

Document d'origine : **ebXML Technical Architecture Specification v 1.04**  
Nom du document : ebTA\_v104\_FR  
Date d'enregistrement : 13/02/2011 22:39:00  
Nature de la traduction : Pour approbation CEN  
Version du document d'origine : 1.04  
Date d'Edition du document d'origine : 16 février 2001  
Agence de Maintenance : UN/CEFACT

---

## **Architecture technique ebXML**

## Sommaire

<b>1</b>	<b>Statut de ce Document .....</b>	<b>4</b>
<b>2</b>	<b>Participants au Groupe « Architecture Technique ebXML » .....</b>	<b>4</b>
<b>3</b>	<b>Introduction.....</b>	<b>5</b>
3.1	Sommaire .....	5
3.2	Lectorat et Aperçu.....	5
3.3	Documents Associés .....	5
3.4	Références Normatives .....	5
<b>4</b>	<b>Objectifs de Conception .....</b>	<b>6</b>
4.1	Problématique et Objectifs pour ebXML.....	6
4.2	Avertissement et Hypothèses .....	6
4.3	Conventions applicables à la Conception des Spécifications ebXML .....	6
<b>5</b>	<b>Généralités sur le Système ebXML.....</b>	<b>8</b>
<b>6</b>	<b>Méthodologie de Modélisation recommandée par ebXML .....</b>	<b>10</b>
6.1	Généralités .....	10
6.2	Vue Opérationnelle des Affaires ebXML (ebXML Operational View).....	11
6.3	Vue Fonctionnelle de Service ebXML (ebXML Functional Service View).....	13
<b>7</b>	<b>Le projet ebXML, phases fonctionnelles .....</b>	<b>15</b>
7.1	Phase de mise en oeuvre.....	15
7.2	Phase de Découverte et d'Extraction .....	15
7.3	Phase d'Exécution.....	16
<b>8</b>	<b>Infrastructure ebXML .....</b>	<b>18</b>
<b>8.1</b>	<b>Information des Partenaires Commerciaux [CPP et CPA].....</b>	<b>18</b>
8.1.1	Introduction.....	18
8.1.2	Description Formelle des Fonctions CPP .....	18
8.1.3	Description Formelle des Fonctions CPA.....	18
8.1.4	Interfaces CPP.....	19
8.1.5	Interfaces CPA.....	19
8.1.6	Détails de Mise en Œuvre (non normatifs) [CPP et CPA] .....	20
<b>8.2</b>	<b>Processus d'Affaires et Modélisation de l'Information .....</b>	<b>20</b>
8.2.1	Introduction.....	20
8.2.2	Description Formelle des Fonctions .....	21
8.2.3	Interfaces .....	22
8.2.4	Détails de Mise en Œuvre (non normatifs) .....	22
<b>8.3</b>	<b>Composants Communs et Fonctions de la Librairie Centrale .....</b>	<b>23</b>
8.3.1	Introduction.....	23
8.3.2	Description Formelle des Fonctions .....	23
8.3.3	Interfaces .....	24
8.3.4	Détails de Mise en Œuvre (non normatifs) .....	24
<b>8.4</b>	<b>Fonction du Service de Registre .....</b>	<b>25</b>
8.4.1	Introduction.....	25
8.4.2	Description Formelle des Fonctions .....	25
8.4.3	Interfaces .....	27
8.4.4	Détails de Mise en Œuvre (non normatifs) .....	27
<b>8.5</b>	<b>Fonction du Service de Messagerie.....</b>	<b>27</b>

8.5.1	Introduction.....	27
8.5.2	Description formelle des Fonctions.....	29
8.5.3	Interfaces.....	30
8.5.4	Détails de Mise en Œuvre (non normatifs).....	30
<b>9</b>	<b>Conformité.....</b>	<b>32</b>
<b>9.1</b>	<b>Introduction.....</b>	<b>32</b>
<b>9.2</b>	<b>Conformité à ebXML.....</b>	<b>32</b>
<b>9.3</b>	<b>Conformité à la Spécification d'Architecture Technique.....</b>	<b>32</b>
<b>9.4</b>	<b>Schéma Directeur des Tests de Conformité.....</b>	<b>32</b>
<b>10</b>	<b>Considérations relatives à la Sécurité.....</b>	<b>33</b>
<b>10.1</b>	<b>Introduction.....</b>	<b>33</b>

## 1 Statut de ce Document

Ce document est un Draft Final (FINAL DRAFT) pour la communauté du Commerce Electronique. La diffusion de ce document n'est pas limitée. Ce document sera soumis à la Procédure de Contrôle Qualité telle que définie par le Document ebXML, « Expression de Besoins » (« ebXML Requirements »). La présentation de ce document est basée sur la format de la RFC Standard de l'Internet Society.

***Cette version :***

ebXML\_TA\_v1.0.4.doc

***Dernière Version:***

ebXML\_TA\_v1.0.4.doc

***Précédente version :***

ebXML\_TA\_v1.0.3.doc

## 2 Participants au Groupe « Architecture Technique ebXML »

Nous remercions les personnes suivantes pour leur participation active à la rédaction de ce document :

Animateur: Brian Eisenberg, DataChannel

Editeurs: Brian Eisenberg, DataChannel  
Duane Nickull, XML Global Technologies

Participants: Colin Barham, TIE  
Al Boseman, ATPCO  
Christian Barret, GIP-MDS  
Dick Brooks, Group 8760  
Cory Casanave, DataAccess Technologies  
Robert Cunningham, Military Traffic Management Command, US Army  
Christopher Ferris, Sun Microsystems  
Anders Grangard, EDI France  
Peter Kacandes, Sun Microsystems  
Kris Ketels, SWIFT  
Piming Kuo, Worldspan  
Kyu-Chul Lee, Chungnam National University  
Henry Lowe, OMG  
Matt MacKenzie, XML Global Technologies  
Melanie McCarthy, General Motors  
Stefano Pagliani, Sun Microsystems  
Bruce Peat, eProcessSolutions  
John Petit, KPMG Consulting  
Mark Heller, MITRE  
Scott Hinkelman, IBM  
Lynne Rosenthal, NIST  
Nikola Stojanovic, Encoda Systems, Inc.  
Jeff Sutor, Sun Microsystems  
David RR Webber, XML Global Technologies

## 3 Introduction

### 3.1 Sommaire

Les mots-clés PEUT (MUST), NE PEUT PAS (MUST NOT), DOIT (SHALL), NE DOIT PAS (SHALL NOT), DEVRAIT (SHOULD), NE DEVRAIT PAS (SHOULD NOT), RECOMMANDE (RECOMMENDED), POURRAIT (MAY), OPTIONNEL (OPTIONAL) doivent, quand ils apparaissent dans ce document, être interprété suivant les recommandations de la RFC 2119 [Bra97].

Les conventions de présentation utilisées dans ce document sont les suivantes :

- *Italiques Majuscules* : mots définis dans le Glossaire ebXML.
- [NOTES : elles sont utilisées pour clarifier l'exposé ou présenter des suggestions et/ou ressources additionnelles.]

### 3.2 Lectorat et Aperçu

Ce document est destiné principalement à être lu par les participants aux groupes de projet ebXML pour les aider à conduire leurs travaux. Les autres lecteurs attendus comprennent, mais ne sont pas limités à : informaticiens, membres des instances internationales de normalisation, autres organismes et entreprises des secteurs public et privé.

Ce document décrit l'architecture sous-jacente aux applications et services ebXML. Il donne un aperçu général de ebXML et décrit les relations, les interactions et les fonctions de base d'ebXML. Il ne DOIT pas être utilisé comme un support de formation pour apprendre : (1) ce qu'est ebXML, (2) à quels problèmes ebXML peut apporter des solutions, (3) les fonctions et l'architecture qui constituent le cœur d'ebXML.

### 3.3 Documents Associés

Comme mentionné précédemment, d'autres documents fournissent des définitions détaillées des composants ebXML et de leurs interactions. Ces documents comprennent les spécifications ebXML sur les sujets suivants :

1. Expression des Besoins (Requirements)
2. Processus d'Affaires et Métamodèle de l'Information (Business Process and Information Meta Model)
3. Composants Communs (Core Components)
4. Registres et Répertoires (Registry and Repository)
5. Information des Partenaires Commerciaux (Trading Partner Information)
6. Service de Messagerie (Messaging Services)

Ces spécifications sont disponibles pour téléchargement à l'adresse suivante :: <http://www.ebxml.org>.

### 3.4 Références Normatives

Les standards suivants contiennent des spécifications qui, lorsqu'elles sont référencées dans le texte, constituent également des clauses pour cette spécification. Au moment de sa publication, les références d'édition indiquées ci-dessous étaient valides. Tous les standards sont soumis à révision, et les partenaires qui s'accordent sur l'utilisation de cette spécification sont encouragés à envisager la possibilité d'appliquer les versions les plus récentes des standards indiqués ci-dessous.

ISO/IEC 14662: Open-edition Reference Model  
ISO 11179/3 Metadata Repository  
ISO 10646: Character Encoding  
ISO 8601:2000 Date/Time/Number Data typing  
OASIS Registry/Repository Technical Specification  
RFC 2119: Keywords for use in RFC's to Indicate Requirement Levels  
UN/CEFACT Modeling Methodology (UMM)  
W3C XML v1.0 Second Edition Specification

## 4 Objectifs de Conception

### 4.1 Problématique et Objectifs pour ebXML

Depuis plus de 25 ans, l'Echange de Données Informatisé (EDI) a présenté aux entreprises la perspective d'éliminer les documents papier, de réduire les coûts et d'améliorer leur productivité en échangeant de l'information sous forme électronique. Idéalement, des compagnies de toutes tailles pouvaient conduire leurs activités de Commerce Electronique en fonction de leurs besoins, sans agrément préalable de quelque type que ce soit. Mais les promesses de l'EDI n'ont pu être tenues ; les grandes compagnies sont seules à avoir les moyens de développer des services EDI, et beaucoup de communautés EDI de commerce électronique sont centrées sur une entreprise dominante qui impose une démarche d'intégration propriétaire à ses *Partenaires Commerciaux*.

Dans les quelques dernières années, l'*Extensible Markup Language (XML)* s'est rapidement imposé comme la meilleure solution pour la définition des formats d'échange de données des nouvelles applications du *Commerce Electronique* sur l'Internet. De nombreuses personnes ont interprété la « lame de fond » XML comme une preuve évidente de la « disparition de l'EDI » – rendu totalement obsolète par l'émergence d'XML – mais ce point de vue est erroné, sur le plan économique comme sur le plan technique.

Les applications et services EDI existants ont accumulé une grande expérience en matière de *Processus d'Affaires* et les entreprises qui ont consenti des investissements importants dans l'intégration de solutions service EDI ne veulent pas les abandonner sans avoir de raisons suffisantes. XML permet d'exécuter des transactions d'affaires de manière plus ouverte et plus flexible que l'EDI. XML pourrait permettre d'établir les modèles des *Places de Marché Electroniques* de manière plus flexible et plus novatrice que l'EDI.. Mais les challenges que constituent la conception des *Messages* qui répondent aux spécifications des *Processus d'Affaires* et la standardisation de leurs sémantiques sont indépendants de la syntaxe qui encode les *Messages*.

Les spécifications ebXML proposent un cadre directeur qui peut, d'une part préserver les investissements importants de l'EDI sur les *Processus d'Affaires*, d'autre part de les exploiter dans une architecture qui utilise les nouvelles possibilités techniques offertes par XML.

Ceux qui veulent en savoir plus sur les objectifs sous-jacents à ebXML peuvent consulter la spécification des exigences de besoins pour ebXML, à l'adresse <http://www.ebxml.org>.

### 4.2 Avertissement et Hypothèses

Cette spécification est destinée à fournir une présentation d'ordre général d'ebXML, et, en tant que telle, elle ne fournit pas une description suffisamment détaillée pour construire des *Applications ebXML*, leurs composants et les services qui les utilisent. Pour accéder à une description détaillée, se référer aux spécifications détaillées ebXML consacrées spécifiquement à chaque grand composant fonctionnel de l'architecture.

### 4.3 Conventions applicables à la Conception des Spécifications ebXML

En vue d'instituer un traitement cohérent des caractères majuscules et une convention de nommage identiques pour toutes les spécifications ebXML, les styles de mise en majuscule « Upper Camel Case » (UCC) et « Lower Camel Case » (LCC) DOIVENT être utilisés. Le style UCC met en majuscule le premier caractère de chaque mot et retire les espacements. Le style LCC met en majuscule le premier caractère de chaque mot à l'exception du premier et retire les espacements.

1) les DTD, Schémas et instances de documents XML pour ebXML DOIVENT (SHALL) avoir respecter les règles suivantes :

- Les noms des éléments DOIVENT adopter la convention UCC (exemple: <UpperCamelCaseElement/>).
- Les noms des attributs DOIVENT adopter la convention LCC (exemple: <UpperCamelCaseElement lowerCamelCaseAttribute="Whatever"/>).

2) Quand UML et *Object Constrained Language (OCL)* sont utilisés pour spécifier des artefacts ebXML le règles de mise en majuscule suivantes DOIVENT être appliquées :

- Les noms de Classes, d'*Interfaces*, d'*Associations*, de Packages, d'Etats, de Cas d'Utilisation, d'Acteurs DOIVENT utiliser la convention UCC (exemples: ClassificationNode, Versionable, Active, InsertOrder, Buyer).

- Les noms d'Attributs, d'Opérations, de Rôles, de Stéréotypes, d'Instances, d'Evènements, et d'Actions DOIVENT utiliser la convention LCC (exemples: name, notifySender, resident, orderArrived).

3) Les règles générales applicables à tous les noms sont :

- Les Acronymes DOIVENT être évités, mais dans les cas où ils sont utilisés, la mise en majuscules DOIT être conservée SHALL (exemple: XMLSignature).
- Les caractères Underscore ( \_ ), points ( . ) et tirets ( - ) NE DOIVENT PAS être utilisés (ne pas utiliser : header.manifest, stock\_quote\_5, commercial-transaction, les remplacer par HeaderManifest, stockQuote5, CommercialTransaction instead).

## 5 Généralités sur le Système ebXML

La figure 1 ci-dessous montre un scénario d'échange entre deux partenaires pour une première mise en configuration et la mise en œuvre de transactions commerciales.

Ce modèle montre un exemple de processus et des étapes qui sont nécessaires pour configurer et déployer des applications ebXML et les composants associés de l'architecture.

Ces composants peuvent être mis en œuvre de manière incrémentale.

Les spécifications de ebXML ne sont pas limitées à ce simple modèle qui fournit une rapide introduction aux concepts de ebXML. Des exemples de développement ebXML sont décrits dans l'Appendice A.

Les principaux concepts utilisés sont les suivants :

1. Un mécanisme standard pour décrire les *Processus d'Affaires (Business Process)* et leur association avec le modèle d'informations.
2. Un mécanisme pour enregistrer et mettre à disposition les *Processus d'Affaires* et les *Méta modèles d'Informations (Information Meta Models)*. Ils peuvent ainsi être partagés et réutilisés.
3. La découverte des informations relatives aux partenaires comprennent :
  - Les *Processus d'Affaires* qu'ils sont capables de supporter
  - Les *Interfaces de Services d'Affaires (Business Service Interfaces)* utilisables avec les *Processus d'Affaires*
  - Les *Messages d'Affaires* qui peuvent être échangés avec les *Interfaces de Services d'Affaires*
  - La configuration technique des protocoles de transport, de sécurité et de codage
4. Des mécanismes pour enregistrer les informations précédentes, pour les présenter et les restituer.
5. Un mécanisme pour décrire la mise en œuvre d'un agrément mutuel entre deux partenaires à partir des données définies à l'alinéa 3 ci-dessous (*Agrément sur un Protocole de Collaboration (Collaboration Protocol Agreement) – CPA*)
6. Un cadre standardisé de *Services de Messagerie (Messaging Service)* interopérable, sécurisé et fiable pour les échanges entre *Partenaires Commerciaux*.
7. Une solution pour configurer les *Services de Messagerie* respectifs de manière à traiter les *Processus d'Affaires* objets d'un agrément mutuel conformément aux contraintes des accords commerciaux.

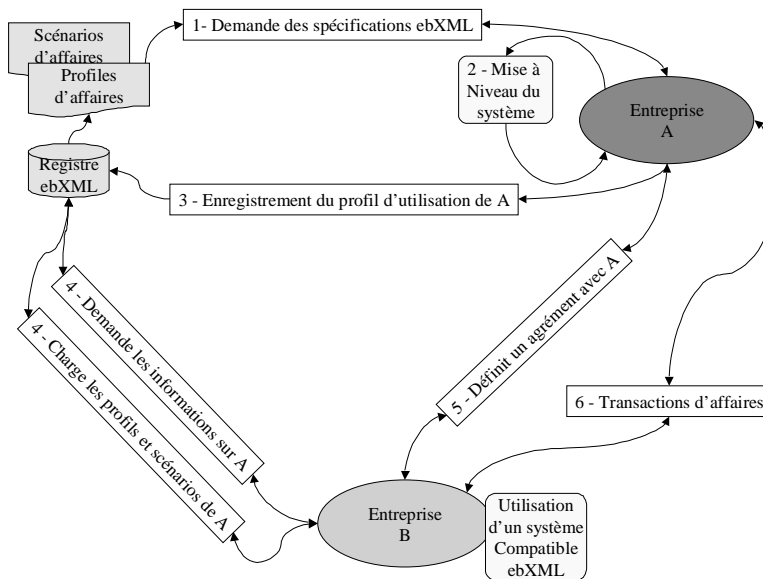


Figure 1- diagramme des interactions entre deux entreprises opérant des transactions ebXML.

Supprimé : -

Dans la Figure 1, Compagnie A est avertie qu'un *Registre ebXML* est accessible sur Internet (Figure 1, Etape 1). Compagnie A, après avoir passé en revue le contenu du *Registre ebXML*, décide de construire et déployer sa propre application conforme ebXML. (Figure 1, Etape 2). Aucun développement de logiciel personnalisé n'est



un pré-requis nécessaire à une participation au domaine d'échanges ebXML. Des applications et composants conformes ebXML peuvent également être disponibles et commercialisées prêtes à l'emploi.

Compagnie A soumet alors l'information constitutive de son propre *Profil d'Affaires* (y compris détails de mise en œuvre et liens de référencement) au *Registre* ebXML (Figure 1, Etape 3). Le profil d'affaires est soumis au *Registre* ebXML décrit les capacités et contraintes techniques, aussi bien que les scénarios d'échange supportés. Ces scénarios d'affaires sont représentés par les versions *XML* des Processus d'Affaires liés aux informations associées (ex. : calcul d'une taxe sur les ventes) que la compagnie est capable d'échanger. Après réception du profil, une vérification portant sur la conformité du format et l'utilisation du scénario est effectuée, un accusé de réception est envoyé par le Registre à Compagnie A (Figure 1, Etape 3).

Compagnie B découvre les scénarios d'affaires supportés par Compagnie A dans le *Registre* ebXML (Figure 1, Etape 4). Compagnie B envoie une requête à Compagnie A déclarant qu'elle souhaiterait engager un échange avec Compagnie A suivant un scénario d'affaires ebXML (Figure 1, Etape 5). Compagnie B fait l'acquisition d'une application conforme ebXML prête à l'emploi.

Avant de commencer à échanger, Compagnie B soumet une proposition d'accord d'affaires directement à Compagnie A. La proposition d'accord d'affaires décrit les principes des scénarios d'affaires et conventions spécifiques sur la base desquels l'accord mutuel doit être établi. L'accord d'affaires contient également les informations relatives aux spécifications de messagerie pour exécution des transactions, plans d'actions préventives et exigences de sécurité (Figure 1, Etape 5). Compagnie A accepte alors la proposition d'accord d'affaires de Compagnie B. Compagnies A et B sont alors prêtes à exécuter des opérations de Commerce Electronique en utilisant ebXML (Figure 1, Etape 6).

## 6 Méthodologie de Modélisation recommandée par ebXML

La *Modélisation des Processus d’Affaires et des Informations* n’est pas obligatoire. Cependant, si les concepteurs et les utilisateurs choisissent de modéliser les *Processus d’Affaires et les Informations*, ils DOIVENT utiliser la *Méthodologie de Modélisation de l’UN/CEFACT* (UN/CEFACT Modeling Methodology, UMM) qui elle-même utilise la notation UML.

### 6.1 Généralités

Même si les pratiques sont éminemment variables d’une organisation à une autre, la plupart des activités peuvent être décomposées en Processus d’Affaires qui sont plus génériques pour un type donné de relation d’affaires. Cette analyse à travers une modélisation des processus permettra d’identifier des *Processus d’Affaires et Méta modèles d’Information* qui sont eux candidats à la standardisation.

La démarche ebXML consiste à rechercher des composants standards et réutilisables à partir desquels il est possible de construire des documents et composants interopérables.

La méthodologie UMM est utilisée les deux vues suivantes pour décrire les différents aspects des transactions de Commerce Electronique. Ce modèle est basé sur le modèle de référence de l’EDI ouvert, ISO/IEC 14662.

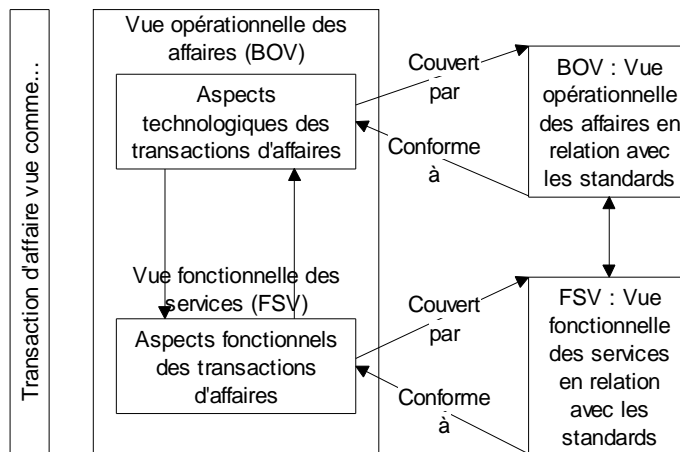


Figure 2- Méthodologie de Modélisation recommandée par ebXML.

Supprimé : -

La *Méthodologie de Modélisation de l’UN/CEFACT* se décompose en Vue Opérationnelle des Affaires (Business Operational View, BOV) et Vue Fonctionnelle des Services (Functional Service View, FSV), qui sont décrites ci-dessous.

Dans l’approche ebXML, la FSV sert de modèle de référence qui PEUT être utilisé par les concepteurs de progiciels de gestion comme guide durant le processus de développement.

L’objectif implicite de la *Méthodologie de Modélisation de l’UN/CEFACT (UMM)* est de faire la distinction entre les vues opérationnelles et les vues fonctionnelles pour assurer le niveau maximum d’interopérabilité des systèmes et de compatibilité descendante entre les systèmes existants.

Aussi, les standards résultants associés à la BOV fournissent à la *Méthodologie de Modélisation de l’UN/CEFACT (UMM)* les ressources nécessaires pour la construction de Processus d’Affaires et Métamodèles d’Information utilisables par les composants et les applications conformes ebXML.

Le BOV s’applique :

- A la sémantique des données d’affaires dans les transactions et à l’association des données dans les échanges.
- A l’architecture des transactions d’affaires, ce qui inclut :
  - Les conventions opérationnelles
  - Les agréments et arrangements
  - Les obligations et besoins réciproques

Ces spécifications s'appliquent à la description des besoins des *Partenaires Commerciaux* ebXML (ebXML *Trading Partners*).

Le FSV décrit les mécanismes de services nécessaires à ebXML. Plus particulièrement, il traite des aspects suivants des technologies de l'information :

- Les capacités fonctionnelles
- Les *Interfaces de Services d'Affaires* (*Business Service Interfaces*)
- Les protocoles et *Services de Messagerie* (*Messaging Service*)

Cela inclut de manière non limitative :

- Les ressources pour implémenter, mettre à disposition, déployer et exécuter des scénarios
- Les *Interfaces* utilisateurs
- Les *Interfaces* pour le transfert des données
- Les *Protocoles* capables d'assurer l'interopérabilité entre les développements de vocabulaire XML émanant de diverses organisations.

## 6.2 Vue Opérationnelle des Affaires ebXML (ebXML Operational View)

Les techniques de modélisation décrites dans cette section ne constituent pas des spécifications obligatoires pour la prise de participation à des transactions d'affaires conformes à ebXML.

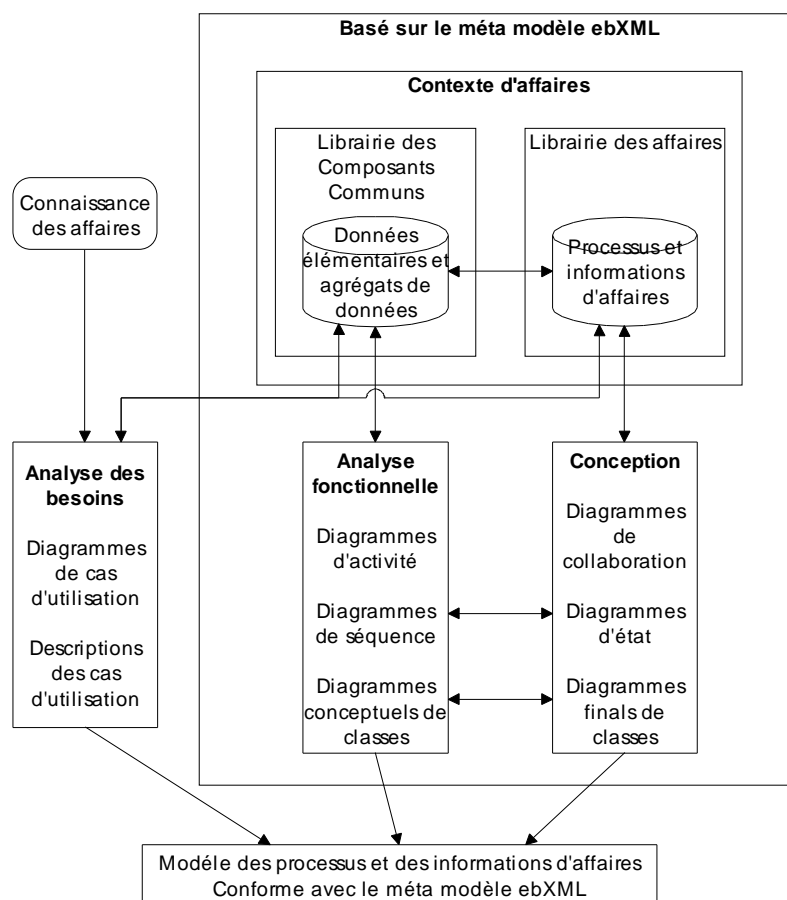


Figure 3.- Représentation Détaillée de la Vue Opérationnelle des Affaires.

Supprimé : -

Dans la Figure 3 précédente, une *Librairie Centrale (Core Library)* constitue une base de *Connaissance sur les Collaborations d’Affaires*. La Librairie Centrale contient les définitions des données et des processus, y compris les relations et références croisées, telles qu’elles sont exprimées dans la terminologie des affaires, laquelle PEUT être associée à un schéma de classification ou à une taxonomie reconnue. La *Librairie Centrale* La *Librairie Centrale* constitue le pont reliant le langage spécifique de l’industrie et des affaires et la connaissance exprimée par les modèles exprimés dans un langage plus généralisé neutre vis-à-vis du contexte.

La première phase consiste dans la définition des besoins pour décrire le problème en utilisant les *Diagrammes et Descriptions des Cas d’Utilisation*.

Si des informations de la *Librairie Centrale* peuvent être fournies par un Registre conforme à ebXML, celles-ci vont être utilisées. Si ce n’est pas le cas, de nouvelles informations à conserver par la *Librairie Centrale* vont être créées et enregistrées par un *Registre* conforme à ebXML.

La seconde phase (analyse) consiste dans la création des diagrammes de séquences et d’activités (en suivant les recommandations de la *Méthodologie de Modélisation UN/CEFACT*) qui décrivent les *Processus d’Affaires*. Les *Diagrammes de Classes* vont capturer les modules de regroupement de l’information associés aux processus (documents d’affaires). La phase d’analyse s’appuie sur la connaissance des processus détenue par la *Librairie Centrale*. Aucun effort n’est fait pour contraindre l’application de principes orientés objet. Le diagramme de classe est un diagramme des données librement structuré. Les *Processus d’Affaires* communs de la Librairie d’Affaires peuvent être référencés durant l’exécution du processus d’analyse et de conception des artefacts.

La phase de conception est la dernière étape de la standardisation. Elle PEUT être accomplie en appliquant principes orientés objet basés sur la *Méthodologie de Modélisation UN/CEFACT*. En complément les diagrammes de collaboration et d’états PEUVENT aussi être créés. Les diagrammes de la phase d’analyse seront revus pour l’harmonisation avec les autres modèles dans le même secteur ou dans des secteurs transversaux. Dans ebXML, l’interopérabilité est obtenue en utilisant de manière systématique les *Objets d’Information d’Affaires* dans tous les modèles de classe.

Les *Processus d’Affaires* sont créés par application de la *Méthodologie de Modélisation UN/CEFACT* qui utilise un ensemble commun *Objets d’Information d’Affaires* et de Composants Communs.

### 6.3 Vue Fonctionnelle de Service ebXML (ebXML Functional Service View)

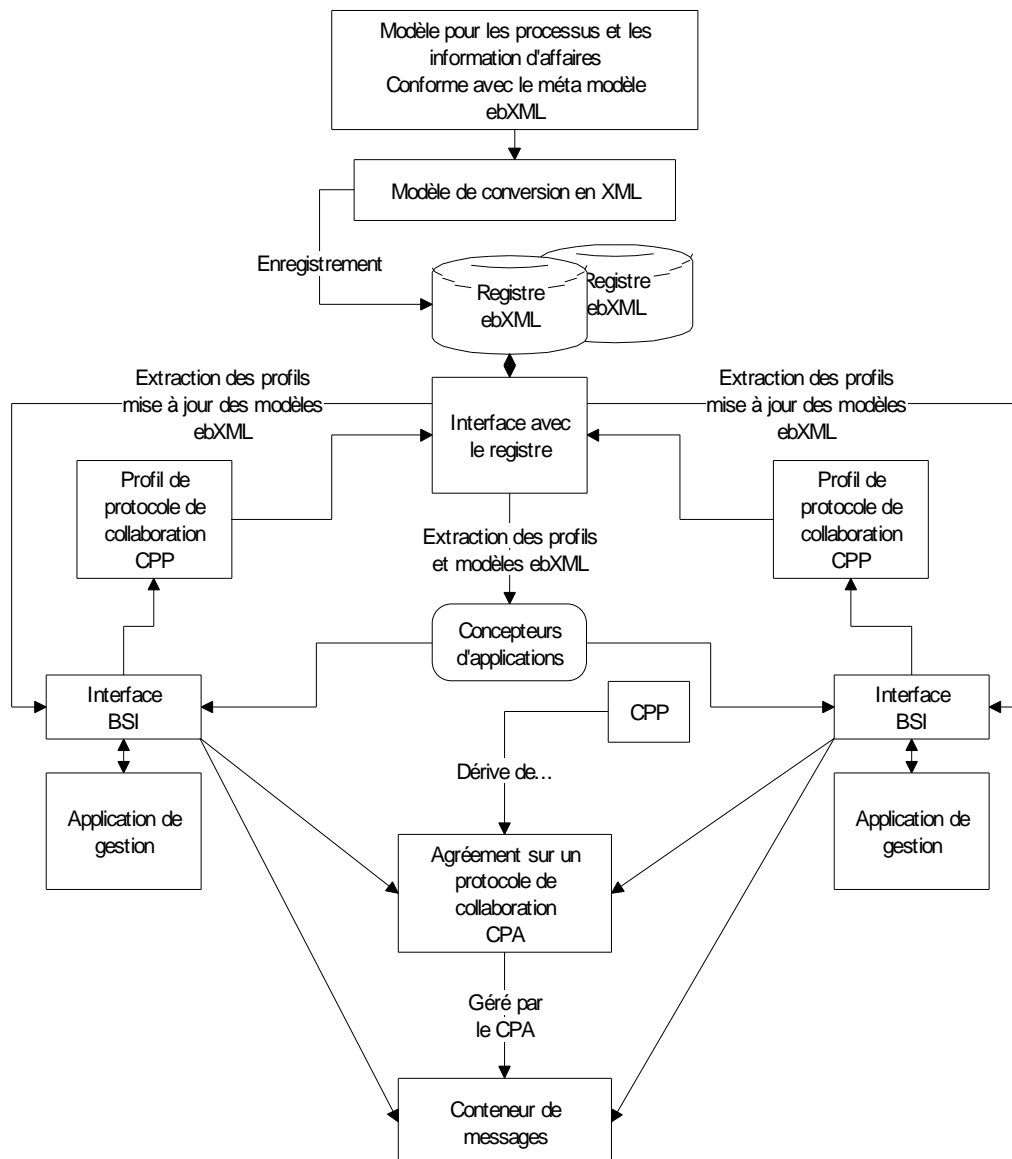


Figure 4- Représentation Détaillée de la Vue Fonctionnelle des Services.

Supprimé : -

Comme le montre la Figure 4 ci-dessus, le *Registre de Services ebXML* sert de lieu de stockage pour les *Modèles de Processus et d'Information d'Affaires*, les représentations basées sur XML de ces modèles, les *Composants Communs* et les *Profils de Protocole de Collaboration*. Les *Méta-modèles de Processus et d'Information d'Affaires* PEUVENT être archivés dans une syntaxe de modélisation, toutefois ils PEUVENT également être archivés comme syntaxe XML dans le *Registre*. Ces Informations d'affaires basées sur XML DOIVENT être exprimées de manière à permettre de découvrir les données élémentaires au travers d'une méthodologie cohérente.

L'architecture sous-jacente à ebXML est distribuée, de manière à minimiser les conséquences potentielles résultant d'un dysfonctionnement localisé en un seul point de l'infrastructure ebXML. Ce paragraphe se rapporte

à la spécification des *Services de Registre* (cf.. Fonctions des Registres, Section 8.4 pour information détaillée sur cette architecture).

## 7 Le projet ebXML, phases fonctionnelles

### 7.1 Phase de mise en oeuvre

La phase d'implémentation porte principalement sur les procédures pour créer une application à partir de l'infrastructure ebXML. Un *Partenaire Commercial* qui souhaite engager des transactions conformes à ebXML DEVRAIT en premier lieu faire l'acquisition d'une copie des spécifications ebXML;

Le *Partenaire Commercial* étudie ces spécifications et télécharge dans les *Librairies Communes* et *Librairies d'Affaires* les composants et les processus d'affaires qui lui sont nécessaires. Le *Partenaire Commercial* PEUT aussi demander à un autre *Partenaire Commercial* les informations de ses *Processus d'Affaires* (stockées dans son *Profil d'Affaires*) pour analyse et revue.

De manière parallèle, le *Partenaire Commercial* PEUT implémenter ebXML à partir d'application tiers (cette dernière intègre déjà des modèles ebXML).

Le *Partenaire Commercial* peut également soumettre l'information de son *Processus d'Affaires* à un *Service de Registre* conforme à ebXML.

La Figure 5 illustre cette interaction entre le *Service de Registre ebXML* et un *Partenaire Commercial*.

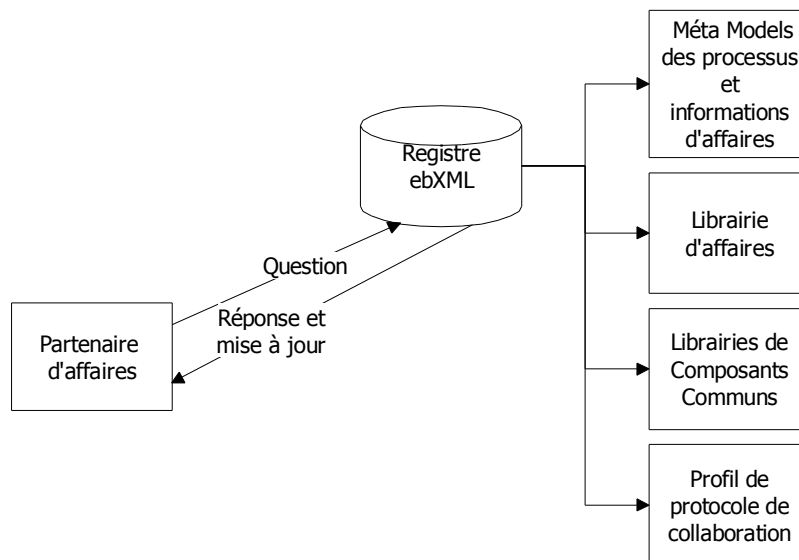


Figure 5 - Représentation Détaillée de la Vue Fonctionnelle des Services.

Supprimé : -

### 7.2 Phase de Découverte et d'Extraction

La Phase de Découverte et d'Extraction couvre tous les aspects de la découverte de ressources associées à ebXML. Un *Partenaire Commercial* qui a implémenté une *Interface de Service* ebXML peut commencer le processus de découverte et d'extraction (Figure 6 ci-dessous).

Une méthode possible de découverte peut être de demander un *Profil de Protocole de Collaboration* à un autre *Partenaire Commercial*. Les demandes de mise à jour des *Librairie Commune* et *Librairie d'Affaires* à partir de *Métamodèles de Processus et d'Informations d'Affaires*, nouveaux ou révisés, DOIVENT être supportées par l'*Interface de Service d'Affaires* ebXML. Dans cette phase, les *Partenaires Commerciaux* découvrent la signification de l'information d'affaires qui est demandée par d'autres *Partenaires Commerciaux*.

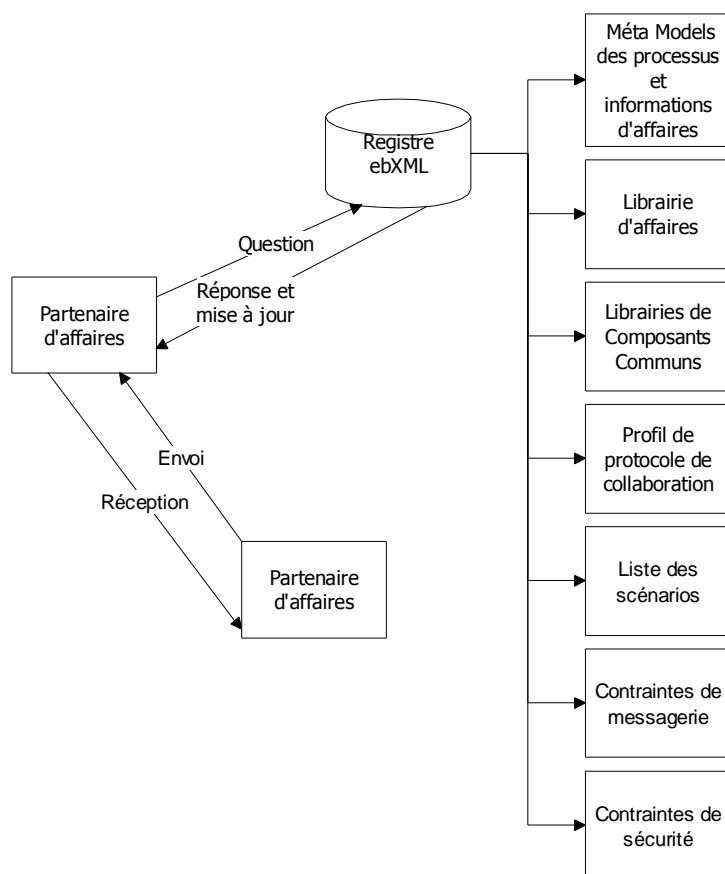


Figure 6- Vue Fonctionnelle des Services, Phase de Découverte et d'Extraction

Supprimé : -

### 7.3 Phase d'Exécution

La phase d'exécution recouvre l'exécution d'un scénario mettant en œuvre des transactions ebXML opérationnelles. Dans cette phase, des *Messages* ebXML sont échangés entre des *Partenaires Commerciaux* en utilisant un *Service de Messagerie* ebXML.

Par exemple, un CPA ebXML est un ensemble ordonné d'échanges de *Messages* d'affaires liés entre eux par chorégraphie bien définie utilisant le *Service de Messagerie* ebXML.

Il n'y a pas d'accès au répertoire pendant la phase d'exécution.

Si cela est nécessaire durant cette phase, cet accès aux répertoires doit être considéré comme une révision de la phase de découverte et de mise en place.

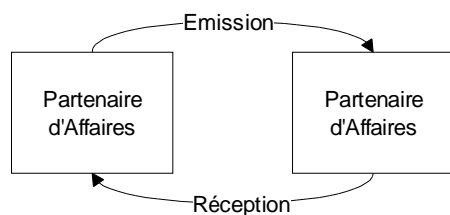


Figure 7- Vue Fonctionnelle des Services, Phase d'Exécution

Supprimé : -



[NOTE : Il n'est pas nécessaire d'accéder au *Registre* en phase d'exécution. S'il s'avère nécessaire d'appeler le registre en cours d'exécution, une telle action doit être considérée comme un retour à la phase précédente de Découverte et d'Extraction]

## 8 Infrastructure ebXML

### 8.1 Information des Partenaires Commerciaux [CPP et CPA]

#### 8.1.1 Introduction

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration* (*Collaboration Protocol Profil : CPP*). Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.

Un agrément spécial appelé *Agrément sur un Protocole de Collaboration* (*CPA : Collaboration Protocol Agreement*) dérive de l'intersection de deux CPP ou plus. Le CPA sert de "poignée de main formelle" entre deux ou plusieurs partenaires qui souhaitent exécuter des transactions d'affaires en utilisant ebXML.

#### 8.1.2 Description Formelle des Fonctions CPP

Le CPP décrit les capacités qui sont supportées par un *Partenaire Commercial* et les spécifications des *Interfaces de Service* qui ont besoin d'être « accordées » avant de pouvoir procéder à l'échange de documents d'affaires avec ce *Partenaire Commercial*.

Le CPP contient des informations essentielles au sujet du *Partenaire d'Affaires*, incluant de manière non limitative : des informations sur les coordonnées de contact, la classification par secteur d'activité, les *Processus d'Affaires* supportés, les spécifications fonctionnelles d'*Interface* et de *Service de Messagerie*. Les CPP PEUVENT également contenir des précisions sur la sécurité et des détails d'implémentation. Chaque *Partenaire Commercial* doit enregistrer ses CPP(s) dans un *Service de Registre* conforme à ebXML, lequel fournit un mécanisme de découverte permettant à des *Partenaires Commerciaux* (1) de trouver un autre (2) de découvrir les *Processus d'Affaires* supportés par d'autres *Partenaires Commerciaux*.

La définition du CPP DOIT donner les informations suffisantes pour opérer une sélection non ambiguë de choix dans tous les cas où existent des sélections (HTTP ou SMTP comme moyen de transport, par exemple).

#### 8.1.3 Description Formelle des Fonctions CPA

Le *Agrément sur un Protocole de Collaboration* est un document qui représente l'intersection entre deux CPP et qui est l'objet d'un accord mutuel entre deux *Partenaires Commerciaux* quand ils prévoient d'exécuter des opérations de *Commerce Electronique* en utilisant ebXML.

Un CPA décrit : (1) le *Service de Messagerie*, (2) les spécifications des *Processus d'Affaires* qui sont agréés par deux ou plusieurs *Partenaires Commerciaux*. Conceptuellement, ebXML supporte une vue à trois niveaux de profils imbriqués en vue de constituer des CPA aptes à déterminer des transactions du *Commerce Electronique*.

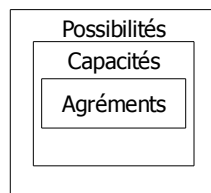


Figure 8- les trois niveaux de vues constitutifs des CPA

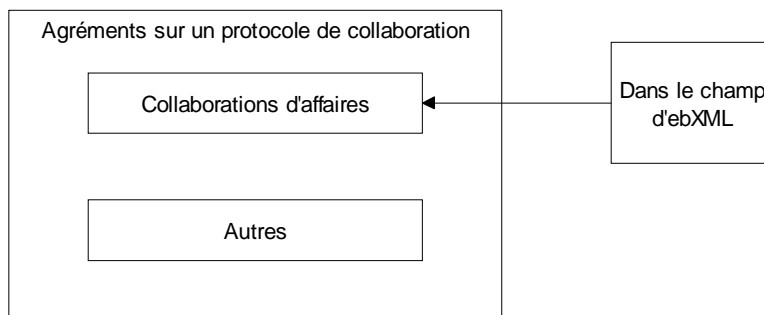
Supprimé : -

La structure de plus haut niveau rattache l'ensemble des capacités qu'un *Partenaire Commercial* pourrait supporter, au sous-ensemble de capacités qu'un *Partenaire Commercial* va réellement supporter.

Un CPA contient les spécifications de besoins de l'*Interface de Service de Messagerie* aussi bien que les détails d'implémentation relatifs aux *Processus d'Affaires* que les deux *Partenaires Commerciaux* conviennent d'utiliser

pour conduire leurs activités de Commerce Electronique. Les *Partenaires Commerciaux* peuvent décider d'enregistrer leurs *CPA* dans un *Service de Registre* conforme à ebXML mais cela n'est pas obligatoire.

Les Collaborations d'Affaires constituent le premier « engagement sur la définition de capacités » qui peut être déclaré par des *Partenaires Commerciaux* ebXML. Cet « déclaration de capacités » est facilitée par un profil distinct spécialement destiné à être publié ou référencé dans un service d'annuaire, comme un Registre ebXML, ou tout autre service disponible. La Figure 9 montre le domaine fonctionnel des *Agréments sur un Protocole de Collaboration* dans ebXML.



**Figure 9.-** *Domaine fonctionnel des CPA*

Supprimé : -

Les spécifications *CPA - CPP* incluent des annexes non normatives qui traitent de la négociation et de la composition d'un *CPA* et présentent des recommandations pour la mise en œuvre des procédures de composition et de négociation.

#### 8.1.4 Interfaces CPP

##### Interface avec les Processus d'Affaires

Un *CPP* DOIT être capable de référencer un ou plusieurs *Processus d'Affaires* supportés par le *Partenaire Commercial* propriétaire de l'instance du *CPP*. Le *CPP* DOIT référencer les *Rôles* dans un *Processus d'Affaires* que l'utilisateur est capable d'exécuter. Par exemple les rôles d'« Acheteur » ou de « Vendeur » dans un *Processus d'Affaires* « Achat ».

Le *CPP* DOIT être capable d'être stocké et restitué par les mécanismes des *Registres* ebXML.

Un *CPP* DEVRAIT également décrire les détails des liens qui sont utilisés pour construire un *En-Tête de Message* ebXML.

#### 8.1.5 Interfaces CPA

Un *CPA* donne des directives sur l'*Interface de Service d'Affaires* utilisée par un *Partenaire Commercial* en vue de contraindre la configuration d'une *Interface de Service d'Affaires* aux valeurs d'un ensemble de paramètres sur lesquelles se sont accordés tous les *Partenaires Commerciaux* qui exécuteront l'agrément.

Les *CPA* ont des *Interfaces* avec les *CPP* puisque le *CPA* est dérivé du *CPP* à la suite d'un processus de négociation permettant de rapprocher les capacités des *Partenaires Commerciaux* (*CPP*) et d'en déduire l'accord sur ce que chaque *Partenaire Commercial* s'engage à faire.

Un *CPA* doit être rattaché à un *Processus d'Affaires* spécifique et référencer les spécifications des interactions nécessaires pour exécuter le *Processus d'Affaires*.

Un *CPA* PEUT être stocké dans un dispositif de *Registre*, ce qui implique l'existence implicite de fonctions d'archivage et de restitution.

### 8.1.6 Détails de Mise en Œuvre (non normatifs) [CPP et CPA]

Un CPA est négocié après la phase de Découverte et d'Extraction. Il donne une vue instantanée de l'information relative au Service de Messagerie et aux Processus d'Affaires que deux ou plus Partenaires Commerciaux ont convenu d'utiliser pour échanger leurs informations d'affaires. Si certains paramètres contenus dans un CPA accepté changent après que l'agrément ait été conclu, un nouveau CPA DEVRAIT être négocié entre les *Partenaires Commerciaux*.

Dans certains cas il peut s'avérer nécessaire ou souhaitable de spécifier des CPA occasionnels, informels ou implicites.

Un des buts éventuels de ebXML est de faciliter la génération automatique des *CPA*. Afin d'atteindre cet objectif, une méthodologie formelle DEVRAIT être spécifiée pour le processus de négociation des *CPA*.

## 8.2 Processus d'Affaires et Modélisation de l'Information

### 8.2.1 Introduction

Le *Méta-modèle d'Information et de Processus d'Affaires ebXML* est un mécanisme qui permet aux *Partenaires Commerciaux* de saisir les détails de leurs scénarios d'affaires en utilisant un modèle méthodologique stable. Un *Processus d'Affaires* décrit de manière détaillée comment des *Partenaires Commerciaux* assument leurs rôles, leurs relations et leurs responsabilités pour faciliter leurs interactions avec d'autres *Partenaires Commerciaux* dans le cadre de collaborations partagées. L'interaction entre les rôles est décrite comme un ensemble chorégraphié de transactions d'affaires. Chaque transaction d'affaires met en jeu un échange électronique de *Documents d'Affaires*. Les *Documents d'Affaires* peuvent être composés d'*Objets d'Information d'Affaires* réutilisables (voir « *Relations avec les Composants Communs* », paragraphe 8.2.3 ci-dessous). Au niveau le plus bas, les *Processus d'Affaires* peuvent être composés de *Processus Communs* réutilisables et les *Objets d'Information d'Affaires* peuvent être composés de *Composants Communs* réutilisables.

Le *Méta-modèle d'Information et de Processus d'Affaires ebXML* supporte différents points de vue pour la description des spécifications des besoins, de l'analyse, et de la conception. A chaque point de vue est associé un ensemble d'éléments sémantiques (vocabulaire) sur lesquels s'appuie la spécification des artefacts utilisés pour faciliter l'intégration de l'information et l'interopérabilité des *Processus d'Affaires*.

Une vue additionnelle vient compléter le Méta-modèle : le *Schéma de Spécification*. Celui-ci est utilisé pour spécifier directement l'ensemble d'éléments nécessaires pour configurer un système exécutable capable d'exécuter un ensemble de transactions d'affaires ebXML. En faisant appel à des éléments de modélisation extraits des autres vues, le Schéma de Spécification forme un sous-ensemble sémantique du *Méta-modèle d'Information et de Processus d'Affaires ebXML*. Le *Schéma de Spécification* est disponible en deux représentations autonomes, un profil UML et un Schéma/DTD XML.

Les relations entre le *Méta-modèle d'Information et de Processus d'Affaires ebXML* et le *Schéma de Spécification ebXML* sont représentées par la Figure 10 ci-dessous. :

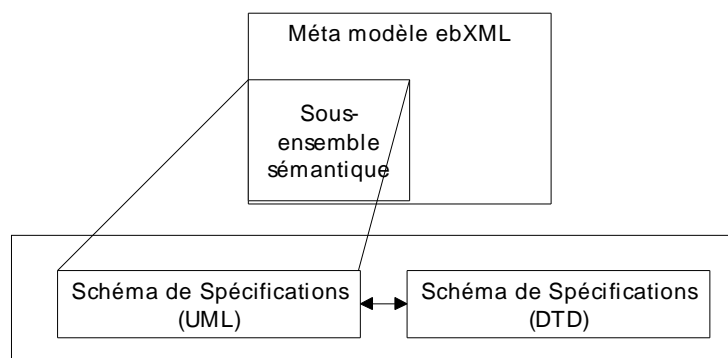


Figure 10.- Méta-modèle ebXML, sous-ensemble sémantique

Supprimé : -

Le *Schéma de Spécification* supporte les spécifications des transactions d'affaires et la chorégraphie de ces transactions dans les *Collaborations d'Affaires* (*Business Collaborations*).  
 Chaque *Transaction d'Affaires* peut être implémentée en utilisant l'un des patterns standards disponibles. Ces patterns spécifient les conditions opérationnelles d'exécution des échanges de *Messages* et de signaux entre des Partenaires Commerciaux en vue d'accomplir la transaction électronique requise. Pour aider à la spécification des patterns, le *Schéma de Spécification* est associé à un ensemble de patterns standards et à un ensemble d'éléments de modélisation commun à ces patterns.  
 La spécification complète d'un *Processus d'Affaires* consiste en un *Méta-modèle de Processus d'Affaires et d'Information* spécifié au moyen du *Schéma de Spécification* associé à l'identification des patterns utilisés. Cette information sert d'entrée initiale pour la formation des *Profils de Protocole de Collaboration* (CPP) et *Agréments sur un Protocole de Collaboration* (CPA), comme le montre la Figure 11 suivante.

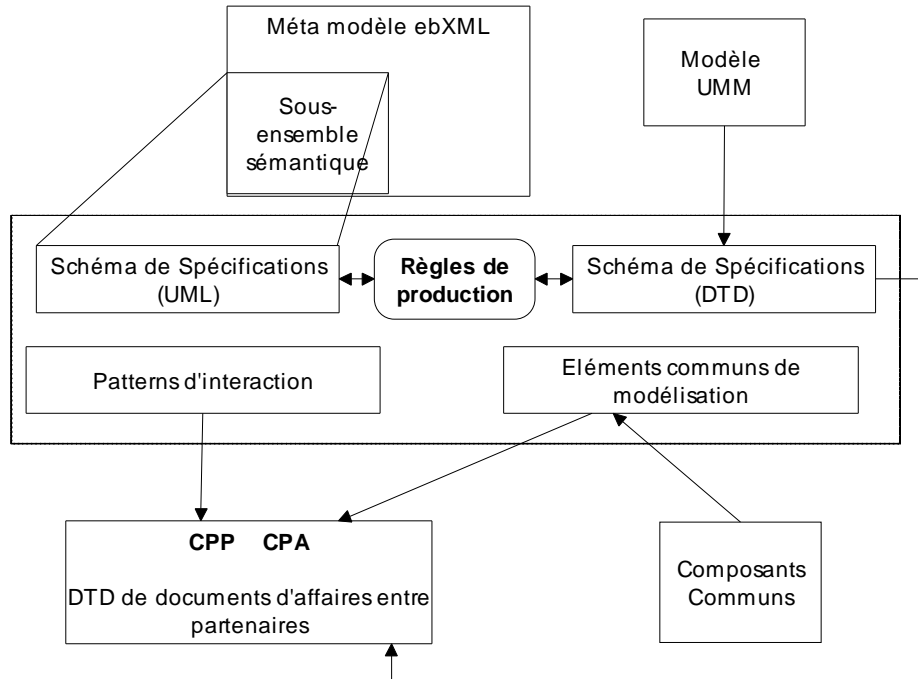


Figure 11- Méta-modèle ebXML

Supprimé : -

Il n'y a pas de recommandation explicite imposant l'utilisation obligatoire d'un langage de modélisation pour composer un nouveau *Processus d'Affaires*. Cependant, si un langage de modélisation est utilisé pour développer des *Processus d'Affaires*, celui-ci DOIT être le *Langage Unique de Modélisation*, (*Unified Modeling Language*, *UML*). Cette exigence garantit qu'il y a qu'une seule méthodologie de modélisation cohérente et stable pour créer de nouveaux *Processus d'Affaires*. Un des principaux avantages à tirer de l'utilisation d'une méthodologie de modélisation unique est qu'il est possible de comparer des modèles et d'éviter la duplication des *Processus d'Affaires* existants.

Plus tard pour faciliter la création de *Processus d'Affaires* et de modèles d'information cohérents, ebXML définira un ensemble de *Processus d'Affaires* partagés parallèlement à la *Librairie Centrale*. Il est possible que les utilisateurs de l'infrastructure ebXML puissent souhaiter étendre cet ensemble ou utiliser leurs propres *Processus d'Affaires*.

## 8.2.2 Description Formelle des Fonctions

La représentation d'une instance de document d'un *Processus d'Affaires* DOIT être formatée de telle manière qu'elle puisse être lue par les hommes comme par les ordinateurs. Cela est nécessaire pour faciliter une transition graduelle vers une automatisation totale des interactions d'affaires.

Un *Processus d'Affaires* DOIT être archivable et restituable par un mécanisme de *Registre*. Les *Processus d'Affaires* PEUVENT être enregistrés dans un *Registre ebXML* pour en faciliter la découverte et la restitution.

Pour être compris par les applications, un *Processus d'Affaires* DOIT pouvoir être exprimé dans la syntaxe XML. Un *Processus d'Affaires* peut être spécifié à partir d'un *Méta-modèle de Processus d'Affaires et d'Information* ou d'une représentation XML de ce modèle. Les *Processus d'Affaires* sont constitués des types d'informations suivantes :

- La chorégraphie des échanges d'instances des documents (par exemple, la chorégraphie des échanges de *Messages* nécessaires entre deux partenaires pour l'exécution d'une transaction ebXML « achat »).
- Les références des *Méta-modèles de Processus d'Affaires et d'Information* ou des *Documents d'Affaires* (qui peuvent être des DTD ou des schémas) qui ajoutent la présentation des structures à celle des données d'affaires.
- La définition des rôles de chaque participant dans le *Processus d'Affaires*

Un *Processus d'Affaires* :

- Fournit les contraintes contextuelles pour l'utilisation des *Composants Communs*
- Fournit un cadre directeur pour l'établissement des CPA
- Spécifie le domaine d'appartenance d'un *Processus d'Affaires*, ainsi que les informations associées aux coordonnées de contact.

[NOTE] : La liste ci-dessus n'est pas limitative.

### 8.2.3 Interfaces

#### Relations avec les CPP et CPA

Une instance d'un CPP pour un *Partenaire Commercial* définit ses capacités fonctionnelles et techniques à supporter zéro, un ou plus *Processus d'Affaires* et un ou plus rôle dans chaque processus.

L'agrément entre deux *Partenaires Commerciaux* définit les conditions opérationnelles que les deux partenaires s'engagent à respecter pour exécuter leurs transactions. L'*Interface* entre le *Processus d'Affaires*, son *Méta-modèle d'Information* et le CPA est définie par le document *Processus d'Affaires*. Celui-ci PEUT être instancié comme un document XML représentant les couches relatives aux transactions et aux collaborations du *Méta-modèle de Processus d'Affaires et d'Information*. L'expression d'une séquence de transactions commerciales en XML est partagée entre le *Processus d'Affaires* et les modèles d'*Information sur le Partenaire Commercial*.

#### Relation avec les Composants Communs

Une instance de *Processus d'Affaires* DEVRAIT spécifier les contraintes pour les échanges de données avec les autres *Partenaires Commerciaux*. L'information d'affaires PEUT être constituée de composants extraits de la *Librairie Centrale ebXML*. Un document de *Processus d'Affaires* DOIT référencer les *Composants Communs* directement ou indirectement en utilisant un document XML qui référence les modèles d'Information et de *Processus d'Affaires* et/ou *Documents d'Affaires* appropriés (qui peuvent être des DTD ou Schéma XML). Le mécanisme pour l'interfaçage avec les *Composants Communs* et la *Librairie Centrale* DOIT utiliser l'identifiant unique associé à chaque composant.

#### Relation avec le Service de Messagerie de ebXML

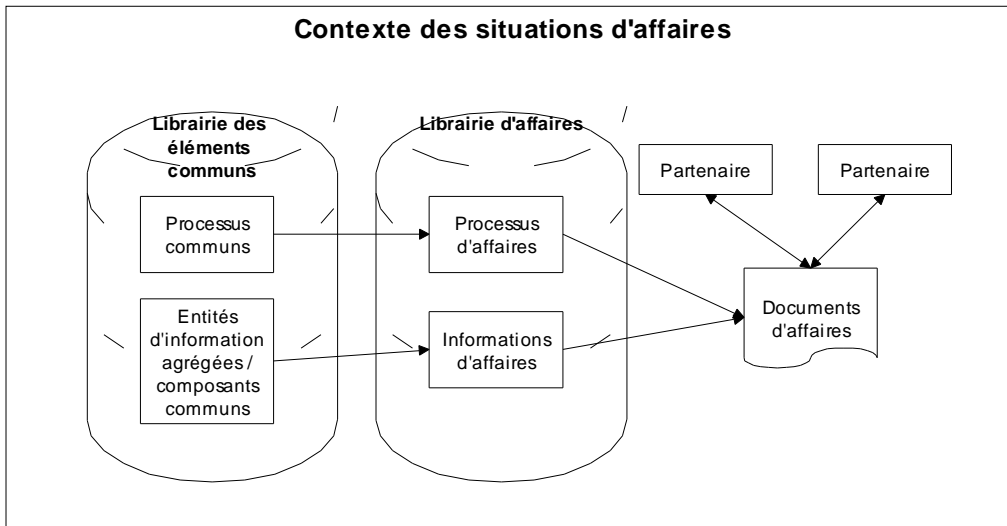
Une instance de *Processus d'Affaires* DOIT pouvoir être transportée d'un *Service de Registre* à un autre *Service de Registre* via un *Message ebXML*. Il DOIT aussi pouvoir être transporté entre un *Service de Registre* et une application utilisatrice via le *Service de Messagerie ebXML*.

#### Relation avec le Système d'Enregistrement

Une instance de *Processus d'Affaires* destinée à être utilisée dans une infrastructure ebXML DOIT pouvoir être retrouvée en adressant une requête à un *Registre*. En conséquence, chaque *Processus d'Affaires* DOIT être associé à un identifiant unique.

### 8.2.4 Détails de Mise en Œuvre (non normatifs)

La composition exacte des *Objets d'Information d'Affaires* ou des *Documents d'Affaires* et des messages est pilotée par un ensemble de catégories de contexte dérivées des *Processus d'Affaires*. Le niveau correspondant de modélisation de l'architecture ebXML est présenté dans la Figure 12 ci dessous.



**Figure 12-** *Processus d’Affaires ebXML, couche Modélisation de l’Information*

Supprimé : -

Un *Méta-modèle de Processus d’Affaires et d’Information ebXML* PEUT être créé en suivant la *Méthodologie de Modélisation UN/CEFACT (UMM)* recommandée, ou PEUT être élaboré en suivant d’autres démarches, pour autant qu’elles soient adaptées à la production de *Méta-modèles de Processus d’Affaires et d’Information* conformes à ebXML.

### 8.3 Composants Communs et Fonctions de la Librairie Centrale

#### 8.3.1 Introduction

Un *Composant Commun* s’empare de l’information sur un concept du monde réel des affaires, ainsi que des relations entre ce concept, d’autres *Objets d’Information d’Affaires* et une description contextuelle qui précise les modalités d’utilisation d’un *Composant Transversal* dans un scénario donné de *Commerce Electronique ebXML*.

Un *Composant Commun* peut être un ensemble cohérent d’information d’affaires (*Objet d’Information d’Affaire*), ou une famille cohérente d’*Objets d’Information d’Affaires* regroupés par affinité (*Entités d’Information Agrégées*).

Le groupe de projet « Composants Communs ebXML » DOIT définir un ensemble initial de Composants Communs ebXML qui pourront être adoptés par les utilisateurs et à partir desquels la Librairie des Composants Communs pourra être progressivement constituée.

#### 8.3.2 Description Formelle des Fonctions

Les *Composants Communs* DOIVENT être conçus pour faciliter l’exécution des fonctions suivantes, ce qui constitue un ensemble minimal de spécifications fonctionnelles :

Les *Composants Communs* DOIVENT être archivés et pouvoir être retrouvés en utilisant un dispositif de *Registre ebXML*.

Les *Composants Communs* DOIVENT capturer et maintenir un ensemble minimal d’information répondant aux besoins du *Commerce Electronique*.

Les *Composants Communs* DOIVENT pouvoir être exprimés dans la syntaxe XML.

Un *Composant Commun* DOIT pouvoir contenir :

- Un autre *Composant Commun* en combinaison avec un ou plusieurs items d’*Objets d’Information d’Affaires*

- D'autres *Composants Communs* en combinaison avec zéro ou plusieurs items d'*Objets d'Information d'Affaires*

Un *Composant Commun* DOIT pouvoir être identifié de manière unique.

### 8.3.3 Interfaces

Un *Composant Commun* PEUT être référencé directement ou indirectement à partir d'une instance de *Document d'Affaires*. Un *Processus d'Affaires* PEUT spécifier un *Composant Commun* ou un groupe de *Composants Communs* comme étant une information obligatoire ou facultative à partir d'une instance de *Document d'Affaires*.

Un *Composant Commun* DOIT être interfacé avec un mécanisme de *Registre* capable de l'archiver et de le restituer à la demande.

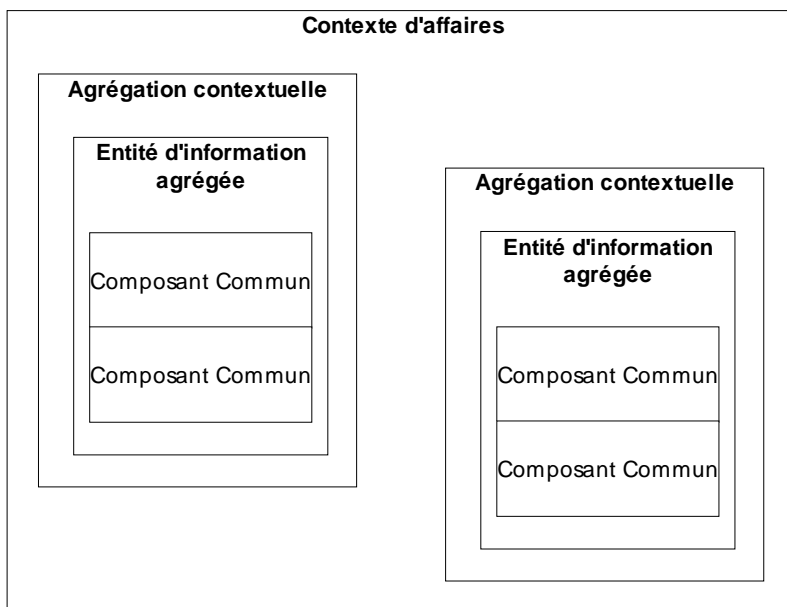
Un *Composant Commun* PEUT être interfacé avec un Élément *XML* provenant d'un autre vocabulaire *XML* à partir du moment où il est unilatéralement ou bilatéralement référencé comme un équivalent sémantique.

### 8.3.4 Détails de Mise en Œuvre (non normatifs)

Un *Composant Commun* PEUT contenir un/des attribut(s) ou être contenu dans un autre *Composant Commun*, en fonction du contexte précis ou de la combinaison de contextes dans lequel il est utilisé.

Le processus d'agrégation des *Composants Communs* pour leur utilisation par un contexte d'affaires spécifique DOIT comprendre un moyen d'identification de la localisation d'un *Composant Commun* dans un autre *Composant Commun*. Ce processus PEUT utiliser une combinatoire de structures de contextes pour faciliter la ré-utilisation des *Composants Communs* à différents niveaux d'imbrication dans un autre *Composant Commun* ou une autre *Entité d'Information Agrégée*. Une combinaison sera désignée sous le nom de *Contexte d'Affaires*.

Le Contexte PEUT également être défini en utilisant le *Méta-modèle de Processus d'Affaires et d'Information ebXML* qui définit les instances des *Objets d'Information d'Affaires* dans lesquelles le *Composant Commun* est impliqué.



**Figure 13-** Contexte d'Affaires, défini en termes de d'Agrégation Contextuelle, d'Entité d'Information Agrégée, et de Composant Commun

Supprimé : -



Les items d'Objets d'Information d'Affaires ou de Composants Communs contenus dans un Composant Commun générique peuvent être facultatifs ou obligatoires. Un Composant Commun placé dans un contexte spécifique ou une combinaison de contextes (agrégation contextuelle ou contexte d'affaires) peuvent voir leur statut de base obligatoire/facultatif modifié.

## 8.4 Fonction du Service de Registre

### 8.4.1 Introduction

Un Registre ebXML fournit un ensemble de services destinés à partager l'information entre des Partenaires Commerciaux. Un Registre est un composant qui maintient une interface d'accès aux méta-données d'un item enregistré. L'accès au Registre ebXML est fourni par des Interfaces (APIs) publiées par les services du Registre.

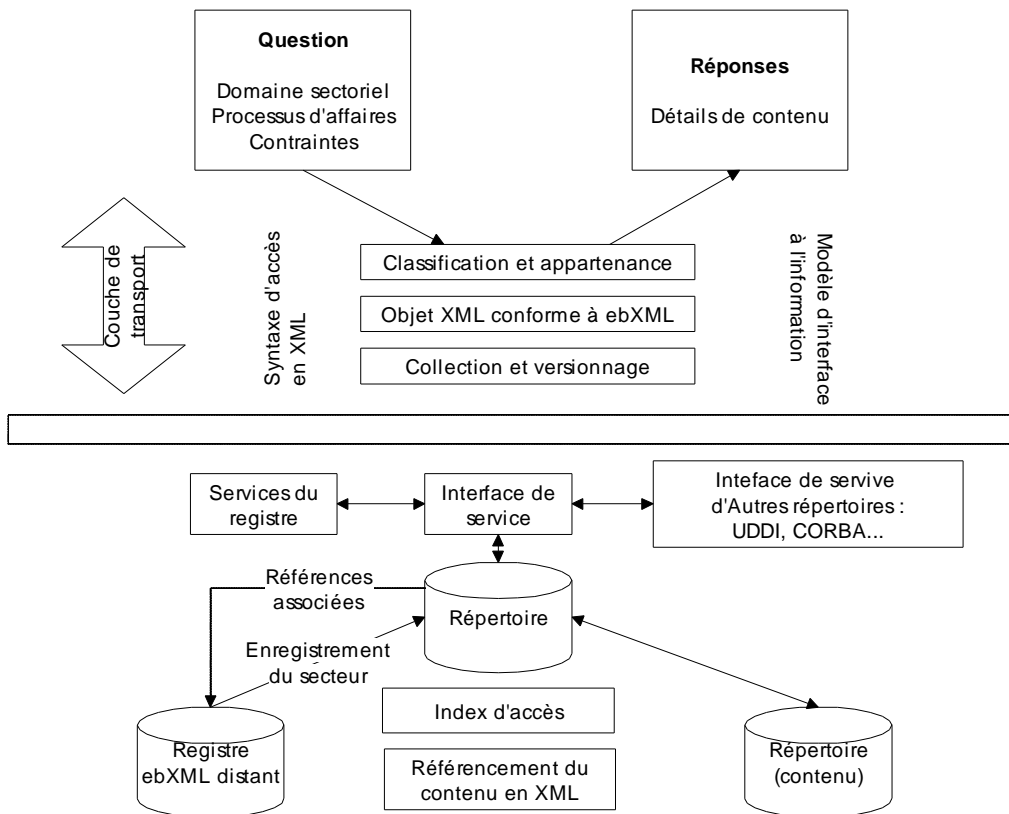


Figure 14.- Architecture globale d'un Registre ebXML

Supprimé : -

### 8.4.2 Description Formelle des Fonctions

Un Registre DOIT être capable de gérer des données utilisant des normes de codage de caractères différentes.

Chaque Enregistrement du Registre, à chaque niveau de granularité tel que défini par l'Organisme Soumissionnaire, DOIT être identifiable de manière unique. Cela est essentiel pour faciliter l'exécution des requêtes adressées par les applications au Registre.

Un Registre DOIT renvoyer zéro ou un résultat positif en réponse à une requête contextuelle portant sur un identifiant et un seul. Dans les cas où plusieurs résultats positifs peuvent correspondre à la requête, un message d'erreur DEVRAIT être rapporté à l'Autorité d'Enregistrement.

Un *Enregistrement du Registre* DOIT être structuré de manière à pouvoir utiliser les relations structurant l'informations pour l'identifier, la nommer, la décrire, indiquer les statuts relatifs à son administration et aux contraintes d'accessibilité, définir sa persistance et sa variabilité, la classer conformément aux schémas de classification prédéfinis, déclarer le type de format des fichiers, et identifier les organismes soumissionnaires et responsables.

L'*Interface du Registre* sert de mécanisme d'accès applications – *Registre*. Les interactions directes homme – *Registre* DOIVENT être traitées comme un composant de l'*Interface du Registre* (à partir par exemple d'un navigateur) et non comme une *Interface* séparée.

L'*Interface du Registre* DOIT être conçue pour être indépendante de la pile des protocoles réseau de niveaux inférieurs (par exemple HTTP/SMTP ou TCP/IP). Des instructions spécifiques sur les modalités de mise en œuvre des interactions avec l'Interface du Registre PEUVENT être contenues dans l'Interchange du Message ebXML.

Les processus supporter par un répertoire ebXML PEUVENT également inclure :

- Un *CPA* spécifique pour la mise en relation entre le *Registre* et les *Registres Clients*
- Un ensemble de processus fonctionnels réglant les relations *Registre* et *Registres Clients*.
- Un ensemble de *Messages d'Affaires* échangés entre *Registre* et *Registres Clients* appartenant à un *Processus d'Affaires* spécifique.
- Un ensemble de mécanismes de primitives d'*Interface* capables de supporter les *Messages d'Affaires* et les mécanismes de mise en œuvre des requêtes et des réponses associées aux *Messages*.
- Un *CPA* spécifique pour orchestrer les interactions entre des *Registres* conformes ebXML.
- Un ensemble de processus fonctionnels réglant les interactions *Registre* à *Registre*.
- Un ensemble de réponses contenant des notifications d'erreur et les instructions de mise en œuvre des actions correctives.

Pour faciliter le processus de découverte, des requêtes et des agents de recherche et d'extraction des informations PEUVENT être utilisées pour la mise en œuvre des interactions homme – *Registre* (par exemple à partir d'un navigateur). Un utilisateur DEVRAIT être capable d'explorer directement et transversalement le contenu en s'appuyant sur les schémas de classification disponibles et qui structurent les accès au *Registre*.

Les *Services de Registre* sont destinés à créer, modifier, et supprimer les *Enregistrements du Registre* et leurs méta-données.

Des protocoles de sécurité appropriés PEUVENT être déployés pour garantir l'authentification des correspondants et la protection des informations du *Répertoire* quand celui-ci est interrogé par le *Registre*.

Des *Identifiants Uniques (UID)* DOIVENT être affectés à tous les items dans un *Système de Registre* ebXML. Des clés *UID* sont des références OBLIGATOIRES pour accès à tout élément de contenu d'un *Registre*. Des Identifiants Uniques Universels (UUID) PEUVENT être utilisés pour garantir que les entrées dans le *Registre* sont effectivement globalement uniques, et donc que, quand un système adresse une requête à un *Registre* basée sur un *UUID*, un résultat et un seul DOIT être retourné.

Pour faciliter la reconnaissance sémantique des *Méta-modèles d'Information* et des *Processus d'Affaires*, le Service de *Registre* DOIT fournir un mécanisme qui lui permette de fournir des descriptions des items du *Registre* lisibles par des hommes. Les *Méta-modèles d'Information* et des *Processus d'Affaires* existants (par exemple les PIPs de RosettaNet) et les *Composants Communs* DOIVENT être associés à des clés *UID* quand ils sont enregistrés par un *Service de Registre* conforme à ebXML. Ces clés *UID* PEUVENT être exprimées dans une syntaxe physique *XML*, et ce de plusieurs façons. Ces mécanismes PEUVENT inclure (la liste suivante n'est pas limitative) :

- Un mécanisme de référencement explicite (exemple : méthode URN:*UID*)
- Une méthode de référencement (exemple : URI :*UID* / namespace :*UID*)
- Une méthode de référencement objet compatible avec les schémas W3C (exemple : URN:NomTypeComplexe)
- Une référence associée à un type de données (exemple : ISO 8601 :2000 type de données Date/Time/Number puis typage de données de l'application existante)

Des composants ebXML DOIVENT facilement supporter le multi-linguisme. Une référence par *UID* est particulièrement appropriée puisque celui-ci est par nature indépendant du langage. Pour supporter le multilinguisme, les spécifications ebXML DOIVENT être conformes avec les normes Unicode et ISO/IEC 10646 pour les jeux de caractère, et UTF8 ou UTF 16 pour l'encodage des caractères.

### 8.4.3 Interfaces

#### **Relations avec la Messagerie ebXML**

La syntaxe de requêtes utilisée par les mécanismes d'accès au Registre doit être indépendante de l'implémentation physique du système de restauration.

Le *Service de Messagerie* ebXML PEUT servir de mécanisme de transport pour toutes les transmissions internes et externes du *Registre*.

#### **Relation avec les Processus d'Affaires**

Les *Processus d'Affaires* sont publiés et retrouvés via les *Services de Registre* ebXML.

#### **Relation avec les Composants Communs**

Les *Composants Communs* sont publiés et retrouvés via les *Services de Registre* ebXML.

#### **Relation Enregistrement avec Méta-données**

Les éléments *XML* constituent des méta-données standard relatives aux items qui sont administrés par le *Service de Registre* ebXML. Puisque les *Registres* ebXML sont distribués, chaque *Registre* PEUT interagir avec un autre *Registre* ebXML et réciproquement.

### 8.4.4 Détails de Mise en Œuvre (non normatifs)

Les *Méta-modèles de Processus d'Affaires et d'Information* dans un *Registre* peuvent être archivés suivant différents schémas de classification.

Les recommandations de la norme ISO11179/3 existante sur le développement des *Registres* PEUVENT être utilisées et fournir un modèle pour le développement des *Registres* ebXML.

Les enregistrements des **Registres** et leurs méta-données PEUVENT être adressables à travers une référence URI basée sur *XML* utilisant seulement HTTP comme protocole support d'accès direct.

Des exemples de fonctions étendues des *Services de Registre* peuvent être proposés pour traitement dans une phase ultérieure de l'initiative ebXML. Ceci inclut de manière non limitative des services de transformation et traduction, des services de workflow, des services d'assurance qualité et des mécanismes de sécurité renforcés.

Un *Service de Registre* PEUT avoir de nombreux modèles de déploiement sous réserve que les *Interfaces du Registre* soient conformes à ebXML.

Le *Méta-modèles de Processus d'Affaires et d'Information* pour un *Service de Registre* ebXML peut être une extension de la spécification existante d'OASIS portant sur les spécifications techniques des services de *Registre/Répertoire*, spécialement conçue pour l'archivage et la recherche/extraction des informations d'affaires, appuyée sur le modèle OASIS qui constitue un sur-ensemble destiné à représenter les processus étendus d'administration d'un contenu d'information ouvert puisque générique.

## 8.5 Fonction du Service de Messagerie

### 8.5.1 Introduction

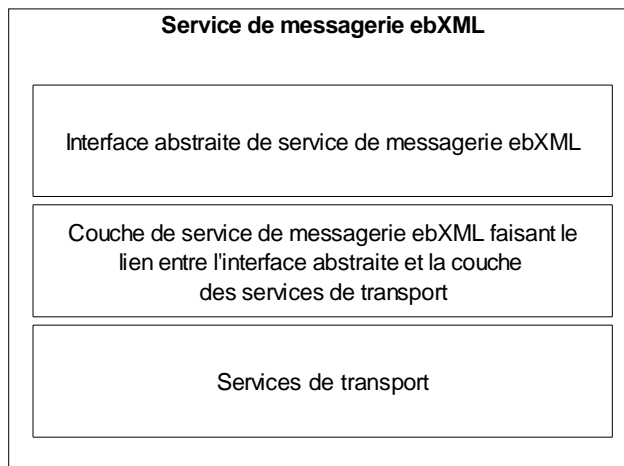
Le mécanisme du *Service de Messagerie* ebXML doit permettre l'échange de *Messages d'Affaires* de manière standardisée entre *Partenaires Commerciaux*. Le *Service de Messagerie* ebXML fournit un moyen fiable pour échanger des *Messages d'Affaires* qui doit pouvoir être implémenté sans faire appel à des solutions et technologies propriétaires. Un Message ebXML contient des structures d'*En-Tête de Message* (nécessaire pour son routage et sa remise) et une section *Conteneur (Payload)*.

Le service de messagerie ebXML est conceptuellement composée de trois parties :

1. un *Service d'Interface* logique
2. les fonctions fournies par le *Niveau Service de Messagerie*,

### 3. la liaison avec les services de transport

Les relations entre ces trois parties peuvent être représentées par la figure 15 ci-dessous:



**Figure 15.- Service de Messagerie ebXML**

Le diagramme suivant donne la structure logique des modules fonctionnels de l'architecture des *Services de Messagerie* ebXML. Ces modules sont disposés de manière à indiquer leurs interrelations et leurs dépendances. Ce diagramme d'architecture illustre la flexibilité du Service de Messagerie ebXML et reflète l'étendue de la gamme de services et de fonctions qui peuvent être implémentées dans un système ebXML.

Supprimé : -

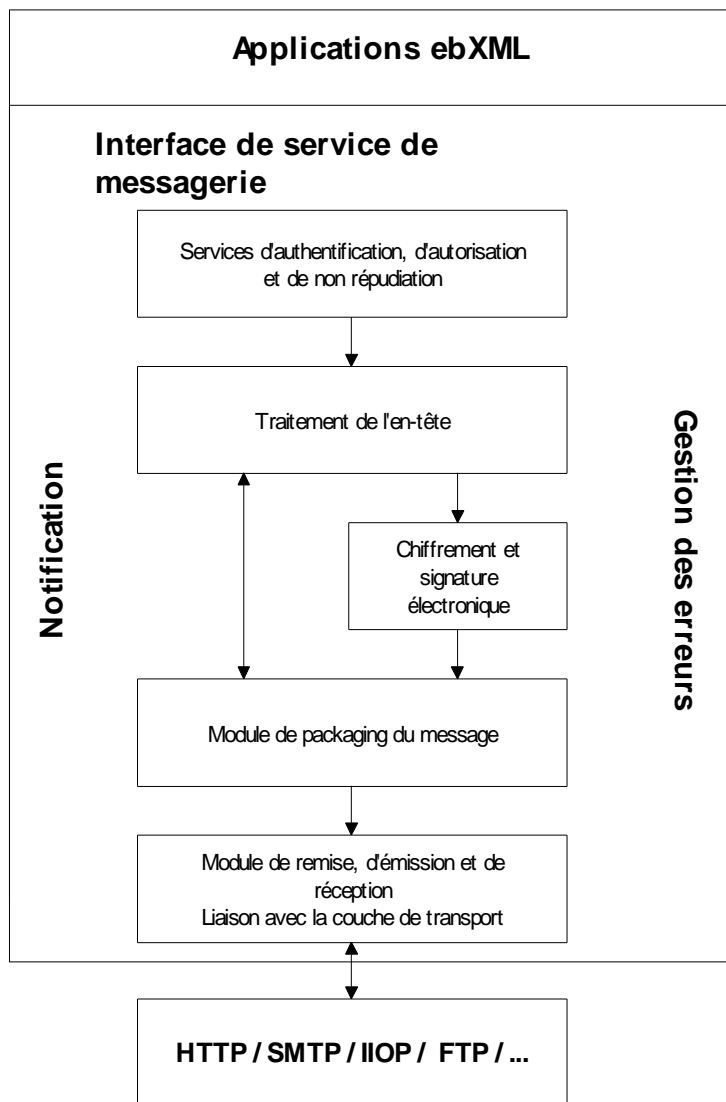


Figure 16- Architecture du Service de Messagerie

Supprimé : -

### 8.5.2 Description formelle des Fonctions

Le *Service de Messagerie* ebXML fournit un mécanisme sécurisé, cohérent et fiable pour l'échange de *Messages* ebXML entre des utilisateurs de l'infrastructure ebXML et un ensemble de Protocoles de transport incluant par exemple SMTP, HTTP/S, FTP, ...).

Le *Service de Messagerie* ebXML prescrit des formats pour tous les *Messages* ebXML échangés entre des *Composants* de l'architecture ebXML distribués, y compris les mécanismes des *Registres*, et les *Applications* des utilisateurs.

Le *Service de Messagerie* ebXML n'impose aucune restriction quant au contenu des *Conteneurs*.

Le *Service de Messagerie* ebXML supporte les échanges de *Messages* unilatéraux (simplex) et bilatéraux (requête/réponse) synchrones et asynchrones.

Le *Service de Messagerie* ebXML supporte le séquençement des *Conteneurs* en instances ou plusieurs *Conteneurs* et/ou *Messages* sont échangés entre des *Partenaires Commerciaux*

Le *Niveau Service de Messagerie* ebXML est capable de traiter les « règles d'engagement » telles que définies par les *Partenaires Commerciaux* dans un *Agrément sur un Protocole de Collaboration* (y compris, mais de manière non limitative, la sécurité et les fonctions des *Processus d'Affaires* relatives à la remise des *Messages*) Ce service de messagerie permet les échanges synchrones et asynchrones. Les messages supportent les conteneurs multiples et les messages multiples entre partenaires. L' *Agrément sur un Protocole de Collaboration* définit les règles de fonctionnement acceptables auquel chaque *Partenaire Commercial* accepte de se conformer. La définition de ces règles de base peut prendre des formes variées parmi lesquelles les *Agréments sur un Protocole de Collaboration* formels, des agréments interactifs établis au moment où la transaction d'affaires est exécutée (par exemple, achat d'un livre en ligne), ou toute autre forme d'agrément. Certaines fonctions du *Niveau Service de Messagerie* exécutent ces règles de base. Une violation quelle qu'elle soit des règles de base doit entraîner une détection de l'erreur et la production d'un message d'erreur au moyen de mécanismes appropriés.

Le *Service de Messagerie* ebXML accomplit les fonctions suivantes relatives à la sécurité :

- Identification
- Authentification (vérification d'identité)
- Autorisation (contrôle d'accès)
- Confidentialité (Chiffrement)
- Intégrité
- Non-répudiation
- Journalisation

### 8.5.3 Interfaces

Le *Service de Messagerie* ebXML dote l'infrastructure ebXML d'une *Interface* abstraite dont les fonctions logiques sont les suivantes :

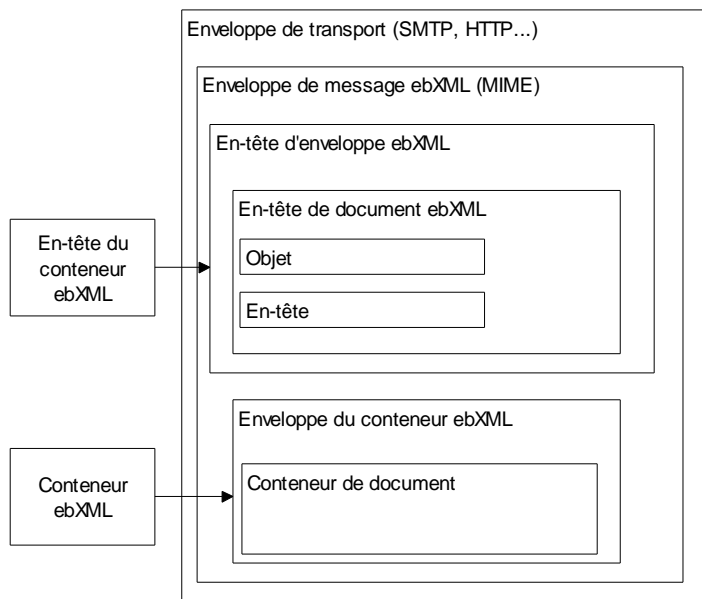
- Send – envoie un message ebXML – dont les valeurs de paramétrage sont extraites des *En-Tête de Messages* ebXML,
- Receive – indique la volonté de recevoir un message ebXML
- Notify – notifie les événements attendus et inattendus
- Send – envoie un message ebXML
- Inquire – fournit une méthode d'interrogation sur le statut d'un interchange particulier de *Message* ebXML

Le *Service de Messagerie* ebXML DOIT comporter une interface avec les systèmes internes, y compris :

- Routage des *Messages* reçus vers les systèmes internes.
- Notification des erreurs

### 8.5.4 Détails de Mise en Œuvre (non normatifs)

La figure 17 illustre la structure logique des *Messages* ebXML :



**Figure 16.-** Architecture du Service de Messagerie

Supprimé : -

Un message ebXML est constitué d'un *Protocole* de transport spécifique et optionnel indépendant de l'*Enveloppe du Protocole de Communication* et d'un *Protocole* ebXML indépendant d'*Enveloppe de Message*. Les groupages de l'*Enveloppe de Message* ebXML sont formés en utilisant les types de contenus prévus dans les attachements de MIME. MIME est utilisé comme solution de groupage car il permet d'échanger une information de nature variée entre les environnements de *Partenaires Commerciaux*. Par exemple, une *Transaction d'Affaires* complexe pourrait nécessiter la constitution d'un *Conteneur* qui contienne une collection de documents d'affaires (*XML* ou autres formats de document), des images, et/ou d'autres Informations d'Affaires associées.

## 9 Conformité

### 9.1 Introduction

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration (Collaboration Protocol Profil : CPP)*. Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.

### 9.2 Conformité à ebXML

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration (Collaboration Protocol Profil : CPP)*. Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.

### 9.3 Conformité à la Spécification d'Architecture Technique

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration (Collaboration Protocol Profil : CPP)*. Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.

### 9.4 Schéma Directeur des Tests de Conformité

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration (Collaboration Protocol Profil : CPP)*. Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.



## 10 Considérations relatives à la Sécurité

### 10.1 Introduction

Pour faciliter le processus de mise en œuvre du *Commerce Electronique*, les *Partenaires Commerciaux* potentiels ont besoin d'un mécanisme permettant de publier les informations au sujet des *Processus d'Affaires* qu'ils supportent avec les modalités détaillées de mise en œuvre de technologies spécifiques concernant leurs capacités d'échange d'informations d'affaires. Cela est réalisé par l'utilisation d'un *Profil de Protocole de Collaboration* (*Collaboration Protocol Profil : CPP*). Le CPP est un document qui permet à un *Partenaire Commercial* d'exprimer les *Processus d'Affaires* qu'il supporte et les spécifications des de son *Interface de Service d'Affaires* de manière à ce qu'ils puissent être compris de manière universelle par tous les autres *Partenaires Commerciaux* utilisant des systèmes conformes à ebXML.

### Avertissement

Les points de vue et spécifications exprimées dans ce document sont celles de leurs auteurs et peuvent ne pas être celles de leurs employeurs. La responsabilité des auteurs et de leurs employeurs ne peut être engagée à la suite de toute réclamation portant sur des problèmes consécutifs à une utilisation et/ou une implémentation correcte ou incorrecte des concepts et/ou des clauses de ces spécifications.

### Copyright Statement

Copyright © ebXML 2001. All Rights Reserved.

This document and translations of it MAY be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation MAY be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself MAY not be modified in any way, such as by removing the copyright notice or references to ebXML, UN/CEFACT, or OASIS, except as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by ebXML or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and ebXML DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.