CHOISIR UN ENVIRONNEMENT LOGICIEL APPROPRIÉ AU CYCLE DE VIE DU DOCUMENT

Rapport du Groupe de travail
sur les applications et les logiciels

Collection en ingénierie documentaire : 3

Janvier 1999

Réalisé dans le cadre du Chantier en ingénierie documentaire
Coordonné par : Richard Parent et Nicole Boulet

Conseil du trésor
Sous-secrétariat à l'inforoute gouvernementale
et aux ressources informationnelles

REMERCIEMENTS

Ce rapport est issu de la réflexion d'un groupe de travail qui s'est penché sur les (objet)

Nous remercions de leur collaboration les personnes suivantes :

Membres du groupe de travail :

Jean Asselin, ministère de la Culture et des Communications

Marthe Baril, ministère de l'Éducation

Jean-Éric Fiorito, ministère du Conseil exécutif

Louis Houle, Secrétariat du Conseil du trésor

Yves Marcoux, École de bibliothéconomie et des sciences de l'information,
Université de Montréal

Richard Parent, Secrétariat du Conseil du trésor

Sylvain Sénécal, Hydro-Québec

TABLE DES MATIÈRES

INTRODUCTION

Dans le cadre du Chantier en ingénierie documentaire, un groupe de travail a été formé pour étudier le volet pratique lié aux applications et aux logiciels. Le premier objectif était la description des phases du cycle de vie des documents, des fonctions pertinentes pour les activités de chaque phase, et de voir comment certaines classes de documents peuvent poser des exigences fonctionnelles distinctes. La première partie du présent rapport répond à cet objectif en traçant un portrait global des aspects production et gestion de documents.

Un deuxième objectif visait à faire ressortir la convergence, accélérée par l'intranet, des applications traditionnellement liées aux logiciels d'édition de texte, de gestion de base de données, de gestion documentaire, aux « moteurs de recherche », aux outils collecticiels pour le travail en collaboration, le circuit de production et la messagerie. C'est l'objet de la deuxième partie du rapport qui offre un aperçu de cette évolution technologique et de l'importance accrue des normes qui favorisent l'infonctionnalité entre des fonctions et des logiciels spécialisés.

Le troisième objectif était de mettre en lumière les grandes options et de produire une liste des principaux critères de choix dans la panoplie des logiciels disponibles. Le domaine étendu des fonctions applicables au document et la grande variété des besoins en contexte d'implantation rendent périlleux un tel exercice. Des lignes générales sont quand mêmes proposées dans la troisième partie du présent rapport en les situant dans le nouveau cadre normatif de l'Internet. Les choix sont aussi situés dans le contexte de l'implantation d'une infrastructure et de services communs pour l'Inforoute gouvernementale visant à accompagner les ministères et les organismes dans le développement de leurs applications : mise en place d'un moteur de recherche des sites Web gouvernementaux, développement d'un répertoire gouvernemental, d'une télémessagerie gouvernementale et d'une infrastructure à clés publiques, sont des exemples d'outils communs proposés comme éléments de solution pour combler l'écart entre le mode de traitement actuel des documents et une ingénierie documentaire moderne.

Les deux dernières parties du rapport présentent et expliquent les principales normes de l'Internet qui permettent l'interfonctionnalité des logiciels reliés aux documents. Une attention toute particulière est accordée aux normes les plus directement pertinentes pour les documents, soit celles concernant les métadonnées des documents ainsi que celles ayant trait à la structure logique du contenu des documents. La place importante donnée à l'explication du nouveau cadre normatif des documents constitue en soi une indication que les fonctions logicielles deviennent de plus en plus interchangeables dans un monde où les données sont partagées et échangées grâce aux normes guidant leur syntaxe et leur transfert.

CHAPITRE 1

CYCLE DE VIE DU DOCUMENT

Le cycle de vie du document est formé de deux processus distincts comportant chacun leurs objectifs spécifiques. Il s'agit de la production (axe horizontal) et de la gestion des documents axe vertical (voir figure 1).

Dans l'axe horizontal, le document est considéré comme un objet à construire parce qu'il est, pour l'utilisateur, un moyen de réaliser une activité quelconque dans un processus de travail. On s'y intéresse donc aux différentes fonctions nécessaires afin qu'un document soit produit, validé, distribué et utilisé à l'intérieur d'un processus de travail particulier.

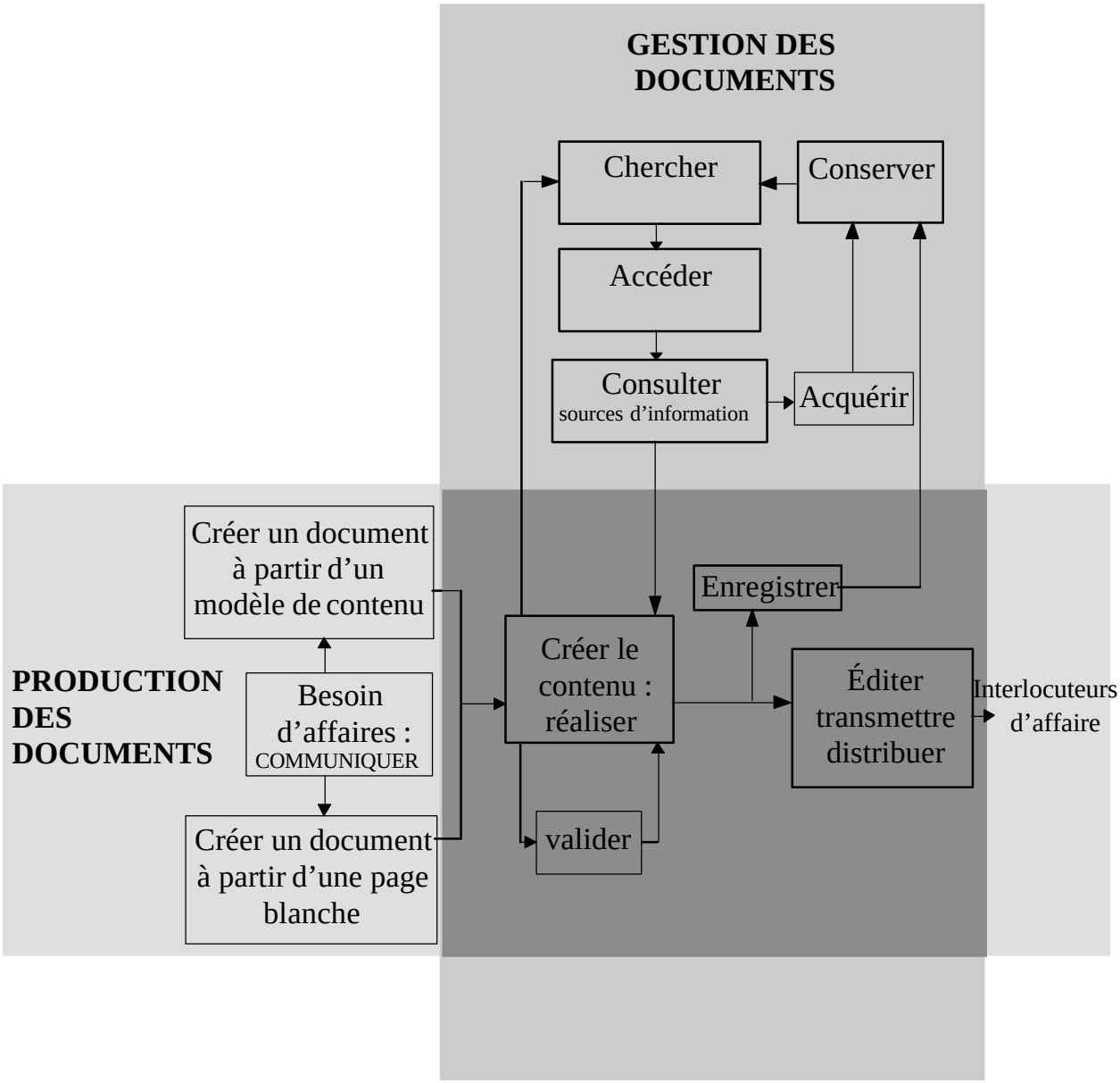
Les besoins d'affaires se traduisent souvent par l'obligation de créer un document. Dans cette perspective, le document est *un produit à construire*. Sont considérées ici les différentes composantes qui doivent être construites et assemblées en un tout cohérent, c'est-à-dire la structure, le contenu, la forme, ainsi que le support physique du document. Le contrôle de cette production est aussi un souci majeur de cet aspect du monde documentaire. Les étapes de la production et de l'assemblage de ces différents éléments en une chaîne de traitement particulière ouvrent la porte à des produits informatiques de gestion de processus, d'événement, de travail en collaboration, de distribution, de circulation et d'approbation de documents. Les logiciels de soutien à la production des documents s'occupent du travail en cours sur le document pour le bénéfice d'un créateur. Ces logiciels contribuent à implanter un cadre de production et de mise à jour de documents souvent strictement « contrôlés » par des politiques et des procédures fréquemment révisées pour suivre l'évolution des processus de travail.

La conception du contenu du document regroupe des activités pour lesquelles le document est considéré uniquement comme un contenu informatif à concevoir et à interpréter, par exemple lorsque quelqu'un, pour répondre à un besoin d'affaires, crée un document de toutes pièces sur une page blanche. Essentiellement, ces activités traitent de problématiques de contenu (souvent textuel) où l'usager requiert une assistance dans l'utilisation et l'interprétation de ce contenu afin d'en produire un autre. Ce regroupement d'activités comprend la conception, l'utilisation, la recherche de contenu, l'analyse, l'assemblage de données, etc. Ces activités sont principalement liées au domaine de l'analyse de texte assistée par ordinateur, et englobent toutes les activités associées à l'accès au contenu informatif des documents via une certaine assistance informatique.

La figure 1 indique trois intrants possibles lors de la création d'un document : l'utilisation d'un modèle de contenu (par exemple un formulaire), la consultation de sources d'information existantes et la validation interactive du modèle de contenu ou de l'aspect linguistique. Une fois créé, le document est édité (en fonction possiblement de modèles d'édition) et inséré dans un média quelconque avec l'encodage de données approprié pour sa transmission (messagerie) ou sa distribution (page Web, canal HTTP, papier).

Une fois qu'il a été créé, le document considéré comme extrant de ce cycle est géré selon des règles établies en fonction de son utilisation dans le processus de travail. Dès qu'il est enregistré, il est pris en charge par la gestion des documents institutionnels.

Figure 1. Cycle de vie du document : entrecroisement des processus de production et de gestion des documents



1.1 La gestion du document

Les activités liées à la gestion du cycle de vie des documents sont : les activités de description et d'indexation selon des fiches de référence afin de repérer les documents ; les activités de prêt et d'acquisition de documents pour permettre leur consultation et constituer un fonds documentaire ; les activités de conservation afin de respecter la valeur administrative, légale, historique ou de référence des documents et les activités d'entreposage temporaire ou permanent des documents.

La gestion de document, présentée dans l'axe vertical de la figure 1, est ce qui est communément appelé la gestion de documents institutionnels. Cette expression a le mérite d'englober toutes les activités documentaires relatives aux documents déjà produits et possédant une quelconque valeur (administrative, légale, historique, de référence, etc.) pour les organisations.

L'axe vertical correspond à des activités spécialisées du type gestion des documents administratifs, gestion des archives ou gestion de la documentation *qui sont directement centrées sur la gestion d'un processus documentaire et non sur la gestion du processus de l'utilisateur*, cette dernière étant plutôt associée à l'axe horizontal. Ainsi, l'axe vertical a pour finalité de conserver et de rendre disponible un document, par ailleurs déjà produit, selon des normes de description spécifiques d'un domaine d'activité donné. Ce processus se traduit par l'ajout de métadonnées qui facilitent l'activité de recherche sur le document.

La gestion des documents institutionnels se positionne au premier plan en fonction d'une problématique de la mémoire de l'organisation et place donc au cœur de ses préoccupations la fonction conservation et gestion de la conservation des documents.

Les activités de gestion documentaire présentées à la figure 1 peuvent se définir de la façon suivante.

Chercher : Rechercher et repérer un document selon certains critères, fournis en partie par l'indexation issue des métadonnées créées ou encore directement explicités dans le contenu des documents. De plus, cette activité inclut la formulation d'une demande de recherche, la recherche comme telle ou des activités de veille informationnelle.

D'une façon plus spécialisée et proactive, cette recherche inclut aussi l'analyse, la prévision et la validation des besoins de la clientèle ainsi que l'analyse des documents, notamment l'évaluation qualitative et quantitative des fonds.

Accéder : De façon générale, cette activité vise autant un document sollicité qu'un document reçu sans sollicitation. Elle concerne la capacité physique de manipuler un document. Il s'agit fondamentalement de définir et de gérer les droits d'accès aux documents. Dans un contexte électronique, le transfert de documents actifs et semi-actifs, le versement de documents d'archives peuvent n'être que des changements dans les droits d'accès à ces documents.

Consulter : Activité cognitive relative à la lecture du document, à la prise en compte de son contenu signifiant.

Acquérir : Acquisition d'un document provenant d'un fournisseur quelconque au moyen de l'achat, de l'emprunt, de l'abonnement, ou du téléchargement d'un document publié sur le Web.

Distribuer : Activités relatives à la diffusion, à la transmission, à la circulation et au prêt des documents. Elles consistent à désigner les documents à distribuer, à établir la stratégie de distribution, la liste des destinataires, à préparer la transmission, à transmettre les documents et à gérer les accusés de réception.

Conserver : Activités relatives à la sélection des documents à être conservés selon un calendrier de conservation ou d'autres moyens, à la description de leurs métadonnées, à leur préparation pour le rangement physique et le rangement comme tel. Elles comprennent en outre la disposition des documents non sélectionnés pour la conservation à court, moyen ou long terme, c'est-à-dire leur destruction, leur vente ou leur donation, ainsi que la gestion des versions des documents.

1.2 La combinaison des processus de production et de gestion de documents : un besoin

La production et la gestion de documents doivent pouvoir être combinées de façon à permettre à un groupe d'utilisateurs en situation de travail courant de mettre en commun leurs efforts et leurs ressources informationnelles afin de réaliser au bon moment, selon la bonne séquence de travail et à l'aide des intervenants pertinents, les différents documents nécessaires à leurs activités.

Plusieurs logiciels de soutien à la production de documents offrent la gestion des versions de document. Mais cette gestion sert beaucoup plus à l'utilisateur dans son besoin de conserver un document en tant qu'il est relié à un autre document dans une relation de modification de son contenu signifiant à l'intérieur d'un processus de travail. Ce besoin de conservation n'est pas lié à une problématique de préservation d'une valeur institutionnelle (corporative), il correspond plutôt à un besoin de simple sauvegarde qui trouve sa source dans l'utilisation effective d'un document dans un cycle de création-réalisation-» consommation ».

La nature des encadrements et des contraintes appliqués à cette activité de conservation est différente selon que celle-ci est considérée du point de vue des besoins de l'ensemble de l'organisation ou en fonction de l'efficacité du document dans l'action de l'utilisateur.

Les logiciels de soutien à la production de documents dans un processus de travail ne sont pas des outil de gestion électronique de documents administratifs, de référence, d'archives, etc. Le besoin de conservation institutionnel réside dans le contrôle des documents et des encadrements de gestion qui leur sont appliqués.

Entre l'archiviste, le gestionnaire de document, les créateurs de contenu et le gestionnaire des logiciels de soutien aux applications de production et de gestion, il existe une incertitude sur les responsabilités respectives. Les gestionnaires de documents institutionnels deviennent responsables des documents une fois que ceux-ci sont enregistrés, ainsi que des différents encadrements qui permettent ce traitement.

Par contre, les logiciels de soutien à la production, quant à eux, appartiennent typiquement aux utilisateurs ; leur implantation comme leur utilisation ainsi que les documents qu'ils visent à traiter sont directement sous le contrôle des « propriétaires de processus de travail ». Il existe donc un triple enjeu, technologique, fonctionnel et administratif, dans le mariage des applications de production et de gestion de documents, ainsi que dans leur contrôle.

CHAPITRE 2

ÉVOLUTION TECHNOLOGIQUE

Le domaine du travail avec le document exige une grande variété de fonctions, tant pour leur production que pour leur gestion, comme l'illustre la figure 1. Des logiciels variés ont été utilisés dès les débuts de la micro-informatique pour produire du texte, des dessins, des statistiques et des banques de données : WordPerfect, Paint, Excel, Dbase sont des noms évocateurs de cette première manière. Ces logiciels ont été progressivement intégrés dans des « suites » comme Lotus et Office il y a quelques années, facilitant ainsi la création de documents. Mais de façon générale, dans l'architecture client-serveur du début de la décennie, il y avait des **logiciels distincts** pour ces fonctions de création de documents, la messagerie, l'imagétique, le repérage textuel, la gestion documentaire et le circuit de production.

La concurrence entre ces créneaux de l'offre a conduit à une **intégration croissante** dans des « collecticiels » intégrant ces divers volets en un produit. Au moment même où certains de ces produits devenaient réalité, l'essor des intranets et de ses produits à coût moindre est venu modifier la perspective. En effet, le potentiel visé n'est plus une simple interfonctionnalité des fonctions au cours du cycle de vie du document **au sein d'une organisation** : ce marché est en effet bien desservi par des **collecticiels « propriétaires »** tels que ceux offerts par Microsoft et Lotus. Mais la promesse du **commerce électronique interentreprises** a rendu essentielle l'existence de **normes ouvertes** telles celles de l'ISO, de l'Internet Engineering Task Force (IETF) et du World Wide Web Consortium (W3C). Les avantages des produits avec des formats « propriétaires » intégrés continuent donc de s'éroder à mesure que le marché y pousse les nouveaux outils de développement basés sur des formats standards. Il devient évident qu'ils doivent se conformer à des protocoles et à des normes ouvertes de description des données et des métadonnées qui deviennent plus tentaculaires que même les « suites » bien établies. En effet, développer avec un formalisme quelconque particulier aux produits de Lotus, Microsoft, Netscape ou Novell devient de plus en plus onéreux en créant des îlots difficiles d'accès dans l'information à partager, à échanger ou à diffuser.

L'intranet, qui consiste en un déploiement de ressources qui utilisent, sous protection, les protocoles de l'Internet, contient par le fait même des ressources qui facilitent l'interfonctionnalité des logiciels reliés aux documents. Le format HTML a été un triomphe, la syntaxe XML est une amélioration importante mieux en mesure de représenter la structure logique des documents et leurs métadonnées diverses, dont celles concernant les fichiers physiques de données. L'intranet est un terrain propice au déploiement d'applications de circuit de production en tirant profit du fonctionnement formel de certains processus d'application (par exemple, validation et vérification de déclarations fiscales), tout en soutenant les processus plus souples de groupes de travail interministériels par exemple.

Il y a d'autres initiatives qui favorisent l'interfonctionnalité, comme la convention ODMA (*Object Document Management API*) et ses « composantes conformes » qui permettent de camoufler plusieurs fonctions de gestion documentaire derrière des menus de logiciels de traitement de texte. Pour certains développeurs de logiciels, c'est un premier pas vers les normes

ouvertes. Ce genre de convention est fortement relativisé dans le contexte global des normes ouvertes, mais c'est une convention particulière quand même utile.

L'idée d'une ingénierie documentaire est celle d'un ensemble de ressources aménagées en services, lesquels sous-tendent l'organisation des diverses fonctions pouvant s'appliquer au cours du cycle de vie du document. La possibilité d'y parvenir est aujourd'hui accrue par le protocole XML (*Extensible Markup Language*) qui, en facilitant la création de documents structurés, permet une interfonctionnalité plus profonde que celle que procurent les seules métadonnées en rendant également partageables les données internes du document au moyen de leur balisage et de leur marquage en éléments logiques. Les « suites » servant à la création de documents (Microsoft, Lotus) ne conservent leur intérêt pour de nouveaux clients qu'en se conformant aux normes ouvertes.

Importance des normes relatives au document

Il a été souligné précédemment dans ce rapport que les normes deviennent de plus en plus importantes pour les applications documentaires, qu'elles soient vues sous l'angle de la production des documents ou sous celui de la gestion des documents institutionnels. En effet, les normes permettent de tirer parti de fonctionnalités éclatées parmi plusieurs progiciels distincts et de les orchestrer pour répondre à des besoins spécifiques d'une organisation ou d'un contexte organisationnel, rendant ainsi possible l'élaboration sur mesure de solutions complètes qu'aucun progiciel individuel ne saurait offrir.

Mais l'importance des normes pour les applications documentaires va beaucoup plus loin que l'exploitation de cet éclatement de fonctionnalités. En effet, l'essence même des applications documentaires réside dans la communication entre individus, et l'on sait que les communications entre systèmes informatiques exigent la compatibilité des systèmes sur un très grand nombre de points (matériel, formats, protocoles, etc.). Or, la façon la plus robuste d'atteindre cette compatibilité est la normalisation.

Un avantage fondamental associé à la normalisation est la réutilisation des données rendue possible par les *documents structurés*. Un document structuré est un document électronique qui porte en lui-même des données explicites (généralement, sous forme de *balises*) concernant sa propre structure logique. La raison pour laquelle les documents structurés permettent une plus grande réutilisation des données que, par exemple, les bases de données relationnelles ou les formats traditionnels de traitement de texte, est que les balises d'un document structuré associent à ses différents segments une sémantique semblable à celle des champs d'une base de données, mais en laissant l'information *dans son emplacement naturel* et en respectant la flexibilité structurelle exigée par la nature même du document. En comparaison, les bases de données relationnelles imposent une structure trop rigide aux données pour pouvoir représenter convenablement les documents, et les formats traditionnels de traitement de texte ne représentent que très peu d'aspects sémantiques des documents.

Dans une application documentaire, il est clair que la finalité du repérage d'un document est, sous une forme ou une autre, sa réutilisation : soit pour créer un nouveau document, soit pour en faire une lecture qui peut être très différente de la lecture originelle. Toute méthodologie facilitant la réutilisation des documents est donc d'un grand intérêt pour les applications documentaires.

Les documents structurés sont habituellement associés à la normalisation parce qu'une des façons les plus avantageuses et les plus courantes d'implanter les documents structurés est d'adopter un format normalisé de documents structurés (comme XML, SGML ou HTML). En fait, il existe très peu de formats de documents structurés qui ne soient pas normalisés (probablement, parce que les documents structurés sont orientés vers la réutilisation, qui n'atteint son plein potentiel que dans un contexte multi-plate-forme et multi-producteur).

CHAPITRE 3

FONCTIONS ET CRITÈRES POUR LE CHOIX DE SOLUTIONS LOGICIELLES

3.1 Contexte technologique

Il est incontestable que le marché des logiciels est turbulent et que les façons de faire sont en changement constant. Rétrospectivement, il y a eu la période IBM ou des « mainframes », puis la période Microsoft ou du micro-ordinateur et de la bureautique. L'inforoute domine de plus en plus la scène. Alors qu'il y a deux ans encore la question du choix d'une « suite logicielle » (traitement de texte, chiffrier, dessin) nous plaçait devant divers produits commerciaux incompatibles entre eux, aujourd'hui les normes ouvertes sont portées par le raz-de-marée Internet et modifient nos façons de percevoir le choix des outils de développement et d'utilisation de l'information. L'effet des protocoles ouverts de l'inforoute est d'amener une vision de l'informatique centrée sur les données.

La multiplicité des fonctions dans un seul logiciel est un atout fortement relativisé par les normes ouvertes de l'Internet puisque l'interfonctionnalité tend à être mieux assurée via les données et les métadonnées que par toute intégration de fonctions sur des formats privés ou sur le système d'exploitation Windows (Novell, Lotus, Microsoft).

Si on n'a plus à attendre d'un seul logiciel qu'il puisse accomplir toutes les fonctions, il n'en demeure pas moins qu'il est plus simple pour les développeurs d'utiliser certaines fonctions déjà rassemblées en modules cohérents. Ce serait le cas de plusieurs des fonctions liées à la gestion documentaire (indexation des mots du texte et indexation par attributs en vue d'un repérage combinant les deux types d'argument dans une requête).

Les logiciels de gestion des documents électroniques sont généralement des produits fonctionnant en réseau et qui se relient aux commandes Ouvrir et Sauvegarder des logiciels de création de documents (traitement de texte, chiffrier, messagerie). Ils permettent aux utilisateurs d'établir un profil de métadonnées pour un document sans quitter leur outil de création. Ces documents sont souvent sauvegardés sur un serveur de fichiers de réseau local, alors que les métadonnées sont placées dans une base de données centrales d'où est contrôlé l'accès aux documents, le suivi des révisions des documents, et qui fournit le repérage en texte intégral et par métadonnées.

3.2 Liste de fonctions et critères

Les fonctions et les critères recherchés pour ce genre de logiciel peuvent être décrits ainsi :

- **Intrant/extrant :** l'important est la capacité de travailler avec les normes ouvertes, en particulier le protocole XML pour le document lui-même, le protocole RDF pour les métadonnées, le modèle objet du document (DOM) pour les interfaces de programmation d'application (API) et les langages de définition d'interface (IDL), et le protocole LDAP pour les échanges avec les services de répertoire. Divers convertisseurs entre formats de document peuvent faire partie du logiciel ou simplement venir le compléter (par exemple Omnimark).

- **Sauvegarde/enregistrement/conservation :** capacité de simple sauvegarde sous responsabilité personnelle d'un document ; capacité d'enregistrement d'un document institutionnel par un utilisateur ou une application en le situant dans le Plan de classification des documents ; capacité de rendre opérationnel le Calendrier de conservation ; capacité de reprendre les métadonnées transmises par les outils de création ou de transmission/diffusion des documents en même temps que ceux-ci ; pour soutenir les révisions, capacité de gestion des versions d'un document. Les documents institutionnels sur support papier peuvent être enregistrés selon le même profil de métadonnées que les documents sur support électronique. Cet enregistrement se fait avec la même interface de saisie, mais tout en sollicitant en plus de l'utilisateur les métadonnées qui auront été reçues automatiquement dans le cas des documents électroniques.

- **Indexation/repérage/consultation :** opérations adéquates d'indexation des documents sauvegardés et enregistrés : à la fois indexation de tous les mots du texte et indexation en fonction des attributs présents dans le profil de métadonnées ; en outre, cette indexation doit pouvoir être effectuée dans le respect du niveau de confidentialité attribué à chaque document ; capacité de requêtes simples et élaborées pour le repérage dans les documents, et capacité de sélection de sujets dans le Plan de classification et de termes dans un thésaurus ; affichage des résultats conformément aux protocoles pertinents (LDAP, ISO 23950) et sous le contrôle des paramètres par défaut ou ajustés par les utilisateurs.

- **Collaboration/circuit de production :** d'une part, flexibilité d'arrangements interpersonnels ou de groupes en réseau pour la rédaction en collaboration, la discussion, dans lesquelles l'initiative et la spontanéité sont des valeurs positives ; d'autre part, capacité d'imposer une gestion centralisée du contrôle de la circulation des documents entre des rôles et des organisations (circuit de production) pour des étapes d'analyse, de révision, d'approbation, etc.

- **Interface d'utilisation :** capacité de livrer ses fonctions via l'une des interfaces de navigation courantes du Web (Netscape, Microsoft ou autre). Cette interface a l'immense avantage d'offrir les divers algorithmes de hachage et de chiffrement qui servent à transmettre ou à enregistrer confidentiellement un document, à garantir l'intégrité d'un document et à produire la signature numérique d'un document électronique.

Les critères à l'intérieur des fonctions doivent être adaptés ou pondérés selon le contexte d'utilisation. Les besoins documentaires varient beaucoup, et surtout l'environnement de

support est variable. L'annexe 1 fournit un énoncé détaillé d'exigences fonctionnelles pouvant figurer dans la définition d'un cahier de charges.

3.3 Infrastructure de l'inforoute gouvernementale

Si ce n'était de la mise en place, amorcée en 1998, d'un service de répertoire gouvernemental pour l'inforoute, beaucoup de fonctions supplémentaires seraient requises dans chaque intranet, en particulier évidemment en ce qui concerne le contrôle de l'accès, l'établissement de listes, la disponibilité des modèles de contenu et autres structures logiques pertinentes. Ce service de répertoire s'explique notamment par l'accroissement des ressources en réseau qui doivent être recensés, décrites et contrôlées et dont l'accès doit être connu. L'économie des moyens, les exigences de la sécurité et la recherche de la meilleure qualité commandent sa mise en place parmi les services communs d'une infrastructure de réseau. Ce répertoire permettra de réduire la complexité du logiciel documentaire tout en permettant une plus grande spécialisation de ses fonctions spécifiques. Divers services rendus par ce répertoire couvriront plusieurs besoins communs à un grand nombre d'applications :

- trouver des renseignements sur les personnes, par exemple l'adresse des destinataires de messages envoyés, distribuer un document à un groupe circonscrit de personnes, vérifier grâce à son certificat de clé publique, si une personne a bel et bien signé un message ;
- permettre de contrôler l'accès aux ressources, par exemple aux documents, en bloquant/autorisant l'accès des personnes, des classes d'utilisateurs, à des classes de documents ou à des parties spécifiques de documents, ou certaines opérations sur ces documents ;
- enregistrer les métadonnées pour les documents institutionnels aux fins de conservation, d'archivage, ou de diffusion ;
- rendre accessibles divers modèles de contenu pour des types de document (DTD : définitions de types de document).

Des distinctions importantes doivent être établies entre trois modes d'usage des SGBD (systèmes de gestion de base de données) pour emmagasiner les documents à gérer et l'information utile à leur gestion. Dans une architecture ouverte comme l'inforoute gouvernementale, il est pertinent et utile de définir et de distinguer trois fonctions importantes des SGBD :

- **Recueil** (*database, repository*) : mécanisme d'emmagasinage d'une collection de fichiers, les fichiers du contenu des documents. Un recueil emmagasine des *documents concrets.*

- **Référentiel** (*referential, repository*) : mécanisme d'emmagasinage pour des modèles et des outils permettant de traiter les données modélisées. Dans le cas des documents, ce sont les modèles de contenu ou structures logiques de documents (DTD) qui y seront enregistrés. Des modèles d'échange (XML/EDI) et d'édition (XLL, CSS) pourraient aussi y être emmagasinés pour les étapes ultérieures de la production des documents. Cette partie du référentiel est basée sur des *documents abstraits*, soit en général des DTD (définitions de types de document). Ces modèles peuvent servir à la création, à la validation, au traitement, à l'échange et à la diffusion en permettant d'y faire respecter des règles.

- **Répertoire** (*directory*) : modèle de base de données répartie pour enregistrer systématiquement plusieurs classes d'objets informationnels fréquemment consultés, pour se faire référer aux ressources en consultant les entrées définies pour ces objets. Plusieurs sont pertinents pour l'ingénierie documentaire. Un serveur de répertoire est une base de données optimisée pour fournir des références (par opposition à son usage comme un recueil des documents eux-mêmes). En outre, des interfaces de consultation spéciales donnent accès à certaines ressources, ce qui est le cas des Pages vertes pour l'accès aux documents par repérage dans les métadonnées.

Un répertoire qualifie une forme d'organisation et d'utilisation d'une base de données qui est constituée, de façon intégrée ou séparée, de deux formes proches d'organisation et d'utilisation d'une base de données : recueil et référentiel. Le référentiel peut être une partie du répertoire puisque ses modèles se représentent comme toute autre classe d'objets. Le répertoire se distingue en effet à titre de mécanisme d'emmagasinage des attributs en fonction des classes d'objets, en particulier les personnes, les organisations, les équipements, les documents et les modèles. Il est organisé en fonction de *noms* uniques donnés aux entrées. Le noyau essentiel d'un service de répertoire est d'offrir un service de résolution d'*adresses* (ou pointeurs) ou d'autres éléments d'information orientés sur le fonctionnement en réseau via des services aux applications. Pour la classe d'objet Document, ce sont les références (groupement de métadonnées) qui sont enregistrées et pointent sur les adresses des documents. Ces documents eux-mêmes se trouvent dans des recueils qui sont généralement des SGBD implantés à moindre coût (moins de mémoire vive et plus d'espace de stockage).

3.4 Ressources pour aider à faire des choix

Certains sites spécialisés du Web font office de centres d'information sur les logiciels. Un des plus intéressants, mis à jour en 1997, se trouve à http://www.cs.uku.fi/~kuikka/systems.html. Une liste de catégories y est offerte pour quelque 200 logiciels relatifs aux documents structurés ; cette liste est reprise ici avec notre traduction en français :

- traitement de texte (*text editor*)
- création de document structuré (*structure editor*)
- logiciel d'édition et de mise en page (*desktop publishing software, formatter/page layout program*)
- moteur de recherche ou de repérage (*search program*)
- logiciel de diffusion électronique (*electronic delivery too*l)
- navigateur ou fureteur (*browser*)
- base de données (*database*)
- base de données textuelles (*text database*)
- base de données de documents structurés (*structured text database*)
- base de données de documents (*document database*)
- interface de base de données (*database front-end*)
- convertisseur (*conversion program*)
- parseur ou analyseur (*parser*)
- outil d'API (*API tool*)

– outil de création de DTD (*DTD tool, structure design tool*)
– outil d'édition par style (*DSSSL tool, layout design tool*)

Le site le plus complet et constamment tenu à jour est celui de Steve Pepper sur les outils SGML et XML à http://www.infotek.no/sgmltool/guide.htm. Il établit un nombre plus restreint de catégories :

– édition et mise en page (*edition and composition*)
– diffusion électronique (*electronic delivery*)
– emmagasinage et gestion de document (*document storage and management*)
– élaboration des modèles (*control information development*)
– parseurs et moteurs (*parsers and engines*).

3.5 Saisie et gestion des métadonnées dans le cycle de vie

Comment gérer la saisie et le stockage des métadonnées au cours de la vie d'un document ? Clairement, il y a une panoplie de solutions possibles, allant de la gestion entièrement manuelle des métadonnées (irréaliste, sauf dans des cas très particuliers) jusqu'à la gestion entièrement intégrée aux formats de documents, ainsi qu'aux logiciels de production et d'exploitation des documents. Entre les deux s'étend pratiquement un continuum de solutions dans lesquelles le soutien technologique occupe une place plus ou moins importante.

Le but de la présente section est de montrer comment un support technologique plus ou moins sophistiqué peut être complété par des protocoles à l'intention des humains pour assurer une gestion satisfaisante des métadonnées. Nous illustrerons ces possibilités par deux scénarios, utilisant des outils technologiques de deux niveaux de sophistication différents. La présentation des deux scénarios tient pour acquis que l'on a déjà établi le schéma de métadonnées avec lequel on veut travailler (*i.e.*, l'ensemble des attributs et les différentes valeurs possibles pour chacun). Un cas concret sera ensuite présenté, qui combine des éléments des deux scénarios.

Premier scénario

Dans le premier scénario, les documents et les métadonnées (celles provenant des rédacteurs) sont saisis avec Word 97. Le « cadre de métadonnées » utilisé (si l'on peut parler ainsi) est le mécanisme des « propriétés des documents » de Word. L'ensemble des propriétés gérées par Word est personnalisé de façon à se conformer le mieux possible au schéma de métadonnées adopté. Ainsi, le « type » d'une propriété, au niveau de Word, sera choisi de façon à exercer le meilleur contrôle possible sur les données saisies, bien que ce contrôle soit de toutes façons très limité. Concrètement, cela signifie que pour une métadonnée qui serait entièrement numérique, on choisirait le type de propriété « nombre », plutôt que « texte », de façon à ce que le logiciel effectue une certaine validation.

Il est possible que le schéma de métadonnées choisi doive être adapté pour se conformer aux possibilités du mécanisme des propriétés de Word. Ainsi, des métadonnées qui seraient

structurées hiérarchiquement devraient d'abord être « aplaties » de façon à entrer dans la structure strictement « plate » des propriétés des documents de Word.

On créera un modèle de document qui intégrera les propriétés personnalisées correspondant au schéma de métadonnées choisi. Ce modèle de document sera obligatoirement utilisé par les rédacteurs. On pourra en profiter pour inclure dans le modèle une feuille de styles qui puisse être utilisée pour identifier certains éléments de la structure logique des documents ; ainsi, advenant une conversion subséquente dans un format de document structuré (comme XML), ces éléments de structure pourraient être automatiquement détectés et exploités pour faciliter la conversion. Dans un protocole de rédaction auquel les rédacteurs seraient tenus de se conformer, on présentera les différents styles et les circonstances dans lesquelles ils doivent être utilisés.

Il n'y a aucune gestion automatisée de la rédaction en collaboration. Si plusieurs coauteurs travaillent sur un même document, ils se l'échangent par courriel (ou autrement via réseau), et si des métadonnées doivent être ajoutées ou modifiées, elles le sont explicitement par l'entremise de la fonction d'édition des propriétés du logiciel.

On prendra soin d'activer l'option « Demander les propriétés du document » sur chaque copie de Word utilisée par les rédacteurs, de sorte que le logiciel propose la saisie des propriétés à chaque sauvegarde d'un nouveau document. Malheureusement, il n'y a aucun moyen de s'assurer que le rédacteur saisit effectivement toutes les métadonnées obligatoires, et il faut donc ici pallier cette impossibilité en mentionnant, dans le protocole de rédaction, qu'il est de sa responsabilité de saisir cette information.

Puis, la fonction d'indexation en texte intégral de la « Recherche accélérée » d'Office est utilisée pour indexer le texte intégral des documents (incluant les métadonnées) et la fonction de recherche d'Outlook est utilisée pour le repérage. Cette fonction permet de restreindre la recherche à certaines propriétés seulement, et possède même certaines capacités limitées de recherche booléenne.

Une disposition judicieuse des documents et des index sur le réseau évite la duplication des index et rend la recherche accessible à tous.

Lorsqu'un document passe du statut de document en élaboration au statut de document institutionnel, les permissions réseau sont modifiées pour le mettre sous la responsabilité du service de gestion des documents et certaines métadonnées sont ajoutées et/ou modifiées pour tenir compte du nouveau statut du document. Le document peut aussi changer physiquement de serveur.

Second scénario

Dans le second scénario, un logiciel de gestion documentaire sur intranet (comme LiveLink) est utilisé. Le logiciel inclut des fonctions de circuit de production aptes à soutenir la rédaction en collaboration des documents. Les documents sont en format XML et les métadonnées provenant des rédacteurs sont incluses dans les documents eux-mêmes ; elles sont validées dès la saisie par des procédures personnalisées intégrées à l'éditeur XML utilisé. Un filtre d'importation des

documents (développé sur mesure puis intégré au logiciel de gestion documentaire) extrait les métadonnées des documents et les inscrit dans la base de données documentaire du système, et ce, après chaque modification d'un document. Les documents sont aussi indexés en texte intégral en tenant compte de leur structure logique (*i.e.*, le contenu de différents éléments XML se retrouve dans différents index). La base de données contient également des métadonnées de gestion, entre autres sur les accès permis aux documents.

La recherche de documents s'effectue avec la fonction de recherche habituelle du logiciel. On a alors accès aux index correspondant aux différentes métadonnées, de même qu'à la recherche en texte intégral. Tous les changements et/ou ajouts aux métadonnées au cours de la vie du document sont effectués dans la base de données documentaire du logiciel, conformément aux permissions de chaque usager.

Comme dans le premier scénario, lorsqu'un document passe du statut de document en élaboration au statut de document institutionnel ; les différentes métadonnées indiquant le nouveau statut du document et le fait qu'il est maintenant sous la responsabilité du service de gestion des documents sont ajoutées et/ou modifiées. Le document peut aussi changer physiquement de serveur.

Si ce second scénario semble plus simple que le premier, c'est qu'il bénéficie d'un plus grand soutien technologique. Les différentes opérations de gestion des métadonnées sont donc mieux intégrées aux autres opérations sur les documents.

Cas concret

« The Wall Street Journal Interactive Edition » (www.wsj.com) offre quotidiennement son contenu à ses 200 000 abonnés avec un succès reconnu. La production qui sous-tend son site Web est basée sur un système misant sur Java, les scripts en langage PERL, SGML/XML, ainsi que sur le logiciel Word 6.0 et son interface sur mesure. L'outil Konstructor d'Omnimark sert à transférer les extrants en RTF de Word en documents XML. Ce logiciel était connu de tout le personnel de rédaction et la décision fut prise de bâtir avec cet outil d'utilisateur. Au moyen de scripts et de macrocommandes, plus de 95 % du code SGML/XML est produit par les auteurs sans qu'ils aient à s'en préoccuper.

En 1998, ce n'est peut-être plus la meilleure solution à cause de la multiplication des outils pour créer des documents XML, mais on peut voir qu'il y a beaucoup d'espace pour des variations techniques importantes dans la façon de donner de la valeur au document pour son cycle de vie. C'est en raison du poids accru de la normalisation dans les facteurs de choix de logiciels que les deux dernières sections de ce rapport seront consacrées à ces questions.

CHAPITRE 4

DOMAINES DE NORMALISATION

4.1 Vue d'ensemble

La normalisation des applications documentaires, qui exigent en général la compatibilité des systèmes informatiques, peut s'appliquer à quatre domaines :

1. Les formats des documents
2. Les métadonnées associées aux documents
3. Le repérage des documents
4. L'accès aux documents

Chacun de ces domaines inclut des caractéristiques avec lesquelles les systèmes informatiques peuvent être compatibles ou non, et cela peut donner lieu à l'établissement de sous-domaines de normalisation. Avant de donner plus de détails sur chacun de ces domaines, nous introduisons certaines caractéristiques ou certains concepts qui leur sont communs.

Ces domaines et leur composition exacte ne sont pas les seuls possibles. Ils constituent un cadre de référence utile à l'analyse des normes pertinentes pour les applications documentaires.

4.2 Principes généraux

4.2.1 Formats de données et protocoles de communication

Les deux premiers domaines sont plutôt de nature statique, alors que les deux derniers, comme ils se rapportent à des processus, sont donc plutôt de nature dynamique. Sans être une règle absolue, on trouvera dans les deux premiers des *formats de données* et dans les seconds des *protocoles de communication.*

Un format de données détermine une façon d'encoder de l'information. Il est constitué, d'une part, de conventions *syntaxiques*, régissant la façon de *disposer* les données les unes par rapport aux autres en une succession d'octets et, d'autre part, de conventions *sémantiques*, régissant la façon d'*interpréter* les données disposées selon les conventions syntaxiques.

Les diverses conventions d'un format de données sont de nature *statique* parce qu'elles ne font aucune référence à une dimension temporelle dans la disposition des données. Par opposition, un protocole de communication est de nature *dynamique* parce qu'il détermine une façon de *cadencer* dans le temps les échanges « en direct » de données. Il est constitué d'une série de règles, incluant des contraintes d'ordonnancement temporel, régissant la succession des échanges de données entre deux systèmes informatiques.

Les protocoles se superposent souvent l'un à l'autre. Cette caractéristique, qui découle naturellement des façons optimales de partager les ressources matérielles et logicielles nécessaires aux communications par réseaux informatiques, est particulièrement évidente dans le modèle OSI (*Open Systems Interconnection*), qui comporte sept couches superposées de protocoles et qui constitue le modèle de référence de l'ISO pour les systèmes ouverts. Un protocole peut aussi faire référence à des formats de données en en prescrivant l'usage à certains stades de la communication. On peut dire alors que le protocole se superpose au format.

Les formats de données peuvent aussi se superposer, bien que le phénomène soit moins apparent que pour les protocoles. C'est le cas, par exemple, du format MIME (*Multimedia Internet Mail Extension*) pour les messages de courrier électronique. Ce format prescrit une façon de regrouper en un seul message électronique différentes composantes pouvant avoir chacune son propre format. Le format MIME est donc défini en fonction d'autres formats, auxquels il se superpose.

Les formats de données et les protocoles de communication peuvent être normalisés ou non. Ceux qui ne sont reconnus ou respectés que par les logiciels d'un même producteur sont qualifiés de *propriétaires* ou *privés*. Un format propriétaire peut devenir tellement populaire qu'il s'impose comme « norme *de facto* », par exemple le format Word 6.0 pour les documents textuels et le format PDF pour les documents Web). Cependant, on ne peut parler de norme véritable tant que le format ou le protocole n'est pas sanctionné comme norme par un organisme officiel, indépendant d'un producteur.

Parmi les organismes les plus actifs en normalisation dans le domaine des technologies de l'information, on retrouve l'ISO (Organisation internationale de normalisation) et le Consortium W3. Les normes émanant de ces organismes peuvent être considérées comme sérieuses et valables. Cependant, le véritable intérêt d'une norme dépend autant de son degré d'adoption dans le monde que de son statut de norme officielle reconnu par les producteurs et les utilisateurs.

4.2.2 Principe de superposition

Le principe de superposition est omniprésent dans le contexte des formats de données et des protocoles de communication. Chacun des domaines de normalisation est constitué d'une *superposition* de points de compatibilité. Il est important de réaliser que, dans toute application concrète opérationnelle, les systèmes en communication doivent être compatibles simultanément sur la *totalité* de ces points.

Les solutions adoptées pour des points de compatibilité superposés doivent bien sûr elles-mêmes se superposer, et peuvent ainsi avoir une certaine dépendance entre elles. Par exemple, si l'on désire accéder à des documents avec le protocole HTTP, il faut obligatoirement utiliser comme protocole de communication sous-jacent TCP/IP, puisque HTTP est défini en fonction de TCP/IP. Dans ce cas, on dira que ces solutions sont dépendantes. En revanche, on peut très bien utiliser SGML avec n'importe quel jeu de caractères sous-jacent ; ainsi, si l'on adopte SGML avec comme jeu de caractères l'ISO 8859-1, on dira que ces solutions sont indépendantes.

Comme les formats de données et les protocoles de communication, toute solution de compatibilité peut être normalisée ou non. Les normes qui se superposent pour offrir une solution de compatibilité à des points superposés peuvent être dépendantes ou indépendantes. Les exemples du paragraphe précédent sont justement des normes qui illustrent les deux modalités possibles.

4.2.3 Cadres et schémas

Il y a lieu de distinguer les notions de *cadre* et de *schéma*. Un cadre (en anglais, *framework*) est un modèle général, souvent abstrait, à l'intérieur duquel il est possible de définir différentes façons de structurer ou de traiter des données. Un schéma spécifie, *dans le contexte d'un cadre déterminé*, une façon spécifique de structurer ou de traiter les données.

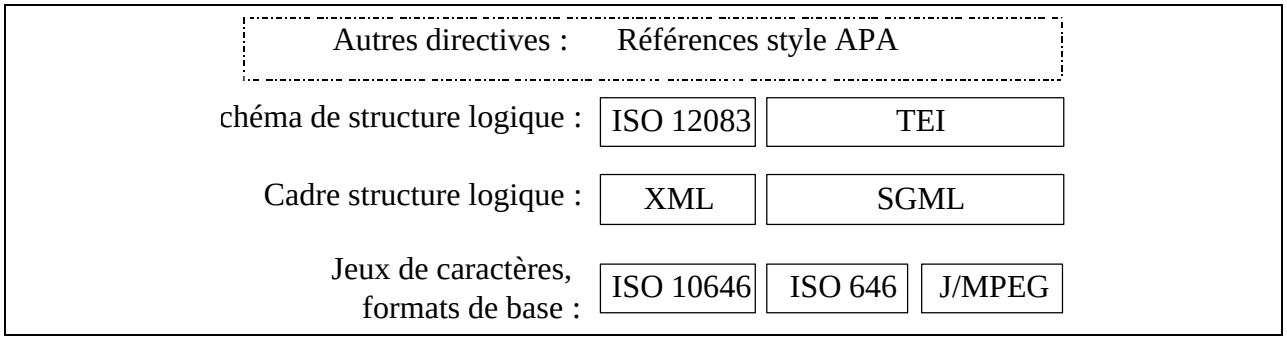
Dans le monde des bases de données, le modèle relationnel constitue un cadre, alors qu'un schéma relationnel particulier constitue effectivement un schéma. Dans le monde des algorithmes de chiffrement, un algorithme donné constitue un cadre, alors qu'une clé particulière constitue un schéma. Dans le monde des langages de programmation, un langage particulier constitue un cadre, et un programme particulier, un schéma. La superposition cadre-schéma donne souvent et naturellement lieu à une superposition de normes.

Il est possible de considérer un cadre seul, sans parler des schémas qu'il peut accueillir, mais jamais un schéma ne pourra être considéré sans le cadre qui lui donne sa signification.

4.3 Domaine des formats des documents

Le domaine des formats des documents peut être divisé en trois points de compatibilité superposés, présentés à la figure 2. Les points de compatibilité sont présentés dans la colonne de gauche ; les autres colonnes affichent certaines normes particulières correspondant à ces points.

Figure 2 - Domaine des formats des documents



La couche 1 est constituée des jeux de caractères et des formats de base dans le cas de contenus multimédias (son, image, etc.). Évidemment, les choix naturels sont les jeux de caractères normalisés et les formats normalisés.

La couche 2 est constituée du cadre de structure logique, c'est-à-dire du modèle ou métalangage utilisé pour décrire la structure logique générique des documents gérés par l'application. Les choix normalisés naturels sont les formats XML (norme proposée par le Consortium W3) et SGML (norme ISO 8879). Il est à noter que le langage XML est dépendant du jeu de caractères ISO 10646, qui intervient dans la définition même de XML.

Le schéma de structure logique de la couche 3 se rapporte à la ou aux structures logiques génériques, définies dans le contexte du cadre de structure logique choisi, auxquelles les documents gérés par l'application se conforment. Dans le contexte des formats SGML et XML, il s'agit des DTD (*Document Type Definitions*). Ici, les choix les plus appropriés ne sont pas nécessairement normalisés, dépendant de l'application spécifique considérée. Cependant, il existe plusieurs DTD normalisées par différents organismes ou groupes d'intérêt, et lorsqu'une telle DTD existe, c'est certainement l'un des choix les plus appropriés.

Une quatrième couche a été ajoutée dans un encadré pointillé pour montrer que la superposition des trois premiers points suit une progression logique qui pourrait être continuée par des directives à l'intention des rédacteurs humains quant à la forme à respecter dans les documents. La directive mentionne que les références doivent respecter le style bien connu de l'APA.

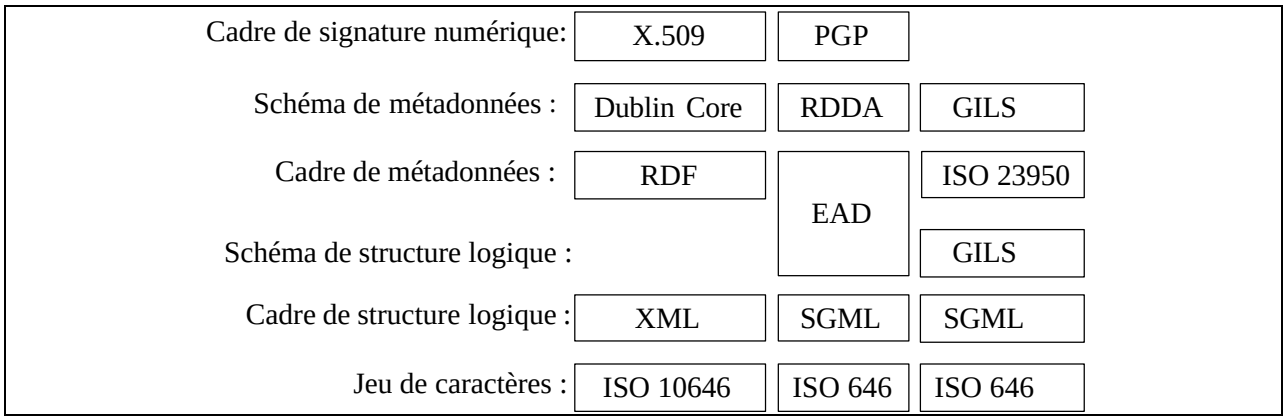
Cette présentation du domaine des formats des documents est légèrement orientée vers l'approche des documents structurés. La division en trois couches serait cependant toujours pertinente avec d'autres approches. Par exemple, dans une application gérant des documents en format Word 6.0, la couche inférieure serait constituée des polices et des formats multimédias admis dans les documents, la couche intermédiaire correspondrait au choix (imposé par l'outil) du mécanisme de feuilles de style de Word, et la troisième couche serait constituée d'une feuille de style Word particulière. (La quatrième couche à l'intention des rédacteurs humains serait encore applicable à cet exemple. Elle pourrait d'ailleurs être complétée par un protocole de rédaction plus ou moins serré destiné à augmenter la prédictibilité du contenu des documents.)

On peut développer une application gérant des documents Word 6.0 sans prescrire l'utilisation de feuilles de style, mais cela revient à laisser la couche 3 au choix des rédacteurs. On ne peut alors compter sur aucune régularité dans la structure ni dans le contenu des documents, à moins de s'appuyer sur un protocole de rédaction extrêmement serré pour les rédacteurs, diminuant ainsi la « réutilisabilité » des données contenues dans les documents.

4.4 Domaine des métadonnées

Les métadonnées associées aux documents sont d'abord et avant tout des données. Le domaine des métadonnées présenté dans la figure 3 est constitué des mêmes couches que le domaine des formats des documents. De nouvelles couches propres aux métadonnées s'ajoutent à cette figure. La couche 1, constituée des jeux de caractères, présuppose que les métadonnées sont exclusivement textuelles. Si ce n'était pas le cas, comme par exemple si l'on voulait utiliser des icônes comme métadonnées, il faudrait alors prévoir les formats de base permis, comme dans le domaine des formats des documents.

Figure 3 - Domaine des métadonnées



Les couches 2 et 3 servent aux mêmes fins que les mêmes couches du domaine des formats des documents.

Le cadre de métadonnées, dans la couche 4, définit (habituellement, dans un modèle abstrait) la structuration permise des métadonnées de même que les relations qui peuvent exister entre les métadonnées et les ressources décrites. Il peut également comporter des règles d'encodage syntaxique des métadonnées. Dans ce cas, le cadre se trouve à déterminer un format de données avec une syntaxe et, éventuellement, une certaine sémantique de base. Un cadre de métadonnées peut également être défini sans règles syntaxiques spécifiques ; il demeure alors un modèle abstrait qui doit être complété par des règles syntaxiques avant d'être implantable concrètement.

Il est à noter que lorsqu'un cadre prévoit des règles syntaxiques d'encodage des métadonnées, il peut être possible d'inclure celles-ci dans les documents mêmes, ou bien de les stocker séparément. Certains cadres, dont le format RDF, prévoient les deux possibilités.

Pour sa part, le schéma de métadonnées, dans la couche 5, consiste en un ensemble d'attributs avec une description généralement assez précise de leur sémantique et possiblement certaines règles d'écriture. Certains schémas sont définis uniquement pour un cadre de métadonnées ; d'autres sont définis indépendamment d'un cadre particulier et peuvent donc être implantés dans différents cadres.

Le format RDF (*Resource Description Framework*), élaboré par le Consortium W3, est présentement le point focal d'un ensemble d'initiatives dans le domaine des métadonnées, qui voient là l'occasion de s'aligner sur un cadre normalisé quasi-universel. Nous en reparlerons plus en détail au point 5.3. La norme XML est présentée dans le format RDF comme un substrat privilégié pour l'expression syntaxique des métadonnées. L'axe XML—RDF semble donc particulièrement fort présentement pour les documents **et** leurs métadonnées.

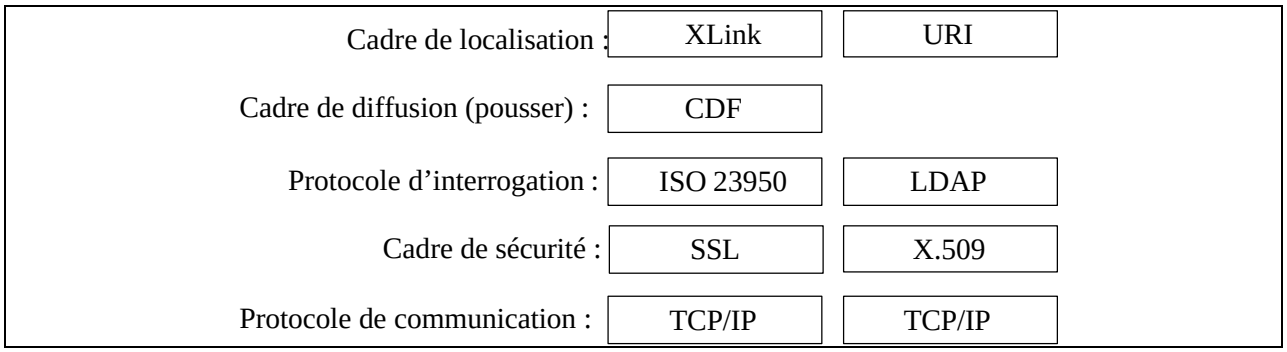
Dans la couche 6, nous avons inclus la signature numérique (ou, éventuellement, *les* signatures numériques) comme métadonnées. Pour être de quelque utilité ou valeur, elle doit être réalisée

dans un cadre prévu dans les spécifications de l'application. La norme X.509, développée par l'ISO et l'UIT, est un exemple d'un tel cadre. PGP (*Pretty Good Privacy*) est un autre cadre de signature numérique assez répandu. Notons qu'un cadre de signature numérique s'appuie en général, implicitement ou explicitement, sur une infrastructure d'attribution et de distribution de clés publiques.

4.5 Domaine du repérage des documents

Le domaine du repérage des documents, présenté dans la figure 4, est constitué de protocoles et de formats.

Figure 4 - Domaine du repérage des documents



Le cadre de sécurité, dans la couche 2, peut servir à assurer la confidentialité et l'authenticité des documents, de même que le cadre de diffusion, qui est un mécanisme du « pousser » (*push*), dans la couche 4. L'exemple indiqué est la norme CDF (*Channel Definition Format*), un format de description de canaux de pousser basé sur le format RDF (et, donc, sur XML) et proposé par Microsoft. Le format CDF est d'ores et déjà reconnu par Internet Explorer 4.

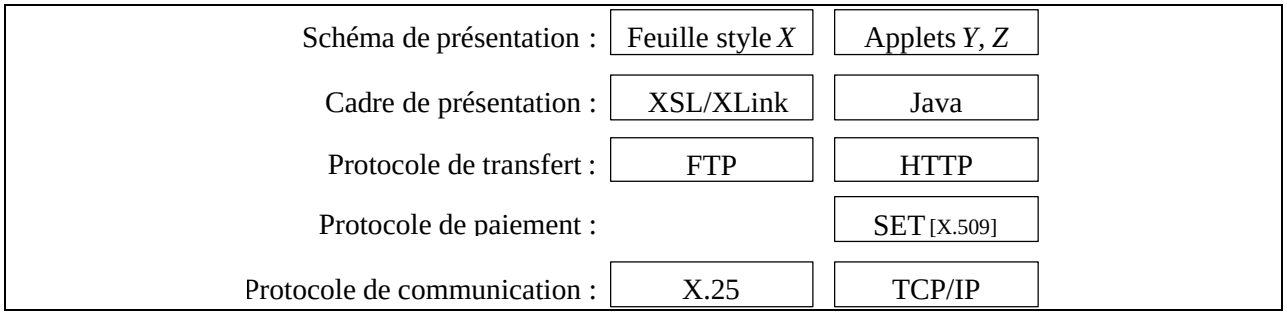
Le protocole d'interrogation, dans la couche 3, inclut un langage d'interrogation permettant la formulation de requêtes de recherche pouvant exploiter, entre autres, les métadonnées. La norme ISO 23950 (le pendant international de la norme américaine Z39.50) comporte non seulement un protocole d'interrogation, mais également un cadre de métadonnées. L'autre exemple, la norme LDAP, est un protocole d'interrogation de répertoires.

Ce domaine inclut également, dans la couche 5, un cadre de localisation, c'est-à-dire un formalisme permettant de pointer les documents repérés. La norme XLink propose, dans la lignée de XML, une façon normalisée de représenter des liens hypertextuels plus riches que ceux permis par HTML. XLink n'est actuellement qu'un projet de norme du Consortium W3. L'autre exemple, le formalisme des URI (*Uniform Resource Identifier*), est en cours d'élaboration par le W3C ; ce concept chapeaute la syntaxe URL (*Uniform Resource Locator*) et URN (*Uniform Resource Name*). Actuellement, un des seuls cadres de localisation un tant soit peu normalisés et réellement implantés est le formalisme des URL, sur lequel sont basés les liens hypertextuels de HTML.

4.6 Domaine de l'accès aux documents

L'accès aux documents dépend de plusieurs points de compatibilité, présentés à la figure 5.

Figure 5 - Domaine de l'accès aux documents



Le protocole de paiement, dans la couche 2, a été prévu parce qu'on recourra de plus en plus fréquemment au paiement à la pièce pour la récupération de documents d'archives, d'autant plus que les micro-paiements (paiements de très petits montants) devraient être possibles quand le commerce électronique sera chose courante. La norme SET (*Secure Electronic Transaction*) est un protocole entériné par plusieurs intervenants dans le domaine du commerce électronique, dont Visa et MasterCard. La norme SET peut s'appuyer sur la norme X.509 pour les aspects de sécurité.

Le cadre et le schéma de présentation dans les couches 4 et 5 représentent l'environnement informatique requis pour que la restitution des documents s'effectue correctement. Dans le cas de documents traditionnels (mais tout de même électroniques), un simple mécanisme de feuilles de style peut suffire à gouverner complètement la restitution. Dans le cas de documents dynamiques, par exemple comportant des liens hypertextuels internes ou externes, ou devant réagir à certaines actions du lecteur (les documents « interactifs »), un environnement plus sophistiqué peut être nécessaire. Les deux exemples incluent un mécanisme de feuilles de style (XSL), un langage de liens (XLink) et un environnement complètement programmable, soit Java, qui inclut la notion d'» applet » et le langage de scriptage JavaScript, dont il existe une version normalisée par l'ECMA, l'ECMAScript (norme ECMA-262).

La norme XSL (*eXtensible Styling Language*) fait partie de l'axe XML—XLink et consiste en un langage de feuilles de styles qui généralise les CSS (*Cascading Style Sheets*) et, incidemment, permet le recours à l'ECMAScript pour la spécification de traitements particulièrement complexes. La norme XSL n'en est cependant qu'au stage de la proposition au sein du W3C, bien que certains producteurs (dont Microsoft et Arbortext) aient déjà développé des prototypes de processeurs XSL.

CHAPITRE 5

NORMES DE STRUCTURES LOGIQUES ET DE MÉTADONNÉES

5.1 Importance prépondérante des données stockées

Parmi toutes les solutions de compatibilité que doivent comporter les applications documentaires, certaines sont d'importance plus cruciale que d'autres. En effet, certaines ont une incidence sur les données stockées elles-mêmes, et donc influent sur leur accessibilité à long terme.

À titre d'exemple, mettons en parallèle le schéma de métadonnées adopté par une application et une feuille de style servant à présenter ces métadonnées sous forme de fiches à l'écran. Le premier influence la nature des métadonnées stockées par le système. Tout changement dans ce schéma influencera la validité des métadonnées déjà stockées, et donc leur « utilisabilité » pour le repérage futur. À l'opposé, une modification apportée à la feuille de style de la présentation des métadonnées n'invalide en rien les métadonnées déjà stockées et ne modifie aucunement leur « utilisabilité ».

Bien que le choix d'un schéma de métadonnées et celui d'une feuille de style pour les présenter influencent tous deux « l'utilisabilité » immédiate d'une application documentaire, le schéma a une incidence beaucoup plus fondamentale sur l'accessibilité à long terme des documents que la feuille de style.

Les points de compatibilité qui ont une incidence sur les données stockées sont ceux des domaines des formats des documents et des métadonnées (le cadre de présentation, faisant partie du domaine de l'accès aux documents, est aussi important, mais seulement pour les documents dynamiques). Il existe deux normes actuellement en plein essor et qui, à elles seules, couvrent une gamme considérable de points de compatibilité des deux domaines en question. Il s'agit des formats XML et RDF.

5.2 Un cadre normalisé de structure logique : la norme XML

XML signifie *eXtensible Markup Language*. Il s'agit d'un format de documents structurés, en filiation avec la norme SGML (*Standard Generalized Markup Language* ; ISO 8879). La norme XML est une recommandation du W3C (le plus haut niveau de normalisation possible au sein du W3C), portant le code REC-xml-19980210.

5.2.1 Balisage

Le format XML (comme SGML) est basé sur le concept de *balisage*. Une balise est une courte chaîne de caractères qui indique le début ou la fin d'un segment de document. Les règles syntaxiques du langage assurent que les balises ne peuvent être confondues avec le texte proprement dit du document. En XML, il existe trois types de balises : les balises de début, les balises de fin et les balises de début vides. Toutes les balises commencent par le caractère « < » et se terminent par « > ».

Une balise de début comporte obligatoirement, immédiatement après le « < », un *identificateur générique*, soit un nom destiné à identifier le type d'information qui se trouve dans le segment qui commence. Une balise de début peut aussi comporter une ou plusieurs spécifications d'*attributs*. Les attributs qualifient le type d'information correspondant à l'identificateur générique, ou fournissent autrement de l'information additionnelle sur le segment qui commence. Voici des exemples de balises de début : « `<titre>` », « `<adresse type="courriel">` », « `<nom genre="f" lang="fr">` ».

Une balise de fin comporte, immédiatement après le « < », le caractère « / », suivi d'un identificateur générique, et rien d'autre. Exemples de balise de fin : « `</titre>` », « `</adresse>` », « `</nom>` ». Une balise de fin indique la fin d'un segment et doit comporter le même identificateur générique que la balise de début qui en indique le début. Le segment de document délimité par une balise de début et la balise de fin correspondante s'appelle *élément*. Des éléments peuvent être imbriqués l'un dans l'autre, mais ne peuvent se chevaucher :

valide : `<A> <B> </B> </A>`
invalide : `<A> <B> </A> </B>`

Une balise de début vide a la même forme qu'une balise de début, mais contient le caractère « / » immédiatement avant le « > » final. Elle sert à représenter un élément dont le contenu est vide ; en fait, on peut la considérer comme une abréviation d'une balise de début suivie immédiatement d'une balise de fin correspondante. Par exemple :

```
        <BR type="big"/>
équivaut à <BR type="big"></BR>
```

Un élément vide sert habituellement à marquer un endroit dans un document, et non à délimiter un segment.

5.2.2 Métaformat

Comme SGML, XML est en fait un « métaformat », c'est-à-dire un langage qui permet de définir différents formats, adaptés à différents types de documents. On définit un format particulier en rédigeant une DTD (*Document Type Definition*), qui détermine quelles balises peuvent être utilisées, ainsi que les règles d'utilisation de ces balises (l'emplacement des différentes balises les unes par rapport aux autres dans le document).

5.2.3 Vocation de la norme XML

Étant un métaformat de documents, le langage XML est en fait un cadre de structure logique. Pour les applications documentaires, il peut donc faire partie des solutions de compatibilité dans le domaine des formats des documents et dans celui des métadonnées. Il sert également de substrat syntaxique au format CDF (*Channel Definition Format*), un cadre de diffusion utilisé pour le repérage des documents.

La norme XML est d'une complexité formelle moindre que le format SGML, ce qui en fait un format plus adapté au Web. L'objectif de la conception de XML était clairement le remplacement de la norme HTML comme langage universel du Web, cette dernière étant sémantiquement trop pauvre, comme le démontre le besoin constant de plusieurs producteurs de lui adjoindre des extensions propriétaires. La norme XML étant intrinsèquement extensible (comme son nom l'indique), les producteurs n'ont plus la crainte d'être limités dans les fonctionnalités de leurs documents par le langage de description utilisé.

En dépit de la plus grande simplicité de XML, plusieurs experts croient qu'il n'y a rien de ce que l'on peut faire avec SGML, que l'on ne pourrait faire aussi avec XML.

5.2.4 Documents valides et bien-formés

Un document XML *valide* est un document conforme à une DTD particulière. Un document valide comporte nécessairement un préambule qui identifie la DTD à laquelle se conforme le document. Les applications de traitement du document peuvent donc récupérer la DTD et l'utiliser pour la validation ou les autres opérations à effectuer sur le document.

Certaines des simplifications apportées à la norme XML par rapport au format SGML rendent possible la définition de la notion de documents « bien-formés » (*well-formed*) qui ne sont pas nécessairement valides. Un document bien-formé est un document XML dont le balisage est correct, mais qui ne respecte pas nécessairement les règles spécifiques d'une DTD particulière. Dans un document bien-formé non valide, les identificateurs génériques peuvent être quelconques ; de plus, n'importe quel élément peut survenir n'importe où (pour autant, bien sûr, que les balises sont bien appariées et qu'il n'y a pas de chevauchement d'éléments).

Bien qu'*a priori* il puisse sembler que les documents bien-formés non valides soient trop peu prévisibles pour être d'une quelconque utilité, ils s'avèrent adéquats pour bien des applications, en particulier pour des applications Web. Ainsi, cette simple possibilité d'accommoder les documents non valides ouvre à la norme XML plusieurs domaines d'application hors de la portée du format SGML.

5.2.5 Utilisation de la norme XML

Un nombre impressionnant de producteurs de logiciels (dont tous les producteurs actuels de logiciels SGML) ont déjà annoncé leur soutien à la norme XML. Certains produits le supportant ont déjà atteint le marché et sont dès maintenant disponibles commercialement. Microsoft, en particulier, appuie avec force XML, ayant entre autres déjà annoncé (mais non officiellement)

que la prochaine version d'Office supporterait XML. Internet Explorer 4.0 inclut déjà deux analyseurs syntaxiques XML, en particulier pour le support du format CDF (*Channel Definition Format*), qui est reconnu par le navigateur.

Les applications de la norme XML sont d'ores et déjà légion ; plusieurs des « grandes » DTD SGML sont en cours de réécriture en XML (ce processus est souvent très simple, étant donné la grande similarité entre les caractéristiques de SGML couramment utilisées et XML). Mentionnons, entre autres, la DTD ISO 120 83 (pour les périodiques électroniques et les articles de périodiques), la DTD TEI (*Text Encoding Initiative*), utilisée pour l'encodage de textes littéraires classiques, et la DTD EAD (*Encoded Archival Description*), utilisée en archivistique.

Par ailleurs, une foule de nouvelles applications (avec DTD correspondantes) ont été développées directement en XML. Mentionnons le *Chemical Markup Language* (CML), le MathML (langage de balisage pour les formules mathématiques), etc.

5.2.6 Document XML : un exemple

Voici un exemple de document XML (figure 6). Il s'agit d'un document bien-formé, mais non valide, puisqu'il ne comporte aucune référence à une DTD spécifique.

Figure 6 - Exemple de document XML

```
<?xml version="1.0" encoding="utf-8"?>
<mémo>
<auteur> Julia Royer </auteur>
<destinataires>
   <nom> Jean Picard </nom>
   <nom> Émilie Dugré </nom>
</destinataires>
<sujet> Invitation </sujet>
<corps>
   <par> Veuillez noter que ... le 27
         septembre 1999. </par>
   <par> SVP, avisez-moi ... dans les plus
         brefs délais. </par>
</corps>
</mémo>
```

On remarque que les balises traduisent la structure logique naturelle du document.

5.3 Un cadre normalisé de métadonnées : RDF

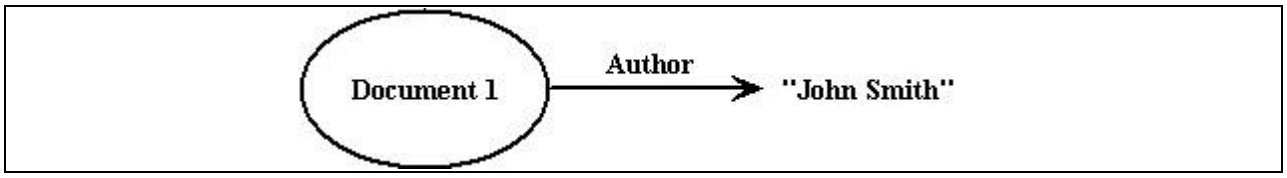
RDF signifie *Resource Description Framework.* C'est une ébauche de norme au sein du W3C (l'ébauche WD-rdf-syntax-19980216). Le format RDF est un cadre de métadonnées ; il propose donc un modèle abstrait permettant d'attribuer certaines propriétés à certaines ressources (principalement, mais non exclusivement, Web). Il propose également une syntaxe permettant de représenter les objets de ce modèle abstrait sous une forme échangeable entre systèmes informatiques, spécialement sur le Web. Cette syntaxe est basée sur la syntaxe XML. Comme nous l'avons mentionné précédemment, le format RDF constitue actuellement l'axe dans lequel plusieurs initiatives de métadonnées s'orientent, dont PICS et le Dublin Core.

Comme tout les cadres très généraux, la norme RDF possède un modèle de base extrêmement simple. Le modèle de RDF est basé sur la notion mathématique de *graphe*. Un graphe est un ensemble de *nœuds*, que relient un certain nombre d'*arcs*, c'est-à-dire des flèches qui partent d'un nœud et aboutissent à un autre. Un arc peut également aboutir à un objet qui n'est pas un nœud, mais plutôt ce qui est appelé une *chaîne RDF*, soit, en l'occurrence, une chaîne de caractères du jeu ISO 10646. Tout arc possède une *étiquette*, qui indique un *type de propriété*.

Intuitivement, les nœuds du modèle de base sont les ressources d'information auxquelles on veut attribuer des propriétés ; à la limite, ce pourrait être le Web au complet, mais ce peut être également des ensembles plus restreints, selon l'application. Les étiquettes des arcs représentent les types de propriété et les points d'arrivée des arcs sont les valeurs des propriétés. Ce qui est appelé « propriété » dans le contexte de la norme RDF correspond à ce que nous appelons « attribut » ailleurs dans ce document.

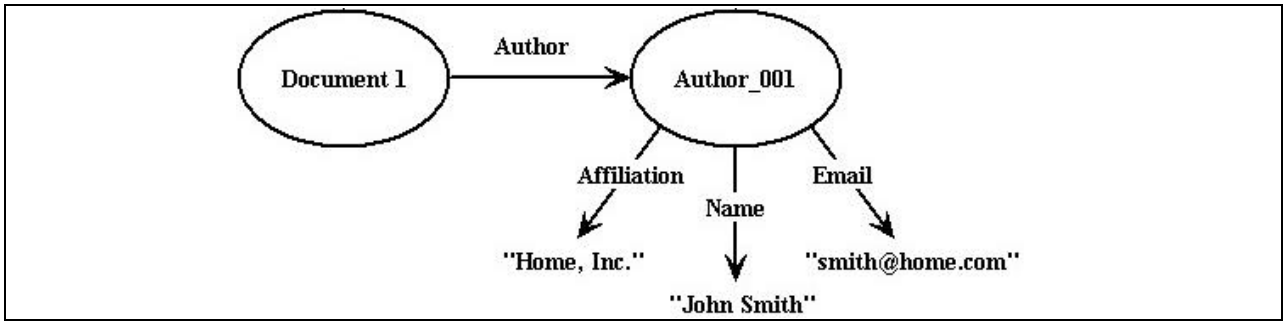
Pour attribuer une propriété à une ressource, on ajoute au graphe un arc, dont l'étiquette correspond au type de la propriété que l'on désire attribuer, dont le point de départ est la ressource à laquelle on désire attribuer la propriété, et dont le point d'arrivée est la valeur que l'on veut donner à la propriété. La figure 7 montre une propriété énonçant que John Smith est l'auteur de la ressource Document 1.

Figure 7 – Spécification d'une propriété en RDF (tiré de Miller, 1998)



Dans cet exemple, la valeur de la propriété est atomique (la chaîne "John Smith") ; il est cependant possible que la valeur de la propriété soit structurée. Pour ce faire, il suffit d'utiliser comme point d'arrivée de l'arc une autre ressource (*i.e.*, un autre nœud du graphe), laquelle possédera l'ensemble des propriétés désirées. La figure 8 illustre cette possibilité.

Figure 8 - Propriété avec valeur complexe (tiré de Miller, 1998)



La norme RDF prévoit également l'attribution de propriétés à des propriétés, de même que la possibilité de signer numériquement une description, de sorte que l'on puisse être certain de son intégrité et de l'authenticité de sa provenance. L'attribution d'une propriété à une propriété permettrait d'exploiter la provenance d'une métadonnée comme critère de recherche d'une ressource ; par exemple, on pourrait demander le repérage de toutes les ressources attribuées à l'auteur Michel Tremblay, mais seulement si cette attribution a été faite par la Bibliothèque nationale du Québec ou la Bibliothèque nationale du Canada.

Une autre initiative du Consortium W3 reliée à RDF est celle des *RDF Schemas*. Cette ébauche de norme (l'ébauche WD-rdf-schema-19980409) prévoit une façon normalisée de définir, dans le modèle RDF lui-même, des schémas spécifiques de métadonnées, incluant la spécification des valeurs permises pour certains attributs.

Bien que le format RDF semble voué à un avenir brillant, il n'existe encore aucune implantation fonctionnelle fondée sur ce formalisme. Son incidence sur la conception d'applications documentaires est encore marginale. Cependant, étant donné l'aval massif accordé à RDF par plusieurs intervenants dans le monde des métadonnées, il est de rigueur de surveiller attentivement son développement. À notre avis, il ne serait pas étonnant d'assister à l'arrivée de produits supportant le format RDF d'ici au plus un an ou deux.

CONCLUSION

Le cycle de vie des documents fait intervenir plusieurs fonctions généralement réparties entre plusieurs logiciels distincts. La façon d'harmoniser ces logiciels est de s'éloigner au maximum des formats propriétaires et de miser sur les normes ouvertes de l'Internet pour rechercher l'interfonctionnalité des logiciels. Nous avons vu qu'il ne saurait y avoir de solution dans un logiciel unique, du moins à court terme, mais que chaque organisation doit interpréter ses besoins d'affaires et son contexte technique pour décider d'une stratégie à l'égard des applications visées et des choix d'outils adéquats. Cet éclairage peut sembler indirect, mais l'importance accordée aux normes, en particulier XML et RDF, est nécessaire pour porter un regard rafraîchi sur le marché des logiciels.

ANNEXE 1

EXIGENCES FONCTIONNELLES DÉTAILLÉES POUR LE CHOIX D'UN PROGICIEL

Veuillez préciser si oui ou non votre progiciel répond à chaque exigence exprimée dans les pages suivantes. Si vous répondez « non », vous devez nous indiquer quelle(s) solution(s) vous proposez pour y arriver, par exemple par un développement particulier du progiciel, par un lien avec un autre progiciel (mentionner lequel), etc. Si vous répondez « oui », vous pouvez également ajouter un court commentaire.

GESTION DES DOCUMENTS

Entrepôt de documents

Le progiciel comporte un entrepôt de documents où l'on peut stocker et gérer les documents ayant ou non une source électronique et tout autre objet prévu par le progiciel d'une même façon.

L'usager peut classer les objets dans l'entrepôt en utilisant un progiciel de dossiers hiérarchiques avec un nombre illimité d'objets et de niveaux.

Il est possible de créer des références vers les objets (« alias », « raccourcis », « liens ») et l'intégrité de ces références est constamment garantie par le progiciel.

L'entrepôt de documents doit pouvoir gérer des documents composés de plusieurs parties.

Le progiciel est doté de fonctions d'assemblage de documents complexes, par exemple pour permettre la consultation ou l'impression en un seul bloc d'un document comprenant plusieurs parties.

L'usager peut conserver plusieurs représentations (formats) d'un même objet tout en traitant ces différentes représentations comme un seul objet logique.

L'entrepôt de documents doit pouvoir gérer tout type de fichier ou de document.

L'entrepôt de documents doit permettre de créer des liens entre différents objets, par exemple d'un document à un autre.

L'entrepôt de documents peut contenir des références vers des objets extérieurs au progiciel, par exemple des documents sur l'intranet ou l'Internet.

On peut facilement référencer les objets de l'entrepôt à partir d'autres applications ou d'autres sites de l'intranet ou de l'Internet.

Un usager utilisant les applications bureautiques normalisées à Hydro-Québec peut stocker et consulter directement, sans quitter son application, les objets de l'entrepôt de documents.

Le progiciel permet de gérer différents statuts d'archivage pour les documents qui incluent des délais ouverts, fermés et conditionnels. Parmi ces statuts, on doit retrouver :

- les documents « actifs », avec les métadonnées et les documents qui sont disponibles en ligne ;

- les documents « inactifs », avec les métadonnées disponibles en ligne, et les documents pouvant être sur des supports tels que des bandes ou des disques optiques ;

- les documents « archivés », dont les métadonnées et les documents sont stockés sur des supports externes et hors ligne.

Le progiciel permet de documenter l'utilisation des statuts d'archivages relativement à des regroupements d'objets.

Métadonnées

Les objets stockés dans l'entrepôt peuvent être décrits à l'aide de champs que l'on peut déterminer nous-mêmes.

Un ensemble de documents peut partager un même « profil » de métadonnées, c'est-à-dire les mêmes champs. Il n'y a pas de limite quand au nombre de profils que l'on peut utiliser.

Il n'y a pas de limite quant au nombre de champs que l'on peut attribuer à un objet.

Les champs de description peuvent être de différents types, notamment texte, long texte, date, numérique.

Certains champs doivent être saisis automatiquement par le progiciel, tels que les dates (création, modification), les numéros de version, les propriétaires, etc.

Il est possible de définir des fonctions de validation pour les valeurs de certains champs, par exemple des masques de saisie, des intervalles de dates ou numériques, validation en fonction de la valeur d'autres champs.

Il est possible de définir un champ ayant une taille illimitée.

Certains champs peuvent être définis comme étant à saisie obligatoire.

Certains champs ne peuvent être modifiés par les usagers, tels que des dates, des codes d'usager, etc.

Les champs peuvent comprendre plusieurs valeurs (« occurrences multiples »).

Les champs peuvent inclure une valeur par défaut afin de faciliter la saisie par les usagers.

Les valeurs par défaut des champs peuvent être déterminées par l'usager.

Les valeurs de certains champs peuvent être saisies à partir d'une liste de valeurs déterminées par les responsables du progiciel.

Les valeurs de certains champs peuvent être liées à d'autres sources de données (tables externes, par exemple).

Les valeurs de certains champs peuvent être saisies automatiquement en fonction de règles de numérotation séquentielle (caractères, numérique, mixte).

Le progiciel comporte des éléments d'interface pour les champs (listes déroulantes, zones partiellement remplies, etc.) qui permettent une saisie efficace de l'information, et ce en fonction des caractéristiques du champ.

Le progiciel est doté d'un dictionnaire qui corrige automatiquement les mots saisis par les usagers dans les champs (dictionnaires des langues française et anglaise, au moins).

La spécification des champs est suffisamment souple pour que les objets puissent être décrits en suivant les Règles de catalogage anglo-américaines (RCAA) et les Règles de description des documents d'archives (RDDA).

Le contenu de tous les champs de description de tous les objets peut être cherché à l'aide de l'outil de recherche.

Il existe des fonctions permettant de modifier en lot la valeur d'un champ d'un grand nombre d'objets, par exemple tous les objets d'un dossier.

Pour les messages de courrier électronique, le progiciel peut capter automatiquement les informations relatives à la transmission (expéditeur, récepteur, sujet, dates, etc.) et les insérer dans des champs de métadonnées.

Pour les documents de bureautique (progiciels bureautiques normalisés Hydro-Québec et documents formatés selon la norme XML), le progiciel peut capter automatiquement les informations déjà saisies dans les propriétés du document (titre, sujet, mots clés, auteur, résumé, dates, catégorie, etc.).

Un objet peut hériter automatiquement des métadonnées associées à l'objet le contenant (*e.g.* un dossier), et inversement les métadonnées associées à un contenant peuvent s'appliquer à tous les objets qu'il contient.

Le progiciel permet d'intégrer un thésaurus, à la fois pour aider les usagers à saisir des métadonnées, pour valider ces métadonnées, ainsi que comme aide à la recherche.

Le progiciel est capable de conserver un historique des différents propriétaires (*e.g.* une unité administrative) d'un document.

Consultation des documents

Les usagers peuvent consulter les documents dans leur format natif, c'est-à-dire tels qu'ils ont été entrés dans le progiciel.

Le progiciel peut convertir en HTML les principaux formats de documents utilisés pour les outils bureautiques normalisés à Hydro-Québec.

MANIPULATION DES DOCUMENTS

Le progiciel comporte des fonctions permettant de synchroniser le travail de plusieurs personnes sur un même document : « check-out », « check-in », réservation, etc.

L'usager peut s'ajouter à une liste de réservation de tout objet géré par le progiciel ainsi que de ses différentes versions.

Les usagers peuvent savoir qui a réservé ou sorti un document et à quel moment.

Le progiciel permet la gestion de la localisation, de l'inventaire et de l'entreposage des objets matériels.

Le progiciel permet de conserver plusieurs versions d'un même document.

Les différentes versions d'un même document peuvent être décrites différemment.

On peut limiter le nombre de versions à conserver pour un même document.

On peut identifier des versions d'un document qui sont à conserver et qui ne seront pas supprimées même si le nombre limite de versions permises est dépassé.

TRAVAIL EN COLLABORATION

Le progiciel est doté d'outils pour favoriser le travail en collaboration dans un groupe d'individus.

Les groupes de travail peuvent partager un espace pour stocker et gérer les documents relatifs à leurs activités.

Le progiciel permet aux individus travaillant en collaboration de tenir des discussions « électroniques ».

Le progiciel permet d'assigner des tâches aux individus membres d'un groupe de travail et comporte des outils de gestion de ces tâches.

Le progiciel permet à un groupe de travail de partager un agenda commun.

Le progiciel offre des fonctions de réservation de locaux et de matériel mobile.

CIRCUIT DE PRODUCTION (*workflow*)

Le progiciel prévoit un outil simple et graphique qui permet de modéliser et d'automatiser des circuits de production (*workflow*) complexes.

Les spécifications du circuit de production peuvent contenir des routes en série, en parallèle, ainsi que des retours en arrière.

Les spécifications du circuit de production peuvent inclure des champs descriptifs qui seront saisis pendant l'exécution et qui peuvent servir à prendre des décisions pour le déroulement du circuit de production.

Les spécifications du circuit de production permettent à l'usager de modifier, s'il en a les droits, la route suivie par le circuit de production, dans le cas d'un changement *ad hoc* du processus de travail.

On peut constituer en modules les spécifications du circuit de production, dans le but de créer des parties indépendantes qui peuvent être intégrées dans un tout et réutilisées dans plusieurs spécifications.

Une spécification du circuit de production peut être exécutée aussi souvent que désirée.

Les spécifications du circuit de production doivent être traitées comme les autres objets du progiciel : on peut les décrire, gérer les versions, les réserver, etc.

Les droits d'accès au circuit de production permettent à l'usager de déterminer qui peut modifier la spécification, qui peut l'exécuter, qui peut en suivre le déroulement et qui peut en disposer à la fin de l'exécution.

On peut attacher à l'exécution d'un circuit de production des objets provenant de l'entrepôt de documents, et les métadonnées associées à ces objets peuvent être modifiées automatiquement en fonction des actions entreprises pendant l'exécution du circuit de production.

À la fin de l'exécution d'un circuit de production, on peut stocker toute l'information (métadonnées, documents, commentaires) qui lui est attachée et la consulter ultérieurement.

Les résultats de l'exécution d'un circuit de production contiennent de l'information précise sur la date et l'heure où les différentes actions ont été effectuées.

Le gestionnaire de l'exécution d'un circuit de production peut être averti lorsque des retards surviennent, ou encore lorsque des circonstances particulières altèrent le déroulement du processus de travail.

RECHERCHE

Un outil de recherche permet de faire des recherches dans l'ensemble des objets gérés par le progiciel.

Les objets retournés par l'outil de recherche sont ceux qui satisfont aux critères de recherche et dont l'usager a (au moins) le droit de lecture.

L'outil de recherche permet également de faire des recherches à l'extérieur du progiciel, soit sur le réseau local, l'Internet, ou dans d'autres bases de données, ou de collaborer avec des outils de recherche déjà en place.

Pour les objets gérés par le progiciel, l'outil permet de chercher dans les valeurs de tous les champs de description.

L'outil de recherche permet de chercher dans le contenu des documents (texte intégral).

L'outil permet de chercher dans les métadonnées d'un document dont on ne possède pas la version électronique, ou encore dans celles d'un document qui est stocké sur un support d'archivage (hors ligne).

L'outil de recherche supporte les principaux opérateurs booléens (ET, OU, SAUF) et de proximité (PRÈS, AVANT, APRÈS).

L'outil de recherche supporte les principaux opérateurs mathématiques (<, >, <=, >=, =, <>) sur des champs de type numérique ou date.

L'outil de recherche supporte la troncature explicite à droite, à gauche ou au centre des mots.

L'usager peut combiner à l'aide des opérateurs booléens (ET, OU, SAUF) différents critères de recherche qui portent sur différents mots ou phrases ou différents champs.

L'outil de recherche et d'indexation utilise un anti-dictionnaire pour éliminer les mots courants dans les recherches par mots clés.

Le progiciel propose un anti-dictionnaire pour la langue française, et il est possible de le modifier.

Le progiciel propose la correction automatique des mots mal écrits dans les requêtes de recherche à partir d'un dictionnaire.

Les dictionnaires offerts pour les corrections incluent au moins les langues française et anglaise.

Les usagers ont accès à des aides à la recherche pour certains champs, tels des thésaurus, des index, des tables de validation, etc.

L'usager peut conserver une requête de recherche et l'exécuter autant de fois qu'il le désire par la suite.

L'usager peut sauvegarder des résultats d'une recherche et y revenir ultérieurement.

Le progiciel peut déclencher automatiquement, à des moments prédéfinis, des requêtes de recherche.

Les résultats d'une recherche peuvent être classés de différentes façons, y compris par ordre de pertinence probable et par ordre alphabétique ou numérique sur tous les champs de métadonnées.

L'usager peut configurer la présentation des résultats d'une recherche, y compris le nombre de résultats à retourner de même que la clé de tri.

La présentation des résultats d'une recherche transmet d'abord des informations générales sur les résultats (nombre de résultats, requêtes effectuées, etc.) et une liste contenant le sommaire des objets trouvés (titre, date, résumé, etc.).

Le progiciel peut mettre en évidence les mots clés trouvés à la suite d'une recherche, autant dans les métadonnées que dans les documents eux-mêmes.

COMMUNICATIONS AVEC LES USAGERS

Le progiciel comporte un mécanisme par lequel l'usager peut être averti par courrier électronique lorsque certains événements se produisent.

L'usager peut être averti lorsque des nouveaux documents sont ajoutés dans l'entrepôt ou une dans certaine partie de l'entrepôt.

L'usager peut être averti lorsque des documents sont modifiés dans l'entrepôt, par exemple lorsqu'un nouvelle version est ajoutée ou lorsqu'on modifie les métadonnées.

L'usager peut être averti lorsqu'une tâche lui est assignée, que ce soit dans le cadre de l'exécution d'un circuit de production ou dans le cadre du travail en collaboration.

L'usager peut être averti lorsqu'il est intégré à un groupe de travail.

TRANSACTIONS

Le progiciel permet de gérer différents types de transaction impliquant tout objet : transactions ponctuelles (achat, vente), location ou prêt, transactions à fréquence déterminée ou non (abonnements), etc.

Le progiciel permet le contrôle de la réception et de l'expédition des objets matériels faisant partie des transactions.

Le progiciel permet de gérer la facturation reliée aux différentes transactions.

Le progiciel permet de gérer une banque de clients et de fournisseurs avec qui on peut effectuer des transactions.

Le progiciel permet de faire le suivi du budget alloué aux différents types de transactions.

ANNEXE 2

PRÉSENTATION DE QUELQUES NORMES RELATIVES À L'INGÉNIERIE DOCUMENTAIRE

(par ordre alphabétique de leurs acronymes)

* **CDF** (*Channel Definition Format*) : proposition de Microsoft au W3C, CDF est un langage basé sur XML et qui sert à programmer des canaux pour la diffusion sélective de l'information.

* **CML** (*Chemical Markup Language*) : implanté en SGML puis en XML, il s'agit d'un ensemble de conventions propres à l'industrie chimique.

* **CSS** ou **CSS2** (*Cascading Style Sheet*) : langage de style pour la présentation des documents diffusés en HTML, et potentiellement en XML (voir aussi XSL)

* **DCD** (*Document Content Description*) : langage de schémas basé sur la syntaxe XML pour manipuler plus facilement les DTD elles-mêmes en tant qu'objets formels et permettant de les combiner dans un style orienté objet de la même façon que les formes architecturales de *HyTime*. Les types de données (*data types*) définis dans la proposition *XML-data* y sont repris, de même qu'un vocabulaire de base est fourni pour la norme RDF.

* **DOM** (*Document Object Model*) : le modèle objet du document définit une API orientée objet pour les opérations de lecture et d'écriture par les applications sur les pages Web ou les documents XML. Un noyau en a été défini par le W3C à l'automne 1997. Les pages dynamiques en HTML ont particulièrement besoin du DOM pour permettre aux scripts la manipulation du contenu, en réconciliant les DOM respectifs de Netscape et de Microsoft. Les DOM offrent aussi un moyen pour séparer ce qui relève de la gestion du document et ce qui relève de l'édition ou présentation. En DOM, le document est une arborescence de noeuds dont il est la racine ; le premier niveau de noeuds peut comprendre l'en-tête, les métadonnées, les commentaires ; les attributs des éléments et le texte se situent aux niveaux de noeuds suivants selon les décompositions en éléments. Le modèle peut être bâti avec le langage de définition d'interface de CORBA, ainsi qu'en Java. Un noeud spécial d'un objet DOM est DocumentContext, qui contient les métadonnées sur le document.

* **DSig Label** (*Digital Signature Label*) : les étiquettes signées numériquement sont l'objet d'une norme proposée par le W3C afin de permettre des énoncés garantis à propos d'un document, ou des « manifestes » signés contenant divers énoncés à propos de plusieurs objets rassemblés. Ces étiquettes sont associées à des étiquettes PICS, mais vraisemblablement elles deviendront des énoncés RDF. Par exemple, une fois qu'un programme, sa documentation et des exemples ont été créés, l'utilisateur crée un manifeste qui pointe sur ces trois ressources avec l'énoncé « J'ai créé ceci » pour chacun des éléments Le créateur ajoute une étiquette pointant sur le manifeste avec l'énoncé « applet à sécurité

garantie »). Il signe ensuite cette étiquette et insère le "bloc de signature" résultant dans l'étiquette du manifeste. L'intervention d'une tierce partie dans la certification des énoncés peut créer l'étiquette « applet à sécurité garantie », la signer et la distribuer.

* **ICE** (*Information Content and Exchange*) : application XML conçue pour faciliter l'échange et la gestion du contenu et des données orientées transaction entre sites Web. Le but est de fournir des règles d'affaires qui facilitent l'échange des données grâce à des étiquettes de balise (rassemblées dans des profils) dont le sens est convenu par des communautés plus ou moins étendues. La compatibilité sera assurée avec le *Open Profiling Standard*. Avec ICE, les producteurs de contenu n'auront pas à utiliser différents formats et méthodes de transfert. Cette norme comporte également certaines conventions spécifiques de métadonnées : date d'embargo, date de destruction. (Consortium dirigé par Vignette et Firefly, avec l'appui de Microsoft. Travaux amorcés en février 1998).

* **MCF** (*Meta Content Format*) : proposition de norme pour l'Internet en cours d'élaboration en 1997. Tous les moyens d'organiser le contenu y sont décrits dans une syntaxe basée sur le langage XML et une sémantique restreinte de catégories (descripteur, classification, table des matières...) et d'attributs (domaine, portée, typeDe, unité). Le format MCF offre un langage de représentation et un langage de requête pour diverses structures (répertoires de fichiers, hiérarchies, catégories de sujets, répertoires de personnes) d'une façon qui n'est pas liée à leurs supports compartimentés. La proposition MCF est supposée avoir été entièrement incluse dans la norme RDF.

* **ODMA** (*Open Document Management API*) : l'interface de gestion ouverte de document est une convention d'un consortium visant à faciliter l'intégration d'applications sur les documents électroniques et à permettre leur usage (version 1.5, 1997). Des « composantes conformes » peuvent, par exemple, cacher plusieurs fonctions de gestion documentaire derrière des menus de logiciel de traitement de texte. Son avenir incertain eu égard au développement pour le Web des formats DOM et RDF qui se présentent comme des solutions de plus grande envergure.

* **PGML** (*Precision Graphics Markup Language*) : proposition de norme au W3C par Adobe, Sun, Netscape et IBM, qui consiste en un format XML pour les graphiques construits par vecteurs. Son importance vient du fait que son modèle image est le même que celui des formats PostScript et PDF, produits d'Adobe et normes *de facto*. Seulement quelques modifications sont nécessaires pour faire des logiciels d'imagétique des outils de création de documents PGML. Le langage PGML est implanté en Java et rend possible une précision accrue de la publication sur le Web et l'uniformité de la présentation à l'écran et de l'impression sur papier. Il apporte son vocabulaire : une *figure* (*drawing*) comprend un ou plusieurs objets graphiques : un *tracé* (*path*), une *forme* (*shape*), une *image* (*image*), du *texte* (*text*). Propositttion initiale : http://www.w3.org/TR/1998/NOTE-PGML-19980410.html

* **PICS** (*Platform for Internet Content Selection*) : format pour la classification du contenu véhiculé sur les protocoles de transport HTTP (Web) et SMTP (messagerie). Dans la pratique, nombre de sites Web sont étiquetés selon des degrés pour la rectitude-vulgarité (*profanity*), pour la sexualité, la violence, le jeu d'argent, etc. Les navigateurs peuvent être programmés pour ignorer les sites jugés indésirables en fonction de ces étiquettes et choisir

ceux qui sont jugés désirables. Les préoccupations des parents à l'égard de leurs enfants sont la principale motivation de cet effort. Cette étiquette peut être signée numériquement (Dsig Label).

* **RDF** (*Resource Description Framework*) : basé sur la syntaxe XML, le format RDF offre un langage pour encadrer la description des ressources via les métadonnées. La carte de sujets d'un site Web peut être générée automatiquement en fonction des métadonnées des documents qu'il contient. Ces métadonnées sont aussi utiles aux moteurs de recherche pour obtenir un repérage d'une plus grande précision. Parmi les usages, le langage RDF facilite le recours aux multiples vocabulaires d'évaluation de sites, par exemple (voir PICS). Il comporte un modèle syntaxique (*RDF Model and Syntax*) et un vocabulaire de *base (RDF Schema*).

* **SSML** (*Speech Synthesis Markup Language*) : langage pour le balisage élocutoire, avec le marquage de la prononciation et de la prosodie dans un texte en vue de son utilisation éventuelle par des synthétiseurs vocaux (en développement).

* **TML** (*Tutorial Markup Language*) : le langage de balisage pour l'apprentissage est un format d'échange conçu pour séparer le contenu sémantique d'une question de son mode de présentation. Plusieurs types de question sont possibles pour avoir accès aux automatismes TML de présentation à l'apprenant via le Web, de réception des réponses et de leur analyse. TML fournit des étiquettes spéciales de balise pour les questions : *Multiple-Choice, Poly-Choice, Word-Match, Hot-Image*. Un fichier typique TML comprend un titre, le texte de la question en HTML, les réponses correctes et incorrectes, un ensemble de réactions aux réponses, un ensemble de résultats liés aux réponses et un ensemble d'indices pour aider (*hints*).

* **WebDAV** (*Distributed Authoring and Versioning on the Web*) : spécifie un ensemble de méthodes et de types de contenu pour la gestion des métadonnées dans un processus de rédaction en collaboration, pour un verrouillage simple des ressources et un contrôle des versions.

* **X-ACT** (*XML Active Content Technologies*) : lancée en mars 1998, initiative de promotion de l'utilisation de la norme XML pour des applications nouvelles sur le Web, au-delà des activités conventionnelles de diffusion. Ces applications nouvelles sont basées sur l'échange d'information structurée entre toutes les ressources en réseau (poste de travail, réseau local, ordinateurs centraux, Internet, intranet, etc.). La caractéristique nouvelle est le « contenu actif » représenté en XML, qu'il s'agisse de documents, tant les données que les métadonnées, de toute donnée, de tout objet, de tout modèle qui peuvent être réutilisés, y compris être transformés, dans toute application en tout point rejoint par le réseau. La représentation en XML sert de véhicule à ces contenus et à ces structures logiques.

* **XLL** (*Extensible Link Language*) : le langage extensible de liens est une norme associée aux documents XML dans l'architecture du W3C. Il emprunte aux concepts de *HyTime* et de *Text Encoding Initiative,* et tout en supportant les formats URL existants, il permet la création de liens bidirectionnels entre les deux noeuds reliés, il permet de pointer dans l'arborescence d'un document, par exemple dans une position hiérarchique correspondant à

une section précise ; il permet aussi d'interposer un service de résolution d'adresse sur le lien entre deux noeuds de façon à n'avoir à modifier l'adresse d'un document à un seul endroit plutôt que dans toutes les instances du document ayant ce lien. Le langage XLL a été divisé en deux sections : Xpointer (hyperlien interne du document) et Xlink (entre documents).

* **XML** (*Extensible Markup Language)* : adopté comme norme (XML 1.0) par le W3C en février 1998, le langage extensible de document sert à décrire une classe d'objets de données, des documents XML. La norme XML est une syntaxe servant à représenter des données structurées au moyen de balises étiquetées avec un vocabulaire convenu dans un domaine. Elle ne fournit pas ce vocabulaire. On s'attend à ce que les institutions et les industries se dotent de vocabulaires convenus pour faciliter les échanges (genre EDI), comme cela s'est produit depuis quelques années avec la norme SGML pour les industries de l'automobile, pharmaceutique, des microprocesseurs, etc. La structure de représentation comporte commodément une branche physique et une branche logique. Les documents XML sont composés d'un ou des deux types d'entité (unités d'emmagasinage) suivants :

  - texte : composé de caractères qui soit représentent le contenu du document, soit sont des caractères liés au balisage et au marquage. Ce marquage encode une description de l'arrangement du stockage du document, de sa structure et des paires arbitraires d'attribut-valeur associées aux éléments de la structure ;

  - données binaires : elles ne sont pas traitées par le processeur XML mais par l'indication de la notation dans laquelle se trouvent les données binaires.

* **XML-Data** : proposition de Microsoft au W3C pour distinguer les données XML en fonction de leur type (numérique, date, texte, chaîne de caractères) de façon à améliorer les occasions propices de l'automatisation des échanges entre applications.

* **XSL** (*Extensible Style Language*) : langage extensible de style qui établit la relation entre les données XML et leur affichage, leur impression ou toute forme d'exécution multimédia du contenu de ces données XML.