

# RFC 9039 : Uniform Resource Names for Device Identifiers

Stéphane Bortzmeyer  
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 1 juillet 2021

Date de publication du RFC : Juin 2021

<https://www.bortzmeyer.org/9039.html>

---

Voici un nouvel espace de noms pour des URN, *dev*, afin de stocker des identificateurs pour des équipements matériels, par exemple pour les inventaires.

Les URN, un sous-ensemble des URI, et des cousins des URL, sont normalisés dans le RFC 8141<sup>1</sup>. On peut placer des URN partout où on peut mettre des URI, par exemple comme noms dans SenML (RFC 8428). Pour des objets contraints, les URN risquent d'être un peu longs par rapport à, par exemple, une adresse IPv4, mais ils sont plus souples. Un URN commence par le plan `urn:`, suivie d'un espace de noms. Ces espaces sont stockés dans un registre IANA <<https://www.iana.org/assignments/urn-namespaces/urn-namespaces.xml>>. Ce nouveau RFC crée un nouvel espace de noms, *dev*. On pourra donc désormais avoir des URN comme `urn:dev:os:32473-123456` (qui identifie la machine 123456 de l'organisation 32473). Ces identificateurs de machines pourront être utilisés toutes les fois où on a besoin de désigner une machine, dans les données du RFC 8428, dans des inventaires, etc.

Passons au concret maintenant, avec la section 3 du RFC, qui donne la définition formelle de l'espace de noms *dev*. Ces URN sont évidemment conformes à la norme des URN, le RFC 8141. Comme tous les URN, ceux sous `urn:dev:` ne sont pas prévus pour être résolus automatiquement. Contrairement aux URL, ils ne fournissent pas de moyen d'accès à une ressource. Bien sûr, si on les met dans une base de données, d'inventaire de ses machines, par exemple, on pourra retrouver l'information, mais ce RFC ne spécifie aucun mécanisme de résolution standard (section 3.6 du RFC).

Après le `urn:dev:`, ces URN prennent un composant supplémentaire, qui identifie le type d'identificateur dont dérive l'URN. Cela peut être :

---

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc8141.txt>

- `mac`, où l'identificateur est une adresse MAC, enregistrée selon les procédures IEEE, au format EUI-64. C'est par exemple `urn:dev:mac:acde48234567019f` (pour l'adresse MAC `ac:de:48:23:45:67`). Si vous faites varier l'adresse MAC, par exemple pour protéger votre vie privée, l'URN n'est plus un identifiant stable.
- `ow`, qui s'appuie sur 1-Wire, un système privé.
- `org`, avec des identificateurs spécifiques à une organisation, identifiée par son PEN (les PEN sont normalisés dans le RFC 2578). Les PEN sont enregistrés à l'IANA <<https://www.iana.org/assignments/enterprise-numbers/>>. Prenons comme exemple celui d'une entreprise dont j'étais un co-fondateur, 9319. Si cette entreprise utilise des noms pour ses machines, la machine `marx` aurait comme URN `urn:dev:org:9319-marx`.
- `os`, comme les `org` mais plutôt prévus pour des numéros de série. Si cette entreprise numérote toutes ses machines en partant de 1, un URN serait `urn:dev:os:9319-1`. (Notez que des versions précédentes des URN `dev` utilisaient `os` pour les identificateurs LwM2M.) Ainsi, le RIPE-NCC qui a le PEN 15854 pourrait nommer ses sondes Atlas <<https://atlas.ripe.net/>> d'après leur numéro unique et donc la sonde 6593 <<https://atlas.ripe.net/probes/6593/>> pourrait être désignée par `urn:dev:os:15854-6593`. Notez qu'on pourrait avoir ici une intéressante discussion sur l'intérêt respectif des URN (`urn:dev:os:15854-6593`) et des URL (<https://atlas.ripe.net/probes/6593/>).
- `ops`, qui ressemble à l'`os`, mais avec un identificateur de produit entre le PEN et le numéro de série, par exemple `urn:dev:ops:9319-coffeemachine-2` pour la deuxième machine à café de l'organisation. (Comme `os`, il avait été utilisé pour l'Open Mobile Alliance.)
- Un autre type, qui sera défini dans le futur (cf. section 7 du RFC). En attendant, on peut utiliser exemple pour des exemples, comme `urn:dev:example:1234`.

Enfin, une machine identifiée par un de ces URN peut avoir une partie particulière de la machine désignée par une chaîne de caractères après un tiret bas. Ainsi, `urn:dev:os:9319-1_alimentation` serait l'alimentation de la machine `urn:dev:os:9319-1`.

Notez que l'équivalence de deux URN est sensible à la casse donc attention, par exemple, à la façon dont vous écrivez les adresses MAC. Le RFC recommande de tout mettre en minuscules.

Idéalement, on veut bien sûr qu'un URN `dev` identifie une machine et une seule. Mais, en pratique, cela peut dépendre du type d'identificateurs utilisé. Ainsi, les adresses MAC ne sont pas forcément uniques, entre autres parce que certains fabricants ont déjà réutilisé des adresses.

Petit avertissement sur la vie privée : les identificateurs décrits dans ce RFC sont prévus pour être très stables sur le long terme (évidemment, puisque leur but est de garder trace d'une machine) et leur utilisation imprudente (par exemple si on envoie un de ces URN avec les données d'un utilisateur anonyme) peut permettre une surveillance accrue (sections 3.4 et 6.1 du RFC). Le RFC 7721 détaille les risques de ces identificateurs à longue durée de vie.

Le RFC note (section 1) qu'il existe d'autres catégories d'identificateurs qui, selon le cas, pourraient concurrencer nos URN de l'espace de noms `dev`. C'est le cas par exemple des condensats du RFC 6920, des IMEI du RFC 7254, des MEID du RFC 8464 et bien sûr des UUID du RFC 9562. Tous peuvent se représenter sous forme d'URI, et parfois d'URN. Ils ont leurs avantages et leurs inconvénients, le choix est vaste.

Pour les gens qui utilisent le SGBD PostgreSQL, notez qu'il n'a pas de type de données « URI » donc, si on veut stocker les URN de notre RFC dans PostgreSQL, il faut utiliser le type `TEXT`, ou bien installer une extension comme `pguri` <<https://github.com/petere/pguri>>. Selon ce qu'on veut faire de ces URN, on peut aussi prendre une solution plus simple qui ne nécessite pas d'installer d'extension, ici pour une organisation qui met toutes ces machines en `urn:dev:os:9319-...` :

```
CREATE DOMAIN urndev AS text CHECK (VALUE ~ '^urn:dev:os:9319-[0-9]+(_[a-z0-9]+)?$');
COMMENT ON DOMAIN urndev IS 'URN DEV (RFC 9039) for our devices';
CREATE TABLE Devices (id SERIAL, urn urndev UNIQUE NOT NULL, comments TEXT);
INSERT INTO Devices (urn, comments) VALUES ('urn:dev:os:9319-2', 'No comment');
INSERT INTO Devices (urn) VALUES ('urn:dev:os:9319-1_alimentation');
INSERT INTO Devices (urn, comments) VALUES ('urn:dev:os:9319-666', 'Beast');
```