la minute.

Un autre point mérite d'être détaillé. Dans la FAQ [Dokuwiki](#), pour que « [L^AT_EX](#) »⁴³ apparaisse bien sous la forme de son logo, le contributeur devait recourir à une syntaxe assez lourde : `<latex>\LaTeX</latex>` ; ensuite, à l'enregistrement du fichier source, un plugin [Dokuwiki](#) lançait à la volée une compilation [L^AT_EX](#) puis en faisait une image [PNG](#) qui, enfin, était intégrée à la page HTML générée. Pour le lecteur, le résultat était très sous-optimal d'un point de vue typographique, ainsi que le montre la figure 3 : l'image était pixellisée, et la fonte et la famille (serif/sans serif) étaient différentes de celles de l'ensemble du site.

FIGURE 3 – Exemple de logos [T_EX](#) et [L^AT_EX](#) sur la FAQ sous [Dokuwiki](#)

Vous trouverez ici les réponses aux questions les plus courantes sur [T_EX](#), [L^AT_EX](#) et outils associés.

Lors de la migration vers [Sphinx](#), il s'est avéré qu'une telle astuce à base de compilation [L^AT_EX](#) à la volée n'était pas possible. Et, comme j'estimais de toute façon que la méthode basée sur l'inclusion d'images était très sous-optimale, j'ai cherché à faire autrement. J'ai pu mettre au point une solution *en aval*, basée sur un code JavaScript qui, à l'affichage des

43. Ainsi que « [T_EX](#) », etc.

pages HTML, remplaçait les occurrences du code « `{\latexlogo}`LaTeX`` » (syntaxe restant assez lourde pour le contributeur) par le code du listing 1.

Listing 1

```
<span class="t-logo">
  L<span class="a">a</span>T<span class="e">e</span>X
</span>
```

Une classe `css` que j'avais conçue produisait alors un rendu à la fois :

- conforme au véritable logo de Leslie Lamport (ou presque, cf. ci-dessous);
- en HTML et pas sous forme d'image;
- ne jurant pas d'un point de vue typographique (fonte et famille identiques à celles de l'ensemble du site).

FIGURE 4 – Exemple de logos `TeX` et `LaTeX` sur la FAQ sous `Sphinx`

Vous trouverez ici les réponses aux questions les plus courantes sur `TeX`, `LaTeX` et outils associés.

Or, lors du `tug` à Bonn mi-juillet 2023, Gerd Neugebauer, connu en particulier pour être le mainteneur du site Web du `CTAN`, a (bien) voulu regarder comment fonctionnait la FAQ `Sphinx`, notamment du point de vue du contributeur. Je ne sais plus pourquoi, la page HTML que j'ai choisie au hasard affichait « `LaTeX` » et non « `LaTeX` », et il en a fait le reproche. Je lui ai montré qu'il s'agissait là d'un accident, l'ensemble du site affichant bien le logo; ensuite, par simple clic sur une icône de la page Web, nous avons abouti au fichier source de cette page et Gerd a vu par quelle syntaxe cela passait pour le contributeur (« `{\latexlogo}`LaTeX`` », donc), ce qu'il m'a signalé comme étant inutilement lourd. En effet, pour le site du `CTAN`, il a pour cela une solution *en amont* : avant leur publication, les fichiers HTML sont tous scannés par un script `Perl` qui y remplace les occurrences de « `LaTeX` » (seulement) par le code HTML du listing 1; ensuite, là aussi grâce à un fichier `css` de son cru, le rendu est conforme au véritable logo de Leslie Lamport. Gerd m'a très aimablement fourni son script et son fichier `css`, ce qui fait que, désormais, les contributeurs de la FAQ `Sphinx` n'ont plus qu'à écrire « `LaTeX` » dans les fichiers sources pour que, dans les pages HTML résultantes, ce soit bien « `LaTeX` » qui apparaisse! En outre, dans la classe `css` que j'avais conçue, mes choix de positions des lettres « `A` » et « `E` » étaient selon moi élégants mais n'étaient en fait pas tout à fait conformes au logo de Leslie Lamport; en revanche, avec le fichier `css` de Gerd, le rendu est conforme. Nous remercions vivement Gerd pour tout cela et pour avoir été à l'écoute des retours que Bastien Dumont (qui a récemment rejoint de `ca` de `GUTenberg`) et moi avons faits pour signaler de légers bugs et proposer des pistes visant à rendre plus puissante encore sa solution⁴⁴.

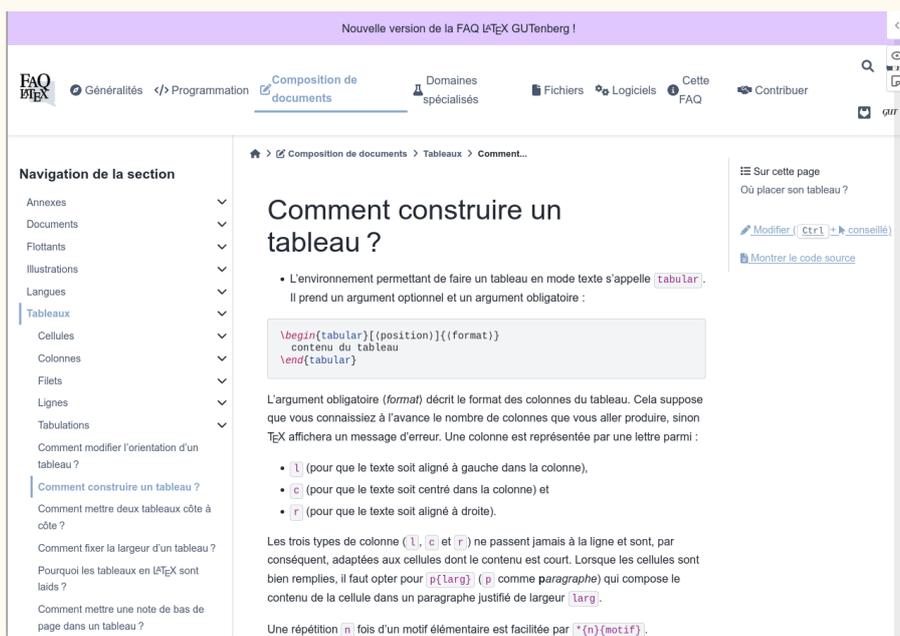
44. C'est désormais un *fork*, créé par Bastien, qui est en œuvre sur le site de la FAQ.

FAQ GUTenberg (2023–aujourd’hui) : Dokuwiki → Sphinx!

En septembre 2023, le CA de GUTenberg a considéré que le projet était suffisamment mature pour que la FAQ [Dokuwiki](#) soit migrée vers [Sphinx](#). Néanmoins, nous souhaitions réduire encore le temps de compilation et pour cela, il semblait nécessaire (et suffisant) que nous disposions de notre propre instance [GitLab](#) ainsi que de notre propre [GitLab runner](#)⁴⁵. Ceci nécessitait une montée en puissance de notre serveur, ce qui a été acté par le CA.

Maxime Chupin a alors réalisé un considérable travail d’administration système pour installer, via des images [Docker](#), nos propres instances de [GitLab](#)⁴⁶ et [GitLab runner](#), mais aussi pour trouver une alternative « manuelle » à [Pages](#) que nous ne parvenions pas à faire fonctionner sur notre instance; et ça n’aura pas été un travail inutile puisque, désormais, le temps mis au processus pour actualiser le site après modification d’un unique fichier source [Markdown](#) est passé d’un peu plus d’une minute à environ 30 s. Maxime a également testé la FAQ du point de vue du contributeur et en a créé un nouveau logo.

FIGURE 5 – Exemple de page de la FAQ sous Sphinx (vue ordinateur)



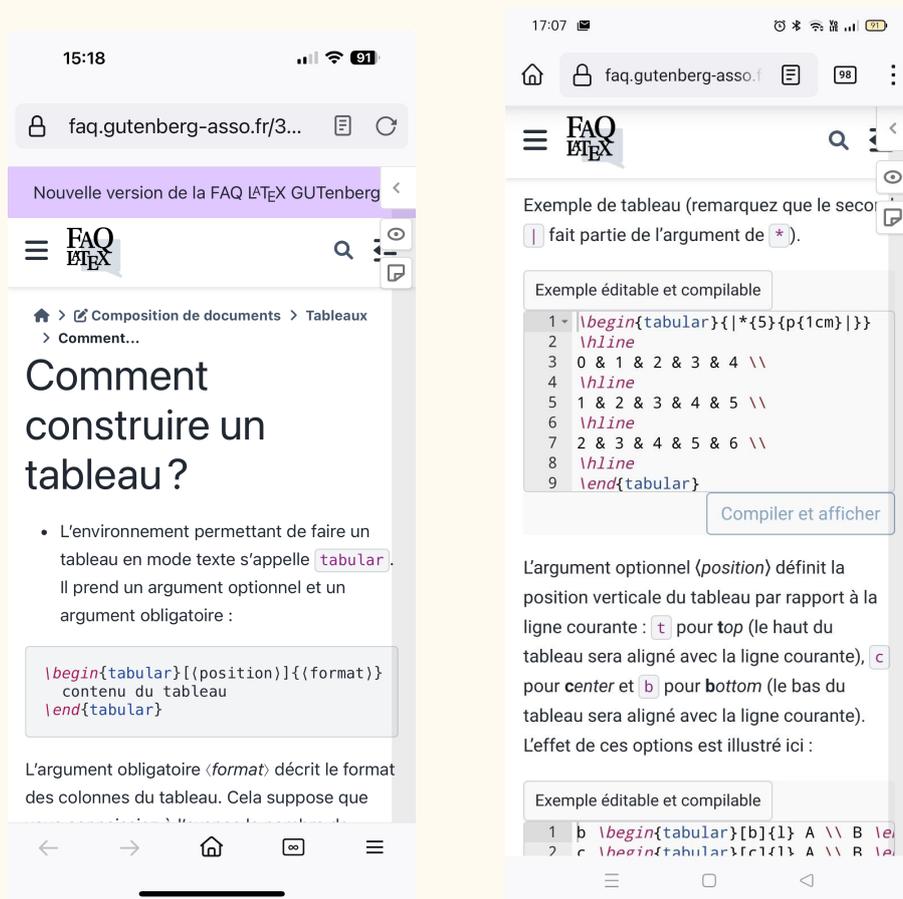
Maxime et moi avons aussi cherché à élucider un mystère... En effet, début 2023, JérémY Just a mis en ligne un *fork* de la FAQ GUTenberg et ce, sans l’annoncer au CA de l’association, ni avant, ni après l’avoir fait. Il va sans dire que nous déplorons au plus haut point cet état de fait car il dessert considérablement la communauté des utilisateurs francophones de L^AT_EX : ceux qui cherchent de l’aide se retrouvent désormais avec deux FAQ divergeant progressivement! Mais ce qui reste mystérieux, c’est que

45. Le *runner* est un logiciel, connecté à l’instance [GitLab](#), qui se met en écoute des demandes de tâches, notamment donc de la compilation de la FAQ lorsqu’un *commit* est « poussé » sur le dépôt.

46. Cette instance se trouve à l’adresse <https://gitlab.gutenberg-asso.fr> et a pour but d’héberger aussi les dépôts des productions francophones autour de (L^A)T_EX.

depuis début 2023 et jusqu'à récemment (donc que ce soit avant ou après la migration vers *Sphinx*), les moteurs de recherche, notamment Google, n'ont plus *du tout* répertorié la FAQ GUTenberg (qui n'a pas changé d'URL entre temps); et seule celle de Jérémie Just était répertoriée, et très bien répertoriée⁴⁷! Même si nous avons notre idée sur la question, on peut se demander quelle est l'explication de ce mystère.

FIGURE 6 – Exemple de page de la FAQ sous *Sphinx* (vue téléphone portable)



Finalement, après de nombreux ajustements complémentaires, la migration de la FAQ de *Dokuwiki* vers *Sphinx* a eu lieu fin novembre 2023. Elle peut être trouvée à l'adresse suivante :

<https://faq.gutenberg-asso.fr>

mais voici des captures d'écran, à comparer avec les précédentes, montrant le gain en terme de design et d'ergonomie.

47. Heureusement, grâce à Maxime qui s'est plongé dans *Google Search Console*, la FAQ GUTenberg est désormais (depuis début 2024 environ) répertoriée, et régulièrement mieux que celle de Jérémie Just.

Plusieurs contributeurs ont rejoint cette nouvelle version de la FAQ GUTenberg :

- certains anciens : Maxime Chupin, moi-même, Bastien Dumont et Yannick Tanguy (toujours aussi extraordinairement actif!);
- mais aussi des nouveaux : Patrick Bideault, Yann Denichou, Bernard Peyréga et Wissam Sabra.

Un grand merci à eux tous!

FAQ GUTenberg (aujourd’hui – futur)

Nous espérons que cette nouvelle version [Sphinx](#) de la FAQ, avec toutes les fonctionnalités qu’elle apporte⁴⁸, rendra service à la communauté francophone des utilisateurs de (L)A^AT_EX.

Nous espérons également qu’elle suscitera des vocations de contributeur⁴⁹ : cher lecteur, c’est avec plaisir que la joyeuse équipe des contributeurs actuels vous accueillera en son sein!

Pour le futur, il est évident que nous allons continuellement améliorer, corriger, actualiser et développer la FAQ. Parmi les pistes possibles de développement, il serait intéressant et certainement très utile est de concevoir une intelligence artificielle basée sur le trésor que représente le très riche contenu de cette FAQ.

Annexe : détails de mes péripéties

Je livre ici les détails des péripéties évoquées ci-dessus.

1. Je me suis donc battu de début juillet à mi-août 2023, parfois pendant des journées entières, pour régler un problème de la nouvelle FAQ extrêmement bloquant : la modification d’un unique fichier source provoquait la régénération, non pas seulement de l’unique page HTML correspondante, mais des 1226 pages HTML des 1226 fichiers sources! J’ai remué ciel et terre, posé la question sur un nombre incalculable de forums, ai sollicité Maxime Chupin, Paul Gaborit, Georges Khaznadar, ai passé des heures à voir si ça marcherait mieux sur une instance auto-hébergée de [GitLab](#) avec un *runner* auto-hébergé (ça ne marchait pas mieux mais j’ai réussi à créer une telle instance), etc. Je suis passé par des moments de doute terribles. Et puis, finalement, je suis tombé tout à fait par hasard et, surtout, par miracle (cf. ci-dessous) sur une [solution, extrêmement simple](#), qui a fait passer la régénération, après modification d’un unique fichier source, d’environ 15 minutes à un peu plus d’une minute.
2. Ce qui est tout à fait ahurissant dans cette histoire, c’est que je suis donc tombé sur la solution ci-dessus, non seulement tout à fait par hasard, mais aussi par pur miracle. En effet, la machinerie `ci/cd` décrite ci-dessus nécessite un système d’exploitation et un certain nombre de programmes qui, pour des questions de sécurité, de reproductibilité,

48. Détaillées dans l’article « [Fonctionnement de la FAQ L^AT_EX GUTenberg](#) » page 33.

49. Cf. l’article « [Fonctionnement de la FAQ L^AT_EX GUTenberg](#) », section « [Améliorations](#) » page 46.

d'automatisation, etc., sont rendus disponibles dans des environnements virtualisés isolés via des « images », par exemple des images **Docker**. Pendant des semaines, l'image **Docker** que j'ai utilisée était une brique immuable constituée d'une distribution Linux minimale agrémentée d'une distribution **Python** (nécessaire puisque **Sphinx** est codé en **Python**) sur laquelle, à chaque modification de la FAQ, étaient ensuite installés un certain nombre d'outils dont notamment **Sphinx**, l'ensemble lançant enfin la compilation **Sphinx** des fichiers sources. Je me disais depuis longtemps qu'il faudrait que je parte plutôt d'une image **Docker** intégrant déjà **Sphinx** : Paul Gaborit et Georges Khaznadar l'ont fait pour moi début juillet 2023 et ça montrait une légère accélération du processus de démarrage, mais rien au niveau de la compilation ; j'avais cessé d'utiliser ces images car elles utilisaient la version 6.2.1 de **Sphinx** alors que, à partir de mi-juillet, je pouvais (et devais pour des raisons de compatibilité) utiliser la version la plus récente (7.1.2) de **Sphinx**. Lorsque, en désespoir de cause, j'ai décidé mi-août d'à nouveau tester une image **Docker** intégrant **Sphinx**, je ne pouvais me baser sur celle officiellement fournie par **Sphinx** car elle était basée sur une version de **Sphinx** antérieure à la 7.1.2 ; en butinant sur Internet, j'ai trouvé une autre image **Docker** intégrant **Sphinx 7.1.2** qui, elle, a comme par magie réglé le problème de compilation interminable ! J'ai signalé la chose aux développeurs de **Sphinx** qui quelques jours plus tard ont fourni une nouvelle image **Docker** officielle intégrant **Sphinx 7.1.2** ; je l'ai testée mais, pour une raison toujours pas claire, le problème de compilation interminable persistait avec cette image officielle. La solution trouvée relève donc du miracle : j'aurais pu tomber sur mille autres images **Docker** intégrant **Sphinx 7.1.2** avec lesquelles le problème aurait persisté !

Références

- [1] Bobby BODENHEIMER. « $\text{T}_{\text{E}}\text{X}$, $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, etc. questions et réponses ». fr. Trad. par Philippe LOUARN et al. In : *Cahiers GUTenberg* 13 (1992), p. 55-77. URL : http://www.numdam.org/item/CG_1992__13_55_0/.
- [2] GUTENBERG'S OFFICE. « FAQ : questions souvent posées sur $(\text{L})\text{T}_{\text{E}}\text{X}$ ». fr. In : *Cahiers GUTenberg* 23 (1996), p. 1-114. URL : http://www.numdam.org/item/CG_1996__23_1_0/.
- [3] Marie-Paule KLUTH. *FAQ $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ française. Pour débutants et confirmés*. Paris : Vuibert, 1999. 440 p. ISBN : 2-7117-8641-2.
- [4] Marie-Paule KLUTH. *FAQ $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$. Foire aux questions LATEX*. 2^e éd. Paris : Vuibert, 2000. 441 p. ISBN : 2-7117-8662-5.

Denis Bitouzé

