

Groupe de travail Réseau
Request for Comments : 8089
RFC mise à jour : 1738
 Catégorie : Sur la voie de la normalisation
 ISSN: 2070-1721

M. Kerwin, QUT

février 2017

Traduction Claude Brière de L'Isle

Schéma d'URI "file"

Résumé

Le présent document donne une spécification plus complète du schéma d'identifiant de ressource universel (URI, *Uniform Resource Identifier*) "file" et remplace la très brève définition du paragraphe 3.10 de la RFC 1738. Il définit une syntaxe commune destinée à interopérer à travers le large spectre des usages existants. En même temps, il note quelques autres pratiques courantes autour de l'utilisation des URI "file".

Statut de ce mémoire

Ceci est un document de l'Internet en cours de normalisation.

Le présent document a été produit par l'équipe d'ingénierie de l'Internet (IETF). Il représente le consensus de la communauté de l'IETF. Il a subi une révision publique et sa publication a été approuvée par le groupe de pilotage de l'ingénierie de l'Internet (IESG). Plus d'informations sur les normes de l'Internet sont disponibles à la Section 2 de la RFC7841.

Les informations sur le statut actuel du présent document, tout errata, et comment fournir des réactions sur lui peuvent être obtenues à <http://www.rfc-editor.org/info/rfc8089>

Notice de droits de reproduction

Copyright (c) 2017 IETF Trust et les personnes identifiées comme auteurs du document. Tous droits réservés.

Le présent document est soumis au BCP 78 et aux dispositions légales de l'IETF Trust qui se rapportent aux documents de l'IETF (<http://trustee.ietf.org/license-info>) en vigueur à la date de publication de ce document. Prière de revoir ces documents avec attention, car ils décrivent vos droits et obligations par rapport au présent document. Les composants de code extraits du présent document doivent inclure le texte de licence simplifié de BSD comme décrit au paragraphe 4.e des dispositions légales du Trust et sont fournis sans garantie comme décrit dans la licence de BSD simplifiée.

Table des matières

1. Introduction.....	2
1.1 Conventions de notation.....	2
2. Syntaxe.....	2
3. Opérations qui impliquent des URI <file>.....	3
4. Codage du nom de système de fichiers.....	3
5. Considérations sur la sécurité.....	4
6. Considérations relatives à l'IANA.....	4
7. Références.....	4
7.1 Références normatives.....	4
7.2 Références pour information.....	5
Appendice A. Différences avec les spécifications précédentes.....	5
Appendice B. Exemples d'URI.....	6
Appendice C. Technologies similaires.....	6
Appendice D. Opérations spécifiques du système.....	6
D.1 Systèmes POSIX.....	6
D.2 Systèmes de style DOS et Windows.....	7
D.3 Systèmes Mac OS X.....	7
D.4 Systèmes Files-11 OpenVMS.....	7
Appendice E. Variantes de syntaxe non standard.....	7
E.1 Informations d'utilisateur.....	7
E.2 Lettres de pilote DOS et Windows.....	7
E.3 Chaînes UNC.....	8
E.4 Barre oblique inverse comme séparateur.....	9

Appendice F. Règles non standart collectées.....	9
Remerciements.....	10
Adresse de l'auteur.....	10

1. Introduction

Un URI "file" identifie un objet (un "fichier") mémorisé dans un environnement structuré de dénomination d'objet et d'accès sur un hôte (un "système de fichiers"). L'URI peut être utilisé dans des discussions au sujet du fichier, et si d'autres conditions sont satisfaites, il peut être déréférencé pour accéder directement au fichier.

Le présent document spécifie une syntaxe fondée sur la syntaxe générique de la [RFC3986] qui est compatible avec la plupart des usages existants. Lorsque des incompatibilités surviennent, elles sont généralement dans des parties du schéma qui étaient sous spécifiées dans les définitions antérieures et avaient été resserrées par des spécifications plus récentes. L'Appendice A fait la liste des changements significatifs à la syntaxe.

Les extensions à la syntaxe qui pourraient être rencontrées dans la pratique sont recensées dans l'Appendice E ; ces extensions sont mentionnées à des fins d'information et ne sont pas une exigence de mise en œuvre.

Le schéma d'URI "file" n'est pas couplé avec un protocole ni un type de support spécifiques [RFC6838]. Voir à la Section 3 une discussion des opérations qui peuvent être effectuées sur l'objet identifié par un URI "file".

1.1 Conventions de notation

Les mots clés "DOIT", "NE DOIT PAS", "EXIGÉ", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" dans le présent document sont à interpréter comme décrit dans la [RFC2119] quand ils apparaissent tout en majuscules. Ils peuvent aussi apparaître en minuscules ou en casse mixte dans des mots français sans signification normative.

Tout au long du présent document, le terme "fichier local" est utilisé pour décrire les fichiers auxquels on peut accéder par l'API de système de fichier local en utilisant seulement les informations incluses dans le chemin de fichiers, et non en s'appuyant sur d'autres informations (comme les adresses réseau). Il est important de noter qu'un fichier local peut n'être pas situé physiquement sur la machine locale, par exemple, si un système de fichier mis en réseau est monté de façon transparente dans le système de fichier local.

Le terme "URI de fichier local" est utilisé pour décrire les URI "file" qui n'ont pas de composant "autorité" ou où l'autorité est la chaîne spéciale "hôte" ou un nom de domaine pleinement qualifié qui se résout en la machine de laquelle l'URI est interprété (Section 2).

2. Syntaxe

La syntaxe d'URI "file" est définie ici en forme Backus-Naur augmentée (ABNF) [RFC5234], important les règles "hôte" et "chemin-absolu" de la [RFC3986] (telle que mise à jour par la [RFC6874]).

La syntaxe générique de la [RFC3986] inclut les composants "chemin" et "autorité", pour chacun desquels seul un sous ensemble est utilisé dans la définition du schéma d'URI "file". Le sous ensemble pertinent de "chemin" est "chemin-absolu", et le sous ensemble de "autorité" est "autorité-fichier", donnée ci-dessous.

La définition de la syntaxe ci-dessous est différente de celle donnée dans les [RFC1630] et [RFC1738] car elle est déduite de la syntaxe générique de la [RFC3986], qui périmé des spécifications antérieures de l'URI "file". L'Appendice A énumère les différences significatives.

file-URI = file-scheme ":" file-hier-part

file-scheme = "file"

file-hier-part = ("/" auth-path) / local-path

auth-path = [file-auth] path-absolute

local-path = path-absolute

file-auth = "localhost" / host

Le "host" est le nom de domaine pleinement qualifié du système sur lequel le fichier est accessible. Cela permet à un client sur un autre système de savoir qu'il ne peut pas accéder au système de fichiers, ou peut-être qu'il a besoin d'utiliser quelque autre mécanisme local pour accéder au fichier.

Cas particulier, la règle "file-auth" peut correspondre à la chaîne "localhost" qui est interprétée comme "la machine à partir de laquelle l'URI est interprété", exactement comme si aucune autorité n'était présente. Certaines utilisations courantes du schéma interprètent de façon incorrecte toutes les valeurs dans l'autorité d'un fichier URI, incluant "localhost", comme non local. D'autres encore interprètent toute valeur comme locale, même si le "host" ne se résout pas en la machine locale. Pour maximiser la compatibilité avec les spécifications antérieures, les utilisateurs PEUVENT choisir d'inclure un "auth-path" sans "file-auth" lors de la création d'un URI.

Le composant "path" représente le chemin absolu vers le fichier dans le système de fichiers. Voir à l'Appendice D une discussion des problèmes spécifiques de système incluant les chemins de fichier absolus et les racines de système de fichier.

Certains systèmes de fichiers ont des noms de fichiers sensibles à la casse et certains autres pas. À ce titre, l'URI "file" prend en charge la sensibilité à la casse afin de conserver la casse comme elle est. Tout traitement en rapport avec le transport du schéma d'URI "file" DOIT conserver la casse comme elle est. Toute transposition de ou vers une forme insensible à la casse est de la seule responsabilité du traitement de mise en œuvre de l'URI "file" au nom du système de fichier référencé.

Voir aussi l'Appendice E, qui donne la listes de quelques variantes syntaxiques non standard qui peuvent se rencontrer en pratique.

3. Opérations qui impliquent des URI <file>

Voir le fichier POSIX et les opérations de répertoire [POSIX] pour des exemples d'opérations normalisées qui peuvent être effectuées sur les fichiers.

Un URI "file" ne peut être déréférencé ou traduit en un chemin de fichier local que si il est local. Un URI "file" est considéré comme "local" si il n'a pas de "file-auth", ou si le "file-auth" est la chaîne spéciale "localhost", ou un nom de domaine pleinement qualifié qui se résout en la machine de laquelle l'URI est interprété (Section 2).

La présente spécification ne définit ni n'interdit aucun ensemble d'opérations qui pourraient être effectuées sur un fichier identifié par un URI "file" non local.

4. Codage du nom de système de fichiers

Les systèmes de fichiers utilisent divers schémas de codage pour mémoriser les noms de fichier et de répertoire. De nombreux systèmes modernes de fichiers mémorisent les noms de fichiers et de répertoires comme des séquences arbitraires d'octets, auquel cas la représentation comme chaîne codée dépend souvent des réglages de localisation de l'utilisateur ou par défaut se fait en UTF-8 [RFC3629].

Quand un URI "file" produit représente des données de texte consistant en caractères de l'ensemble de caractères de la norme Unicode [UNICODE], les données DEVRAIENT être codées comme des octets conformément au schéma de codage de caractères UTF-8 [RFC3629] avant que soit appliqué le codage en pourcentage (conformément au paragraphe 2.5 de la [RFC3986]).

Une décision de ne pas utiliser l'UTF-8 codé en pourcentage sort du domaine d'application de la présente spécification. Elle va normalement exiger l'utilisation d'une heuristique ou d'une connaissance explicite de la façon dont la chaîne va être traitée.

5. Considérations sur la sécurité

De nombreuses considérations de sécurité sur les schémas d'URI sont discutées dans la [RFC3986].

L'accès au fichier et l'attribution de privilèges pour des opérations spécifiques sont des sujets complexes, et l'utilisation des URI "file" peut compliquer le modèle de sécurité appliqué pour les privilèges de fichier.

Historiquement, les agents d'utilisateur ont accordé au contenu provenant du schéma d'URI "file" une quantité énorme de privilèges. Cependant, accorder à tous les fichiers locaux des privilèges aussi larges peut conduire à des attaques d'escalade de privilèges. Certains agents d'utilisateur ont eu des succès en accordant aux fichiers locaux des privilèges fondés sur le répertoire, mais cette approche n'a pas été largement adoptée. D'autres agents d'utilisateur utilisent des identifiants uniques au monde comme origine pour chaque URI "file" [RFC6454], ce qui est l'option la plus sûre.

Traiter un URI "file" non local comme local, ou tenter autrement d'effectuer des opérations locales sur un URI non local peut résulter en des problèmes de sécurité.

Les systèmes de fichiers allouent normalement une signification opérationnelle à des caractères spéciaux, tels que les caractères "/", "\", ":", "[", et "]", et aux noms d'appareils spéciaux tels que ".", "..", "...", "aux", "lpt", etc. Dans certains cas, vérifier simplement l'existence d'un tel nom va causer une pause du système d'exploitation ou invoquer des appels système sans relation, conduisant à de sérieux problèmes de sécurité concernant des déni de service et des transferts de données involontaires. Il n'est pas possible à la présente spécification de faire la liste de tous ces caractères et noms d'appareil significatifs. Les développeurs devraient chercher les noms et caractères réservés pour les types d'appareils de mémorisation qui peuvent être rattachés à leur application et restreindre en conséquence l'utilisation des données obtenues de composants d'URI.

Les systèmes de fichiers varient dans la façon dont ils traitent la casse. Il faut faire attention à éviter les problèmes résultant de possibles alias inattendus provenant de la seule différence de casse entre les chemins de fichiers ou des URI, ou de discordances de codage ou d'équivalences Unicode [UAX15] (voir la Section 4).

6. Considérations relatives à l'IANA

Le présent document définit le schéma d'URI permanent suivant. Le registre "Schémas d'identifiant de ressource universel (URI)" r a été mis à jour en conséquence. Cet enregistrement se conforme à la [RFC7595].

Nom de schéma : file

Statut : permanent

Applications/protocoles qui utilisent ce nom de schéma : Couramment utilisé par les documents hypertexte pour se référer à des fichiers sans dépendre de l'accès réseau. Pris en charge par les navigateurs majeurs.

Utilisé dans des bibliothèques de développement telles que :

* Windows Shell (PathCreateFromUrl, UrlCreateFromPath)

* libwww-perl - The World-Wide Web library for Perl

Contact : Applications and Real-Time Area <art@ietf.org>

Contrôleur des changements : IETF <ietf@ietf.org>

Références : cette RFC

7. Références

7.1 Références normatives

[RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))

[RFC3629] F. Yergeau, "[UTF-8, un format de transformation](#) de la norme ISO 10646", STD 63, novembre 2003.

[RFC3986] T. Berners-Lee, R. Fielding et L. Masinter, "[Identifiant de ressource uniforme \(URI\) : Syntaxe générique](#)", STD 66, janvier 2005, DOI 10.17487/RFC3986.

[RFC5234] D. Crocker, éd., P. Overell, "[BNF augmenté pour les spécifications de syntaxe](#) : ABNF",

DOI 10.17487/RFC5234, janvier 2008. ([STD0068](#))

- [RFC6454] A. Barth, "[Le concept d'origine de la Toile](#)", DOI 10.17487/RFC6454, décembre 2011. (*P.S.*)
- [[RFC6874](#)] B. Carpenter, S. Cheshire, R. Hinden, "Représentation des identifiants de zone IPv6 dans les identifiants de ressource et les littéraux d'adresse", DOI 10.17487/RFC6874, février 2013. (*MàJ RFC3986*) (*P.S.*)

7.2 Références pour information

- [Bash-Tilde] Free Software Foundation, Inc, "Bash Reference Manual: Tilde Expansion", septembre 2016, <http://www.gnu.org/software/bash/manual/html_node/Tilde-Expansion.html>.
- [Bug107540] Bugzilla@Mozilla, "Bug 107540", octobre 2001, <https://bugzilla.mozilla.org/show_bug.cgi?id=107540>.
- [MS-DTYP] Microsoft, "Windows Data Types: 2.2.57 UNC", octobre 2015, <<http://msdn.microsoft.com/en-us/library/gg465305.aspx>>.
- [POSIX] IEEE, "IEEE Std 1003.1, 2013 Edition - Standard for Information Technology-- Portable Operating System Interface (POSIX(R)) Base Specifications, Issue 7", DOI 10.1109/IEEESTD.2013.6506091, avril 2013.
- [RFC1630] 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", DOI 10.17487/RFC1630, juin 1994. (*Information*)
- [RFC1738] T. Berners-Lee et autres, "[Localisateurs uniformes de ressource](#) (URL)", DOI 10.17487/RFC1738, décembre 1994. (*P.S.*, *Obsolète*, voir les RFC4248 et 4266 ; *MàJ par RFC8089*)
- [RFC2396] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiants de ressource uniformes (URI) : Syntaxe générique", DOI 10.17487/RFC2396, août 1998. (*Obsolète*, voir [RFC3986](#))
- [[RFC6838](#)] N. Freed, J. Klensin et T. Hansen, "Spécifications de type de support et procédures d'enregistrement", DOI 10.17487/RFC6838, BCP 13, janvier 2013.
- [[RFC7595](#)] D. Thaler, et autres, "[Lignes directrices et procédures d'enregistrement](#) pour les schémas d'URI", juin 2015. BCP 35, (*Remplace RFC4395 ; MàJ par RFC8615*)
- [UAX15] Davis, M., Ed. and K. Whistler, Ed., "Unicode Standard Annex #15: Unicode Normalization Forms", février 2016, <<http://www.unicode.org/reports/tr15/tr15-44.html>>.
- [UNICODE] The Unicode Consortium, "The Unicode Standard, Version 9.0.0", ISBN 978-1-936213-13-9, juin 2016, <<http://www.unicode.org/versions/Unicode9.0.0/>>.
- [WHATWG-URL] WHATWG, "URL Living Standard", January 2017, <<https://url.spec.whatwg.org/>>.
- [Win32-Namespaces] Microsoft Developer Network Blogs, "Naming Files, Paths, and Namespaces", juin 2013, <[https://msdn.microsoft.com/en-us/library/windows/desktop/aa365247\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa365247(v=vs.85).aspx)>.
- [Zsh-Tilde] "The Z Shell Manual: 14.7 Filename Expansion", décembre 2015, <<http://zsh.sourceforge.net/Doc/Release/Expansion.html#Filename-Expansion>>.

Appendice A. Différences avec les spécifications précédentes

La définition de la syntaxe de la Section 2 hérite des différences incrémentaires de la syntaxe générale de la [RFC1738], comme décrit par l'Appendice G de la [RFC2396] et l'Appendice D de la [RFC3986].

Conformément à la définition de la [RFC1738], un URL "file" commence toujours par le jeton "file://", suivi par un nom d'hôte (facultativement blanc) et un caractère "/". La syntaxe donnée à la Section 2 rend facultatif le composant d'autorité entier, y compris la double barre oblique "//".

Appendice B. Exemples d'URI

La syntaxe de la Section 2 est destinée à prendre en charge les URI "file" qui prennent les formes suivantes :

Fichiers locaux :

- o Un URI "file" traditionnel pour un fichier local avec une autorité vide. C'est le format le plus courant en usage aujourd'hui. Par exemple : "file:///path/to/file"
- o La représentation minimale d'un fichier local sans champ d'autorité et un chemin absolu qui commence par une barre oblique "/". Par exemple : "file:/path/to/file"

Fichiers non locaux :

- o Un fichier non local avec une autorité explicite. Par exemple : "file://host.exemple.com/path/to/file"

Appendice C. Technologies similaires

- o La spécification d'URL WHATWG [WHATWG-URL] définit le comportement du navigateur pour diverses entrées, incluant les URI "file". Document vivant, il change pour refléter les mises à jour du comportement du navigateur. Par suite, ses définitions d'algorithmes et de syntaxe peuvent ou non être cohérentes avec la présente spécification. Les développeurs devraient connaître cette possible discordance si ils s'attendent à partager des URI "file" avec des navigateurs qui suivent la spécification WHATWG.
- o La convention universelle de dénomination (UNC, *Universal Naming Convention*) [MS-DTYP] définit un format de chaîne qui peut remplir un rôle similaire à celui du schéma d'URI "file" pour décrire la localisation des fichiers, sauf que les fichiers localisés par les chaînes de sélecteur de l'espace de fichiers UNC sont normalement mémorisés sur une machine distante et qu'on y accède en utilisant un protocole réseau. L'Appendice E.3 présente quelques unes de façons dont les chaînes de sélecteur de l'espace de fichiers UNC sont actuellement faites pour interopérer avec le schéma d'URI "file".
- o L'API Microsoft Windows définit les espaces de noms Win32 [Win32-Namespaces] pour interagir avec les fichiers et appareils qui utilisent les fonctions d'API Windows. Les chemins de cet espace de noms sont précédés de "\\?\\" pour les espaces de noms de fichiers Win32 et de "\\.\\" pour les espaces de noms d'appareil Win32. Il y a aussi un cas particulier pour les chemins de fichiers UNC dans les espaces de noms de fichier Win32, auquel on se réfère sous le nom de "Long UNC", qui utilise le préfixe "\\?\UNC\". La présente spécification ne définit pas de mécanisme pour traduire les chemins de cet espace de noms de ou vers les URI "file".

Appendice D. Opérations spécifiques du système

Cet appendice n'est pas normatif. Il souligne certains comportements observés et donne des lignes directrices spécifiques du système pour interagir avec les URI et chemins "file". Ce n'est pas une liste exhaustive des systèmes d'exploitation ou de fichiers ; elle est plutôt destinée à illustrer certains types d'interactions qui peuvent être rencontrés.

D.1 Systèmes POSIX

Dans un système de fichiers POSIX, la racine du système de fichiers est représentée comme un répertoire avec un nom de longueur zéro, généralement écrit comme "/" ; la présence de cette racine dans un URI "file" peut être prise comme la barre oblique initiale dans la règle "path-absolute".

Les coquilles UNIX courantes telles que Bourne-Again SHell (bash) et Z SHell (zsh) fournissent une fonction connue sous le nom de "expansion de tilde" [Bash-Tilde] ou "expansion de nom de fichier" [Zsh-Tilde], où un chemin qui commence par un caractère tilde "~" peut être expansé à partir d'un nom de répertoire particulier. Une telle facilité n'existe pas avec l'utilisation du schéma d'URI "file" ; un tilde dans un URI "file" est toujours juste un tilde.

D.2 Systèmes de style DOS et Windows

Quand on transpose un chemin de fichier de type DOS ou Windows en un URI "file", la lettre de pilote (par exemple, "c:") est normalement transposée en le premier segment de chemin.

L'Appendice E mentionne des techniques non standard pour interagir avec les chemins de fichiers de style DOS ou Windows et les URI.

D.3 Systèmes Mac OS X

Le système de fichiers hiérarchisé enrichi (HFS+, *Hierarchical File System Plus*) utilise une forme de normalisation non standard, similaire à la forme de normalisation D [UAX15]. Il faut faire attention quand on transforme des chemins de fichiers HFS+ de et vers des URI (Section 4).

D.4 Systèmes Files-11 OpenVMS

Quand on transpose un chemin de fichier d'un système de mémoire virtuelle (VMS, *Virtual Memory System*) en un URI "file", le nom d'appareil est transposé en le premier segment de chemin. Noter que le signe dollars "\$" est un caractère réservé selon la définition du paragraphe 2.2 de la [RFC3986], de sorte qu'il devrait être codé en pourcentage si il est présent dans le nom d'appareil.

Si le chemin de fichiers VMS inclut une référence de nœud, cette référence est utilisée comme autorité. Lorsque la référence de nœud d'origine inclut un nom d'utilisateur et un mot de passe dans une chaîne de contrôle d'accès, ils peuvent être transcrits en l'autorité en utilisant l'extension de syntaxe non standard de l'Appendice E.1.

Appendice E. Variantes de syntaxe non standard

Ces variantes peuvent être rencontrées dans les utilisations existantes du schéma d'URI "file" mais ne sont pas prises en charge par la syntaxe normative de la Section 2. Le présent appendice n'est pas normatif.

E.1 Informations d'utilisateur

Il peut être nécessaire d'inclure des informations d'utilisateur telles qu'un nom d'utilisateur dans un URI "file", par exemple, quand on transpose un chemin de fichier VMS avec une référence de nœud qui inclut une chaîne de contrôle d'accès.

Pour permettre que soient incluses les informations d'utilisateur dans un URI "file", la règle "file-auth" de la Section 2 peut être remplacée par :

```
file-auth = "localhost" / [ userinfo "@" ] host
```

Cela utilise la règle "userinfo" de la [RFC3986].

Comme discuté dans l'article de la documentation de système OpenVMS de HP disponible à <http://h71000.www7.hp.com/doc/84final/ba554_90015/ch03s09.html>, "les chaînes de contrôle d'accès incluent des informations suffisantes pour permettre à quelqu'un de faire intrusion dans le compte distant, [donc] elles créent une sérieuse exposition de la sécurité". Dans une veine similaire, la présence d'un mot de passe dans un champ userinfo "user:password" est déconseillée par la [RFC3986]. Il faut faire attention quand on traite ces informations qui peuvent être utilisées pour identifier un utilisateur ou accorder l'accès à un système.

E.2 Lettres de pilote DOS et Windows

Sur les systèmes de fichier du style Windows ou DOS, un chemin de fichier absolu peut commencer par une lettre de pilote. Pour faciliter cela, la règle "local-path" de la Section 2 peut être remplacée par la suivante :

```
local-path = [ drive-letter ] path-absolute  
drive-letter = ALPHA ":"
```

La règle "ALPHA" est définie dans la [RFC5234].

Ceci est destiné à prendre en charge la représentation minimale d'un fichier local dans un environnement de style DOS ou Windows, sans champ autorité et avec un chemin absolu qui commence par une lettre de pilote. Par exemple :

```
"file:c:/path/to/file"
```

Les URI de la forme "file:///c:/path/to/file" sont déjà pris en charge par la règle "path-absolute".

Noter que la comparaison des lettres de pilote dans les chemins de fichiers DOS ou Windows est insensible à la casse. Dans certaines utilisations des URI "file", les lettres de pilote sont canonisées en les convertissant en majuscules ; d'autres usages traitent les URI qui ne diffèrent que par la casse de la lettre de pilote comme identiques.

E.2.1 Résolution relative

Pour imiter le comportement des systèmes de fichiers de style DOS ou Windows, les références relatives qui commencent par une barre oblique "/" peuvent être résolues par rapport à la lettre de pilote quand elle est présente ; la résolution des segments de points "." (selon le paragraphe 5.2.4 de la [RFC3986]) peut être modifiée pour ne jamais écraser la lettre de pilote. Par exemple :

```
L'URI de base :   file:///c:/path/to/file.txt
référence relative : /some/other/thing.bmp
se résout en :   file:///c:/some/other/thing.bmp
```

```
L'URI de base :   file:///c:/foo.txt
référence relative : ../bar.txt
se résout en:     file:///c:/bar.txt
```

Une référence relative qui commence par une lettre de pilote va être interprétée par un analyseur d'URI générique comme un URI dont la lettre de pilote est son schéma. Au lieu de cela, une telle référence devrait être construite avec un caractère barre oblique "/" en tête (par exemple, "/c:/foo.txt").

Les références relatives avec une lettre de pilote suivie par un caractère autre qu'une barre oblique (par exemple, "/c:bar/baz.txt" ou "/c:../foo.txt") peuvent n'être pas acceptées comme des URI déréréférencables dans les systèmes de style DOS ou Windows.

E.2.2 Caractère ligne verticale

Historiquement, certaines utilisation des URI "file" ont inclus un caractère ligne verticale "|" au lieu d'un caractère deux-points ":" dans la construction de la lettre de pilote. La [RFC3986] interdit l'utilisation de la ligne verticale ; cependant, il peut être nécessaire d'interpréter ou mettre à jour de vieux URI.

Pour interpréter de tels URI, les règles "auth-path" et "local-path" de la Section 2 et la règle "drive-letter" ci-dessus peuvent être remplacées par les suivantes :

```
auth-path    = [ file-auth ] path-absolute / [ file-auth ] file-absolute
local-path   = [ drive-letter ] path-absolute / file-absolute
file-absolute = "/" drive-letter path-absolute
drive-letter = ALPHA ":" / ALPHA "|"
```

Ceci est destiné à prendre en charge des URI "file" DOS ou Windows réguliers avec des caractères ligne verticale dans la construction de lettre de pilote. Par exemple :

- o "file:///c:/path/to/file"
- o "file:c:/path/to/file"
- o "file:c|path/to/file"

Pour mettre à jour ces vieux URI, on remplace la ligne verticale "|" par un deux-points ":".

E.3 Chaînes UNC

Certains usages du schéma d'URI "file" permettent que des chaînes de sélecteur d'espace de fichiers UNC [MS-DTYP] soient traduites en et des URI "file", soit en transposant les segments équivalents des deux schémas (hostname en authority, sharename+objectnames en path) soit en transposant la chaîne UNC entière en le segment de chemin d'un URI.

E.3.1 URI <file> avec autorité

Voici une description algorithmique du processus de traduction d'une chaîne de sélecteur d'espace de fichiers UNC en un URI "file" en transposant les segments équivalents des deux schémas :

1. Initialiser l'URI avec l'identifiant de schéma "file:".
2. Ajouter l'autorité :
 1. Ajouter le cachet d'autorité "/" à l'URI.
 2. Ajouter le champ nom d'hôte de la chaîne UNC à l'URI.
3. Ajouter le nom de partage "share-name" :
 1. Transformer le nom de partage en un segment de chemin (voir le paragraphe 3.3 de la [RFC3986]) pour se conformer aux règles de codage de la Section 2 de la [RFC3986].
 2. Ajouter un caractère barre oblique "/" délimiteur et le segment transformé à l'URI.
4. Pour chaque nom d'objet :
 1. Transformer le nom d'objet en segment de chemin comme ci-dessus. Le caractère deux-points ":" est permis comme délimiteur avant le nom de flux et le type de flux dans le nom de fichier, si ils sont présents.
 2. Ajouter un caractère délimiteur barre oblique "/" et le segment transformé à l'URI.

Par exemple, la chaîne UNC : \\host.example.com\Share\path\to\file.txt
serait transformée en l'URI : file://host.example.com/Share/path/to/file.txt

L'algorithme inverse pour traduire un URI "file" en une chaîne de sélecteur d'espace de fichiers UNC est laissé comme exercice pour l'entraînement du lecteur.

E.3.2 URI <file> avec chemin UNC

Il est courant de rencontrer des URI "file" qui codent les chaînes UNC entières dans le chemin, généralement avec tous les caractères barre oblique inverse "\" remplacés par des caractères barre oblique "/".

Pour interpréter de tels URI, la règle "auth-path" de la Section 2 peut être remplacée par la suivante :

```
auth-path    = [ file-auth ] path-absolute / unc-authority path-absolute
unc-authority = 2*3"/" file-host
file-host    = inline-IP / IPv4address / reg-name
inline-IP    = "%5B" ( IPv6address / IPvFuture ) "%5D"
```

Cette syntaxe utilise les règles "IPv4address", "IPv6address", "IPvFuture", et "reg-name" de la [RFC3986].

Noter que la règle "file-host" est la même que "host" mais avec le codage en pourcentage appliqué aux caractères "[" et "]". Cette syntaxe étendue est destinée à prendre en charge les URI qui prennent les formes suivantes, en plus de celles de l'Appendice B :

Fichiers non locaux :

- o La représentation d'un fichier non local avec une autorité vide et une chaîne UNC complète (transformée) dans le chemin. Par exemple : "file:///host.example.com/path/to/file"
- o Comme ci-dessus, avec une barre oblique supplémentaire entre l'autorité vide et la chaîne UNC transformée, selon la syntaxe définie dans la [RFC1738]. Par exemple : "file://///host.example.com/path/to/file"

Cette représentation est notoirement utilisée par le navigateur Firefox. Voir [Bug107540].

Elle limite aussi la définition d'un "URI de fichier local" en excluant tout URI "file" avec un chemin qui code une chaîne UNC.

E.4 Barre oblique inverse comme séparateur

Historiquement, certaines utilisations ont copié des chemins de fichiers entiers dans les composants de chemin des URI "file". Lorsque des chemins de fichiers DOS ou Windows étaient ainsi copiés, les chaînes d'URI résultantes contenaient des caractères barre oblique inverse "\" non codés, ce qui est interdit à la fois par la [RFC1738] et par la [RFC3986].

Il est possible de traduire ou mettre à jour un tel URI "file" invalide en remplaçant toutes les barres obliques inverses "\" par des barres obliques "/" si on peut déterminer avec une certitude raisonnable que les barres obliques inverses sont destinées à être des séparateurs de chemin.

Appendice F. Règles non standard collectées

Voici les règles de syntaxe collectées de tous les appendices facultatifs, présentées pour un résumé pratique. Cette collection de règles syntaxiques n'est pas normative.

```

file-URI      = file-scheme ":" file-hier-part
file-scheme   = "file"
file-hier-part = ( "/" auth-path ) / local-path
auth-path     = [ file-auth ] path-absolute / [ file-auth ] file-absolute / unc-authority path-absolute
local-path    = [ drive-letter ] path-absolute / file-absolute
file-auth     = "localhost" / [ userinfo "@" ] host
unc-authority = 2*3 "/" file-host
file-host     = inline-IP / IPv4address / reg-name
inline-IP     = "%5B" ( IPv6address / IPvFuture ) "%5D"
file-absolute = "/" drive-letter path-absolute
drive-letter  = ALPHA ":" / ALPHA "|"

```

Cette collection de règles de syntaxe est destinée à prendre en charge les URI "file" qui prennent les formes suivantes :

Fichiers locaux :

- o Un URI "file" traditionnel pour un fichier local avec une autorité vide. Par exemple : "file:///path/to/file"
- o La représentation minimale d'un fichier local sans champ "authority" et un chemin absolu qui commence par une barre oblique "/". Par exemple : "file:/path/to/file"
- o La représentation minimale d'un fichier local dans un environnement fondé sur le DOS ou Windows sans champ "authority" et avec un chemin absolu qui commence par une lettre de pilote. Par exemple : "file:c:/path/to/file"
- o Les URI "file" DOS ou Windows réguliers avec des caractères ligne verticale dans la construction de lettre de pilote. Par exemple :
 - * "file:///c:/path/to/file"
 - * "file:c/path/to/file"
 - * "file:c/path/to/file"

Fichiers non locaux :

- o La représentation d'un fichier non local avec une autorité explicite. Par exemple :
 - * "file://host.example.com/path/to/file"
- o La représentation "traditionnelle" d'un fichier non local avec une autorité vide et une chaîne UNC complète (transformée) dans le chemin. Par exemple :
 - * "file:///host.example.com/path/to/file"
- o Comme ci-dessus, avec une barre oblique supplémentaire entre l'autorité vide et la chaîne UNC transformée. Par exemple : "file://///host.example.com/path/to/file"

Remerciements

Les contributions de nombreux membres des communautés de l'IETF et du W3C -- en particulier Dave Crocker, Graham Klyne, Tom Petch, et John Klensin -- ont été très appréciées.

Des remerciements supplémentaires sont dus à Dave Risney, auteur de l'article d'information IEBlog qui figure à <<http://blogs.msdn.com/b/ie/archive/2006/12/06/file-uris-in-windows.aspx>>, et à Dave Thaler pour leurs commentaires précoces et leurs suggestions, ainsi qu'à Paul Hoffman, dont les travaux ont inspiré cette entreprise.

Adresse de l'auteur

Matthew Kerwin
 Queensland University of Technology
 Victoria Park Road
 Kelvin Grove, QLD 4059
 Australia

mél : matthew.kerwin@qut.edu.au