

Rapport hebdomadaire semaine 7

Yann DUBOIS

May 21, 2016

1 Semaine du 16/05 au 20/05

1.1 Lundi 16/05/2016

Féié

1.2 Mardi 17/05/2016

J'ai bougé un transformer pour voir les soucis qui en découlé, j'ai donc changé les phases pour qu'elles se comporte encore comme avant mais qu'elle puisse dorénavant visiter un document. J'ai ajouté les méthodes "key" et "priority" pour pouvoir organiser les phases, étant donné que maintenant elles se comporte plus comme un pipe. Une phases prend une collection d'objet et sort une collection d'objet, ces collections peuvent être un document, une configuration ou autre. Suite à ça j'ai corrigé des tests qui n'allaient plus. Le premier transformer que j'ai bougé était celui qui permet d'inclure un fichier pillar dans un autre. J'ai eu quelques problèmes lié à Pharo, la synchronisation lors d'une modification est assez capricieuses et j'ai perdu beaucoup de temps à essayer de resynchroniser cette classe avec sa classe parente. J'ai finalement réussi à contourner le problème en changeant sa classe parente puis la remettre comme au début. J'ai aussi changé la façon dont on trie le cycle d'exécution des phases pour qu'il prenne maintenant en compte les catégories de phases et les priorités.

1.3 Mercredi 18/05/2016

Ma première idée en arrivant était de vérifier que l'inclusion des fichiers fonctionnait toujours. J'étais assez fier quand j'ai vu que ça fonctionnait toujours, l'idée de fusionner le concept de phase et de transformer est bon. J'ai donc ajouté une option dans la configuration pour activer une phase et ai changé l'action d'une phase qui transforme. Je me suis rendu compte que les tests ne passaient plus, en effet, comme les transformers se lancent uniquement si ils sont dans la liste des phases activées, il ne se lance plus. J'ai donc fait cette petite modification dans les tests. J'ai ensuite continué à bouger les transformers, je me suis rendu compte lorsque j'ai bougé les transformers qui permettent de numéroté les titres, les figures, etc..., que certains transformers fonctionnaient de paire, ici avec la numérotation des liens. Mais grâce à ma solution, le problème était vite résolu, je leur ai donné la même clé mais pas la même priorité au sein de leur catégorie. Ce qui a pour effet de faire l'un puis l'autre en les activant en même temps. L'autre problème, c'est que la numérotation des liens a besoin de connaître des données provenant de la phase qui s'est produite juste avant. Là encore le problème a été résolu grâce à ma solution. Une phase n'est pas forcée de prendre un document entré ou bien même de sortir un document, du moment que ce que l'on sort est une collection, il n'y a pas de problème. Je sors donc un dictionnaire lorsque c'est nécessaire pour y ranger l'information nécessaire. Ce qui aura pour effet que la prochaine phase aura en entrée, un dictionnaire, et il se trouve que la prochaine phase est obligatoirement la numérotation des liens.

1.4 Jeudi 19/05/2016

J'ai continué de bouger les transformers, maintenant que tout est en place, il est devenu plus simple de faire ce travail. Les footnotes, la capitalisation des titres, l'évaluation des scripts, l'inclusion d'un script à partir d'un fichier et le fait de cacher des scripts sont maintenant des phases. Chacun d'eux ont été testés sur des cas réels. Ce qui a permis de trouver des bugs sur certains transformers. Les scripts vides ne sont pas parsés correctement, ce qui est dommage car on veut que lorsqu'on inclut un script, on ait pas besoin de devoir écrire dans le script du fichier principal. J'ai donc ouvert une issue : #111¹. L'autre bug que j'ai trouvé était sur la capitalisation des titres,

¹<https://github.com/pillar-markup/pillar/issues/111>

il ne capitalisait pas que les titres, mais tout le texte qui suivait un titre qui devait être capitalisé. J'ai perdu un peu de temps à vouloir corriger ce bug, j'ai fini par le corriger avec l'aide de Thibault car c'est lui qui avait créé ce transformer. On a juste remis à zéro la variable qui permettait la capitalisation après avoir lu le texte d'un titre.

1.5 Vendredi 20/05/2016

J'ai terminé la migration des transformers, il n'en resté plus beaucoup mais m'on tout de même pris du temps car certains n'avaient pas le même déroulé qu'un transformer normal, j'ai voulu garder les plus difficiles pour la fin. Le dernier transformer qui a été bougé est le transformer des structures. Ce transformer a failli prouvé que les transformers étaient bien bien différents des phases. J'en ai parlé avec Thibault car il est aussi l'implémenteur de ce transformer en lui proposant un nouveau moyen de le faire. La discussion a été un peu longue car on cherche le pour et le contre de chaque solution. Le problème de ma solution est qu'on visite plusieurs fois les structures jusqu'à tomber sur la bonne transformation, mais d'un autre côté elle rend plus simple la lecture du code et supprime une dépendance cyclique dans les Models. La solution de Thibault permet de garder une bonne performance mais on garde la dépendance, ce qui n'est vraiment pas bon. On a trouvé une solution raisonnable mais qui nous ennuie un peu, elle consiste à mettre directement dans la classe qui définit la structure son rendu dans l'arbre de document. Ce qui nous embête c'est le rendu, on pense que ce n'est pas le rôle de la structure de savoir ça, où du moins pas directement, on en parlera à Damien lorsqu'il reviendra du Togo lundi.