



IFLA  
2005  
OSLO

## World Library and Information Congress: 71th IFLA General Conference and Council

### "Libraries - A voyage of discovery"

August 14th - 18th 2005, Oslo, Norway

*Conference Programme:*

<http://www.ifla.org/IV/ifla71/Programme.htm>

septembre 28, 2005

**Code Number:**

**065-F**

**Meeting:**

**121 UNIMARC with Information Technology**

**Fourniture de notices du catalogue de la Bibliothèque du Congrès en format MARC/XML via les protocoles SRW/SRU et Z39.50**

**Mike Taylor**

Index Data, London, UK

**Adam Dickmeiss**

Index Data, Copenhagen, Denmark

*Traduit par Annick Bernard, conservateur général honoraire des bibliothèques (France)*

### **Résumé**

*Pendant plus de dix ans, le format MARC comme structure des notices bibliographiques et la norme Z39.50 comme moyen de les localiser et de les récupérer ont favorisé l'interopérabilité dans le monde des bibliothèques. Avec la généralisation croissante d'XML, ces normes sont en cours de remplacement par MARCXML et MarcXchange pour structurer les notices et par SRW/U pour les rechercher et les obtenir. Les fournisseurs qui basculent des anciennes normes aux nouvelles doivent en général maintenir les deux types de service pendant la période de transition. YAZ Proxy applique une méthode originale pour offrir un accès SRW/MARCXML au catalogue de la Bibliothèque du Congrès, méthode qui consiste à traduire les requêtes en Z39.50 et à attaquer directement le système existant. Avantage subsidiaire, l'outil accélère significativement l'accès Z39.50.*

## De l'usage des normes pour structurer et rechercher les notices bibliographiques

L'intérêt d'appliquer les normes pour structurer les notices des catalogues est bien compris depuis longtemps. L'adoption universelle de la famille des formats MARC (*Library of Congress (2005)*) et de la syntaxe ISO 2709 (*ISO (1996)*) ont rendu possible un niveau de coopération entre bibliothèques que nous tenons aujourd'hui pour acquis, mais dont on n'ose pas encore rêver dans d'autres domaines, comme par exemple celui des collections muséographiques.

Ce qui est loin d'avoir été aussi largement reconnu, est la valeur ajoutée des normes pour la recherche dans les catalogues et le transfert des notices bibliographiques au travers des réseaux. La norme majeure apparue dans la dernière décennie en ce domaine est l'ANSI/NISO Z39.50 (*ANSI/NISO (2003)*). Approuvée pour la première fois en 1992 et développée dans la deuxième version publiée en 1995, Z39.50 a été ratifiée en 1998 en tant que norme internationale ISO 23950, et réaffirmée par NISO par l'édition révisée publiée en 2003. Les structures de données transmises au cours des sessions Z39.50 sont décrites en ASN.1, ou « Notation de syntaxe abstraite numéro 1 »<sup>1</sup> et codées à l'aide des règles de codage normalisé BER [Basic encoding rules], technologies qui toutes deux permettent un codage compact et efficace mais sont considérées dans certains milieux comme dépassées. Le protocole met à disposition un ensemble d'opérations puissantes que les clients-Z bibliographiques peuvent utiliser pour rechercher et récupérer les notices de réservoirs distants. La plupart des grandes bibliothèques, parmi elles la Bibliothèque du Congrès et la British Library<sup>2</sup>, sont accessibles par Z39.50, ce qui permet à des utilisateurs du monde entier de profiter de leur force de catalogage. Bien que la Z39.50 ait été utilisée dans beaucoup d'autres domaines, les catalogues de bibliothèque sont restés son terrain de prédilection.<sup>3</sup>

L'association d'un format MARC pour structurer les notices des catalogues et de Z39.50 pour les localiser et les télécharger rend possible un large éventail d'applications bibliographiques qui atteignent des niveaux de cohérence sémantique inégalables dans d'autres domaines. La mise à disposition - en tant que logiciels libres - d'outils de programmation tels que YAZ (*Index Data (1995)*) et d'API<sup>4</sup> pour le client telles que ZOOM (*Taylor et al. (2004)*) a donné naissance à une communauté dynamique de producteurs d'applications Z39.50.

## Les standards XML pour structurer et rechercher les notices bibliographiques et la recherche

Avec la généralisation croissante d'XML comme métaformat standard de fait pour échanger des données, le besoin s'est fait sentir de rajeunir les bons vieux formats MARC en les

---

<sup>1</sup> [Note de la traductrice] : « ASN.1 (*Abstract Syntax Notation number One*) est une norme internationale dont la vocation première est la spécification de données utilisées dans les protocoles de communication. », cela indépendamment des langages informatiques et de la représentation physique de ces données, cf. <http://asn1.elibel.tm.fr/fr/index.htm>.

<sup>2</sup> [Note de la traductrice] Le catalogue de la Bibliothèque nationale de France, BN-OPALE PLUS, est accessible via Z39.50 pour la recherche et le téléchargement de notices bibliographiques et de notices d'autorité.

<sup>3</sup> [Note de la traductrice] Lorsque c'est possible, nous utilisons le vocabulaire en usage dans les documents en français relatifs à la norme Z39.50, par exemple dans la version française du Profil Bath (version 2.0 de mars 2003) disponible sur le site de Bibliothèque et Archives Canada <http://www.collectionscanada.ca/bath/tp-bath2-f.htm#a>.

<sup>4</sup> [Note de la traductrice] API : *Application program interface*, ou interface de programmation.

convertissant en XML. La première tentative bien reçue a été le MARCXML de la Bibliothèque du Congrès (*Library of Congress (2003)*), schéma XML applicable aux notices en MARC21. Ce format étant limité au MARC américain, et donc incapable de s'appliquer aux différents formats nationaux, le Comité technique Information et documentation de l'Organisation internationale de normalisation, ISO TC46, a décidé en mai 2003 de mettre en chantier une généralisation de MARCXML qui soit applicable à tous les formats MARC. Le résultat de cette initiative, MarcXchange (*ISO (2005)*), est doué de l'estimable propriété d'être un sur-ensemble de MARCXML : tout document MARCXML est aussi un document MarcXchange valide. Les spécifications de MarcXchange en sont actuellement au stade NWI [New work item] de la certification ISO, assorti de la recommandation que la version de travail actuelle constitue la base du document soumis au vote comme projet de norme internationale, DIS [Draft International standard] <sup>5</sup>.

En même temps que l'on bascule des formats binaires historiques vers XML pour structurer les notices des catalogues de bibliothèque, on assiste à un mouvement similaire pour abandonner les protocoles binaires au profit d'un mode de transfert des notices basé sur XML. A tort ou à raison, l'utilisation d'ASN-1 et de BER dans Z39.50 est très largement perçue comme un obstacle à l'adoption du protocole. Pour répondre à cet état de choses, un travail a commencé en décembre 2000 en vue de refondre la puissante et expressive sémantique Z39.50 dans des mécanismes plus aisément compréhensibles dans le contexte actuel du traitement de l'information. Il en est résulté une famille de deux protocoles : SRW [Search/Retrieve web service] et SRU [Search/Retrieve URL]. Leurs spécifications sont administrées par le SRW Editorial board, constitué de 11 membres appartenant à des sociétés privées, des bibliothèques et des universités.

SRW (*SRW Editorial board (2004a)*) utilise SOAP <sup>6</sup> pour envoyer des requêtes du client-Z au serveur-Z et renvoyer du serveur-Z au client-Z des réponses contenant zéro ou plusieurs notices. Les requêtes et les réponses sont exprimées en XML. Plusieurs opérations sont spécifiées, chacune d'elle étant constituée d'un couple requête-réponse. Parmi eux, « SearchRetrieve » (Rechercher et Transférer), « Explain » (Expliquer, où le client demande au serveur de décrire ses possibilités) et « Scan » (Feuilleter, pour parcourir les entrées d'index). Les spécifications d'une opération « Update » (Mettre à jour) sont en cours d'élaboration et devraient être disponibles en 2005.

SRU (*SRW Editorial board (2004b)*) est sémantiquement équivalent à SRW, mais utilise un mécanisme plus simple, proche de « REST » <sup>7</sup>, pour transporter les données. Les requêtes SRU sont exprimées sous forme d'URLs avec des paramètres de requêtes qui véhiculent une information équivalente à celle que l'on trouve dans les requêtes XML correspondantes de SRW. Les réponses SRU sont identiques à celles de SRW, mais sont renvoyées directement en tant que contenu de la réponse HTTP au lieu d'être emballées dans une enveloppe SOAP comme dans SRW.

---

<sup>5</sup> [Note de la traductrice] Le New Work Item a été approuvé, et le « projet de norme internationale » devrait être prêt pour être soumis au vote vers la fin de l'été 2005 (information donnée par Tommy Schomacker, rédacteur du projet de norme).

<sup>6</sup> [Note de la traductrice] SOAP [*Simple object access protocol*] est une recommandation du W3C, le World Wide Web consortium.

<sup>7</sup> [Note de la traductrice] REST=*Representational state transfer*, ce terme a été introduit en 2000 par l'Américain Roy Fielding dans sa thèse de philosophie pour désigner de nouveaux principes d'architecture du web.

Il faut souligner que SRW et SRU sont tous deux basés sur HTTP, et peuvent à juste titre être vus comme des profils de ce protocole. Ils apportent à l'information transmise via HTTP un raffinement et une plus grande précision, de même qu'à son interprétation. Alors que HTTP ne convient pas lui-même pour la recherche et le transfert d'information, il fournit un substrat très convenable à un protocole qui favorise une puissante fonction de recherche d'information.

Si SRW a d'abord été considéré comme le meilleur des deux, l'adoption de SRU a été plus rapide et plus étendue. En général, les applications de SRU paraissent dépasser les performances des applications SRW en raison de la couche additionnelle d'analyse de la syntaxe XML (« XML-parsing ») inhérente au protocole basé sur SOAP. Puisqu'ils sont, de toute façon, sémantiquement équivalents, nous utiliserons le terme « SRW » pour désigner les deux protocoles de la famille SRW/U, sauf lorsqu'il ressort du contexte que nous donnons au terme une interprétation plus spécifique.

Une caractéristique très importante des protocoles structurés de recherche et de transfert d'information tels que Z39.50 et la famille SRW/U est le recours à un langage de recherche lui-même structuré qui permet de formuler des requêtes élaborées. Z39.50 utilise un format de requêtes binaires plutôt ésotérique appelé « Type-1 query », dont nous n'avons pas à nous occuper ici. SRW/U apporte avec lui un format de requêtes-texte à la fois intuitivement compréhensible et puissamment expressif, car capable de véhiculer toutes les requêtes qui peuvent être exprimées par la notation Type-1 de Z39.50. Il s'agit du format appelé CQL [Common query language] (*SRW Editorial board (2004c)*), que l'on peut illustrer par l'exemple ci-après :

```
dinosaur
title=dinosaur
title=(dinosaur or pterosaur) and author=martill
dc.title=*saur and dc.author=martill
title exact "the complete dinosaur" and date < 2000
name=/phonetic "smith"
fish prox/distance<3/unit=sentence frog
```

Ce langage autorise une précision beaucoup plus grande que dans les requêtes traditionnelles consistant en un « sac de mots » sans relation particulière entre eux et qui n'ont pas de lien spécifique à aucun des éléments constituant les documents objets de la recherche.

### **Trois méthodes possibles pour assurer l'accès XML**

Chaque fois que de nouvelles technologies se présentent pour en remplacer de plus anciennes, on retrouve le vieux problème de l'œuf et de la poule. Personne ne veut abandonner l'ancien standard avant que la communauté tout entière ne soutienne l'adoption du nouveau ; mais la communauté elle-même n'acceptera jamais de promouvoir le nouveau standard avant que des développeurs ne le fassent. Il y a donc inévitablement une période de recouvrement pendant laquelle l'ancien et le nouveau standard co-existent. Le défi auquel sont confrontés les fournisseurs de services consiste à garantir aux utilisateurs le minimum de perturbation pendant la période de transition, ce qui implique nécessairement qu'il faut assurer les deux services simultanément. C'est le scénario auquel la Bibliothèque du Congrès s'est résolue lorsqu'elle a pris en 2003 la décision de fournir des notices MARCXML via SRW tout en

maintenant la fourniture de notices MARC21 par Z39.50. Il y a trois manières de résoudre le problème.

La première consiste à gérer deux services séparés, à partir de deux bases de données distinctes, en utilisant des logiciels différents pour fournir au moyen de deux protocoles différents des notices formatées différemment. Si l'approche « deux bases de données – deux services » a le mérite de la simplicité conceptuelle, elle est obérée par des difficultés d'exploitation : les deux bases de données doivent être maintenues à jour l'une par rapport à l'autre, ce qui suppose que la conversion soit possible dans un sens au moins. La redondance des données est généralement à éviter pour plusieurs raisons, parmi lesquelles le gaspillage de ressources informatiques, la charge de maintenance accrue, et le risque d'incohérence entre les deux bases.

Lorsqu'elle est possible, une meilleure méthode consiste à assurer les deux services à partir de la même base de données. Les notices de la base maîtresse peuvent être structurées selon l'un ou l'autre format (ici MARC21 ou MARCXML) pourvu que la conversion au fil de l'eau dans l'autre format soit possible ; autre possibilité, les notices peuvent être stockées sous une forme complètement différente – par exemple comme elles peuvent l'être dans une base de données relationnelle - et traduites à la demande dans les deux formats à fournir. Ce modèle « une base de données – deux services » élimine la duplication des données, source d'erreur potentielle.

Le modèle « une base de données – deux services » marche bien lorsque l'on peut accéder à toutes les composantes du système, de sorte que les deux services puissent obtenir les notices de la base maîtresse. Cependant, en situation réelle, c'est loin d'être toujours le cas. Par exemple, le serveur Z39.50 actuel de la Bibliothèque du Congrès fait partie du système intégré de gestion de bibliothèque produit par la société Endeavor information systems Inc.<sup>8</sup> La base bibliographique au cœur de ce système n'étant généralement pas accessible par une interface API, il devenait donc impossible de réaliser un serveur SRW pour fournir les notices bibliographiques dans la version MARCXML. Cette difficulté conduisit à la troisième approche, une passerelle (« gateway ») jouant le rôle de serveur SRW/MARCXML et servant les requêtes produites en jouant aussi le rôle de client Z39.50/MARC21. Une fois ce schéma retenu, plusieurs avantages apparaissent sur le champ.

- Il n'est pas nécessaire de dupliquer les données.
- Nul besoin de solliciter la collaboration du système de gestion de bibliothèque existant.
- La conformité aux normes Z39.50 et MARC21 à l'extrémité interne du serveur SRW/MARCXML signifie que le même serveur SRW peut être utilisé comme frontal en face de n'importe quel serveur conforme aux normes Z39.50 et MARC21.

Ce dernier point appelle quelques commentaires, car il détermine en majeure partie la puissance du modèle. Une passerelle SRW → Z39.50 fonctionnant en surcouche d'un serveur Z39.50 existant peut être vue comme un cas particulier du modèle « une base de données – deux services », l'interface API entre le serveur SRW et la base de données étant la Z39.50 elle-même. Si l'on appliquait dans sa forme native le modèle évoqué, chaque installation du serveur SRW obligerait, à son extrémité interne, à développer une nouvelle application en fonction du SGBD gérant la base de données maîtresse. Au contraire, en utilisant le protocole Z39.50 lui-même au bout de la chaîne, la passerelle esquive ces développements multiples et tire sa puissance de la généralisation même du protocole qu'elle est destinée à remplacer.

---

<sup>8</sup> [Note de la traductrice] Ce SIGB s'appelle Voyager.

La Bibliothèque du Congrès a choisi de suivre cette troisième voie, et passé un contrat avec Index Data pour la réalisation d'une passerelle SRW → Z39.50 à installer en frontal du serveur Z39.50 d'Endeavor.

La passerelle SRW est une application plutôt originale de Z39.50, en ce sens que le protocole Z39.50 a été conçu au premier chef pour assurer à distance la communication entre clients-Z et serveurs-Z. Dans le modèle de passerelle, le client et le serveur Z39.50 résident généralement sur la même machine, ou au moins sur des machines proches connectées au même réseau local : le protocole est utilisé comme interface API uniquement. Les protocoles réseau comme Z39.50 sont bien supérieurs aux API ordinaires, car ils n'imposent pas la configuration physique des composants du logiciel. Par exemple, bien qu'il soit usuel que la passerelle soit proche du serveur Z39.50, cela n'est pas obligatoire : il est tout à fait possible à un tiers situé au Danemark d'héberger une passerelle SRW → Z39.50 qui permette à des clients SRW situés en Angleterre d'accéder à un serveur Z39.50 aux Etats-Unis.

On aurait pu s'attendre à ce que l'étape supplémentaire introduite par la passerelle dans le service SRW entraîne une certaine dégradation des performances. Dans la pratique, c'est le contraire qui se produit, en raison des mesures prises pour améliorer les performances que l'on trouvera décrites ci-dessous.

### **La passerelle YAZ Proxy <sup>9</sup>**

La passerelle SRW vers Z39.50 d'Index Data s'appelle « YAZ Proxy » (*Index Data (2004)*)<sup>10</sup>, car elle fonctionne aussi comme un protocole Z39.50, c'est-à-dire comme une passerelle de Z39.50 à Z39.50. Ce rôle était aussi important pour la Bibliothèque du Congrès que le rôle de passerelle SRW. Pourquoi est-il utile d'avoir un « proxy » Z39.50 ? Pour la même raison que les « proxys » HTTP tels que Squid <sup>11</sup> sont utiles : un « proxy » peut réduire la charge d'un serveur, améliorer la performance d'un client en mémorisant (« caching ») et en réutilisant les sessions déjà ouvertes sur le serveur, protéger celui-ci en normalisant les requêtes du client, et équilibrer la charge entre de multiples serveurs. Le YAZ Proxy installé à la Bibliothèque du Congrès fait tout cela, et procure ainsi un service Z39.50 remarquablement amélioré.

Comme il se doit, la passerelle fournit aussi des notices MARCXML via Z39.50, en traduisant au fil de l'eau les notices MARC21 de la base de données. Dans la suite de cet exposé, nous nous concentrerons sur la fonction SRW qu'elle assure (ce qui inclut le support à la fois de SRW au sens strict et de SRU).

Le principe du service SRW de YAZ Proxy est simple : pour chaque requête de recherche SRW qu'il reçoit du client-Z, il accomplit les étapes énumérées ci-après :

- il reçoit la requête SRW, qui inclut une requête en langage CQL ;
- il traduit cette requête CQL en requête Z39.50 de Type-1 équivalente ;
- il encapsule la requête traduite dans une requête de recherche Z39.50 ;
- il l'envoie au serveur de la base de données ;
- il attend la réponse ;

---

<sup>9</sup> [Note de la traductrice] On traduit habituellement « proxy » par « mandataire », et « serveur proxy » par « serveur mandataire ».

<sup>10</sup> [Note de la traductrice] Comme on peut le voir dans la référence citée, « YAZ » est depuis 1995 l'acronyme de « Yet another Z39.50 toolkit » !

<sup>11</sup> [Note de la traductrice] Squid est un serveur de mémoire cache associé au client http.

- il extrait les notices MARC de la réponse ;
- il les convertit en MARCXML ;
- il les encapsule dans une réponse SRW, et pour finir,
- il envoie celle-ci au client Z.

Vu du client-Z, ceci est un couple requête-réponse SRW parfaitement normal, et vu du serveur-Z, une réponse normale à la requête. Cependant, comme il arrive bien souvent, « le diable est dans les détails » : bien que SRW ait été conçu comme équivalent direct de Z39.50, il y a une certaine quantité de défauts de concordance entre les protocoles que la passerelle doit absorber. Nous en évoquons ci-après quelques-uns.

- Réutilisation des sessions – Pour certains serveurs Z39.50, et c'est le cas pour celui du système Voyager d'Endeavor, l'initialisation de la session est une opération très lourde, dont la longueur dépasse souvent le temps nécessaire à la recherche elle-même. Dans le cas de longues sessions, cette surcharge est amortie sur de nombreuses recherches, et n'est pas tellement pénalisante. Mais les statistiques d'utilisation montre que de nombreuses sessions ne comptent qu'une seule opération de recherche, qui prend alors dix fois plus de temps qu'elle ne devrait. Pour atténuer cet inconvénient, YAZ Proxy met en mémoire cache les sessions ouvertes sur le serveur-Z, et les réutilise pour plusieurs clients-Z. Ceci est paramétrable dans le détail. Ainsi, il est possible de spécifier : la durée maximale d'une session ouverte sur le serveur-Z, la charge maximale de données qui peuvent être envoyées et le nombre maximum de requêtes qui peuvent être satisfaites pendant la session. Cette limitation de la durée d'une session permet d'atténuer les difficultés des serveurs-Z qui souffrent d'une déperdition de ressources au cours de sessions trop longues. En complément, la passerelle peut être paramétrée pour pré-initialiser au moment de la mise en route un certain nombre de connexions au serveur-Z. Quand un client-Z qui a déjà utilisé la passerelle se reconnecte, il est mis en relation avec la connexion au serveur-Z précédemment utilisée, à condition que celle-ci soit toujours disponible. Ceci améliore la réutilisation des résultats mémorisés par les serveurs-Z qui savent le faire.
- Mise en mémoire cache des requêtes – Les simples clients-Z « amnésiques » (« stateless » = sans état, c'est-à-dire qui ne gardent pas d'état qui leur soit propre), envoient souvent des recherches identiques dans une période de temps relativement courte, par exemple pour obtenir la première page de la liste des résultats, la page suivante, une notice complète, etc. Pour de nombreux serveurs-Z, il est beaucoup plus coûteux de produire un nouveau jeu de résultats que de réutiliser un jeu de résultats existant. La passerelle accélère les recherches ainsi répétées en gardant en mémoire la dernière requête envoyée au serveur-Z, de sorte que si une requête identique à la précédente arrive, elle est transformée en requête Z39.50 « Present » (Présenter), qui fait référence au jeu de résultats existant, plutôt qu'en requête Search (Rechercher), qui crée de nouveaux jeux de résultats.
- Traduction de la requête – La part la plus complexe et la plus paramétrable de la tâche qui revient à la passerelle est de traduire les requêtes CQL venant de ses clients-Z en requêtes Z39.50 de Type-1 équivalentes. Cette complexité vient pour une part de ce que les deux langages sont eux-mêmes paramétrables du fait de l'usage de « jeux contextuels » (« context sets ») pour les requêtes CQL, et de jeux d'attributs (« attribute sets ») pour les requêtes de Type-1. Ces ensembles peuvent être élaborés par les communautés d'utilisateurs pour les rendre aptes à exprimer les concepts spécifiques à leur domaine, comme par exemple « trouver les stations météorologiques situées entre 79 et 82 degrés de longitude est ». D'une manière générale, pour mettre

en concordance une requête CQL et une requête de Type-1 équivalente, il faut connaître non seulement la syntaxe des deux langages, mais aussi les jeux contextuels en usage et les jeux d'attributs équivalents. Ce savoir, plutôt que d'être inscrit « en dur » dans la passerelle, est obtenu d'un fichier de paramètres dédié qui indique quels attributs de Type-1 doivent être utilisés pour exprimer les index, relations, caractères jokers, etc. de CQL <sup>12</sup>.

- Equilibre de la charge – La passerelle peut être paramétrée pour associer plusieurs serveurs Z39.50 à un unique serveur frontal SRW qu'elle fournit. Dans ce cas, les sessions client-Z entrantes sont dirigées vers le serveur-Z qui, à ce moment précis, compte le moins de sessions déjà initialisées à servir. Quand cette possibilité est utilisée, il revient à l'administrateur des serveurs-Z de faire en sorte que ceux-ci restent synchronisés.
- Traduction des notices – La passerelle inclut le code natif assurant la traduction de MARC21 en MARCXML. En outre, elle peut être paramétrée pour utiliser des feuilles de style XSLT pour convertir les documents MARCXML obtenus en d'autres formats XML, parmi ceux-ci Dublin Core, MODS et METS <sup>13</sup>.
- Mémorisation des notices – La passerelle peut mettre en mémoire cache les notices obtenues lors d'une recherche, et renvoyer cette copie locale si la même notice est demandée plusieurs fois. Ce mode d'utilisation est propre à certains types de clients SRW seulement. La passerelle peut aussi être paramétrée pour pré-mémoriser des notices au moment de la recherche initiale, en anticipation des requêtes « Retrieve » (Transférer) qui vont suivre.
- Journal de bord – Des mouchards peuvent être générés pour inclure soit les requêtes reçues du client-Z, soit celles envoyées au serveur-Z, soit les deux. Les requêtes peuvent être journalisées soit en détail soit en résumé.

En dehors du fichier spécifiant la conversion de CQL en Type-1 mentionné plus haut, tout le paramétrage du YAZ Proxy est spécifié dans un unique fichier de paramètres, exprimés en XML ; on utilisera XInclude si l'on veut inclure des fichiers annexes de paramètres. Les règles décrites dans le fichier de paramétrage peuvent être court-circuitées par des lignes de commande envoyées au démarrage.

Voici un bref exemple de fichier de paramètres :

```
<?xml version="1.0"?>
<proxy xmlns="http://indexdata.dk/yazproxy/schema/0.9/"
xmlns:xi="http://www.w3.org/2001/XInclude"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://indexdata.dk/yazproxy/schema/0.9/yazproxy.xsd">
  <target name="localhost" default="1">
    <url>localhost:9999</url>
    <target-timeout>30</target-timeout>
    <client-timeout>60</client-timeout>
    <keepalive>
```

---

<sup>12</sup> [Note de la traductrice] Voir : « *The Bath profile for SRW* » <http://zing.z3950.org/srw/bath/2.0/#2> . Voir aussi : « *A gentle introduction to CQL* », <http://zing.z3950.org/cql/intro.html> .

<sup>13</sup> [Note de la traductrice] Dublin Core, MODS et METS sont trois formats de métadonnées. MODS (Metadata object description schema), sous-ensemble de MARC21 pour les notices bibliographiques, et METS (Metadata encoding & transmission standard) sont maintenus par la Bibliothèque du Congrès. Celle-ci maintient aussi MADS (Metadata authority description schema) pour les notices d'autorité.

```

    <bandwidth>1000000</bandwidth>
    <pdu>1000</pdu>
</keepalive>
<limit><!-- per minute limits .. -->
    <bandwidth>2000000</bandwidth>
    <pdu>50</pdu>
    <retrieve>100</retrieve>
</limit>
<attribute type="1" value="1-11,13-1016"/>
<attribute type="1" value="*" error="114"/>
<syntax type="opac"/>
<syntax type="usmarc"/>
<syntax type="none"/>
<syntax type="xml" marcxml="1"/>
<syntax type="*" error="238"/>
<preinit>0</preinit>
<xi:include href="explain.xml"/>
<cql2rpn>pqf.properties</cql2rpn>
<query-charset>ISO-8859-1</query-charset>
</target>
<target name="*">
    <target-timeout>60</target-timeout>
    <client-timeout>30</client-timeout>
</target>
<max-clients>50</max-clients>
<log>client-requests server-requests</log>
</proxy>

```

## Conclusion

Une application originale de la norme Z39.50 rend YAZ Proxy capable d'offrir un accès SRW au contenu de serveurs Z39.50. Le paramétrage consiste à écrire un seul fichier décrivant les paramètres opérationnels et les exigences en matière d'optimisation des performances. Un fichier additionnel indique comment traduire CQL en requêtes Z39.50 de Type-1. La passerelle peut être mise en frontal de n'importe quel serveur Z39.50. Plusieurs installations sont actuellement en usage dans le monde, la plus remarquable étant celle de la Bibliothèque du Congrès. En sus de fournir des notices MARCXML à partir des notices MARC21 du catalogue de la Bibliothèque, la passerelle est aussi un proxy d'optimisation Z39.50 qui résout plusieurs problèmes connus de longue date par le serveur Z39.50 existant. YAZ Proxy est disponible librement sous licence GNU GPL <sup>14</sup> et bénéficie d'un support complet.

## Références

- ANSI/NISO. 2003. Information Retrieval (Z39.50): Application Service Definition and Protocol Specification. <http://www.niso.org/standards/resources/Z39-50-2003.pdf>
- ISO (International Standards Organization). 1996. ISO 2709: Information and documentation: format for information interchange. <http://www.iso.org/cate/d7675.html>

<sup>14</sup> [Note de la traductrice] GNU GPL : Licence publique générale GNU, ou *GNU GPL* pour *GNU General Public License* [http://fr.wikipedia.org/wiki/Licence\\_publicque\\_g%C3%A9n%C3%A9rale\\_GNU](http://fr.wikipedia.org/wiki/Licence_publicque_g%C3%A9n%C3%A9rale_GNU)

ISO (International Standards Organization). 2005. MarcXchange.  
<http://www.niso.org/international/SC4/n569.pdf>

Index Data. 1995. YAZ - Yet Another Z39.50 toolkit. <http://indexdata.com/yaz/>

Index Data. 2004. YAZ Proxy. <http://indexdata.com/yazproxy/>

Library of Congress. 2003. MARCXML. <http://www.loc.gov/standards/marcxml/>

Library of Congress. 2005. MARC Standards. <http://www.loc.gov/marc/>

SRW Editorial Board. 2004a. SRW - Search/Retrieve Web-service.  
<http://www.loc.gov/z3950/agency/zing/srw/>

SRW Editorial Board. 2004b. SRU - Search/Retrieve URL.  
<http://www.loc.gov/z3950/agency/zing/srw/sru.html>

SRW Editorial Board. 2004c. CQL - Common Query Language.  
<http://www.loc.gov/z3950/agency/zing/cql/>

Taylor, M., S. Hammer, A. Sanders, A. Dickmeiss, R. Sanderson, and A. Lav. 2004.  
ZOOM: The Z39.50 Object-Orientation Model, v1.4. <http://zoom.z3950.org/>