

RFC 8288 : Web Linking

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 25 octobre 2017

Date de publication du RFC : Octobre 2017

<https://www.bortzmeyer.org/8288.html>

Le **lien** est à la base du Web. Mais c'est seulement récemment, avec le RFC 5988¹ que notre RFC remplace, que certains manques ont été comblés :

— Un registre des **types** de lien,

— Un mécanisme standard pour indiquer un lien dans une réponse HTTP.

Bien sûr, des formats normalisés qui permettent des liens, il y en a plusieurs, et avant tout HTML, avec le fameux élément `<A>` (pour "*anchor*"). Il y a aussi le plus récent Atom (RFC 4287, notamment la section 4.2.7). Comme HTML, Atom avait l'idée de registre des types de liens, mais ces types étaient spécifiques à Atom. L'une des idées phares du RFC 5988 et de celui-ci, son successeur RFC 8288, est de généraliser le concept de type de lien et de le rendre accessible à tous les formats et protocoles qui en ont besoin. Ce RFC décrit un cadre général pour les types de liens, en partant de celui d'Atom.

Second apport de cette norme, une (re-)définition de l'en-tête HTTP `Link:`, utilisant évidemment le nouveau cadre général. Cet en-tête permettant d'indiquer un lien dans la réponse HTTP, indépendamment du document servi, avait été normalisé dans le RFC 2068, section 19.6.2.4, puis, peu utilisé, avait été supprimé par le RFC 2616, avant de faire sa réapparition ici, sous une forme quasi-identique à l'original. On peut voir cet en-tête comme une représentation concrète du cadre de notre RFC. D'autres apparaîtront sans doute.

Pour un exemple réel, regardez les en-têtes `Link:` de mon blog, il y en a un de type `licence`, suivant le RFC 4946. Avec Apache, cela se configure simplement avec le module `headers` et la directive `Header set Link "<http://www.gnu.org/copyleft/fdl.html>; rel=\"license\"; title=\"GFDL\""`.

Donc, qu'est-ce qu'un lien? La section 2, la principale du RFC, le définit comme une connexion **typée** entre deux ressources (une ressource étant typiquement une page Web), nommées respectivement **contexte** et **cible**. Les deux ressources sont représentées par leur IRI (cf. RFC 3987, en notant que, dans la plupart des cas, les IRI des liens seront des URI). Le lien comprend :

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc5988.txt>

- Un IRI qui indique le contexte, c'est-à-dire la ressource de départ,
- Le type de la connexion (détaillé en section 2.1),
- L'IRI cible,
- D'éventuels attributs de la cible (comme par exemple le hreflang de HTML `<https://www.bortzmeyer.org/attribut-lang-liens.html>`). Ce sont des couples clé/valeur.

Par exemple, dans le flux de syndication Atom de mon blog, on trouvera un lien `<atom:link rel="alternate" href="http://www.bortzmeyer.org/expose-go.html"/>` qui se décompose en un contexte (l'entrée Atom dont l'IRI est `tag:bortzmeyer.org,2006-02:Blog/expose-go`), un type (`alternate`, qui indique une version alternative de la ressource, ici une page HTML au lieu d'une entrée Atom), et une cible (ici `<http://www.bortzmeyer.org/expose-go.html>`). Il n'y a pas dans cet exemple d'attributs de la cible mais Atom en permet (par exemple `hreflang` pour indiquer la langue de la cible ou bien `length` pour indiquer sa longueur - afin de prévenir d'un long téléchargement, par exemple).

Cette définition du lien ne place aucune limite sur la cardinalité. Il peut y avoir zéro, un ou plusieurs liens partant d'une ressource et c'est la même chose pour les liens entrants.

La section 2 s'arrête là. Puisque ce RFC propose un cadre général, il ne formalise pas une syntaxe unique pour représenter les liens. Chaque format, chaque protocole, aura la sienne, la **sérialisation**.

Un des points les plus importants de cette définition des liens, et qui est souvent ignorée des gens qui écrivent des pages Web, est la notion de **type** d'un lien (section 2.1). Par exemple, on a un type *"copyright"* qui associe, via un lien, un document à l'auteur de celui-ci. Point à retenir : ce type du lien ne doit pas être confondu avec le type de médium du RFC 6838 comme `text/html` ou `audio/ogg`.

Il y a deux sortes de type de lien : enregistrés ou bien extensions. Les premiers font l'objet de la section 2.1.1. Ils ont fait l'objet d'un processus formel d'enregistrement et leur liste est publiée sous forme d'un registre IANA `<https://www.iana.org/assignments/link-relations/link-relations.xml>`. On y trouve par exemple `via` (RFC 4287) ou `hub` (`<https://github.com/pubsubhubbub>`). La politique d'enregistrement est délibérément ouverte, l'idée étant que si un type est utilisé sur le Web, il faut vraiment qu'il soit indiqué dans le registre, sauf s'il est gravement dangereux pour la santé du Web.

Les extensions sont spécifiées dans la section 2.1.2. L'idée est que, si on n'a pas envie de se fatiguer à enregistrer un type de lien, et qu'on veut quand même créer un type **unique**, n'ayant pas de risque de collision avec le travail des autres, on peut simplement se servir d'un URI (forcément unique) pour indiquer le type. Cette URI peut (mais ce n'est pas obligé) mener à une page Web qui décrira le type en question. Ainsi, on pourrait imaginer de réécrire le lien plus haut en `<atom:link rel="http://www.bortzmeyer.org/reg/my-link-type" href="http://www.bortzmeyer.org/expos">` (en pratique, le format Atom ne permet pas actuellement de telles valeurs pour l'attribut `rel`.)

Après le cadre général de description des liens, notre RFC introduit une syntaxe concrète pour le cas de l'en-tête `Link` : des requêtes HTTP. Les autres formats et protocoles devront s'ajuster à ce cadre chacun de son côté. Pour HTTP, la section 3 décrit l'en-tête `Link` : . La cible doit être un URI (et un éventuel IRI doit donc être transformé en URI), le contexte (l'origine) est la ressource qui avait été demandée en HTTP et le type est indiqué dans le paramètre `rel`. (Le paramètre `rev` qui avait été utilisé dans des vieilles versions est officiellement abandonné.) Plusieurs attributs sont possibles comme `hreflang`, `type` (qui est le type MIME, pas le type du lien) ou `title` (qui peut être noté `title*` s'il utilise les en-têtes étendus du RFC 8187). Pour la plupart de ces attributs, leur valeur est juste une indication, la vraie valeur sera obtenue en accédant à la ressource cible. Ainsi, `hreflang` dans le lien ne remplace pas `Content-Language` : dans la réponse et `type` ne gagnera pas si un `Content-Type` : différent est indiqué dans la réponse.

Voici des exemples d'en-têtes, tirés de la section 3.5 du RFC :

`https://www.bortzmeyer.org/8288.html`

```
Link: <http://www.example.com/MyBook/chapter2> rel="previous";
      title="Previous chapter"
```

Ici, cet en-tête, dans une réponse HTTP, indique que `http://www.example.com/MyBook/chapter2` est une ressource liée à la ressource qu'on vient de récupérer et que ce lien est de type `previous`, donc précède la ressource actuelle dans l'ordre de lecture. L'attribut `title` indique un titre relatif, alors que la vraie ressource `http://www.example.com/MyBook/chapter2` aura probablement un titre du genre « Chapitre 2 ». En application des règles de la section 3.4, c'est ce dernier titre qui gagnera au final.

Un seul en-tête `Link` : peut indiquer plusieurs liens, comme dans l'exemple suivant :

```
Link: </TheBook/chapter2>;
      rel="previous"; title*=UTF-8'de'letztes%20Kapitel,
      </TheBook/chapter4>;
      rel="next"; title*=UTF-8'de'n%c3%a4chstes%20Kapitel
```

Ce dernier montre également les en-têtes complètement internationalisés du RFC 8187, ici en allemand (étiquette de langue `<http://www.langtag.net/> de`).

Cet en-tête a été enregistré à l'IANA, en application du RFC 3864 dans le registre des en-têtes `<https://www.iana.org/assignments/message-headers/perm-headers.html>` (section 4.1).

D'autre part, un registre des types de liens `<https://www.iana.org/assignments/link-relations/link-relations.xml>` existe. La section 4.2 décrit en détail ce registre. Voici, à titre d'exemple, quelques-uns des valeurs qu'on peut y trouver :

- `blocked-by` indique l'entité qui a exigé le blocage d'une page Web (avec le fameux code 451, voir RFC 7725),
- `copyright` qui indique le "*copyright*" du document (issu de la norme HTML `<http://www.w3.org/TR/html401/types.html#type-links>`),
- `edit` qui indique l'URI à utiliser pour une modification de ce document, comme le permet le protocole APP (RFC 5023),
- `first`, qui pointe vers le premier document de la série (défini par ce RFC 8288, même s'il était déjà enregistré),
- `hub` qui indique l'endroit où s'abonner pour des notifications ultérieures, suivant le protocole PubSubHubbub),
- `latest-version` qui indique où trouver la dernière version d'un document versionné (RFC 5829),
- `licence`, qui associe un document à sa licence d'utilisation (RFC 4946),
- `nofollow`, qui indique qu'on ne recommande pas la ressource vers laquelle on pointe `<https://www.w3.org/TR/html5/links.html#link-type-nofollow>`, et qu'il ne faut donc pas considérer ce lien comme une approbation,
- `related`, qui indique un document qui a un rapport avec celui-ci (créé pour Atom, dans le RFC 4287),
- `replies`, qui indique les réponses faites à ce document (pour mettre en œuvre le "*threading*", RFC 4685),
- Et bien d'autres encore...

<https://www.bortzmeyer.org/8288.html>

Ce registre est peuplé par le mécanisme dit Spécification Nécessaire (cf. RFC 8126), avec exigence d'un examen par un expert (l'actuel expert est Mark Nottingham, auteur de plusieurs RFC, dont celui-ci). Pour chaque type, il faudra indiquer le type (aussi nommé relation, comme par exemple `previous` plus haut), une description et une référence à la norme qui le formalise. Les demandes d'enregistrement sont reçues par `link-relations@ietf.org`.

Attention, ce n'est pas parce qu'il y a un lien qu'il faut le suivre automatiquement. La section 5, sur la sécurité, met en garde contre la confiance accordée à un lien.

L'annexe A.1 contient une discussion de l'utilisation des liens avec HTML 4, d'où le cadre actuel de définition des liens est issu. Le type `y` est indiqué par l'attribut `rel`. Un exemple indiquant la licence en XHTML est donc :

```
<link rel="license" type="text/html" title="GFDL in HTML format"
      href="http://www.gnu.org/copyleft/fdl.html"/>
```

L'annexe A.2 discute, elle, de l'utilisation du cadre de définition des liens en Atom, qui utilise l'élément `<atom:link>` avec les attributs `href` pour indiquer la cible et `rel` pour le type. Par exemple, `<atom:link rel="license" type="text/html" title="GFDL in HTML format" href="http://www.gnu.org/copyleft/fdl.html"/>` indiquera la licence du flux Atom qui contient cet élément (et, oui, Atom et XHTML ont quasiment la même syntaxe).

L'annexe C de notre RFC indique les changements depuis son prédécesseur, le RFC 5988. Rien de très crucial :

- Les enregistrements faits par le RFC 5988 sont faits, donc on ne les répète pas dans ce RFC,
- Un registre des données spécifiques aux applications, qui n'a apparemment pas été un grand succès (il me semble qu'il n'a jamais été créé), n'est plus mentionné,
- Quatre errata `<https://www.rfc-editor.org/errata_search.php?rfc=5988&rec_status=15&presentation=table>` ont été pris en compte, dont un était technique et portait sur la grammaire,
- La nouvelle annexe B donne un algorithme pour l'analyse des en-têtes,
- Et enfin ce nouveau RFC apporte pas mal de clarifications et de détails.

Ce nouveau RFC a fait l'objet de peu de discussions à l'IETF, mais il y en a eu beaucoup plus sur GitHub `<https://github.com/mnot/I-D/issues?q=is:issuelabel:rfc5988bis>`.

Notez qu'à Paris Web 2017, le t-shirt officiel portait une allusion à ce RFC, avec le texte `<link rel="human" ... >`, encourageant les relations entre humains.