RFC 8081 : The "font" Top-Level Media Type

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 1 mars 2017

Date de publication du RFC : Février 2017

https://www.bortzmeyer.org/8081.html

Les types de contenu, servant à indiquer le type des données envoyées (par le Web, par le courrier, etc) sont composés de deux parties, un type de premier niveau ("*top level type*", ou type proprement dit, c'est la catégorie des données) et un sous-type (il indique le format des données). Type et sous-type sont séparés par une barre oblique. Par exemple, `image/png` est un type MIME identifiant une image au format PNG. Des nouveaux sous-types sont enregistrés très souvent, c'est un événement banal. Mais des nouveaux types de premier niveau sont bien plus rares. Ce RFC en décrit un, le type `font/`, qui sert à identifier les formats pour des polices de caractères. Ainsi, on pourra envoyer un fichier de polices au format TTF en l'étiquetant `font/ttf`. (Notre RFC procède également à l'enregistrement de plusieurs sous-types pour des formats de polices particuliers.)

Une police de caractères, c'est une description de comment représenter visuellement un ensemble de caractères, description qu'un programme peut lire et comprendre. Il existe bien des façons de faire cette représentation. Les premiers formats de polices numériques étaient matriciels mais on est depuis passés à des formats vectoriels, qui permettent des changements de taille à volonté. Ces descriptions de caractères peuvent être distribuées via l'Internet et la question se pose alors du type de média à utiliser. En pratique, cela a souvent été `application`, un type un peu fourre-tout. On trouve ainsi enregistré, par exemple, un `application/font-woff`. Le RFC 6838 [1], sur l'enregistrement des types et sous-types de contenus, permet (dans sa section 4.2.7, qui ajoute « "*Such cases are expected to be quite rare*" ») l'enregistrement d'un nouveau type de premier niveau. C'est ainsi que ce RFC 8081 crée `font/`.

Le besoin provient entre autres de l'usage de plus en plus important des "*Web fonts*". C'est ansi qu'HTTP Archive <http://httparchive.org/trends.php?s=All&minlabel=Nov+15+2010&maxlabel=Feb+15+2015#perFonts> a vu passer le pourcentage de sites Web utilisant cette technique de 1 % en 2010 à 50 % en 2015. L'analyse de Kuetell <https://docs.google.com/document/

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc6838.txt`

d/1Tsju6EOP4LqJ1RcFJRpqxryggbWZYbq5jvsIhVYNoyU/edit> montrait une certaine confusion chez les utilisateurs quant au type MIME à utiliser pour ces polices. Certains utilisaient le type de premier niveau font/ avant même son enregistrement officiel et on voyait donc déjà des font/truetype pour le format TrueType. D'autres se servaient d'application/ pour des application/octet-stream (fichier binaire quelconque) ou des application/x-font-ttf (utilisant le préfixe x-, pourtant abandonné par le RFC 6648). On voit même des text/plain pour des ressources pourtant clairement binaires... Les rares types officiellement enregistrés, comme application/font-woff, enregistré par un groupe du W3C <https://www.w3.org/TR/WOFF/>, sont peu utilisés.

Au fait, pourquoi est-ce qu'application/ est une mauvaise idée ? Une des principales raisons est qu'il est regardé avec suspicion par les logiciels de filtrage, qui se méfient de la capacité de ces fichiers à transporter du logiciel malveillant. (Certains formats de police incluent un langage de Turing, et peuvent donc offrir des possibilités insoupçonnées. . .) Ensuite, en l'absence d'un type de premier niveau, il n'était pas possible de définir un jeu commun de paramètres précisant le type. Enfin, les polices de caractères ne sont pas des logiciels et posent des problèmes spécifiques, notamment de licence. Bref, il fallait un type pour les formats de polices.

Ah, et puisque j'ai parlé de sécurité, la section 3 du RFC fait le point sur les problèmes que peuvent poser les polices de ce côté. Un fichier décrivant une police contient des données, mais aussi des programmes ("*hinting instructions*") pour les opérations de rendu les plus sophistiquées. Par exemple, quand on agrandit un caractère, il ne faut pas agrandir uniformément selon toutes les dimensions ; ou bien certaines caractéristiques d'un caractère dépendent des caractères qui l'entourent. Bref, le rendu est une chose trop compliquée pour être spécifié sans un langage de programmation. C'est par exemple ce qu'on trouve dans les polices TrueType (cf. l'article de Wikipédia). Bien sûr l'exécution de ces « programmes » se fait en général dans un bac à sable, et ils n'ont pas accès à l'extérieur, mais certaines attaques restent possibles, par exemple des attaques par déni de service visant à bloquer le moteur de rendu. Les langages utilisés sont en général trop riches pour que des protections simples suffisent.

Et même si on se limite aux données, la plupart des formats (comme SFNT) sont extensibles et permettent l'ajout de nouvelles structures de données. Cette extensibilité est une bonne chose mais elle présente également des risques (par exemple, elle facilite la dissimulation de données dans les fichiers de polices).

Bon, je vous ai assez fait peur avec les risques de sécurité, place à l'enregistrement de font/ à l'IANA (section 4 du RFC). font/ n'indique pas un format particulier, mais une catégorie de contenu. Le format sera indiqué dans le sous-type et c'est là seulement que le logiciel qui reçoit ce contenu saura s'il peut en faire quelque chose d'utile. (Le RFC suggère que les sous-types inconnus devraient être traités comme du binaire quelconque, comme s'ils étaient application/octet-stream.) Six sous-types sont enregistrés par notre RFC.

On peut utiliser un identificateur de fragment (RFC 3986, section 3.5, cet identificateur est le truc après le croisillon dans un URI), pour désigner une police particulière au sein d'une collection présente dans les données envoyées. L'identificateur est le nom PostScript. Attention, certains caractères peuvent être utilisés dans un nom PostScript mais interdits pour un identificateur de fragment, et doivent donc être échappés avec la notation pour-cent. Par exemple, l'identificateur de la police Caret{}stick sera #Caret%5Estick.

Le RFC enregistre plusieurs sous-types. Si on veut en ajouter au registre des polices <https://www.iana.org/assignments/media-types/media-types.xml#font>, il faut suivre les procédures du RFC 6838. Il est recommandé que la spécification du format soit librement accessible (ce qui n'est pas évident dans ce milieu).

Le RFC se termine avec les six sous-types de font/ officiellement enregistrés. D'abord, sfnt pour le format générique SFNT. Il prend des paramètres optionnels, outlines (qui prend comme valeur TTF, CFF ou SVG) et layout (valeurs OTL, AAT et SIL). On pourra donc écrire, par exemple, font/sfnt; layout=SIL. Ce font/sfnt remplace l'ancien type enregistré, application/font-sfnt. Notez que la spécification de ce format est la norme ISO ISO/IEC 14496-22 <http://www.iso.org/iso/iso_catalogue/catalogue_ts/catalogue_detail.htm?csnumber=43466>, dite « *"Open Font Format"* ».

SFNT est un format générique, qui sera sans doute rarement utilisé tel quel. On verra plutôt ttf ou otf.

Un exemple d'un format spécifique est en effet TrueType. Ce sera le sous-type ttf. Il aura également un paramètre optionnel layout (mêmes valeurs possibles). On pourra donc voir dans une réponse HTTP Content-Type: font/ttf.

Troisième sous-type enregistré, otf pour OpenType.

On trouve aussi un sous-type collection pour mettre plusieurs polices ensemble.

Viennent enfin WOFF versions 1 (woff) et 2 (woff2). Il s'agit cette fois d'une norme W3C. Ce nouveau type font/woff remplace l'ancien application/font-woff.

Voilà, c'est tout, le nouveau type de premier niveau font est désormais inclus dans le registre IANA des types de premier niveau <https://www.iana.org/assignments/media-types/media-types.xml>, et les polices enregistrées sont dans cet autre registre IANA <https://www.iana.org/assignments/media-types/media-types.xml#font>.