

Groupe de travail Réseau

**Request for Comments : 2227**

Catégorie : En cours de normalisation

Traduction : Claude Brière de L'Isle

J. Mogul, DECWRL

P. Leach, Microsoft

octobre 1997

# Simple mesure du nombre d'accès et limitation d'usage pour HTTP

## Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

## Notice de Copyright

Copyright (C) The Internet Society (1997). Tous droits réservés.

## Résumé

Le présent document propose une simple extension à HTTP, utilisant un nouvel en-tête "Meter", qui permet une forme limitée d'informations démographiques (appelé familièrement "compte de touches", *hit metering*) à rapporter par les antémémoires aux serveurs d'origine, d'une manière plus efficace que les techniques de "court-circuit d'antémémoire" (*cache busting*) actuellement utilisées. Il permet aussi à un serveur d'origine de contrôler le nombre de fois qu'une antémémoire utilise une réponse mémorisée, et souligne une technique que les serveurs d'origine peuvent utiliser pour capturer des informations de référence sans "court-circuit d'antémémoire".

## Table des Matières

1.	<a href="#">Introduction.....</a>
1.1	<a href="#">Buts, non buts, et limitations.....</a>
1.2	<a href="#">Bref résumé du concept.....</a>
1.3	<a href="#">Terminologie.....</a>
2	<a href="#">Vue d'ensemble.....</a>
2.1	<a href="#">Discussion.....</a>
3	<a href="#">Arrangement des concepts.....</a>
3.1	<a href="#">Mise en œuvre de la "sous arborescence de mesure".....</a>
3.2	<a href="#">Format de l'en-tête Meter.....</a>
3.3	<a href="#">Négociation du compte de touches et de la limitation d'usage.....</a>
3.4	<a href="#">Transmission des rapports d'usage.....</a>
3.5	<a href="#">Quand envoyer les rapports d'usage.....</a>
3.6	<a href="#">Subdivision des limites d'usage.....</a>
4	<a href="#">Analyse.....</a>
4.1	<a href="#">Approximation de la précision du comptage des usagers.....</a>
4.2	<a href="#">Qu'en est-il des ordinateurs en réseau ?.....</a>
4.3	<a href="#">Analyse des délais du chemin critique.....</a>
5	<a href="#">Spécification.....</a>
5.1	<a href="#">Spécification de l'en-tête Meter et directives.....</a>
5.2	<a href="#">Abréviations pour les directives Meter.....</a>
5.3	<a href="#">Règles de comptage.....</a>
5.4	<a href="#">Règles de comptage : interaction avec les demandes Range.....</a>
5.5	<a href="#">Mise en œuvre par les mandataires qui ne mettent pas en antémémoire.....</a>
5.6	<a href="#">Mise en œuvre par des antémémoire coopératives.....</a>
6	<a href="#">Exemples.....</a>
6.1	<a href="#">Exemple d'un ensemble d'échanges complets.....</a>
6.2	<a href="#">Protection contre les mandataires HTTP/1.0.....</a>
6.3	<a href="#">Exemples plus élaborés.....</a>
7	<a href="#">Interactions avec la négociation de contenu.....</a>
7.1	<a href="#">Traitement des réponses portant un en-tête Vary.....</a>
7.2	<a href="#">Interaction avec la négociation de contenu transparent.....</a>
8	<a href="#">Note sur la capture des Referer.....</a>
9	<a href="#">Autres propositions.....</a>
10	<a href="#">Considérations pour la sécurité.....</a>

<a href="#">11</a>	<a href="#">Remerciements.....</a>
<a href="#">12</a>	<a href="#">Références.....</a>
<a href="#">13</a>	<a href="#">Adresse des auteurs.....</a>
<a href="#">14</a>	<a href="#">Déclaration complète de droits de reproduction.....</a>

## 1. Introduction

Pour diverses raisons, les fournisseurs de contenu veulent être capables de collecter des informations sur la fréquence à laquelle on accède à leurs contenus. Ce désir conduit à certains des "courts-circuits d'antémémoire" effectués par les serveurs existants. (Le "court-circuit d'antémémoire" est l'utilisation par les serveurs de techniques destinées à empêcher la mise en antémémoire des réponses ; on ne sait pas exactement à quel point il est répandu.) Cette espèce de court-circuit d'antémémoire n'est pas faite dans le but de maintenir les propriétés de transparence ou de sécurité, mais simplement de collecter des informations démographiques. Le court-circuit d'antémémoire est aussi effectué pour faire apparaître différentes images publicitaires sur la même page (c'est-à-dire que chaque restitution de la page voit une publicité différente).

La présente proposition prend en charge un modèle similaire à celui des éditeurs de publications sur papier : ces éditeurs rapportent (essayent de rapporter) à leurs annonceurs le nombre de personnes qui lisent au moins une fois leur publication ; ils n'essayent pas de savoir combien de fois un lecteur la lit. Ils le font en comptant les copies publiées, et essayent ensuite d'estimer, pour leur publication, combien de personnes en moyenne lisent une même copie au moins une fois. Le point clé est que le résultat n'est pas exact, mais c'est quand même utile. Un autre modèle est de faire des enquêtes codées de façon telle que l'annonceur puisse dire quelle publication a produit l'enquête.

### 1.1 Buts, non buts, et limitations

HTTP/1.1 permet déjà aux serveurs d'origine d'empêcher la mise en antémémoire des réponses, et la preuve existe [9] qu'au moins parfois, cela est fait dans le seul but de collecter le compte du nombre d'accès à des pages spécifiques. Une partie de ces preuves est déduite de l'étude des traces des mandataires ; une autre partie se fonde sur des déclarations explicites de l'intention des opérateurs des serveurs de la Toile. Les informations collectées de cette façon peuvent ou non être d'une utilité réelle pour ceux qui les collectent ; le fait est qu'ils veulent les collecter, ou le font déjà.

L'objectif de la présente proposition est de fournir une optimisation facultative des performances pour cette utilisation de HTTP/1.1.

Cette spécification est :

- Facultative : aucun serveur ni mandataire n'est obligé de la mettre en œuvre.
- Centrée sur le mandataire : il n'y a aucune implication de la part des mises en œuvre du client final.
- Seulement une optimisation des performances : elle ne fournit pas d'information ou de fonctionnalité qui ne soit déjà disponible dans HTTP/1.1. L'intention est d'améliorer les performances globales, et de réduire la latence pour presque toutes les interactions ; la latence peut être augmentée pour une petite fraction des interactions HTTP.
- Au mieux : elle ne garantit pas la précision des informations rapportées, bien qu'elle fournisse des résultats précis en l'absence de défaillance persistante du réseau ou de l'hôte.
- Neutre par rapport à la confidentialité : elle ne révèle aux serveurs aucune information sur les clients qui ne soit déjà disponible par les caractéristiques existantes de HTTP/1.1.

Les buts de cette spécification n'incluent pas :

- de résoudre la totalité du problème de l'obtention efficace d'informations détaillées sur les demandes faites via les mandataires,
- d'améliorer la protection de la confidentialité des utilisateurs (bien que notre proposition puisse réduire le transfert aux serveurs d'informations spécifiques de l'utilisateur, elle ne les empêche pas),
- d'empêcher ou d'encourager l'utilisation de mécanismes d'échange de journaux d'événements,
- d'éviter toutes les formes de "court-circuit d'antémémoire", ou même de tout le court-circuit d'antémémoire fait pour rassembler les comptes.

Ce concept a certaines limitations potentielles :

- Si il n'est pas déployé largement à la fois chez les mandataires et chez les serveurs, il ne fournira que peu d'avantages.
- Il peut, en résolvant partiellement le problème du compte d'accès, réduire la pression pour adopter des solutions plus complètes, si il en devenaient de disponibles.
- Même largement déployé, il peut n'être pas largement utilisé, et ne pourrait ainsi pas accroître significativement les performances.

Ces limitations potentielles peuvent n'être pas des problèmes dans la pratique réelle.

## 1.2 Bref résumé du concept

Ce paragraphe est inclus pour les gens qui ne souhaitent pas lire la totalité du document ; ce n'est pas la spécification du concept proposé, et il simplifie à l'extrême de nombreux aspects du concept.

Le but de ce concept est d'éliminer le besoin que les serveurs d'origine utilisent les techniques de "court-circuit d'antémémoire" (*cache-busting*), lorsque elles ne sont utilisées que pour le comptage du nombre d'utilisateurs d'une ressource. (Le court-circuit d'antémémoire inclut des techniques telles que d'établir des dates d'expiration immédiates, ou l'envoi de "Cache-control: private" dans chaque réponse.)

Le concept ajoute un nouvel en-tête "Meter" à HTTP ; l'en-tête est toujours protégé par l'en-tête "Connection", et est donc toujours bond par bond. Ce mécanisme permet la construction d'une "sous arborescence de mesure", qui est une sous arborescence connectée de mandataires (*proxy*), qui ont pour racine un serveur d'origine. Seuls les mandataires qui sont explicitement volontaires pour se joindre à la sous arborescence de mesure pour une ressource participent au compte des accès, mais les mandataires qui sont volontaires sont obligés de faire de leur mieux pour fournir des comptes précis. Lorsque une réponse de compte d'accès est transmise en dehors de la sous arborescence de mesure, le mandataire qui la transmet ajoute "Cache-control: s-maxage=0", de sorte que les autres mandataires (en dehors de la sous arborescence de mesure) soient forcés de transmettre toutes les demandes à un serveur dans la sous arborescence de mesure.

Note : La spécification HTTP/1.1 ne définit pas actuellement une directive de contrôle d'antémémoire "s-maxage". Le groupe de travail HTTP a décidé d'ajouter une telle directive à la prochaine révision de HTTP/1.1 [7].

L'en-tête Meter porte zéro, une ou plusieurs directives, de façon similaire à celle dont l'en-tête Cache-control porte des directives. Les mandataires peuvent utiliser certaines directives Meter pour se porter volontaires pour faire le compte d'accès pour une ressource. Si un mandataire se porte volontaire, le serveur peut utiliser certaines directives pour demander qu'une réponse soit comptée dans les accès. Finalement, les mandataires utilisent une directive Meter "count" pour faire rapport des comptes d'accès accumulés.

Le mécanisme Meter peut aussi être utilisé par un serveur pour limiter le nombre d'utilisations qu'une antémémoire peut faire d'une réponse mise en antémémoire, avant de la revalider.

La spécification complète comporte toutes les règles du comptage des "utilisations" d'une réponse (par exemple, des GET non conditionnels) et des "réutilisations" (les GET conditionnels). Ces règles assurent que les résultats sont entièrement cohérents dans tous les cas, sauf ceux de défaillance des systèmes ou des réseaux.

## 1.3 Terminologie

Le présent document utilise des termes définis et expliqués dans la spécification HTTP/1.1 [4], incluant "serveur d'origine", "ressource", "bond par bond", "GET inconditionnel", et "GET conditionnel". Le lecteur est supposé familier avec la spécification HTTP/1.1 et sa terminologie.

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans la RFC2119 [1].

## 2 Vue d'ensemble

Le concept décrit dans le présent document introduit plusieurs nouvelles caractéristiques dans HTTP :

- Hit-metering : permet à un serveur d'origine d'obtenir des comptes raisonnablement précis du nombre de clients qui utilisent une instance de ressource via une antémémoire mandataire, ou une hiérarchie d'antémémoires mandataires.
- Usage-limiting : permet à un serveur d'origine de contrôler le nombre de fois qu'une réponse en antémémoire peut être utilisée par une antémémoire mandataire, ou une hiérarchie d'antémémoires mandataires, avant revalidation auprès du serveur d'origine.

Ces nouvelles caractéristiques non obligatoires n'exigent qu'une prise en charge minimale du nouveau protocole, aucun changement de la version du protocole, relativement peu de redondance dans les en-têtes de message. Le concept n'ajoute pas d'allers retours supplémentaires sur le réseau dans un chemin critique qui affecte directement la latence perçue par l'utilisateur (voir l'analyse au paragraphe 4.3).

Le but principal du compte de touches et de la limitation d'usage est d'éviter le besoin qu'un serveur d'origine envoie un "Cache-control: s-maxage=0" avec les réponses pour des ressources dont la valeur ne va probablement pas changer immédiatement. En d'autres termes, dans les cas où la seule raison pour contacter le serveur d'origine sur chaque demande qui pourrait autrement être satisfaite par une entrée d'antémémoire de mandataire est de permettre au serveur de collecter des informations démographiques ou de contrôler le nombre de fois qu'est utilisée une entrée d'antémémoire, l'extension proposée ici va éviter une quantité significative de trafic réseau et de latence inutiles.

Ce concept introduit un nouvel en-tête "Meter" qui sera utilisé à la fois dans les messages de demande HTTP et dans les messages de réponse HTTP. L'en-tête Meter est utilisé pour transmettre un certain nombre de directives et de rapports. En particulier, toute la négociation de l'utilisation du compte de touches et de la limitation d'usage est faite en utilisant cet en-tête. Aucun autre changement à la spécification existante de HTTP/1.1 [4] n'est proposé dans le présent document.

Le concept introduit plusieurs nouvelles idées :

1. Le concept d'une "utilisation" d'une entrée d'antémémoire, qui est quand un mandataire retourne son corps d'entité en réponse à une demande conditionnelle ou non conditionnelle, et de la "réutilisation" d'une entrée d'antémémoire, qui est quand un mandataire retourne une réponse 304 (Non modifié) pour une demande conditionnelle qui est satisfaite par cette entrée d'antémémoire.
2. Le concept d'une ressource dont le nombre de touches est compté, pour laquelle les antémémoires mandataires font une tentative au mieux pour rapporter des comptes d'utilisation et/ou réutilisation précis au serveur d'origine.
3. Le concept de ressource à limitation d'usage, pour laquelle le serveur d'origine attend que les antémémoires mandataires limitent le nombre d'utilisations et/ou réutilisations.

Les nouvelles directives Meter et Reports interagissent pour permettre aux antémémoires mandataires et aux serveurs de coopérer à la collecte de données démographiques. Le but est une approximation au mieux du vrai nombre d'utilisations et/ou réutilisations, et non un compte exact garanti.

Les nouvelles directives Meter permettent aussi à un serveur de limiter l'imprécision d'un compte de touches particulier, en mettant des bornes au nombre des utilisations entre les rapports. Il peut aussi, par exemple, limiter le nombre de fois que la même publicité est montrée parce qu'elle est dans l'antémémoire.

Le paragraphe 7.1 décrit un moyen pour utiliser la négociation de contenu pilotée par le serveur (l'en-tête Vary) qui permet à un serveur d'origine HTTP de séparer de façon souple les demandes en catégories et de compter les demandes par catégorie. La mise en œuvre d'un tel comptage de touches par catégories sera vraisemblablement une très petite modification à la plupart des mises en œuvre de Vary ; certaines mises en œuvre peuvent même ne requérir aucune modification du tout.

## 2.1 Discussion

En transposant cela dans le modèle de publication, une antémémoire mandataire va incrémenter le compte d'utilisations pour une entrée d'antémémoire une fois pour chaque GET inconditionnel effectué pour l'entrée, et une fois pour chaque GET conditionnel qui résulte en l'envoi d'une copie de l'entrée pour mettre à jour la copie invalide de l'antémémoire du client. Les GET conditionnels qui résultent en réponses 304 (Non modifié) ne sont pas incluses dans le compte d'utilisations, parce qu'elle ne résultent pas en ce qu'un nouvel utilisateur voit la page, mais signifient plutôt la répétition par un usager d'une consultation faite précédemment. Cependant, les réponses 304 sont comptées dans le compte des réutilisations. Les HEAD ne sont pas comptés du tout, parce que leurs réponses ne contiennent aucun corps d'entité.

Les directives Meter ne s'appliquent qu'aux antémémoires mandataires partagées, pas à celle de client final (ou autre utilisateur unique). Les antémémoires d'utilisateurs singuliers ne devraient pas utiliser Meter, parce que leurs touches seront automatiquement comptées comme résultant du GET inconditionnel avec lequel elles sont allées chercher la page pour la première fois, soit au serveur d'origine, soit dans une antémémoire mandataire. Les GET conditionnels suivants ne résultent pas en ce qu'un nouvel utilisateur voit la page.

Le mécanisme spécifié ici compte les GET ; d'autres méthodes ne résultent pas en une page à lire pour l'utilisateur, ne sont pas mises en antémémoire, ou sont "écrites à travers" et peuvent ainsi être directement comptées par le serveur d'origine. (Si

à l'avenir un "POST mettable en antémémoire" venait à exister, par lequel le corps d'entité dans la demande POST serait utilisé pour choisir une réponse dans l'antémémoire, de tels POST devraient alors être traités exactement comme les GET.) L'applicabilité du compte de touches à toute nouvelle méthode HTTP qui pourrait être définie à l'avenir ne peut être spécifiée pour l'instant.

Dans le cas de plusieurs antémémoires le long d'une chemin, une antémémoire mandataire fait les additions évidentes lorsque elle reçoit un compte d'utilisations ou de réutilisations dans une demande provenant d'une autre antémémoire.

### 3 Arrangement des concepts

Afin de permettre l'introduction du compte de touches et la limitation d'usage sans exiger de révision d'un protocole, et pour assurer une approximation raisonnablement proche de comptes précis, la négociation de la mesure et de la limitation d'usage est faite bond par bond, et non de bout en bout. Si on considère "l'arbre" des mandataires qui reçoivent, mémorisent, et transmettent une réponse spécifique, l'intention de ce concept est qu'au sein d'un "sous arbre de mesure" (éventuellement nul) dont la racine est au serveur d'origine, tous les mandataires utilisent les comptes de touches et/ou la limitation d'usage demandés par le serveur d'origine.

Les mandataires aux extrémités de ce sous arbre vont insérer une directive "Cache-control: s-maxage=0", qui force tous les autres mandataires (en dessous de ce sous arbre) à vérifier auprès d'une extrémité du sous arbre de mesure sur chaque demande. Cependant, cela ne les empêche pas de mémoriser et d'utiliser la réponse, si la revalidation réussit.

Aucun mandataire n'est obligé de mettre en œuvre le compte de touches ou la limitation d'usage. Cependant, tout mandataire qui transmet l'en-tête Meter dans une demande DOIT mettre en œuvre toutes les exigences inconditionnelles de la présente spécification, sans exception ni amendement.

C'est une conception prudente, qui peut parfois manquer à tirer parti de la prise en charge du compte de touches chez les mandataires en dehors du sous arbre de mesure. Cependant, il est vraisemblable que sans la fiabilité offerte par une conception prudente, les gestionnaires de serveurs d'origine avec une exigence d'approximations précises ne tireraient parti d'aucune proposition de compte de touches.

Le mécanisme de compte de touches/limitation d'usage- est conçu de façon à éviter tous allers retours supplémentaires sur le réseau dans le chemin critique de toute demande de client, et (autant que possible) à éviter d'allonger excessivement les messages HTTP.

L'en-tête Meter est utilisé pour transmettre à la fois les informations de négociation et les informations numériques.

Une spécification formelle de l'en-tête Meter figure à la section 5 ; l'exposé qui suit utilise une approche informelle dans un souci de clarté.

#### 3.1 Mise en œuvre de la "sous arborescence de mesure"

L'approche du "sous arbre de mesure" est mise en œuvre d'une façon simple et directe en définissant le nouvel en-tête "Meter" comme DEVANT toujours être protégé par un en-tête Connection dans toute demande ou réponse. C'est-à-dire que si l'en-tête Meter est présent dans un message HTTP, ce message :

1. DOIT contenir "Connection: meter", et DOIT être traité conformément à la spécification HTTP/1.1 de l'en-tête Connection.
2. NE DOIT PAS être envoyé en réponse à une demande provenant d'un client dont le numéro de version est inférieur à HTTP/1.1.
3. NE DOIT PAS être accepté d'un client dont le numéro de version est inférieur à HTTP/1.1.

La raison des deux dernières restrictions est de protéger contre les mandataires qui pourraient ne pas mettre en œuvre correctement l'en-tête Connection. Autrement, un sous arbre qui comporte un mandataire HTTP/1.0 pourrait apparaître à tort dans un sous arbre de mesure.

Note: Il apparaît que pour que le mécanisme d'en-tête Connection fonctionne correctement, un système qui reçoit un message HTTP/1.0 (ou de version inférieure) comportant un en-tête Connection doit agir comme si cet en-tête, et tous les en-têtes qu'il protège, devaient avoir été retirés du message par un mandataire intermédiaire.

Bien que la RFC2068 n'exige pas spécifiquement ce comportement, il apparaît implicite. Autrement, on ne pourrait pas s'appuyer sur la propriété déclarée (au paragraphe 14.10) que les options protégées "NE DOIVENT PAS être communiquées par les mandataires sur d'autres connexions." Cela devrait probablement être précisé dans un projet ultérieur

de la spécification de HTTP/1.1.

La présente spécification ne propose, en aucune façon une modification de l'en-tête Connection.

Du point de vue d'un serveur d'origine, les mandataires fonctionnent ensemble dans un sous arbre de mesure pour respecter les limites d'usage et pour maintenir des comptes d'usage précis. Lorsque un serveur d'origine spécifie une limite d'usage, un mandataire dans le sous arbre de mesure peut subdiviser cette limite parmi ses descendants dans le sous arbre si cela lui paraît convenable. De même, lorsque un mandataire dans le sous arbre reçoit un rapport d'usage, il s'assure que les touches représentées par ce rapport sont correctement additionnées et rapportées au serveur d'origine.

Lorsque un mandataire transmet une réponse de compte de touches ou de limite d'usage à un client (mandataire ou client final) qui n'est pas dans le sous arbre de mesure, il DOIT omettre l'en-tête Meter, et il DOIT ajouter "Cache-control: s-maxage=0" à la réponse.

### 3.2 Format de l'en-tête Meter

L'en-tête Meter est utilisé pour porter zéro, une ou plusieurs directives. Les en-têtes Meter multiples peuvent survenir dans un message HTTP, mais selon les règles du paragraphe 4.2 de la spécification HTTP/1.1 [4], ils peuvent être combinés en un seul en-tête (et devraient être combinés ainsi) pour réduire la redondance.

Par exemple, la séquence suivante d'en-têtes Meter

```
Meter: max-uses=3
Meter: max-reuses=10
Meter: do-report
```

peut être exprimée par :

```
Meter: max-uses=3, max-reuses=10, do-report
```

### 3.3 Négociation du compte de touches et de la limitation d'usage

Un serveur d'origine qui veut collecter les comptes de touches pour une ressource, en forçant simplement toutes les demandes à outrepasser toutes les antémémoires de mandataire, répondrait aux demandes sur la ressource avec "Cache-control: s-maxage=0". (Un serveur d'origine qui souhaite empêcher les mandataires HTTP/1.0 de mettre indûment en antémémoire la réponse pourrait aussi envoyer à la fois "Expires: <now>", pour empêcher une telle mise en antémémoire, et "Cache-control: max-age=NNNN", pour permettre à de nouveaux mandataires de mettre la réponse en antémémoire).

L'objet de l'en-tête Meter est de suppléer au besoin de "Cache-control: s-maxage=0" au sein d'un sous arbre de mesure. Ainsi, tout mandataire peut négocier l'utilisation du compte de touches et/ou de limitation d'usage avec le serveur du prochain bond. Si ce serveur est le serveur d'origine, ou fait déjà partie d'un sous arbre de mesure (dont la racine est au serveur d'origine) il peut alors achever la négociation, étendant par là le sous arbre de mesure pour inclure le nouveau mandataire.

Pour commencer la négociation, un mandataire envoie sa demande avec une des directives Meter suivantes :

will-report-and-limit

indique que le mandataire est d'accord et capable de retourner des rapports d'usage et va obéir à toute limite d'usage.

wont-report

indique que le mandataire va obéir aux limites d'usage mais ne va pas envoyer de rapports d'usage.

wont-limit

indique que le mandataire ne va pas obéir aux limites d'usage mais va envoyer des rapports d'usage.

Un mandataire qui ne veut ni obéir aux limites d'usage ni envoyer de rapports d'usage NE DOIT PAS transmettre un en-tête Meter dans la demande.

Par définition, un en-tête Meter vide :

```
Meter:
```

est équivalent à "Meter: will-report-and-limit", et ainsi, par la définition de l'en-tête Connection (voir au paragraphe 14.10 de la spécification HTTP/1.1 [4]), une demande qui contient

Connection: Meter

et pas d'en-tête Meter explicite est équivalente à une demande qui contient

Connection: Meter

Meter: will-report-and-limit

Cela rend le cas par défaut plus efficace.

Un serveur d'origine qui n'est pas intéressé par les mesures ou les limites d'usage de la ressource demandée ignore simplement l'en-tête Meter.

Si le serveur veut que le mandataire ne fasse pas le compte de touches et/ou la limitations d'usage, sa réponse devrait inclure une ou plusieurs des directives Meter suivantes :

Pour le compte de touches :

do-report

spécifie que le mandataire DOIT envoyer des rapports d'usage au serveur.

dont-report

spécifie que le mandataire NE DEVRAIT PAS envoyer de rapports d'usage au serveur.

timeout=NNN

règle un temporisateur de mesure de NNN minutes, à partir du moment où cette réponse a été générée, pour les rapports de compte de touches. Si le mandataire a un compte de touches différent de zéro pour cette réponse lorsque le temporisateur arrive à expiration, il DOIT envoyer un rapport au serveur à ce moment ou avant ce moment. Cela implique "do-report".

Par définition, un en-tête Meter vide dans une réponse, ou tout en-tête Meter qui ne contient pas "dont-report", signifie "Meter: do-report" ; cela rend le cas courant plus efficace.

Note : Un serveur d'origine qui utilise le mécanisme de temporisation du compte de touches pour limiter la période de collecte sur laquelle les comptes de touches sont obtenus devrait ajuster les valeurs du temporisateur dans les réponses qu'il envoie de telle sorte que toutes les réponses générées dans cette période atteignent leur fin de temporisation de mesure à la fin ou avant la fin de cette période.

Si le serveur d'origine envoie simplement une temporisation de mesure constante T avec chaque réponse pour une ressource, les rapports qu'il reçoit vont refléter l'activité sur une période dont la durée est entre T et N\*T (dans le pire des cas) où N est la profondeur maximum du sous arbre de mesure.

Pour la limitation d'usage

max-uses=NNN

établit une limite supérieure de NNN "utilisations" de la réponse, sans compter sa transmission immédiate au client final demandeur, pour tous les mandataires pris ensemble dans le sous arbre suivant.

max-reuses=NNN

établit une limite supérieure de NNN "réutilisations" de la réponse pour tous les mandataires pris ensemble dans le sous arbre suivant.

Lorsque un mandataire a épuisé son allocation d'utilisations ou de réutilisations pour une entrée d'antémémoire, il DOIT revalider l'entrée d'antémémoire (en utilisant une demande conditionnelle) avant de la retourner dans une réponse. (Le mandataire DEVRAIT utiliser ce message de revalidation pour envoyer un rapport d'usage, s'il en était demandé un et s'il est temps de l'envoyer. Voir aux paragraphes 3.4 et 3.5.)

Ces directives de réponse à Meter ne s'appliquent qu'à la réponse spécifique à laquelle elles sont rattachées.

Noter que la limite aux "utilisations" établie par la directive max-uses n'inclut pas l'utilisation de la réponse pour satisfaire la demande du client final qui a causé la demande du mandataire au serveur. Cette règle de comptage prend en charge la notion d'une prélecture (*prefetch*) à l'initiative de l'antémémoire : une antémémoire peut produire une demande prefetch, recevoir une réponse max-uses=0, mémoriser cette réponse, et retourner ensuite cette réponse (sans revalidation) lorsque un client fait une demande réelle pour la ressource. Cependant, chacune de ces réponse peut être utilisée au plus une fois de

cette façon, de sorte que le serveur d'origine conserve un contrôle précis du nombre d'utilisations réelles.

Un serveur NE DOIT PAS envoyer un en-tête Meter qui exigerait d'un mandataire qu'il fasse quelque chose qu'il n'a pas déjà offert de faire. Un mandataire qui reçoit une directive de réponse à Meter qui lui demande de faire quelque chose à quoi il ne s'est pas porté volontaire DEVRAIT ignorer cette directive.

Un mandataire qui reçoit un en-tête Meter dans une réponse DOIT soit y obéir, soit DOIT revalider l'entrée d'antémémoire correspondante sur tous les accès. (C'est à dire, si il choisit de ne pas obéir à l'en-tête Meter dans une réponse, il DOIT agir comme si la réponse incluait "Cache-control: s-maxage=0".)

Note : Un mandataire qui n'a pas envoyé l'en-tête Meter dans une demande pour la ressource en question, et qui ne s'est donc pas porté volontaire pour honorer les directives Meter dans une réponse, n'est pas obligé des les honorer. Si, dans cette situation, le serveur n'envoie pas un en-tête Meter dans une réponse, c'est une erreur de protocole. Cependant, sur la base du principe de robustesse, le mandataire peut choisir d'interpréter l'en-tête Meter comme une demande implicite d'inclure "Cache-control: s-maxage=0" lorsque il transmet la réponse, car cela préserve l'intention apparente du serveur.

Un mandataire qui reçoit l'en-tête Meter dans une demande ne peut l'ignorer que dans la mesure où cela est cohérent avec son propre devoir à l'égard du serveur du prochain bond. Si l'en-tête de demande Meter reçu n'est pas cohérent avec ce devoir, ou si aucun en-tête de demande Meter n'est reçu et si la réponse provenant du serveur de prochain bond ne demande aucune forme de mesure ou de limitation, le mandataire DOIT alors ajouter "Cache-control: s-maxage=0" à toute réponse qu'il transmet pour cette demande. (Un mandataire NE DEVRAIT PAS ajouter ou changer l'en-tête Expires ou la directive de contrôle d'antémémoire max-age.)

Par exemple, si le mandataire A reçoit une demande GET du mandataire B pour l'URL X avec "Connection: Meter", mais si la réponse en antémémoire du mandataire A pour l'URL n'inclut aucune directive Meter, le mandataire A peut alors ignorer l'offre de mesure du mandataire B.

Cependant, si le mandataire A a dit précédemment au serveur d'origine "Meter: wont-limit" (ce qui implique "will-report") et si la réponse en antémémoire contient "Meter: do-report", et que la demande du mandataire B comporte "Meter: wont-report", l'offre du mandataire B est alors incohérente avec le devoir du mandataire envers le serveur d'origine. Donc, dans ce cas, le mandataire A doit ajouter "Cache-control: s-maxage=0" lorsque il retourne la réponse en antémémoire au mandataire B, et doit ne pas inclure un en-tête Meter dans cette réponse.

Si un serveur ne veut pas utiliser le mécanisme Meter, et n'envisage pas de le faire dans un proche avenir, il peut envoyer cette directive :

wont-ask

recommande que le mandataire NE DEVRAIT PAS envoyer de directives Meter à ce serveur.

Le mandataire DEVRAIT se souvenir de ce fait pendant jusqu'à 24 heures. Cela évite presque toutes les surcharges inutiles pour les serveurs qui ne souhaitent pas utiliser ou prendre en charge l'en-tête Meter. (Cette directive implique aussi "dont-report".)

### 3.4 Transmission des rapports d'usage

Pour transmettre un rapport d'usage, un mandataire envoie l'en-tête Meter suivant dans une demande portant sur la ressource appropriée:

```
Meter: count=NNN/MMM
```

Le premier entier indique le compte des utilisations de l'entrée d'antémémoire depuis le dernier rapport ; le second entier indique le compte de réutilisations de l'entrée (voir au paragraphe 5.3 les règles du compte des utilisations et réutilisations) La transmission d'une directive "count" dans une demande sans autre directive Meter est aussi définie comme une transmission implicite de la directive "will-report-and-limit", pour optimiser le cas général. (Un mandataire qui ne veut pas honorer les limites d'usage enverrait "Meter: count=NNN/MMM, wont-limit" comme rapport.)

Noter que lorsque un mandataire transmet la demande d'un client et reçoit une réponse, la réponse que le mandataire envoie immédiatement au client demandeur n'est pas comptée comme "utilisation". C'est-à-dire, le compte rapporté est le nombre de fois que l'entrée d'antémémoire a été utilisée, et non le nombre de fois que cette réponse a été utilisée.

Un mandataire NE DEVRAIT PAS transmettre "Meter: count=0/0", car cela ne porte aucune information utile.

Les rapports d'usage DOIVENT toujours être transmis au titre d'une demande conditionnelle telle que GET ou HEAD) car les informations dans l'en-tête conditionnel (par exemple, If-Modified-Since ou If-None-Match) sont nécessaires pour le serveur d'origine pour savoir quelle instance d'une ressource est en cours de comptage. Les rapports d'usage transmis par les mandataires le long du sous arbre de mesure NE DOIVENT PAS changer le contenu de l'en-tête conditionnel, car autrement il en résulterait un compte incorrect.

Un rapport d'usage NE DOIT PAS être transmis au titre d'une demande transmise qui comporterait plusieurs étiquettes d'entité dans un en-tête If-None-Match ou If-Match.

Note : Un mandataire qui se porte volontaire pour faire le compte de touches (pour rapporter l'utilisation) doit compter les utilisations et les réutilisations. Il n'est pas possible de négocier le rapport de l'une mais pas de l'autre.

### 3.5 Quand envoyer les rapports d'usage

Un mandataire qui a offert d'envoyer des rapports d'usage à son parent dans le sous arbre de mesure DOIT envoyer un rapport d'usage dans chacune de ces situations :

1. Lorsque il transmet un GET conditionnel sur l'instance de ressource au nom d'un ses clients (si le GET est conditionnel sur au moins une étiquette d'entité).
2. Lorsque il transmet un HEAD conditionnel sur l'instance de ressource au nom d'un de ses clients.
3. Lorsque il doit générer un GET conditionnel pour satisfaire la demande d'un client parce que la limite max-uses a été dépassée.
4. À expiration d'une temporisation de mesure associée à une entrée d'antémémoire qui a un compte de touches différent de zéro.
5. Lorsque il retire l'entrée correspondante de compte de touches différente de zéro de sa mémoire pour une raison quelconque, y compris:
  - parce que le mandataire a besoin de l'espace mémoire pour une autre entrée de compte de touches,
  - parce que le mandataire n'est pas capable de mémoriser plus d'une réponse par ressource, et qu'une demande transmise au nom d'un client a résulté en la réception d'une nouvelle réponse (avec une étiquette d'entrée différente ou une autre heure de dernière modification).

Noter qu'une antémémoire pourrait continuer de mémoriser les informations de compte de touches même après avoir supprimé le corps de la réponse, de sorte qu'il ne soit pas nécessaire de faire rapport du compte de touches lors de la suppression du corps ; il est seulement nécessaire de faire rapport si le mandataire est sur le point "d'oublier" une valeur différente de zéro.

(Le paragraphe 5.3 explique comment le compte de touches devient zéro ou différent de zéro.)

Si le rapport d'usage est sur le point d'être envoyé parce que le mandataire va retirer l'entrée de compte de touches de sa mémoire, ou à cause de l'arrivée à expiration d'une temporisation de mesure :

- le mandataire DOIT envoyer le rapport au titre d'une demande HEAD conditionnelle sur l'instance de ressource,
- le mandataire n'est pas obligé de réessayer la demande HEAD si elle échoue (c'est un concept au mieux). Pour améliorer la précision, le mandataire DEVRAIT cependant réessayer les demandes HEAD qui ont échoué, sous réserve des contraintes de ressources.
- le mandataire n'est pas obligé d'enchaîner d'autres opérations à l'achèvement de cette demande.

Note : Une mise en œuvre de mandataire est vivement encouragée à faire un lot de plusieurs rapports fondés sur HEAD au même serveur, lorsque possible, sur une seule connexion persistante, pour réduire la redondance du réseau autant que possible. Cela peut impliquer un algorithme complexe pour programmer la suppression des entrées de compte de touches.

Si le compte d'usage est envoyé parce que une demande arrivante porte aussi une directive "count", le mandataire DOIT combiner ses propres comptes d'utilisation et de réutilisation (éventuellement à zéro) avec les comptes arrivants, puis tenter de transmettre la demande.

Cependant, le mandataire n'est pas obligé de transmettre une demande arrivante avec une directive "count", pourvu que :

- il puisse répondre à la demande en utilisant une réponse en antémémoire, en conformité avec les autres exigences de la spécification HTTP,
- une telle réponse n'excède pas une limite max-uses,
- il ne soit pas obligé de transmettre la demande à cause de l'arrivée à expiration du temporisateur de mesures.

Si une demande arrivante porte une directive "count", et si le mandataire n'a plus d'entrée d'antémémoire pour la ressource,

le mandataire DOIT transmettre la directive "count". (C'est-à-dire, dans tous les cas, ce qu'un mandataire qui n'a pas une entrée d'antémémoire convenable ferait normalement pour toute demande valide qu'il reçoit.)

### 3.6 Subdivision des limites d'usage

Lorsque un serveur d'origine spécifie une limite d'usage, un mandataire qui se trouve dans le sous arbre de mesures peut subdiviser cette limite parmi ses enfants dans le sous arbre à sa convenance.

Par exemple, considérons la situation avec deux mandataires P1 et P2, chacun d'eux utilisant le mandataire P3 comme moyen d'atteindre le serveur d'origine S. Imaginons que S envoie à P3 une réponse avec

```
Meter: max-uses=10
```

Les mandataires utilisent cette réponse pour satisfaire le client final actuel qui fait la demande. La directive max-uses dans cet exemple permet de combiner ensemble P1, P2, et P3 pour satisfaire dix utilisations supplémentaires de client final (GET inconditionnels) pour la ressource.

La présente spécification n'établit pas de contrainte sur la façon dont P3 divise son allocation entre lui-même et les autres mandataires. Par exemple, P3 pourrait garder toute l'allocation de max-use pour lui-même. Dans ce cas, il transmettrait la réponse à P1 et/ou à P2 avec

```
Meter: max-uses=0
```

P3 pourrait aussi diviser également l'allocation entre P1 et P2, n'en gardant rien pour lui-même (ce qui peut être le bon choix si P3 a peu ou pas d'autres clients). Dans ce cas, il pourrait envoyer

```
Meter: max-uses=5
```

au mandataire (P1 ou P2) qui a fait la demande initiale, puis enregistrer le reste de l'allocation dans une structure de données internes qu'il "doit" à l'autre mandataire.

Noter que cette liberté de choisir la valeur de max-uses s'applique aussi bien au serveur d'origine. Il n'est pas exigé qu'un serveur d'origine envoie la même valeur de max-uses à toutes les antémémoires. Par exemple, cela pourrait avoir un sens que d'envoyer "max-uses=2" la première fois qu'on voit une antémémoire, puis de doubler la valeur (jusqu'à une limite maximum) chaque fois qu'on obtient un "compte d'utilisations" de cette antémémoire. L'idée est que plus vite une antémémoire utilise son quota de max-use, plus il est probable qu'elle va rapporter une valeur de compte d'utilisation avant de supprimer l'entrée d'antémémoire. Aussi, des comptes d'utilisation élevés et fréquents impliquent de bénéficier d'une efficacité élevée correspondante de permettre la mise en antémémoire.

Là encore, les détails d'une telle heuristique sortiraient du domaine d'application de la présente spécification.

## 4 Analyse

Cette section comporte une analyse informelle de plusieurs aspects du compte de touches :

1. la précision des résultats lorsque appliqués au comptage des utilisateurs (paragraphe 4.1),
2. le problème du comptage des usagers dont le navigateur ne comporte pas d'antémémoire, comme les ordinateurs en réseau (paragraphe 4.2),
3. les délais imposés aux "chemins critiques" pour les opérations de HTTP (paragraphe 4.3).

### 4.1 Approximation de la précision du comptage des usagers

Pour de nombreux opérateurs de services (mais pas tous) le seul aspect le plus important du flux de demandes est le nombre d'usagers distincts qui ont récupéré une entité particulière au sein d'une période donnée (par exemple, durant une certaine journée). Le mécanisme de compte de touches est conçu pour fournir à un serveur d'origine une approximation du nombre d'usagers qui font référence à une certaine ressource. L'intention du concept est que la précision de cette approximation est cohérente avec les objectifs de simplicité et de mise en œuvre facultative.

Presque tous les usagers de la Toile utilisent des logiciels clients qui conservent des antémémoires locales, et l'état de l'art de la technologie des antémémoires locales est assez efficace. (Le paragraphe 4.2 expose le cas des antémémoires clients qui sont petites ou non existantes.) Donc, en supposant une antémémoire client efficace et persistante, chaque usager individuel qui restitue une entité fait exactement une demande GET qui résulte en réponses 200 ou 203, ou une réponse 206

qui comporte le premier octet de l'entité. Si une antémémoire mandataire conserve et fait rapport d'un compte d'utilisations précis de telles restitutions, le rapport de ce compte d'utilisations va approximer de près le nombre d'usagers distincts qui ont restitué l'entité.

Il y a certaines circonstances dans lesquelles cette approximation peut échouer. Par exemple, si une entité reste dans une antémémoire mandataire pendant beaucoup plus longtemps qu'elle ne persiste dans l'antémémoire normale de client, et si les usagers re-référencent souvent l'entité, ce schéma va alors tendre à sur estimer le nombre d'usagers. Ou, si la politique de gestion d'antémémoire mise en œuvre dans les antémémoires normales de client est biaisée en conservant certaines sortes d'entités fréquemment re-référencées (comme de très grosses images) les comptes d'utilisation rapportés vont tendre à surestimer le compte des usagers pour de telles entités.

Les analyses des journaux des navigateurs ont montré que lorsque un usager revisite une ressource, c'est presque toujours très peu de temps après la précédente visite, presque toujours avec moins de huit autres références intercalées [11]. Bien que ce résultat ne soit sans doute pas d'application universelle, il implique que presque toutes les réutilisations vont impacter l'antémémoire du client final, et ne seront pas vues comme des GET inconditionnels par une antémémoire mandataire.

Les mécanismes existants (HTTP/1.0) de "court-circuit d'antémémoire" pour compter les usagers distincts vont certainement surestimer le nombre d'usagers derrière un mandataire, car il ne fournit pas de façon fiable pour distinguer entre la demande initiale d'un usager et les demandes répétées suivantes qui pourraient avoir été des GET conditionnels, si aucun court-circuit d'antémémoire n'avait été employé. Le dispositif "Cache-control: s-maxage=0" de HTTP/1.1 permet bien la séparation du compte d'utilisations et des comptes de réutilisations, pourvu qu'aucune antémémoire de mandataire HTTP/1.0 n'intervienne.

Noter que si il y a un doute sur la validité des résultats du compte de touches d'un certain ensemble de ressources, le serveur peut employer les techniques de court-circuit d'antémémoire pour de courtes périodes, pour établir un point de départ pour la validation des résultats du compte de touches. Diverses approches de ce problèmes sont exposées dans un article de James Pitkow [9].

## 4.2 Qu'en est-il des ordinateurs en réseau ?

L'analyse du paragraphe 4.1 supposait que "presque tous les utilisateurs de la Toile" ont des antémémoires client. Si le modèle de l'ordinateur en réseau (NC, *Network Computer*) devient populaire, cette hypothèse peut cependant se révéler fausse : la plupart des NC proposés n'ont pas de disque de mémorisation, et relativement peu de RAM. De nombreux assistants numériques personnels (PDA, *Personal Digital Assistant*), qui ont parfois un accès réseau, ont des contraintes similaires. De tels systèmes clients ne peuvent faire que peu ou pas du tout de mise en antémémoire de réponses HTTP. Cela signifie qu'un seul usager peut très bien générer de nombreux GET inconditionnels qui donnent la même réponse à partir d'une antémémoire mandataire.

On notera d'abord que dans le présent document, le concept de compte de touche, même avec de tels clients, donne une approximation qui n'est pas pire que ce qui est disponible avec HTTP/1.1 non modifié : les comptes qu'un mandataire va retourner à un serveur d'origine vont représenter exactement le nombre de demandes que le mandataire aurait transmis au serveur, si celui-ci avait simplement spécifié "Cache-control: s-maxage=0".

Cependant, il est possible d'améliorer la précision de ces comptes de touches en utilisant une heuristique chez le mandataire. Par exemple, le mandataire pourrait noter l'adresse IP du client, et ne compter qu'un GET par adresse client et par réponse. Ce n'est pas parfait : par exemple, cela ne distingue pas entre les NC et certaines autres sortes d'hôtes. Le mandataire pourrait aussi utiliser une heuristique selon laquelle seuls les clients qui n'envoient jamais un GET conditionnel devraient être traités de cette façon, bien qu'on ne soit pas tout à fait certain que les NC n'enverront jamais de GET conditionnels.

Comme la solution à ce problème paraît exiger une heuristique fondée sur le comportement réel des NC (ou peut-être un nouveau dispositif du protocole HTTP qui permettrait la détection sans ambiguïté des clients sans antémémoire) il apparaît prématuré de spécifier une solution.

## 4.3 Analyse des délais du chemin critique

Dans les systèmes (comme la Toile) où la latence est un problème, il y a généralement une arborescence d'étapes qui dépendent les unes des autres, d'une façon telle que le résultat final ne peut pas être envisagé tant que tous ses prédécesseurs n'ont pas été parcourus. Comme la structure de l'arbre admet un certain parallélisme, il n'est pas nécessaire d'ajouter le temps de chaque étape pour découvrir la latence du processus entier. Mais tous les chemins de cet arbre de

dépendance ne peuvent pas être mis en parallèle, et le plus long de ces chemins est celui dont la longueur (en secondes) détermine la latence globale. C'est le "chemin critique", parce que aussi court que soit tout autre chemin, il ne peut pas changer la latence globale pour le résultat final.

Si on voit le résultat final, pour une demande sur la Toile, comme la restitution d'une page sur un navigateur, ou une autre action sur le résultat d'une demande, on voit clairement que quelques délais d'aller retour de réseau (par exemple, échanger des paquets TCP SYN si la connexion n'existe pas encore) sont sur le chemin critique. Ce concept de compte de touches ajoute bien quelques allers retours pour le rapport des comptes différents de zéro lorsque une entrée d'antémémoire est supprimée, mais, par conception, ils sont en dehors de tout chemin critique : ils peuvent être faits en parallèle avec toutes les autres opérations, et n'exigent que des efforts "au mieux", de sorte qu'un mandataire n'a pas à mettre en série d'autres opérations avec leur réussite ou leur échec.

On voit clairement que tout ce qui change l'utilisation du réseau (en l'augmentant ou en la diminuant) peut affecter indirectement la latence perçue par l'utilisateur. Notre attente est qu'en moyenne le compte de touches va réduire la charge et donc même ses effets indirects ne devraient pas ajouter d'allers retours de réseau dans un chemin critique. Mais il peut y avoir quelques instances spécifiques où l'ajout d'opérations sur le chemin non critique (en particulier, les rapports d'usage sur suppression d'entrée d'antémémoire) retardent une opération sur un chemin critique. C'est un problème inévitable dans les réseaux de datagrammes.

## 5 Spécification

### 5.1 Spécification de l'en-tête et des directives Meter

Le champ d'en-tête général Meter est utilisé pour :

- négocier l'utilisation du compte de touches et de la limitation d'usage entre les serveurs d'origine et les antémémoires mandataires,
- rapporter les comptes d'utilisation et de réutilisation.

La mise en œuvre de l'en-tête Meter est facultative à la fois pour les mandataires et les serveurs d'origine. Cependant, tout mandataire qui transmet l'en-tête Meter dans une demande DOIT mettre en œuvre toutes les exigences de la présente spécification, sans exception ni amendement.

L'en-tête Meter DOIT toujours être protégé par un en-tête Connection. Un mandataire qui ne met pas en œuvre l'en-tête Meter NE DOIT PAS le passer à travers un autre système (voir au paragraphe 5.5 comment un mandataire qui ne met pas en antémémoire peut se conformer à cette spécification). Si un en-tête Meter est reçu dans un message dont la version est inférieure à HTTP/1.1, il DOIT être ignoré (parce qu'il est clair qu'il est passé par un mandataire qui ne met pas Meter en œuvre).

Un mandataire qui a reçu une réponse avec une version inférieure à HTTP/1.1, et donc provenant d'un serveur (ou autre mandataire) qui ne met pas en œuvre l'en-tête Meter, NE DEVRAIT PAS envoyer de directives de demande Meter à ce serveur, parce que ce serait simplement gâcher la bande passante. Cette recommandation ne s'applique que si le mandataire est actuellement en train de compter les touches ou de limiter l'usage de réponses provenant de ce serveur. Si le mandataire reçoit une réponse HTTP/1.1 ou supérieure d'un tel serveur, il devrait cesser sa suppression des directives Meter.

Tous les mandataires qui envoient l'en-tête Meter DOIVENT adhérer au concept de "sous arbre de mesure" décrit à la section 3.

```
Meter = "Meter" ":" 0#meter-directive
```

```
meter-directive = meter-request-directive
                  | meter-response-directive
                  | meter-report-directive
```

```
meter-request-directive = "will-report-and-limit"
                          | "wont-report"
                          | "wont-limit"
```

```
meter-report-directive =
  | "count" "=" 1*DIGIT "/" 1*DIGIT
```

```
meter-response-directive =
```

```

"max-uses" "=" 1*DIGIT
| "max-reuses" "=" 1*DIGIT
| "do-report"
| "dont-report"
| "timeout" "=" 1*DIGIT
| "wont-ask"

```

Une directive de demande de mesures ou une directive de rapport de mesures ne peut apparaître que dans un message de demande HTTP. Une directive réponse de mesure ne peut apparaître que dans une directive de réponse HTTP.

Un en-tête Meter vide dans une demande signifie "Meter: will-report-and- limit". Un en-tête Meter vide dans une réponse, ou toute autre réponse incluant un ou plusieurs en-têtes Meter sans la directive "dont-report" ou "wont-ask" implique "Meter: do-report".

La signification des directives demande de mesure est la suivante :

will-report-and-limit

indique que le mandataire est volontaire et capable de retourner des rapports d'usage et va respecter toutes les limites d'usage.

wont-report

indique que le mandataire va respecter les limites d'usage mais n'enverra pas de rapports d' usage.

wont-limit

indique que le mandataire ne va pas respecter les limites d'usage mais va envoyer les rapports d'usage.

Un mandataire qui ne veut pas respecter les limites d'usage ni envoyer de rapports d'usage NE DOIT PAS transmettre d'en-tête Meter dans la demande.

La signification des directives de rapport de mesures est la suivante :

count "=" 1\*DIGIT "/" 1\*DIGIT

Les deux chaîne de chiffres codent des entiers décimaux. Le premier entier indique le compte d'utilisations de l'entrée d'antémémoire depuis le dernier rapport ; le second entier indique le compte de réutilisations de l'entrée.

Le paragraphe 5.3 spécifie les règles de comptage.

La signification des directives de réponse de mesure est la suivante :

max-uses "=" 1\*DIGIT

fixe une limite supérieure au nombre "d'utilisations" de la réponse, sans compter sa transmission immédiate au client final demandeur, pour tous les mandataires pris ensemble dans le sous arbre suivant.

max-reuses "=" 1\*DIGIT

établit une limite supérieure du nombre de "réutilisations" de la réponse pour tous les mandataires pris ensemble dans le sous arbre suivant.

do-report

spécifie que le mandataire DOIT envoyer des rapports d'usage au serveur.

dont-report

spécifie que le mandataire NE DEVRAIT PAS envoyer de rapports d'usage au serveur.

timeout "=" 1\*DIGIT

établit une temporisation de mesures du nombre spécifié de minutes (et non de secondes) après la génération de cette réponse (comme indiqué par son en-tête "Date"). Si le mandataire a un compte de touches différent de zéro pour cette réponse lors de l'arrivée à expiration de ce temporisateur, il DOIT envoyer un rapport au serveur à ce moment ou avant. Les temporisations devraient être mises en œuvre avec une précision de plus ou moins une minute. Elle implique "do-report".

wont-ask

spécifie que le mandataire NE DEVRAIT PAS envoyer d'en-têtes Meter au serveur. Le mandataire devrait oublier cet avis après une période de pas plus de 24 heures.

Le paragraphe 5.3 spécifie les règles de comptage, et en particulier spécifie une interprétation non évidente de la valeur de max-uses.

## 5.2 Abréviations pour les directives Meter

Pour permettre le codage le plus efficace possible des en-têtes Meter, on a défini des formes abrégées de toutes les directives Meter. Elles sont sémantiquement exactement équivalentes à leur contrepartie non abrégée. Tous les systèmes qui mettent en œuvre l'en-tête Meter DOIVENT mettre en œuvre les deux formes abrégée et non abrégée. Les mises en œuvre DEVRAIENT utiliser les formes abrégées en utilisation normale.

Les formes abrégées de directive Meter sont indiquées ci-dessous, avec les littéraux non abrégés correspondants dans les commentaires :

```
Abb-Meter = "Meter" ":" 0#abb-meter-directive
```

```
abb-meter-directive = abb-meter-request-directive
                    | abb-meter-response-directive
                    | abb-meter-report-directive
```

```
abb-meter-request-directive =
    "w"                ; "will-report-and-limit"
    | "x"                ; "wont-report"
    | "y"                ; "wont-limit"
```

```
abb-meter-report-directive =
    | "c" "=" 1*DIGIT "/" 1*DIGIT          ; "count"
```

```
abb-meter-response-directive =
    "u" "=" 1*DIGIT          ; "max-uses"
    | "r" "=" 1*DIGIT        ; "max-reuses"
    | "d"                    ; "do-report"
    | "e"                    ; "dont-report"
    | "t" "=" 1*DIGIT        ; "timeout"
    | "n"                    ; "wont-ask"
```

Note : bien que la règle BNF Abb-Meter soit définie à part de la règle Meter, on peut librement mêler les directives Meter abrégées et non abrégées dans le même en-tête.

## 5.3 Règles de comptage

Note : On se souviendra que les comptes de touches et les comptes d'usage sont associés à des réponses individuelles, et non à des ressources. Une entrée d'antémémoire qui, sur sa durée de vie, contient plus d'une réponse n'est aussi pas une "réponse", dans ce sens particulier.

Soit R une réponse en antémémoire, et soit V la valeur de l'URI de demande qui va sélectionner les en-têtes de demandes (si il en est, voir le paragraphe 14.43 de la spécification HTTP/1.1 [4]) qui vont retenir R si elle est contenue dans une demande. On définit une "utilisation" de R comme survenant lorsque le mandataire retourne la copie de R qu'il a en mémoire dans une réponse avec un des codes d'état suivants : un état 200 (OK), un état 203 (Informations qui ne sont pas d'autorité), ou un état 206 (Contenu partiel) lorsque la réponse contient l'octet n° 0 de l'entité (voir au paragraphe 5.4 un exposé sur les demandes Range).

Note : Lorsque un mandataire transmet la demande d'un client et reçoit une réponse, la réponse que le mandataire envoie immédiatement au client demandeur n'est pas comptée comme une "utilisation". C'est-à-dire, le compte objet du rapport est le nombre de fois que l'entrée d'antémémoire a été utilisée, et non le nombre de fois que la réponse a été utilisée.

On définit une "réutilisation" de R comme survenant lorsque le mandataire répond à une demande qui choisit R avec un état 304 (Non modifié) sauf si cette demande est une demande Range (*gamme*) qui ne spécifie pas l'octet n° 0 de l'entité.

### 5.3.1 Règles de comptage pour les touches

Un mandataire qui participe au compte de touches pour une réponse en antémémoire R entretient deux compteurs, CU et CR, associés à R. Lorsque un mandataire mémorise R pour la première fois dans son antémémoire, il règle CU et CR tous deux à 0 (zéro). Lorsque une demande de client ultérieure résulte en une "utilisation" de R, le mandataire incrémente CU. Lorsque une demande de client suivante résulte en une "réutilisation" de R, le mandataire incrémente CR. Lorsque une demande de client suivante qui choisit R (c'est-à-dire, incluant V) comporte une directive Meter "count", le mandataire incrémente CU et CR en utilisant les valeurs correspondantes dans la directive.

Lorsque le mandataire envoie une demande choisissant R (c'est-à-dire, incluant V) sur le serveur entrant, il inclut une directive Meter "count" avec le CU et CR actuels comme valeurs de paramètres. Si la demande était causée par la réception par le mandataire d'une demande provenant d'un client, à réception de la réponse du serveur, le mandataire règle CU et CR au nombre d'utilisations et réutilisations respectives qui ont pu survenir pendant que la demande était en cours. (Ces chiffres seront très vraisemblablement zéro, mais ce n'est pas certain.) Si la demande du mandataire était un rapport final fondé sur HEAD, il n'a plus besoin de conserver les valeurs CU et CR, mais il peut aussi les régler au nombre d'utilisations et réutilisations intervenues et les conserver.

### 5.3.2 Règles de comptage pour la limitation d'usage

Un mandataire qui participe à la limitation d'usage pour une réponse R conserve l'un et/ou l'autre des compteurs TU et TR, selon le cas approprié, pour cette ressource. TU et TR sont incrémentés de la même façon que, respectivement, CU et CR. Cependant, TU n'est mis à zéro que à réception d'une directive Meter "max-uses" pour cette réponse (incluant la réception initiale). De même, TR n'est mis à zéro qu'à réception d'une directive Meter "max-reuses" pour cette réponse.

Un mandataire qui participe à la limitation d'usage pour une réponse R mémorise aussi les valeurs MU et/ou MR associées à R. Lorsque il reçoit une réponse qui ne comporte qu'une valeur de max-uses, il règle MU à cette valeur et MR à l'infini. Lorsque il reçoit une réponse comportant seulement une valeur max-reuses, il règle MR à cette valeur et MU à l'infini. Lorsque il reçoit une réponse incluant à la fois des valeurs pour max-uses et max-reuses, il règle MU et MR à ces valeurs respectives. Lorsque il reçoit une réponse ultérieurs ne comportant de valeur ni pour max-uses ni pour max-reuses, il les règle toutes deux à l'infini.

Si un mandataire participant à la limitation d'usage pour une réponse R reçoit une demande qui causerait une "utilisation" de R, et que  $TU > MU$ , il DOIT transmettre la demande au serveur. Si il reçoit une demande qui causerait une "réutilisation" de R, et que  $TR \geq MR$ , il DOIT transmettre la demande au serveur. Si (dans l'un ou l'autre cas) le mandataire a déjà transmis une demande précédente au serveur et en attend la réponse, il devrait différer tout traitement de la nouvelle demande jusqu'à l'arrivée de la réponse (ou la fin de sa temporisation) ; il NE DEVRAIT PAS avoir deux demandes de revalidation en cours à la fois qui choisissent la même réponse, sauf si ce sont des demandes Range qui choisissent des sous gammes différentes.

Il y a un cas particulier de ces règles pour la directive "max-uses" : si le mandataire reçoit une réponse avec "max-uses=0" et si il ne la transmet pas à son client demandeur, le mandataire devrait établir un fanion PF associé à R. Si R est vrai, alors quand une demande arrive alors que si  $TU \geq MU$ , si le fanion PF est mis, la demande n'a alors pas besoin d'être transmise au serveur (pourvu que ce ne soit pas requis par d'autres règles de mise en antémémoire). Cependant, le fanion PF DOIT être ôté à chaque utilisation de la réponse.

Note : Le fanion "PF" est ainsi nommé parce que ce dispositif n'est utile que pour les antémémoires qui pourraient produire une demande de "prélecture" (*prefetch*) avant qu'un client réel ne demande la réponse. Un mandataire qui ne met pas en œuvre la prélecture n'a pas besoin de mettre en œuvre le fanion PF.

### 5.3.3 Les algorithmes équivalents sont admis

Tout autre algorithme qui exhibe le même comportement externe (c'est-à-dire, génère exactement les mêmes demandes du mandataire au serveur) que celui décrit dans ce paragraphe est explicitement permis.

Note : Dans la plupart des cas, TU ne sera pas égal à CU, et TR sera égal à CR. Les deux seuls cas où ils pourraient différer sont :

1. Le mandataire produit une demande non conditionnelle pour la ressource en utilisant V, alors que TU et/ou TR sont différents de zéro, et la réponse du serveur comporte une nouvelle directive "max-uses" et/ou "max-reuses" (mettant donc à zéro TU et/ou TR, mais pas CU et CR).
2. Le mandataire produit une demande conditionnelle rapportant les comptes de touches (et mettant donc à zéro CU et CR, mais pas TU ou TR) mais la réponse du serveur n'inclut pas les nouvelles directives "max-uses" et/ou "max-reuses".

Pour résoudre le premier cas, le mandataire a plusieurs options de mise en œuvre :

- toujours mémoriser TU et TR séparément de CU et CR,

- créer des copies "dans l'ombre" de TU et TR lorsque survient cette situation (analogue à "copie sur écriture"),
- générer un rapport d'usage fondé sur HEAD lorsque la demande non conditionnelle est envoyée (ou quand "max-uses=0" est reçu) causant la mise à zéro de CU et CR (analogue d'une certaine façon à une instruction "barrière de mémoire").

Dans le second cas, le serveur a implicitement supprimé la ou les limites d'usage sur la réponse (en réglant MU et/ou MR à l'infini) et donc le fait que, disons, TU soit différent de CU n'est pas significatif.

Note : Il est aussi possible d'éliminer le fanion PF en envoyant des demandes supplémentaires de rapport d'usage fondées sur HEAD, mais nous le déconseillons; il est préférable d'allouer un bit supplémentaire par entrée plutôt que de transmettre des demandes supplémentaires.

#### 5.4 Règles de comptage : interaction avec les demandes Range

HTTP/1.1 permet à un client de demander une sous gamme d'une ressource. Un client peut finir par produire plusieurs demandes dont le résultat net sera la réception d'une copie de la ressource. Pour l'uniformité des résultats vus par les serveurs d'origine, les mandataires doivent observer une règle pour le comptage de ces références, bien qu'on ne sache pas trop si une règle génère des résultats précis dans tous les cas.

La règle établie dans la présente spécification est que les mandataires ne comptent comme utilisation ou réutilisation que les demandes Range qui résultent en le retour de l'octet n° 0 de la ressource. Les raisons de cette règle sont que dans la plupart des cas, un client final va restituer le début de toute ressource à laquelle il fait référence, et qu'il va rarement restituer une portion plus d'une fois. Donc, cette règle paraît satisfaire aux objectifs d'une approximation "au mieux".

#### 5.5 Mise en œuvre par les mandataires qui ne mettent pas en antémémoire

Un mandataire qui ne met pas en antémémoire peut participer au sous arbre de mesure; c'est vivement recommandé.

Un mandataire qui ne met pas en antémémoire (HTTP/1.1 ou supérieur) et qui participe au sous arbre de mesures DEVRAIT transmettre les en-têtes Meter sur les demandes et sur les réponses, avec les en-têtes Connection appropriés.

Si un mandataire qui ne met pas en antémémoire transmet des en-têtes Meter, il DOIT se conformer aux restrictions suivantes :

1. Si le mandataire transmet des en-têtes Meter en réponse, une telle réponse NE DOIT PAS être retournée à une autre demande que celle qui l'a suscitée.
2. Une fois qu'un mandataire qui ne met pas en antémémoire commence à transmettre des en-têtes Meter, il ne devrait pas arrêter arbitrairement de les transmettre (sinon des rapports vont se perdre).

Un mandataire qui met en antémémoire certaines réponses et pas d'autres, quelle qu'en soit la raison, peut choisir de mettre en œuvre l'en-tête Meter comme un mandataire qui met en antémémoire pour les réponses qu'il met en antémémoire, et comme un mandataire qui ne met pas en antémémoire pour les réponses qu'il ne met pas en antémémoire, pour autant que son comportement externe par rapport à toute réponse particulière soit pleinement cohérent avec la présente spécification.

#### 5.6 Mise en œuvre par des antémémoires coopératives

Plusieurs mises en œuvre d'antémémoire de HTTP, dont la plus notable est l'antémémoire Harvest/Squid [2], créent des arrangements coopératifs entre plusieurs antémémoires. Si de telles antémémoires utilisent un protocole autre que HTTP pour communiquer entre elles, tel que le protocole d'antémémoires Internet (ICP, *Internet Cache Protocol*) [12], et si elles mettent en œuvre l'en-tête Meter, elles DOIVENT alors agir pour s'assurer que leur coopération ne viole pas les intentions de la présente spécification.

En particulier, si un membre d'un groupe d'antémémoires coopérantes se met d'accord avec un serveur pour compter les touches d'une réponse particulière, puis passe cette réponse via un protocole non HTTP à une seconde antémémoire dans le groupe, les antémémoires DOIVENT s'assurer que le serveur qui a demandé la mesure reçoive les rapports qui tiennent compte de façon appropriée de toutes les utilisations ou réutilisations faites par la seconde antémémoire. De même, si la première antémémoire est d'accord pour la limitation d'usage de la réponse, le nombre total d'utilisations par le groupe d'antémémoires DOIT être limité au nombre accepté.

## 6 Exemples

### 6.1 Exemple d'un ensemble d'échanges complets

Cet exemple montre comment le protocole est destiné à être utilisé la plupart du temps : pour le compte de touches sans limitation d'usage. Les corps d'entité sont omis.

Un client envoie une demande à un mandataire :

```
GET http://foo.com/bar.html HTTP/1.1
```

Le mandataire transmet la demande au serveur d'origine :

```
GET /bar.html HTTP/1.1
Host: foo.com
Connection: Meter
```

offrant donc (implicitement) "will-report-and-limit".

Le serveur répond au mandataire :

```
HTTP/1.1 200 OK
Date: Fri, 06 Dec 1996 18:44:29 GMT
Cache-control: max-age=3600
Connection: meter
Etag: "abcde"
```

demandant donc (implicitement) "do-report" (mais sans demander la limitation d'usage).

Le mandataire répond au client :

```
HTTP/1.1 200 OK
Date: Fri, 06 Dec 1996 18:44:29 GMT
Etag: "abcde"
Cache-control: max-age=3600, proxy-mustcheck
Age: 1
```

Comme le mandataire ne sait pas si son client est un système d'extrémité, ou un mandataire qui ne fait pas les mesures, il ajoute la directive "proxy-mustcheck".

Un autre client demande bientôt la ressource :

```
GET http://foo.com/bar.html HTTP/1.1
```

et le mandataire envoie la même réponse qu'il a envoyée à l'autre client, sauf (peut-être) la valeur de Age.

Après une heure, un troisième client demande la réponse :

```
GET http://foo.com/bar.html HTTP/1.1
```

Mais maintenant, le max-age de la réponse a été dépassé, de sorte que le mandataire revalide la réponse avec le serveur d'origine :

```
GET /bar.html HTTP/1.1
If-None-Match: "abcde"
Host: foo.com
Connection: Meter
Meter: count=1/0
```

remplissant donc simultanément son devoir de validation de la réponse et de faire rapport de l'utilisation qui n'avait pas été transmise.

Le serveur d'origine répond :

```
HTTP/1.1 304 Not Modified
Date: Fri, 06 Dec 1996 19:44:29 GMT
Cache-control: max-age=3600
Etag: "abcde"
```

de sorte que le mandataire peut utiliser la réponse originale pour répondre au nouveau client ; le mandataire met aussi à zéro le compte d'utilisations qu'il associe à cette réponse.

Un autre client demande bientôt la ressource :

```
GET http://foo.com/bar.html HTTP/1.1
```

et le mandataire envoie la réponse appropriée.

Après quelques heures, le mandataire décide de supprimer l'entrée d'antémémoire. Lorsque il le fait, il envoie au serveur d'origine :

```
HEAD /bar.html HTTP/1.1
If-None-Match: "abcde"
Host: foo.com
Connection: Meter
Meter: count=1/0
```

rapportant qu'une utilisation de plus de la réponse a été satisfaite à partir de l'antémémoire.

## 6.2 Protection contre les mandataires HTTP/1.0

Un serveur d'origine qui ne veut pas que les antémémoires HTTP/1.0 mémorisent du tout la réponse, et qui veut que les clients HTTP/1.0 de systèmes d'extrémité génèrent des GET supplémentaires (qui seront transmis par les mandataires HTTP/1.0) pourrait envoyer ceci comme réponse :

```
HTTP/1.1 200 OK
Cache-control: max-age=3600
Connection: meter
Etag: "abcde"
Expires: Sun, 06 Nov 1994 08:49:37 GMT
```

Les antémémoires HTTP/1.0 vont voir l'ancien en-tête Expires, mais les antémémoires HTTP/1.1 verront la directive max-age et vont ignorer Expires.

Note : Bien que la plupart des mises en œuvre majeures de mandataire HTTP/1.0 observent l'en-tête Expires, il est possible que certaines ne le fassent pas. L'utilisation de l'en-tête Expires pour empêcher la mise en antémémoire par les mandataires HTTP/1.0 pourrait n'être pas entièrement fiable.

## 6.3 Exemples plus élaborés

Voici une demande d'un mandataire qui veut faire le compte de touches mais ne veut pas faire de limitation d'usage :

```
GET /bar.html HTTP/1.1
Host: foo.com
Connection: Meter
Meter: wont-limit
```

On a ici une réponse d'un serveur d'origine qui ne veut pas compter les touches, mais veut limiter les "utilisations" à 3, et les "réutilisations" à 6 :

```
HTTP/1.1 200 OK
Cache-control: max-age=3600
Connection: meter
```

```
Etag: "abcde"
Expires: Sun, 06 Nov 1994 08:49:37 GMT
Meter: max-uses=3, max-reuses=6, dont-report
```

Voici le même exemple avec les noms abrégé de directive Meter :

```
HTTP/1.1 200 OK
Cache-control: max-age=3600
Connection: meter
Etag: "abcde"
Expires: Sun, 06 Nov 1994 08:49:37 GMT
Meter:u=3,r=6,e
```

## 7 Interactions avec la négociation de contenu

Cette section décrit deux aspects de l'interaction entre compte de touches et ressources à "contenu négocié":

1. traitement des réponses portant un en-tête Vary (paragraphe 7.1).
2. traitement des réponses qui utilisent le mécanisme proposé de négociation transparente de contenu (paragraphe 7.2).

### 7.1 Traitement des réponses portant un en-tête Vary

On devrait tenir des comptes séparés pour chaque combinaison des en-têtes désignés dans l'en-tête Vary pour l'URI de demande (ce que [4] appelle "les en-têtes de demande sélectifs") même si ils se transposent en la même étiquette d'entité. Cette règle a pour effet de compter les touches sur chaque variante, si il y a plusieurs variantes disponibles d'une page.

Note : Cette interaction entre Vary et les directives de comptage de touches permet au serveur d'origine beaucoup de souplesse pour spécifier comment les touches devraient être comptées. Par nature, le serveur d'origine utilise le mécanisme Vary pour diviser les demandes sur une ressource en catégories arbitraires, sur la base des en-têtes des demandes. (On appellera ces catégories "schéma de demandes".) Comme un mandataire conserve son compte de touches pour chaque schéma de demandes, plutôt que pour chaque ressource, le serveur d'origine peut obtenir des statistiques distinctes pour de nombreux aspects d'une demande HTTP.

Par exemple, si une page a changé sur la base de la valeur de l'en-tête User-Agent dans la demande, les comptes de touches seront conservés pour chaque nuance différente de navigateur. Mais en fait, c'est plus général que cela; comme plusieurs combinaisons d'en-tête peuvent se transposer en la même variante, cela permet aussi au serveur d'origine de compter le nombre de fois (par exemple) où la version swahili d'une page a été demandée, quand bien même elle ne serait disponible qu'en anglais.

Si un mandataire ne prend pas en charge le mécanisme Vary, [4] dit alors qu'il NE DOIT PAS mettre de réponse en antémémoire si elle porte un en-tête Vary, et donc qu'il n'a besoin de mettre en œuvre aucun aspect de ce concept de compte de touches ou de limitation d'usage pour les variations de ressources.

Note : Cela implique aussi que si un mandataire prend en charge le mécanisme Vary mais ne veut pas tenir des comptes de touches indépendants pour chaque variante de réponse dans son antémémoire, il doit alors suivre au moins une des ces règles :

1. il ne doit pas utiliser l'en-tête Meter dans une demande pour offrir le compte de touches ou la limitation d'usage des réponses,
2. si il n'offre pas de compter les touches ou de limiter l'usage des réponses, et reçoit alors une réponse qui comporte à la fois un en-tête Vary et un en-tête Meter avec une directive qu'il ne peut satisfaire, le mandataire ne doit alors pas mettre la réponse en antémémoire.

En d'autres termes, il est permis à un mandataire de mettre partiellement en œuvre le mécanisme Vary en ce qui concerne le compte de touches, pour autant que cela n'ait pas d'effet visible extérieurement sur sa capacité à se conformer à la spécification Meter.

Cette approche fonctionne pour le comptage de presque tous les aspects du flux de demandes, sans incorporer de liste spécifique des aspects comptables dans la mise en œuvre de la spécification ou du mandataire.

### 7.2 Interaction avec la négociation de contenu transparent

[Une description de l'interaction entre ce concept et celui proposé de négociation de contenu transparent (TCN,

*Transparent Content Negotiation*) [6] sera disponible dans un document ultérieur.]

## 8 Note sur la capture des Referer

On prétend que certains annonceurs veulent payer des fournisseurs de contenus, non par nombre de "touches", mais par "consommateur" -- le nombre de personnes qui cliquent réellement sur l'annonce pour avoir plus d'information.

HTTP a déjà un mécanisme pour cela: l'en-tête "Referer". Cependant, peut-être devrait il être désactivé pour des raisons de confidentialité -- selon la spécification HTTP/1.1 :

"parce que la source du lien peut être une information privée ou peut révéler une source d'information par ailleurs confidentielle, il est fortement recommandé que l'utilisateur soit capable de choisir si le champ Referer est envoyé ou non."

Cependant, dans le cas de publicités, la source du lien veut en fait laisser la page de référence faire savoir d'où vient la référence.

Cela ne nécessite pas l'ajout d'un autre mécanisme, mais plutôt de pouvoir utiliser les schémas qui incorporent le référant dans l'URI d'une manière similaire à:

`http://www.blah.com/ad-reference?from=site1`

Un tel URI devrait pointer sur une ressource (peut-être un script CGI) qui retourne une réponse 302 de redirection sur la page réelle :

`http://www.blah.com/ad-reference.html`

Les mandataires qui ne mettent pas en antémémoire les 302 seront cause d'une touche sur la page de redirection par utilisation, mais la page réelle sera mise en antémémoire. Les mandataires qui mettent en antémémoire les 302 et font rapport des touches sur les 302 en antémémoire se comporteront de façon optimale.

Cette approche présente l'avantage de fonctionner que le client final ait ou non désactivé l'utilisation de Referer. Combiné avec le reste de la proposition de compte de touches dans ce concept, cette approche permet, par exemple, à un annonceur de savoir avec quelle fréquence une référence a été faite à une annonce à partir d'une certaine page.

## 9 Autres propositions

Il peut y avoir un certain nombre d'autres façons de rassembler les informations démographiques et d'usage ; d'autres mécanismes peuvent répondre à un ensemble différent de besoins que ne le fait la présente proposition. Elle n'empêche certainement pas la proposition ou le déploiement d'autres mécanismes, et beaucoup d'entre eux peuvent être complémentaires et compatibles avec le mécanisme proposé ici.

Il y a eu quelques spéculations sur les méthodes d'échantillonnage statistique qui pouvaient être utilisées pour rassembler des données d'une précision raisonnable. Une de ces propositions est de manipuler l'arrivée à expiration des antémémoires de sorte que les ressources sélectionnées soient inaccessibles pendant des périodes soigneusement choisies, permettant aux serveurs de compter précisément les accès durant ces périodes. Le mécanisme de comptage des touches proposé ici est entièrement complémentaire de cette approche, car il pourrait être utilisé pour réduire le coût du rassemblement de ces comptes. James Pitkow a écrit un article qui compare un projet antérieur de cette proposition de comptage des touches avec les approches d'échantillonnage [9].

Phillip Hallam-Baker a proposé d'utiliser un protocole d'échange de journaux [5], par lequel un serveur pourrait demander les journaux d'un mandataire en faisant une demande HTTP au mandataire. Cette proposition affirme qu'on "pense qu'elle fonctionne correctement dans les configurations qui impliquent plusieurs mandataires", mais on ne sait pas si c'est vrai si un mandataire externe est utilisé comme pare-feu (unidirectionnel). La proposition laisse aussi ouvertes un certain nombre de questions, comme comment un serveur d'origine peut être sûr que tous les mandataires dans le sous arbre des demandes prennent effectivement en charge l'échange de journaux. On ne voit pas très bien non plus comment cette proposition se marie avec la prise en charge par un mandataire de l'échange de journaux avec la permission d'un serveur de mettre une réponse en antémémoire.

Pour les fondements généraux sur le sujet des normes de mesure sur la Toile, voir l'exposé de Thomas P. Novak et Donna L. Hoffman [8]. Voir aussi le page "Privacy and Demographics Overview" tenue par le World Wide Web Consortium [10], qui comporte un pointeur sur certaines propositions pour rassembler les informations démographiques sur les

consommateurs (pas seulement de compter les références) [3].

## 10 Considérations pour la sécurité

À quels clients externes un serveur (mandataire ou d'origine) peut-il se fier pour rapporter les comptes de touches ? Un mandataire malveillant pourrait facilement rapporter un grand nombre de touches sur certaines pages, et donc peut-être causer un gros paiement à un fournisseur de contenu de la part d'un annonceur. Pour aider à éviter cette possibilité, un mandataire peut choisir de ne relayer les comptes d'usage reçus des ses mandataires externes à ses serveurs internes que lorsque les mandataires se sont authentifiés en utilisant Proxy-Authorization et/ou qu'ils sont sur une liste de mandataires approuvés.

Il n'est pas possible de mettre en application des limites d'usage si un mandataire veut tricher (c'est-à-dire, si il offre de limiter l'usage mais ignore ensuite une directive Meter du serveur).

Concernant la confidentialité : il apparaît que le concept de ce document ne révèle pas plus d'informations sur les usagers individuels que n'en serait déjà révélé par la mise en œuvre de la prise en charge existante par HTTP/1.1 de "Cache-control: max-age=0, proxy-revalidate" ou "Cache-control: s-maxage=0". Il peut, en fait, aider à dissimuler certains aspects de la structure organisationnelle sur le côté externe d'un mandataire. En tous cas, le conflit entre les exigences d'anonymat de l'utilisateur et les exigences du serveur d'origine pour les informations démographiques ne peut pas être résolu par des moyens purement techniques.

## 11 Remerciements

Nous tenons à manifester notre reconnaissance pour les commentaires constructifs reçus de Anselm Baird-Smith, Ted Hardie, Koen Holtman (qui a suggéré la technique décrite à la section 8) Dave Kristol, Ari Luotonen, Patrick R. McManus, Ingrid Melve, et James Pitkow.

## 12 Références

- [1] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", RFC2119, BCP 14, mars 1997.
- [2] Anwat Chankhunthod, Peter B. Danzig, Chuck Neerdaels, Michael F. Schwartz, and Kurt J. Worrell, "A Hierarchical Internet Object Cache" Proc. 1996 USENIX Technical Conf., San Diego, janvier, 1996, pp. 153-163.
- [3] Daniel W. Connolly, "Proposals for Gathering Consumer Demographics" <http://www.w3.org/pub/WWW/Demographics/Proposals.html>.
- [4] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, "Protocole de transfert Hypertext -- HTTP/1.1", RFC2068, janvier 1997. (*Obsolète, voir RFC2616*) (P.S.)
- [5] Phillip M. Hallam-Baker, "Notification for Proxy Caches", W3C Working Draft WD-mandataire-960221, World Wide Web Consortium, février 1996. <http://www.w3.org/pub/WWW/TR/WD-mandataire.html>.
- [6] K. Holtman, A. Mutz, "Négociation de contenu transparent dans HTTP", RFC2295, mars 1998. (*Expérimentale*)
- [7] Mogul, J., "Forcing HTTP/1.1 proxies to revalidate responses", Travail en cours.
- [8] Thomas P. Novak and Donna L. Hoffman, "New Metrics for New Media: Toward the Development of Web Measurement Standards". Ce projet d'article est actuellement disponible à <http://www2000.ogsm.vanderbilt.edu/novak/web.standards/webstand.html>. Cité avec la permission de l'auteur ; ne pas citer sans permission.
- [9] James Pitkow, "In search of reliable usage data on the WWW". Proc. Sixth International World Wide Web Conference, Santa Clara, CA, avril 1997.
- [10] Joseph Reagle, Rohit Khare, Dan Connolly, and Tim Berners-Lee, "Privacy and Demographics Overview" <http://www.w3.org/pub/WWW/Demographics/>.
- [11] Linda Tauscher and Saul Greenberg. "Revisitation Patterns in World Wide Web Navigation". Research Report 96/587/07, Department of Computer Science, University of Calgary, mars 1996.

<http://www.cpsc.ucalgary.ca/projects/grouplab/papers/96WebReuse/TechReport96.html>.

- [12] D. Wessels, K. Claffy, "Protocole des antémémoires de l'Internet (ICP), version 2", RFC [2186](#), septembre 1997. (*Information*)

### 13 Adresse des auteurs

Jeffrey C. Mogul  
Western Research Laboratory  
Digital Equipment Corporation  
250 University Avenue  
Palo Alto, California, 94305, U.S.A.  
mél : [mogul@wrl.dec.com](mailto:mogul@wrl.dec.com)  
téléphone : 1 415 617 3304 (email preferred)

Paul J. Leach  
Microsoft  
1 Microsoft Way  
Redmond, Washington, 98052,  
U.S.A.  
mél : [paulle@microsoft.com](mailto:paulle@microsoft.com)

### 14 Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (1997). Tous droits réservés.

Ce document et les traductions de celui-ci peuvent être copiés et diffusés, et les travaux dérivés qui commentent ou expliquent autrement ou aident à sa mise en œuvre peuvent être préparés, copiés, publiés et distribués, partiellement ou en totalité, sans restriction d'aucune sorte, à condition que l'avis de droits de reproduction ci-dessus et ce paragraphe soient inclus sur toutes ces copies et œuvres dérivées. Toutefois, ce document lui-même ne peut être modifié en aucune façon, par exemple en supprimant le droit d'auteur ou les références à l'Internet Society ou d'autres organisations Internet, sauf si c'est nécessaire à l'élaboration des normes Internet, auquel cas les procédures pour les droits de reproduction définies dans les processus des normes de l'Internet doivent être suivies, ou si nécessaire pour le traduire dans des langues autres que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Société Internet ou ses successeurs ou ayants droit.

Ce document et les renseignements qu'il contient sont fournis "TELS QUELS" et l'INTERNET SOCIETY et l'INTERNET ENGINEERING TASK FORCE déclinent toute garantie, expresse ou implicite, y compris mais sans s'y limiter, toute garantie que l'utilisation de l'information ici présente n'enfreindra aucun droit ou aucune garantie implicite de commercialisation ou d'adaptation à un objet particulier.