# RFC 6497 : BCP 47 Extension T - Transformed Content

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 23 février 2012

Date de publication du RFC : Février 2012

https://www.bortzmeyer.org/6497.html

---

Le RFC 5646 [1], qui normalise les étiquettes de langue <http://www.langtag.net/>, prévoit un mécanisme d'extension pour ajouter à ces étiquettes des informations maintenues dans des registres extérieurs. Ces extensions sont identifiées par une lettre unique et une nouvelle extension est ajoutée par notre RFC 6497, « T » (pour "*transformation*"), qui permet d'indiquer les transformations qu'a subies un texte (par exemple une traduction), en se référant au registre CLDR.

Le RFC 5646 est également connu sous l'identificateur « BCP 47 » (BCP pour "*Best Common Practice*"), ce qui explique le titre de notre RFC 6497. Les transformations qu'on pourra noter dans une étiquette de langue sont notamment la translittération, la transcription et la traduction. Sans cette extension « T », on ne peut étiqueter que l'état actuel du document. Par exemple, `pes-Latn` désignerait un texte en persan dans l'alphabet latin (ce qui est inhabituel) ; l'extension « T » permettra d'écrire `pes-Latn-t-pes-arab`, indiquant que le texte persan a été translitteré ou transcrit depuis l'alphabet arabe. Autre exemple, donné dans le RFC, une carte d'Italie pour des japonais où les noms des villes ont été translittérés en Katakana. N'indiquer que `ja` (langue japonaise) pour ce texte serait insuffisant. Cette information sur la source, donnée par l'extension « T » est souvent importante, pour aider à comprendre certaines particularités du texte (comme des fautes lors de la traduction). Le mécanisme des variantes du RFC 5646 ne suffisait pas pour cela, d'où la nouvelle extension. C'est la deuxième enregistrée, après la « U » du RFC 6067. Le mécanisme d'extension est normalisé dans le RFC 5646, sections 2.2.6 et 3.7. Les extensions sont enregistrées dans le registre IANA <https://www.iana.org/assignments/language-tag-extensions-registry>.

Parfois, il existe des méthodes standard de translittération ou de transcription. C'est par exemple le cas de celles spécifiées par l'UNGEGN. La nouvelle extension « T » permet également d'indiquer comment et par quelle méthode standard un texte a été translitteré ou transcrit.

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc5646.txt`

Maintenant, les détails techniques : l'extension est notée par un `t` suivi d'une étiquette de langue légale (section 2, notamment 2.2). Ainsi, `es-t-pid` signifie que le texte est en espagnol, traduit du piaroa. Mais, comme le texte après `t` peut être une étiquette de langue complète, on peut noter bien plus de détails, comme l'écriture utilisée (comme dans l'exemple du perse plus haut). Comme l'étiquette de langue doit comporter obligatoirement une langue en première position, si celle-ci n'est pas connue ou tout simplement pas pertinente (rappelez-vous que la différence principale entre translittération et transcription est que la première ne dépend **pas** de la langue, c'est une opération purement mécanique sur les caractères), on la note par `und`, ce qui veut dire « inconnue ». Ainsi, `und-Latn-t-und-cyrl` signifie que le texte a été translitteré de l'alphabet cyrillique vers le latin (par exemple par l'algorithme ISO 9), sans tenir compte des langues en jeu.

Enfin, il est possible d'indiquer le mécanisme de transformation, en utilisant le séparateur `m0`, puis l'organisme qui spécifie ces mécanismes, puis un identificateur d'un mécanisme particulier. Ainsi, `und-Cyrl-t-und-` est du texte en alphabet latin, transformé en cyrillique, suivant la spécification UNGEGN de 2007. D'autres séparateurs que `m0` seront peut-être créés dans le futur.

Quels termes peut-on mettre après `m0` ? Ce qui vous passe par la tête ? Non. Il faut suivre la section 3 du document Unicode UTR #35 `<http://www.unicode.org/reports/tr35/>`. C'est d'ailleurs le consortium Unicode qui est l'autorité gérant l'extension « T » (section 2.4). Cela a suscité de vigoureuses discussions (pourquoi pas l'IETF ?). La conséquence en a été la définition d'un mécanisme plus ouvert pour soumettre des propositions de changement. Si on veut enregistrer de nouvelles possibilités (section 2.6), cela se fait en créant un ticket en `<http://cldr.unicode.org/index/bug-reports/>`.

La liste complète des possibilités de mécanismes de transformation figure dans un fichier structuré (section 2.9), librement accessible en `<http://cldr.unicode.org/index/downloads>`. Ce point a suscité beaucoup de débats sur la liste de diffusion de l'ancien groupe de travail LTRU `<https://www.bortzmeyer.org/fin-ltru.html>`. En effet, le fichier officiel est un gros `.zip` rempli de tas de choses qui n'ont rien à voir. Il faut donc tout télécharger puis trier (les transformations sont dans le répertoire `common/transforms`).