ARTRAGIS

[Chronique]Zeste Of Dev 14

1er octobre 2019

Table des matières

1. La roadmap zds 1
   1.1. v28.2, ça c'est fait 1
   1.2. Roadmap de la version 29 2
2. Pérégrination du développement : le SVG 2
   2.1. Contexte 2
   2.2. Le problème 3
   2.3. Intégrer SVG à easy-thumbnail 3
   2.4. Chercher l'alternative 3
3. Et ce qui est en cours ? 4
   3.1. Zds-Site 4
   3.2. Zmarkdown 4
   3.3. zds-editor 4

Y'a pas, cette chronique bat des record de ponctualité, c'est donc non pas un mois mais un semestre après la dernière chronique que je vous livre l'avancement et les pérégrinations de l'équipe de développement de zeste de savoir !

1. La roadmap zds

1.1. v28.2, ça c'est fait

Comme promis en septembre, nous avons lancé la version 28.2 du site, elle n'a pas intégré l'API des contenus mais nous avons vraiment eu de grosses améliorations :

— nouvelle page de profile, avec la fonctionnalité de signalement
— améliorations globale sur les exports (corrections des epub, améliorations des PDF, lanceur d'export plus stable...)
— bugfixes d'accessbilité
— amélioration des imagettes twitter
— commande de publication pour les plus gros tutos (comme le tuto de C)

Pour autant nous avons dû passer par un hotfix très rapidement, en effet bien que le lanceur d'export soit beaucoup plus fiable que le précédant il possède encore quelques faiblesses qui force à le relancer (mais bon contrairement au précédant le rattrapage est ultra rapide).

On essaie de trouver des solutions aux derniers problèmes mais là, nous allons passer au développement de la version 29.

1.2. Roadmap de la version 29

Cette version incorporera, suite à votre vote une interface de revue collaborative et les parcours.

A ce propos, nous avons lancé une consultation pour savoir ce que vous désirez vraiment dans les parcours. Un premier élément qui en est sorti, c'est que dès la v29 les validateurs pourront lier des contenus entre eux pour vous les proposer sur la page de lecture.

Ensuite, nous avons lancé un sondage pour définir ce que nous allons vraiment faire.

Enfin, nous avons aussi lancé la suite du chantier de la refonte de l'interface de rédaction. Une branche a été tirée pour faciliter le travail collaboratif.

2. Pérégrination du développement : le SVG

On entend souvent que le développement, en 2019 c'est vraiment trop facile par rapport à ce qui se faisait avant, que les développeurs n'ont plus besoin de réfléchir, juste d'intégrer entre elles des briques déjà conçues par d'autres.

Évidemment beaucoup contredisent ce jugement à l'emporte pièce, en rappelant que les défis contemporains ne sont plus les mêmes (architecture distribuées, parallélisme, IA...) mais on oublie trop souvent que l'intégration n'est pas non plus un long fleuve tranquille. Pour illustrer cela, j'aimerais vous parler d'une fonctionnalité demandée par les graphistes sur zds : le support du SVG.

2.1. Contexte

SVG, c'est l'acronyme de *Scalable Vector Graphics*, c'est un type d'image qui a une capacité assez utile : on peut l'agrandir (ou la réduire) à l'infini, elle ne sera pas pixélisée ni déformée. Autre avantage : que votre image soit donnée en 250 par 250 pixel ou en 1920 par 1700 le poids sera... le même. En effet SVG n'est qu'un descriptif de comment il faut construire l'image, c'est votre navigateur qui interprétera les instructions pour l'afficher à la bonne taille.

i

Pour information, on dit que les images SVG sont des images *vectorielles*, à l'opposée les formats tels que PNG, JPG, bitmap... sont appelés *binaires*.

Si vous essayez aujourd'hui d'ajouter un SVG à une galerie de ZDS, il vous dira que ce n'est pas une image. Vous serez donc obligé de choisir entre :

— héberger le SVG ailleurs ;
— le convertir en PNG pour ensuite l'importer.

?

Mais du coup c'est quoi le problème ?

2.2. Le problème

Afin de nous faciliter la tâche dans la gestion des images et de leurs miniatures nous avons choisi d'utiliser une bibliothèque déjà existante, solide, sécurisée qui se nomme easy-thumbnails .

Mais voilà cette bibliothèque est fortement liée à PIL, c'est-à-dire la bibliothèque standard en python pour gérer les images binaires. Autrement dit, si vous ne pouvez pas ajouter de SVG c'est parce que nous utilisons easy-thumbnail.

2.3. Intégrer SVG à easy-thumbnail

Cette bibliothèque possède un ensemble de méthodes qui permettent d'en avoir une utilisation personnalisée et souple. On parle de :

- `Field` : l'élément qui permet de garder une trace de l'image dans la BDD mais aussi de mettre un formulaire adapté
- `Processor` : un bout de programme qui transforme une image en une autre (par exemple retailler, mettre en nuance de gris...)
- `Thumbnailer` : un créateur d'imagette

J'ai donc tenté de créer des éléments personnels pour que le `Field` accepte le SVG, que le `thumbnailer` ne transforme pas un SVG en miniature etc.

Mais voilà, à un moment où à un autre vous êtes obligé de passer par `engine.process` car ce dernier est une fonction interne de la bibliothèque. `engine` est un module de la bibliothèque qui permet de gérer les images et il appelle PIL en dur, sans vous laisser le choix, et il n'est pas personnalisable.

Impossible d'intégrer le SVG à easy-thumbnail.

?

Donc on abandonne ?

Non !

2.4. Chercher l'alternative

Développer demande un peu de créativité et de recul. Pour cela une sorte de devise est placardée sur tous les murs des écoles d'ingénieur (enfin, j'espère) : KISS. Il ne s'agit pas de déposer un baiser sur votre ordinateur mais de *"Keep It Simple and Stupid"*, c'est-à-dire de faire en sorte de garder notre programme simple et le plus "stupide" à utiliser.

J'ai donc tenté de voir si on ne pouvait pas faire plus simple. Et quitte à ne pas avoir une solution parfaite tout de suite, au moins vous proposer un Produit Minimum Viable (MVP en anglais si vous voulez chercher sur votre moteur de recherche préféré).

J'ai donc fait comme si on partait de zéro, j'ai ajouté à la galerie un nouveau formulaire qui s'appelle "Ajouter un dessin" et qui permet de télécharger un SVG dans la galerie. Puis j'ai ajouté cette possibilité à l'API des galeries pour enfin modifier le script qui vous permet de télécharger une image par simple glisser/déposer.

Et voilà, vous avez un support minimum viable du SVG. Il n'est pas parfait, le fait d'avoir deux formulaires avec des vocabulaires différents n'est pas la chose la plus ergonomique, mais aujourd'hui on suppose que vous allez surtout utiliser le glisser/déposer plutôt que le formulaire dans la galerie, alors cela ressemble à un détail.

Ce MVP possède sa PR .

3. Et ce qui est en cours ?

3.1. Zds-Site

— La refonte de l'interface de rédaction, @**A-312** a démarré les hostilités avec le début du support du drag&drop pour déplacer les parties, chapitres section.
— Un script unifié d'installation sur windows
— Amélioration de l'historique des éditions sur le forum
— Flux RSS personnel des notifications

3.2. Zmarkdown

— Mise à jour de remark, pour avoir un meilleur support de GFM mais aussi (enfin !) le retour des `hl_lines` dans remark-code
— Correction du bug qui empêche d'afficher les smiley dans les PDF

3.3. zds-editor

Un projet tout nouveau issu du travail de @Sandhose pour créer un éditeur de contenu qui affiche le rendu en temps réel.

Liste des abréviations

IA Intelligence Artificielle. 2

MVP Minimum Viable Product. 3, 4

PIL Python Imaging Library. 3

SVG Scalable Vectorial Graphics. 1–4

ZDS Zeste De Savoir, Banane !. 2