Travail de maturité

L’archivage et la citation de ressources web

Nicholas Helke

bonjour@nhelke.com

+41 77 406 7350

Crêts de Pregny 14B
1218 Gd Saconnex
Switzerland

nhelke.com

Résumé

La motivation de l'auteur pour ce mémoire était de rechercher pourquoi le web était considéré comme sorcier par tant de monde. Avant l'écriture du mémoire la conclusion auquel l'auteur est arrivé est que cette peur vient d'une angoisse. Une angoisse qui est lié à l'incompréhension de l'enorme espace du web en métamorphose perpétuelle.

Ce travail cherche à s'attaquer au problème d'archivage de ressources sur le web car elles sont aujourd'hui volatiles. La résolution de ce problème permettrait de se référer au web dans des rapports ou articles là où jusqu'ici on ne l'osait à cause de sa volatilité.

La précision de la problématique et l'étude de pratiques actuelles lié à l'utilisation du web comme source font l'objet des chapitres 1 à 4.

Au chapitre 5 est étudié la problématique posé par les droits d'auteurs qui interdisent à priori toute copie, y compris pour l'archivage. Trois scénarios pour contourner cette interdiction sont décrits.

Ce travail propose, au chapitre 6, une solution théorique, mais qui est réalisable à court terme. Cette solution est un programme informatique sur un serveur accessible via un site web qui permet de commander l'archivage de n'importe quelle ressource.

Ces ressources doivent alors être identifiables. Le choix d'un identifiant est évoqué et l'algorithme NHA1, choisi, est introduit. Son expression intégrale en language de programmation Ruby est en annexe.

Enfin, au chapitre 7, l'auteur tente d'imaginer la suite. La possibilité de réaliser la solution est évoquée. Les projets de la communauté internationale pour la révision des droits d'auteur font aussi l'objet du chapitre.

Ce mémoire à été écrit en LaTeX 2$_\varepsilon$ qui est une norme *de facto* pour la préparation de documents de recherche scientifique. Ce mémoire ne prétend pas l'être, mais le chemin futur envisagé par l'auteur l'a encouragé à se familiariser avec cet outil à l'avance.

Ce mémoire compte environ 4036 mots.

Table des matières

1 Introduction

Tout ce qui à été publié a un moment ou à un autre est une ressource consultable et susceptible d'être citée.

Un problème existe cependant si on cite une source. Il faut qu'elle reste accesible afin qu'un lecteur puisse à une date ultérieure consulter la dite source. Sinon il ne pourra pas vérifier la citation ni s'instruire sur son contexte.

Ce problème a toujours existé. C'est le problème de la volatilité. Les archives et les bibliothèques – qui ne sont en fin de compte qu'une sorte d'archive – ont pour but de lutter contre ce problème.

En 1989 Sir Tim Berners-Lee invente le web, en 1991 sa création quitte les locaux du CERN, lieu de conception, et rejoint le domaine public où il prend son incroyable essor.

Cet invention a porté le problème de la volatilité à des proportions exorbitantes. Les ressources sur le web identifiées à l'aide d'un URL[1] peuvent être modifiées, déplacées à une autre adresse (URL) ou même détruites sans laisser de traces.

Pour prévenir des éventuelles divergences entre la citation et la ressource référencée via l'URL, on recommande aujourd'hui d'ajouter à une citation de ressource web, la date de consultation.[2] Cette pratique ne résout cependant pas le problème. Par exemple :

Simon écrit un mémoire. Il veut citer un excellent article qu'il a trouvé sur la page d'un étudiant au Massachusetts Institute of Technology (MIT). Le problème est que l'espace web d'étudiants au MIT est réservé aux étudiants. Lorsque l'auteur de l'article en question matriculera, son espace sera effacé, et la référence bibliographique dans le mémoire de Simon sera obsolète.

La pérennité de livres – et d'une façon générale d'imprimés – est beaucoup plus sûre. Un texte imprimé jouit d'une existence physique permanente et souvent dans la mesure où il est publié, il existe physiquement en plusieurs exemplaires qui seront distribués. Cette existence plurielle rend la destruction du texte plus difficile. Ainsi une ressource imprimée et référencée pourra beaucoup plus probablement être retrouvée à des fins de vérification ou de recherche approfondie.

Par extension il en va de même pour les différentes éditions d'une œuvre. La parution sur un média physique fait qu'une nouvelle édition ne détruit pas d'office les exemplaires des éditions antérieurs déjà en circulation.

Avec les médias virtuels – le web en tête en tant qu'espace le plus foisonnant en information – une nouvelle édition écrase par défaut entièrement l'édition précédente.

Il faut cependant reconnaître l'intérêt des libertés offertes par le web. Elles ont notamment permises aux applications webs dynamiques (parfois regroupées sous le terme « Web 2.0 ») de se développer comme Amazon, facebook, ou Gmail.

Il ne s'agit pas d'estropier ces services dynamiques en proposant un web plus statique. Ni de s'embarquer dans la tâche, impossible, de changer la norme si largement adoptée des URL, pour la rendre consciente des besoins de pérennité.

[1] *Universal Ressource Locator*, soit la localisation universelle de ressources.
[2] *The Chicago Manual of Style*, 15^{e} édition, §17.12.

2 Solutions d'archivage actuelles de ressources web

À ma connaissance il n'existe que deux organisations dignes d'êtres mentionnées ici pour être déjà conscient de l'importance de la pérennité des sources, mêmes virtuelles. Ce sont Wikipédia et Archive.org

2.1 Wikipédia

Wikipédia garde toutes les éditions de toutes ses ressources. Leur implémentation de cette politique est également digne de compliments. Par défaut la toute dernière édition de la ressource demandée est affichée. Un petit lien accompagnant toutes les ressources permet d'accéder à l'historique des éditions. Si cette pratique s'était démocratisée ce mémoire n'aurait pas lieu d'être.

Profitons de la mention de Wikipédia pour signaler que son utilisation comme source est déconseillée. Jimmy Wales, co-fondateur de Wikipédia, la qualifie d'un portail vers le savoir plutôt qu'une source définitive. D'ailleurs c'est la pratique de pérennité qui permet à Wikipédia de rester un assez bon portail malgré son anarchisme.

Wikipédia est un exemple d'une situation où les sources sur le web énumérés dans les références d'articles bénéficierait énormément d'archivage sûre de ressources web. En tant que portail ses références sont clés. Elles sont la destination automatique pour approfondir un sujet.

2.2 Archive.org

Un site web se soucie déjà de l'archivage du web. Archive.org s'efforce depuis 1995 de prendre des clichés du web à intervales variables. Ces intervales varient pour des raisons économiques et les clichés se limitent au « surface web ». C'est-à-dire les pages web les plus populaires comme : `http://www.apple.com`, `http://www.cnet.com`, `http://www.google.com`, `http://www.microsoft.com`, etc.

Archive.org rempli déjà une partie du marché d'archivage du web, la clé de cette phrase étant « une partie ». Archive.org a d'ailleurs servi de source à deux reprises pour ce mémoire (Cf. p. 11).

L'intérêt de ce mémoire est dans la partie restante du marché d'archivage du web. L'archivage des ressource plus obscurs – provenant du « deep web » – qu'il faut aussi pouvoir archiver. À condition que la demande existe.

3 Le marché de services d'archivage

Le web n'est pas une source comme on les connaissait jusqu'ici. Il a la particularité de se défaire d'éditeurs. Ses ressources sont pour la plupart la responsabilité d'un individu, auteur. Cependant le web est une ressource extrêmement riche à de nombreux niveaux. Sans se soucier de la valeur littéraire des sources, elles ont au minimum une valeur psychologique et à terme historique.

L'archivage du web est tout aussi important que l'archivage de sources imprimés voir plus important vu l'importance que prend le web dans la distribution d'informations. Peu importe le fond ou la forme de ces ressources.

Nos vies migrent vers ce média virtuel volatile sans que, jusqu'ici, l'on se souci du moyen à long terme, de la pérennité. Cependant le vent tourne. Le monde se réveille et se soucie maintenant de ces choses. L'exemple le plus parlant au niveau du consommateur est la multiplication d'applications de sauvegarde des données locales.

Ce souci va rapidement se tourner vers les contenus en réseau, sur le web. Le marché d'archivage du web est aussi vaste que le web entier si ce n'est plus grand dans la mesure ou il faudra aussi archiver les éditions antérieures du web.

4 L'utilisation du web comme source

Le web est une enorme source. En tout cas c'est une source psychologique et, à terme, historique. Certaines parties du web sont aussi des sources littéraires.

L'apparente anarchie fait peur aux académiciens. Plutôt que d'apprendre à être lecteur averti, critique des sources, on décourage sérieusement l'utilisation du web comme source à tous les élèves.

4.1 Lecture critique

L'absence de bibliographie dans beaucoup d'articles sur le web ampute la possibilité de vérifier les sources des affirmations. Inhibant aussi la possibilité de faire la différence entre un travail original sans fondements, médiocre, et un travail original brillant, mais lacunaire. Sans compter les travaux plagiés. Cela est en partie la conséquence de l'élimination de l'éditeur, intermédiaire tuteur, censeur et applicateur de normes. Chose qu'il ne faut cependant pas reprocher au web. Certaines perles dont les éditeurs ne veulent rien savoir, trouvent sur le web un lectorat.

Le web peut être utilisé pour toute recherche qui ne nécessite pas une référence. Par exemple cartes, dates de naissance et de mort. Tout ce qui compte dans la sagesse populaire ou la culture générale, et qui en tant que telle ne requiert pas de référence peut être emprunté au web.

Cependant si le texte ne respecte pas les mêmes critères de qualité qu'on demanderait à un article sur papier il faut l'écarter comme source de documents académiques à référencer. Ces exigences sont structuration, énumération des références, nom, prénom et coordonnés de l'auteur. Doivent être écartées toutes ressources ne respectant pas quelque critères supplémentaires spécifiques au web. Il faut éviter les pages hébergées en Polynésie ou en Afrique, leur législation ne protégeant pas contre la fraude. Privilégier les pages dépourvues de publicités.

Ces critères techniques ne sont malheureusement pas absolues. Elles pourraient être rendues plus sûre par un label de qualité.

4.2 Utilisation de services d'archivage

Le marché de l'archivage du web peut aussi prendre de l'importance pour les références bibliographiques. L'usage veut qu'on cite une édition spécifique et que la référence corresponde. Hors cette démarche est entravée par l'auto-écrasement des données.

Les services d'archivage pourraient, si ils sont correctement comuniqué au monde, devenir un outil clé pour citer les ressources web. L'utilisation d'un service d'archivage pour la citation d'une source à été illustrée dans ce mémoire à deux reprises. (Cf. p. 11)

L'application proposée par ce mémoire (cf. 6.1) produirait automatiquement le texte bibliographique lorsque l'on y entrera l'URL d'une ressource, chose qui encouragerait à énumérer ses sources. Elle pourrait à terme aussi délivrer ou non un label de qualité. (Cf. 4.1)

5 Conflit avec la volonté de protéger la propriété intellectuelle

Je ne parle pas de plagiat ce problème est beaucoup plus vaste que l'étendu de ce mémoire. C'est un problème qui existe depuis que les sources existent et qui n'a que peu de choses à voir avec le média.

La convention de Berne sur les droits d'auteurs prévoit une protection automatique pour les droits d'auteurs sur toute propriété intellectuelle sauf indication contraire.

Le problème avec l'archivage du web est qu'on doit faire une copie des ressources qu'on veut archiver. Les imprimés ne sont pas copiés pour êtres conservés. Ils changent de mains pour arriver dans ceux de l'archiviste. Les ressources web ne changent pas de mains. Elles restent sur le serveur de l'auteur à bonne foi et il n'y a pas d'autre moyen d'archiver ces ressources sans les copier.

L'archivage de ressources qui ne sont pas clairement indiqués comme appartenant au domaine public est interdit. C'est une mauvaise conception de dire que le contenu du web est dans le domaine public. Il n'est que consultable publiquement.

Tous les protagonistes sont d'accords qu'il faut introduire de nouvelles exceptions pour permettre au niveau mondiale l'archivage du web. Actuellement les exceptions aux droits d'auteurs ne sont pas fixés par la convention de Berne. Chaque état fixe lui-même ces exceptions.

Pour archiver des ressources du web il faut soit :

- Dans certains pays des associations peuvent obtenir de la part de l'état une autorisation spéciale pour l'archivage. Cependant l'inconvénient de cette solution est que l'autorisation est uniquement valable dans un pays pour les contenus du même pays.
- Le demandeur pourrait s'assurer que il a le droit de faire une copie éducative. Ce droit varie d'état en état. Ça serait au demandeur d'assurer la légalité de sa commande. Puis l'archive devrait rester uniquement consultable par lui – l'autorisation étant personnelle.
- Dernièrement, la solution ubiquiste. La demande au détenteur des droits. Cette démarche est prévue par la convention de Berne. Elle nécessite une grande logistique

pour traiter les demandes. En plus le détenteur des droits peut décider arbitrairement s'il veut déroger l'autorisation ou non. Aucune convention ne l'y encourage.

Digital Object Identifier (DOI)

Le *Digital Object Identifier* (DOI) est une nouvelle norme analogique à l'*International Standard Book Number* (ISBN) pour les ressources numériques, virtuelles.

Le DOI ne règle en rien l'archivage.

Comme les numéros ISBN les DOI sont délivrés par des autorités compétentes enregistrés auprès de l'organisation mère. Relevons l'incompatibilité de ce procédé avec la majorité des contenus sur le web qui sont publiés par un individu, une personne physique.

Le DOI est une norme concurrente des URL. Elle a certes un avantage, elle suit les déplacements de ressources, réduisant un peu la volatilité. C'est une norme mal adoptée encore, qui ne fait pas la même unanimité que la norme des URL.

Le scénario envisagé par la fondation des DOI prévoit que les éditeurs continuent d'héberger à bonne foi leur contenu – évitant ainsi de se confronter au problème de droits d'auteurs – mais ne règlent pas le problème de pérennité garantie.

6 Développement d'une solution

6.1 L'expérience pour l'utilisateur

Simon écrit un mémoire. Il veut citer un article excellent du web et il veux s'assurer que cette référence sera toujours consultable dans 10, 15 ou 20 ans.

Pour éviter le risque de voir disparaître la ressource il faudrait pouvoir stocker un exemplaire dans une bibliothèque virtuelle. C'est une pratique millénaire. Les universités qui voulaient être en position de toujours pouvoir consulter certaines sources en ont obtenues des exemplaires, qu'elles ont rangées soigneusement dans leurs bibliothèques. En d'autres termes : afin de garantir l'accessibilité aux ressources référencées, il faut les archiver.

C'est à ce stade que l'utilisateur emploiera l'application présentée par ce mémoire. L'utilisateur copiera l'URL de la ressource qu'il désire référencer et la collera dans le champ prévu à cet effet sur le site de l'application. (L'URL de l'application reste encore à fixer.) Il entrera en plus son adresse e-mail et validera le formulaire.

Après avoir validé le formulaire l'utilisateur verra une page lui indiquant la progression de sa requête, puis après 30 à 60 secondes la copie de la ressource originale sera affichée avec le texte bibliographique et le nouvel URL permanent de l'exemplaire.

Peuvent être archivés ainsi : articles de presse, editions de pages avant erratum, images ou documents PDF susceptibles d'être modifiés ou effacés. En somme tout ce qui ressemble à une source qui pourrait être sur papier. L'application devrait à terme se construire la réputation d'être une autorité en archivage de ressources web. Vous voulez avoir la preuve plus tard qu'une ressource ressemblait à X, Y ou Z à un moment donné, il vous suffit de commander l'archivage de la ressource via l'application.

Certaines ressources ne sont pas concernées par cette solution. Les raisons sont techniques et sont énumérés au point 6.2.3. Ne sont pas concerné : contenus Adobe (ancien-

nement Macromedia) Flash (exemple : YouTube), contenus faisant appel à du JavaScript, toute forme d'élément dynamique (par exemple citations dynamiques) ou interactif (exemple : diaporama) et Wikipedia[3].

6.2 Logique du programme informatique théorique

6.2.1 Insertion de références dans la base de données

Il y a un certain nombre de vérifications à effectuer avant d'accepter une requête en provenance du formulaire d'inscription d'une ressource.

Avant de traiter la moindre chose l'application vérifie si le formulaire à été envoyé depuis une session valable. C'est une fonction de protection contre le spam, aucun formulaire n'est considéré si il n'a pas été envoyé depuis la page d'accueil.

Ensuite l'URL est comparé à une *black-list*, liste des site interdits de citation. C'est une liste qui servira en cas de dispute legale. D'office l'application pourra employer des black-lists publiques pour interdire la validation de sites pornographiques et malwares[4].

L'URL est ensuite cherché dans la base de données de l'application, pour voir si elle a déjà été archivée. Si c'est le cas la page est comparée avec sa version archivée. L'application choisira ensuite, en fonction de cette comparaison, soit de renvoyé la requête vers la ressource précédemment archivée, soit de continuer le processus d'archivage.

Enfin la page est vérifiée. Si il y a une ambiguïté sur le dynamisme de la page le processus est halté avec une erreur, de même pour les pages masquant leur vrai URL. Un filtre de contenu pour raisons légale pourra en cas de besoin intervenir à ce stade.

Enfin, si le document passe toutes ces conditions, il est enregistré puis analysé pour retrouver ses ressources incluses comme ses illustrations. Ces ressources passent par par les mêmes filtres avant d'être enregistré. Enfin la page d'origine est modifiée, au besoin, pour inclure les illustrations archivées. Ainsi la page archivée est entièrement indépendente du capharnaüm du web.

Ressources dynamiques

Revenons un moment aux ressources dynamiques du web. Les services 2.0. Gmail, facebook, etc.

Le fait que les ressources sur le web ne sont pas toutes statiques et donc susceptibles d'être référencées n'est pas un problème en soi. Le problème est la transparence des ressources dynamiques pour l'utilisateur. C'est-à-dire que l'utilisateur n'est pas averti et ne peut pas identifier à l'aide de son navigateur si la page en cours de consultation est dynamique ou statique (donc citable).

Par exemple le service Gmail ne doit pas répondre la même chose à tous ses visiteurs. Chaque utilisateur reçoit des données très différentes et personnelles pour la même adresse. Quand Simon visite `http://gmail.com/` il veut voir ses emails et non pas ceux d'Eric.

[3] Wikipédia n'est pas concerné car il se charge déjà d'assurer la pérennité de ses ressources. Cf. 2.1

[4] À but malicieux. Souvent ces sites servent à répandre des virus.

Techniquement la différenciation est claire et simple. Une ressource statique est obtenue avec la commande « GET[5] » et une ressource dynamique est renvoyée après une requête « POST[6] ».

Pour reprendre l'exemple de Gmail, par défaut quand on tape un URL dans le navigateur, celui-ci va le chercher à l'aide de la commande GET. La commande GET sur l'adresse `http://gmail.com/` reçoit en réponse le formulaire d'identification. Le contenu du formulaire est envoyé dans la commande POST lorsqu'on soumet le formulaire. Cette commande POST reçoit selon le contenu du formulaire soit :

- en cas de vérification de l'identification, les e-mails personnelles, ou ;
- en cas d'échec de l'identification, la page d'identification est renvoyé avec un avertissement.

Pour éviter que l'on soit tenté à tort de référencer une ressource dynamique il faudrait que les navigateurs indiquent à l'utilisateur si la page est dynamique ou non. Dans la solution proposée par ce mémoire ce point est vérifié et le cas échéant l'utilisateur est averti qu'il ne pourra pas citer cette ressource.

6.2.2 Identification des références

Une fois archivée, la resource doit pouvoir être retrouvé, sinon l'archivage n'a pas de sens. Pour les raisons données plus haut, l'identifiant sera un URL. Les URL sont connus de tous les utilisateurs du web c'est-à-dire tous les utilisateurs potentiels de ce système.

Un URL est composé d'un domaine et d'un chemin d'accès, *domain* et *file path* en anglais. Le domaine est ce qui vient après le `http://` et avant le premier `/`. Le chemin d'accès est tout ce qui suit le premier `/`. Ses niveaux se comptent depuis le premier `/` et sont séparés entre eux par des `/`.

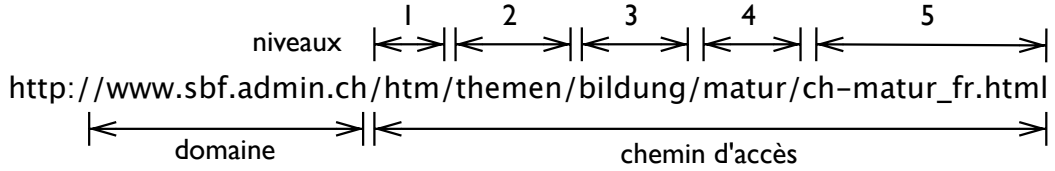

Fig. 1: Structure d'un URL

En l'occurrence, le domaine sera simplement celui du serveur d'archivage – qui doit encore être fixé. La partie variable est donc le chemin d'accès. Le chemin d'accès sera composé d'un seul niveau. Les ressources seront identifié à ce niveau par l'identifiant NHA.

[5] GET, littéralement « chercher » en anglais.

[6] POST, littéralement « envoyer » en anglais. Commande appelé ainsi car elle contient des informations personnelles à destination du server. Par exemple : Nom d'utilisateur et mot de passe.

Un bon identifiant

Un bon identifiant doit être unique, il devrait aussi avoir une unité de forme et être pratique. Un identifiant n'est pas pratique si il ne permet pas d'identifier avec certitude un objet plus succinctement que par la voie descriptive. L'unité de forme rend l'identifiant reconnaissable et équitable. Ces deux dernières raisons ont contribué à l'obligation de composer l'indicatif des numéros de téléphone en Suisse. En plus l'unité de forme peut simplifier l'implémentation de systèmes informatiques gérant les identifiants.

Les tableaux de bases de données informatiques identifient leurs objets[7] grace à des identifiants entiers positifs non nul et successifs. $i_1 = 1; i_{n+1} = i_n + 1; n \in \mathbb{N}^*$ L'identifiant répond à la condition d'unicité. Si un objet est effacé de la base de donnée son identifiant n'est jamais réutilisé. Ces identifiants sont utilisés à travers toute l'application pour l'identification d'objets.

Ces identifiants ne respectent cependant pas toutes les conditions énumérées ci-dessus. Elle ne sont pas équitables. Les 9 premiers objets ont un identifiant à un seul chiffre, les 90 suivants à deux chiffres, les 900 suivants à trois chiffres et ainsi de suite. De plus, comme ils sont séquentiels, ils sont prévisibles. On risquerait, dans ce cas, d'assister à un phénomène de « *cybersquatting* ». C'est-à-dire que certaines personnes malintentionnées pourraient être tentées d'occuper de diverses manières les identifiants remarquables, tels 100, 666, 1 000, 1 111, etc.

Pour éviter ce phénomène il faut donner à l'espace des identifiants de l'URL de l'entropie. C'est-à-dire qu'il faut qu'il soit imprévisible. Il faut éviter deux choses. Premièrement que la suite soit prévisible pour lutter contre le *cybersquatting*. Secundo, il ne faut pas qu'à chaque identifiant permis par la forme corresponde un objet pour éviter le tourisme.

Le tourisme est indésirable car le but de l'application n'est pas de donner libre accès aux ressources effacées du web. La politique est que si un contenu à été retiré c'est pour une bonne raison, cependant ceux qui avaient vu la ressource devrait avoir le droit de continuer à consulter une version archivée. Après tout ils pourraient très bien en avoir conservé eux-mêmes une copie. Cependant une copie individuelle vaut peu de choses lorsqu'il s'agit de vérifier une source. Surtout quand on sait comment il est facile dans l'ère numérique de falsifier une ressource électronique. En plus la notion d'original est ambigu dans l'existence de copies parfaites – 0 pour 0 et 1 pour 1.

Le web comptait en 2001 environ 550 milliards de ressources[8]. Ce nombre a sans doute augmenté. En 2001, Google avait indexé 1 millard de pages[9] et en 2005 lorsqu'ils arrêtèrent de publier la taille de leur index il contenait 8 milliards de pages[10]. Le lieu géométrique des identifiants doit être grand.

Pour raccourcir les identifiants elles seront exprimés en base 62. C'est-à-dire 0 à 9, a à z et A à Z. Profitions de rendre les identifiants équitables par la même occasion. Fixons la forme comme étant un nombre à six chiffre en base 62, soit `aZiL09`, par exemple.

Pour éviter de devoir garder un index des identifiants utilisés pour l'URL, il faut

[7] Souvent une rangée d'un tableau correspond à un objet qui sera traité par l'application. C'est le cas de l'application présentée par ce mémoire. Les objets sont des éditions de ressources webs.

[8] Bergman, Michael K., *The Deep Web: Surfacing Hidden Value.*

[9] Google, *Page d'accueil*, janvier 2001.

[10] *Idem*, septembre 2005.

développer un algorithme qui traduit l'identifiant de l'URL en l'identifiant employé par l'application et vice versa.

Cet algorithme s'appelle NHA1.

NHA1

NHA1 est un algorithme qui établit une relation entre l'espace de nombres à six chiffres en base 62 et l'espace des entiers naturels entre 1 et 5 680 023 558. Cette limite est imposé par la division par 10 de 62^6, la dimension de tous les nombres de six chiffres en base 62.

La division par dix permet d'ajouter une somme de contrôle Verhoeff au dernier chiffre de la représentation en base dix. Cette somme de contrôle ne prévient pas les erreurs de frappe dans l'expression en base 62, mais elle sert à intégrer l'entropie voulue. Grace à elle, seul 10% de tous nombres de six chiffres en base 62 ont une relation biunivoque avec l'algorithme NHA1.

L'expression en Ruby de NHA1 est en annexe.

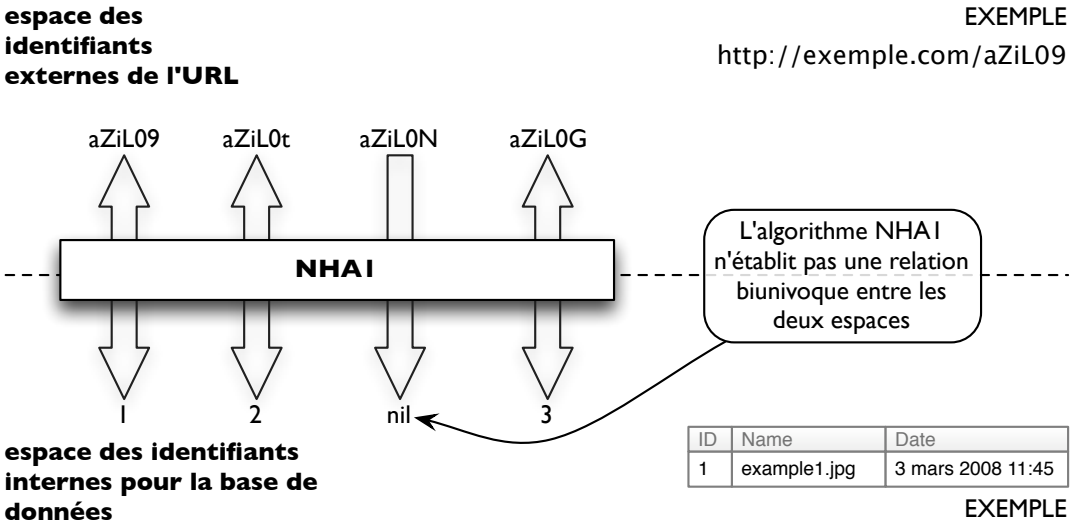

Fig. 2: NHA1

Dans le futur, comme l'entropie est prévisible par l'algorithme de Verhoeff, on peut facilement envisager une fois NHA1 saturé de réutiliser une partie de l'espace des nombre à six chiffres en base 62, avec un NHA2, par exemple.

6.2.3 Affichage des ressources archivées

Cette fonction est en fait la plus risquée de l'application. C'est à ce stade que le code malicieux éventuellement caché dans une ressource archivée pourrait être exécuté. C'est ce qu'on appelle un « *Cross-Site Scripting attack* » ou attaque *XSS*. Le moyen le plus efficace de lutter contre cette pratique exécrable est de s'assurer qu'absolument aucun

code n'est exécuté. Au prix, malheureusement, de fonctionnalités légitimes et voulues.

Pour ce faire le code de la page doit être modifié avant d'être affiché. Cette modification peut se faire lors de l'enregistrement de la page, scénario statique, ou à la volée, lors de l'affichage de la page, scénario dynamique.

Le scénario statique requiert plus d'espace de stockage, puisqu'il faut obligatoirement stocker une copie pure et une copie anti-XSS.

Le scénario dynamique requiert plus de puissance CPU pour effectuer en temps réel, à chaque requête, la procédure XSS.

Le juste milieu serait de faire des copies statiques des ressources ayant un traffic fréquent et de rester en mode dynamique pour les autres.

7 Implementation

Cette solution devrait être facilement réalisable. Le développement de la solution théorique tient compte d'une réalisation par étapes. La capacité d'une réalisation par étapes de s'adapter à la croissance pourrait faire l'objet de l'oral.

Le 15 juillet, l'Organisation Mondiale de la Propriété Intellectuelle (OMPI) à Genève organise un séminaire mondiale sur préservation numérique de ressources soumis aux droits d'auteurs. Les conclusion de cette rencontre internationale feront sûrement l'objet d'une partie de l'oral.

Annexe : NHA1

```
#!/usr/bin/env ruby -wKU

NHAT=[
  "aMzbgxhue2fHkU8vPTnI4X6KcqmBNJQjGFLyl3o5WV9CtDsZp71diOEYRS0rwA",
  "Zv0w2x4y6z8AaBcCeDgEiFkGmHoIqJsKuL3M7NbOfPjQnRrS1T9UhVpW5XlYdt",
  "i2IDFawpjyuKdh6Xn0Hb1ezlfQsGRP8Eox3V4TBkZLWmJ7trYgUNOCcqS5MA9v",
  "LpFmcqzWukCXKaxoQfYn6BVO7e5lhtw1UZ2dHvPE4b8DIS0G9MTiJ3ysAgjRNr",
  "0lw9Edg1eaSNk7UH5KC2tTrPV36YmFpfDcZOXsxvRqBouMnzG8yiLA4IhQWJjb",
  "irU8nwFXzPhjYgK9p2fAbNdS0Jot7MuyLxBsGmETlvW5Hck46D3qVeCZ1RIaOQ"
]

def d(j,k)
  table = [
    [0,1,2,3,4,5,6,7,8,9],
    [1,2,3,4,0,6,7,8,9,5],
    [2,3,4,0,1,7,8,9,5,6],
    [3,4,0,1,2,8,9,5,6,7],
    [4,0,1,2,3,9,5,6,7,8],
    [5,9,8,7,6,0,4,3,2,1],
    [6,5,9,8,7,1,0,4,3,2],
    [7,6,5,9,8,2,1,0,4,3],
    [8,7,6,5,9,3,2,1,0,4],
    [9,8,7,6,5,4,3,2,1,0]
  ]
  return table[j][k]
end
def p(pos,num)
  table = [
    [0,1,2,3,4,5,6,7,8,9],
    [1,5,7,6,2,8,3,0,9,4],
    [5,8,0,3,7,9,6,1,4,2],
    [8,9,1,6,0,4,3,5,2,7],
    [9,4,5,3,1,2,6,8,7,0],
    [4,2,8,6,5,7,3,9,0,1],
    [2,7,9,3,8,0,6,4,1,5],
    [7,0,4,6,9,1,3,2,5,8]
  ]
  return table[pos%8][num]
end
def inv(j)
  table = [0,4,3,2,1,5,6,7,8,9]
```

```
  return table[j]
end

def generate_nha(id)
  if id <= 62**6/10
    id = id.to_s
    c = 0
    id.split(//).reverse.each_with_index do |n,i|
      c = d(c,p(i+1,n.to_i))
    end
    id <<= inv(c).to_s
    id = id.to_i
    res=""
    ndx = 5
    while id>0
      digit = id % 62
      res <<= NHAT[ndx].split(//)[digit]
      id /= 62
      ndx -= 1
    end
    while ndx>=0
      res <<= NHAT[ndx].split(//)[0]
      ndx -= 1
    end
    return res.reverse
  end
end

def lookup_nha(nha)
  if nha.length==6
    id=0
    nha.split(//).each_with_index do |digit,ndx|
      id = id*62+NHAT[ndx].index(digit.to_s)
    end
    id = id.to_s
    c = 0
    id.split(//).reverse.each_with_index do |n,i|
      c = d(c,p(i,n.to_i))
    end
    if c==0
      return id.split(//)[0,id.length-1].to_s
    end
  end
end
```

Références

- Bergman, Michael K., *The Deep Web: Surfacing Hidden Value*. Bright Planet, septembre 2001. `http://www.brightplanet.com/pdf/deepwebwhitepaper.pdf`
- *The Chicago Manual of Style*, 15e édition. The University of Chicago Press, 2003.
- *The DOI Handbook*, édition 4.4.1. International DOI Foundation, Inc., Oxford, octobre 2006.
- Google, *Page d'acceuil*. Archive.org, janvier 2001. `http://web.archive.org/web/20010118214400/http://www.google.com/`
- ——— , *Page d'acceuil*. Archive.org, septembre 2005. `http://web.archive.org/web/20050924172505/www.google.com/`
- *ISBN User' Manual*, 5e édition. International ISBN Agency, Berlin, 2005.
- Longcroft, Lucinda, Juriste principale, Division du commerce électronique, des techniques et de la gestion du droit d'auteur, Organisation Mondiale de la Propriété Intellectuelle (OMPI), Genève. Entretien sur le sujet des droits d'auteurs et la copie numérique, 6 mai 2008.
- Verhoeff, Jacobus, *Error detecting decimal codes*. Mathematisch centrum, Amsterdam, 1969.