

# L'intégration des technologies XML dans le commerce électronique professionnel

Youcef AKLOUF  
Université des Sciences et de la Technologie Houari Boumediene  
Faculté d'Electronique et d'Informatique  
Département d'Informatique  
Laboratoire de recherche en Intelligence Artificielle  
[Yaklouf@usthb.dz](mailto:Yaklouf@usthb.dz), [aklouf@wissal.dz](mailto:aklouf@wissal.dz)

## Résumé

Le commerce électronique professionnel (Business To Business) concerne l'informatisation des échanges interentreprises ou d'un autre terme dresser un système d'information inter-organisation qui permet de faire tout type d'échanges d'information d'une manière automatique et sans faire recours aux humains.

La première opération à faire dans ce type de commerce est la réorganisation des applications internes de l'entreprise afin qu'elle soit adaptée a cette nouvelle tendance d'échange, c'est le rôle de l'EAI, la deuxième étape est la réalisation des mécanismes d'échanges proprement dite, c'est le B2B. Dans cet article nous allons faire un petit survol sur les technologies XML utilisés dans l'EAI et le B2B, les deux standards les plus utilisés à savoir ebXML et RosettaNet seront étudiés en détail, finalement un modèle de convergence regroupant la plupart des acteurs du commerce sera présenté, ce modèle en couche défini les couches essentielles pour la réalisation d'un système B2B

**Mots clés** : XML, Intégration des applications, échanges interentreprises, ebXML, RosettaNet.

## 1. Introduction

XML est une technologie incontournable dans l'entreprise. On retrouve XML dans des Contextes aussi divers que les applications distribuées, la configuration de produits, les annuaires, l'édition de documents, la diffusion de contenus sur le web ou la gestion de la connaissance. XML est plus qu'un langage, c'est un méta-langage. Actuellement on estime que plusieurs centaines de langages basés sur XML ont été décrits, preuve que XML répond à un véritable besoin. Le problème est qu'actuellement le monde XML est proche de la tour de Babel. En effet, il est difficile de comprendre ce qui se passe. L'objectif est donc d'arriver à classer rapidement une technologie. Pour ceci les développeurs de la famille XML proposent quatre catégories, les langages XML pour le *Business to Customer* (B2C), les langages XML pour les services web, les langages XML pour l'*Enterprise Application Integration* (EAI) et les langages XML pour le *Business to*

*Business* (B2B). Notre centre d'intérêt tourne autour des deux dernières catégories à savoir l'EAI et le B2B. Dans ce qui suit nous allons graduer la complexité de ces systèmes, ainsi que la nature des échanges intra et interentreprises.

## 2. EAI - Définition

Le but de l'EAI (Enterprise Application Integration) est de faire communiquer et collaborer les applications d'une même entreprise de manière non-intrusive. L'intégration des applications est devenue l'une des priorités des entreprises. Elle permet d'uniformiser leur système informatique. En effet, pour le moment les systèmes informatiques sont souvent composés d'applications disparates incapables de communiquer. Prenons l'exemple d'une commande client. Si les applications ne savent pas communiquer, cette commande sera saisie en plusieurs endroits :

- dans le logiciel de traitement des commandes ;
- dans le logiciel de facturation ;
- etc.

Il peut y avoir des erreurs lors de ces multiples saisies. L'intégration des différentes applications permet l'échange de ces informations, et ainsi d'éviter les doublons, les informations erronées et d'améliorer les possibilités de traçabilité des commandes.

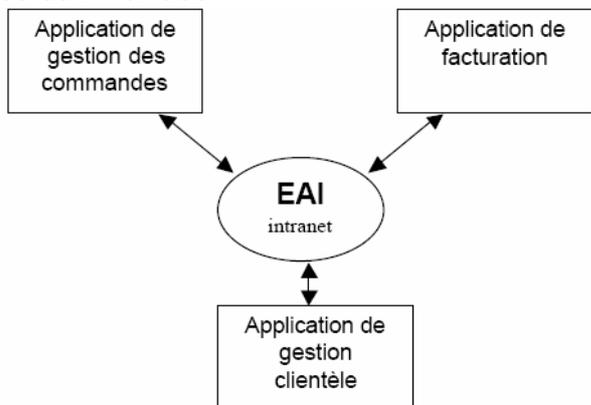


Figure 1 : EAI

Un EAI doit remplir 4 fonctions [2]:

1. Interfacer : extraire et injecter des données dans une application. Pour cela on utilise les services Web (et en particulier WSDL(Web Service Description Language)), qui ont l'avantage de pouvoir être accédés à partir de n'importe quel langage de programmation qui fonctionne sur n'importe quelle plate-forme. WSDL permet de donner une description normalisée des fonctionnalités offertes par les applications.

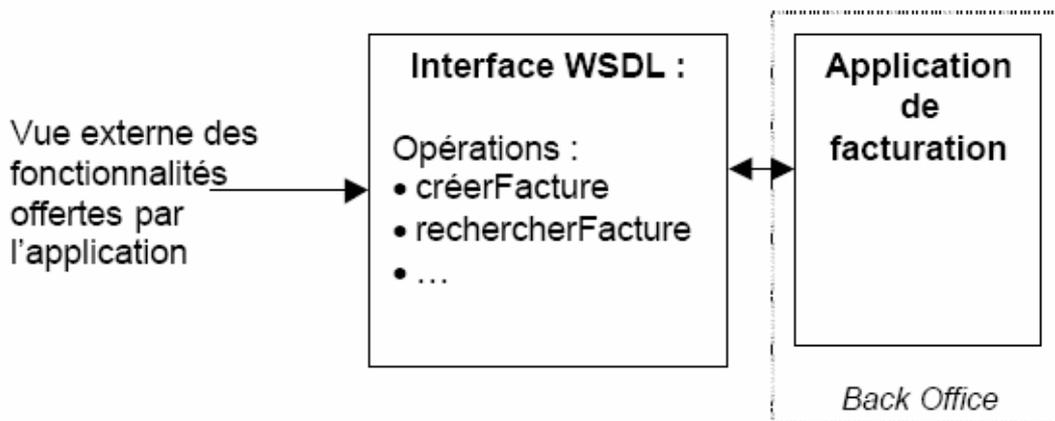


Figure 2 : EAI Interfacer

2. Transformer : convertir les données. C'est le rôle de XSLT. Cela permet en particulier de convertir des informations comprises par une application en informations comprises par une autre application. Par exemple, si une application de gestion des commandes

transmet une description de commandes à une application de facturation dans le but de générer une facture, et que ces deux applications n'ont pas la même façon de modéliser une commande, une opération de transformation est nécessaire :

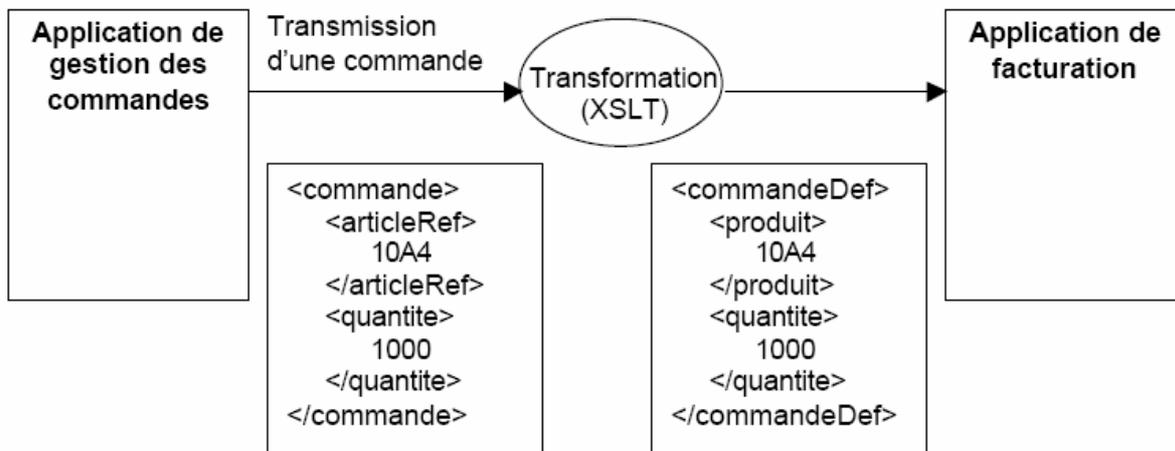


Figure 3 : EAI Transformer

3. Router : transporter les données vers le destinataire. Pour cela on utilise SOAP, qui peut être utilisé pour une communication synchrone ou asynchrone.

4. Gérer : suivre l'état du processus. Pour cela on utilise la gestion des processus métiers avec BPML, comme expliqué ci dessous.

### 2.1 Définition des processus métiers

Afin de permettre l'intégration d'applications, les technologies XML doivent comprendre le fonctionnement interne de l'entreprise. La première étape est de permettre aux applications de communiquer : c'est le besoin adressé par les normes d'infrastructures telles que SOAP. L'étape suivante est de formaliser l'exécution d'activités par des applications de manière collaborative dans le but d'atteindre un objectif métier : c'est l'objectif des processus métiers.

Formellement, un processus métier peut être défini comme un enchaînement d'activités incluant une interaction entre participants, dans le but de réaliser un objectif métier. Dans un processus métier, on tient compte des différents participants d'une opération, de leur rôle, de l'objectif de cette opération et des moyens mis en

oeuvre (messages, documents). Un processus métier décrit des activités et leur séquençement. Le problème de la représentation des processus métiers n'est pas nouveau : plusieurs produits existent déjà pour permettre leur définition et leur exécution (Excelon B2Bis, BEA Weblogic Process Integrator, MEGA Process, Biztalk Server, etc.), mais ce sont des solutions propriétaires.

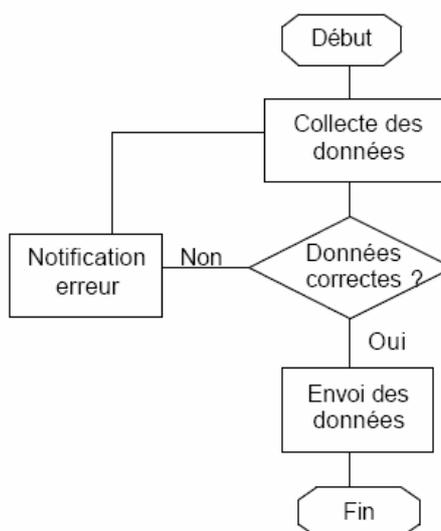


Figure 4 : Exemple de processus métiers

Un processus métier est interne à une entreprise et une seule. Il décrit les activités nécessitant la collaboration de plusieurs entités de cette même entreprise.

## 2.2 Les Dialogues :

Un dialogue définit une collaboration entre plusieurs applications sous la forme d'échanges de messages. Un dialogue comprend :

- La définition des messages échangés entre partenaires (exemple : demande de devis, réponse à une demande de devis, etc.) ;
- Le séquençement de ces messages dans le but de réaliser un objectif commun (exemple : pour obtenir un devis, on envoie tout d'abord un message 'demande de devis' à un fournisseur, et on attend un acquittement, puis la réponse de type 'réponse à une demande de devis') ;
- Les rôles des différents partenaires.

Il existe de nombreuses normes pour la définition de dialogues : RosettaNet, ebXML et Biztalk.org, etc.

Par abus de langage, on dit souvent que ces normes définissent des processus métiers (Business Process) inter-entreprise permettant à plusieurs entreprises de collaborer. Pour éviter toute confusion, nous utiliserons le terme de « dialogue » pour les caractériser, et nous détaillerons ce concept et les normes associées (ebXML et RosettaNet) dans la partie dédiée au B2B.

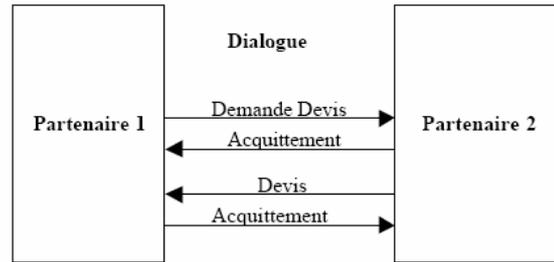


Figure 5 : Dialogue

L'EAI reste fondamentalement intra-entreprise, les problématiques pour l'intégration de plusieurs entreprises sont tout autres et sont abordés dans le B2B.

## 2.3 Les processus e-business

Un dialogue fait abstraction des processus métiers implémentés par chaque partenaire. Pour lier dialogues et processus métiers, BPMI.org (Business Process Management Initiative, regroupement d'acteurs de l'EAI et du B2B) définit la notion de processus e-business.

Un processus e-business est composé de deux parties :

- Une interface publique : définition du dialogue (RosettaNet, ebXML, Biztalk.org ou autre), précisant l'interaction entre les participants du processus e-business. Elle est commune à tous les participants.
- Deux interfaces privées (une par partenaire) : définition des processus métiers implémentés par chaque partenaire.

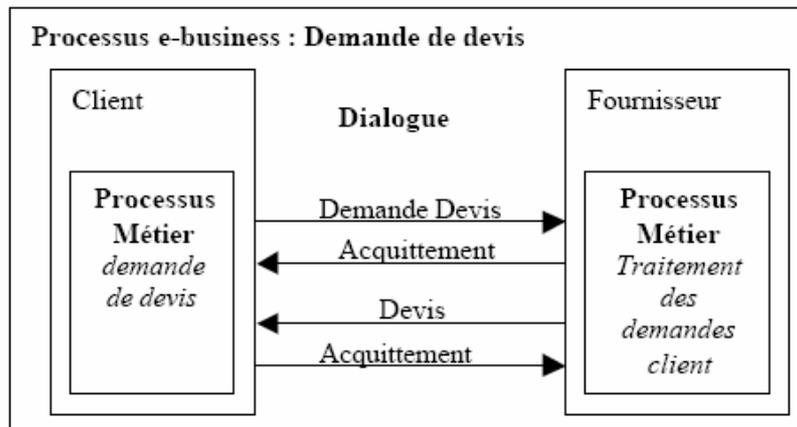


Figure 6 : Processus e-business (le dialogue ne connaît pas le processus métiers)

Figure 7 : B2B

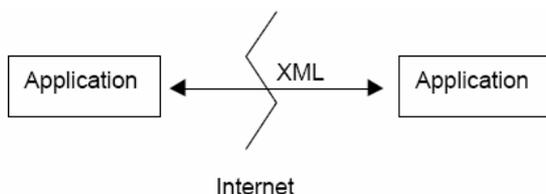
### 3. Le commerce électronique professionnel (B2B) [5]

L'EAI permet de connecter les applications d'une entreprise et ainsi d'avoir un système informatique uniformisé. L'étape suivante est de connecter ce système informatique à celui de ses partenaires dans le but de faire des transactions commerciales électroniques. C'est l'objectif des normes XML définissant le B2B : Business To Business. Le but est de résoudre les problèmes récurrents : quelle infrastructure de communication utiliser, comment représenter mes requêtes (demande de devis, etc.) de manière formalisée afin que tous mes fournisseurs puissent me comprendre, comment découvrir de nouveaux partenaires et établir un dialogue, etc.

On répertorie deux types de normes adressant les besoins du B2B :

- Des normes d'infrastructure (Horizontales), définissant les infrastructures de base : communication, référentiels, etc. Les normes d'infrastructures permettent aux applications de communiquer. Parmi ces normes on retrouve SOAP, WSDL, UDDI et ebXML. ebXML sera détaillé dans la suite de cet article ;

- Des normes de contenu (Verticales) adressant la sémantique des informations échangées et les dialogues. Les dialogues permettent aux applications de se comprendre et de réaliser des transactions par le biais d'échanges de messages XML, dont le contenu (bon de commande, facture, etc.) est normalisé. L'objectif final est que toutes les entreprises utilisent les mêmes documents.



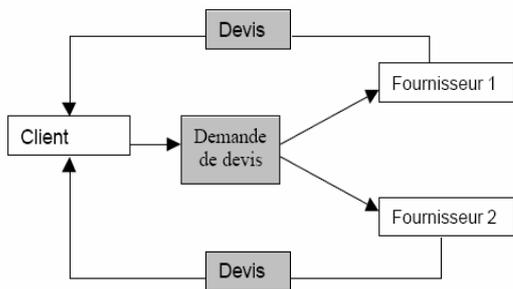


Figure 8 : Exemple de dialogue B2B

### 3.1 Norme d'infrastructure : ebXML [1]

ebXML est une initiative commune des Nations Unies (United Nation/CEFACT) et d'OASIS (Organization for the Advancement of Structured Information Standards) dans le but de définir une architecture XML pour créer "un marché électronique unique et global", où des entreprises de n'importe quelle taille et de n'importe quelle provenance géographique peuvent se rencontrer et interagir via l'échange de messages XML. L'objectif est d'ouvrir l'EDI aux PME-PMI. La version 1.0 de la norme a été approuvée à Vienne le 14 Mai 2001. ebXML fournit un référentiel comme UDDI, mais ne se limite pas à cette fonctionnalité : ebXML va plus loin en introduisant la notion de processus métier (*business process*), de profile de collaboration ainsi que d'accord de collaboration (*Collaboration Protocol Agreement ou CPA*). On tient ici compte du métier de l'entreprise et de sa façon de travailler.

Les collaborations B2B nécessitent plus qu'un protocole XML et un référentiel. ebXML fournit une architecture globale définissant le commerce électronique sous tous ses aspects : infrastructure (messages, transport, repository, etc.), support pour définir les processus métiers, les accords de collaboration, la description des différents acteurs, les

conditions de sécurité, l'interopérabilité, etc. En particulier, ebXML ne spécifie pas le type ou le contenu des transactions métiers. Il s'agit d'une norme d'infrastructure avec laquelle on peut échanger des documents de type RosettaNet, OAGI ou encore ses propres documents. Le groupe des « Core Components » (qui n'a pas réussi à terminer sa spécification dans les temps pour faire partie de la norme 1.0) travaille sur la spécification d'éléments d'information (tels que Date, Adresse, ...) qui peuvent être re-utilisés dans des schéma. Des groupes de travail tels que l'OAGI suivent de très près les travaux pour pouvoir ré-écrire leur norme avec une base de Core Components.

#### 3.1.1 Architecture

L'architecture ebXML fournit :

- Une méthode de description des entreprises ;
- Une méthode de description des processus métiers, les documents associés (DTD ou XML-Schéma), un vocabulaire XML commun, et les échanges de messages XML associés ;
- Un repository commun où les entreprises stockent leurs informations, enregistrent et découvrent des services proposés ;
- Un repository où les entreprises peuvent stocker leurs business models avec les messages associés ;
- Une méthode de définition et de négociation d'accords de collaboration, ou *collaboration partner agreement (CPA)*. Le langage TpaML d'IBM (*Trading Partner Agreement Modeling Language*) permet en particulier de gérer les négociations et de définir les accords de collaboration entre partenaires ;
- Une architecture de transport de messages XML, utilisant des standards tels que : HTTP, TCP/IP, Mime, SMTP et FTP. A noter que cette architecture

intègre SOAP. L'annuaire ebXML est plus complet que celui d'UDDI, dans le sens où UDDI ne permet que de référencer des entreprises et les services Web proposés.

### 3.1.2 Scénario d'utilisation

L'entreprise A cherche un partenaire pour l'achat d'un produit spécifique. L'entreprise B est à même de répondre à sa demande. Dans le scénario suivant nous expliquons les étapes nécessaires pour que ces deux entreprises puissent se découvrir, et inter opérer.

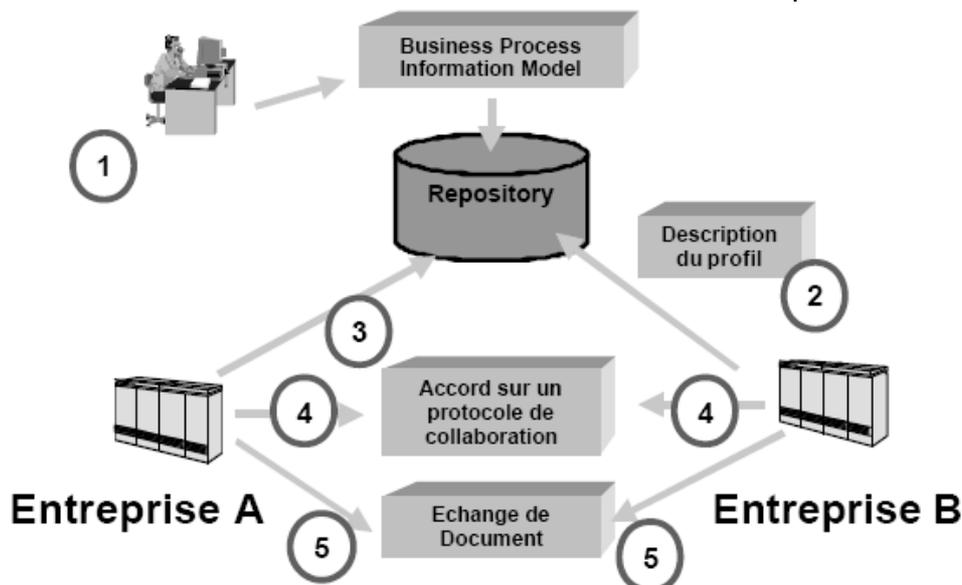


Figure 9 : Scénario d'utilisation de ebXML

1. Un organisme de normalisation (RosettaNet, OAGIS, ...) définit le processus 'Demande de devis', sous forme de *Business Process Information Model*, et l'enregistre dans le repository ebXML. Ce modèle précise la séquence des messages échangés au cours du processus, et fournit les documents métiers, communs à toute la profession (devis, formulaire de demande de devis, acquittement, reçu, etc.);

2. L'entreprise B décrit son profil et s'enregistre dans le référentiel ebXML. Elle précise qu'elle fournit le processus 'Demande de devis', et enregistre les détails de l'implémentation qu'elle fournit ;

3. L'entreprise A recherche la liste des entreprises vendant un produit particulier, et fournissant le processus 'Demande de devis' dans le référentiel ebXML, et trouve l'entreprise B ;

4. Optionnellement, les entreprises A et B négocient le protocole de collaboration : infrastructure de communication, format des messages échangés, conditions de sécurité, etc., et produisent un document TPA, base de la collaboration ;

5. Les entreprises A et B s'échangent les documents métiers (demande de devis, réponse, ...) en suivant les règles de séquence du processus défini. Ces documents peuvent alors être utilisés par les applications des entreprises A et B.

### 3.1.3 Modélisation des processus métier (ebBPSS)

Un processus métier ebXML est en fait une spécification de « Collaboration métier ». Le terme processus métier est souvent utilisé comme qualificatif pour désigner des concepts assez différents. Il est bon de distinguer les processus métiers d'entreprise, les processus métiers exécutables et les collaborations

métiers. Les processus métiers d'entreprise documentent les étapes logiques qui se succèdent pour atteindre un but commercial donné. Ils ne font référence à aucun système et peuvent souvent documenter des étapes qui sont exécutées chez des partenaires commerciaux. Cette modélisation permet de documenter et communiquer la succession d'opérations avec pour but dans, certains cas, d'optimiser les processus. Les processus métiers exécutables sont définis en association avec un moteur de processus métiers. Les moteurs modernes permettent de spécifier des processus métiers qui font intervenir des échanges de messages avec des partenaires commerciaux, des systèmes d'entreprises et des employés chargés de gérer les exceptions. Les « collaborations métiers » représentent le dialogue entre deux processus métiers exécutables. La combinaison d'un document BPSS, deux CPPs et un CPA constitue un contrat commercial et technique qui définit sans ambiguïté, tous les paramètres du dialogue entre deux partenaires. La norme BPSS précise:

- Les transactions métiers qui sont constituées d'un échange de messages et de signaux. Les messages contiennent eux même un ou plusieurs Business

Documents regroupés dans une enveloppe de documents. Tout type de document peut être échangé (Binaires, XML, documents signés, ...);

- Les séquencements de ces transactions (et non pas des messages) dans le but de réaliser une collaboration particulière ;
- Les rôles des partenaires de la transaction.

- La norme peut aussi être utilisée pour spécifier des collaborations qui font intervenir plus de deux partenaires (*Multi-party Collaborations*)

- La norme BPSS a été conçue comme un sous-ensemble sémantique de la méthodologie de modélisation définie par UN/CEFACT : UMM, UN/CEFACT Unified Modeling Methodology. Le livrable de la norme est un document XML qui respecte le schéma défini par la norme. La norme ne propose pas de notation particulière pour représenter les collaborations. La norme BPSS fait souvent référence au composant « *Business Service Interface* Ce composant logique (qui ne fait pas partie de la norme) a pour rôle de contrôler l'exécution du contrat indépendamment de l'architecture sous-jacente (dans ce cas un moteur de processus métiers) :

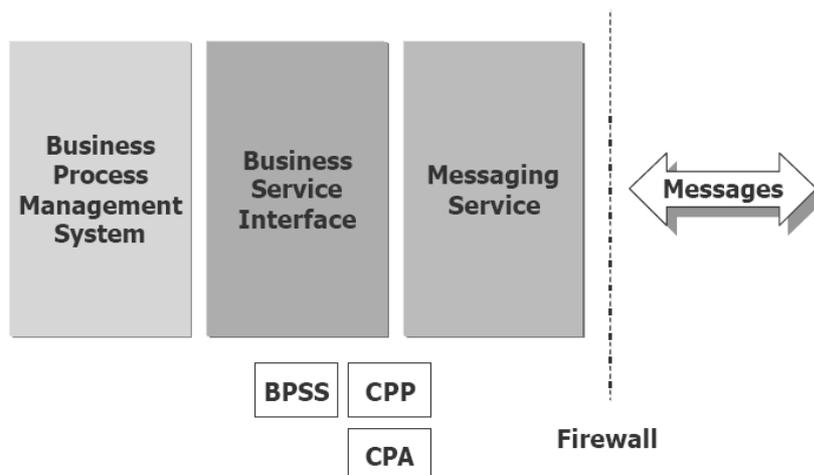


Figure 10 : Vue logique de l'architecture d'un système ebXML

### 3.1.4 Le protocole de messagerie

SOAP ne suffisait pas comme protocole d'échange de messages lors de transactions B2B. En effet, il est nécessaire d'ajouter des notions de sécurité, d'authentification, d'intégrité, de non-répudiation des données, et avant tout de garantie de réception. Et ceci dans le but d'établir des transactions commerciales équivalentes à celles que l'on peut réaliser avec du papier, de l'encre, une signature, un tampon, une enveloppe, la poste ou un fax. En fait ce n'est pas du tout trivial de remplacer ces instruments séculaires par leurs équivalents électroniques !

Le protocole de messagerie ebXML se base sur « SOAP Messages With Attachments » pour le transport des données. SOAP Messages With Attachments (SwA) est une extension de SOAP pour permettre la transmission de pièces attachées au format MIME (à l'origine utilisé pour l'attachement de fichiers aux emails). C'est un candidat à la recommandation par le W3C. Par ce biais, la messagerie ebXML se rend indépendante du protocole utilisé : HTTP, SMTP, ...

ebXML définit un ensemble de balises permettant la gestion de la sécurité, etc., au dessus de SOAP. Structure d'un message ebXML :

- Un message SwA est transporté par un protocole de communication (HTTP, SMTP, etc.). A ce titre, le message est contenu dans une enveloppe spécifique à ce protocole ;
- Le contenu est sous format MIME : on a donc ensuite une enveloppe MIME ;
- La première partie de ce message MIME contient l'enveloppe SOAP, qui contient l'entête et le corps SOAP. ebXML défini ensuite le format des éléments contenus dans l'entête et le corps SOAP,

pour répondre à ses besoins spécifiques de sécurité, authentification, etc.

- La seconde partie du message MIME contient les messages que s'échangent les partenaires de la transaction.

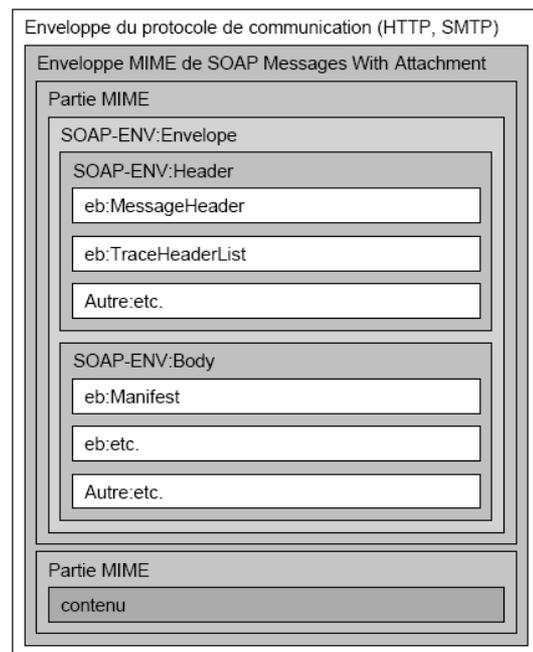


Figure 11 : Structure d'un message SOAP

### 3.1.5 Les accords de protocole de collaboration

Pour faciliter les collaborations, les partenaires d'une transaction ont besoin d'un mécanisme pour publier des informations sur les processus métiers qu'ils supportent, et sur les détails techniques de leurs implémentations (en particulier les protocoles d'accès). C'est l'objectif des *Collaboration Protocol Profile* ou *CPP* (profile d'accord de collaboration). Un CPP, enregistré sur le registre ebXML, est en quelque sorte la vitrine d'un partenaire, précisant :

- la classification de l'entreprise,
- la collaboration métier supportée,
- les personnes à contacter pour un processus particulier,
- le protocole de messagerie utilisé
- etc.

Un CPP supporte une seule collaboration métiers. La collection complète de CPP d'une entreprise représente un TPP (trading partner profile). La norme ne fournit pas de représentation particulière d'un TPP. Dans la phase de négociation, deux partenaires s'entendent sur un accord, intersection de leurs CPA : un *Collaboration Protocol Agreement*, qui précise les termes pour une collaboration métier donnée. Un CPA est en fait deux CPP qui sont associés à la même collaboration métier, et qui possède une « intersection technique » non nulle (e.g. les deux partenaires utilisent http,SSL, ...)

- Un CPA, accord de partenariat, définit comment des partenaires vont interagir lors d'une transaction. Les informations contenues dans un tel document servent de base à la mise en place du partenariat. Un CPA est valide pour une certaine durée, exactement comme un contrat commercial. La collection de tous les CPA signés par deux partenaires représente un TPA (Trading Partner Agreement). Une fois encore, la norme ne propose pas de représentation particulière pour cette entité.

### 3.2 B2B - Les spécifications de contenu

Beaucoup d'applications sont compatibles XML, et sont capables de communiquer avec d'autres applications par échange de messages XML. Cela ne signifie pourtant pas que ces applications soient capables de se comprendre. Pour que des applications puissent communiquer, il est nécessaire qu'elles s'entendent sur la structure des données échangées. Prenons un exemple simple : un client désire demander un devis à un fournisseur et formate sa requête de la manière suivante :

```
<requete>
<demandeDevis>
<refProduit> 10A45 </refProduit>
<quantité> 1000 </quantite>
</demandeDevis>
</requete>
```

Rien ne nous garantit que le fournisseur en question sera à même d'analyser et de comprendre cette requête. Il accepte peut être en entrée des requêtes formatées de la façon suivante :

```
<requeteClient>
<objet> demande de devis </objet>
<refArticle> 10A45 </refArticle>
<quantite> 1000 </quantite>
</requeteClient>
```

Même si les applications client et fournisseur parlent XML, il ne s'agit pas du même XML. XML est un méta langage, permettant de définir des langages particuliers. Les acteurs doivent ensuite s'entendre sur le langage particulier à utiliser. Un parallèle peut être fait avec le langage naturel : un français et un italien parlent tous deux une langue basée sur le même jeu de caractères. Et pourtant si ils ne trouvent pas une langue commune pour communiquer, ils ne comprendront pas les documents qu'ils s'échangeront. Les spécifications de contenu adressent

- la définition de termes communs (dictionnaires),
- la définition de messages échangés entre partenaires (par exemple une formalisation de « demande de devis ») ;

Et pour certaines :

- La définition de dialogues, séquences de messages dans le but d'atteindre un objectif commun, une description des rôles des différents partenaires. Certaines de ces spécifications, les spécifications verticales, adressent les besoins d'un corps de métier particulier. Par exemple RosettaNet fournit une spécification de contenu pour les fabricants de composants électroniques ou semi-conducteurs, et les entreprises des technologies de l'information. Il existe certainement une spécification verticale qui correspond au métier de votre entreprise. D'autres spécifications, les spécifications horizontales, sont indépendantes du métier de l'entreprise. C'est le cas d'OAGIS, qui définit par

exemple les messages liés à la commande de produits, l'envoi de factures, etc., et pouvant être utilisés quelle que soit l'activité de l'entreprise.

### 3.3 Exemple de spécification verticale : RosettaNet [3]

RosettaNet est un consortium d'entreprises de l'industrie des technologies de l'information (hard et softs) réunies pour créer des standards de dialogues métier. Le point fort de RosettaNet est d'avoir défini une structure d'échange des informations entre partenaires (RNIF - Rosettanet Implementation Framework) mais surtout d'avoir su élaborer un processus de définition et d'approbation des processus métiers qui a permis d'obtenir un consensus rapide des entreprises impliquées.

L'architecture RosettaNet comprend 3 parties : la définition des dialogues, les dictionnaires et un protocole de transport.

- La définition des dialogues ou PIP (*Partner Interface Processes*) PIP définit les interactions des différents acteurs dans le but d'accomplir un objectif métier, chaque acteur devant se conformer aux obligations spécifiées dans ce contrat. Les PIP définissent aussi les messages utilisés lors du dialogue. Un exemple de PIP est « Request Price and Availability » qui définit tous les éléments permettant à une entreprise de demander le prix et les stocks d'un produit particulier à une autre entreprise. Les spécifications des PIP définissent un dialogue en trois vues :

- Les dictionnaires établissent un langage commun pour définir les produits, les informations marketing, et les spécifications techniques. Cette partie est aussi liée à la définition des codes produits et partenaires communs, permettant d'identifier de manière unique les produits, catégories et les acteurs enregistrés ;

- Le RosettaNet Implementation Framework (RNIF) définit les formats et le protocole d'échange des messages (messages XML transportés par HTTP). Ce framework est appelé à disparaître au profit de la messagerie ebXML lorsque cette dernière aura atteint son niveau de maturité.

- Le modèle de convergence proposé dans la dernière partie explique en quoi les multiples initiatives (ebXML, RosettaNet, SOAP, UDDI, ...) sont complémentaires dans le but de mettre en place une solution d'intégration B2B.

#### 3.3.1 Les PIPs : Partner Interface Processes

Un PIP définit un dialogue entre différents acteurs dans le but d'effectuer une transaction particulière. Tous les éléments (messages échangés, interactions, éléments spécifiques à la communication point à point entre les partenaires) sont spécifiés par le PIP. Un exemple de PIP est « Request Price and Availability » qui définit tous les éléments permettant à une entreprise de demander le prix et les stocks d'un produit particulier à une autre entreprise. Les PIP sont classés en suivant l'arborescence suivante : les Clusters, groupes de processus métier, sont composés de plusieurs segments, chaque segment contenant les PIP. Il existe 7 Clusters.

Exemple de Cluster :

Cluster 3 : Order Management

- Segment A : Quote and Order Entry

- PIP 3A1 : Request Quote

Supports a process for trading partners to request and provide quotes

- PIP 3A2 : Request Price and Availability

Provides a quick, automated process for trading partners to request and provide product price and availability

...

- Segment B : Transportation and Distribution

- Segment C : Returns and Finance
- Segment D : Product Configuration

#### 6.3.1.1 Documents spécifiant un PIP :

La vue métier d'un PIP est décrite dans un document appelé « **PIP Blueprint** ». Ce document spécifie les rôles des partenaires et leurs interactions pour accomplir un objectif métier.

Un PIP est ensuite décrit formellement dans un document appelé « **PIP Specification** ».

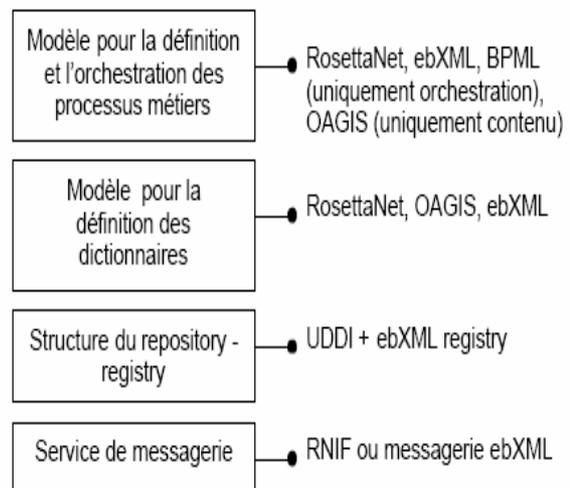
Cette spécification comporte trois parties :

- Business Operational View (BOV : Vue métier) : décrit la sémantique des entités métier et les échanges entre les différents rôles d'un dialogue métier. Le contenu de cette partie se base sur le PIP Blueprint, et contient une description formelle du système, indépendante de l'infrastructure utilisée et des messages échangés ;
- Functional Service View (FSV : Vue fonctionnelle) : décrit les composants (services et agents), et les interactions nécessaires pour exécuter un PIP. Un service est un composant répondant aux requêtes de transactions métier, et dirigeant l'exécution de ces derniers. Un agent est un composant qui agit comme un intermédiaire entre un utilisateur et un service. La vue fonctionnelle définit un protocole dérivé de la vue métier du PIP.
- Implementation Framework View (IFV : vue implémentation) : décrit le format des messages échangés entre les composants (messages guideline), et les éléments nécessaires pour la communication point à point.

### 3.3.2 Le modèle de convergence du commerce B2B

Le modèle de convergence définit comment les multiples initiatives (SOAP, ebXML, RosettaNet, ...) sont complémentaires dans le but de mettre en place une solution d'intégration B2B dans une problématique de « Supply Chain ». C'est une initiative intéressante car c'est

la première fois qu'un acteur explique la complémentarité des normes XML. Nous exposerons ci-dessous les éléments de ce modèle qui sont valables dans une problématique horizontale, c'est-à-dire indépendante du métier de l'entreprise. Pour ce modèle, une solution B2B contient quatre couches suivantes en plus des couches spécifiques au corps de métier de l'entreprise [4]:



#### a. Le service de messagerie

Le service de messagerie permet de packager, transporter et router les messages échangés lors de dialogues entre partenaires d'une transaction commerciale. La sécurité est assurée au moyen de signatures digitales au niveau du transport. RosettaNet encourage l'utilisation de RNIF, et pense passer à la messagerie ebXML dès que cette dernière sera stable.

#### b. Le structure du repository – registry

Le système de repository – registry permet à une entreprise d'enregistrer son profil, ses points d'accès, les activités qu'elle supporte, etc. C'est un mécanisme structuré permettant aux entreprises de rechercher des partenaires de façon dynamique. RosettaNet ne s'occupe pas de cette couche et préconise l'utilisation de SOAP+UDDI et du registry ebXML.

### c. Le modèle pour la définition des dictionnaires

Le modèle pour la définition des dictionnaires comprend en fait deux parties:

- La structure et le contenu du dictionnaire métier : le dictionnaire métier définit le contenu horizontal des documents métiers. Par exemple, ce dictionnaire peut préciser qu'un bon de commande comprend généralement la référence de l'article à acheter, la quantité, etc. Cette partie spécifie la structure du dictionnaire et le contenu horizontal du dictionnaire.
- La structure du dictionnaire technique : le dictionnaire technique permet de spécifier les fonctionnalités d'un produit, service, etc. Cette partie définit la structure du dictionnaire, et la manière d'obtenir des informations spécifiques à un domaine particulier. Pour cela RosettaNet recommande l'utilisation de sa norme, conjointement avec ebXML. OAGIS permet aussi de résoudre ce type de problématiques.

### d. Le modèle pour la définition et l'orchestration des processus métiers

Cette couche définit le format des messages échangés lors de transactions commerciales, et une façon de définir leur orchestration. Cette partie contient aussi la définition de dialogues indépendants du métier de l'entreprise (par exemple : demande de devis). Pour cela RosettaNet recommande l'utilisation de sa norme + ebXML. OAGIS répond à la définition du contenu des dialogues, et BPML permet uniquement de faire leur orchestration.

## 4. Conclusion

Nous avons présenté dans cet article les différentes étapes pour la mise en place d'un projet B2B dans une entreprise, ainsi que les standards les plus utilisés dans ce domaine. Ces étapes sont. La première

est la cartographie et la formalisation des processus métiers. Ainsi l'entreprise maîtrise ses ressources et uniformise son système informatique. Ce dernier est capable d'intégrer une dimension métier, et de comprendre les acteurs de l'organisation, leurs rôles, et leur façon de collaborer. La deuxième étape est de mettre en place des infrastructures B2B, permettant la connexion de ses applications avec celles de ses partenaires. L'étape finale est la mise en place de dialogues et l'adoption d'un langage commun pour pouvoir gérer des transactions avec ses partenaires. Le point fort de XML est de pouvoir s'adresser à tous ces besoins, et de mettre d'accord les différents acteurs sur des standards.

## 5. Bibliographie

- [1] ebXML. 2001. Business Process Team 10 May 2001 Business Process Analysis Worksheets and Guidelines. *Procedures for Developing Business Processes in ebXML v1.0*, July.
- [2] Li, H.: XML and Industrial Standards for Electronic Commerce. *Knowledge and Information Systems* 2:4 (2000) 487-497
- [3] RosettaNet. 2001. RosettaNet Architecture Conceptual Model, July.
- [4] BIC. 2001. XML Convergence Workgroup High-Level ConceptualModel for B2B Integration, *Business Internet Consortium*, Version: 1.0, 5 October.
- [5] XML : eXtensible Markup Language [www.w3c.org/xml](http://www.w3c.org/xml)