

Intégrations graphiques Publik - Development #38205

CSS: gestion automatique des prefix navigateurs

04 décembre 2019 16:05 - Thomas Jund (congé, retour le 29/04)

Statut:	Nouveau	Début:	04 décembre 2019
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Non		
Description			
Je propose l'utilisation d'autoprefixer https://github.com/postcss/autoprefixer Pour générer automatiquement les préfixes navigateurs.			
Cet outil est basé sur browserlist et l'api de caniuse pour appliquer automatiquement les préfixes nécessaires à une liste de navigateur qu'on lui passe en paramètre.			
Avantages :			
<ol style="list-style-type: none">1. Notre base de code reste propre, on peut nettoyer tous les préfixes qui traînent et qui ne servent plus (comme les `vendor-prefix(...)`).2. On ne se soucie plus des préfixes pour flexbox et le jour prochain où on abandonne IE10, pas de travail de nettoyage à faire :, on enlève juste IE10 de la liste des navigateurs à supporter.3. On peut commencer à utiliser css-grid, qui apporterait des solutions à bien des problèmes de layout que nous avons.4. Et plein d'autres trucs nouveau en amélioration progressive pour les mobiles avec @supports (comme snap-scroll, position sticky, ...).			
Demandes liées:			
Lié à Intégrations graphiques Publik - Development #38662: Supprimer les vend...		Fermé	20 décembre 2019
Lié à Intégrations graphiques Publik - Development #42287: CSS: supprimer les...		Fermé	29 avril 2020

Historique

#1 - 04 décembre 2019 16:20 - Frédéric Péters

On doit dépendre d'outils qui sont disponibles dans Debian (stretch).

Il y a bien <https://packages.debian.org/search?keywords=node-postcss> et <https://packages.debian.org/search?keywords=node-autoprefixer>, ce dernier pouvant être autre chose, mais ce n'est pas pour stretch.

Pour les mini-trucs à faire, genre ajouter des préfixes pour les flexbox, ça sera plus gérable de faire le script moi/nous-même; pour virer les vendor-prefix obsolètes, de faire le taf une fois et voilà.

Pour les trucs sophistiqués, ce serait à voir au cas par cas, sans doute des trucs possibles, sans doute d'autres pas, clairement moins satisfaisant que se dire que toutes les fonctionnalités d'un module existant sont disponibles, j'en ai conscience.

#2 - 05 décembre 2019 12:07 - Thomas Jund (congé, retour le 29/04)

Je pose juste l'idée. On peut bien sûr tout gérer manuellement (avec les oublis et le fait qu'on laisse toujours traîner des trucs). Mais si on peut éviter de devoir maintenir ce code qui est forcément transitoire et nous permettre de nous lancer dans du CSS un peu plus moderne sans avoir à apprendre des syntaxes de code déjà obsolètes, je préférerais.

Pour l'utiliser depuis des années, c'est pratique, ça marche super, et c'est dur de revenir sur un confort acquis :).

#3 - 05 décembre 2019 12:09 - Frédéric Péters

On doit dépendre d'outils qui sont disponibles dans Debian (stretch).

#4 - 05 décembre 2019 12:34 - Frédéric Péters

Et je répète également « pour les mini-trucs à faire, genre ajouter des préfixes pour les flexbox, ça sera plus gérable de faire le script moi/nous-même », qui devait répondre à : « On peut bien sûr tout gérer manuellement (avec les oublis et le fait qu'on laisse toujours traîner des trucs). ».

#5 - 05 décembre 2019 13:12 - Frédéric Péters

Et dans l'idée poc, pour autoprefixer les flexbox, (en m'occupant juste de la propriété display, c'est pour l'idée); ça se générerait facilement via :

```
import css_parser

sheet = css_parser.parseFile('style.css')

for rule in sheet:
    if rule.type == rule.STYLE_RULE:
        display_property = rule.style.getProperty('display')
        if display_property and display_property.value == 'flex':
            rule.style.removeProperty('display')
            rule.style.setProperty('display', '-ms-flexbox')
            rule.style.setProperty('display', 'flex', replace=False)

open('style2.css', 'wb').write(sheet.cssText)
```

Ça me renforce dans l'idée qu'on peut avancer sur un mode "tiens il y a un plugin postcss qui fait ça, ce serait super d'avoir pareil".

Alors oui ça n'est pas le confort de pouvoir piocher dans mille plugins et avoir directement le job à disposition, mais c'est une méthode qui nous permettrait à la fois de conserver des builds automatiques, sur une infra saine, et de ne pas avoir à gérer manuellement les préfixes, ou d'autres trucs derrière.

#6 - 05 décembre 2019 16:27 - Thomas Jund (congé, retour le 29/04)

- Statut changé de Nouveau à Rejeté

#7 - 05 décembre 2019 17:14 - Frédéric Péters

J'aurais imaginé continuer à couvrir ça ici mais ça peut bien sûr être un autre ticket,

- 1/ Notre base de code reste propre, on peut nettoyer tous les préfixes qui traînent et qui ne servent plus (comme les `vendor-prefix(...)`).
- 2/ On ne se soucie plus des préfixes pour flexbox et le jour prochain où on abandonne IE10, pas de travail de nettoyage à faire :), on enlève juste IE10 de la liste des navigateurs à supporter.

sont des trucs qui, il me semble, peuvent facilement être mis en œuvre.

#8 - 06 décembre 2019 10:11 - Thomas Jund (congé, retour le 29/04)

- Sujet changé de CSS: gestion automatique des prefix navigateurs avec autoprefixer à CSS: gestion automatique des prefix navigateurs

- Statut changé de Rejeté à Nouveau

Oups Pardon.

Dans le titre de ce ticket, j'ai proposé une solution qui résoudrait certains problèmes, plutôt que d'exposer d'abord les problèmes pour débattre sur la solution à apporter.

Maladroit. J'ai donc fait le raccourci "solution proposée rejetée == ticket rejeté".

Je vire donc "avec autoprefixer" du titre et rouvre le débat sur la pertinence de l'idée et des solutions à apporter.

#9 - 06 décembre 2019 10:26 - Thomas Jund (congé, retour le 29/04)

sont des trucs qui, il me semble, peuvent facilement être mis en œuvre.

Et quid d'ouvrir la possibilité d'utiliser css grid qui demande comme flexbox l'ajout d'une ancienne spécification pour support de IE10/11? Ou l'utilisation de toutes les autres propriétés non supportés par IE10/11 en amélioration progressive ?

#10 - 06 décembre 2019 11:02 - Frédéric Péters

Et quid d'ouvrir la possibilité d'utiliser css grid qui demande comme flexbox l'ajout d'une ancienne spécification pour support de IE10/11? Ou l'utilisation de toutes les autres propriétés non supportés par IE10/11 en amélioration progressive ?

Il faut commencer par décrire ce que ça représente. (pour flexbox, on a déjà ça utilisé, donc je peux juste voir que ça veut dire taper display: -ms-flexbox, taper -ms-flex: whatever, etc.). Ensuite, c'est sans doute possible. (et oui, je notais « moins satisfaisant que se dire que toutes les fonctionnalités d'un module existant sont disponibles »).

#11 - 20 décembre 2019 11:33 - Thomas Jund (congé, retour le 29/04)

- Lié à Development #38662: Supprimer les vendor-prefix pour les propriétés transitions ajouté

#12 - 20 décembre 2019 11:33 - Thomas Jund (congé, retour le 29/04)

- Lié à Development #38662: Supprimer les vendor-prefix pour les propriétés transitions ajouté

#13 - 20 décembre 2019 11:34 - Thomas Jund (congé, retour le 29/04)

- Lié à Development #38662: Supprimer les vendor-prefix pour les propriétés transitions supprimé

#14 - 20 décembre 2019 12:12 - Thomas Jund (congé, retour le 29/04)

Concernant les préfixes pour les propriétés transform, multi-column et backface-visibility, ils dépendent du support du browser android jusqu'à la version 4.4.4 (usage 0.26%).

Pour les préfixes Flexbox, ils sont nécessaires pour IE10 et android browser 4.4.3 (usage 0.17%)

Les stats d'usage sont tirés de Canluse.

#15 - 29 avril 2020 16:13 - Thomas Jund (congé, retour le 29/04)

- Lié à Development #42287: CSS: supprimer les propriétés flexbox préfixées ajouté