

Groupe de travail Réseau  
**Request for Comments : 2141**  
 Catégorie : En cours de normalisation

R. Moats, AT&T  
 mai 1997  
 Traduction Claude Brière de L'Isle

## Syntaxe d'URN

### Statut du présent Mémo

La présente RFC spécifie un protocole de normalisation pour la communauté Internet et appelle à des discussions et suggestions pour son amélioration. Prière de se reporter à l'édition en cours des "Internet Official Protocol Standards" (*normes officielles du protocole Internet*) (STD 1) pour connaître l'état de la normalisation et le statut du présent protocole. La distribution du présent mémo n'est soumise à aucune restriction.

### Résumé

Les noms de ressource uniformes (URN, *Uniform Resource Name*) sont destinés à servir d'identifiants de ressource persistants, indépendants de la localisation. Le présent document établit la syntaxe canonique pour les URN. On présente à la fois les espaces de dénomination existants traditionnels et nouveaux ainsi que les exigences pour la présentation et la transmission des URN. Il y a enfin une présentation de l'équivalence des URN et la façon de la déterminer.

## 1. Introduction

Les noms de ressource uniformes (URN, *Uniform Resource Name*) sont destinés à servir d'identifiants de ressource persistants, indépendants de la localisation et sont conçus pour faciliter la transposition des autres espaces de noms (qui partagent les propriétés des URN) dans l'espace des URN. Donc, la syntaxe d'URN donne le moyen de coder les données de caractères sous une forme qui peut être envoyée dans les protocoles existants, transcrits sur la plupart des claviers, etc.

## 2. Syntaxe

Tous les URN ont la syntaxe suivante (les phrases entre guillemets sont EXIGÉES) :

```
<URN> ::= "urn:" <NID> ":" <NSS>
```

où <NID> est l'identifiant d'espace de nom, et <NSS> est la chaîne spécifique d'espace de nom. La séquence "urn:" en tête est insensible à la casse. L'identifiant d'espace de nom détermine l'interprétation syntaxique de la chaîne spécifique d'espace de nom (comme exposé dans [1]).

La RFC1630 [2] et la RFC1737 [3] présentent chacune des considérations supplémentaires pour le codage des URN, qui ont des implications sur la limitation de la syntaxe. Par ailleurs, l'exigence de prise en charge des systèmes existants traditionnels de dénomination ont pour effet d'élargir la syntaxe. On expose donc séparément la syntaxe acceptable à la fois pour l'identifiant d'espace de nom et pour la chaîne spécifique d'espace de nom.

### 2.1 Syntaxe d'identifiant d'espace de nom

Ce qui suit est la syntaxe pour l'identifiant d'espace de nom. Pour (a) être cohérent avec tous les schémas potentiels de résolution, et (b) ne pas créer de contraintes injustifiées sur un schéma de résolution potentiel, la syntaxe pour l'identifiant d'espace de nom est :

```
<NID> ::= <let-num> [ 1,31<let-num-hyp> ]
```

```
<let-num-hyp> ::= <upper> | <lower> | <number> | "-"
```

```
<let-num> ::= <upper> | <lower> | <number>
```

```
<upper> ::= "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" |
           "U" | "V" | "W" | "X" | "Y" | "Z"
```

```
<lower> ::= "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" |
           "x" | "y" | "z"
```

<number> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

C'est un peu plus restrictif que ce qui était dit dans [4] (qui permet les caractères "." et "+"). De plus, l'identifiant d'espace de nom est insensible à la casse, de sorte que "ISBN" et "isbn" se réfèrent au même espace de nom.

Pour éviter la confusion avec l'identifiant "urn:", le NID "urn" est réservé et NE DOIT PAS être utilisé.

## 2.2 Syntaxe de chaîne spécifique d'espace de nom

Comme l'exige la RFC1737, il y a une seule représentation canonique de la portion NSS d'un URN. Le format de cette seule forme canonique est le suivant :

<NSS> ::= 1\*<URN chars>

<URN chars> ::= <trans> | "%" <hex> <hex>

<trans> ::= <upper> | <lower> | <number> | <other> | <reserved>

<hex> ::= <number> | "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" | "d" | "e" | "f"

<other> ::= "(" | ")" | "+" | "," | "-" | "." | ":" | "=" | "@" | ";" | "\$" | "\_" | "!" | "\*" | ""

Selon les règles qui gouvernent un espace de nom, les identifiants valides dans un espace de nom peuvent contenir des caractères qui ne font pas partie du jeu de caractères d'URN ci-dessus (<URN chars>). De telles chaînes DOIVENT être traduites en format NSS canonique avant de les utiliser comme éléments de protocole ou de les passer à d'autres applications. La traduction est faite en codant chaque caractère en dehors du jeu de caractères d'URN comme une séquence de un à six octets en utilisant le codage UTF-8 [5], et le codage de chacun de ces octets avec "%" suivi de deux caractères tirés du jeu <hex> ci-dessus. Les deux caractères donnent la représentation hexadécimale de cet octet.

## 2.3 Caractères réservés

Les caractères qui restent à discuter sont le jeu des caractères réservés, qui contient divers caractères réservés en utilisation normale. Le jeu des caractères réservés est donné ci-après, avec une discussion sur les spécificités qui font que chacun est réservé.

Le jeu des caractères réservés est :

<reserved> ::= "%" | "/" | "?" | "#"

### 2.3.1 Caractère "%"

Le caractère "%" est réservé dans la syntaxe d'URN pour introduire la séquence d'échappement pour un octet. L'utilisation littérale du caractère "%" dans un espace de nom doit être codée en utilisant "%25" dans les URN pour cet espace de nom. La présence d'un caractère "%" dans un URN DOIT être suivi par deux caractères du jeu de caractères <hex>.

Les espaces de nom PEUVENT désigner un ou plusieurs caractères provenant du jeu de caractères d'URN comme ayant une signification particulière pour cet espace de nom. Si l'espace de nom utilise aussi ce caractère au sens littéral, le caractère utilisé au sens littéral DOIT être codé avec "%" suivi de la représentation hexadécimale de cet octet. De plus, un caractère NE DOIT PAS être codé avec "%" si le caractère n'est pas un caractère réservé. Donc, le processus d'enregistrement d'un identifiant d'espace de nom devra inclure la publication de la définition des caractères qui ont une signification particulière pour cet espace de nom.

### 2.3.2 Autres caractères réservés

La RFC1630 [2] réserve les caractères "/", "?", et "#" pour des usages particuliers. Le groupe de travail URN n'a pas encore débattu de l'applicabilité et de la sémantique précise de ces objets lorsque appliquée aux URN. Donc, ces caractères sont RESERVÉS pour des développements futurs. Les développeurs d'espace de nom NE DEVRAIENT PAS utiliser ces caractères en forme non codée, mais plutôt utiliser le codage en % approprié pour chaque caractère.

## 2.4 Caractères exclus

La liste suivante n'est donnée que dans le souci d'être complet. Tout octet/caractère de cette liste est explicitement EXCLU du jeu de caractères d'URN, et si il est utilisé dans un URN, il DOIT être codé en % :

<excluded> ::= octets 1-32 (1-20 hex) | "\" | "" | "&" | "<" | ">" | "[" | "]" | "^" | "`" | "{" | "|" | "}" | "~" | octets 127-255 (7F-FF hex)

De plus, l'octet 0 (0 hex) NE DEVRAIT JAMAIS être utilisé, ni non codé, ni codé en forme %.

Un URN se termine quand un octet/caractère tiré du jeu de caractères exclus (<excluded>) est rencontré. Le caractère tiré du jeu de caractères exclus NE FAIT PAS partie de l'URN.

## 3. Prise en charge des systèmes de dénomination existants et nouveaux systèmes

Tout espace de nom (existant ou nouvellement conçu) qui est proposé comme espace de nom d'URN et qui satisfait aux critères des espaces de nom d'URN DOIT être exprimé dans cette syntaxe. Si les noms dans ces espaces de nom contiennent des caractères autres que ceux définis pour le jeu de caractères d'URN, ils DOIVENT être traduits en forme canonique telle qu'exposée au paragraphe 2.2.

## 4. Présentation et transport des URN

La syntaxe d'URN définit le format canonique des URN et tout transport et échange d'URN DOIT avoir lieu dans ce format. De plus, toute application en relation avec les URN DOIT offrir l'option d'afficher les URN en forme canonique pour permettre une transcription directe (par exemple par des techniques de copier coller). De telles applications PEUVENT prendre en charge l'affichage des URN sous une forme plus lisible par l'homme et peuvent utiliser un jeu de caractères qui comporte des caractères qui ne sont pas permis dans la syntaxe d'URN telle que définie dans la présente RFC (c'est-à-dire qu'elles peuvent remplacer la notation % par des caractères tirés d'un jeu de caractères étendu pour l'affichage à l'homme).

## 5. Équivalence lexicale dans les URN

Pour diverses raisons comme la mise en mémoire, il est souvent souhaitable de déterminer si deux URN sont les mêmes sans avoir à les résoudre. Le moyen général pour le faire est d'essayer de trouver une "équivalence lexicale" telle que définie ci-dessous.

Deux URN sont lexicalement équivalents si ils sont égaux octet par octet après le pré-traitement suivant :

1. normaliser la casse du jeton "urn:" en tête
2. normaliser la casse du NID
3. normaliser la casse de tout échappement en %.

Noter que l'échappement en % NE DOIT PAS être retiré.

Certains espaces de nom peuvent définir des équivalences lexicales supplémentaires, telles que la non sensibilité à la casse de la NSS (ou de parties d'elle). Les équivalences lexicales supplémentaires DOIVENT être documentées au titre de l'enregistrement de l'espace de nom, lexical DOIVENT toujours avoir pour effet d'éliminer les doubles négations obtenues par la procédure ci-dessus, et NE DOIVENT JAMAIS dire que deux URN ne sont pas équivalents si la procédure ci-dessus dit qu'ils sont équivalents.

## 6. Exemples d'équivalence lexicale

Les comparaisons d'URN ci-dessous mettent en lumière les définitions d'équivalence lexicale :

- 1- URN:foo:a123,456
- 2- urn:foo:a123,456
- 3- urn:FOO:a123,456
- 4- urn:foo:A123,456
- 5- urn:foo:a123%2C456
- 6- URN:FOO:a123%2c456

Les URN 1, 2, et 3 sont tous lexicalement équivalents. L'URN 4 n'est lexicalement équivalent à aucun des autres URN de l'ensemble ci-dessus. Les URN 5 et 6 ne sont lexicalement équivalents que l'un avec l'autre.

## 7. Équivalence fonctionnelle dans les URN

L'équivalence fonctionnelle est déterminée par la pratique au sein d'un espace de nom donné et gérée par les résolveurs pour cet espace de nom. Thus, it is beyond the scope of this document. Namespace registration must include guidance on how to determine functional equivalence for that namespace, i.e. when two URNs are the identical within a namespace.

## 8. Considérations pour la sécurité

Le présent document spécifie la syntaxe des URN. Bien que certains résolveurs d'espace puissent allouer une signification particulière à certains des caractères de la chaîne spécifique de l'espace de dénomination, toute considération de sécurité résultant d'une telle allocation est en dehors du domaine d'application du présent document. Il est fortement recommandé que le processus d'enregistrement d'identifiant d'espace de dénomination inclue de telles considérations.

## 9. Remerciements

Merci au divers membres du groupe de travail URN pour leurs commentaires sur les premières versions de ce document. Ce document est partiellement pris en charge par la National Science Foundation, accord de coopération NCR-9218179.

## 10. Références

Les documents d'Appel à commentaires (RFC) et les projets Internet sont disponibles sur <URL:ftp://ftp.internic.net> et de nombreux sites miroir.

- [1] K. Sollins, "Principes d'architecture de la résolution de nom de ressource uniforme", RFC2276, janvier 1998. (*MàJ par RFC3401*) (*Information*)
- [2] T. Berners-Lee, "Identifiants de ressource universels dans la Toile mondiale ; syntaxe unificatrice pour l'expression des noms et adresses des objets du réseau utilisés sur la Toile mondiale", RFC1630, juin 1994. (*Information*)
- [3] K. Sollins et L. Masinter, "Exigences fonctionnelles pour les noms de ressource uniformes", RFC1737, décembre 1994.
- [4] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiants de ressource uniformes (URI) : Syntaxe générique", RFC2396, août 1998. (*Obsolète, voir RFC3986*)
- [5] Appendice A.2 du Consortium Unicode, "The Unicode Standard, Version 2.0", Addison-Wesley Developers Press, 1996. ISBN 0-201-48345-9.

## 11. Adresse de l'éditeur

Ryan Moats  
AT&T  
15621 Drexel Circle  
Omaha, NE 68135-2358  
USA

téléphone : +1 402 894-9456  
mél : jayhawk@ds.internic.net

## **Appendice A Traitement des URN par les résolveurs/navigateurs URL**

La syntaxe des URN a été définie de telle sorte que les URN puissent être utilisés dans des endroits où on attend des URL. Un résolveur qui se conforme à la spécification actuelle de syntaxe des URL [3] va extraire une valeur de schéma de "urn:" plutôt qu'une valeur de schéma de "urn:<nid>".

Un URN DOIT être considéré comme un URL opaque par les résolveurs d'URL et passé (avec l'étiquette "urn:") à un résolveur d'URN pour résolution. Le résolveur d'URN peut être un résolveur externe que le résolveur d'URL connaît, ou il peut être fonctionnellement incorporé dans le résolveur d'URL.

Pour éviter d'embrouiller les utilisateurs, un navigateur d'URL DEVRAIT afficher l'URN complet (y compris l'étiquette "urn:") pour s'assurer qu'il n'y a pas de confusion entre les identifiants d'espace de nom d'URN et les identifiants de schéma d'URL.