

TCVN ISO/TS 15000-1 : 2007

ISO/TS 15000-1 : 2004

Xuất bản lần 1

**NGÔN NGỮ ĐÁNH DẤU MỞ RỘNG KINH DOANH ĐIỆN TỬ (ebXML)
PHẦN 1: QUY ĐỊNH KỸ THUẬT VỀ HỒ SƠ VÀ THỎA THUẬN
GIAO THỨC HỢP TÁC (ebCPP)**

Electronic business eXtensible Markup Language (ebXML)

Part 1: Collaboration-protocol profile and agreement specification (ebCPP)

Mục lục	Trang
Lời nói đầu	5
1 Phạm vi áp dụng	7
2 Ban kỹ thuật tiêu chuẩn quốc tế của OASIS	7
2.1 Các bên tham gia xây dựng ebXML	8
3 Giới thiệu	8
3.1 Khái quát nội dung tiêu chuẩn	8
3.2 Các quy ước sử dụng trong tiêu chuẩn	9
3.3 Phiên bản tài liệu ebXML	10
3.4 Định nghĩa	10
3.5 Độc giả	10
3.6 Giả định	11
3.7 Các tài liệu liên quan	11
4 Mục tiêu thiết kế	11
5 Tổng quan hệ thống	12
5.1 Tiêu chuẩn này thực hiện	12
5.2 Tạo một CPA từ hai CPP	14
5.3 Tạo một CPA từ mẫu CPA	16
5.4 Phương thức làm việc của CPA	16
5.5 CPA có thể được thực hiện	17
6 Định nghĩa của CPP	18
6.1 Định danh toàn cầu-duy nhất của tài liệu CPP cụ thể	19
6.2 Cấu trúc CPP	19
6.3 Phần tử CollaborationProtocolProfile (hồ sơ giao thức hợp tác)	19
6.4 Phần tử PartyInfo (Thông tin bên tham gia)	20
6.5 Phần tử SimplePart (Thành phần đơn giản)	68
6.6 Phần tử Packaging (Đóng gói)	69
6.7 Phần tử Signature (Chữ ký)	72
6.8 Phần tử Comment (Chú giải)	73
7 Định nghĩa CPA	73
7.1 Cấu trúc CPA	74
7.2 Phần tử CollaborationProtocolAgreement (Thỏa thuận giao thức hợp tác)	74
7.3 Phần tử Status (Trạng thái)	75
7.4 Khoảng thời gian của CPA	76
7.5 Phần tử ConversationConstraints (Quy định hội thoại)	77
7.6 Phần tử PartyInfo (Thông tin bên tham gia)	78
7.7 Phần tử SimplePart (Thành phần đơn giản)	78
7.8 Phần tử Packaging (Đóng gói)	79
7.9 Phần tử Signature (Chữ ký)	79
7.10 Phần tử Comment (Chú giải)	81
7.11 Soạn một CPA từ hai CPP	81
7.12 Sửa đổi các tham số tài liệu quy định quá trình dựa trên cơ sở thông tin trong CPA	82
8 Tài liệu tham khảo	83
9 Sự phù hợp	84
10 Thông tin về điểm liên lạc	85
Phụ lục A	87
Phụ lục B	100
Phụ lục C	113
Phụ lục D	115
Phụ lục E	123
E.1 Các đề xuất đối với việc thiết kế các thủ tục tính toán	123
E.2 Các phần nhiệm vụ hình thành CPA	125
Phụ lục F	147
Phụ lục G	150

Lời nói đầu

TCVN ISO/TS 15000-1 : 2007 hoàn toàn tương đương với tiêu chuẩn **ISO/TS 15000-1 : 2004**.

TCVN ISO/TS 15000-1 : 2007 do Ban kỹ thuật tiêu chuẩn TCVN/TC 154 "*Quá trình, các yếu tố dữ liệu và tài liệu trong thương mại, công nghiệp và hành chính*" biên soạn, Tổng cục Tiêu chuẩn Đo lường Chất lượng đề nghị, Bộ Khoa học và Công nghệ công bố.

Ngôn ngữ đánh dấu mở rộng kinh doanh điện tử (ebXML)

Phần 1: Quy định kỹ thuật về hồ sơ và thỏa thuận giao thức hợp tác (ebCPP)

Electronic business eXtensible Markup Language (ebXML)

Part 1: Collaboration-protocol profile and agreement specification (ebCPP)

1 Phạm vi áp dụng

Tiêu chuẩn này là một QUY ĐỊNH ebXML cho cộng đồng doanh nghiệp điện tử (eBusiness). Định dạng tiêu chuẩn này dựa trên dạng thức tiêu chuẩn RFC của Internet Society (cộng đồng người sử dụng Internet).

2 Ban kỹ thuật tiêu chuẩn quốc tế của OASIS

Selim Aissi	Intel
Arvola Chan	TIBCO
James Bryce Clark	Individual Member
David Fischer	Drummond Group
Tony Fletcher	Individual Member
Brian Hayes	Collaborative Domain
Neelakantan Kartha	Sterling Commerce
Kevin Liu	SAP
Pallavi Malu	Intel
Dale Moberg	Cyclone Commerce
Himagiri Mukkamala	Sybase
Peter Ogden	Cyclone Commerce
Marty Sachs	IBM
Yukinori Saito	Individual Member
David Smiley	Mercator
Tony Weida	Individual Member
Pete Wenzel	SeeBeyond
Jean Zheng	Vitria

2.1 Các bên tham gia xây dựng ebXML

Các tác giả trên ghi nhận việc tham gia đáng kể trong việc xây dựng tiêu chuẩn này (phiên bản 1.0) của các bên tham gia sau đây:

David Burdett, CommerceOne

Tim Chiou, United World Chinese Commercial Bank

Chris Ferris, Sun

Scott Hinkelman, IBM

Maryann Hondo, IBM

Sam Hunting, ECOM XML

John Ibbotson, IBM

Kenji Itoh, JASTPRO

Ravi Kacker, eXcelon Corp.

Thomas Limanek, iPlanet

Daniel Ling, VCHEQ

Henry Lowe, OMG

Dale Moberg, Cyclone Commerce

Duane Nickull, XML Global Technologies

Stefano Pogliani, Sun

Rebecca Reed, Mercator

Karsten Riemer, Sun

Marty Sachs, IBM

Yukinori Saito, ECOM

Tony Weida, Edifecs

3 Giới thiệu

3.1 Khái quát nội dung tiêu chuẩn

Như đã được định nghĩa trong giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], một *bên tham gia kinh doanh* là một thực thể tham gia vào các *giao dịch kinh doanh* cùng với (các) *bên tham gia kinh doanh* khác. Các khả năng trao đổi *thông điệp* của một *bên tham gia* CÓ THỂ được mô tả trong phần *hồ sơ giao thức hợp tác (Collaboration Protocol Profile) (CPP)*. thỏa thuận trao đổi *thông điệp* giữa hai *bên tham gia* CÓ THỂ được mô tả trong *thỏa thuận giao thức hợp tác (Collaboration Protocol Agreement) (CPA)*. Một *CPA* có thể được tạo ra thông qua việc tính toán phần chung của các *CPP* của hai *bên tham gia*.

Các chi tiết về truyền tải, truyền *thông điệp*, quy định an ninh và các ràng buộc đối với một tài liệu về *quy định quá trình kinh doanh* (hoặc viết tắt là *quy định quá trình*) được chứa trong *CPP* và *CPA* đó, bao gồm định nghĩa các phần chung giữa hai *bên tham gia* vào một *hồ sơ hợp tác kinh doanh* điện tử đã xác định.

Tiêu chuẩn này bao gồm các định nghĩa được chi tiết trong *hồ sơ giao thức hợp tác (Collaboration Protocol Profile) (CPP)* và *thỏa thuận giao thức hợp tác (Collaboration Protocol Agreement) (CPA)*.

Tiêu chuẩn này là một phần trong bộ tiêu chuẩn về ebXML.

Phần còn lại trong tiêu chuẩn này được tổ chức như sau:

- Phần 6 xác định các mục tiêu, đối tượng của tiêu chuẩn này;
- Phần 7 đưa ra một tổng quan về hệ thống;
- Phần 8 bao gồm định nghĩa *CPP*, xác định cấu trúc của *CPP* và toàn bộ các trường cần thiết;
- Phần 9 bao gồm định nghĩa *CPA*;
- Phần 10 liệt kê tất cả các tài liệu được tham chiếu trong tiêu chuẩn này;
- Phần 11 đưa ra một khẳng định về sự phù hợp;
- Phần 12 bao gồm phần chưa được khai báo;
- Phần 13 liệt kê thông tin liên hệ về các tác giả đóng góp và người hợp tác soạn thảo tiêu chuẩn này;
- Các phụ lục bao gồm các ví dụ trong các tài liệu *CPP* và *CPA* (không quy định), một ví dụ về *quy định quá trình kinh doanh* của XML (không quy định), một tài liệu về giản đồ XML (quy định), mô tả cách thức soạn thảo một *CPA* từ hai *CPP* (không quy định), một tổng quan về các tham số của *CPP* và dịch vụ *thông điệp* của ebXML tương ứng (quy định) và bảng chú giải các thuật ngữ.

3.2 Các quy ước sử dụng trong tiêu chuẩn

Các thuật ngữ dưới dạng chữ nghiêng được định nghĩa trong phụ lục G (Bảng chú giải các thuật ngữ). Các thuật ngữ dưới dạng chữ nghiêng đậm trình bày nội dung phần tử và/hoặc thuộc tính của XML *CPP*, *CPA* hoặc các định nghĩa liên quan.

Trong tiêu chuẩn này, các đoạn được bắt đầu bằng "CHÚ THÍCH:" đưa ra các giải thích hoặc đề nghị tham khảo không bắt buộc trong tiêu chuẩn này.

Các tham chiếu tới các tài liệu bên ngoài được trình bày trong các khối văn bản được đóng trong dấu ngoặc vuông, ví dụ; [RFC2396]. Các tham chiếu này được liệt kê trong phần 10, "Tham chiếu".

Các từ khoá BẮT BUỘC, KHÔNG BẮT BUỘC, YÊU CẦU, PHẢI, KHÔNG PHẢI, NÊN, KHÔNG NÊN, KHUYẾN CÁO, CÓ THỂ và TỰY CHỌN, khi xuất hiện trong tiêu chuẩn này được dịch như đã mô tả trong [RFC 2119].

TCVN ISO/TS 15000-1 : 2007

CHÚ THÍCH: Người sử dụng NÊN xem xét cẩn thận việc hỗ trợ các phần tử trong các tập hợp (0 hoặc 1) hoặc (0 hoặc nhiều). Việc hỗ trợ một phần tử như vậy có nghĩa là phần tử đó được xử lý thích hợp đối với chức năng đã xác định của nó và không chỉ là được công nhận và bỏ qua. Một *bên tham gia* cho trước có thể sử dụng các phần tử này trong một số *CPP* hoặc *CPA* và không sử dụng trong các *CPP* hoặc *CPA* khác. Một số phần tử trong các phần tử này xác định các phương thức hoạt động hoặc các tham số NÊN được thi hành với tất cả người sử dụng. Điều thích hợp là thi hành các phần tử lựa chọn để trình bày các chức năng trong thời gian thực chính, như là các chức năng an ninh hoặc thủ tục truyền thông khác nhau, bằng các phương tiện plug-ins. Vì vậy, một *bên tham gia* cho trước CÓ THỂ chỉ yêu cầu các chức năng cần thiết hơn là cài đặt tất cả các chức năng.

Theo quy ước, các giá trị của các thuộc tính [XML] nói chung được đính kèm trong các nhãn trích dẫn, tuy nhiên, các nhãn trích dẫn đó không phải là một phần của chính các giá trị của chúng.

3.3 Phiên bản tài liệu ebXML

Ngay khi tiêu chuẩn được sửa đổi, tiêu chuẩn PHẢI được đánh số phiên bản mới.

Được dự liệu trước trong khoảng thời gian soát xét, các lỗi và mâu thuẫn trong tiêu chuẩn này có thể được phát hiện và chỉnh cho đúng. Tất cả các lỗi được nhận biết trong tiêu chuẩn này cũng như các thay đổi cần thiết đối với các giản đồ phải được kết luận tại:

http://www.oasis-open.org/committees/ebxml-cppa/documents/ebCPP-2_0-Errata.shtml

Tiêu chuẩn này khi được xây dựng được thông qua bởi Ban kỹ thuật về Thỏa thuận và hồ sơ hồ sơ hợp tác để soát xét công khai, PHẢI được đánh số hiệu phiên bản "2_0". Tại thời điểm đó, giản đồ này PHẢI có số hiệu phiên bản là "2_0b" và chữ cái sau "2_0" sẽ được tăng lên khi lỗi sửa giản đồ đó được đưa ra. Các phiên bản như vậy của giản đồ đó PHẢI được tìm thấy tại danh mục:

<http://www.oasis-open.org/committees/ebxml-cppa/schema/>

Ngoài ra, phiên bản mới nhất của giản đồ luôn PHẢI được đưa ra tại:

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd

Do đường dẫn sau là tên miền URI được sử dụng cho tiêu chuẩn này và giản đồ tương ứng được hỗ trợ để có thể giải quyết một cách trực tiếp từ tên miền URI đó.

Giá trị thuộc tính **version** (phiên bản) của phần tử **Schema** (giản đồ) trong một phiên bản cho trước của giản đồ đó PHẢI bằng số phiên bản của giản đồ đó.

3.4 Định nghĩa

Các thuật ngữ kỹ thuật trong tiêu chuẩn này được định nghĩa trong phụ lục G.

3.5 Độc giả

Độc giả của tiêu chuẩn này là những người thực thi các dịch vụ ebXML, người thiết kế khác và người phát triển phần mềm ứng dụng và phần sụn được sử dụng để tạo ra *kinh doanh* điện tử. Độc giả của tiêu chuẩn này cũng là những người trong mỗi tổ chức doanh nghiệp có trách nhiệm tạo ra các *CPP* và *CPA*.

3.6 Giả định

Tiêu chuẩn này mong muốn người đọc hiểu biết về XML và biết rõ các khái niệm về *kinh doanh* điện tử (eBusiness).

3.7 Các tài liệu liên quan

Các tài liệu liên quan bao gồm các quy định ebXML trong các chủ đề sau:

- quy định kỹ thuật về dịch vụ thông điệp ebXML [ebMS];
- giản đồ và quy định kỹ thuật về *quá trình kinh doanh* ebXML [ebBPSS];
- khái quát về thành phần chính ebXML [ccOVER];
- quy định kỹ thuật về dịch vụ đăng ký ebXML [ebRS];

Danh mục đầy đủ các tham chiếu xem phần 10.

4 Mục tiêu thiết kế

Mục tiêu của tiêu chuẩn này là đảm bảo khả năng hoạt động tương tác giữa hai *bên tham gia* thậm chí hai *bên tham gia* này CÓ THỂ sử dụng phần mềm ứng dụng và phần mềm hỗ trợ thời gian thực của các nhà cung cấp khác nhau. CPP xác định các khả năng trao đổi *thông điệp* của một *bên tham gia* và các quá trình hợp tác *kinh doanh* mà nó hỗ trợ. CPA xác định phương thức hai *bên tham gia* sẽ giao tiếp trong việc thực thi các quá trình hợp tác *kinh doanh* đã chọn. Cả hai *bên tham gia* PHẢI sử dụng các bản sao đồng dạng của CPA đó để lập cấu hình hệ thống thời gian thực của họ. Điều này đảm bảo rằng họ lập cấu hình một cách tương thích để trao đổi *thông điệp* dù họ có duy trì các hệ thống thời gian thực của họ từ cùng một nhà cung cấp hay không. Quá trình tạo cấu hình này CÓ THỂ được tự động bằng các công cụ phù hợp để đọc CPA đó và thực hiện quá trình tạo cấu hình này.

Ngoài việc hỗ trợ giao tiếp trực tiếp giữa hai *bên tham gia*, tiêu chuẩn này cũng CÓ THỂ được sử dụng để hỗ trợ giao tiếp giữa hai *bên tham gia* thông qua một bên trung gian như bưu điện hoặc người môi giới.

Một mục đích nữa của tiêu chuẩn này là PHẢI có khả năng soạn một CPA thông qua các phần chung của các CPP tương ứng của các *bên tham gia* liên quan. CPA Kết quả PHẢI chỉ bao gồm các phần tử chung đó hoặc có thể tương thích, giữa hai *bên tham gia*. Số lượng các biến, như số lần thử lỗi, sau đó được thương lượng giữa hai *bên tham gia*. Việc thiết kế các giản đồ CPP và CPA tạo thuận lợi cho quá trình thương lượng/tạo kết cấu (composition/negotiation) này. Tuy nhiên, các quá trình thương lượng và tạo kết cấu đó tự chúng nằm ngoài phạm vi của tiêu chuẩn này. Phụ lục E (tham khảo) đề cập đến vấn đề này.

Mục đích sâu hơn của tiêu chuẩn này để tạo thuận lợi cho việc chuyển dịch các ứng dụng dựa trên cơ sở EDI truyền thống và ứng dụng mang tính kế thừa khác sang các ứng dụng dựa trên cơ sở các quy định ebXML.

Cụ thể, *CPP* và *CPA* là các thành phần của việc chuyển đổi các ứng dụng dựa trên cơ sở X12 838 Trading-Partner Profile[X12] sang phương pháp tự động hóa hơn để thiết lập các mối quan hệ *kinh doanh* và tiến hành *kinh doanh* trong chúng.

5 Tổng quan hệ thống

5.1 Tiêu chuẩn này thực hiện

Việc trao đổi thông tin giữa hai *bên tham gia* đòi hỏi mỗi *bên tham gia* phải hiểu *hồ sơ hợp tác kinh doanh* của *bên tham gia* kia, vai trò của *bên tham gia* kia trong *hồ sơ hợp tác kinh doanh* và chi tiết công nghệ về cách thức mà *bên tham gia* kia gửi và nhận các *thông điệp*. Trong một số trường hợp, điều cần thiết đối với hai *bên tham gia* đó để đạt được thỏa thuận trong một số chi tiết đó.

Cách thức mà mỗi *bên tham gia* có thể trao đổi thông tin, theo nội dung của một thủ tục quá trình hợp tác *kinh doanh* có thể được mô tả bằng một *hồ sơ giao thức hợp tác (Collaboration Protocol Profile) (CPP)*. Thỏa thuận giữa các *bên tham gia* này có thể coi như một thỏa thuận về giao thức *hồ sơ hợp tác (CPA)*.

Một *bên tham gia* CÓ THỂ tự mô tả trong một *CPP* đơn. *Bên tham gia* CÓ THỂ tạo nhiều *CPP* để mô tả, ví dụ, các *hồ sơ hợp tác kinh doanh* khác nhau mà nó hỗ trợ, các hoạt động của nó trong các vùng khác nhau trên thế giới hoặc các bộ phận khác nhau trong tổ chức của *bên tham gia* đó.

Để đảm bảo các *bên tham gia* mong muốn tiến hành *kinh doanh* phù hợp với các *bên tham gia kinh doanh* khác, các *CPP* CÓ THỂ được lưu trữ trong một kho như được cung cấp bởi sổ đăng ký ebXML [ebRS]. Việc sử dụng quá trình khôi phục được cung cấp như một phần của tiêu chuẩn này của một kho, một *bên tham gia* CÓ THỂ sau đó sử dụng các phương tiện kho để tìm kiếm các *bên tham gia kinh doanh*.

Tiêu chuẩn này định nghĩa các giao dịch giữa hai *bên tham gia* là một tài liệu về *quy định-quá trình* CÓ THỂ phù hợp với giản đồ *quy định quá trình kinh doanh* [ebBPSS]. *CPP* và *CPA* bao gồm các tham chiếu tới tài liệu *quy định quá trình*. Tài liệu *quy định quá trình* CÓ THỂ được lưu trữ trong một kho như là sổ đăng ký ebXML. Xem phần chú thích về các mô tả *hồ sơ hợp tác kinh doanh* trong phần 8.4.4.

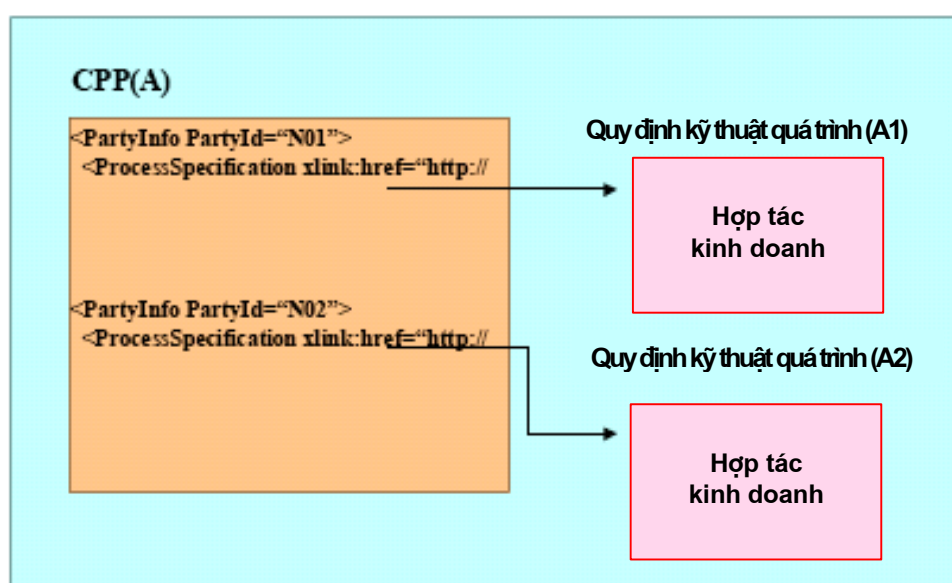
Hình 1 minh họa mối quan hệ giữa một *CPP* và hai tài liệu *quy định quá trình*, A1 và A2, trong một sổ đăng ký ebXML. Bên trái là một *CPP* bao gồm thông tin về hai công ty đại diện như các *bên tham gia* khác nhau. Bên phải chỉ ra hai tài liệu *quy định quá trình*. Mỗi phần tử **PartyInfo** (thông tin bên tham gia) trong *CPP* này bao gồm một tham chiếu tới một trong các tài liệu *quy định quá trình*. Điều này xác định các *hồ sơ hợp tác kinh doanh* mà *bên tham gia* đó có thể thực hiện.

Tiêu chuẩn này định nghĩa từ vựng đối với việc tạo các *CPP* và *CPA* điện tử. Các *CPP* và *CPA* là các tài liệu [XML]. Trong các phụ lục của tiêu chuẩn này là hai ví dụ minh họa một *CPP*, một ví dụ *CPA* được tạo ra từ các *CPP* đó, một ví dụ *quy định quá trình* được tham chiếu bởi các *CPP* và *CPA* đó và giản đồ XML áp đặt các cấu trúc của các *CPP* và *CPA*.

CPP mô tả khả năng của một *bên tham gia* riêng. CPA mô tả các khả năng để hai *bên tham gia* đã thỏa thuận sử dụng để thực hiện các *hồ sơ hợp tác kinh doanh* cụ thể. Các CPA này định nghĩa "các thuật ngữ và nội dung công nghệ thông tin" để đảm bảo rằng các tài liệu *kinh doanh* được trao đổi dưới dạng điện tử giữa các *bên tham gia*. Nội dung thông tin của một CPA tương tự các quy định công nghệ thông tin đôi khi được bao gồm trong trao đổi dữ liệu điện tử (EDI)

Các thỏa thuận của *bên tham gia* thương mại (TPA). Tuy nhiên, các CPA này không phải là các tài liệu bản giấy. Hơn nữa, chúng là các tài liệu điện tử để có thể được xử lý bằng máy tính tại vị trí của các *bên tham gia* để thiết lập và sau đó thực thi các trao đổi thông tin *kinh doanh* mong muốn. Các thuật ngữ và hoàn cảnh "hợp pháp" của một Thỏa thuận *kinh doanh* nằm ngoài phạm vi của tiêu chuẩn này và vì vậy không được bao gồm trong CPP và CPA.

Kho



Hình 1 - Cấu trúc của CPP và quy định *quá trình kinh doanh* trong một sổ đăng ký ebXML

Một tổ chức *kinh doanh* CÓ THỂ lựa chọn để tự đại diện như nhiều *bên tham gia*. Ví dụ, nó có thể biểu diễn một văn phòng trung tâm của tổ chức buôn bán và cơ sở sản xuất của tổ chức buôn bán như các *bên tham gia* riêng. Tổ chức *kinh doanh* đó sau đó CÓ THỂ xây dựng một CPP bao gồm tất cả các đơn vị của nó được biểu diễn như các *bên tham gia* riêng biệt. Trong CPP đó, mỗi một đơn vị trong các đơn vị đó sẽ được biểu diễn bởi một phần tử **PartyInfo** (thông tin bên tham gia) riêng biệt.

Quy định CPA liên quan đến phần mềm tạo ra công việc *kinh doanh* đại diện cho các *bên tham gia* bằng việc trao đổi các *thông điệp* [ebMS]. Cụ thể, nó tập trung vào các chương trình bên Server và Client phù hợp với các *giao dịch kinh doanh* [ebBPSS] bằng việc gửi và nhận các *thông điệp*.

Các *thông điệp* đó chuyển các tài liệu *kinh doanh* và/hoặc các tín hiệu *kinh doanh* đã được chấp nhận [ebBPSS] trong thiết bị mang của nó. Dưới các điều kiện của một CPA:

- một bên Client khởi tạo một kết nối với bên Server;
- một bên *yêu cầu* khởi tạo một *giao dịch kinh doanh* với một bên *phản hồi*;
- một bên *gửi* gửi một *thông điệp* đến một bên *nhận*.

TCVN ISO/TS 15000-1 : 2007

Vì vậy, bên Client và bên Server là các bản sao phần mềm, bên yêu cầu và bên phản hồi là các bên tương ứng có cùng chức năng *kinh doanh* và bên gửi và bên nhận là các bên tạo *thông điệp* tương ứng nhau. Không có mối quan hệ cố định giữa các bên tương ứng cùng chức năng của các kiểu khác nhau. Ví dụ, xem xét một hợp tác mua sắm. Bên *Client* đại diện *bên tham gia* mua có thể kết nối tới bên *Server* đại diện *bên tham gia* bán và sau đó tạo ra một yêu cầu mua sắm bằng việc gửi một *thông điệp* chứa một đơn đặt hàng mua sắm trên kết nối đó. Nếu *CPA* đó quy định một phản hồi *kinh doanh* đồng thời, bên *Server* sau đó có thể phản hồi lại bằng việc gửi một *thông điệp* chứa một thông báo chấp nhận trở lại bên *Client* trên cùng kết nối đó. Nói cách khác, nếu *CPA* đó quy định một phản hồi *kinh doanh* đồng thời, bên *Client* đại diện *bên tham gia* bán có thể sau đó phản hồi bởi kết nối tới *bên Server* đại diện *bên tham gia* mua và sau đó gửi một *thông điệp* chứa một thông báo chấp nhận.

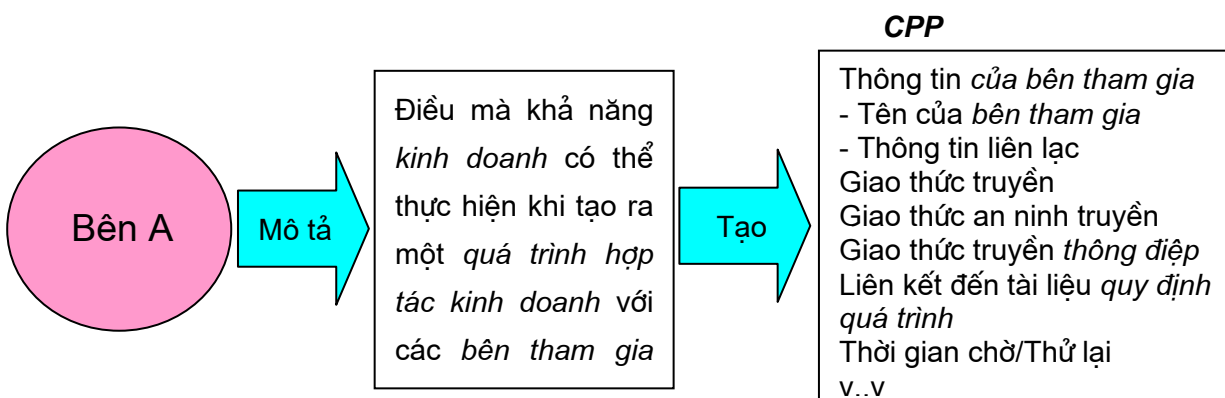
Nói chung, các *bên tham gia* tới một *CPA* có thể có cả hai đặc điểm Client (khách) và Server (chủ). Một bên *Client* yêu cầu các dịch vụ và một bên *server* cung cấp các dịch vụ tới *bên tham gia* yêu cầu các dịch vụ. Trong một số trình ứng dụng, một *bên tham gia* chỉ yêu cầu các dịch vụ và một *bên tham gia* chỉ cung cấp các dịch vụ. Các trình ứng dụng này có một số tương đồng với các trình ứng dụng client-server (khách-chủ) truyền thống. Trong các trình ứng dụng khác, mỗi *bên tham gia* CÓ THỂ yêu cầu các dịch vụ của *bên tham gia* khác. Trong trường hợp đó, mối quan hệ giữa hai *bên tham gia* có thể được mô tả như một mối quan hệ ngang hàng hơn là một mối quan hệ client-server (khách-chủ).

5.2 Tạo một CPA từ hai CPP

Phần này khái quát quá trình phát hiện một *bên tham gia* để tiến hành *kinh doanh* và tạo một *CPA* từ hai *CPP* của *bên tham gia*. Nói chung, phần này là phần khái quát của một thủ tục có thể áp dụng và không được xem xét như một quy định quy chuẩn. Xem phụ lục E "Thành phần cấu tạo của *CPA* (Tham khảo)" để biết thêm thông tin.

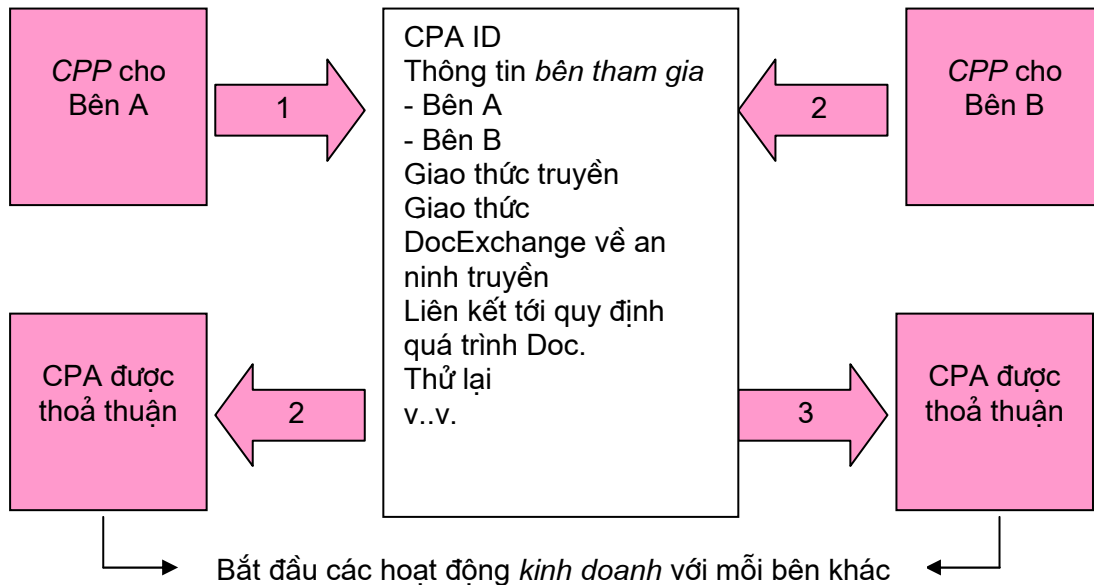
Hình 2 minh họa việc tạo một *CPP*. Bên tham gia A lập bảng kê thông tin được đặt trong một kho cho quá trình phát hiện ở trên, tạo ra một *CPP* chứa thông tin này và nhập nó vào trong một sổ đăng ký ebXML hoặc kho tương tự cùng với thông tin bổ sung về *bên tham gia* đó. Thông tin bổ sung đó có thể bao gồm một mô tả về công việc *kinh doanh* mà *bên tham gia* đó đang thực hiện. Ngay khi thông tin của *bên tham gia* A được chứa trong kho, các *bên tham gia* khác có thể phát hiện *bên tham gia* A bằng việc sử dụng các dịch vụ phát hiện của kho đó.

Hình 2 - Khái quát của Hồ sơ giao thức hợp tác (Collaboration Protocol Profile) (CPP)



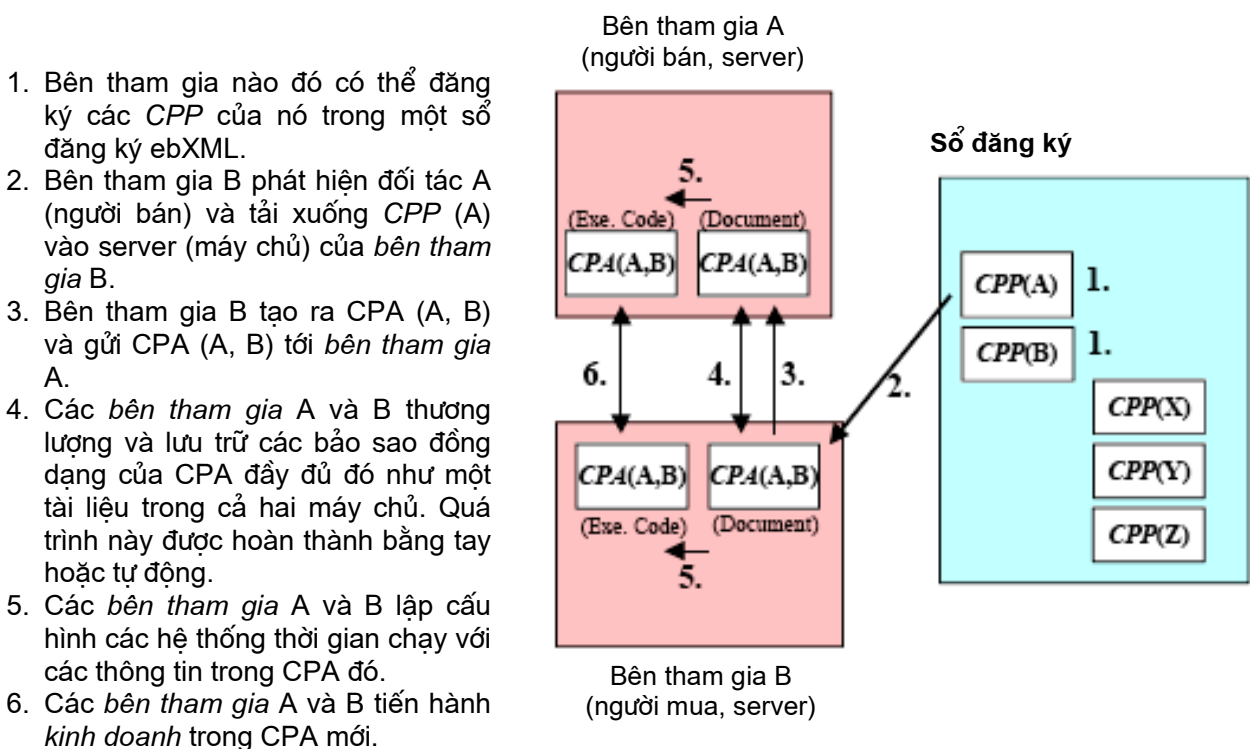
Trong Hình 3, bên *tham gia* A và bên *tham gia* B sử dụng các *CPP* của họ để xây dựng chung một bản sao đơn của một *CPA* bằng việc tính toán phần chung của thông tin trong các *CPP* của họ. *CPA* kết quả xác định cách thức hai bên *tham gia* sẽ xử sự trong việc thực hiện *hồ sơ hợp tác kinh doanh* của họ.

Hình 3 - Khái quát của thỏa thuận giao thức hợp tác (Collaboration Protocol Agreement)s (CPA)



Hình 4 minh họa toàn bộ quá trình đó. Các bước được liệt kê ở bên trái. Cuối quá trình đó là để hai bên *tham gia* cấu hình các hệ thống của họ từ các bản sao đồng dạng của *CPA* được thông qua và sau đó chúng sẵn sàng để thực hiện công việc *kinh doanh*.

Hình 4 - Khái quát kiến trúc vận hành của CPP/CPA trong sổ đăng ký



1. Bên tham gia nào đó có thể đăng ký các *CPP* của nó trong một sổ đăng ký ebXML.
2. Bên tham gia B phát hiện đối tác A (người bán) và tải xuống *CPP* (A) vào server (máy chủ) của bên tham gia B.
3. Bên tham gia B tạo ra *CPA* (A, B) và gửi *CPA* (A, B) tới bên tham gia A.
4. Các bên tham gia A và B thương lượng và lưu trữ các bản sao đồng dạng của *CPA* đầy đủ đó như một tài liệu trong cả hai máy chủ. Quá trình này được hoàn thành bằng tay hoặc tự động.
5. Các bên tham gia A và B lập cấu hình các hệ thống thời gian chạy với các thông tin trong *CPA* đó.
6. Các bên tham gia A và B tiến hành *kinh doanh* trong *CPA* mới.

5.3 Tạo một CPA từ mẫu CPA

Nói cách khác, một mẫu CPA có thể được sử dụng để tạo một CPA. Một mẫu CPA đại diện một mẫu đề nghị "điền vào chỗ trống" của bên tham gia với đối tác tham gia thương mại trong tương lai để thực hiện một hoặc nhiều *quá trình kinh doanh*. Ví dụ, như một mẫu có thể bao gồm các giá trị trình giữ chỗ (*placeholder*) đối với việc xác định các khía cạnh của bên tham gia khác. Để tạo một CPA từ một mẫu CPA, thì các giá trị trình giữ chỗ sẽ được thay thế bởi các giá trị thực đối với bên tham gia thương mại khác. Các giá trị thực có thể đạt được từ CPP của bên tham gia khác, nếu có thể hoặc bởi mục nhập dữ liệu trong một biểu mẫu HTML, giữa các khả năng khác. Phiên bản hiện tại của tiêu chuẩn này không hướng vào cách thức các giá trị trình giữ chỗ có thể được trình bày trong một CPA. Tuy nhiên, quá trình điền đầy đủ một mẫu CPA BẮT BUỘC dẫn đến một CPA hợp lệ. Chi tiết hơn về các mẫu CPA được đưa ra trong phụ lục E.

5.4 Phương thức làm việc của CPA

Một CPA mô tả tất cả các hiển nhiên hợp lệ, vì vậy có thể thực hiện, các tương tác giữa các *bên tham gia* và cách thức mà các tương tác này được tiến hành. Nó độc lập với các quá trình bên trong được thực thi tại mỗi *bên tham gia*. Mỗi *bên tham gia* thực thi các quá trình bên trong của chính nó và giao tiếp chúng với *hồ sơ hợp tác kinh doanh* được mô tả bởi CPA và tài liệu *quy định quá trình*. CPA không đưa ra các chi tiết của một các quá trình bên trong của *bên tham gia* với *bên tham gia* khác. Mục đích của CPA là để cung cấp một quy định mức-cao mà có thể được thông hiểu dễ dàng bởi con người và chưa đủ mức độ chính xác để thực hiện bởi máy tính.

Thông tin trong CPA được sử dụng để lập cấu hình các hệ thống của các *bên tham gia* để cho phép việc trao đổi các *thông điệp* trong giai đoạn thực hiện *hồ sơ hợp tác kinh doanh* được lựa chọn. Điển hình, phần mềm mà để thực hiện các trao đổi *thông điệp* đó và *thông điệp* khác hỗ trợ các tương tác giữa các *bên tham gia* là phần sụn để có thể hỗ trợ bất kỳ *hồ sơ hợp tác kinh doanh* được lựa chọn. Một thành phần của phần sụn này có thể là trình điều khiển dịch vụ *thông điệp* ebXML [ebMS]. Trong tiêu chuẩn này, các thuật ngữ "hệ thống thời gian thực" hoặc "phần mềm thời gian thực" được sử dụng có nghĩa là phần sụn.

CPA và tài liệu *quy định quá trình* để tham chiếu xác định một hội thoại giữa hai *bên tham gia*. Đối thoại này trình bày một đơn vị đơn của *kinh doanh* như được xác định bởi phần tử **BinaryCollaboration** của tài liệu *quy định quá trình*. Đối thoại này bao gồm một hoặc nhiều *giao dịch kinh doanh*, mỗi giao dịch trong chúng là một *thông điệp* yêu cầu từ một *bên tham gia* và không hoặc một *thông điệp* phản hồi từ *bên tham gia* khác. Tài liệu *quy định quá trình* xác định giữa *thông điệp* yêu cầu và *thông điệp* phản hồi đối với mỗi *giao dịch kinh doanh* và thứ tự mà *giao dịch kinh doanh* đó ĐƯỢC YÊU CẦU xuất hiện. Giải thích chi tiết xem [ebBPSS]. CPA thực CÓ THỂ tham chiếu nhiều hơn một tài liệu *quy định quá trình*. Khi một CPA tham chiếu nhiều hơn một tài liệu *quy định quá trình*, mỗi tài liệu *quy định quá trình* xác định một kiểu phân biệt của hội thoại. Một hội thoại bất kỳ chỉ liên quan một tài liệu *quy định quá trình* đơn.

Một hội thoại mới được bắt đầu mỗi thời điểm một khối đơn vị mới của *kinh doanh* được bắt đầu. Thủ tục *kinh doanh* này cũng xác định thời điểm hội thoại này kết thúc. Từ quan điểm của một *CPA* giữa *bên tham gia A* và *bên tham gia B*, hội thoại này bắt đầu tại *bên tham gia A* khi *bên tham gia A* gửi *thông điệp yêu cầu* khởi tạo tới *bên tham gia B*. Tại *bên tham gia B*, đối thoại này bắt đầu khi nó nhận yêu cầu khởi tạo của khối đơn vị của *kinh doanh* từ *bên tham gia A*. Một hội thoại kết thúc khi các *bên tham gia* đã hoàn thành khối đơn vị của *kinh doanh*.

CHÚ THÍCH: Hệ thống thời gian thực NÊN cung cấp một giao thức bằng ứng dụng *kinh doanh* có thể yêu cầu khởi tạo và kết thúc các hội thoại.

5.5 CPA có thể được thực hiện

Dựa trên khái niệm, máy server (máy chủ) doanh nghiệp-tới-doanh nghiệp (B2B) tại địa điểm của mỗi *bên tham gia* thực hiện *CPA* và tài liệu *quy định quá trình*. Máy server (máy chủ) B2B bao gồm phần mềm thời gian thực, Nghĩa là; phần sụn đó để hỗ trợ việc truyền thông với *bên tham gia* khác, việc thi hành của các chức năng được quy định trong *CPA* đó, việc ghép nối tới các quá trình phụ trợ (back-end) của mỗi *bên tham gia* và ghi lại các tương tác giữa các *bên tham gia* đối với các mục đích như kiểm tra và khôi phục. Phần sụn có thể hỗ trợ khái niệm của một hội thoại thực hiện trong thời gian dài như hiện thân của một đơn vị đơn của *kinh doanh* giữa các *bên tham gia*. Để cấu hình các hệ thống của hai *bên tham gia* đối với các thao tác B2B, thông tin đó trong bản sao của *CPA* đó và tài liệu *quy định quá trình* tại địa điểm của mỗi *bên tham gia* được cài đặt trong hệ thống thời gian thực đó. Thông tin tĩnh có thể được ghi lại trong một cơ sở dữ liệu cục bộ và thông tin khác trong *CPA* đó và tài liệu *quy định quá trình* có thể được sử dụng trong việc tạo ra hoặc tùy chỉnh mã cần thiết để hỗ trợ *CPA* đó.

CHÚ THÍCH: Nó có thể để cung cấp một công cụ tạo ra-*CPP/CPA* đồ họa để hiểu cả ngữ nghĩa của *CPP/CPA* và cú pháp XML đó. Điều quan trọng tương tự, các định nghĩa trong tiêu chuẩn này tạo thuận lợi cho việc thực hiện tự động, tại vị trí của mỗi *bên tham gia*, mã cần thiết để thực thi *CPA* đó, tuân theo các quy tắc của nó và giao thức với các quá trình phụ trợ (back-end) của *bên tham gia* đó.

5.6 Định nghĩa và phạm vi

Tiêu chuẩn này định nghĩa và giải thích nội dung các tài liệu XML của *CPP* và *CPA*. Phạm vi của nó được giới hạn trong các định nghĩa này. Nó không xác định cách thức soạn thảo một *CPA* từ hai *CPP* cũng không xác định bất kỳ điều gì liên quan đến hỗ trợ thời gian thực đối với *CPP* và *CPA*. Nó bao gồm một số khuyến cáo và đề xuất tham khảo về thành phần cấu tạo của *CPA* từ hai *CPP* và hỗ trợ thời gian thực, các chú thích này nhằm làm cho các định nghĩa *CPP* và *CPA* sáng sủa dễ hiểu hơn. Xem phần 11 một giải thích phù hợp với tiêu chuẩn này.

CHÚ THÍCH: Tiêu chuẩn này được giới hạn trong việc định nghĩa các nội dung của *CPP* và *CPA* và nó có thể phù hợp với nó một cách đơn thuần bằng việc tạo ra một tài liệu *CPP* hoặc *CPA* để phù hợp với tài liệu về giản đồ XML được định nghĩa trong tài liệu này. Tuy nhiên, quan trọng để hiểu rằng giá trị của tiêu chuẩn này nằm ở điều mà nó cho phép một hệ thống thời gian thực để hỗ trợ thương mại điện tử giữa hai *bên tham gia* dưới hướng dẫn thông tin trong *CPA*.

6 Định nghĩa của CPP

CPP xác định khả năng của một *bên tham gia* nhằm thực hiện trong *kinh doanh* điện tử với các *bên tham gia* khác. Các khả năng này bao gồm cả các khả năng về công nghệ, như các thủ tục truyền *thông điệp* và truyền thông được hỗ trợ và các khả năng *kinh doanh* trong điều kiện mà các *hỗ trợ hợp tác kinh doanh* bên tham gia đó hỗ trợ.

Phần này định nghĩa và đề cập đến các chi tiết trong CPP dưới dạng các phần tử XML riêng. Nội dung đề cập này được minh họa với một số đoạn XML. Xem phụ lục D đối với giản đồ XML và phụ lục A đối với các tài liệu CPP minh họa.

Các phần tử **ProcessSpecification** (quy định quá trình), **DeliveryChannel** (kênh truyền), **DocExchange** (trao đổi tài liệu) và **Transport** (truyền tải) của CPP mô tả việc xử lý của một khối đơn vị *kinh doanh* (hội thoại). Các phần tử này từ một cấu trúc phân lớp ở mức độ nào đó tương tự một mô hình truyền thông được phân lớp.

Lớp quy định-quá trình - Lớp quy định-quá trình xác định điểm cốt lõi của thỏa thuận *kinh doanh* giữa các *bên tham gia*: Các dịch vụ (giao dịch *kinh doanh*) mà các *bên tham gia* đối với CPA đó có thể yêu cầu bên khác và các quy tắc chuyển tiếp để xác định thứ tự các yêu cầu. Lớp này được xác định bởi tài liệu quy định quá trình riêng biệt đó là được tham chiếu bởi CPP và CPA.

Các kênh truyền - Một kênh truyền mô tả các đặc điểm gửi-*thông điệp* và nhận-*thông điệp* của một *bên tham gia*. Nó bao gồm một định nghĩa trao đổi-tài liệu và một định nghĩa truyền. Nhiều kênh truyền có thể được định nghĩa trong một CPP.

Lớp trao đổi-tài liệu - Lớp trao đổi-tài liệu quy định việc xử lý của các tài liệu *kinh doanh* bởi Chức năng trao đổi-*thông điệp*. Các đặc tính được quy định bao gồm các đặc điểm mật mã hoá, chữ ký dạng số và truyền thông *điệp* xác thực. Các lựa chọn đối với lớp trao đổi-tài liệu là bổ sung cho các lựa chọn đó đối với lớp truyền. Ví dụ, nếu an ninh *thông điệp* được yêu cầu và giao thức truyền được lựa chọn không cung cấp mật mã hoá *thông điệp*, thì mật mã hoá *thông điệp* BẮT BUỘC được quy định trong Lớp trao đổi-tài liệu. Thủ tục đối với việc trao đổi *thông điệp* giữa hai *bên tham gia* được xác định bởi quy định dịch vụ *thông điệp* ebXML[ebMS] hoặc các dịch vụ truyền *thông điệp* tương tự khác.

Lớp truyền - Lớp truyền xác định thủ tục truyền tới được sử dụng trong việc gửi các *thông điệp* thông qua mạng và định nghĩa các địa chỉ đầu cuối, cùng với các đặc tính khác đa dạng của thủ tục truyền. Các lựa chọn của các đặc tính trong lớp truyền là bổ sung trong lớp trao đổi-tài liệu (xem "Lớp trao đổi-tài liệu" ở trên.)

Chú ý rằng các lớp chức năng đó được chứa bởi CPP độc lập với các nội dung của thiết bị mang các tài liệu *kinh doanh*.

6.1 Định danh toàn cầu-đuy nhất của tài liệu CPP cụ thể

Khi một *CPP* được thay thế trong một sổ đăng ký ebXML hoặc đăng ký khác, sổ đăng ký này ấn định cho nó một định danh toàn cầu duy nhất (GUID) là một phần trong siêu dữ liệu của nó. GUID đó có thể được sử dụng phân biệt giữa các *CPP* thuộc cùng *bên tham gia*.

CHÚ THÍCH: Sổ đăng ký không thể chèn GUID vào *CPP* đó. Nói chung, sổ đăng ký không thay đổi nội dung của các tài liệu được đệ trình cho nó. Hơn nữa, một *CPP* có thể được ký và thay đổi sẽ làm mất hiệu lực của chữ ký đó.

6.2 Cấu trúc CPP

Sau đây là cấu trúc tổng thể của *CPP*. Trừ khi có chú thích khác, các phần tử của *CPP* **BẮT BUỘC** theo thứ tự được chỉ ra ở đây. Các phần tiếp theo mô tả mỗi phần tử trong các phần tử chi tiết hơn.

```
<tp:CollaborationProtocolProfile
  xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-
2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
cppa/schema/cpp-cpa-2_0.xsd http://www.oasis-open.org/committees/ebxml-
cppa/schema/cpp-cpa-2_0.xsd"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  tp:cppid="uri:companyA-cpp"
  tp:version="2_0b">
  <tp:PartyInfo> <!-- one or more -->
  ...
  </tp:PartyInfo>
  <tp:SimplePart id="..."> <!-- one or more -->
  ...
  </tp:SimplePart>
  <tp:Packaging id="..."> <!-- one or more -->
  ...
  </tp:Packaging>
  <tp:Signature> <!-- zero or one -->
  ...
  </tp:Signature>
  <tp:Comment>text</tp:Comment> <!-- zero or more -->
</tp:CollaborationProtocolProfile>
```

6.3 Phần tử CollaborationProtocolProfile (hồ sơ giao thức hợp tác)

Phần tử **CollaborationProtocolProfile** (hồ sơ giao thức hợp tác) là phần tử gốc của tài liệu XML của *CPP*.

Các khai báo XML YÊU CẦU tên miền [XML] [XMLNS] đối với tài liệu cơ sở như sau:

- tên miền của *CPP/CPA*: xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd";
- tên miền của chữ ký dạng số XML: xmlns:ds="http://www.w3.org/2000/09/xmldsig#";
- và tên miền XLink: xmlns:xlink="http://www.w3.org/1999/xlink".

Ngoài ra, phần tử **CollaborationProtocolProfile** (hồ sơ giao thức hợp tác) bao gồm thuộc tính **cppid** (id của cpp) ĐƯỢC YÊU CẦU để cung cấp một định danh duy nhất đối với tài liệu đó, cộng với một

TCVN ISO/TS 15000-1 : 2007

thuộc tính **version** (phiên bản) ĐƯỢC YÊU CẦU để chỉ ra phiên bản của giản đồ đó. Mục đích của nó là để định danh phiên bản của giản đồ đó mà *CPP* đó phù hợp. Giá trị thuộc tính **version** (phiên bản) NÊN là chuỗi ký tự như "2_0a", "2_0b", v..v.

CHÚ THÍCH: Phương pháp ấn định các giá trị cppid duy nhất được cho phép thực hiện.

Phần tử **CollaborationProtocolProfile** (hồ sơ giao thức hợp tác) PHẢI bao gồm các phần tử con sau đây:

- một hoặc nhiều phần tử **PartyInfo** (thông tin bên tham gia) YÊU CẦU để định danh tổ chức đó (hoặc một phần tổ chức đó) khả năng của tổ chức đó được mô tả bởi *CPP*;
- một hoặc nhiều phần tử **SimplePart** (thành phần đơn giản) YÊU CẦU để mô tả các cấu thành được sử dụng tạo nên các *thông điệp* hỗn hợp;
- một hoặc nhiều phần tử **Packaging** (đóng gói) YÊU CẦU để mô tả cách thức *tiêu đề thông điệp* đó và các thành phần cấu thành của vùng mang thông tin được đóng gói để truyền;
- không hoặc một phần tử **Signature** (chữ ký) bao gồm chữ ký dạng số để ký tài liệu *CPP*;
- không hoặc nhiều phần tử **Comment** (chú giải).

Một tài liệu *CPP* có thể ký dạng số để cung cấp đối với biện pháp đảm bảo rằng tài liệu đó chưa bị sửa đổi (toàn vẹn) và để cung cấp đối với biện pháp xác thực tác giả của tài liệu đó. Một *CPP* số được ký PHẢI được ký có sử dụng công nghệ để phù hợp với quy định chữ ký dạng số XML của W3C/IETF[XMLDSIG].

6.4 Phần tử PartyInfo (Thông tin bên tham gia)

Phần tử **PartyInfo** (thông tin bên tham gia) định danh tổ chức có khả năng được mô tả trong *CPP* này và bao gồm toàn bộ các chi tiết về *bên tham gia* này. Hơn một phần tử **Info** (thông tin) của *bên tham gia* có thể được cung cấp trong một *CPP* nếu tổ chức đó chọn để tự đại diện như các phòng ban với các đặc điểm khác nhau. Mỗi phần tử con của **PartyInfo** (thông tin bên tham gia) được đề cập sau đó. Cấu trúc tổng thể của phần tử **PartyInfo** (thông tin bên tham gia) là như sau:

```
<tp:PartyInfo
tp:partyName="..."
  tp:defaultMshChannelId="..."
  tp:defaultMshPackageId="...">
  <tp:PartyId tp:type="..."> <!-- one or more -->
  ...
  </tp:PartyId>
  <tp:PartyRef xlink:href="..."/>
<tp:CollaborationRole> <!-- one or more -->
...
  </tp:CollaborationRole>
  <tp:Certificate> <!-- one or more -->
  ...
  </tp:Certificate>
<tp:SecurityDetails> <!-- one or more -->
...
  </tp:SecurityDetails>
```

```

    <tp:DeliveryChannel> <!-- one or more -->
    ...
  </tp:DeliveryChannel>
  <tp:Transport> <!-- one or more -->
  ...
  </tp:Transport>
  <tp:DocExchange> <!-- one or more -->
  ...
  </tp:DocExchange>
  <tp:OverrideMshActionBinding> <!-- zero or more -->
  ...
  </tp:OverrideMshActionBinding>
</tp:PartyInfo>

```

Phần tử **PartyInfo** (thông tin bên tham gia) bao gồm một thuộc tính **partyName** (tên bên tham gia) YÊU CẦU để chỉ thị tên chung và con người có thể đọc của tổ chức đó. Không giống **PartyID**, **partyName** không phải là duy nhất. Tuy nhiên, giá trị thuộc tính **name** (tên) của mỗi *bên tham gia* PHẢI đầy đủ ý nghĩa để định danh một cách trực tiếp tổ chức đó hoặc phòng ban của một tổ chức được mô tả trong phần tử **PartyInfo** (thông tin bên tham gia).

Ví dụ sau minh họa tên có thể của *bên tham gia*.

```

  <tp:PartyInfo tp:partyName="Example, Inc."...</tp:PartyInfo>
  <tp:PartyInfo tp:partyName="Example, Inc. US Western Division">
  ...
</tp:PartyInfo>

```

Phần tử **PartyInfo** (thông tin bên tham gia) cũng bao gồm một thuộc tính **defaultMshChannelId** (ID của kênh MSH mặc định) YÊU CẦU và một thuộc tính **defaultMshPackageId** (ID của gói MSH mặc định) YÊU CẦU. Thuộc tính **defaultMshChannelId** (id kênh truyền mặc định) định danh **DeliveryChannel** (kênh truyền) mặc định được sử dụng đối với việc gửi các thông điệp mức trình điều khiển dịch vụ thông điệp độc lập [ebMS] (như là., Acknowledgement (xác thực), Error (Báo lỗi), StatusRequest (yêu cầu tình trạng), StatusResponse (phản hồi tình trạng), Ping (một tiện ích đi kèm theo internet), Pong (một tiện ích đi kèm theo internet)) là để được truyền không đồng bộ. Khi sử dụng phương thức trả lời đồng bộ, các thông điệp mức trình điều khiển dịch vụ *thông điệp* được truyền trở lại mặc định đồng bộ. Trạng thái mặc định này có thể được ghi đè thông qua việc sử dụng các phần tử **OverrideMshActionBinding** (quy định hoạt động MSH ghi đè). Thuộc tính **defaultMshPackageId** (ID gói MSH mặc định) định danh việc đóng gói mặc định được sử dụng đối với việc gửi các thông điệp mức trình điều khiển dịch vụ *thông điệp* độc lập [ebMS].

Phần tử **PartyInfo** (thông tin bên tham gia) bao gồm các phần tử con sau đây:

- một hoặc nhiều phần tử **PartyId** (ID bên tham gia) YÊU CẦU để cung cấp các định danh logic đối với tổ chức đó;
- một hoặc nhiều phần tử **PartyRef** (tham chiếu bên tham gia) YÊU CẦU để cung cấp các con trỏ bổ sung thông tin về *bên tham gia* đó;
- một hoặc nhiều phần tử **CollaborationRole** (vai trò hợp tác) YÊU CẦU để định danh các vai trò mà *bên tham gia* này đang diễn theo ngữ cảnh của *quy định quá trình*;

TCVN ISO/TS 15000-1 : 2007

- một hoặc nhiều phần tử **Certificate** (chứng chỉ) YÊU CẦU để định danh các chứng chỉ được sử dụng bởi *bên tham gia* này trong các chức năng một mình;
- một hoặc nhiều phần tử **SecurityDetails** (chi tiết an ninh) YÊU CẦU để định danh các mẫu neo chứng thực và quy định chính sách an ninh được sử dụng bởi *bên tham gia* này trong các chức năng một mình;
- một hoặc nhiều phần tử **DeliveryChannel** (kênh truyền) YÊU CẦU để định nghĩa các đặc điểm mà *bên tham gia* đó có thể sử dụng để gửi và/hoặc nhận các *thông điệp*. Nó bao gồm cả hai thủ tục truyền (như là; HTTP) và thủ tục truyền *thông điệp* (như là; dịch vụ *thông điệp* ebXML);
- một hoặc nhiều phần tử **Transport** (truyền tải) YÊU CẦU để định nghĩa các đặc điểm của (các) thủ tục truyền mà *bên tham gia* đó có thể hỗ trợ để gửi và/hoặc nhận các *thông điệp*;
- một hoặc nhiều phần tử **DocExchange** (trao đổi tài liệu) YÊU CẦU để định nghĩa các đặc điểm trao đổi-*thông điệp*, như các giao thức mật mã hoá ký, mà *bên tham gia* đó có thể hỗ trợ;
- không hoặc nhiều phần tử **OverrideMshActionBinding** (quy định hoạt động MSH ghi đè) để quy định **DeliveryChannel** (kênh truyền) sử dụng đối với các *thông điệp* mức trình điều khiển dịch vụ *thông điệp* được truyền không đồng bộ.

6.4.1 Phần tử PartyId (ID bên tham gia)

Phần tử **PartyId** (ID bên tham gia) YÊU CẦU cung cấp một định danh PHẢI được sử dụng để định danh lôgic *bên tham gia* đó. Các phần tử **PartyId** (ID bên tham gia) bổ sung có thể có mặt trong cùng phần tử **PartyInfo** (thông tin bên tham gia) để cung cấp cho các định danh lôgic thay thế đối với *bên tham gia* đó. Nếu *bên tham gia* đó được ưu tiên như đối với định danh được sử dụng, thì các phần tử **PartyId** (ID bên tham gia) NÊN được liệt kê theo thứ tự của sự ưu tiên bắt đầu với định danh được ưu tiên nhất.

Trong một *CPP* bao gồm nhiều phần tử **PartyInfo** (thông tin bên tham gia), các phần tử **PartyInfo** (thông tin bên tham gia) khác nhau có thể chứa các phần tử **PartyId** (ID bên tham gia) để định nghĩa các định danh lôgic khác nhau. Điều này cho phép lượng lớn các tổ chức, ví dụ, để có các định danh khác nhau đối với các mục đích khác nhau.

Giá trị của phần tử **PartyId** (ID bên tham gia) là bất kỳ chuỗi ký tự nào để cung cấp một định danh duy nhất. Định danh đó có thể là bất kỳ định danh nào được thông hiểu bởi cả hai *bên tham gia* với một *CPA*. Điển hình, định danh đó sẽ được liệt kê trong một danh mục được nhiều người biết đến như DUNS (Dun và Bradstreet) hoặc trong bất kỳ hệ thống đặt tên được quy định bởi [ISO6523].

Phần tử **PartyId** (ID bên tham gia) có một thuộc tính MẶC NHIÊN đơn: **type** (kiểu) mà có bất kỳ giá trị URI [XMLSCHEMA-2] nào đó.

Nếu thuộc tính **type** (kiểu) có mặt, thì nó cung cấp một phạm vi hoặc tên miền cho nội dung của phần tử **PartyId** (ID bên tham gia) đó.

Nếu thuộc tính **type** (kiểu) không có mặt, thì nội dung của phần tử **PartyId** (ID bên tham gia) BẮT BUỘC là một URI để phù hợp với [RFC2396]. Khuyến cáo rằng giá trị thuộc tính **type** (kiểu) là một URN để định nghĩa một tên miền đối với giá trị của phần tử **PartyId** (ID bên tham gia). Điển hình, URN sẽ được đăng ký trong một danh mục được nhiều người biết đến của các định danh của tổ chức.

Ví dụ sau minh họa hai tham chiếu URI.

```
<tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:duns">123456789</tp:PartyId>
<tp:PartyId>urn:icann:example.com</tp:PartyId>
```

Ví dụ đầu tiên là số hiệu DUNS của *bên tham gia* đó. Giá trị đó là số hiệu DUNS của tổ chức đó.

Ví dụ thứ hai chỉ ra một URN tùy ý. Đây có thể là một URN mà *bên tham gia* đó đã đăng ký với IANA, tổ chức có thẩm quyền cấp số hiệu ấn định Internet (*Internet Assigned Numbers Authority*) (<http://www.iana.org>) để tự định danh trực tiếp.

Tài liệu sau đây thảo luận các đại diện đặt tên và cách chúng được định danh thông qua các giá trị URI của thuộc tính **type** (kiểu) đó:

http://www.oasis-open.org/committees/ebxml-cppa/documents/PartyID_Types.shtml

6.4.2 Phần tử **PartyRef** (Tham chiếu bên tham gia)

Phần tử **PartyRef** (tham chiếu bên tham gia) cung cấp một liên kết, dưới dạng một URI, để thông tin bổ sung về *bên tham gia* đó. Điển hình, đây sẽ là URL từ nơi mà thông tin đó có thể được thu được. Thông tin đó có thể tại trang web của *bên tham gia* đó hoặc trong một kho có thể truy cập công khai như một sổ đăng ký ebXML, một kho UDDI (www.uddi.org) hoặc một danh mục giao thức truy cập danh mục ít quan trọng [RFC2251] (LDAP). Thông tin sẵn có tại URI đó có thể bao gồm thông tin về điểm liên lạc như là các tên, địa chỉ và số điện thoại hoặc thông tin về nội dung như là các vị trí địa lý và các đoạn thuộc ngành công nghiệp hoặc có thể thông tin thêm về các *hồ sơ hợp tác kinh doanh* mà *bên tham gia* đó hỗ trợ. Thông tin này có thể dưới dạng một thành phần chính của ebXML [ccOVER]. Việc xác định nội dung hoặc dạng thức của thông tin tại URI đó không nằm trong phạm vi của tiêu chuẩn này.

Phần tử **PartyRef** (tham chiếu bên tham gia) là một đường liên kết đơn giản [XLINK]. Nó có các thuộc tính sau:

- một thuộc tính **xlink:type** CỐ ĐỊNH;
- một thuộc tính **xlink:href** YÊU CẦU;
- một thuộc tính **type** (kiểu) MẶC NHIÊN;
- một thuộc tính **schemaLocation** (vị trí giản đồ) MẶC NHIÊN.

Các nội dung của tài liệu này được tham chiếu bởi phần tử **PartyRef** (tham chiếu bên tham gia) dễ bị thay đổi tại bất kỳ thời điểm nào. Vì vậy, KHÔNG NÊN lưu trữ nó trong một khoảng thời gian dài. Hơn

nữa, giá trị của **xlink:href** NÊN được xóa tham chiếu chỉ khi các nội dung của của tài liệu này là cần thiết.

6.4.2.1 Thuộc tính **xlink:type**

Thuộc tính **xlink:type** CỐ ĐỊNH PHẢI có một giá trị "đơn giản". Điều này xác định phần tử đó như là một liên kết [XLINK] đơn giản.

6.4.2.2 Thuộc tính **xlink:href**

Thuộc tính **xlink:href** YÊU CẦU PHẢI có một giá trị đó là một URI để phù hợp với [RFC2396] và xác định vị trí của thông tin bên ngoài về *bên tham gia* đó.

6.4.2.3 Thuộc tính **type** (Kiểu)

Giá trị của một thuộc tính **type** (kiểu) MẶC NHIÊN xác định kiểu tài liệu thông tin bên ngoài về *bên tham gia* đó. Nó BẮT BUỘC là một URI để định nghĩa tên miền được kết hợp với thông tin về *bên tham gia* đó. Nếu thuộc tính **type** (kiểu) bị lược bỏ, thông tin bên ngoài về *bên tham gia* đó BẮT BUỘC là một trang web HTML.

6.4.2.4 Thuộc tính **schemaLocation** (Vị trí giản đồ)

Giá trị của một thuộc tính **schemaLocation** (vị trí giản đồ) MẶC NHIÊN cung cấp một URI cho giản đồ để mô tả cấu trúc của thông tin bên ngoài.

Một ví dụ về phần tử **PartyRef** (tham chiếu bên tham gia) là:

```
<tp:PartyRef xlink:type="simple"
  xlink:href="http://example2.com/ourInfo.xml"
  tp:type="urn:oasis:names:tc:ebxml-cppa:contact-info"
  tp:schemaLocation="http://example2.com/ourInfo.xsd"/>
```

6.4.3 Phần tử **CollaborationRole** (Vai trò hợp tác)

Phần tử **CollaborationRole** (vai trò hợp tác) kết hợp một *bên tham gia* với một vai trò cụ thể trong *hồ sơ hợp tác kinh doanh*. Nói chung, *quy định-quá trình* này được xác định dưới dạng các vai trò như "người mua" và "người bán". Kết nối giữa một *bên tham gia* cụ thể và (các) vai trò có khả năng hoàn thành trong một nội dung của một *quy định-quá trình* được xác định trong cả hai tài liệu *CPP* và *CPA*. Trong một *CPP*, phần tử **CollaborationRole** (vai trò hợp tác) xác định vai trò mà *bên tham gia* đó có khả năng đóng vai trong mỗi tài liệu *quy định quá trình* được tham chiếu bởi *CPP* đó. Một ví dụ về phần tử **CollaborationRole** (vai trò hợp tác), dựa trên cơ sở RosettaNet™ PIP 3A4 là:

```
<tp:CollaborationRole >
  <tp:ProcessSpecification
    tp:version="2.0a"
    tp:name="PIP3A4RequestPurchaseOrder"
    xlink:type="simple"
    xlink:href="http://www.rosettanet.org/processes/3A4.xml"/>
  <tp:Role
    tp:name="Buyer"
    xlink:type="simple"
```



```

        xlink:href="http://www.rosettanet.org/processes/3A4.xml#BuyerId"/>
    <tp:ApplicationCertificateRef tp:certId="CompanyA_AppCert"/>
    <tp:ServiceBinding>
        <tp:Service
tp:type="anyURI">urn:icann:rosettanet.org:bpid:3A4$2.0</tp:Service>
        <tp:CanSend>
            <tp:ThisPartyActionBinding
                tp:id="companyA_ABID1"
                tp:action="Purchase Order Request Action"
                tp:packageId="CompanyA_RequestPackage">
            <tp:BusinessTransactionCharacteristics
                tp:isNonRepudiationRequired="true"
                tp:isNonRepudiationReceiptRequired="true"
                tp:isConfidential="transient"
                tp:isAuthenticated="persistent"
                tp:isTamperProof="persistent"
                tp:isAuthorizationRequired="true"
                tp:timeToAcknowledgeReceipt="PT2H"
                tp:timeToPerform="P1D"/>
            <tp:ActionContext
                tp:binaryCollaboration="Request Purchase Order"
                tp:businessTransactionActivity="Request Purchase Order"
                tp:requestOrResponseAction="Purchase Order Request
Action"/>
                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
            </tp:ThisPartyActionBinding>
        </tp:CanSend>
        <tp:CanSend>
            <tp:ThisPartyActionBinding
                tp:id="companyA_ABID2"
                tp:action="ReceiptAcknowledgment"
                tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
            <tp:BusinessTransactionCharacteristics
                tp:isNonRepudiationRequired="true"
                tp:isNonRepudiationReceiptRequired="true"
                tp:isConfidential="transient"
                tp:isAuthenticated="persistent"
                tp:isTamperProof="persistent"
                tp:isAuthorizationRequired="true"
                tp:timeToAcknowledgeReceipt="PT2H"
                tp:timeToPerform="P1D"/>
                <tp:ChannelId>asyncChannelA1</tp:ChannelId>
            </tp:ThisPartyActionBinding>
        </tp:CanSend>
        <tp:CanReceive>
            <tp:ThisPartyActionBinding
                tp:id="companyA_ABID3"
                tp:action="Purchase Order Confirmation Action"
                tp:packageId="CompanyA_ResponsePackage">
            <tp:BusinessTransactionCharacteristics
                tp:isNonRepudiationRequired="true"
                tp:isNonRepudiationReceiptRequired="true"
                tp:isConfidential="transient"
                tp:isAuthenticated="persistent"
                tp:isTamperProof="persistent"
                tp:isAuthorizationRequired="true"
                tp:timeToAcknowledgeReceipt="PT2H"
                tp:timeToPerform="P1D"/>
            <tp:ActionContext
                tp:binaryCollaboration="Request Purchase Order"
                tp:businessTransactionActivity="Request Purchase Order"
                tp:requestOrResponseAction="Purchase Order Confirmation
Action"/>
                <tp:ChannelId>asyncChannelA1</tp:ChannelId>

```

```

        </tp:ThisPartyActionBinding>
    </tp:CanReceive>
    <tp:CanReceive>
        <tp:ThisPartyActionBinding
            tp:id="companyA_ABID4"
            tp:action="ReceiptAcknowledgment"
            tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
            tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"
        tp:timeToPerform="P1D"/>
        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    </tp:CanReceive>
    <tp:CanReceive>
        <tp:ThisPartyActionBinding
            tp:id="companyA_ABID5"
            tp:action="Exception"
            tp:packageId="CompanyA_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"

        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"
        tp:timeToPerform="P1D"/>
        <tp:ChannelId>asyncChannelA1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    </tp:CanReceive>
</tp:ServiceBinding>
</tp:CollaborationRole>

```

Để chỉ ra vai trò mà *bên tham gia* đó có thể đóng vai trong nhiều hơn một *hồ sơ hợp tác kinh doanh* hoặc nhiều hơn một vai trò trong một *hồ sơ hợp tác kinh doanh* cho trước, phần tử **PartyInfo** (thông tin bên tham gia) PHẢI chứa nhiều hơn một phần tử **CollaborationRole** (vai trò hợp tác). Mỗi phần tử **CollaborationRole** (vai trò hợp tác) PHẢI chứa kết nối phù hợp giữa phần tử **ProcessSpecification** (quy định quá trình) và phần tử **Role** (vai trò).

Phần tử **CollaborationRole** (vai trò hợp tác) PHẢI bao gồm các phần tử con sau đây: một phần tử **ProcessSpecification** (quy định quá trình) YÊU CẦU, một phần tử **Role** (vai trò) YÊU CẦU, không hoặc một phần tử **ApplicationCertificateRef** (tham chiếu chứng chỉ ứng dụng), không hoặc một phần tử **ApplicationSecurityDetailsRef** (tham chiếu chi tiết an ninh của ứng dụng) và một phần tử **ServiceBinding**. Phần tử **ProcessSpecification** (quy định quá trình) xác định tài liệu *quy định quá trình* để định nghĩa vai trò. Phần tử **Role** (vai trò) xác định vai trò mà *bên tham gia* đó có khả năng hỗ trợ. Phần tử **ApplicationCertificateRef** (tham chiếu chứng chỉ của ứng dụng) xác định chứng chỉ được sử dụng đối với chữ ký mức ứng dụng và mật mã hóa. Phần tử **ApplicationSecurityDetailsRef** (tham

chiếu chi tiết an ninh của chứng chỉ) xác định các mẫu neo chứng thực và chính sách an ninh sẽ được áp dụng cho bất kỳ chứng chỉ lớp-ứng dụng nào được đề nghị bởi *bên tham gia* khác. Phần tử **ServiceBinding** (quy định dịch vụ) PHẢI bao gồm không hoặc nhiều Phần tử **CanSend** (có thể gửi) và không hoặc nhiều phần tử **CanReceive** (có thể nhận). Phần tử **CanReceive** (có thể nhận) và **CanSend** (có thể gửi) định danh các phần tử **DeliveryChannel** (kênh truyền) là để được sử dụng đối với việc gửi và nhận các *thông điệp* tiến hành *kinh doanh* bằng các vai trò theo truy vấn. Chúng cũng có thể được sử dụng đối với việc quy định các kênh truyền đối với các *thông điệp* báo hiệu *kinh doanh*.

Mỗi *bên tham gia* PHẢI có một kênh truyền mặc định đối với việc phân phát của các tín hiệu mức trình quản lý dịch vụ thông điệp độc lập như (truyền thông điệp tin cậy) Acknowledgment (báo nhận), Error (lỗi), StatusRequest (yêu cầu trạng thái), StatusResponse (phản hồi trạng thái), v..v.

6.4.4 Phần tử ProcessSpecification (Quy định quá trình)

Phần tử **ProcessSpecification** (quy định quá trình) cung cấp liên kết tới tài liệu *quy định quá trình* để định nghĩa các tương tác giữa hai *bên tham gia*. KHUYẾN CÁO rằng mô tả hợp tác *kinh doanh* này được chuẩn bị phù hợp với giản đồ quy định quá trình kinh doanh ebXML [ebBPSS]. Tài liệu *quy định quá trình* có thể được giữ lại trong một số đăng ký ebXML.

CHÚ THÍCH: Một *bên tham gia* có thể mô tả *hồ sơ hợp tác kinh doanh* có sử dụng bất kỳ thay thế mong muốn cho giản đồ quy định quá trình kinh doanh ebXML. Khi một mô tả *hồ sơ hợp tác kinh doanh* thay thế được sử dụng, các *bên tham gia* tới một CPA BẮT BUỘC đồng ý về cách phiên dịch mô tả *hồ sơ hợp tác kinh doanh* đó và cách phiên dịch các phần tử đó trong CPA đó để tham chiếu thông tin trong mô tả *hồ sơ hợp tác kinh doanh* đó. Các phần tử chịu ảnh hưởng trong CPA đó là phần tử **Role** (vai trò), phần tử **CanReceive** (có thể nhận) và **CanSend** (có thể gửi), phần tử **ActionContext** (ngữ cảnh hành động) và một số thuộc tính của phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh).

Cú pháp của phần tử **ProcessSpecification** (quy định quá trình) là:

```
<tp:ProcessSpecification
  tp:version="2.0a"
  tp:name="PIP3A4RequestPurchaseOrder"
  xlink:type="simple"
  xlink:href="http://www.rosettanet.org/processes/3A4.xml"
  uuid="urn:icann:rosettanet.org:bpid:3A4$2.0">
  <ds:Reference ds:URI="http://www.rosettanet.org/processes/3A4.xml">
    <ds:Transforms>
      <ds:Transform
ds:Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    </ds:Transforms>
    <ds:DigestMethod
      ds:Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</ds:DigestValue>
  </ds:Reference>
</tp:ProcessSpecification>
```

Phần tử **ProcessSpecification** (quy định quá trình) có không hoặc nhiều phần tử con **ds:Reference** và các thuộc tính sau:

- một thuộc tính **name** (tên) YÊU CẦU;

- một thuộc tính **version** (phiên bản) YÊU CẦU;
- một thuộc tính **xlink:type** CỐ ĐỊNH;
- một thuộc tính **xlink:href** YÊU CẦU;
- một thuộc tính **uuid** MẶC NHIÊN.

Phần tử **ProcessSpecification** (quy định quá trình) bao gồm không hoặc nhiều phần tử **ds:Reference** được lập tuân theo quy định chữ ký dạng số XML[XMLDSIG]. Phần tử **ds:Reference** đầu tiên, nếu có mặt, thì liên quan với thuộc tính **xlink:type** và **xlink:hrefs** như sau. Mỗi phần tử **ProcessSpecification** (quy định quá trình) PHẢI bao gồm một thuộc tính **xlink:href** và một thuộc tính **xlink:type** với một giá trị "simple". trong trường hợp tài liệu CPP (CPA) được ký, phần tử **ds:Reference** đầu tiên có mặt BẮT BUỘC bao gồm một thuộc tính **ds:URI** mà giá trị của nó đồng dạng với giá trị thuộc tính **xlink:href** trong phần tử **ProcessSpecification** (quy định quá trình) kèm theo. Phần tử **ds:Reference** quy định một phương pháp hệ thống và giá trị hệ thống để cho phép kiểm tra tài liệu *quy định quá trình* được tham chiếu đã bị thay đổi chưa. Phần tử **ds:Reference** bổ sung cần thiết Nếu **ProcessSpecification** được tham chiếu lần lượt bao gồm (như là., các tham chiếu) các **ProcessSpecification** khác. Cụ thể, phần tử **ds:Reference** BẮT BUỘC được cung cấp tương đương với kết thúc ngoài của tất cả **ProcessSpecification** được tham chiếu trực tiếp hoặc gián tiếp để đảm bảo rằng không phần tử nào trong số đó bị thay đổi.

6.4.4.1 Thuộc tính name (Tên)

Phần tử **ProcessSpecification** (quy định quá trình) BẮT BUỘC bao gồm một thuộc tính **name** (tên) YÊU CẦU: một chuỗi để xác định *quy định-quá trình kinh doanh* đang thực hiện. Nếu tài liệu *quy định quá trình* được xác định bởi quy định ebXML về *quá trình kinh doanh* [ebBPSS], thì thuộc tính này BẮT BUỘC được đặt tên đối với phần tử **ProcessSpecification** (quy định quá trình) tương đương trong *quy định quá trình kinh doanh* cụ thể.

6.4.4.2 Thuộc tính version (Phiên bản)

Phần tử **ProcessSpecification** (quy định quá trình) bao gồm một thuộc tính **version** (phiên bản) YÊU CẦU để chỉ ra phiên bản của tài liệu *quy định quá trình* được xác định bởi thuộc tính **xlink:href** (và cũng được xác định bởi phần tử **ds:Reference**, nếu có).

6.4.4.3 Thuộc tính xlink:type

Thuộc tính **xlink:type** có một giá trị "simple" CỐ ĐỊNH. Điều này xác định phần tử đó đang liên kết [XLINK] đơn giản.

6.4.4.4 Thuộc tính **xlink:href**

Thuộc tính **xlink:href** YÊU CẦU PHẢI có một giá trị để bao gồm tài liệu *quy định quá trình* và là một URI để phù hợp với [RFC2396].

6.4.4.5 Thuộc tính **uuid**

Thuộc tính **uuid** MẶC NHIÊN xác định duy nhất **ProcessSpecification** (quy định quá trình). Nếu tài liệu *quy định quá trình* được xác định bởi quy định ebXML về *quá trình kinh doanh* [ebBPSS], thì thuộc tính này BẮT BUỘC được thiết lập **uuid** đó đối với phần tử **ProcessSpecification** (quy định quá trình) tương đương trong *quy định quá trình kinh doanh* cụ thể.

6.4.4.6 Phần tử **ds:Reference**

Phần tử **ds:Reference** xác định cùng tài liệu *quy định quá trình* như thuộc tính **xlink:href** của phần tử **ProcessSpecification** (quy định quá trình) kèm theo và phần tử bổ sung cung cấp cho việc kiểm tra tài liệu *quy định quá trình* không bị thay đổi khi CPP đã được tạo ra, thông qua việc sử dụng một phương pháp hệ thống và giá trị hệ thống như được mô tả dưới đây.

CHÚ THÍCH: Các *bên tham gia* có thể kiểm tra tính hợp lệ của CPP hoặc CPA tại bất kỳ thời điểm nào. Các kiểm tra tính hợp lệ sau đây có thể là sự quan tâm riêng:

- kiểm tra tính hợp lệ của một CPP và tài liệu *quy định quá trình* được tham chiếu tại thời điểm thành phần cấu tạo của một CPA bắt đầu trong trường hợp chúng thay đổi khi chúng được tạo ra;
- kiểm tra tính hợp lệ của một CPA và tài liệu *quy định quá trình* được tham chiếu tại thời điểm một CPA được cài đặt tới một hệ thống của *bên tham gia*;
- kiểm tra tính hợp lệ của một CPA tại các khoảng thời gian khi CPA đó được cài đặt vào trong một hệ thống của *bên tham gia*. CPA đó và tài liệu *quy định quá trình* được tham chiếu có thể được xử lý bởi một công cụ cài đặt vào trong một biểu mẫu phù hợp phần sụn cụ thể. Vì vậy, các biến đổi tới CPA đó và tài liệu *quy định quá trình* được tham chiếu không cần thiết ảnh hưởng lên các thao tác thời gian chạy. Các biến đổi như vậy có thể không được phát hiện đến khi cần phải cài đặt lại CPA đó và tài liệu *quy định quá trình* được tham chiếu.

Cú pháp và ngữ nghĩa của phần tử **ds:Reference** và các phần tử con của nó được xác định trong định danh tài liệu số XML[XMLDSIG].

Ngoài ra, để định danh tài liệu *quy định quá trình*, phần tử **ds:Reference** đầu tiên BẮT BUỘC bao gồm một thuộc tính **ds:URI** mà giá trị của nó đồng dạng với giá trị thuộc tính **xlink:href** trong phần tử **ProcessSpecification** (quy định quá trình) kèm theo.

Theo [XMLDSIG], một phần tử **ds:Reference** có thể có một phần tử con **ds:Transforms** lần lượt có một danh sách có thứ tự của một hoặc nhiều các phần tử con **ds:Transforms** để quy định một trình tự các phép biến đổi. Tuy nhiên, tiêu chuẩn XML hiện tại YÊU CẦU theo quy tắc truyền tiêu chuẩn [XMLC14N] và ngăn cấm các biến đổi khác. Vì vậy, các yêu cầu bổ sung sau đây áp dụng cho một phần tử **ds:Reference** trong một phần tử **ProcessSpecification** (quy định quá trình):

TCVN ISO/TS 15000-1 : 2007

- phần tử **ds:Reference** BẮT BUỘC có một phần tử con **ds:Transforms**;
- phần tử **ds:Transforms** BẮT BUỘC chính xác là có chính xác một phần tử con **ds:Transform**;
- phần tử **ds:Transform** BẮT BUỘC quy định theo quy tắc truyền XML tiêu chuẩn [XMLC14N] thông qua giá trị YÊU CẦU sau đây đối với thuộc tính **ds:Algorithm** YÊU CẦU của nó:
<http://www.w3.org/TR/2001/Rec-xml-c14n-20010315>.

Chú ý rằng sự thi hành XML theo quy tắc tiêu chuẩn ĐƯỢC YÊU CẦU bởi định danh tài liệu số XML[XMLDSIG].

Để cho phép việc kiểm tra tài liệu *quy định quá trình* được xác định và truyền không bị thay đổi, phần tử **ds:DigestMethod** quy định thuật toán hệ thống được áp dụng cho tài liệu *quy định quá trình* và phần tử **ds:DigestValue** quy định giá trị mong muốn. Tài liệu *quy định quá trình* được coi là không bị thay đổi nếu và chỉ nếu kết quả của việc áp dụng thuật toán hệ thống đối với các kết quả của tài liệu *quy định quá trình* nằm trong giá trị mong muốn.

Một phần tử **ds:Reference** trong một phần tử **ProcessSpecification** (quy định quá trình) ám chỉ tính hợp lệ của CPP:

- một CPP BẮT BUỘC phải xem xét hợp lệ nếu bất kỳ phần tử **ds:Reference** trong một phần tử **ProcessSpecification** (quy định quá trình) không có khả năng kiểm tra tính hợp lệ tham chiếu như được xác định bởi quy định chữ ký dạng số XML[XMLDSIG];
- một CPP BẮT BUỘC phải xem xét hợp lệ nếu bất kỳ phần tử **ds:Reference** trong nó không thể xóa được tham chiếu.

Các hàm ý về kiểm tra tính hợp lệ khác của phần tử **ds:Reference** như vậy được quy định trong phần mô tả phần tử **Signature** (chữ ký); phần 9.9.

CHÚ THÍCH: Quy định chữ ký dạng số XML[XMLDSIG] khẳng định "Trình ứng dụng về chữ ký có thể trả lời việc định danh (URI) và truyền được cung cấp bởi người ký trong phần tử **Reference** hoặc nó có thể đạt được nội dung thông qua các phương tiện khác như một bộ điều khiển cục bộ" (tầm quan trọng trên có thể thêm vào). Tuy nhiên, KHUYẾN CÁO là các thực hiện CPP/CPA ebXML không tạo ra việc sử dụng của các kết quả được lưu trữ như vậy khi ký hoặc kiểm tra tính hợp lệ.

CHÚ THÍCH: Nhận thấy rằng định danh tài liệu số XML[XMLDSIG] cung cấp cho việc ký một tài liệu XML cùng với các tài liệu tham chiếu bên ngoài. trong các trường hợp mà một tài liệu CPP hoặc CPA thực tế được ký phù hợp, phương tiện đó cũng có thể được sử dụng để đảm bảo rằng tài liệu *quy định quá trình* được tham chiếu là không thay đổi. Tuy nhiên, tiêu chuẩn hiện tại không bắt buộc một CPP hoặc CPA được ký.

CHÚ THÍCH: Nếu các *bên tham gia* một CPA mong muốn làm theo yêu cầu khách hàng về một tài liệu *quy định quá trình* tồn tại trước đó, Họ có thể sao chép tài liệu hiện tại, thay đổi nó và làm cho CPA của họ tham chiếu đến bản sao được sửa đổi. Nhận thấy rằng đối với các yêu cầu về tính rõ ràng, súc tích hoặc báo cáo tóm lược, các *bên tham gia* có thể để tham chiếu một tài liệu *quy định quá trình* tồn tại trước đó dưới dạng gốc của nó và phần

phụ để tham chiếu tới một quy định của các thay đổi đã thỏa thuận. Vì vậy, việc sử dụng *CPP* của phần tử phụ **ds:Transforms** của phần tử **ds:Reference** trong một phần tử **ProcessSpecification** (quy định quá trình) có thể được mở rộng trong tương lai để cho phép các biến đổi khác như được quy định trong quy định chữ ký dạng số XML[XMLDSIG]. Ví dụ, các sửa đổi đối với tài liệu gốc có thể sau đó được giải thích như các biến đổi XSLT. Sau khi áp dụng biến đổi nào đó, có thể cần thiết kiểm tra tính hợp lệ tài liệu được biến đổi đối với giản đồ quy định quá trình kinh doanh ebXML [ebBPSS].

6.4.5 Phần tử Role (Vai trò)

Phần tử **Role** (vai trò) YÊU CẦU xác định vai trò mà trong *quy định quá trình* của *bên tham gia* đó có khả năng hỗ trợ thông qua (các) phần tử **ServiceBinding** (quy định dịch vụ) cùng thuộc phần tử **CollaborationRole** (vai trò hợp tác) này.

Phần tử **Role** (vai trò) có các thuộc tính sau:

- một thuộc tính **name** (tên) YÊU CẦU;
- một thuộc tính **xlink:type** CỐ ĐỊNH;
- một thuộc tính **xlink:href** YÊU CẦU.

6.4.5.1 Thuộc tính name (Tên)

Thuộc tính **name** (tên) YÊU CẦU là một chuỗi để đưa ra một tên đối với vai trò. Giá trị của nó được lấy từ một thuộc tính **name** (tên) của một trong các phần tử **Role** (vai trò) của một **BinaryCollaboration** được mô tả trong *quy định quá trình* [ebBPSS].

Xem CHÚ THÍCH trong phần 8.4.4 về các mô tả *hồ sơ hợp tác kinh doanh* thay thế.

6.4.5.2 Thuộc tính xlink:type

Thuộc tính **xlink:type** có một giá trị "simple" CỐ ĐỊNH. Điều này xác định phần tử đó như là một liên kết [XLINK] đơn giản.

6.4.5.3 Thuộc tính xlink:href

Thuộc tính **xlink:href** YÊU CẦU PHẢI có một giá trị là một URI để phù hợp với [RFC2396]. Nó xác định thuộc tính hoặc vị trí của phần tử đó trong tài liệu *quy định quá trình* để định nghĩa vai trò trong nội dung của *hồ sơ hợp tác kinh doanh*. Ví dụ:

```
xlink:href="http://www.rosettanet.org/processes/3A4.xml#Buyer"
```

Ở đây, "người mua" là giá trị thuộc tính **ID** của phần tử đó trong tài liệu *quy định quá trình* để định nghĩa tên vai trò.

6.4.6 Phần tử ApplicationCertificateRef (Tham chiếu chứng chỉ của ứng dụng)

Phần tử **ApplicationCertificateRef** (tham chiếu chứng chỉ của ứng dụng), nếu có mặt, thì xác định một chứng chỉ để sử dụng bởi lớp ứng dụng/ *quá trình kinh doanh*. Chứng chỉ này không được sử dụng bởi

TCVN ISO/TS 15000-1 : 2007

hệ thống truyền *thông điện* đó, nhưng nó được chứa trong *CPP*, vì vậy nó có thể được xem xét trong quá trình thương lượng *CPA* đó. Phần tử ***ApplicationCertificateRef*** (tham chiếu chứng chỉ của ứng dụng) có thể xuất hiện không hoặc nhiều lần.

CHÚ THÍCH: Phần mềm ứng dụng trên cả hai vị trí của một hợp tác để xác định cách sử dụng dự kiến/được cho phép của một chứng chỉ ứng dụng bằng việc kiểm tra sự mở rộng sử dụng khóa trong chính chứng chỉ của nó.

CHÚ THÍCH: Phần tử này được chứa trong *CPP/CPA* để hỗ trợ khả năng hoạt động tương tác với các hệ thống pháp lý thực sự đã thực hiện các chức năng mật mã hóa như chữ ký dạng số hoặc mật mã hóa. Những người thực hiện nên hiểu rằng việc sử dụng của ***ApplicationCertificateRef*** chỉ cần thiết trong các trường hợp khi khả năng hoạt động tương tác với các hệ thống pháp lý như vậy được yêu cầu.

Phần tử ***ApplicationCertificateRef*** (tham chiếu chứng chỉ của ứng dụng) có:

- một thuộc tính ***certId*** (id của chứng chỉ) YÊU CẦU.

6.4.6.1 Thuộc tính ***certId*** (id của chứng chỉ)

Thuộc tính ***certId*** (id của chứng chỉ) YÊU CẦU là một IDREF [XML] để liên kết phần tử ***CollaborationRole*** (vai trò hợp tác) với một chứng chỉ. Nó **BẮT BUỘC** có một giá trị bằng giá trị thuộc tính ***certId*** (id của chứng chỉ) của một trong các phần tử ***Certificate*** (chứng chỉ) thuộc phần tử ***PartyInfo*** (thông tin bên tham gia).

6.4.7 Phần tử ***ApplicationSecurityDetailsRef*** (tham chiếu chi tiết an ninh của chứng chỉ)

Phần tử ***ApplicationSecurityDetailsRef*** (tham chiếu chi tiết an ninh của chứng chỉ), nếu có mặt, thì xác định các mẫu neo chứng thực và chính sách an ninh *bên tham gia* này sẽ áp dụng cho bất kỳ chứng chỉ lớp-ứng dụng được đề nghị bởi *bên tham gia* khác. Các mẫu neo chứng thực và chính sách này không được sử dụng bởi hệ thống truyền *thông điện*, nhưng được chứa trong *CPP* đó, vì vậy chúng có thể được xem xét trong quá trình thương lượng *CPA* đó.

Phần tử ***ApplicationSecurityDetailsRef*** (tham chiếu chi tiết an ninh của chứng chỉ) có một thuộc tính ***securityId*** (id an ninh) YÊU CẦU.

6.4.7.1 Thuộc tính ***securityId*** (id an ninh)

Thuộc tính ***securityId*** (id an ninh) YÊU CẦU là một IDREF [XML] để liên kết phần tử ***CollaborationRole*** (vai trò hợp tác) với một phần tử ***SecurityDetails*** (chi tiết an ninh) quy định một tập các mẫu neo chứng thực và một chính sách an ninh. Nó **BẮT BUỘC** có một giá trị bằng với giá trị thuộc tính ***securityId*** (id an ninh) của một trong các phần tử ***SecurityDetails*** (chi tiết an ninh) thuộc phần tử ***PartyInfo*** (thông tin bên tham gia) đó.

6.4.8 Phần tử ***ServiceBinding*** (Quy định dịch vụ)

Phần tử ***ServiceBinding*** (quy định dịch vụ) xác định một phần tử ***DeliveryChannel*** (kênh truyền) đối với tất cả lưu lượng *thông điện kinh doanh* đó là để được gửi hoặc nhận bởi *bên tham gia* đó trong nội

dung của tài liệu *quy định quá trình* được xác định. Nó **BẮT BUỘC** bao gồm ít nhất một phần tử con **CanReceive** (có thể nhận) hoặc **CanSend** (có thể gửi).

Phần tử **ServiceBinding** (quy định dịch vụ) có một phần tử con **Service** (dịch vụ), không hoặc nhiều phần tử con **CanSend** (có thể gửi) và không hoặc nhiều phần tử con **CanReceive** (có thể nhận).

6.4.9 Phần tử Service (Dịch vụ)

Giá trị của phần tử **Service** (dịch vụ) là một chuỗi **PHẢI** được sử dụng như giá trị của phần tử **Service** (dịch vụ) trong *tiêu đề thông điệp ebXML[ebMS]* hoặc một phần tử tương tự trong *tiêu đề thông điệp* đó của một dịch vụ *thông điệp* thay thế. Phần tử **Service** (dịch vụ) có một thuộc tính **type** (kiểu) **MẶC NHIÊN**.

Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], thì giá trị của phần tử **Service** (dịch vụ) **BẮT BUỘC** là thuộc tính **uuid** (URI) được quy định đối với phần tử **ProcessSpecification** (quy định quá trình) trong tài liệu quy định giản đồ *quy định quá trình kinh doanh*.

CHÚ THÍCH: Mục đích của phần tử **Service** (dịch vụ) là để cung cấp thông tin về lộ trình đối với *tiêu đề thông điệp ebXML*. Phần tử **CollaborationRole** (vai trò hợp tác) và các phần tử con của nó định danh thông tin trong tài liệu **ProcessSpecification** có liên quan đến *CPP* hoặc *CPA*. Phần tử **Service** (dịch vụ) có thể được sử dụng cùng với phần tử **CanReceive** và **CanSend** (và các dẫn xuất của chúng) để cung cấp định tuyến của các *thông điệp* nhận được tới mục nhập đúng của ứng dụng.

6.4.9.1 Thuộc tính type (Kiểu)

Nếu thuộc tính **type** (kiểu) có mặt, thì nó chỉ ra nhận biết việc gửi và nhận *thông điệp* của các *bên tham gia* bằng một số phương tiện khác như cách phiên dịch giá trị của phần tử **Service** (dịch vụ). Cả hai *bên tham gia* có thể sử dụng giá trị thuộc tính **type** (kiểu) để trợ giúp việc phiên dịch đó.

Nếu thuộc tính **type** (kiểu) không có mặt, thì giá trị của phần tử **Service** (dịch vụ) **BẮT BUỘC** là một URI [RFC2396]. Nếu sử dụng quy định ebXML về *quá trình kinh doanh* [ebBPSS] để định nghĩa tài liệu *quy định quá trình*, thì thuộc tính **type** (kiểu) **BẮT BUỘC** đó là một URI[RFC2396].

6.4.10 Phần tử CanSend (Có thể gửi)

Phần tử **CanSend** (có thể gửi) xác định một thông điệp **Action** (hành động) để một *bên tham gia* có khả năng gửi. Phần tử **CanSend** (có thể gửi) có ba phần tử con: **ThisPartyActionBinding** (quy định hoạt động bên tham gia này), **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) và **CanReceive** (có thể nhận). Phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) **ĐƯỢC YÊU CẦU** đối với cả *CPP* và *CPA*.

Nó xác định **DeliveryChannel** (kênh truyền) và **Packaging** (đóng gói) của *bên tham gia* đó được mô tả bởi việc bao gồm phần tử **PartyInfo** (thông tin bên tham gia) sẽ sử dụng đối với việc gửi *thông điệp* dẫn

TCVN ISO/TS 15000-1 : 2007

chứng hành động theo truy vấn. Phần tử **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) chỉ được sử dụng trong trường hợp các CPA. Trong một CPA và thuộc cùng phần tử **CanSend**, **DeliveryChannels** và **Packaging** được mong đợi/sử dụng bởi hai bên tham gia BẮT BUỘC tương thích. Phần tử **CanReceive** (có thể nhận) có thể xuất hiện không hoặc nhiều lần. Nếu có mặt, thì nó chỉ ra rằng một hoặc nhiều hành động phản hồi đồng thời được mong đợi.

Điều này được minh họa trong các ví dụ CPP và CPA ở các phụ lục.

CHÚ THÍCH: Trong khi giản đồ cho phép các mức lồng nhau tùy ý thuộc phần tử **CanSend** (có thể gửi), sử dụng các trường hợp đối với việc lồng nhau ngoài phạm vi hai mức không được trình bày. Hai mức có thể cần thiết đối với một yêu cầu với một phản hồi lại đồng bộ được quy định thêm một xác thực trở lại đồng bộ đối với phản hồi đó.

6.4.11 Phần tử CanReceive (Có thể nhận)

Phần tử **CanReceive** (có thể nhận) xác định một *thông điệp* dẫn chứng hành động mà một bên tham gia có khả năng nhận. Nó có ba phần tử con: **ThisPartyActionBinding** (quy định hoạt động bên tham gia này), **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) và **CanSend** (có thể gửi). Phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) ĐƯỢC YÊU CẦU đối với cả CPP và CPA. Nó xác định **DeliveryChannel** (kênh truyền) bên tham gia đó được mô tả bởi bao gồm phần tử **PartyInfo** (thông tin bên tham gia) sẽ sử dụng đối với việc nhận thông điệp **Action** (hành động) theo truy vấn và phần tử **Packaging** (đóng gói) đang mong đợi. Phần tử **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) chỉ được sử dụng trong trường hợp CPA. trong một CPA và cùng phần tử **CanReceive**, **DeliveryChannel** và **Packaging** được mong đợi/sử dụng bởi hai bên tham gia BẮT BUỘC tương thích. Phần tử **CanSend** (có thể gửi) có thể xuất hiện không hoặc nhiều lần. Nếu có mặt, thì nó chỉ ra rằng một hoặc nhiều hành động phản hồi đồng thời được mong đợi. Điều này được minh họa trong các ví dụ CPP và CPA trong các phụ lục.

CHÚ THÍCH: Trong khi giản đồ đó cho phép các mức lồng nhau tùy ý trong phần tử **CanReceive**, các trường hợp sử dụng đối với việc lồng nhau ngoài phạm vi hai mức chưa được trình bày. Hai mức có thể cần thiết đối với một yêu cầu tới một phản hồi lại đồng bộ để được quy định thêm một xác thực trở lại đồng bộ đối với phản hồi đó.

6.4.12 Phần tử ThisPartyActionBinding (Quy định hoạt động bên tham gia này)

Phần tử **ThisPartyActionBinding** quy định một hoặc nhiều phần tử **DeliveryChannel** (kênh truyền) đối với các *thông điệp* cho một hành động được chọn lựa và việc đóng gói đối với các *thông điệp* đó là để được gửi hoặc nhận bởi bên tham gia đó theo nội dung của *quy định quá trình* đó được kết hợp với phần tử **ServiceBinding** (quy định dịch vụ) gốc.

Phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) có một phần tử con **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) YÊU CẦU, không hoặc một phần tử con **ActionContext** (nội dung hành động) và một hoặc nhiều phần tử con **ChannelID**.

Phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) có thuộc tính sau:

- một thuộc tính **action** (hành động) YÊU CẦU;
- một thuộc tính **packageld** (id của gói) YÊU CẦU;
- một thuộc tính **xlink:href** MẶC NHIÊN;
- một thuộc tính **xlink:type** CỐ ĐỊNH.

Thuộc một phần tử **ServiceBinding** (quy định dịch vụ) cho trước, có thể có nhiều phần tử con **CanSend** (có thể gửi) hoặc **CanReceive** (có thể nhận) với cùng hành động để cho phép các điểm nhập phần mềm khác nhau và các lựa chọn truyền. trong một kịch bản như vậy, **DeliveryChannel** (kênh truyền) được đề cập bởi các phần tử con **ChannelID** (id kênh truyền) của phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) PHẢI chỉ ra các **EndPoint** (điểm đầu cuối) riêng biệt đối với việc nhận MSH để định danh một cách duy nhất phần tử **DeliveryChannel** (kênh truyền) đang được sử dụng cho trao đổi *thông điệp* cụ thể này.

CHÚ THÍCH: Một thực hiện có thể cung cấp khả năng của việc ấn định động các kênh truyền trên mỗi *thông điệp* dựa trên cơ sở khoảng thời gian thực hiện của **BinaryCollaboration**. Kênh truyền được chọn sẽ được lựa chọn, dựa trên cơ sở các điều kiện hiện tại, từ những cái đó được xác định bởi các Phần tử **CanSend** (có thể gửi) đề cập tới **BinaryCollaboration** đó là gửi *thông điệp*. Tại vị trí *bên nhận*, MSH có thể sử dụng các **EndPoint** (điểm đầu cuối) riêng biệt để định danh phần tử **DeliveryChannel** (kênh truyền) được sử dụng cho việc trao đổi *thông điệp* này.

Trong một phần tử **CanSend** (có thể gửi) hoặc một phần tử **CanReceive** (có thể nhận), khi cả hai Phần tử **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) và **ThisPartyActionBinding** có mặt (như là., trong một *CPA*), chúng BẮT BUỘC có các giá trị hành động đồng dạng hoặc tương đương phần tử **ActionContext** (nội dung hành động). Ngoài ra, **DeliveryChannel** và đóng gói đó để chúng tham chiếu BẮT BUỘC tương thích.

6.4.12.1 Thuộc tính action (Hành động)

Giá trị thuộc tính **action** (hành động) YÊU CẦU là một chuỗi để xác định việc trao đổi tài liệu *kinh doanh* được kết hợp với **DeliveryChannel** (kênh truyền) và xác định bởi phần tử con **ChannelID**.

Giá trị thuộc tính **action** (hành động) PHẢI được sử dụng như giá trị của phần tử **Action** (hành động) trong *tiêu đề thông điệp ebXML[ebMS]* hoặc một phần tử tương tự trong *tiêu đề thông điệp* của một dịch vụ *thông điệp* thay thế. Mục đích của thuộc tính **action** (hành động) là để cung cấp một ánh xạ giữa cách đặt tên theo thứ tự được kết hợp với một *quá trình kinh doanh* ứng dụng và phần tử **Action** (hành động) trong *tiêu đề thông điệp ebXML[ebMS]*. Việc ánh xạ này có thể thực hiện bằng việc sử dụng phần tử **ActionContext** (nội dung hành động). Xem chú thích trong phần 8.4.4 về các mô tả *hồ sơ hợp tác kinh doanh* thay thế.

TCVN ISO/TS 15000-1 : 2007

Các tín hiệu kinh doanh, khi được gửi riêng (nghĩa là; không được gói cùng với các tài liệu phản hồi theo phương thức trả lời đồng bộ) PHẢI sử dụng các giá trị *ReceiptAcknowledgment* (báo việc nhận), *AcceptanceAcknowledgment* (báo chấp nhận) hoặc *Exception* (ngoại lệ) như giá trị thuộc tính **action** (hành động) của chúng. Ngoài ra, chúng NÊN quy định dịch vụ giống như dịch vụ được sử dụng đối với *thông điệp gốc*.

CHÚ THÍCH: Nói chung, tên hành động được chọn bởi hai *bên tham gia* để biểu diễn một yêu cầu hoạt động kinh doanh cụ thể hoặc phản hồi hoạt động kinh doanh trong nội dung của một **BinaryCollaboration** để tạo ra việc sử dụng của các **BinaryCollaboration** lồng nhau có thể không đồng dạng. Vì vậy, khi soạn hai *CPP* để tạo một biểu mẫu *CPA*, cần thiết để tạo việc sử dụng của thông tin từ **ActionContext** kèm theo (xem phần 8.4.16) để xác định hai tên hành động khác nhau từ hai *CPP* biểu diễn thực cùng **ActionContext**. Khi các *giao dịch kinh doanh* không được sử dụng trong các ngữ cảnh khác nhau, khuyến cáo rằng các tên của yêu cầu hoạt động kinh doanh và phản hồi hoạt động kinh doanh được sử dụng cùng tên hoạt động.

6.4.12.2 Thuộc tính **packaged** (id của gói)

Thuộc tính **packaged** (id của gói) YÊU CẦU là một IDREF [XML] để xác định phần tử **Packaging** (đóng gói) được kết hợp với *thông điệp* được xác định bởi thuộc tính **action** (hành động).

6.4.12.3 Thuộc tính **xlink:href**

Thuộc tính **xlink:href** MẶC NHIÊN, nếu có mặt, thì PHẢI cung cấp một biểu thức URI [XPOINTER] tuyệt đối để xác định cụ thể phần tử **RequestingBusinessActivity** (yêu cầu hoạt động kinh doanh) hoặc **RespondingBusinessActivity** (phản hồi hoạt động kinh doanh) trong tài liệu *quy định quá trình* [ebBPSS] kèm theo được xác định bởi phần tử **ProcessSpecification** (quy định quá trình).

6.4.12.4 Thuộc tính **xlink:type**

Thuộc tính **xlink:type** MẶC NHIÊN có một giá trị "simple" CỐ ĐỊNH. Điều này xác định phần tử đó như một liên kết [XLINK] đơn giản.

6.4.13 Phần tử **OtherPartyActionBinding** (Quy định hoạt động bên tham gia khác)

Phần tử **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) chỉ được sử dụng trong các *CPA*. Nó thuộc kiểu IDREF và xác định một phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) phù hợp đó thuộc phần tử **PartyInfo** (thông tin bên tham gia) của bên tham gia hợp tác. Nó xác định gián tiếp **DeliveryChannel** của *bên tham gia* khác sẽ sử dụng đối với việc gửi hoặc nhận thông điệp *action* (hành động) theo truy vấn và **Packaging** mong đợi. Trong một *CPA* và thuộc cùng phần tử **CanReceive** (có thể nhận) hoặc **CanSend** (có thể gửi), **DeliveryChannel** (kênh truyền) và **Packaging** (đóng gói) được mong đợi/ sử dụng bởi cả hai *bên tham gia*, như được chỉ ra bởi phần tử **OtherPartyActionBinding** (quy định hoạt động bên tham gia khác) và **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) BẮT BUỘC tương thích.

6.4.14 Phần tử **BusinessTransactionCharacteristics** (Đặc điểm giao dịch kinh doanh)

Phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) mô tả các đặc điểm an ninh và các thuộc tính khác của kênh truyền, như được tạo ra từ (các) **ProcessSpecification** của các thông điệp được truyền có sử dụng kênh truyền đó. Các thuộc tính của phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh), có thể được sử dụng để ghi đè lên các giá trị của các thuộc tính tương ứng trong tài liệu *quy định quá trình*.

Xem CHÚ THÍCH trong phần 8.4.4 đề cập tới các mô tả *hồ sơ hợp tác kinh doanh* thay thế.

CPP và các công cụ cấu tạo CPA và các công cụ phát triển CPA PHẢI kiểm tra các định nghĩa về kênh truyền đối với bên gửi và bên nhận (truyền và trao đổi tài liệu) đối với tính nhất quán bên trong cũng như khả năng giữa hai đối tác tham gia. Điển hình, khi một thuộc tính có một giá trị riêng biệt, thì phần tử con tương ứng thuộc các phần tử **Transport** (truyền tải) và phần tử **DocExchanges** (trao đổi tài liệu) sẽ tồn tại để mô tả kỹ hơn các tham số thực hiện được đề cập.

Phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) có các thuộc tính sau:

- một thuộc tính **isNonRepudiationRequired** (yêu cầu không từ chối) MẶC NHIÊN;
- một thuộc tính **isNonRepudiationReceiptRequired** (yêu cầu không từ chối nhận) MẶC NHIÊN;
- một thuộc tính **isConfidential** (bảo mật) MẶC NHIÊN;
- một thuộc tính **isAuthenticated** (được xác thực) MẶC NHIÊN;
- một thuộc tính **isAuthorizationRequired** (yêu cầu được xác thực) MẶC NHIÊN;
- một thuộc tính **isTamperProof** (chứng cứ can thiệp) MẶC NHIÊN;
- một thuộc tính **isIntelligibleCheckRequired** (yêu cầu kiểm tra khả năng hiểu được) MẶC NHIÊN;
- một thuộc tính **timeToAcknowledgeReceipt** (thời gian thông báo việc nhận) MẶC NHIÊN;
- một thuộc tính **timeToAcknowledgeAcceptance** (thời gian công nhận việc nhận) MẶC NHIÊN;
- một thuộc tính **timeToPerform** (thời gian thực hiện) MẶC NHIÊN;
- một thuộc tính **retryCount** (đếm lần thử lại) MẶC NHIÊN.

Các thuộc tính này cho phép các tham số được quy định tại mức *quy định-quá trình* được ghi đè. Nếu một trong các thuộc tính này không được quy định, thì giá trị mặc định tương ứng nên đạt được từ tài liệu *quy định quá trình*.

6.4.14.1 Thuộc tính **isNonRepudiationRequired** (Yêu cầu không từ chối)

Thuộc tính **isNonRepudiationRequired** (yêu cầu không từ chối) là một giá trị nhị phân với các giá trị có thể là "true" và "false". Nếu giá trị đó là "true" thì kênh truyền BẮT BUỘC quy định *thông điệp* đó là được ký dạng số có sử dụng chứng chỉ của *bên tham gia* gửi *thông điệp* và đạt được bởi cả hai *bên*

tham gia. Phần tử **SenderNonRepudiation** (không từ chối bên gửi) thuộc **DocExchange/ebXMLSenderBinding** (xem phần 8.4.43) và phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) thuộc **DocExchange/ebXMLReceiverBinding** (xem phần 0) mô tả kỹ hơn các tham số biến đổi liên quan đến việc thực hiện dịch vụ không từ chối gốc, như thuật toán băm, thuật toán ký, việc ký chứng chỉ, mấu neo chứng thực, v..v.

6.4.14.2 Thuộc tính **isNonRepudiationReceiptRequired** (Yêu cầu không từ chối nhận)

Thuộc tính **isNonRepudiationReceiptRequired** (yêu cầu không từ chối nhận) là một giá trị nhị phân với các giá trị có thể là "true" và "false". Nếu giá trị đó là "true" thì kênh truyền BẮT BUỘC quy định rằng *thông điệp* đó là để được xác thực bởi một *thông điệp* tín hiệu xác thực việc nhận được ký dạng số, có sử dụng chứng chỉ được ký đó của *bên tham gia* đó để nhận *thông điệp* đó, bao gồm (các) hệ thống tài liệu của thiết bị mang theo của *thông điệp* được chấp nhận. Phần tử **SenderNonRepudiation** (không từ chối bên gửi) thuộc phần tử **DocExchange/ebXMLSenderBinding** (xem phần 8.4.43) và **ReceiverNonRepudiation** thuộc **DocExchange/ebXMLReceiverBinding** (xem phần 0) mô tả kỹ hơn các tham số biến đổi liên quan đến việc thực hiện dịch vụ không từ chối việc nhận.

6.4.14.3 Thuộc tính **isConfidential** (Bảo mật)

Thuộc tính **isConfidential** (bảo mật) có các giá trị có thể là "none", "transient", "persistent" và "transient-and-persistent". Các giá trị này BẮT BUỘC được phiên dịch như được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS]. Nói chung, tính bảo mật tạm thời có thể được thực hiện nhờ việc sử dụng một giao thức truyền an toàn như SSL; tính bảo mật liên tục có thể được thực hiện có sử dụng một cơ chế đường bao số như S/MIME. Thông tin truyền an toàn được cung cấp chi tiết hơn trong các phần tử **TransportSender** (truyền tải bên gửi) (xem phần 8.4.25) và **TransportReceiver** (truyền tải bên nhận) (xem phần 8.4.32) thuộc phần tử **Transport** (truyền tải). Thông tin việc mã hóa lâu dài được cung cấp chi tiết hơn trong phần tử **SenderDigitalEnvelope** (đường bao số bên gửi) thuộc **DocExchange/ebXMLSenderBinding** (xem phần 8.4.48) và phần tử **ReceiverDigitalEnvelope** (đường bao số của bên nhận) thuộc **DocExchange/ebXMLReceiverBinding** (xem phần 8.4.56).

6.4.14.4 Thuộc tính **isAuthenticated** (Được xác thực)

Thuộc tính **isAuthenticated** (được xác thực) có các giá trị có thể là "none", "transient", "persistent" và "persistent-and-transient". Nếu thuộc tính này được thiết lập cho bất kỳ giá trị khác "none", thì *bên nhận* BẮT BUỘC có khả năng kiểm tra định danh của *bên gửi*.

Nói chung, xác thực tạm thời có thể được thực hiện có sử dụng một giao thức truyền an toàn như SSL (cùng với hoặc ngoài việc sử dụng của xác thực hệ thống liệt kê hoặc cơ sở); xác thực lâu dài có thể được thực hiện có sử dụng một cơ chế về chữ ký dạng số. Thông tin truyền an toàn được cung cấp chi tiết hơn trong các phần tử **TransportSender** (truyền tải bên gửi) (xem phần 8.4.25) và **TransportReceiver** (xem phần 8.4.33) thuộc phần tử **Transport**. Thông tin về xác thực lâu dài được cung cấp chi tiết hơn trong phần tử **SenderNonRepudiation** (không từ chối bên gửi) thuộc

DocExchange/ebXMLSenderBinding (xem phần 8.4.43) và phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) thuộc phần tử **DocExchange/ebXMLReceiverBinding** (xem phần 0).

6.4.14.5 Thuộc tính **isAuthorizationRequired** (Yêu cầu được xác thực)

Thuộc tính **isAuthorizationRequired** (yêu cầu được xác thực) là một giá trị nhị phân với các giá trị có thể là "true" và "false". Nếu giá trị đó là "true" thì nó chỉ ra rằng kênh truyền BẮT BUỘC quy định rằng *bên gửi thông điệp* là để được xác thực trước khi truyền tới ứng dụng.

6.4.14.6 Thuộc tính **isTamperProof** (Chứng cứ can thiệp)

Thuộc tính **isTamperProof** (chứng cứ can thiệp) có các giá trị có thể là "none", "transient", "persistent" và "persistent-and-transient". Nếu thuộc tính này được thiết lập cho một giá trị khác "none", thì nó phải có thể được *bên nhận* phát hiện *thông điệp* nhận được đã bị sửa đổi hoặc làm giả chưa. Nói chung, việc xóa giả mạo tạm thời có thể được thực hiện có sử dụng một truyền an toàn như SSL; việc xóa giả mạo lâu dài có thể được thực hiện có sử dụng một cơ chế về chữ ký dạng số. Thông tin truyền an toàn được cung cấp chi tiết hơn trong các phần tử **TransportSender** (truyền tải bên gửi) (xem phần 8.4.25) và **TransportReceiver** (xem phần 8.4.48) thuộc phần tử **Transport**. Thông tin về chữ ký dạng số được cung cấp chi tiết hơn trong phần tử **SenderNonRepudiation** (không từ chối bên gửi) thuộc **DocExchange/ebXMLSenderBinding** (xem phần 8.4.43) và phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) thuộc **DocExchange/ebXMLReceiverBinding** (xem phần 0).

6.4.14.7 Thuộc tính **isIntelligibleCheckRequired** (Yêu cầu kiểm tra khả năng hiểu được)

Thuộc tính **isIntelligibleCheckRequired** (yêu cầu kiểm tra khả năng hiểu được) là một giá trị nhị phân với các giá trị có thể là "true" và "false". Nếu giá trị đó là "true", thì *bên nhận* BẮT BUỘC kiểm tra rằng một tài liệu *kinh doanh* không bị làm sai lạc (như là., vượt qua việc kiểm tra tính hợp lệ của giản đồ) trước khi truyền lại một tín hiệu xác thực việc nhận.

6.4.14.8 Thuộc tính **timeToAcknowledgeReceipt** (Thời gian thông báo việc nhận)

Thuộc tính **timeToAcknowledgeReceipt** (thời gian thông báo việc nhận) là thuộc kiểu khoảng thời gian [XMLSCHEMA-2]. Nó quy định khoảng thời gian trong đó việc nhận của *bên tham gia* phải xác thực việc nhận của một tài liệu *kinh doanh*.

Nếu thuộc tính này được quy định, thì tín hiệu xác thực việc nhận BẮT BUỘC được sử dụng.

6.4.14.9 Thuộc tính **timeToAcknowledgeAcceptance** (Thời gian công nhận việc nhận)

Thuộc tính **timeToAcknowledgeAcceptance** (thời gian công nhận việc nhận) kiểu khoảng thời gian [XMLSCHEMA-2]. Nó quy định khoảng thời gian trong đó *bên nhận* phải thừa nhận tạm thời sự chấp nhận của một tài liệu *kinh doanh* (như là, sau khi nó qua bước kiểm tra tính hợp lệ các quy tắc *kinh doanh*).

Nếu thuộc tính này được quy định, thì tín hiệu xác thực việc chấp nhận BẮT BUỘC được sử dụng.

6.4.14.10 Thuộc tính **timeToPerform** (Thời gian thực hiện)

TCVN ISO/TS 15000-1 : 2007

Thuộc tính **timeToPerform** (thời gian thực hiện) kiểu khoảng thời gian [XMLSCHEMA-2]. Nó quy định khoảng thời gian, bắt đầu từ việc khởi tạo của **RequestingBusinessActivity** (yêu cầu hoạt động kinh doanh), trong đó người khởi tạo giao dịch BẮT BUỘC phải nhận phản hồi đó, nghĩa là, tài liệu *kinh doanh* được kết hợp với **RespondingBusinessActivity** (yêu cầu hoạt động kinh doanh).

CHÚ THÍCH: Thuộc tính **timeToPerform** (thời gian thực hiện) được kết hợp với một **BinaryCollaboration** trong BPSS hiện tại không được lập mô hình trong tiêu chuẩn này. Vì vậy, nó không thể bị ghi đè. Nói cách khác, giá trị được quy định tại mức BPSS BẮT BUỘC được sử dụng.

Khi sử dụng phương thức trả lời đồng bộ (xem phần 8.4.23.1), giá trị **TimeToPerform** (thời gian thực hiện) NÊN được sử dụng như thời gian chờ kết nối.

6.4.14.11 Thuộc tính **retryCount** (Đếm lần thử lại)

Thuộc tính **retryCount** (đếm lần thử lại) là kiểu số tự nhiên. Nó quy định số lần lớn nhất *giao dịch kinh doanh* có thể được thử lại trong điều kiện lỗi nào đó (như là; đợi trong thời gian chờ đối với Tín hiệu xác thực việc nhận) thực thi của nó trong khoảng thời gian phát sinh. Các thử lại như vậy BẮT BUỘC không được sử dụng khi truyền thông điệp xác thực ebXML được thực hiện để truyền các *thông điệp* trong Giao dịch *kinh doanh*. trong trường hợp sau, các lần thử lại được quản lý bởi các phần tử **Retry** (thử lại), **RetryInterval** (khoảng thời gian giữa các lần thử lại) thuộc phần tử **ReliableMessaging** (truyền thông điệp tin cậy).

6.4.15 Phần tử **ChannelId** (id kênh truyền)

Phần tử **ChannelId** (id kênh truyền) xác định một hoặc nhiều phần tử **DeliveryChannel** (kênh truyền) mà có thể được sử dụng đối với việc gửi hoặc nhận thông điệp *action* tương ứng. Nhiều phần tử **ChannelId** có thể được sử dụng để liên kết các phần tử **DeliveryChannel** (kênh truyền) với các đặc điểm khác nhau với cùng phần tử **CanReceive** (có thể nhận) hoặc **CanSend** (có thể gửi). Ví dụ, một *bên tham gia* để hỗ trợ cả hai HTTP và SMTP đối với việc gửi cùng hành động có thể quy định các giá trị thuộc tính **ChannelId** (id kênh truyền) khác nhau đối với các kênh tương ứng. Nếu sử dụng nhiều phần tử **DeliveryChannel** (kênh truyền), thì các phần tử **Endpoint** (điểm cuối) khác nhau BẮT BUỘC được sử dụng, vì vậy việc nhận MSH có thể xác định duy nhất phần tử **DeliveryChannel** (kênh truyền) được sử dụng cho việc trao đổi thông điệp này.

6.4.16 Phần tử **ActionContext** (Nội dung hành động)

Phần tử **ActionContext** (nội dung hành động) cung cấp một ánh xạ từ thuộc tính **action** (hành động) trong phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) cho kế hoạch đặt tên quy định-thực hiện *quá trình kinh doanh* tương ứng, nếu có. Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], thì phần tử **ActionContext** (nội dung hành động) BẮT BUỘC có mặt.

Mọi thực hiện ứng dụng/quá trình kinh doanh có thể sử dụng một kết hợp thông tin trong thuộc tính **action** (hành động) và phần tử **ActionContext** (nội dung hành động) để tạo các quyết định định tuyến *thông điệp*. Nếu có sử dụng các giản đồ mô tả *hồ sơ hợp tác kinh doanh* thay thế, thuộc tính **action** (hành động) của phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) gốc và/hoặc phần tử **wildcard** (đại diện) [XMLSCHEMA-1] trong phần tử **ActionContext** (nội dung hành động) có thể được sử dụng để tạo các quyết định định tuyến trên mức của trình quản lý dịch vụ thông điệp.

Phần tử **ActionContext** (nội dung hành động) có các phần tử sau đây:

- không hoặc một phần tử **CollaborationActivity** (hoạt động hợp tác);
- không hoặc nhiều phần tử **wildcard** (đại diện) [XMLSCHEMA-1].

Phần tử **ActionContext** (nội dung hành động) cũng có các thuộc tính sau:

- một thuộc tính **binaryCollaboration** (hợp tác nhị phân) YÊU CẦU;
- một thuộc tính **businessTransactionActivity** (hoạt động giao dịch kinh doanh) YÊU CẦU ;
- một thuộc tính **requestOrResponseAction** (hành động yêu cầu hoặc phản hồi) yêu cầu.

6.4.16.1 Thuộc tính **binaryCollaboration** (Hợp tác nhị phân)

Thuộc tính **binaryCollaboration** (hợp tác nhị phân) YÊU CẦU là một chuỗi để xác định **BinaryCollaboration** (hợp tác nhị phân) đối với **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) gốc được xác định. Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], thì giá trị thuộc tính **binaryCollaboration** (hợp tác nhị phân) BẮT BUỘC phù hợp giá trị thuộc tính **name** (tên) của phần tử **BinaryCollaboration** (hợp tác nhị phân) như được định nghĩa trong giản đồ quy định quá trình kinh doanh ebXML [ebBPSS].

6.4.16.2 Thuộc tính **businessTransactionActivity** (Hoạt động giao dịch kinh doanh)

Thuộc tính **businessTransactionActivity** (hoạt động giao dịch kinh doanh) YÊU CẦU là một chuỗi để xác định *giao dịch kinh doanh* đối với **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) gốc được xác định. Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], thì giá trị thuộc tính **businessTransactionActivity** (hoạt động giao dịch kinh doanh) BẮT BUỘC phù hợp giá trị thuộc tính **name** (tên) của phần tử **BusinessTransactionActivity** (hoạt động giao dịch kinh doanh) gốc của nó là **BinaryCollaboration** được đề cập bởi thuộc tính **binaryCollaboration** (hợp tác nhị phân).

6.4.16.3 Thuộc tính **requestOrResponseAction** (Hành động yêu cầu hoặc phản hồi)

Thuộc tính **requestOrResponseAction** (hành động yêu cầu hoặc phản hồi) yêu cầu là một chuỗi ký tự xác định cả yêu cầu và phản hồi hoạt động kinh doanh đối với **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) gốc được xác định. Đối với một **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) xác định đối với *bên yêu cầu* của một trao đổi *thông điệp*. Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS]. Thì giá trị thuộc tính

requestOrResponseAction (hành động yêu cầu hoặc phản hồi) BẮT BUỘC phù hợp giá trị thuộc tính **name** (tên) của phần tử **RequestingBusinessActivity** (yêu cầu hoạt động kinh doanh) tương ứng với *giao dịch kinh doanh* được quy định trong thuộc tính **businessTransactionActivity** (hoạt động giao dịch kinh doanh). Tương tự, đối với *bên phản hồi* của một trao đổi *thông điệp*, giá trị thuộc tính **requestOrResponseAction** (hành động yêu cầu hoặc phản hồi) BẮT BUỘC phù hợp giá trị thuộc tính **name** (tên) của phần tử **RespondingBusinessActivity** (phản hồi hoạt động kinh doanh) tương ứng với *giao dịch kinh doanh* được quy định trong thuộc tính **businessTransactionActivity** (hoạt động giao dịch kinh doanh), như được định nghĩa trong giản đồ quy định quá trình kinh doanh ebXML [ebBPSS].

6.4.17 Phần tử **CollaborationActivity** (Hoạt động hợp tác)

Phần tử **CollaborationActivity** (hoạt động hợp tác) hỗ trợ phần tử **ActionContext** (nội dung hành động) bằng việc cung cấp khả năng ánh xạ toàn bộ **BinaryCollaboration** lồng nhau như được định nghĩa trong giản đồ quy định quá trình kinh doanh ebXML [ebBPSS] tới thuộc tính **action** (hành động). Phần tử **CollaborationActivity** (hoạt động hợp tác) BẮT BUỘC có mặt khi **BinaryCollaboration** đề cập bởi thuộc tính **binaryCollaboration** (hợp tác nhị phân) có một **CollaborationActivity** (hoạt động hợp tác) xác định trong định nghĩa *quá trình kinh doanh*.

Một ví dụ về phần tử **CollaborationActivity** (hoạt động hợp tác) là:

```
<tp:CollaborationActivity
  tp:name="Credit Check"/>
```

Phần tử **CollaborationActivity** (hoạt động hợp tác) có không hoặc một phần tử **CollaborationActivity** (hoạt động hợp tác) con để chỉ ra chi tiết hơn việc lồng nhau của **BinaryCollaboration**.

Phần tử **CollaborationActivity** (hoạt động hợp tác) cũng có một thuộc tính:

- Một thuộc tính **name** (tên) YÊU CẦU.

6.4.17.1 Thuộc tính **name** (Tên)

Thuộc tính **name** (tên) YÊU CẦU là một chuỗi để xác định **CollaborationActivity** (hoạt động hợp tác) được chứa trong **BinaryCollaboration**. Nếu tài liệu *quy định quá trình* được xác định bởi giản đồ quy định quá trình kinh doanh ebXML [ebBPSS], giá trị thuộc tính **name** (tên) BẮT BUỘC phù hợp giá trị thuộc tính **name** (tên) của **CollaborationActivity** (hoạt động hợp tác) trong **BinaryCollaboration**, như được định nghĩa trong giản đồ quy định quá trình kinh doanh ebXML [ebBPSS].

6.4.18 Phần tử **Certificate** (Chứng chỉ)

Phần tử **Certificate** (chứng chỉ) xác định thông tin chứng chỉ đối với việc sử dụng trong *CPP* này. Một hoặc nhiều phần tử **Certificate** (chứng chỉ) có thể được cung cấp đối với việc sử dụng trong các chức năng an ninh khác nhau trong *CPP* đó. một ví dụ về phần tử **Certificate** (chứng chỉ) là:

```
<tp:Certificate tp:certId="CompanyA_SigningCert">
  <ds:KeyInfo> . . . </ds:KeyInfo>
</tp:Certificate>
```

Phần tử **Certificate** (chứng chỉ) có một thuộc tính đơn YÊU CẦU: **certId**. Phần tử **Certificate** (chứng chỉ) có một phần tử con đơn: **ds:KeyInfo**.

Phần tử **ds:KeyInfo** có thể bao gồm một chuỗi đầy đủ các chứng chỉ, trừ chứng chỉ leaf là phần tử **Certificate** (chứng chỉ) bao gồm khóa được sử dụng trong thao tác mật mã hóa không đối xứng khác nhau. (Chứng chỉ leaf là một chứng chỉ đã được ấn hành nhưng chưa được sử dụng ấn hành các chứng chỉ.) Nếu chứng chỉ leaf đã được ấn hành bởi một cơ quan chứng nhận trung gian, chuỗi đầy đủ cho cơ quan chứng nhận gốc NÊN được chứa bởi vì nó trợ giúp cho việc kiểm tra tính hợp lệ của chứng chỉ đối với một tập các mẫu neo chứng thực.

6.4.18.1 Thuộc tính certId (id của chứng chỉ)

Thuộc tính **certId** (id của chứng chỉ) YÊU CẦU là một ID [XML] đó là được đề cập bởi một phần tử **CertificateRef** (tham chiếu chứng chỉ) ở trong **CPP** đó. Đây là một ví dụ về cách một **CertificateRef** sẽ đề cập tới phần tử **Certificate** (chứng chỉ) được chỉ trong phần trước:

```
<tp:SigningCertificateRef tp:certId="CompanyA_SigningCert"/>
```

6.4.18.2 Phần tử ds:KeyInfo

Phần tử **ds:KeyInfo** xác định thông tin chứng chỉ. Nội dung của phần tử này và mọi phần tử con được xác định bởi định danh tài liệu số XML[XMLDSIG].

CHÚ THÍCH: Phần mềm tạo ra các **CPP** và **CPA** BẮT BUỘC nhận dạng phần tử **ds:KeyInfo** và chèn cấu trúc phần tử con cần thiết để xác định chứng chỉ đó.

6.4.19 Phần tử SecurityDetails (Chi tiết an ninh)

Phần tử **SecurityDetails** (chi tiết an ninh) xác định một tập **TrustAnchors** (mẫu neo chứng thực) và một **SecurityPolicy** (chính sách an ninh) tương ứng cho việc sử dụng trong **CPP** này. Một hoặc nhiều phần tử **SecurityDetails** (chi tiết an ninh) có thể được cung cấp đối với việc sử dụng trong các chức năng an ninh khác nhau trong **CPP** đó. một ví dụ về phần tử **SecurityDetails** (chi tiết an ninh) là:

```
<tp:SecurityDetails tp:securityId="CompanyA_MessageSecurity"> <tp:TrustAnchors
tp:trustId="MessageTrustAnchors">
  <tp:AnchorCertificateRef tp:certId="TrustedRootCertA3"/>
  <tp:AnchorCertificateRef tp:certId="TrustedRootCertA5"/>
</tp:TrustAnchors>
<tp:SecurityPolicy> ... </tp:SecurityPolicy>
</tp:SecurityDetails>
```

Phần tử **SecurityDetails** (chi tiết an ninh) có không hoặc một phần tử **TrustAnchors** (mẫu neo chứng thực) để bao gồm một tập các chứng chỉ để được chứng thực bởi *bên tham gia* đó. Nó cũng có không hoặc một phần tử **SecurityPolicy**.

Phần tử **SecurityDetails** (chi tiết an ninh) cho phép thỏa thuận đạt được điều mà các chứng chỉ gốc sẽ được sử dụng trong việc kiểm tra tính hợp lệ của các chứng chỉ của *bên tham gia* khác. Nó cũng có thể quy định chính sách đề cập thao tác của hạ tầng khóa công bố.

TCVN ISO/TS 15000-1 : 2007

Phần tử **SecurityDetails** (chi tiết an ninh) có một thuộc tính:

- một thuộc tính **securityId** (id an ninh) YÊU CẦU.

6.4.19.1 Thuộc tính securityId (id an ninh)

Thuộc tính **securityId** (id an ninh) YÊU CẦU là một ID [XML] được đề cập bởi một phần tử nào đó trong CPP đó. Đây là một ví dụ về cách một **SigningSecurityDetailsRef** liên quan đến phần tử **SecurityDetails** (chi tiết an ninh) được chỉ ra trong phần trước:

```
<tp:SigningSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
```

6.4.20 Phần tử TrustAnchors (Mẫu neo chứng thực)

Phần tử **TrustAnchors** (mẫu neo chứng thực) bao gồm một hoặc nhiều phần tử **AnchorCertificateRefs** (tham chiếu mẫu neo chứng thực), mỗi phần tử trong chúng liên quan đến một phần tử **Certificate** (chứng chỉ) (thuộc **PartyInfo**) để biểu diễn một chứng chỉ được chứng thực bởi *bên tham gia* này. Các chứng chỉ chứng thực này được sử dụng trong quá trình kiểm tra tính hợp lệ đường dẫn của chứng chỉ. Nếu một chứng chỉ theo truy vấn không "kết nối" tới một trong các mẫu neo chứng thực của *bên tham gia* này, thì nó được xem là không hợp lệ.

Phần tử **TrustAnchors** (mẫu neo chứng thực) cuối cùng chuyển sang các phần tử **KeyInfo** XMLDsig. Các phần tử này có thể bao gồm nhiều chứng chỉ (một chuỗi) và có thể chỉ dẫn tới các chứng chỉ đó có sử dụng phần tử **RetrievalMethod** (phương pháp khôi phục). Khi có một chain, mẫu neo chứng thực là chứng chỉ "leaf" đối với chứng chỉ của cơ quan chứng nhận (CA) ấn hành "gốc". CA gốc sẽ là một chứng chỉ tự-ấn hành và tự-ký và có sử dụng thông tin về người ấn hành và có thể các thuộc tính sử dụng cho khoá, chứng chỉ leaf ("đã được ấn hành nhưng không đưa ra" trong chuỗi) có thể được xác định. Chuỗi được chứa để tiện lợi các kiểm tra tính hợp lệ điển hình sẽ dẫn tới một CA "gốc". Chú ý rằng việc bao gồm một CA gốc trong một chuỗi không có nghĩa rằng CA gốc đang được thông báo như một mẫu neo chứng thực. Nó có thể là một chính sách PKI trong một số trường hợp, nhưng không phải tất cả, các CA trung gian được chứng thực. Nếu một CA gốc đã được chấp nhận như một mẫu neo chứng thực, tất cả của CA trung gian của nó và tất cả các chứng chỉ chúng ấn hành, sẽ là hợp lệ. Đó có thể không phải là điều đã dự định.

6.4.21 Phần tử SecurityPolicy (Chính sách an ninh)

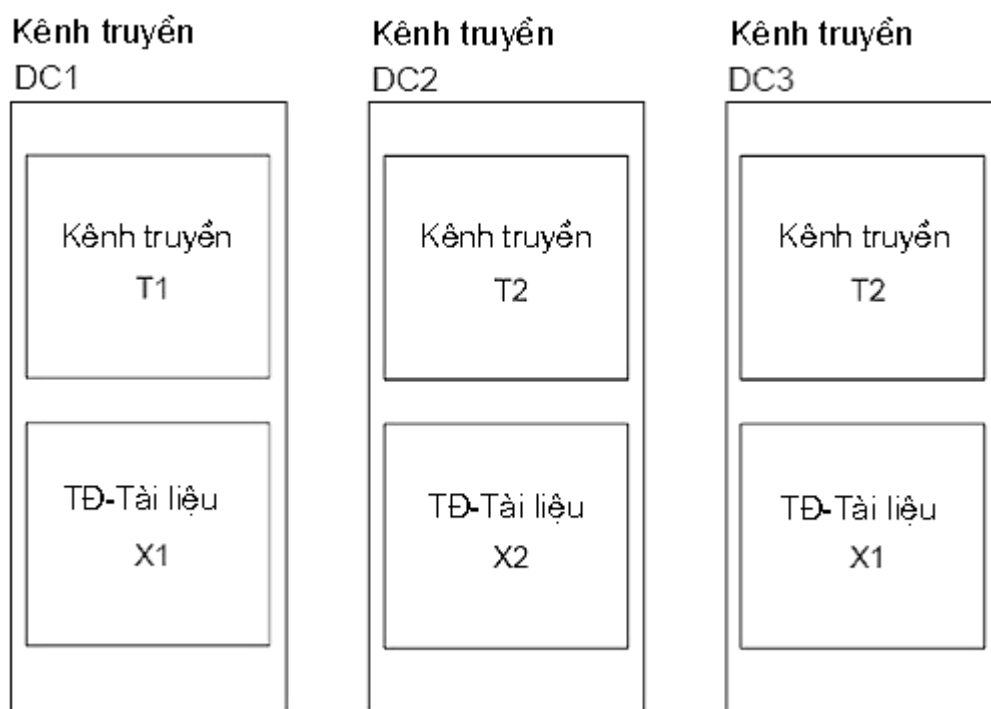
Phần tử **SecurityPolicy** (chính sách an ninh) là một trình giữ chỗ đối với các bộ máy tương lai sẽ cho phép *bên tham gia* đó để quy định chính sách của nó và các thành phần cụ thể theo yêu cầu của hạ tầng khóa công bố của nó. Ví dụ, nó có thể quy định các thủ tục kiểm tra việc hủy bỏ hoặc bắt ép liên quan đến tên, cách sử dụng hoặc độ dài đường dẫn.

6.4.22 Phần tử DeliveryChannel (Kênh truyền)

Một kênh truyền là một kết hợp của một phần tử **Transport** (truyền tải) và một phần tử **DocExchange** (trao đổi tài liệu) để mô tả các đặc điểm truyền thông *thông điệp* của *bên tham gia* đó. CPP PHẢI bao gồm một hoặc nhiều phần tử **DeliveryChannel** (kênh truyền), Một hoặc nhiều phần tử **Transport** (truyền tải) và Một hoặc nhiều phần tử **DocExchange**.

Mỗi kênh truyền PHẢI dẫn tới bất kỳ kết hợp của một phần tử **DocExchange** (trao đổi tài liệu) và một phần tử **Transport** (truyền tải). Cùng phần tử **DocExchange** (trao đổi tài liệu) hoặc cùng phần tử **Transport** (truyền tải) có thể được đề cập bởi nhiều hơn một kênh truyền. Hai kênh truyền có thể sử dụng cùng giao thức truyền và cùng giao thức trao đổi tài liệu và chỉ khác về các chi tiết như các địa chỉ truyền hoặc các định nghĩa an ninh.

Hình 5 - Minh họa ba kênh truyền



Các kênh truyền có các thuộc tính **ID** với các giá trị "DC1", "DC2" và "DC3". Mỗi kênh truyền bao gồm một định nghĩa truyền và một định nghĩa trao đổi-tài liệu. Mỗi định nghĩa truyền và mỗi định nghĩa trao đổi tài liệu cũng có một thuộc tính **ID** mà giá trị của nó được chỉ ra trong Hình.

Chú ý rằng kênh truyền DC3 minh họa rằng một kênh truyền có thể liên quan đến cùng định nghĩa truyền và định nghĩa trao đổi tài liệu được sử dụng bởi các kênh truyền khác trừ một kết hợp khác. Trong trường hợp này kênh truyền DC3 là một kết hợp của định nghĩa truyền T2 (cũng được đề cập bởi kênh truyền DC2) và định nghĩa trao đổi tài liệu X1 (cũng được đề cập bởi kênh truyền DC1).

Sau đây là cú pháp kênh truyền:

```
<tp:DeliveryChannel
  tp:channelId="channel1"
  tp:transportId="transport1"
  tp:docExchangeId="docExchange1"
  <tp:MessagingCharacteristics
    tp:syncReplyMode="none"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"
    tp:actor="urn:oasis:names:tc:ebxml-msg:actor:nextMSH"/>
  </tp:DeliveryChannel>
```

Mỗi phần tử **DeliveryChannel** (kênh truyền) xác định một phần tử **Transport** (truyền tải) và một phần tử **DocExchange** (trao đổi tài liệu) để cùng cấu tạo nên một định nghĩa kênh truyền đơn.

Phần tử **DeliveryChannel** (kênh truyền) có các thuộc tính sau:

- một thuộc tính **channelId** (id kênh truyền) YÊU CẦU;
- một thuộc tính **transportID** (id truyền tải) YÊU CẦU;
- một thuộc tính **docExchangeID** (id tài liệu trao đổi) YÊU CẦU.

Phần tử **DeliveryChannel** (kênh truyền) có một phần tử con YÊU CẦU là **MessagingCharacteristics**.

6.4.22.1 Thuộc tính ChannelId (id kênh truyền)

Thuộc tính **channelId** (id kênh truyền) là một thuộc tính **ID** [XML] xác định duy nhất phần tử **DeliveryChannel** (kênh truyền) để tham chiếu, có sử dụng các thuộc tính **IDREF** (tham chiếu id), từ các phần khác của **CPP** hoặc **CPA**.

6.4.22.2 Thuộc tính transportID (id truyền tải)

Thuộc tính **transportID** (id truyền tải) là một **IDREF** [XML] xác định phần tử **Transport** (truyền tải) để định nghĩa các đặc điểm truyền của kênh truyền. Nó **BẮT BUỘC** có một giá trị đó là bằng với giá trị của một thuộc tính **transportID** (id truyền tải) của một phần tử **Transport** (truyền tải) trong tài liệu **CPP** đó.

6.4.22.3 Thuộc tính docExchangeID (id tài liệu trao đổi)

Thuộc tính **docExchangeID** (id tài liệu trao đổi) là một **IDREF** [XML] xác định phần tử **DocExchange** (trao đổi tài liệu) để định nghĩa các đặc điểm trao đổi-tài liệu của kênh truyền. Nó **BẮT BUỘC** có một giá trị đó là bằng với giá trị của một thuộc tính **docExchangeID** (id tài liệu trao đổi) của một phần tử **DocExchange** (trao đổi tài liệu) trong tài liệu **CPP** đó.

6.4.23 Phần tử MessagingCharacteristics (Đặc điểm truyền thông điệp)

Phần tử **MessagingCharacteristics** (đặc điểm truyền thông điệp) mô tả các thuộc tính được kết hợp với các **thông điệp** được truyền qua một kênh truyền cho trước. Các **bên tham gia** hợp tác có thể đặt quy định để các thuộc tính này được cố định cho tất cả các **thông điệp** được gửi thông qua kênh truyền hoặc họ có thể thỏa thuận các thuộc tính này biến đổi trên cơ sở một "mỗi **thông điệp**".

CPP và các công cụ cấu tạo CPA và các công cụ phát triển CPA PHẢI kiểm tra định nghĩa kênh truyền (truyền và trao đổi tài liệu) để phù hợp với các thuộc tính này.

Phần tử **MessagingCharacteristics** (đặc điểm truyền thông điệp) có các thuộc tính sau:

- một thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) MẶC NHIÊN;
- một thuộc tính **ackRequested** (báo nhận được yêu cầu) MẶC NHIÊN;
- một thuộc tính **ackSignatureRequested** (báo nhận ký được yêu cầu) MẶC NHIÊN;
- một thuộc tính **duplicateElimination** (loại trừ sao chép lại) MẶC NHIÊN;
- một thuộc tính **actor** (chủ thể) MẶC NHIÊN.

6.4.23.1 Thuộc tính syncReplyMode (Phương thức trả lời đồng bộ)

Thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) là một tập đếm bao gồm các giá trị có thể sau đây:

- "mshSignalsOnly";
- "signalsOnly";
- "responseOnly";
- "signalsAndResponse";
- "none".

Thuộc tính này, khi có mặt, chỉ ra cái mà ứng dụng gửi mong muốn trong một phản hồi đồng thời (kênh truyền BẮT BUỘC được giới hạn cho một giao thực truyền đồng độ như HTTP khi **syncReplyMode** là không "none").

Giá trị "mshSignalsOnly" chỉ ra rằng phải hồi quay lại (trên phản hồi HTTP 200 trong trường hợp HTTP) chỉ bao gồm các thông điệp mức *trình quản lý dịch vụ thông điệp* (MSH) độc lập như báo nhận (đối với truyền thông điệp xác thực) và các thông điệp lỗi. Tất cả các phản hồi mức ứng dụng khác được trở lại không đồng bộ (có sử dụng một phần tử **DeliveryChannel** (kênh truyền) được xác định bởi dịch vụ và hành động theo truy vấn).

Giá trị "signalsOnly" chỉ ra rằng phải hồi quay lại (trên phản hồi HTTP 200 trong trường hợp HTTP) chỉ bao gồm một hoặc nhiều tín hiệu kinh doanh như được định nghĩa trong tài liệu *quy định quá trình* [ebBPSS], cộng thêm bất kỳ tín hiệu mức MSH được mang, trừ một *thông điệp phản hồi kinh doanh*. Nếu *quy định-quá trình* gọi đối với việc sử dụng của một thông điệp phản hồi-kinh doanh, thì sau đó BẮT BUỘC được trở lại không đồng bộ. Nếu *quá trình kinh doanh* không gọi đối với việc sử dụng của một tín hiệu xác thực việc chấp nhận, thì phần tử **Action** (hành động) trong *thông điệp ebXML* trở về đồng bộ BẮT BUỘC được đặt "ReceiptAcknowledgment". Nói cách khác, phần tử **Action** (hành động) trong *thông điệp ebXML* trở về đồng bộ (nó bao gồm cả hai; một tín hiệu xác thực việc nhận và một tín hiệu xác thực việc chấp nhận) BẮT BUỘC được đặt "AcceptanceAcknowledgment".

TCVN ISO/TS 15000-1 : 2007

Giá trị "responseOnly" chỉ ra rằng bất kỳ tín hiệu kinh doanh, cho dù chúng được chỉ ra trong *quy định quá trình*, là để bị lược bỏ và chỉ *kinh doanh-thông điệp* phản hồi trở lại đồng bộ, thêm vào bất kỳ tín hiệu mức MSH được mang. để phù hợp, thuộc tính **timeToAcknowledgeReceipt** (thời gian thông báo việc nhận) và timeToAcknowledgeAcceptances thuộc phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) tương ứng NÊN được đặt zero để chỉ ra rằng các tín hiệu đó không được sử dụng được sử dụng chút nào. Phần tử **Action** (hành động) trong *thông điệp ebXML* trở về đồng bộ được xác định bởi tên của hành động trong *CPA* đó điều đó tương ứng với **RespondingBusinessActivity** (phản hồi hoạt động kinh doanh) thích hợp trong *quá trình kinh doanh*.

Giá trị "signalsAndResponse" chỉ ra rằng ứng dụng sẽ phản hồi lại đồng bộ thông điệp phản hồi-kinh doanh thêm Một hoặc nhiều tín hiệu kinh doanh, thêm vào bất kỳ tín hiệu mức MSH được mang. Trong trường hợp này, mỗi tín hiệu và phản hồi đó được bó vào cùng *thông điệp ebXML* phải xuất hiện như một phần MIME phân tách (như là., được đặt trong một bộ chứa tải phân tách). để phù hợp, thuộc tính **timeToAcknowledgeReceipt** (thời gian thông báo việc nhận) và **timeToPerforms** thuộc phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) tương ứng NÊN có các giá trị đồng dạng. Thuộc tính **timeToAcknowledgeAcceptance**, nếu được quy định, NÊN cũng có cùng giá trị như các thuộc tính tính thời gian ở trên. Phần tử **Action** (hành động) trong *thông điệp ebXML* trở về đồng bộ được xác định bởi tên của hành động trong *CPA* đó điều đó tương ứng với **RespondingBusinessActivity** thích hợp trong *quá trình kinh doanh*.

Tín hiệu xác thực việc nhận đối với thông điệp phản hồi-kinh doanh, được gửi từ bên khởi tạo yêu cầu trở lại người đáp ứng, nếu nếu được gọi cho bởi *quy định-quá trình*, cũng BẮT BUỘC được truyền trên cùng kết nối đồng bộ.

CHÚ THÍCH: Đối với máy server (máy chủ) và khác HTTP 1.1, cả hai YÊU CẦU và trả lời HTTP sẽ phải được gửi và nhận trên cùng kết nối. Các thực hiện được giả định hàm là một kết nối HTTP sẽ được đóng sau khi một yêu cầu đồng bộ đơn trả lời trao đổi không thể để hỗ trợ phương thức trả lời đồng bộ "signalsAndResponse".

Giá trị "none", giá trị này giá trị mặc định hàm ẩn trong trường hợp không có mặt của thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ), chỉ ra rằng không có thông điệp phản hồi-kinh doanh cũng không có (các) tín hiệu kinh doanh sẽ được phản hồi lại một cách đồng bộ. Trong trường hợp này, toàn bộ các thông điệp mức trình quản lý dịch vụ *thông điệp* và mức *kinh doanh* sẽ được phản hồi lại như các *thông điệp* không đồng bộ phân tách.

Phần tử **SyncReply** (trả lời đồng bộ) của dịch vụ *thông điệp ebXML* được chứa trong tiêu đề SOAP ngay khi thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) có một giá trị khác "none". Nếu kênh truyền xác định một giao thức truyền mà không có khả năng đồng bộ (như SMTP), phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) KHÔNG PHẢI có một thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) với một giá trị khác "none".

Khi giá trị thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) khác "none", một kênh truyền đồng bộ PHẢI được sử dụng để trao đổi tất cả các *thông điệp* cần thiết để tạo ra một *giao dịch kinh doanh*. Nếu *quy định quá trình* đối với việc sử dụng của không từ chối việc nhận đối với *thông điệp* phản hồi, thì người khởi tạo được trông chờ để phản hồi lại một tín hiệu **ReceiptAcknowledgment** (báo nhận việc nhận) đã ký đối với *thông điệp* phản hồi của người phản hồi.

6.4.23.2 Thuộc tính **ackRequested** (Báo nhận được yêu cầu)

Thuộc tính **ackRequested** (báo nhận được yêu cầu) MẶC NHIÊN là một kiểu số đếm bao gồm các giá trị có thể sau đây:

- "always";
- "never";
- "perMessage".

Thuộc tính này có giá trị mặc định "perMessage" có nghĩa hoặc phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP có mặt hoặc vắng mặt có thể được thay đổi trên một cơ sở "mỗi *thông điệp*". Nếu thuộc tính này được thiết lập là "always", thì mọi *thông điệp* được gửi qua kênh truyền BẮT BUỘC có một phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP. Nếu thuộc tính này được thiết lập là "never", thì mọi *thông điệp* được gửi qua kênh truyền KHÔNG BẮT BUỘC có một phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP.

Nếu thuộc tính **ackRequested** (báo nhận được yêu cầu) không được đặt là "never", thì phần tử **ReliableMessaging** (truyền thông điệp tin cậy) phải có mặt thuộc phần tử **DocExchange** (trao đổi tài liệu) tương ứng để cung cấp các tham số truyền thông điệp xác thực cần thiết.

6.4.23.3 Thuộc tính **ackSignatureRequested** (Báo nhận ký được yêu cầu)

Thuộc tính **ackSignatureRequested** (báo nhận ký được yêu cầu) MẶC NHIÊN là một kiểu số đếm bao gồm các giá trị có thể sau đây:

- "always";
- "never";
- "perMessage".

Thuộc tính này xác định cách thức thuộc tính được ký trong phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP là để được thiết lập.

Nó có giá trị mặc định "perMessage" có nghĩa là thuộc tính được ký trong phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP có thể được đặt là "true" hoặc "false" trên một cơ sở "mỗi *thông điệp*". Nếu thuộc tính này được thiết lập là "always", thì mọi *thông điệp* được gửi qua kênh truyền mà có một phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP BẮT BUỘC có các thuộc tính của nó được ký đặt là "true". Nếu thuộc tính này được thiết lập là "never", thì mọi *thông điệp* được gửi qua kênh truyền that có một phần tử **AckRequested** (báo nhận được yêu cầu) trong tiêu đề SOAP

TCVN ISO/TS 15000-1 : 2007

BẮT BUỘC have các thuộc tính của nó được ký set là "false". Nếu thuộc tính **ackRequested** (báo nhận được yêu cầu) được thiết lập là "never", việc thiết lập thuộc tính **ackSignatureRequested** (báo nhận ký được yêu cầu) không ảnh hưởng.

CHÚ THÍCH: Bằng việc cho phép sử dụng của báo nhận được ký cho các *thông điệp* được gửi một cách tin cậy, một dạng không chắc chắn không từ chối việc nhận có thể được hỗ trợ. Điều này được xem không chắc chắn bằng tín hiệu xác thực việc nhận bởi vì không kiểm tra lược đồ nào có thể được thực hiện trên thiết bị mang trước khi phản hồi lại của báo nhận. Thuộc tính **ackSignatureRequested** (báo nhận ký được yêu cầu) có thể được thiết lập độc lập của giá trị đối với thuộc tính **isNonRepudiationReceiptRequired** (yêu cầu không từ chối nhận) thuộc phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh). Vì vậy, cho dù *quy định-quá trình* gốc quy định rằng không từ chối việc nhận là để được thực hiện, *CPP* và/hoặc *CPA* có thể ghi đè yêu cầu này, đặt **isNonRepudiationReceiptRequired** là "false" và **ackSignatureRequested** là "always" và do đó đạt được dạng không chắc chắn của dịch vụ không từ chối việc nhận.

6.4.23.4 Thuộc tính **duplicateElimination** (Loại trừ sao chép lại)

Thuộc tính **duplicateElimination** (loại trừ sao chép lại) MẶC NHIÊN là một kiểu số đếm bao gồm các giá trị có thể sau đây:

- "always";
- "never";
- "perMessage".

Thuộc tính này xác định phần tử **DuplicateElimination** (loại trừ sao chép lại) trong phần tử **MessageHeader** (tiêu đề thông điệp) trong tiêu đề SOAP là để có mặt hay không. Nó có giá trị mặc định "perMessage" có nghĩa là phần tử **DuplicateElimination** (loại trừ sao chép lại) trong tiêu đề SOAP có thể được có mặt hoặc vắng mặt trên một cơ sở "mỗi *thông điệp*". Nếu thuộc tính này được thiết lập là "always", thì mọi *thông điệp* được gửi qua kênh truyền BẮT BUỘC có một phần tử **DuplicateElimination** (loại trừ sao chép lại) trong tiêu đề SOAP. Nếu thuộc tính này được thiết lập là "never", thì mọi *thông điệp* được gửi qua kênh truyền KHÔNG BẮT BUỘC có một phần tử **DuplicateElimination** (loại trừ sao chép lại) trong tiêu đề SOAP. Nếu thuộc tính **DuplicateElimination** (loại trừ sao chép lại) không được đặt là "never", thì phần tử **PersistDuration** (khoảng thời gian lâu dài) phải có mặt thuộc phần tử **DocExchange** (trao đổi tài liệu) tương ứng để cung cấp tham số lưu trữ lâu dài cần thiết.

6.4.23.5 Thuộc tính **actor** (Chủ thể)

Thuộc tính **actor** (chủ thể) MẶC NHIÊN là một kiểu số đếm của các giá trị có thể sau đây:

- "urn:oasis:names:tc:ebxml-msg:actor:nextMSH";
- "urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH".

Đây là một URI sẽ được sử dụng như giá trị đối với thuộc tính **actor** (chủ thể) trong phần tử **AckRequested** (báo nhận được yêu cầu) (xem [ebMS]) trong trường hợp có mặt sau đó trong tiêu đề

SOAP, như được quản lý bởi thuộc tính **ackRequested** (báo nhận được yêu cầu) trong phần tử **MessagingCharacteristics** (đặc điểm truyền thông điệp) trong *CPA* đó. Nếu thuộc tính **ackRequested** (báo nhận được yêu cầu) được thiết lập là "never", thì việc thiết lập thuộc tính **actor** (chủ thể) không ảnh hưởng.

6.4.24 Phần tử Transport (Truyền tải)

Phần tử **Transport** (truyền tải) xác định khả năng truyền thông trên mạng của *bên tham gia* đó. Một hoặc nhiều phần tử **Transport** (truyền tải) **BẮT BUỘC** có mặt trong một *CPP*, mỗi phần tử mô tả một cơ chế *bên tham gia* đó sử dụng để gửi các *thông điệp*, một cơ chế nó sử dụng để nhận *thông điệp* hoặc cả hai. Ví dụ sau minh họa cấu trúc của một phần tử **Transport** (truyền tải) điển hình:

```
<tp:Transport tp:transportId="transportA1">
  <tp:TransportSender <!-- 0 or 1 time -->
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:Protocol>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">
        SSL
      </tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyA_ClientCert"/>
      <tp:ServerSecurityDetailsRef
        tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportClientSecurity>
  </tp:TransportSender>
  <tp:TransportReceiver <!-- 0 or 1 time -->
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:Protocol>
    <tp:Endpoint
      tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler"
      tp:type="allPurpose"/>
    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">
        SSL
      </tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>
      <tp:ClientSecurityDetailsRef
        tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
```

Phần tử **Transport** (truyền tải) bao gồm không hoặc một phần tử **TransportSender** (truyền tải bên gửi) và không hoặc một phần tử **TransportReceiver** (truyền tải bên nhận).

Một phần tử **Transport** (truyền tải) bao gồm cả hai phần tử **TransportReceiver** (truyền tải bên nhận) và **TransportSender** (truyền tải bên gửi) được gọi là hai chiều trong đó nó có thể được sử dụng để gửi và nhận các *thông điệp*. Nếu *bên tham gia* đó liên quan đến phương thức truyền đồng bộ (ở đây các trả lời được phải hồi lại trên cùng các kết nối TCP các *thông điệp* đã được gửi; Xem phần 8.4.23.1), *CPP* của nó **BẮT BUỘC** cung cấp một **ServiceBinding** (quy định dịch vụ) bao gồm **ActionBindings** (quy định hoạt động) được bó thành một **DeliveryChannel** (kênh truyền) để sử dụng một **Transport** (truyền tải) hai chiều.

Một phần tử **Transport** (truyền tải) bao gồm một **TransportSender** (truyền tải bên gửi) hoặc một phần tử **TransportReceiver**, nhưng không cả hai, được gọi là một chiều. Một **Transport** (truyền tải) một

TCVN ISO/TS 15000-1 : 2007

chiều chỉ có thể được sử dụng đối với việc gửi hoặc nhận các *thông điệp* (không cả hai) phụ thuộc vào phần tử nó bao gồm.

Một *CPP* bao gồm nhiều phần tử **Transport** (truyền tải) như cần thiết để diễn tả đầy đủ các khả năng truyền thông đi vào và đi ra của *bên tham gia*. Ví dụ, *bên tham gia* đó có thể gửi và nhận các *thông điệp* thông qua HTTP và SMTP, *CPP* của nó sẽ bao gồm một phần tử **Transport** (truyền tải) có bao gồm các đặc tính HTTP của nó và một phần tử **Transport** (truyền tải) khác có bao gồm các đặc tính SMTP của nó.

Phần tử **Transport** (truyền tải) có:

- một thuộc tính **transportID** (id truyền tải) YÊU CẦU.

6.4.24.1 Thuộc tính transportID (id truyền tải)

Thuộc tính **transportID** (id truyền tải) YÊU CẦU là một ID [XML] đó là liên quan đến một phần tử **Transport** (truyền tải) nơi nào đó trong *CPP* đó. Đây là một ví dụ về một **DeliveryChannel** liên quan đến phần tử **Transport** (truyền tải) được đưa ra trong phần trước:

```
<tp:DeliveryChannel tp:channelId="channelA1"
  tp:transportId="transportA1"
  tp:docExchangeId="docExchangeA1">
  <tp:MessagingCharacteristics . . . />
</tp:DeliveryChannel>
```

6.4.25 Phần tử TransportSender (truyền tải bên gửi)

Phần tử **TransportSender** (truyền tải bên gửi) bao gồm các đặc tính liên quan đến *bên gửi* của một **DeliveryChannel**. Phần tử **TransportProtocol** (giao thức truyền tải) YÊU CẦU của nó quy định thủ tục truyền sẽ được sử dụng đối với việc gửi các *thông điệp*. Phần tử **AccessAuthentication**(s), nếu có mặt, thì quy định (các) kiểu xác thực truy cập được hỗ trợ bởi khách. Phần tử **TransportClientSecurity**, nếu có mặt, thì xác định các điều khoản của *bên tham gia* đó đối với an ninh lớp truyền bên khách.

Phần tử **TransportSender** (truyền tải bên gửi) không có thuộc tính.

6.4.26 Phần tử TransportProtocol (Giao thức truyền tải)

Phần tử **TransportProtocol** (giao thức truyền tải) xác định một giao thức truyền mà *bên tham gia* đó có khả năng có sử dụng để gửi hoặc nhận Dữ liệu *kinh doanh*. Thuộc tính **version** (phiên bản) MẶC NHIÊN xác định phiên bản quy định của giao thức.

CHÚ THÍCH: Đây là mục tiêu của tiêu chuẩn này để cho phép hỗ trợ cho mọi khả năng truyền của việc mang nội dung của MIME có sử dụng từ vựng được định nghĩa trong tài liệu này.

6.4.27 Phần tử AccessAuthentication (Xác thực truy cập)

Phần tử **AccessAuthentication** (xác thực truy cập), nếu có mặt, thì chỉ ra cơ chế xác thực có thể được sử dụng bởi một máy server (máy chủ) truyền tải để yêu cầu một yêu cầu của ứng dụng khách và bởi một ứng dụng khách để cung cấp thông tin xác thực cho máy server (máy chủ).

Ví dụ, [RFC2617] quy định hai giản đồ xác thực truy cập cho HTTP: "basic" và "digest". một ứng dụng khách để hỗ trợ cả hai nên có hai phần tử **AccessAuthentication** (xác thực truy cập), như được chỉ ra sau đây. khi nhiều giản đồ được hỗ trợ, thứ tự của chúng được quy định trong *CPP* đó chỉ ra thứ tự ưu tiên.

```
<tp:TransportSender>
  <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
  <tp:AccessAuthentication>digest</tp:AccessAuthentication>
  <tp:AccessAuthentication>basic</tp:AccessAuthentication>
  <tp:TransportClientSecurity>
    ...
  </tp:TransportClientSecurity>
</tp:TransportSender>
```

CHÚ THÍCH: Một *CPA* bao gồm, đối với mỗi **TransportSender** (truyền tải bên gửi) hoặc **TransportReceiver** (truyền tải bên nhận), chỉ thỏa thuận về phần tử **AccessAuthentication** (xác thực truy cập).

CHÚ THÍCH: Đối với xác thực cơ sở, các giá trị userid (id người sử dụng) và password (mật khẩu) được cấu hình thông qua các phương tiện nằm ngoài phạm vi của tiêu chuẩn này.

6.4.28 Phần tử TransportClientSecurity (Truyền tải an ninh bên Client)

Phần tử **TransportClientSecurity** (truyền tải an ninh bên Client) cung cấp thông tin về ứng dụng client (khách) truyền tải của *bên tham gia* này cần thiết bởi máy server (máy chủ) truyền tải *bên tham gia* khác để cho phép một kết nối an toàn tới được thiết lập giữa hai *bên tham gia* đó. Nó bao gồm một phần tử **TransportSecurityProtocol** (giao thức an ninh truyền tải) YÊU CẦU, không hoặc một phần tử **ClientCertificateRef** (Tham chiếu chứng chỉ bên client), không hoặc một phần tử **ServerSecurityDetailsRef** (tham chiếu chi tiết an ninh server) và không hoặc nhiều phần tử **EncryptionAlgorithm** (thuật toán mã hóa).

Trong phương thức truyền *thông điệp* không đồng bộ, *bên gửi* luôn luôn là một ứng dụng khách tới ứng dụng chủ của *bên nhận*. Trong phương thức truyền *thông điệp* đồng bộ, mức MSH trả lời (và có thể là một tín hiệu kinh doanh được góivà/hoặc phản hồi *kinh doanh*) được gửi trở lại trên cùng kết nối đó *thông điệp kinh doanh* khởi tạo được đến. Trong các trường hợp như vậy, ở đây *bên gửi* là máy server (máy chủ) và *bên nhận* là ứng dụng khách và kết nối đó đã tồn tại, các phần tử **TransportClientSecurity** (truyền tải an ninh bên Client) của *bên gửi* và **TransportServerSecurity** (truyền tải an ninh bên Server) của *bên nhận* PHẢI được bỏ qua.

6.4.29 Phần tử TransportSecurityProtocol (Giao thức an ninh truyền tải)

Phần tử **TransportSecurityProtocol** (giao thức an ninh truyền tải) xác định giao thức an ninh lớp truyền được hỗ trợ bởi truyền tải gốc. Thuộc tính **version** (phiên bản) MẶC NHIÊN xác định phiên bản quy định của giao thức.

Đối với mật mã hoá, giao thức là TLS phiên bản 1.0[RFC2246], sử dụng sự mật mã hóa khóa công bố. Phụ lục E của quy định TLS phiên bản 1.0 [RFC2246] bao gồm tính tương thích ngược với SSL [SSL].

6.4.30 Phần tử **ClientCertificateRef** (Tham chiếu chứng chỉ bên client)

Phần tử **ClientCertificateRef** (tham chiếu chứng chỉ bên client) xác định chứng chỉ được sử dụng bởi môđun an ninh truyền tải của bên client. **CertId** (id chứng chỉ) của thuộc tính **IDREF** (tham chiếu id) YÊU CẦU xác định chứng chỉ được sử dụng bởi dẫn tới phần tử **Certificate** (chứng chỉ) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính **ID**. Một bên Client có khả năng TLS-HTTP, ví dụ, việc sử dụng chứng chỉ này tự xác thực với máy server (máy chủ) HTTP an toàn của *bên nhận*.

Phần tử **ClientCertificateRef** (tham chiếu chứng chỉ bên client), nếu có mặt, thì chỉ ra rằng việc xác thực lẫn nhau giữa client (khách) và server (chủ) (như là., người khởi tạo và người phản hồi của kết nối HTTP) BẮT BUỘC được thực hiện.

Phần tử **ClientCertificateRef** (tham chiếu chứng chỉ bên client) có:

- một thuộc tính **certId** (id của chứng chỉ) YÊU CẦU.

6.4.31 Phần tử **ServerSecurityDetailsRef** (Tham chiếu chi tiết an ninh server)

Phần tử **ServerSecurityDetailsRef** (tham chiếu chi tiết an ninh server) xác định các mẫu neo chứng thực và chính sách an ninh *bên tham gia* này sẽ áp dụng cho chứng chỉ xác thực máy server (máy chủ) của *bên tham gia* khác.

Phần tử **ServerSecurityDetailsRef** (tham chiếu chi tiết an ninh server) có:

- một thuộc tính **securityId** (id an ninh) YÊU CẦU.

6.4.32 Thuật toán mật mã hóa

Không hoặc nhiều phần tử **EncryptionAlgorithm** (thuật toán mã hóa) có thể được chứa thuộc phần tử **TransportClientSecurity** (truyền tải an ninh bên Client) hoặc **TransportServerSecurity** (truyền tải an ninh bên Server). Nhiều phần tử được sử dụng nhiều trong một nội dung **CPP**, để thông báo các khả năng hoặc tham chiếu; thông thường, một **CPA** bao gồm nội dung được thỏa thuận. Khi không hoặc nhiều hơn một phần tử có mặt trong một **CPA**, các *bên tham gia* đồng ý để cho phép khả năng thương lượng tự động của phần tử **TransportSecurityProtocol** (giao thức an ninh truyền tải) để xác định thuật toán thực được sử dụng.

Các việc đặt thứ tự của phần tử đó sẽ ảnh hưởng tới sự ưu tiên cho các thuật toán. một lý do chính đối với việc bao gồm phần tử này là để cho phép việc sử dụng của thuộc tính **minimumStrength**; một giá trị lớn hơn đối với thuộc tính này có thể chỉ ra rằng khả năng mật mã hóa cao được yêu cầu hoặc đã được thỏa thuận đối với **TransportSecurityProtocol** (giao thức an ninh truyền tải).

Xem phần 8.4.50 đối với mô tả đầy đủ phần tử này.

Đối với SSL và TLS, thông thường để quy định viết mật mã các giá trị phù hợp như các giá trị văn bản đối với phần tử **EncryptionAlgorithm** (thuật toán mã hóa). Các giá trị này bao gồm, không hạn chế đối với:

- **SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA;**
 - TLS_RSA_WITH_3DES_EDE_CBC_SHA;
 - SSL_RSA_WITH_3DES_EDE_CBC_SHA;
 - SSL_RSA_WITH_RC4_128_MD5;
 - SSL_RSA_WITH_RC4_128_SHA;
 - SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA;
 - SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA.

Tham khảo các quy định gốc cho các kiểu số đếm và đề cập của các giá trị này.

6.4.33 Phần tử **TransportReceiver** (truyền tải bên nhận)

Phần tử **TransportReceiver** (truyền tải bên nhận) bao gồm các đặc tính liên quan đến bên đang nhận của một **DeliveryChannel**. Phần tử **TransportProtocol** (giao thức truyền tải) YÊU CẦU của nó quy định thủ tục truyền sẽ được sử dụng đối với việc nhận các *thông điệp*. Một hoặc nhiều phần tử **Endpoint** (điểm cuối) YÊU CẦU quy định các địa chỉ logic mà các *thông điệp* có thể nhận. Phần tử **AccessAuthentication**(s), nếu có mặt, thì chỉ ra (các) kiểu xác thực truy cập được hỗ trợ bởi máy server (máy chủ). Không hoặc một phần tử **TransportServerSecurity** (truyền tải an ninh bên Server) xác định các điều khoản của *bên tham gia* đó cho an ninh lớp truyền bên máy server (máy chủ).

Phần tử **TransportReceiver** (truyền tải bên nhận) không có thuộc tính.

6.4.34 Phần tử **Endpoint** (Điểm cuối)

Một hoặc nhiều các phần tử **Endpoint** (điểm cuối) PHẢI được cung cấp cho mỗi phần tử **TransportReceiver** (truyền tải bên nhận). Mỗi **Endpoint** (điểm cuối) quy định một địa chỉ logic và một chỉ dẫn của các loại *thông điệp* có thể nhận tại địa điểm đó.

Mỗi **Endpoint** có các thuộc tính sau:

- một thuộc tính **URI** YÊU CẦU;
- một thuộc tính **type** (kiểu) MẶC NHIÊN.

6.4.34.1 Thuộc tính **URI**

Thuộc tính **URI** YÊU CẦU quy định một URI việc định danh địa chỉ của một nguồn. Giá trị thuộc tính **URI** PHẢI phù hợp với cú pháp để diễn tả URI như được định nghĩa trong [RFC2396].

6.4.34.2 Thuộc tính **type** (Kiểu)

Thuộc tính **type** (kiểu) xác định mục đích của điểm cuối này. Giá trị của **type** (kiểu) là một kiểu số đếm của các giá trị được phép là "login", "request", "response", "error" và "allPurpose". Có thể có nhiều nhất một giá trị trong mỗi thuộc tính. Nếu thuộc tính **type** (kiểu) bị lược bỏ, giá trị của nó mặc định là

"allPurpose". Điểm cuối "login" được sử dụng đối với địa chỉ cho *thông điệp* khởi tạo giữa hai *bên tham gia*.

Các điểm cuối "request" và "response" được sử dụng để yêu cầu và phản hồi các *thông điệp*, theo thứ tự, để cho phép các thông điệp lỗi được nhận, mỗi phần tử **Transport** (truyền tải) PHẢI bao gồm ít nhất một điểm cuối của kiểu "error", "response" hoặc "allPurpose".

Các kiểu của phần tử **Endpoint** (điểm cuối) trong một phần tử **TransportReceiver** (truyền tải bên nhận) BẮT BUỘC không chồng nhau. Vì vậy, không đúng để bao gồm cả hai một phần tử **Endpoint** (điểm cuối) "allPurpose" cùng với phần tử **Endpoint** (điểm cuối) của kiểu bất kỳ.

6.4.35 Phần tử TransportServerSecurity (Truyền tải an ninh bên Server)

Phần tử **TransportServerSecurity** (truyền tải an ninh bên Server) cung cấp thông tin về máy server (máy chủ) truyền tải của *bên tham gia* này được cần bởi ứng dụng khách truyền tải của *bên tham gia* khác để cho phép một kết nối an toàn tới được thiết lập giữa hai bên. Nó bao gồm một phần tử **TransportSecurityProtocol** (giao thức an ninh truyền tải) YÊU CẦU, một phần tử **ServerCertificateRef** (tham chiếu chứng chỉ bên Server) YÊU CẦU, không hoặc một phần tử **ClientSecurityDetailsRef** (tham chiếu chi tiết an ninh bên Client) và không hoặc nhiều phần tử **EncryptionAlgorithm** (thuật toán mã hóa). Xem phần 8.4.32 đối với một mô tả về phần tử **EncryptionAlgorithm** (thuật toán mã hóa).

CHÚ THÍCH: Xem chú thích trong phần 8.4.27 liên quan đến sự thích hợp của phần tử **TransportServerSecurity** (truyền tải an ninh bên Server) khi sử dụng trả lời đồng bộ.

6.4.36 Phần tử ServerCertificateRef (Tham chiếu chứng chỉ bên Server)

Phần tử **ServerCertificateRef** (tham chiếu chứng chỉ bên Server), nếu có mặt, thì xác định chứng chỉ được sử dụng bởi môđun an ninh truyền tải của máy server (máy chủ). **CertId** của thuộc tính **IDREF** (tham chiếu id) YÊU CẦU xác định chứng chỉ được sử dụng dẫn tới phần tử **Certificate** (chứng chỉ) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính **ID**. Một máy server (máy chủ) HTTP TLS-enabled , ví dụ, sử dụng chứng chỉ này tự xác thực ứng dụng khách của *bên gửi* TLS.

Phần tử **ServerCertificateRef** (tham chiếu chứng chỉ bên Server) BẮT BUỘC có mặt Nếu thủ tục an ninh truyền tải sử dụng các chứng chỉ. Nói cách khác, nó có thể được lược bỏ (như là; Nếu xác thực bằng mật khẩu).

Phần tử **ServerCertificateRef** (tham chiếu chứng chỉ bên Server) có

- một thuộc tính **certId** (id của chứng chỉ) YÊU CẦU.

6.4.37 Phần tử ClientSecurityDetailsRef (Tham chiếu chi tiết an ninh bên Client)

Phần tử **ClientSecurityDetailsRef** (tham chiếu chi tiết an ninh bên Client), nếu có mặt, thì xác định các mẫu neo chứng thực và chính sách an ninh that *bên tham gia* này sẽ áp dụng cho chứng chỉ xác thực ứng dụng khách của *bên tham gia* khác.

Phần tử **ClientSecurityDetailsRef** (tham chiếu chi tiết an ninh bên Client) có:

- một thuộc tính **securityId** (id an ninh) YÊU CẦU.

6.4.38 Giao thức truyền

Trong các phần sau đây, chúng ta đề cập các chi tiết cụ thể của mỗi giao thức truyền được hỗ trợ.

6.4.38.1 HTTP

HTTP là Giao thức truyền siêu văn bản[HTTP]. Đối với HTTP, endpoint là một URI PHẢI phù hợp với [RFC2396]. Phụ thuộc vào ứng dụng, có thể có một hoặc nhiều endpoints, được xác định bởi ứng dụng.

Sau đây là một ví dụ về một **endpoint** HTTP:

```
<tp:Endpoint tp:uri="http://example.com/servlet/ebxmlhandler"
tp:type="request"/>
```

Các **endpoint** (điểm cuối) "request" và "response" có thể được ghi đè đối với một yêu cầu riêng hoặc một phản hồi đồng thời ứng dụng cụ thể URI trong các tài liệu *kinh doanh* được trao đổi thuộc CPA đó.

Đối với một phản hồi đồng thời, **endpoint** (điểm cuối) "response" được bỏ qua nếu có mặt. Một phản hồi đồng thời luôn được phản hồi trở lại trên kết nối đang tồn tại, Nghĩa là; tới URI đó được xác định như nguồn kết nối.

6.4.38.2 SMTP

SMTP là giao thức truyền thư tín đơn giản[SMTP]. Để sử dụng với tiêu chuẩn này, thư tín internet đa năng mở rộng [MIME] BẮT BUỘC được hỗ trợ. Đối với SMTP, địa chỉ truyền thông là địa chỉ thư tín được hạn định đầy đủ của *bên tham gia* đích như được xác định bởi [RFC2822]. Sau đây là một ví dụ về một **endpoint** (điểm cuối) SMTP:

```
<tp:Endpoint tp:uri="mailto:ebxmlhandler@example.com"
tp:type="request"/>
```

CHÚ THÍCH: Đại diện truyền thư tín SMTP (MTA) có thể mã hoá dữ liệu nhị phân khi việc nhận MTA không hỗ trợ truyền nhị phân. Nói chung, truyền SMTP có thể liên quan mã hoá và giải mã nội dung các mã hoá truyền như một *thông điệp* tăng theo một trình tự các MTA. Các thay đổi như vậy có thể trong một số trường hợp không hợp lệ của một vài loại chữ ký thậm chí không có hành động gây hại hoặc lỗi truyền xảy ra.

CHÚ THÍCH: SMTP bởi chính nó (không có xác thực hoặc mật mã hoá nào) tùy thuộc vào sự phủ nhận của dịch vụ và sự lừa đảo bởi các *bên tham gia* không được nhận ra. Điều đề nghị mạnh mẽ là các *bên tham gia* đó mà chọn lựa SMTP như lớp truyền của họ cũng như chọn lựa một phương tiện mật mã hoá và xác thực phù hợp hoặc trong lớp trao đổi-tài liệu hoặc trong lớp truyền như [S/MIME].

CHÚ THÍCH: SMTP là một giao thức không đồng bộ không đảm bảo một một chất lượng dịch vụ riêng. Một báo nhận lớp truyền (như là; một báo nhận SMTP) tới việc nhận của một *thông điệp* thư tín cấu thành một xác nhận

TCVN ISO/TS 15000-1 : 2007

trên phần của máy server (máy chủ) SMTP mà nó biết cách thức phân phối *thông điệp* thư tín và sẽ cố gắng thực hiện tại cùng thời điểm trong tương lai. Tuy nhiên, *thông điệp* đó không được củng cố và có thể không bao giờ được truyền tới người nhận. Hơn nữa, bên *gửi* nhận thấy một báo nhận lớp truyền chỉ từ nút gần nhất. Nếu *thông điệp* đó qua các nút trung gian, SMTP không cung cấp một báo nhận đầu cuối tới đầu cuối. Do đó, việc nhận một báo nhận SMTP không đảm bảo rằng *thông điệp* sẽ được truyền tới ứng dụng và lỗi để nhận một báo nhận SMTP không được chứng tỏ Message chưa được gửi. Khuyến cáo rằng giao thức truyền thông điệp xác thực trong dịch vụ *thông điệp* ebXML được sử dụng cùng với SMTP.

6.4.38.3 FTP

FTP là giao thức truyền tệp[RFC959].

Mỗi *bên tham gia* gửi một *thông điệp* có sử dụng FTP PUT. **Endpoint** (điểm cuối) quy định *id* của người sử dụng và đường dẫn danh mục đầu vào (for PUTs to *bên tham gia* này). Một ví dụ về một endpoint (điểm cuối) FTP là:

```
<tp:Endpoint uri="ftp://userid@server.foo.com"
  tp:type="request"/>
```

Do FTP cần tương thích qua tất cả các thực hiện, FTP cho ebXML sẽ sử dụng các tập tối thiểu các lệnh và tham số sẵn có cho FTP như được quy định trong [RFC959], phần 5.1 và được sửa đổi [RFC1123], phần 4.1.2.13. Phương thức này PHẢI chỉ là dòng và kiểu BẮT BUỘC là ASCII không in (AN), ảnh (I) (nhị phân) hoặc Local 8 (L 8) (nhị phân giữa các máy 8-bit và các máy 36 bit từ – Đối với một máy 8-bit Local 8 là giống như dạng ảnh).

Phương thức luồng đóng kết nối dữ liệu trên điểm kết thúc của tệp. Bên máy server (máy chủ) FTP đó BẮT BUỘC thiết lập điều khiển là "PASV" trước khi mỗi lệnh truyền để đạt được một cặp cổng duy nhất nếu có nhiều phiên làm việc của bên thứ ba.

CHÚ THÍCH: [RFC 959] khẳng định rằng User-FTP NÊN gửi một lệnh của PORT để ấn định một cổng dữ liệu không mặc định trước khi mỗi lệnh truyền được ấn hành để cho phép nhiều phép truyền trong khoảng thời gian một FTP đơn bởi vì trễ trong thời gian dài sau khi một kết nối TCP được đóng đến khi cặp khe cắm của nó có thể được tái sử dụng.

CHÚ THÍCH: Dạng thức của 227 trả lời cho một lệnh PASV chưa được chuẩn hoá tốt và một ứng dụng khách FTP có thể giả định rằng các dấu ngoặc đơn được chỉ ra trong [RFC959] sẽ có mặt trong khi trong một số trường hợp không có mặt. Nếu chương trình của người sử dụng FTP không quét trả lời đối với chữ số đầu tiên của máy server (máy chủ) và các số hiệu cổng, kết quả sẽ là FTP của người sử dụng có thể trở đến máy server (máy chủ) lỗi. trong phản hồi, h1, h2, h3, h4 là địa chỉ IP của máy server (máy chủ) và p1, p2 là một cổng truyền dữ liệu không mặc định mà PASV đã ấn định.

CHÚ THÍCH: Như một khuyến cáo cho tính trong suốt của bức tường lửa, [RFC1579] đề nghị rằng ứng dụng khách gửi một lệnh PASV, cho phép máy server (máy chủ) thực hiện một TCP thụ động mở một số cổng ngẫu nhiên và khai báo ứng dụng khách số hiệu cổng. Ứng dụng khách sau đó có thể thực hiện một hành động chủ động mở để thiết lập kết nối đó.

CHÚ THÍCH: Do phương thức LUÔNG đóng kết nối dữ liệu trên điểm cuối của tệp, việc nhận FTP có thể khẳng định ngắt kết nối không bình thường nếu một mã điều khiển 226 hoặc 250 không được nhận từ máy gửi.

CHÚ THÍCH: [RFC1579] cũng tạo ra sự quan sát để nó có thể được đánh giá để tăng cường giao thức FTP để có ứng dụng khách gửi một lệnh APSV mới (chủ động tất cả) tại khởi động để có thể cho phép máy server (máy chủ) thực hiện quan điểm này để luôn thực hiện một việc mở chủ động. Một mã trả lời mới 151 lên được ấn hành thông phản hồi tới tất cả các yêu cầu truyền tệp không được mở trước một PORT hoặc lệnh PASV; thông điệp này nên bao gồm số hiệu cổng để sử dụng đối với việc truyền đó. Một lệnh PORT vẫn có thể được gửi tới máy server (máy chủ) đã được nhận trước APSV; Điều đó được ghi đề cách hoạt động mặc định cho thao tác truyền tiếp theo, vì vậy cho phép bên thứ ba truyền.

6.4.39 Phần tử DocExchange (Trao đổi tài liệu)

Phần tử **DocExchange** (trao đổi tài liệu) cung cấp thông tin để các *bên tham gia* BẮT BUỘC đồng ý về việc liên quan trao đổi các tài liệu giữa các *bên tham gia* đó. Thông tin này bao gồm các đặc tính dịch vụ truyền *thông điệp* (như là; dịch vụ *thông điệp ebXML[ebMS]*).

Sau đây là cấu trúc của phần tử **DocExchange** (trao đổi tài liệu) của *CPP*. Trình tự các phần mô tả mỗi phần tử con dưới dạng chi tiết hơn.

```
<tp:DocExchange tp:docExchangeId="docExchangeB1">
  <tp:ebXMLSenderBinding tp:version="2.0"> <!-- 0 or 1 -->
    <tp:ReliableMessaging> <!-- 0 or 1 -->
      . . .
    </tp:ReliableMessaging>
    <tp:PersistDuration> <!-- 0 or 1 -->
      . . .
    </tp:PersistDuration>
    <tp:SenderNonRepudiation> <!-- 0 or 1 -->
      . . .
    </tp:SenderNonRepudiation>
    <tp:SenderDigitalEnvelope> <!-- 0 or 1 -->
      . . .
    </tp:SenderDigitalEnvelope>
    <tp:NamespaceSupported> <!-- 0 or more -->
      . . .
    </tp:NamespaceSupported>
  </tp:ebXMLSenderBinding>
  <tp:ebXMLReceiverBinding tp:version="2.0"> <!-- 0 or 1 -->
    <tp:ReliableMessaging> <!-- 0 or 1 -->
      . . .
    </tp:ReliableMessaging>
    <tp:PersistDuration> <!-- 0 or 1 -->
      . . .
    </tp:PersistDuration>
    <tp:ReceiverNonRepudiation> <!-- 0 or 1 -->
      . . .
    </tp:ReceiverNonRepudiation>
    <tp:ReceiverDigitalEnvelope> <!-- 0 or 1 -->
      . . .
    </tp:ReceiverDigitalEnvelope>
    <tp:NamespaceSupported> <!-- 0 or more -->
      . . .
    </tp:NamespaceSupported>
  </tp:ebXMLReceiverBinding>
</tp:DocExchange>
```

TCVN ISO/TS 15000-1 : 2007

Phần tử **DocExchange** (trao đổi tài liệu) bao gồm không hoặc một phần tử con **ebXMLSenderBinding** và không hoặc một phần tử con **ebXMLReceiverBinding**. Nó **BẮT BUỘC** có ít nhất một phần tử con. **CPP** và các công cụ cấu tạo **CPA** và các công cụ phát triển **CPA PHẢI** kiểm tra sự có mặt của một phần tử con.

CHÚ THÍCH: Phần trao đổi tài liệu có thể được mở rộng cho các dịch vụ truyền *thông điệp* khác dịch vụ *thông điệp ebXML* bằng việc thêm các phần tử **xxxSenderBinding** (xxx Quy định bên gửi) và **xxxReceiverBinding** (xxx Quy định bên nhận) bổ sung và các phần tử con của nó mô tả các dịch vụ khác, ở đây xxx được thay thế bởi tên của của quy định bổ sung. Một ví dụ là XMLPSenderBinding/XMLPReceiverBinding, nó có thể hỗ trợ đối với quy định giao thức tương lai.

6.4.39.1 Thuộc tính docExchangeID (id tài liệu trao đổi)

Phần tử **DocExchange** (trao đổi tài liệu) có một thuộc tính **docExchangeID** (id tài liệu trao đổi) **YÊU CẦU** đơn đó là một ID [XML] để cung cấp một định danh duy nhất that có thể được tham chiếu từ một nơi nào khác trong tài liệu **CPP** đó.

6.4.40 Phần tử ebXMLSenderBinding (Quy định bên gửi ebXML)

Phần tử **ebXMLSenderBinding** (quy định bên gửi ebXML) mô tả các đặc tính liên quan đến việc gửi *thông điệp* với dịch vụ *thông điệp ebXML[ebMS]*. Phần tử **ebXMLSenderBinding** (quy định bên gửi ebXML) bao gồm các phần tử con sau đây:

- không hoặc một phần tử **ReliableMessaging** (truyền thông điệp tin cậy) quy định các đặc điểm của truyền thông điệp xác thực;
- không hoặc một phần tử **PersistDuration** (khoảng thời gian lâu dài) quy định khoảng thời gian đối với các *thông điệp* tất yếu phải được lưu trữ lâu dài đối với mục đích loại trừ trùng lặp;
- không hoặc một phần tử **SenderNonRepudiation** (không từ chối bên gửi) quy định các yêu cầu của người gửi và chứng chỉ để ký *thông điệp*;
- không hoặc một phần tử **SenderDigitalEnvelope** (đường bao số bên gửi) quy định các yêu cầu của người gửi đối với mật mã hóa bởi phương pháp đường bao số [DIGENV];
- không hoặc nhiều phần tử **NamespaceSupported** (tên miền được hỗ trợ) để định danh bất kỳ đuôi mở rộng của tên miền được hỗ trợ bởi sự thi hành dịch vụ truyền *thông điệp*.

Phần tử **ebXMLSenderBinding** (quy định bên gửi ebXML) có một thuộc tính:

- một thuộc tính **version** (phiên bản) **YÊU CẦU**.

CHÚ THÍCH: Một **CPA** có thể hợp lệ thậm chí khi lược bỏ toàn bộ phần tử con thuộc **ebXMLSenderBinding** (quy định bên gửi ebXML).

6.4.40.1 Thuộc tính version (Phiên bản)

Thuộc tính **version** (phiên bản) **YÊU CẦU** xác định phiên bản của quy định dịch vụ *thông điệp ebXML* đang được sử dụng.

6.4.41 Phần tử **ReliableMessaging** (Truyền thông điệp tin cậy)

Phần tử **ReliableMessaging** (truyền thông điệp tin cậy) quy định các đặc tính của trao đổi *thông điệp ebXML tin cậy*. Giá trị mặc định áp dụng nếu phần tử **ReliableMessaging** (truyền thông điệp tin cậy) bị lược bỏ là "BestEffort". Sau đây là cấu trúc phần tử đó:

```
<tp:ReliableMessaging>
  <tp:Retries>5</tp:Retries>
  <tp:RetryInterval>PT2H</tp:RetryInterval>
  <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
</tp:ReliableMessaging>
```

Các ngữ nghĩa của truyền thông điệp xác thực được giải thích trong quy định dịch vụ *thông điệp ebXML[ebMS]* chương các phối hợp truyền thông điệp xác thực.

Phần tử **ReliableMessaging** (truyền thông điệp tin cậy) bao gồm các phần tử con sau đây.

- không hoặc một phần tử **Retries**;
- không hoặc một phần tử **RetryInterval**;
- một phần tử **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) **YÊU CẦU**.

6.4.41.1 Phần tử **Retries** (thử lại) và **RetryInterval** (Khoảng thời gian giữa lần thử lại)

Các phần tử **Retries** (thử lại) và **RetryInterval** quy định số hiệu được phép của các phép thử lại và khoảng thời gian, được giải thích như một khoảng thời gian của giản đồ XML[XMLSCHEMA-2], giữa các lần thử lại của việc gửi một *thông điệp* được truyền tin cậy ấy sau đấy một thời gian chờ đối với báo nhận. Mục đích của phần tử **RetryInterval** (khoảng thời gian giữa lần thử lại) là để tận dụng khả năng thành công trong lần thử lại bằng việc trì hoãn lần thử đó đến khi điều kiện tạm thời nào đó gây ra lỗi đó được chỉnh. **RetryInterval** (khoảng thời gian giữa lần thử lại) áp dụng cho thời gian giữa việc gửi *thông điệp gốc* và lần thử đầu tiên, cũng như thời gian giữa các lần thử tiếp theo.

Các phần tử **Retries** (thử lại) và **RetryInterval** (khoảng thời gian giữa lần thử lại) **BẮT BUỘC** được chứa cùng nhau hoặc được lược bỏ cùng nhau. Nếu chúng được lược bỏ, các giá trị của số lượng tương ứng các lần thử và khoảng thời gian giữa các lần thử) là một vấn đề nội bộ của mỗi *bên tham gia*.

6.4.41.2 Phần tử **MessageOrderSemantics** (Ngữ nghĩa thứ tự thông điệp)

Phần tử **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) là một kiểu số đếm bao gồm các giá trị có thể sau đây:

- "Guaranteed";
- "NotGuaranteed".

Sự có mặt của một phần tử **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) trong tiêu đề SOAP đối với các *thông điệp ebXML* xác định nếu việc đặt thứ tự các *thông điệp* được gửi từ *bên tham gia* cần được bảo vệ vì vậy *bên tham gia* nhận các thông điệp này theo thứ tự mà nó đã được gửi. Nếu phần tử **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) được thiết lập là "Guaranteed", thì

thông điệp ebXML BẮT BUỘC bao gồm một phần tử **MessageOrder** (thứ tự thông điệp) trong tiêu đề SOAP. Nếu phần tử **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) được thiết lập là "NotGuaranteed", thì *thông điệp* ebXML KHÔNG BẮT BUỘC bao gồm một phần tử **MessageOrder** (thứ tự thông điệp) trong tiêu đề SOAP. Thứ tự *thông điệp* được đảm bảo hàm ý việc sử dụng sự loại trừ trùng lặp. Vì vậy, phần tử **PersistDuration** (khoảng thời gian lâu dài) BẮT BUỘC cũng xuất hiện nếu **MessageOrderSemantics** (ngữ nghĩa thứ tự thông điệp) được thiết lập là "Guaranteed".

6.4.42 Phần tử PersistDuration (Khoảng thời gian lâu dài)

Giá trị của phần tử **PersistDuration** (khoảng thời gian lâu dài) là độ dài nhỏ nhất của thời gian, được diễn tả như một khoảng thời gian trong giản đồ XML[XMLSCHEMA-2], dữ liệu đó từ một *thông điệp* đó được gửi một cách tin cậy được giữ trong bộ lưu trữ lâu dài bởi một *thông điệp* dịch vụ ebXML thi hành để nhận *thông điệp* đó để tạo thuận lợi cho việc loại trừ các trùng lặp. Khoảng thời gian này cũng áp dụng cho các *thông điệp* phản hồi được lưu giữ lâu dài để cho phép trả lời tự động để chỉ sự trùng lặp các *thông điệp* mà không cần xử lý lặp lại chúng bởi ứng dụng. Đối với các quy tắc để quản lý phần tử **PersistDuration**, xem các phần 8.4.23.4 và 8.4.41.2.

6.4.43 Phần tử SenderNonRepudiation (Không từ chối bên gửi)

Phần tử **SenderNonRepudiation** (không từ chối bên gửi) truyền các yêu cầu của người gửi *thông điệp* và chứng chỉ cho việc không từ chối. Không từ chối chứng tỏ người gửi một *thông điệp* và ngăn ngừa các bảo sao sau đó của các nội dung của *thông điệp* đó. Không từ chối dựa trên cơ sở việc ký *thông điệp* có sử dụng chữ ký dạng số XML[XMLDSIG]. Cấu trúc phần tử đó là như sau:

```
<tp:SenderNonRepudiation>
  <tp:NonRepudiationProtocol>
    http://www.w3.org/2000/09/xmldsig#
  </tp:NonRepudiationProtocol>
  <tp:HashFunction>
    http://www.w3.org/2000/09/xmldsig#sha1
  </tp:HashFunction>
  <tp:SignatureAlgorithm>
    http://www.w3.org/2000/09/xmldsig#dsa-sha1
  </tp:SignatureAlgorithm>
  <tp:SigningCertificateRef tp:certId="CompanyA_SigningCert"/>
</tp:SenderNonRepudiation>
```

Nếu Phần tử **SenderNonRepudiation** (không từ chối bên gửi) bị lược bỏ, *thông điệp* đó không được ký dưới dạng số.

Phần tử **SenderNonRepudiation** (không từ chối bên gửi) bao gồm các phần tử con sau đây:

- một phần tử **NonRepudiationProtocol** (giao thức không từ chối) YÊU CẦU;
- một phần tử **HashFunction** (hàm băm) YÊU CẦU (như là; SHA1, MD5);
- một phần tử **SignatureAlgorithm** (thuật toán ký) YÊU CẦU;
- một phần tử **SigningCertificateRef** (tham chiếu chứng chỉ ký) YÊU CẦU.

6.4.44 Phần tử **NonRepudiationProtocol** (Giao thức không từ chối)

Phần tử **NonRepudiationProtocol** (giao thức không từ chối) YÊU CẦU xác định công nghệ sẽ được sử dụng để ký dạng số một *thông điệp*. Nó có một thuộc tính **version** (phiên bản) MẶC NHIÊN đơn mà giá trị của nó là một chuỗi để xác định phiên bản của công nghệ quy định.

6.4.45 Phần tử **HashFunction** (Hàm băm)

Phần tử **HashFunction** (hàm băm) YÊU CẦU xác định thuật toán được sử dụng để tính toán bản tóm lược của *thông điệp* được ký.

6.4.46 Phần tử **SignatureAlgorithm** (Thuật toán ký)

Phần tử **SignatureAlgorithm** (thuật toán ký) YÊU CẦU xác định thuật toán được sử dụng để tính toán giá trị của chữ ký dạng số. Các giá trị mong muốn bao gồm: RSA-MD5, RSA-SHA1, DSA-MD5, DSA-SHA1, SHA1 cùng với RSA, MD5 cùng với RSA và v...v.

CHÚ THÍCH: Các phần thực hiện nên được chuẩn bị đối với các giá trị trong các trường hợp mức cao hơn và/hoặc thấp hơn và với các sử dụng khác nhau của các dấu nối và liên kết.

Phần tử **SignatureAlgorithm** (thuật toán ký) có ba thuộc tính:

- một thuộc tính **oid** (id của đối tượng) MẶC NHIÊN;
- một thuộc tính **w3c** MẶC NHIÊN;
- một thuộc tính **enumeratedType** (kiểu tập hợp) MẶC NHIÊN.

6.4.46.1 Thuộc tính **oid** (id của đối tượng)

Thuộc tính **oid** (id của đối tượng) phục vụ như một cách để cung cấp một định danh đối tượng đối với thuật toán ký. định nghĩa hình thức của các oid trong khuyến cáo ITU-T X.208 (ASN.1), Chương 28; sự ấn định của "cây chóp" được đưa ra trong Phụ lục B, Phụ lục C và Phụ lục D của X.208 (<http://www.itu.int/POD/>). Các giá trị thường được sử dụng (trong IETF dạng số nguyên có chấm) đối với các thuật toán ký bao gồm:

- 1.2.840.113549.1.1.4 - MD5 with RSA mật mã hoá;
- 1.2.840.113549.1.1.5 - SHA-1 with RSA Encryption.

6.4.46.2 Thuộc tính **w3c**

Thuộc tính **w3c** phục vụ như một cách để cung cấp một định danh đối tượng đối với thuật toán ký. Các định nghĩa của các giá trị này trong các quy định [XMLDSIG] hoặc [XMLENC]. Các giá trị mong muốn đối với các thuật toán ký bao gồm:

- <http://www.w3.org/2000/09/xmlnsig#dsa-sha1>,
- <http://www.w3.org/2000/09/xmlnsig#rsa-sha1>.

6.4.46.3 Thuộc tính **enumeratedType** (Kiểu tập hợp)

Thuộc tính **enumeratedType** (kiểu tập hợp) quy định một cách khác nhau để thông dịch giá trị văn bản của phần tử **SignatureAlgorithm** (thuật toán ký). Thuộc tính này là cho việc xác định giản đồ, sơ đồ, dạng thức định danh thuật toán ký trong tương lai.

6.4.47 Phần tử SigningCertificateRef (Tham chiếu chứng chỉ ký)

Phần tử **SigningCertificateRef** (tham chiếu chứng chỉ ký) YÊU CẦU xác định chứng chỉ *bên gửi* sử dụng để ký các *thông điệp*. YÊU CẦU thuộc tính **IDREF** (tham chiếu id) của nó, certId liên quan đến phần tử **Certificate** (chứng chỉ) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính ID.

6.4.48 Phần tử SenderDigitalEnvelope (Đường bao số bên gửi)

Phần tử **SenderDigitalEnvelope** (đường bao số bên gửi) cung cấp các yêu cầu của người gửi đối với việc mật mã hoá *thông điệp* có sử dụng phương pháp đường bao số [DIGENV].

Đường bao số là một thủ tục trong *thông điệp* được mật mã hóa bằng phương pháp mật mã hoá đối xứng (khóa bí mật được chia sẻ) và khóa bí mật được gửi tới người nhận *thông điệp* được mật mã hóa với khóa công bố của người nhận. Cấu trúc phần tử đó là:

```
<tp:SenderDigitalEnvelope>
  <tp:DigitalEnvelopeProtocol tp:version="2.0">
    S/MIME
  </tp:DigitalEnvelopeProtocol>
  <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
  <tp:EncryptionSecurityDetailsRef
    tp:securityId="CompanyA_MessageSecurity"/>
</tp:SenderDigitalEnvelope>
```

Phần tử **SenderDigitalEnvelope** (đường bao số bên gửi) bao gồm:

- một phần tử **DigitalEnvelopeProtocol** (giao thức đường bao số) YÊU CẦU;
- một phần tử **EncryptionAlgorithm** (thuật toán mã hóa) YÊU CẦU;
- không hoặc một phần tử **EncryptionSecurityDetailsRef** (tham chiếu chi tiết mật mã hóa an ninh).

6.4.49 Phần tử DigitalEnvelopeProtocol (Giao thức đường bao số)

Phần tử **DigitalEnvelopeProtocol** (giao thức đường bao số) YÊU CẦU xác định giao thức mật mã hoá *thông điệp* được sử dụng. Thuộc tính **version** (phiên bản) YÊU CẦU xác định phiên bản của giao thức.

6.4.50 Phần tử EncryptionAlgorithm (Thuật toán mã hóa)

Phần tử **EncryptionAlgorithm** (thuật toán mã hóa) YÊU CẦU xác định thuật toán mật mã hóa được sử dụng. Xem phần 8.4.32.

Phần tử **EncryptionAlgorithm** (thuật toán mã hóa) có bốn thuộc tính:

- một thuộc tính **minimumStrength** (khả năng tối thiểu) MẶC NHIÊN;
- một thuộc tính **oid** (id của đối tượng) MẶC NHIÊN;

- một thuộc tính **w3c** MẶC NHIÊN;
- một thuộc tính **enumeratedType** (kiểu tập hợp) MẶC NHIÊN.

6.4.50.1 Thuộc tính **minimumStrength** (Khả năng tối thiểu)

Thuộc tính **minimumStrength** (khả năng tối thiểu) mô tả khả năng hiệu quả của thuật toán mật mã hóa PHẢI cung cấp dưới dạng "effective" hoặc các bit ngẫu nhiên. Giá trị này nhỏ hơn độ dài của khoá đó theo bit khi các bit kiểm tra được sử dụng trong khoá. Vì vậy, ví dụ, các bit 8 kiểm tra của một khoá DES 64-bit không được chứa trong bộ đếm và để yêu cầu một độ lới nhỏ nhất như được hỗ trợ bởi DES sẽ được báo cáo bằng việc đặt **minimumStrength** (khả năng tối thiểu) là 56.

6.4.50.2 Thuộc tính **oID** (id của đối tượng)

Thuộc tính **oID** (id của đối tượng) là một cách để cung cấp một định danh đối tượng đối với thuật toán mật mã hóa. Định nghĩa hình thức của các oID trong khuyến cáo ITU-T X.208 (ASN.1), chương 28; việc ấn định của "top of tree" (mức đỉnh của cây) trong Phụ lục B, Phụ lục C và Phụ lục D của X.208 (<http://www.itu.int/POD/>). Các giá trị được sử dụng chung (trong IETF dạng số nguyên có dấu) đối với các thuật toán mật mã hoá bao gồm:

- 1.2.840.113549.3.2 (RC2-CBC), 1.2.840.113549.3.4 (RC4 Thuật toán mật mã hóa),
- 1.2.840.113549.3.7 (DES-EDE3-CBC), 1.2.840.113549.3.9 (RC5 CBC Pad),
- 1.2.840.113549.3.10 (DES CDMF), 1.2.840, 1.3.14.3.2.7 (DES-CBC).

6.4.50.3 Thuộc tính **w3c**

Thuộc tính **w3c** phục vụ như một cách để cung cấp một định danh đối tượng đối với thuật toán mật mã hóa. Các định nghĩa của các giá trị này trong quy định [XMLENC]. Các giá trị mong muốn bao gồm:

- <http://www.w3.org/2001/04/xmlenc#3des-cbc>;
- <http://www.w3.org/2001/04/xmlenc#aes128-cbc>;
- <http://www.w3.org/2001/04/xmlenc#aes256-cbc>.

6.4.50.4 Thuộc tính **enumeratedType** (Kiểu tập hợp)

Thuộc tính **enumeratedType** (kiểu tập hợp) quy định một cách thông dịch giá trị văn bản của phần tử **EncryptionAlgorithm** (thuật toán mã hóa). Thuộc tính này đối với việc xác định các giản đồ và dạng thức định danh thuật toán trong tương lai.

6.4.51 Phần tử **EncryptionSecurityDetailsRef** (Tham chiếu chi tiết mật mã hóa an ninh)

Phần tử **EncryptionSecurityDetailsRef** (tham chiếu chi tiết mật mã hóa an ninh) xác định các mẫu neo chứng thực và chính sách an ninh để *bên tham gia* này (gửi) sẽ áp dụng cho chứng chỉ mật mã hoá của *bên tham gia* khác (nhận). Nó YÊU CẦU thuộc tính **IDREF** (tham chiếu id), securityId, liên quan đến phần tử **SecurityDetails** (chi tiết an ninh) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính **ID**.

6.4.52 Phần tử **NamespaceSupported** (Tên miền được hỗ trợ)

Phần tử **NamespaceSupported** (tên miền được hỗ trợ) có thể được chứa không hoặc nhiều lần. Mỗi lần xuất hiện của phần tử **NamespaceSupported** (tên miền được hỗ trợ) xác định một namespace được hỗ trợ bởi thực hiện dịch vụ truyền *thông điệp*. Nó có một thuộc tính **location** YÊU CẦU và một thuộc tính **version** (phiên bản) MẶC NHIÊN. Thuộc tính **location** hỗ trợ một URI để đạt được giản đồ được kết hợp với tên miền đó. Thuộc tính **version** (phiên bản) cung cấp một giá trị phiên bản, khi tồn tại, cho tên miền đó. Trong khi phần tử **NamespaceSupported** (tên miền được hỗ trợ) có thể được sử dụng để liệt kê các tên miền có thể được mong muốn để được sử dụng trong khoảng dạng thức số, động cơ này chủ yếu cho sự mở rộng, các biến số về phiên bản và các tăng cường khác có thể được mong đợi hoặc chỉ cho sự sử dụng. Ví dụ, hỗ trợ cho ngôn ngữ đánh dấu xác nhận an ninh[SAML] xác định như sau:

```
<tp:NamespaceSupported tp:location="http://www.oasis-
open.org/committees/security/docs/draft-sstc-schema-assertion-27.xsd"
tp:version="1.0"> http://www.oasis-open.org/committees/security/docs/draft-
sstc-schema-assertion-27.xsd</tp:NamespaceSupported>
```

Ngoài ra, phần tử **NamespaceSupported** (tên miền được hỗ trợ) có thể được sử dụng để định danh các tên miền đó được kết hợp với các phần thân *thông điệp* (xem phần 8.5) và đặc biệt khi các tên miền này không được chỉ ra một cách hàm ẩn thông qua các phần của **ProcessSpecification** hoặc khi chúng chỉ ra các đuôi mở rộng của Các tên miền đối với các phần thân của thiết bị mang.

6.4.53 Phần tử **ebXMLReceiverBinding** (Quy định bên nhận ebXML)

Phần tử **ebXMLReceiverBinding** (quy định bên nhận ebXML) mô tả các đặc tính liên quan đến việc nhận các *thông điệp* với dịch vụ *thông điệp* ebXML[ebMS]. Phần tử **ebXMLReceiverBinding** (quy định bên nhận ebXML) bao gồm các phần tử con sau đây:

- không hoặc một phần tử **ReliableMessaging** (truyền thông điệp tin cậy) (xem phần 8.4.41);
- không hoặc một phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) quy định các yêu cầu của *bên nhận* cho việc ký *thông điệp*;
- không hoặc một phần tử **ReceiverDigitalEnvelope** (đường bao số của bên nhận) quy định các yêu cầu của *bên nhận* và chứng chỉ cho việc mật mã hóa bởi phương pháp đường bao số [DIGENV];
- không hoặc nhiều phần tử **NamespaceSupported** (tên miền được hỗ trợ) (xem phần 8.4.52).

Phần tử **ebXMLReceiverBinding** (quy định bên nhận ebXML) có một thuộc tính:

- một thuộc tính **version** (phiên bản) YÊU CẦU (xem phần 8.4.40.1)

CHÚ THÍCH: Một CPA có thể hợp lệ thậm chí khi lược bỏ tất cả phần tử con thuộc phần tử **ebXMLReceiverBinding** (quy định bên nhận ebXML).

6.4.54 Phần tử **ReceiverNonRepudiation** (Không từ chối bên nhận)

Phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) truyền các yêu cầu *thông điệp* của *bên nhận* đối với không từ chối. Không từ chối người gửi một *thông điệp* và ngăn sự từ chối sau đó các nội dung của *thông điệp* đó. Không từ chối dựa trên cơ sở việc ký *thông điệp* có sử dụng Chữ ký dạng số XML[XMLDSIG]. Cấu trúc phần tử đó là như sau:

```
<tp:ReceiverNonRepudiation>
  <tp:NonRepudiationProtocol>
    http://www.w3.org/2000/09/xmlldsig#
  </tp:NonRepudiationProtocol>
  <tp:HashFunction>
    http://www.w3.org/2000/09/xmlldsig#sha1
  </tp:HashFunction>
  <tp:SignatureAlgorithm>
    http://www.w3.org/2000/09/xmlldsig#dsa-sha1
  </tp:SignatureAlgorithm>
  <tp:SigningSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
</tp:ReceiverNonRepudiation>
```

Nếu phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) bị lược bỏ, *thông điệp* đó không được ký dạng số.

Phần tử **ReceiverNonRepudiation** (không từ chối bên nhận) bao gồm các phần tử con sau đây:

- một phần tử **NonRepudiationProtocol** (giao thức không từ chối) YÊU CẦU (xem phần 8.4.44);
- một phần tử **HashFunction** (hàm băm) YÊU CẦU (như là; SHA1, MD5) (xem phần 8.4.45);
- một phần tử **SignatureAlgorithm** (thuật toán ký) YÊU CẦU (xem phần 8.4.46);
- không hoặc một phần tử **SigningSecurityDetailsRef** (tham chiếu chi tiết an ninh ký).

6.4.55 Phần tử SigningSecurityDetailsRef (Tham chiếu chi tiết an ninh ký)

Phần tử **SigningSecurityDetailsRef** (tham chiếu chi tiết an ninh ký) xác định các mẫu neo chứng thực và chính sách an ninh mà *bên tham gia* này (nhận) sẽ áp dụng cho chứng chỉ ký của *bên tham gia* khác (gửi). Nó YÊU CẦU thuộc tính **IDREF** (tham chiếu id), **securityId**, liên quan đến phần tử **SecurityDetails** (chi tiết an ninh) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính **ID**.

6.4.56 Phần tử ReceiverDigitalEnvelope (Đường bao số của bên nhận)

Phần tử **ReceiverDigitalEnvelope** (đường bao số của bên nhận) cung cấp các yêu cầu của *bên nhận* để mật mã hoá *thông điệp* có sử dụng phương pháp đường bao số [DIGENV].

Đường bao số là một thủ tục mà trong đó *thông điệp* được mật mã hóa bởi mật mã hóa đối xứng (khóa bí mật được chia sẻ) và khóa bí mật được gửi tới người nhận *thông điệp* được mật mã hóa với khóa công bố của người nhận. Cấu trúc phần tử đó là:

```
<tp:ReceiverDigitalEnvelope>
  <tp:DigitalEnvelopeProtocol tp:version="2.0">
    S/MIME
  </tp:DigitalEnvelopeProtocol>
  <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
  <tp:EncryptionCertificateRef
    tp:certId="CompanyA_EncryptionCert"/>
</tp:ReceiverDigitalEnvelope>
```

TCVN ISO/TS 15000-1 : 2007

Phần tử **ReceiverDigitalEnvelope** (đường bao số của bên nhận) bao gồm:

- một phần tử **DigitalEnvelopeProtocol** (giao thức đường bao số) YÊU CẦU (xem phần 8.4.49);
- một phần tử **EncryptionAlgorithm** (thuật toán mã hóa) YÊU CẦU (xem phần 8.4.50);
- một phần tử **EncryptionCertificateRef** (tham chiếu chứng chỉ mật mã hóa) YÊU CẦU.

6.4.57 Phần tử EncryptionCertificateRef (Tham chiếu chứng chỉ mật mã hóa)

Phần tử **EncryptionCertificateRef** (tham chiếu chứng chỉ mật mã hóa) YÊU CẦU xác định chứng chỉ *bên gửi* sử dụng cho việc mật mã hóa các *thông điệp*. Nó YÊU CẦU thuộc tính **IDREF** (tham chiếu id), certId liên quan đến phần tử **Certificate** (chứng chỉ) (thuộc phần tử **PartyInfo**) điều đó phải phù hợp giá trị thuộc tính **ID**.

6.4.58 Phần tử OverrideMshActionBinding (Quy định hoạt động MSH ghi đè)

Phần tử **OverrideMshActionBinding** (quy định hoạt động MSH ghi đè) có thể xuất hiện không hoặc nhiều lần. Nó có hai thuộc tính YÊU CẦU. Thuộc tính **action** (hành động) xác định hành động mức trình quản lý dịch vụ *thông điệp*, thuộc tính **delivery** không sử dụng **DeliveryChannel** mặc định cho các hành động của trình quản lý dịch vụ thông điệp. Thuộc tính **ChannelId** (id kênh truyền) quy định **DeliveryChannel** được sử dụng thay thế.

6.5 Phần tử SimplePart (Thành phần đơn giản)

Phần tử **SimplePart** (thành phần đơn giản) cung cấp một danh sách lặp lại của các bộ phận cấu thành, chủ yếu được xác định bởi giá trị kiểu nội dung MIME.

Phần tử **SimplePart** (thành phần đơn giản) có hai thuộc tính YÊU CẦU: **id** và **mimetype** (kiểu MIME). Thuộc tính **ID**, của kiểu ID, cung cấp giá trị sẽ được sử dụng sau đó để tham chiếu phần thông điệp này khi quy định cách mà các phần được đóng gói vào các phần tử hỗn hợp, nếu có việc đóng gói phần tử hỗn hợp. Thuộc tính **mimeType** (kiểu MIME) có thể cung cấp các giá trị thực của kiểu nội dung đối với phần *thông điệp* đơn giản đang được quy định. Các Giá trị thuộc tính đó cũng có thể tạo ra việc sử dụng của một ký tự sao đại diện, "*", để chỉ ra một khiểu mức đỉnh tùy ý, một kiểu phụ tùy ý hoặc một kiểu đầy đủ, "**/*". **SimpleParts** (thành phần đơn giản) với các ký tự đại diện các kiểu có thể được sử dụng để chỉ ra nhiều hơn các khả năng xử lý việc đóng gói mở.

SimplePart (thành phần đơn giản) có một thuộc tính **mimparameters** MẶC NHIÊN, việc sử dụng thuộc tính này được mô tả trong phần 8.6.2. **SimplePart** (thành phần đơn giản) cũng có một thuộc tính **xlink:role** MẶC NHIÊN định danh một số nguồn để mô tả phần mime hoặc mục đích của nó; Xem phụ lục F đối với một thảo luận của việc sử dụng của giá trị này trong [ebMS]. Nếu có mặt, thì nó PHẢI có một giá trị đó là một URI hợp lệ phù hợp với quy định [XLINK].

Sau đây là các ví dụ phần tử **SimplePart** (thành phần đơn giản):

```
<tp:SimplePart tp:id="I001" tp:mimetype="text/xml"/>
```

```
<tp:SimplePart tp:id="I002" tp:mimetype="application/xml"/>
<tp:SimplePart tp:id="I002" tp:mimetype="*/xml"/>
```

Phần tử **SimplePart** (thành phần đơn giản) có thể có không hoặc nhiều phần tử **NamespaceSupported**s. Mỗi phần tử trong các phần tử này xác định tên miền nào đó được hỗ trợ đối với XML được đóng gói trong phần thân đơn giản của gốc. Nội dung của **Packaging** (đóng gói) có thể trả lại một cách rất dễ dàng để liệt kê tất cả các tên miền đó được sử dụng trong một **SimplePart** (thành phần đơn giản). Ví dụ, khi xác định **SimplePart** (thành phần đơn giản) đối với một đường bao SOAP, như một phần của một *thông điệp* ebXML, không cần thiết liệt kê tất cả các tên miền đó. Tuy nhiên, các đuôi mở rộng bất thường nào đó, các phiên bản mới hoặc đuôi mở rộng an ninh bất thường có mặt, điều có ích là thông báo các tình trạng bất thường này một cách rõ ràng trong việc đóng gói. Tuy nhiên, lỗi liệt kê tất cả các tên miền được sử dụng trong một **SimplePart** (phần đơn giản), thậm chí ở đây, các tên miền này bị bắt buộc bởi một giao thức truyền *thông điệp* cho trước. Theo quy ước, Khi liệt kê đầy đủ các tên miền được hỗ trợ trong một phần tử **SimplePart** (thành phần đơn giản), phần tử **NamespaceSupported** (tên miền được hỗ trợ) đầu tiên xác định gián đồ đối với **SimplePart** (thành phần đơn giản) khi phần tử **NamespaceSupported**s (tên miền được hỗ trợ) tiếp theo trình bày các tên miền được nhập bởi gián đồ đó. Phần tử **NamespaceSupported**s (tên miền được hỗ trợ) bổ sung nào đó chỉ ra đuôi mở rộng.

CHÚ THÍCH: Việc định danh rõ ràng các tên miền được nói đến được sử dụng khi tyhấy cần thiết. Vì vậy, các ví dụ CPP và CPA trong Phụ lục A và Phụ lục B định danh một cách rõ ràng tên miền dịch vụ truyền *thông điệp* ebXML nhưng lược bỏ tên miền đường bao SOAP và Chữ ký dạng số XML được nhập trong gián đồ đó đối với tên miền dịch vụ truyền *thông điệp* ebXML.

Cùng phần tử **SimplePart** (thành phần đơn giản) có thể được tham chiếu từ (như là, tái sử dụng) nhiều phần tử **Packaging** (đóng gói).

6.6 Phần tử Packaging (Đóng gói)

Cây phụ của phần tử **Packaging** (đóng gói) cung cấp thông tin cụ thể về cách thức các thành phần cấu thành *tiêu đề thông điệp* đó và thiết bị mang được đóng gói để truyền trên đường truyền, bao gồm thông tin quyết định về việc đóng gói an ninh mức tài liệu được sử dụng và cách thức mà các đặc tính an ninh được áp dụng. Cây phụ điển hình thuộc phần tử **Packaging** (đóng gói) chỉ ra cách cụ thể trong đó các bộ phận cấu thành của *thông điệp* được tổ chức. Các khả năng xử lý MIME là các khả năng điển hình hoặc các thỏa thuận được mô tả trong cây phụ này.

Phần tử **Packaging** (đóng gói) cung cấp thông tin về các kiểu nội dung MIME, các tên miền XML, các tham số an ninh và Cấu trúc MIME của dữ liệu được trao đổi giữa các *bên tham gia*.

Sau đây là một ví dụ về một phần tử **Packaging** (đóng gói) tham chiếu phần tử **SimplePart** (thành phần đơn giản) được đưa ra ở ví dụ trong phần 8.5:

```
<!-- Simple ebXML S/MIME Packaging for application-based payload
      encryption -->
<tp:Packaging>
  <tp:ProcessingCapabilities tp:generate="true" tp:parse="true"/>
```

```

<tp:CompositeList>
  <tp:Encapsulation
    <!-- I002 is the payload being encrypted -->
    tp:id="I003"
    tp:mimetype="application/pkcs7-mime"
    tp:mimeparameters="smime-type=&quot;enveloped-data&quot;"
    <Constituent tp:idref="I002"/>
  </tp:Encapsulation>
  <tp:Composite tp:id="I004"
    <!-- I001 is the SOAP envelope. The ebXML message is made
      up of the SOAP envelope and the encrypted payload. -->
    tp:mimetype="multipart/related"
    tp:mimeparameters="type=&quot;text/xml&quot;
version=&quot;1.0&quot;">
    <tp:Constituent tp:idref="I001"/>
    <tp:Constituent tp:idref="I003"/>
  </tp:Composite>
</tp:CompositeList>
</tp:Packaging>

```

Phần tử **Packaging** (đóng gói) có một thuộc tính; thuộc tính **ID YÊU CẦU**, với kiểu ID. Nó dẫn tới phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này), bằng việc sử dụng thuộc tính **IDREF** (tham chiếu id), **packageld** (id gói).

Các phần tử con của phần tử **Packaging** (đóng gói) là **ProcessingCapabilities** (khả năng xử lý) và **CompositeList**. (danh sách phần tử hỗn hợp). Tập các phần tử này có thể xuất hiện một hoặc nhiều lần như một phần tử con của mỗi phần tử **Packaging** (đóng gói).

6.6.1 Phần tử ProcessingCapabilities (Khả năng xử lý)

Phần tử **ProcessingCapabilities** (khả năng xử lý) có hai thuộc tính YÊU CẦU với giá trị nhị phân các giá trị là "true" hoặc "false". Các thuộc tính là **parse** (phân tách) và **generate** (tạo). Thông thường, các thuộc tính này sẽ có giá trị "true" để chỉ ra việc đóng gói các cấu trúc được quy định trong các phần tử con khác có thể được đưa ra cũng như xử lý tại phần mềm lớp dịch vụ *thông điệp*.

Ít nhất một giá trị của các thuộc tính **generate** (tạo) hoặc **parse** (phân tách) BẮT BUỘC là "true" (đúng).

6.6.2 Phần tử CompositeList (Danh sách phần tử hỗn hợp)

Phần tử con cuối cùng của phần tử **Packaging** (đóng gói) là **CompositeList** (danh sách phần tử hỗn hợp), đó là một thiết bị mang đối với cách cụ thể trong các phần đơn giản được kết hợp vào trong các nhóm (đa thành phần MIME) hoặc được rút gọn trong an ninh liên quan các kiểu nội dung MIME. Phần tử **CompositeList** (danh sách phần tử hỗn hợp) PHẢI được lược bỏ khỏi **Packaging** khi sử dụng không tóm lược an ninh hoặc đa thành phần hỗn hợp. Khi phần tử **CompositeList** (danh sách phần tử hỗn hợp) có mặt, mô hình nội dung đối với phần tử **CompositeList** (danh sách phần tử hỗn hợp) là một trình tự lặp lại các lựa chọn của Cửa các phần tử **Composite** (hỗn hợp) hoặc **Encapsulation** (đóng gói).

Các phần tử **Composite** (hỗn hợp) và **Encapsulation** (đóng gói) có thể xuất hiện được trộn lẫn như mong đợi. Trình tự theo các lựa chọn có được là quan trọng bởi vì ký tự đệ quy cho trước của **packaging**

(đóng gói), **composites** (hỗn hợp) hoặc **encapsulations** (sự đóng gói) MIME có thể bao gồm các hỗn hợp trước đó (hoặc rất hiếm khi, **encapsulations** (sự đóng gói)) bổ sung vào các phần *thông điệp* được mô tả trong cây phụ **SimplePart** (phần đơn giản). Vì vậy, đóng gói "top-level" (mức đỉnh) sẽ được mô tả cuối cùng trong trình tự chuỗi.

Phần tử **Composite** (hỗn hợp) có các thuộc tính sau:

- một thuộc tính **mimeType** (kiểu MIME) YÊU CẦU;
- một thuộc tính **ID** YÊU CẦU;
- một thuộc tính **mimeparameters** (tham số MIME yêu cầu) MẶC NHIÊN.

Thuộc tính **mimeType** (kiểu MIME) cung cấp giá trị của kiểu nội dung MIME đối với phần thông điệp này và điều này sẽ là một số kiểu hỗn hợp MIME, như "multipart/related" hoặc "multipart/signed". Thuộc tính **ID**, kiểu ID, cung cấp một cách dẫn đến hỗn hợp này. Nếu cần được đề cập như một thành phần cấu thành của một số phần tử trong chuỗi. Thuộc tính **mimeparameters** (tham số MIME yêu cầu) cung cấp các giá trị của tham số MIME có nghĩa nào đó (như "type=application/xml") điều đó cần thiết để hiểu nhu cầu xử lý của kiểu nội dung.

Phần tử **Composite** (hỗn hợp) có một phần tử con, **Constituent** (thành phần hợp thành).

Phần tử **Constituent** (thành phần hợp thành) có một thuộc tính YÊU CẦU, **idref** (tham chiếu id) của kiểu IDREF, một thuộc tính nhị phân MẶC NHIÊN **excludeFromSignature** (loại bỏ khỏi chữ ký) và hai thuộc tính **nonNegativeInteger** MẶC NHIÊN (số nguyên không âm), **minOccurs** (số lần xuất hiện nhỏ nhất) và **maxOccurs** (số lần xuất hiện lớn nhất).

Thuộc tính **IDREF** (tham chiếu id) có giá trị của nó như Giá trị thuộc tính **ID** của một **Composite**, **Encapsulation** hoặc phần tử **SimplePart** (thành phần đơn giản) trước đó. Mục đích của trình tự các **Constituent** (thành phần hợp thành) là để chỉ ra cả các nội dung và thứ tự của thuộc tính được đóng gói trong **Composite** (hỗn hợp) hoặc **Encapsulation** (sự đóng gói) hiện tại.

Thuộc tính **excludeFromSignature** (loại bỏ khỏi chữ ký) chỉ ra rằng **Constituent** (thành phần hợp thành) này được chứa như một phần của chữ ký *thông điệp* ebXML [XMLDSIG]. Nói cách khác, chữ ký được tạo ra bởi trình quản lý dịch vụ *thông điệp* nên không bao gồm một phần tử **ds:Reference** để cung cấp một hệ liệt kê cho **Constituent** (thành phần hợp thành) của thông điệp này. Thuộc tính này chỉ có thể áp dụng nếu **Constituent** (thành phần hợp thành) là một phần của **Composite** (hỗn hợp) mức đỉnh điều đó tương ứng với toàn bộ *thông điệp* ebXML.

Các thuộc tính **minOccurs** và **maxOccurs** dành để quy định giá trị hoặc dải giá trị dẫn tới hạng mục có thể xảy ra trong **Composite**. Khi không được sử dụng, hiểu rằng hạng mục được sử dụng đúng một lần.

Phần tử **Encapsulation** được dùng một cách điển hình để chỉ ra việc sử dụng các cơ chế an ninh của MIME, như [S/MIME] hoặc Open-PGP[RFC2015]. Một phần thân an ninh có thể tóm lược một phần MIME đã được mô tả trước đó. Để thuận tiện, tất cả các cấu trúc an ninh như vậy chịu ảnh hưởng phần

TCVN ISO/TS 15000-1 : 2007

tử **Encapsulation** (sự đóng gói), thậm chí khi nói một cách kỹ thuật dữ liệu đó không phải "inside" phần thân đó. (Nói cách khác, các cấu trúc ký được gọi là được ký rõ ràng hoặc không lệ thuộc với MIME đa phần/ được ký được tìm thấy một cách đơn giản thuộc phần tử **Encapsulation** (sự đóng gói).

Một cách sử dụng có thể khác của phần tử **Encapsulation** (sự đóng gói) là để trình bày ứng dụng của một thuật toán nén như gzip [ZLIB] đối với cùng các bộ phận của thiết bị mang, trước khi nó được mật mã hóa và hoặc ký.

Phần tử **Encapsulation** có các thuộc tính sau:

- một thuộc tính **mimeType** (kiểu MIME) YÊU CẦU;
- một thuộc tính **ID** YÊU CẦU;
- một thuộc tính **mimeparameters** (tham số MIME yêu cầu) MẶC NHIÊN.

Thuộc tính **mimeType** (kiểu MIME) cung cấp giá trị của kiểu nội dung MIME đối với phần thông điệp này, như "application/pkcs7-mime". Thuộc tính ID, type ID, cung cấp một cách để dẫn tới sự đóng gói này nếu nó cần được đề cập như một cấu thành của một số phần tử sau đó theo trình tự. Thuộc tính **mimeparameters** (tham số MIME yêu cầu) cung cấp các giá trị của tham số MIME có nghĩa nào đó cần thiết để hiểu các nhu cầu xử lý của kiểu nội dung.

Cả phần tử **Encapsulation** (sự đóng gói) và phần tử **Composite** (hỗn hợp) có các phần tử con bao gồm một phần tử **Constituent** (thành phần hợp thành) hoặc của một trình tự lặp lại của các phần tử **Constituent** (thành phần hợp thành), theo trình tự.

Phần tử **Constituent** (thành phần hợp thành) cũng có không hoặc một phần tử con **SignatureTransform** (phép biến đổi chữ ký) và không hoặc một phần tử con **EncryptionTransform** (phép biến đổi mã hóa). Phần tử **SignatureTransform** (phép biến đổi chữ ký) được dự kiến cho việc sử dụng với chữ ký dạng số XML [XMLDSIG]. Nếu có mặt, thì Nó xác định các biến đổi mà có thể được áp dụng cho dữ liệu nguồn trước khi một digest được tính toán phần tử **EncryptionTransform** (phép biến đổi mã hóa) được dự kiến cho việc sử dụng mật mã hóa XML [XMLENC]. Nếu có mặt, thì nó xác định các phép biến đổi có thể được áp dụng cho một **CipherReference** (tham chiếu mật mã) trước khi giải mật mã hóa có thể được thực hiện. Phần tử **SignatureTransforms** (phép biến đổi chữ ký) và phần tử **EncryptionTransforms** (phép biến đổi mật mã) mỗi bao gồm một hoặc nhiều phần tử **ds:Transform** [XMLDSIG].

6.7 Phần tử Signature (Chữ ký)

Phần tử **Signature** (chữ ký) (Các yếu tố trong tập hợp 0 hoặc 1) cho phép CPA đó ký dạng số có sử dụng công nghệ phù hợp với quy định chữ ký dạng số XML [XMLDSIG]. Phần tử **Signature** (chữ ký) là gốc của một cây phụ các phần tử được sử dụng để ký CPP. Cú pháp đó là:

```
<tp:Signature>...</tp:Signature>
```


Phần tử **Signature** (chữ ký) bao gồm một hoặc nhiều phần tử **ds:Signature**. Nội dung của phần tử **ds:Signature** và mọi phần tử con được xác định bởi quy định chữ ký dạng số XML. Chi tiết hơn xem phần 9.9.

CHÚ THÍCH: Cần thiết phải bao trùm phần tử **ds:Signature** với một phần tử **Signature** (chữ ký) trong tên miền đích để cho phép khả năng có phần tử **wildcard** (đại diện) (with namespace="##other") trong phần tử **CollaborationProtocolProfile** (hồ sơ giao thức hợp tác) và **CollaborationProtocolAgreements** (thỏa thuận giao thức hợp tác). Mô hình nội dung tránh nhầm lẫn với việc bao trùm này.

Các bắt buộc bổ sung sau đây về **ds:Signature** buộc phải tuân theo:

- một **CPP** **BẮT BUỘC** phải xem xét tính không hợp lệ nếu có phần tử **ds:Signature** không đủ sự kiểm tra hợp lệ chính như được xác định bởi định danh tài liệu số XML[XMLDSIG];
- ngay khi một **CPP** được ký, mỗi phần tử **ds:Reference** trong một phần tử **ProcessSpecification** (quy định quá trình) **BẮT BUỘC** chuyển qua kiểm tra tính hợp lệ tham chiếu và mỗi phần tử **ds:Signature** **BẮT BUỘC** chuyển qua kiểm tra hợp lệ.

CHÚ THÍCH: Trong trường hợp một **CPP** không được ký, phần mềm dù sao cũng có thể kiểm tra tính hợp lệ phần tử **ds:Reference** trong phần tử **ProcessSpecification** (quy định quá trình)s và báo cáo bất kỳ sự loại trừ nào.

CHÚ THÍCH: Phần mềm tạo ra các **CPP** và **CPA** có thể nhận dạng **ds:Signature** và chèn tự động cấu trúc phần tử đó cần thiết để xác định việc ký của **CPP** và **CPA**. Tạo chữ ký đưa ra vấn đề trong phần 9.9.1.1; các chi tiết của quá trình mật mã hóa nằm ngoài phạm vi của tiêu chuẩn này.

CHÚ THÍCH: Xem chú thích tham khảo trong phần 8.4.4.5 đối với một thảo luận trong lần kiểm tra tính hợp lệ có thể được tiến hành.

6.8 Phần tử Comment (Chú giải)

Phần tử **CollaborationProtocolProfile** (hồ sơ giao thức hợp tác) bao gồm không hoặc nhiều phần tử **Comment** (chú giải). Phần tử **Comment** (chú giải) là một chú thích nguyên văn có thể được bổ sung để dành cho bất kỳ mục đích nào mà tác giả mong muốn. Ngôn ngữ của **Comment** được xác định bởi một thuộc tính **xml:lang** **YÊU CẦU**.

Thuộc tính **xml:lang** **BẮT BUỘC** tuân theo các quy tắc về việc xác định các ngôn ngữ cụ thể trong [XML]. Nếu nhiều phần tử **Comment** (chú giải) có mặt, mỗi có thể có một giá trị thuộc tính **xml:lang** khác nhau. Một ví dụ về một phần tử **Comment** (chú giải) như sau:

```
<tp:Comment xml:lang="en-US">đây là một CPA giữa A và B</tp:Comment>
```

Khi một **CPA** bao gồm từ hai **CPP**, tất cả phần tử **Comment** (chú giải) từ cả hai **CPP** phải được chứa trong **CPA** đó trừ khi hai **bên tham gia** có thỏa thuận khác.

7 Định nghĩa CPA

Một *thỏa thuận giao thức hợp tác* (*Collaboration Protocol Agreement*) (**CPA**) xác định các khả năng mà hai *bên tham gia* cần thỏa thuận để cho phép họ tham gia vào *kinh doanh* điện tử đối với các mục đích của **CPA** riêng.

TCVN ISO/TS 15000-1 : 2007

Phần này định nghĩa và thảo luận các chi tiết của CPA đó. Các thảo luận được minh họa với một số đoạn XML.

Hầu hết các phần tử XML trong phần này được mô tả chi tiết trong phần 8, "Định nghĩa của CPP". Nói chung, phần này không lặp lại thông tin đó. Thảo luận trong phần này được giới hạn các phần tử không nằm trong CPP hoặc đối với việc thảo luận bổ sung cần thiết trong nội dung CPA đó. Cũng xem Phụ lục D đối với Giải đồ XML và Phụ lục B đối với một ví dụ về một tài liệu CPA.

7.1 Cấu trúc CPA

Sau đây là cấu trúc tổng thể của CPA đó:

```
<CollaborationProtocolAgreement
  xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-
2_0.xsd"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xlink="http://www.w3.org/1999/xlink"
    tp:cpaid="YoursAndMyCPA"
    tp:version="2.0a">
  <tp:Status tp:value="proposed"/>
  <tp:Start>1988-04-07T18:39:09</Start>
  <tp:End>1990-04-07T18:40:00</End>
  <!-- ConversationConstraints MAY appear 0 or 1 time -->
    <tp:ConversationConstraints
      tp:invocationLimit="100"
      tp:concurrentConversations="4"/>
  <tp:PartyInfo>
  ...
  </tp:PartyInfo>
  <tp:PartyInfo>
  <tp:PartyInfo>
  ...
  </tp:PartyInfo>
  <tp:SimplePart tp:id="..."> <!-- one or more -->
  ...
  </tp:SimplePart>
  <tp:Packaging tp:id="..."> <!-- one or more -->
  ...
  </tp:Packaging>
  <tp:Signature> <!-- zero or one time -->
  ...
  </tp:Signature>
  <tp:Comment xml:lang="en-GB">any text</Comment> <!-- zero or more -->
</tp:CollaborationProtocolAgreement>
```

7.2 Phần tử CollaborationProtocolAgreement (Thỏa thuận giao thức hợp tác)

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) phần tử gốc của một CPA. Nó có một thuộc tính **cpaid** (id của CPA) YÊU CẦU để cung cấp một định danh duy nhất đối với tài liệu. Giá trị thuộc tính **cpaid** (id của CPA) PHẢI được ấn định bởi một *bên tham gia* và được sử dụng bởi cả hai bên. Khuyến cáo rằng giá trị thuộc tính **cpaid** (id của CPA) là một URI. Giá trị thuộc tính **cpaid** (id của CPA) PHẢI được sử dụng như giá trị của phần tử CPAID đó trong *tiêu đề thông điệp ebXML[ebMS]* hoặc của một phần tử tương tự trong một *tiêu đề thông điệp* của một dịch vụ truyền *thông điệp* thay thế.

CHÚ THÍCH: Mỗi *bên tham gia* có thể liên kết một định danh cục bộ với thuộc tính **cpaid** (id của CPA) đó.

Ngoài ra, phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) có một thuộc tính **version** (phiên bản) YÊU CẦU. Thuộc tính này chỉ phiên bản của giản đồ đó mà CPA đó phù hợp. Giá trị thuộc tính **version** (phiên bản) NÊN là một chuỗi như "2_0a", "2_0b", v..v.

CHÚ THÍCH: Phương pháp của việc ấn định các giá trị **cpaid** (id của CPA) duy nhất được cho phép thực hiện.

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) có các khai báo [XML] tên miền [XMLNS] YÊU CẦU được xác định trong phần 8, "Định nghĩa của CPP".

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) bao gồm các phần tử con sau đây, mà phần lớn được mô tả chi tiết hơn trong các phần tiếp theo:

- một phần tử **Status** (trạng thái) YÊU CẦU để xác định trạng thái của quá trình tạo ra CPA đó;
- một phần tử **Start** (bắt đầu) YÊU CẦU để ghi lại ngày tháng và thời gian mà CPA đó bắt đầu chịu ảnh hưởng;
- một phần tử **End** (kết thúc) YÊU CẦU để ghi lại ngày tháng và thời gian sau khi CPA đó BẮT BUỘC được thương lượng lại bởi các *bên tham gia*;
- không hoặc một phần tử **ConversationConstraints** (quy định hội thoại) để lập tài liệu các thỏa thuận nào đó về việc xử lý hội thoại;
- hai phần tử **PartyInfo** (thông tin bên tham gia) YÊU CẦU, một phần tử cho mỗi *bên tham gia* vào CPA đó;
- một hoặc nhiều phần tử **SimplePart** (thành phần đơn giản);
- một hoặc nhiều phần tử **Packaging** (đóng gói);
- không hoặc một phần tử **Signature** (chữ ký) để cung cấp đối với việc ký của CPA đó có sử dụng tiêu chuẩn chữ ký dạng số XML[XMLDSIG];
- không hoặc nhiều phần tử **Comment** (chú giải).

7.3 Phần tử Status (Trạng thái)

Phần tử **Status** (trạng thái) ghi lại trạng thái của quá trình thương lượng/tạo kết cấu để tạo ra CPA đó. một ví dụ về phần tử **Status** (trạng thái) sau đây:

```
<tp:Status tp:value="proposed"/>
```

Phần tử **Status** (trạng thái) có một thuộc tính value (giá trị) YÊU CẦU để ghi lại trạng thái hiện tại của thành phần cấu tạo của CPA đó. Thuộc tính này là một kiểu số đếm bao gồm các giá trị có thể sau đây:

- "proposed", có nghĩa là CPA đó vẫn đang được thương lượng bởi các *bên tham gia*;
- "agreed", có nghĩa là các nội dung của CPA đó đã được thỏa thuận bởi cả hai *bên tham gia*;
- "signed", có nghĩa là CPA đó đã được "ký" bởi một hoặc nhiều *bên tham gia*. Việc "ký" này đưa ra mẫu của một chữ ký dạng số được mô tả trong phần 9.7 sau.

TCVN ISO/TS 15000-1 : 2007

CHÚ THÍCH: Phần tử **Status** (trạng thái) có thể được sử dụng bởi một thành phần cấu tạo của CPA và công cụ thương lượng để trợ giúp ít trong quá trình xây dựng một CPA.

CHÚ THÍCH: Giá trị thuộc tính value (giá trị) của phần tử **Status** (trạng thái) được thiết lập là "signed" trước khi *bên tham gia* thứ nhất ký. Thậm chí việc loại trừ thuộc tính value (giá trị) từ một chữ ký có thể thực hiện được về kỹ thuật, có thể ưu tiên thay đổi Giá trị thuộc tính là "signed" trước chữ ký đầu tiên và duy trì nó là "signed" đối với tất cả các chữ ký tiếp theo.

7.4 Khoảng thời gian của CPA

Khoảng thời gian của CPA được cho trước bởi các phần tử **Start** (bắt đầu) và **End**. Cú pháp là:

```
<tp:Start>1988-04-07T18:39:09Z</tp:Start>  
<tp:End>1990-04-07T18:40:00Z</tp:End>
```

7.4.1 Phần tử Start (Bắt đầu)

Phần tử **Start** (bắt đầu) quy định ngày tháng và thời gian bắt đầu của CPA đó. Phần tử **Start** (bắt đầu) PHẢI là một chuỗi giá trị phù hợp với nội dung mô hình của một kiểu dateTime chuẩn như được định nghĩa trong Giản đồ quy định các kiểu dữ liệu XML[XMLSCHEMA-2]. Ví dụ, để chỉ ra 1:20 pm UTC (Coordinated Universal Time) vào ngày 31 tháng 5 năm 1999, một phần tử **Start** (bắt đầu) phải có giá trị như sau:

```
1999-05-31T13:20:00Z
```

Phần tử **Start** (bắt đầu) PHẢI được biểu diễn như Coordinated Universal Time (UTC).

7.4.2 Phần tử End (Kết thúc)

Phần tử **End** (kết thúc) quy định ngày tháng và thời gian kết thúc của CPA đó. Phần tử **End** (kết thúc) PHẢI là một chuỗi giá trị để phù hợp với nội dung mô hình của một kiểu ngày giờ chuẩn như được định nghĩa trong giản đồ quy định các kiểu dữ liệu XML[XMLSCHEMA-2]. Ví dụ, để chỉ ra 1:20 pm UTC (Coordinated Universal Time) on May 31, 1999, một phần tử **End** (kết thúc) phải có giá trị sau đây:

```
1999-05-31T13:20:00Z
```

Phần tử **End** (kết thúc) PHẢI được biểu diễn như Coordinated Universal Time (UTC).

Khi đạt tới thời điểm kết thúc của khoảng thời gian CPA đó, mọi *giao dịch kinh doanh* đang tiến hành PHẢI được phép hoàn thành và không có *giao dịch kinh doanh* mới nào PHẢI được bắt đầu.

Khi tất cả *giao dịch kinh doanh* đang tiến hành trên mỗi hội thoại được hoàn thành, đối thoại này PHẢI được kết thúc hoặc không hoàn thành.

Khi một CPA được ký, phần mềm để ký các thỏa thuận PHẢI cảnh báo trước nếu có kiểm tra tính hợp lệ của chứng chỉ ký hết hạn trước khi thời gian được đề nghị để kết thúc CPA đó. Cơ hội để thương lượng lại một giá trị **End** của CPA hoặc để trong một số cách khác hài hòa khoảng thời gian kiểm tra tính hợp lệ của chứng chỉ với khoảng thời gian kiểm tra tính hợp lệ của CPA PHẢI có thể thực hiện. (Các phương

pháp khác để căn chỉnh các khoảng thời gian hợp lệ sẽ bao gồm việc tái ấn hành quá trình ký các chứng chỉ đối với một khoảng thời gian dài hơn hoặc đạt được các chứng chỉ mới cho mục đích này).

Phần mềm ký cũng NÊN chấp nhận căn chỉnh các khoảng thời gian hợp lệ của các chứng chỉ để cập tới CPA đó thực hiện các chức năng an ninh để không hết hạn trước khi CPA đó hết hạn. Việc căn chỉnh này có thể xuất hiện theo nhiều cách bao gồm việc sử dụng **ds:RetrievalMethod** của mô hình nội dung của **ds:KeyInfo** vì vậy một chứng chỉ mới có thể được cài đặt và vẫn có thể đạt được một cách phù hợp với thông tin trong **ds:RetrievalMethod**. Nếu không đạt được sự căn chỉnh, phần mềm ký BẮT BUỘC cảnh báo cho người sử dụng trường hợp tính hợp lệ của CPA vượt quá tính hợp lệ của một số trường hợp của các chứng chỉ để cập tới CPA đó.

CHÚ THÍCH: Nếu một ứng dụng *kinh doanh* xác định một hội thoại bao gồm nhiều *giao dịch kinh doanh*, một hội thoại như vậy có thể được kết thúc mà không có chỉ dẫn lỗi khi đạt tới thời điểm kết thúc của khoảng thời gian. Hệ thống thời gian thực đó có thể cung cấp một chỉ dẫn lỗi cho ứng dụng.

CHÚ THÍCH: Nó có thể không thực hiện được để đợi các vấn đề chưa giải quyết các hội thoại để kết thúc trước khi CPA đó kết thúc do không có giới hạn về thời gian bao lâu một hội thoại có thể kéo dài.

CHÚ THÍCH: Hệ thống thời gian thực đó NÊN phản hồi lại một chỉ dẫn lỗi tới cả hai *bên tham gia* khi một *giao dịch kinh doanh* mới được bắt đầu trong CPA này sau khi ngày tháng và thời gian được quy định trong phần tử **End**.

7.5 Phần tử ConversationConstraints (Quy định hội thoại)

Phần tử **ConversationConstraints** (quy định hội thoại) đặt giới hạn số các hội thoại chịu tác động của CPA đó. một ví dụ về phần tử này như sau:

```
<tp:ConversationConstraints tp:invocationLimit="100"
tp:concurrentconversations="4"/>
```

Phần tử **ConversationConstraints** (quy định hội thoại) có các thuộc tính sau:

- một thuộc tính **invocationLimit** (giới hạn viện dẫn chứng) MẶC NHIÊN;
- một thuộc tính **concurrentconversations** (các hội thoại đồng thời) MẶC NHIÊN.

7.5.1 Thuộc tính invocationLimit (Giới hạn viện dẫn chứng)

Thuộc tính **invocationLimit** (giới hạn viện dẫn chứng) xác định số lớn nhất của các hội thoại có thể được xử lý trong CPA đó. Khi đạt tới số giới hạn này, CPA đó được kết thúc và BẮT BUỘC được thương lượng lại. Nếu không giá trị nào được quy định, Không có giới hạn trên của số các hội thoại và khoảng thời gian của CPA đó được điều khiển đơn bởi phần tử **End** (kết thúc).

CHÚ THÍCH: Thuộc tính **invocationLimit** (giới hạn viện dẫn chứng) thiết lập một giới hạn trên số các đơn vị công việc *kinh doanh* có thể được thực hiện thuộc CPA đó. Nó là một tham số *kinh doanh*, không một tham số đặc tính. Một CPA hết hạn bất cứ lúc nào khi điều kiện kết thúc được tiến tới đầu tiên (**End** (kết thúc) hoặc **invocationLimit** (giới hạn viện dẫn chứng)).

7.5.2 Thuộc tính concurrentconversations (Các hội thoại đồng thời)

Thuộc tính **concurrentconversations** (các hội thoại đồng thời) xác định số lớn nhất của các hội thoại có thể được trong quá trình thuộc CPA này tại cùng thời điểm. Nếu không giá trị nào được quy định, việc xử lý của các hội thoại cùng thời điểm hoàn toàn là một vấn đề cục bộ.

CHÚ THÍCH: Thuộc tính **concurrentconversations** (các hội thoại đồng thời) cung cấp một tham số đối với các *bên tham gia* để sử dụng khi cần thiết hạn chế số các hội thoại có thể được xử lý đồng thời trong một CPA riêng biệt. Ví dụ, quá trình phụ trợ (back-end) chỉ có thể hỗ trợ một số hạn chế các hội thoại cùng thời điểm. Nếu một yêu cầu đối với một hội thoại mới nhận được khi số lớn nhất của các hội thoại cho phép thuộc CPA này đang trong quá trình, một sự thi hành có thể bị loại bỏ hội thoại mới hoặc có thể sắp vào hàng yêu cầu cho đến khi một hội thoại hiện có kết thúc. Nếu không giá trị nào cho trước đối với **concurrentconversations** (các hội thoại đồng thời), cách xử lý một yêu cầu đối với một hội thoại mới để không có khả năng là một vấn đề thực thi nội bộ.

7.6 Phần tử PartyInfo (Thông tin bên tham gia)

Các đặc điểm chung của phần tử **PartyInfo** (thông tin bên tham gia) được trình bày trong phần 8.4.

CPA đó PHẢI có một phần tử Info (thông tin) của *bên tham gia* đối với mỗi *bên tham gia* với CPA đó.

Phần tử **PartyInfo** (thông tin bên tham gia) quy định các thuật ngữ đã thỏa thuận của các *bên tham gia* để mở rộng các *hồ sơ hợp tác kinh doanh* được định nghĩa bởi tài liệu *quy định quá trình* được tham chiếu bởi CPA. Nếu một CPP có hơn một phần tử Info (thông tin) của *bên tham gia*, phần tử **PartyInfo** (thông tin bên tham gia) thích hợp PHẢI được lựa chọn từ mỗi CPP khi soạn một CPA.

Trong CPA đó, PHẢI có một hoặc nhiều phần tử **PartyId** (ID bên tham gia) thuộc Mỗi phần tử Info (thông tin) của *bên tham gia*. Các giá trị của các phần tử này giống các giá trị của các phần tử **PartyId** (ID bên tham gia) trong quy định dịch vụ *thông điệp ebXML[ebMS]* hoặc tương tự quy định dịch vụ truyền *thông điệp*. Các phần tử **PartyId** (ID bên tham gia) này PHẢI được sử dụng trong một phần tử tiêu đề **To** hoặc **From** của một *thông điệp ebXML*.

7.6.1 Phần tử ProcessSpecification (Quy định quá trình)

Phần tử **ProcessSpecification** (quy định quá trình) xác định *hồ sơ hợp tác kinh doanh* để hai *bên tham gia* thỏa thuận thực hiện. CÓ THỂ có một hoặc nhiều phần tử **ProcessSpecification** (quy định quá trình) trong một CPA. Mỗi phần tử PHẢI là một phần tử con của một phần tử **CollaborationRole** (vai trò hợp tác) riêng. Xem phần 8.4.3.

7.7 Phần tử SimplePart (Thành phần đơn giản)

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) PHẢI bao gồm một hoặc nhiều phần tử **SimplePart** (thành phần đơn giản). Xem phần 8.5 để biết thêm chi tiết về cú pháp của phần tử **SimplePart** (thành phần đơn giản).

7.8 Phần tử **Packaging** (Đóng gói)

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) PHẢI bao gồm một hoặc nhiều phần tử **Packaging** (đóng gói). Xem phần 8.6 để biết thêm các chi tiết của cú pháp của phần tử **Packaging** (đóng gói).

7.9 Phần tử **Signature** (Chữ ký)

Một tài liệu **CPA** có thể ký dạng số bởi một hoặc nhiều *bên tham gia* như phương pháp đảm bảo tính toàn vẹn của nó cũng như một phương pháp diễn tả thỏa thuận cho một chữ k của nhân viên đại diện cho tổ chức đối với một tài liệu giấy. Nếu các chữ k đang được sử dụng ký dạng số một ebXML **CPA** hoặc tài liệu **CPP**, thì [XMLDSIG] PHẢI được sử dụng để ký dạng số tài liệu đó.

Phần tử **Signature** (chữ ký), nếu có mặt, thì được tạo ra từ một đến ba phần tử **ds:Signatures**. **CPA** đó có thể được ký bởi một hoặc cả hai *bên tham gia*. Khuyến cáo rằng cả hai *bên tham gia* ký **CPA** đó. Đối với việc ký bởi cả hai *bên tham gia*, một *bên tham gia* ký khởi tạo. Bên tham gia khác sau đó ký theo chữ ký của *bên tham gia* kia. **CPA** kết quả sau đó có thể được ký bởi một người làm chứng.

Phần tử **ds:Signature** là gốc của một cây phụ của các phần tử được sử dụng để ký **CPP**.

Nội dung của phần tử này và mọi phần tử con được xác định bởi định danh tài liệu số XML[XMLDSIG]. Các bắt buộc bổ sung sau đây về **ds:Signature** tuân theo:

- một **CPA** BẮT BUỘC phải xem xét tính không hợp lệ nếu có **ds:Signature** lỗi kỹ thuật sự hợp lệ như được xác định bởi định danh tài liệu số XML;
- bất kỳ khi nào một **CPA** được ký, mỗi **ds:Reference** trong một **ProcessSpecification** BẮT BUỘC vượt qua sự kiểm tra hợp lệ tham chiếu và mỗi **ds:Signature** BẮT BUỘC vượt qua sự kiểm tra hợp lệ kỹ thuật.

CHÚ THÍCH: Trong trường hợp một **CPA** không được ký, phần mềm có thể kiểm tra hợp lệ phần tử **ds:Reference** trong phần tử **ProcessSpecification** (quy định quá trình) và báo cáo bất kỳ sự loại trừ nào.

Phần mềm tạo ra các **CPP** và **CPA** PHẢI nhận dạng **ds:Signature** và chèn tự động cấu trúc phần tử đó cần thiết để xác định việc ký của **CPP** và **CPA**. Việc tạo chữ k, bản thân nó là một quá trình mật mã hóa và nằm ngoài phạm vi của tiêu chuẩn này.

CHÚ THÍCH: Xem chú thích tham khảo trong phần 8.4.4.5 đối với một thảo luận của số lần mà một **CPA** có thể được hợp lệ.

7.9.1 Chữ ký dạng số dài hạn

Nếu [XMLDSIG] được sử dụng để ký một ebXML **CPP** hoặc **CPA**, thì quá trình được xác định trong tiêu chuẩn này PHẢI được sử dụng.

7.9.1.1 Tạo chữ ký

Sau đây là các bước để tạo ra một chữ ký dạng số:

1. tạo một phần tử **SignedInfo**, một phần tử con của **ds:Signature**. **SignedInfo** PHẢI có các phần tử con **SignatureMethod**, **CanonicalizationMethod** và **Reference** như được quy định bởi [XMLDSIG];
2. tuân theo quy tắc và sau đó tính toán **SignatureValue** trên **SignedInfo** dựa trên cơ sở các thuật toán được quy định trong **SignedInfo** như được quy định trong [XMLDSIG];
3. xây dựng phần tử **Signature** (chữ ký) bao gồm các phần tử **SignedInfo**, **KeyInfo** (KHUYẾN CÁO) và **SignatureValue** như được quy định in [XMLDSIG];
4. bao gồm tên miền đó phần tử **Signature** (chữ ký) được hạn định đó chỉ trong tài liệu được ký, theo sau cùng là phần tử **PartyInfo** (thông tin bên tham gia).

7.9.1.2 Phần tử ds:SignedInfo

Phần tử **ds:SignedInfo** PHẢI bao gồm không hoặc một phần tử **ds:CanonicalizationMethod**, phần tử **ds:SignatureMethod** và Một hoặc nhiều phần tử **ds:Reference**.

7.9.1.3 Phần tử ds:CanonicalizationMethod

Phần tử **ds:CanonicalizationMethod** như được định nghĩa trong [XMLDSIG], có thể xuất hiện không hoặc một lần, có nghĩa là phần tử đó không cần xuất hiện trong một trường hợp của một phần tử **ds:SignedInfo**. Phương pháp phù hợp với quy tắc tiêu chuẩn được áp dụng cho dữ liệu được ký [XMLC14N] trong trường hợp vắng mặt một phần tử **ds:CanonicalizationMethod** để quy định trường hợp khác. Phương pháp mặc định này cũng PHẢI dành cho phương pháp phù hợp với quy tắc tiêu chuẩn đối với các tài liệu **CPP** và **CPA** của ebXML.

7.9.1.4 Phần tử ds:SignatureMethod

Phần tử **ds:SignatureMethod** PHẢI có mặt và PHẢI có một thuộc tính **Algorithm**. Giá trị Được Khuyến cáo đối với thuộc tính **Algorithm** là:

"http://www.w3.org/2000/09/xmlsig#sha1"

Giá trị KHUYẾN CÁO này PHẢI được hỗ trợ bởi tất cả trình thực hiện phần mềm **CPP** hoặc **CPA** của ebXML.

7.9.1.5 Phần tử ds:Reference

Phần tử **ds:Reference** đối với Tài liệu **CPP** hoặc **CPA** PHẢI có một Giá trị thuộc tính **URI YÊU CẦU** là "" để cung cấp cho chữ ký được áp dụng cho tài liệu bao gồm phần tử **ds:Signature** (tài liệu **CPA** đó hoặc **CPP**). Phần tử **ds:Reference** đối với tài liệu **CPP** hoặc **CPA** có thể bao gồm một thuộc tính **type** (kiểu) MẶC NHIÊN có một giá trị là:

"http://www.w3.org/2000/09/xmldsig#Object"

Phù hợp với [XMLDSIG]. Thuộc tính này đơn thuần chỉ là tham khảo. Nó có thể được lược bỏ. Việc thi thực thi của phần mềm được thiết kế cho người tạo ra hoặc quá trình một tài liệu CPA hoặc CPP ebXML PHẢI được chuẩn bị để quản lý các trường hợp khác. Phần tử **ds:Reference** có thể bao gồm thuộc tính **ID**, **type ID**, bởi phần tử **ds:Reference** được tham chiếu từ một phần tử **ds:Signature**.

7.9.1.6 Phần tử ds:Transform

Phần tử **ds:Reference** cho tài liệu CPA đó hoặc CPP PHẢI bao gồm một phần tử **ds:Transform** con để loại trừ việc bao gồm phần tử **ds:Signature** và tất cả phần tử con của nó. Việc loại trừ này đạt được bằng phương pháp quy định thuộc tính **ds:Algorithm** của phần tử **Transform** (truyền tải) là:

"http://www.w3.org/2000/09/xmldsig#enveloped-signature"

Ví dụ:

```
<ds:Reference ds:URI="">
  <ds:Transforms>
    <ds:Transform
      ds:Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    </ds:Transforms>
    <ds:DigestMethod
      ds:Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
```

7.9.1.7 Thuộc tính ds:Algorithm

Phần tử **ds:Transform** PHẢI bao gồm một thuộc tính **ds:Algorithm** có giá trị:

http://www.w3.org/2000/09/xmldsig#enveloped-signature

CHÚ THÍCH: Khi ký dạng số một CPA, khuyến cáo rằng mỗi *bên tham gia* ký tài liệu phù hợp với quá trình được mô tả ở trên.

Khi hai *bên tham gia* ký CPA đó, *bên tham gia* thứ nhất để ký CPA chỉ PHẢI ký các nội dung CPA đó, loại trừ các chữ tự ký. *Bên tham gia* thứ hai PHẢI ký thông qua các nội dung của CPA đó cũng như phần tử **ds:Signature** bao gồm chữ ký của *bên tham gia* thứ nhất. Nếu cần thiết, một người làm chứng có thể ký trên cơ sở cả hai chữ ký của các *bên tham gia*.

7.10 Phần tử Comment (Chú giải)

Phần tử **CollaborationProtocolAgreement** (thỏa thuận giao thức hợp tác) bao gồm không hoặc nhiều phần tử **Comment** (chú giải). Các chi tiết của cú pháp của phần tử **Comment** (chú giải) xem phần 8.8.

7.11 Soạn một CPA từ hai CPP

Phần này đề cập đến các ấn bản chuẩn trong việc soạn một CPA từ hai CPP. Xem Phụ lục E, "Thành phần cấu tạo của CPA (Tham khảo)".

7.11.1 Sao chép thuộc tính ID

Việc soạn một CPA từ hai CPP, có một rủi ro là các thuộc tính **ID** từ hai CPP có thể có các giá trị sao chép. Khi một CPA được soạn từ hai CPP, giá trị sao chép của thuộc tính ID PHẢI được thử nghiệm.

TCVN ISO/TS 15000-1 : 2007

Nếu một giá trị sao chép của thuộc tính **ID** có mặt, một trong các giá trị sao chép PHẢI đưa ra một giá trị mới và các giá trị tương ứng của thuộc tính **IDREF** (tham chiếu id) từ **CPP** tương ứng PHẢI chính đúng.

CHÚ THÍCH: Một *bên tham gia* có thể tìm kiếm để ngăn ngừa sự chuyển nhượng lại ID/IDREF trong **CPA** đó bằng việc lựa chọn các giá trị ID và IDREF gần như là duy nhất giữa các *bên tham gia* thương mại của nó. Ví dụ, phần tử **Certificate** (chứng chỉ) sau đây được tìm thấy trong một **CPP** có một thuộc tính **certId** (id của chứng chỉ) là đã có đặc điểm chung để nó có thể mâu thuẫn với một thuộc tính **certId** (id của chứng chỉ) tìm thấy trong một cộng tác **CPP** của *bên tham gia*:

```
<tp:Certificate tp:certId="EncryptionCert"><ds:KeyInfo/></tp:Certificate>
```

Để ngăn ngừa việc chuyển nhượng lại của ID này (và các IDREF liên quan) trong một **CPA**, một cách lựa chọn tốt hơn của **certId** trong **CPP** của công ty A **CPP** là:

```
<tp:Certificate tp:certId="CompanyA_EncryptionCert"><ds:KeyInfo/></tp:Certificate>
```

7.12 Sửa đổi các tham số tài liệu quy định quá trình dựa trên cơ sở thông tin trong CPA

Một tài liệu *quy định quá trình* bao gồm số các tham số, được diễn tả như các thuộc tính XML. Một ví dụ là các thuộc tính an ninh là các bản sao đối chiếu của các thuộc tính của phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) **CPA** đó. Các giá trị của các thuộc tính này có thể được xem xét là các giá trị mặc định hoặc các khuyến cáo.

Khi một **CPA** được tạo ra, các *bên tham gia* có thể quyết định chấp nhận các khuyến cáo trong *quy định-quá trình* hoặc họ có thể đồng ý về các giá trị của các tham số này để phản ánh tốt hơn các nhu cầu của họ.

Khi một **CPA** được sử dụng để lập cấu hình một hệ thống thời gian thực, các chọn lựa được quy định trong **CPA** đó BẮT BUỘC luôn khẳng định quyền ưu tiên trên các chọn lựa được quy định trong tài liệu *quy định quá trình* được tham chiếu. Cụ thể, tất cả các chọn lựa được diễn tả trong một phần tử **Packaging** (đóng gói) và **BusinessTransactionCharacteristics** (các đặc điểm giao dịch kinh doanh) của **CPA** BẮT BUỘC được thực hiện như đã thỏa thuận bởi các *bên tham gia*. Các chọn lựa này PHẢI ghi đè các giá trị mặc định được diễn tả trong tài liệu *quy định quá trình*. Quá trình cài đặt thông tin đó từ **CPA** và tài liệu *quy định quá trình* BẮT BUỘC kiểm tra rằng tất cả các chọn lựa kết quả nhất quán lẫn nhau và BẮT BUỘC báo hiệu một lỗi nếu chúng không nhất quán.

CHÚ THÍCH: Có nhiều cách ghi đè thông tin đó trong tài liệu *quy định quá trình* bởi thông tin từ **CPA** đó.

Ví dụ:

- thành phần cấu tạo của công cụ **CPA** có thể tạo ra một bản sao phân tách của tài liệu *quy định-quá trình*. Công cụ này sau đó có thể sửa đổi tài liệu *quy định quá trình* một cách trực tiếp với thông tin từ **CPA** đó. Một lợi ích của phương pháp này là quá trình ghi đè được thực hiện hoàn toàn bởi thành phần cấu tạo của công cụ **CPA**;

- một CPA công cụ cài đặt có thể ghi đè động các tham số trong tài liệu *quy định quá trình* có sử dụng thông tin từ các tham số tương ứng trong CPA đó tại thời điểm CPA và tài liệu *quy định quá trình* được cài đặt trong các hệ thống của các *bên tham gia*. Các loại trừ cần thiết này để tạo ra một bản sao phân tách của tài liệu *quy định quá trình*;
- các phương pháp có thể khác dựa trên cơ sở các phép biến đổi XSLT của thông tin tham số trong CPA đó và/hoặc tài liệu *quy định quá trình*.

8 Tài liệu tham khảo

[ccOVER] Tổng quan về thành phần chính của ebXML, <http://www.ebxml.org/specs/ccOVER.pdf>.

[DIGENV] Đường bao số, các phòng thử nghiệm RSA, <http://www.rsasecurity.com/rsalabs/faq/2-2-4.html>. CHÚ THÍCH: Tại thời điểm này, quy định sẵn có cho đường bao số là quy định các thư viện của RSA.

[ebBPSS] giản đồ quy định quá trình kinh doanh ebXML, <http://www.ebxml.org/specs/ebBPSS.pdf>.

[ebMS] quy định dịch vụ thông điệp ebXML, http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS_v2_0.pdf.

[ebRS] quy định dịch vụ đăng ký ebXML, <http://www.oasis-open.org/committees/regrep/documents/2.0/specs/ebrs.pdf>.

[HTTP] Giao thức truyền siêu văn bản, nhóm thiết kế mạng RFC 2616, <http://www.rfc-editor.org/rfc/rfc2616.txt>.

[IPSEC] bản đồ chỉ dẫn tài liệu an ninh IP, nhóm thiết kế mạng RFC 2411, <http://www.ietf.org/rfc/rfc2411.txt>.

[ISO6523] cấu trúc đối với định danh các tổ chức và các phần của tổ chức, tiêu chuẩn quốc tế ISO-6523.

[MIME] MIME (thư tín internet đa năng mở rộng) Phần 1: các cơ chế đối với việc quy định và mô tả dạng thức của các cơ quan truyền thông điệp internet. Nhóm thiết kế internet RFC 1521, <http://www.ietf.org/rfc/rfc1521.txt>.

[RFC959] giao thức truyền tệp (FTP), Nhóm thiết kế internet RFC 959, <http://www.ietf.org/rfc/rfc959.txt>.

[RFC1123] các yêu cầu đối với máy server (máy chủ) internet -- ứng dụng và hỗ trợ, Nhóm thiết kế internet RFC 1123, <http://www.ietf.org/rfc/rfc1123.txt>.

[RFC1579] FTP thân thiện-tường lửa, Nhóm thiết kế internet RFC 1579, <http://www.ietf.org/rfc/rfc1579.txt>.

[RFC2015] An ninh MIME tịn cậy đẹp mắt, Nhóm thiết kế internet, RFC 2015, <http://www.ietf.org/rfc/rfc2015.txt>.

[RFC2119] Các từ khóa cho việc sử dụng trong các RFC để chỉ ra các mức yêu cầu, Nhóm thiết kế internet RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>.

[RFC2246] Giao thức TLS, Nhóm thiết kế internet RFC 2246, <http://www.ietf.org/rfc/rfc2246.txt>.

[RFC2251] Giao thức truy cập danh mục ít quan trọng (v3), Nhóm thiết kế internet RFC 2251, <http://www.ietf.org/rfc/rfc2251.txt>.

[RFC2396] Các định danh nguồn thống nhất (URI): Cú pháp chung, Nhóm thiết kế internet RFC 2396, <http://www.ietf.org/rfc/rfc2396.txt>.

TCVN ISO/TS 15000-1 : 2007

[RFC2617] Xác thực HTTP: Xác thực tài liệu liệt kê và cơ sở, Nhóm thiết kế internet RFC 2617, <http://www.ietf.org/rfc/rfc2617.txt>.

[RFC2822] Dạng thức thông điệp internet, Nhóm thiết kế internet RFC 2822, <http://www.ietf.org/rfc/rfc2822.txt>.

[S/MIME] Quy định thông điệp S/MIME phiên bản 3, Nhóm thiết kế internet RFC 2633, <http://www.ietf.org/rfc/rfc2633.txt>.

[SAML] ngôn ngữ đánh dấu xác nhận an ninh, <http://www.oasis-open.org/committees/security/-documents>

[SMTP] giao thức truyền thư tín đơn giản, Nhóm thiết kế internet RFC 2821, <http://www.faqs.org/rfcs/rfc2821.html>.

Lớp khe cắm an ninh [SSL], Netscape Communications Corp., <http://www.netscape.com/eng/ssl3/>

CHÚ THÍCH: Tại thời điểm này, xuất hiện trong quy định của Netscape chỉ là các quy định sẵn có của SSL.

[X12] Tiêu chuẩn ANSI X12 cho trao đổi dữ liệu điện tử, X12 Standard Release 4050, December 2001.

[XAML] Ngôn ngữ đánh dấu bản quyền giao dịch (Transaction Authority Markup Language,) <http://xaml.org/>.

[XLINK] Ngôn ngữ liên kết XML (XML Linking Language), <http://www.w3.org/TR/xlink/>.

[XML] Ngôn ngữ đánh dấu mở rộng (XML), World Wide Web Consortium, <http://www.w3.org/XML>.

[XMLC14N] XML theo quy tắc tiêu chuẩn, Ver. 1.0, Worldwide Web Consortium, <http://www.w3.org/TR/2001/REC-xml-c14n-20010315>.

[XMLDSIG] Xử lý và cú pháp ký XML (XML Signature Syntax and Processing), Worldwide Web Consortium, <http://www.w3.org/TR/xmlsig-core/>.

[XMLENC] Xử lý và mật mã hóa XML (XML Encryption Syntax và Processing), Worldwide Web Consortium, <http://www.w3.org/TR/2002/CR-xmlenc-core-20020304/>.

[XMLNS] Các tên miền trong (Namespaces in XML), Worldwide Web Consortium, <http://www.w3.org/TR/REC-xml-names/>. [XMLSCHEMA-1] Giản đồ XML phần 1: Structures, Worldwide Web Consortium, <http://www.w3.org/TR/xmlschema-1/>.

[XMLSCHEMA-2] Giản đồ XML phần 2: Kiểu dữ liệu, Worldwide Web Consortium, <http://www.w3.org/TR/xmlschema-2/>.

[XPOINTER] Ngôn ngữ con trỏ trong XML (XML Pointer Language), Worldwide Web Consortium, <http://www.w3.org/TR/xptr/>.

[ZLIB] Zlib: Massively Spiffy Yet Delicately Unobtrusive Compression Library, <http://www.gzip.org/zlib/>.

9 Sự phù hợp

Để phù hợp với tiêu chuẩn này, một thực hiện:

- a) PHẢI hỗ trợ toàn bộ các yêu cầu giao tiếp và chức năng được định nghĩa trong tiêu chuẩn này,
- b) KHÔNG PHẢI quy định mọi yêu cầu mâu thuẫn hoặc gây ra sự không-phù hợp với tiêu chuẩn này.

Một sự thi hành phù hợp PHẢI thỏa mãn các yêu cầu về sự phù hợp với các phần có thể của tiêu chuẩn này.

Sự thi hành của một công cụ hoặc dịch vụ để tạo ra hoặc duy trì các tài liệu về trường hợp *CPP* hoặc *CPA* của ebXML PHẢI được xác định là phù hợp bởi sự kiểm tra tính hợp lệ của các tài liệu về trường hợp *CPP* hoặc *CPA* đó, được tạo ra hoặc sửa đổi bởi công cụ và dịch vụ đặc nói, trái với định nghĩa của giản đồ XML [XMLSCHEMA-1] của *CPP* hoặc *CPA* trong Phụ lục D và sẵn có tại:

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd bằng việc sử dụng hai hoặc nhiều phép kiểm tra tính hợp lệ của các giản đồ phân tích cú pháp XML để phù hợp với các quy định giản đồ W3C XML [XMLSCHEMA-1, XMLSCHEMA-2].

Mục tiêu của các thử nghiệm sự phù hợp là để xác định sự thi hành đang được thử nghiệm có phù hợp với các yêu cầu đã khẳng định trong tiêu chuẩn này hay không. phép thử nghiệm sự phù hợp cho phép các nhà cung cấp thi hành các hệ thống có khả năng cùng làm việc và tương thích. Các thực thi và ứng dụng PHẢI được thử nghiệm có sử dụng các phép thử có sẵn phù hợp với tiêu chuẩn này.

Thử nghiệm sẵn có công khai phù hợp với các tổ chức trung lập cung cấp như OASIS và Viện khoa học và công nghệ quốc gia U.S.A. (NIST) NÊN được sử dụng để kiểm tra sự phù hợp của các thực thi, ứng dụng và các thành phần khẳng định phù hợp với tiêu chuẩn này. Các thực thi tham khảo mã nguồn mở sẵn có để cho phép các nhà cung cấp thử nghiệm sản phẩm của họ đối với khả năng cùng làm việc, sự phù hợp, tính tương thích của giao diện.

10 Thông tin về điểm liên lạc

Arvola Chan (Author)

TIBCO Software

3303 Hillview Avenue

Palo Alto, CA 94304

USA

Phone: 650-846-5046

email: <mailto:arvola@tibco.com>

Dale W. Moberg (Author)

Cyclone Commerce

8388 E. Hartford Drive

Scottsdale, AZ 85255

USA

Phone: 480-627-2648

email: <mailto:dmoberg@cyclonecommerce.com>

TCVN ISO/TS 15000-1 : 2007

Himagiri Mukkamala (Author)

Sybase Inc.

5000 Hacienda Dr

Dublin, CA, 84568

USA

Phone: 925-236-5477

email: <mailto:himagiri@sybase.com>

Peter M. Ogden (Author)

Cyclone Commerce, Inc.

8388 East Hartford Drive

Scottsdale, AZ 85255

USA

Phone: 480-627-1800

email: <mailto:pogden@cyclonecommerce.com>

Martin W. Sachs (Author)

IBM T. J. Watson Research Center

P.O.B. 704

Yorktown Hts, NY 10598

USA

Phone: 914-784-7287

email: <mailto:mwsachs@us.ibm.com>

Tony Weida (Coordinating Editor)

535 West 110th St., #4J

New York, NY 10025

USA

Phone: 212-678-5265

email: <mailto:rweida@hotmail.com>

Jean Zheng

Vitria 945 Stewart Drive Sunnyvale, CA 94086 USA Phone: 408-212-2468 email:
<mailto:jzheng@vitria.com>

Phụ lục A

(tham khảo)

Ví dụ về Tài liệu

Ví dụ này bao gồm hai CPP được sử dụng để tạo biểu mẫu CPA trong Phụ lục B. Chúng sẵn có dưới dạng tệp ASCII tại

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-example-companyA-2_0b.xml

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-example-companyB-2_0b.xml

cpp-example-companyA-2_0b.xml:

```
<?xml version="1.0"?>
<!-- Copyright UN/CEFACT and OASIS, 2001. All Rights Reserved. -->
<tp:CollaborationProtocolProfile
  xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
    cpp-cpa-2_0.xsd"
  tp:cppid="uri:companyA-cpp" tp:version="2_0b">
  <!-- Party info for CompanyA-->
  <tp:PartyInfo
    tp:partyName="CompanyA"
    tp:defaultMshChannelId="asyncChannelA1"
    tp:defaultMshPackageId="CompanyA_MshSignalPackage">
    <tp:PartyId
      tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
    <tp:PartyRef xlink:href="http://CompanyA.com/about.html"/>
    <tp:CollaborationRole>
      <tp:ProcessSpecification
        tp:version="2.0"
        tp:name="PIP3A4RequestPurchaseOrder"
        xlink:type="simple"
        xlink:href="http://www.rosettanet.org/processes/3A4.xml"
        tp:uuid="urn:icann:rosettanet.org:bpid:3A4$2.0"/>
      <tp:Role
        tp:name="Buyer"
        xlink:type="simple"
        xlink:href="http://www.rosettanet.org/processes/3A4.xml#Buyer"/>
      <tp:ApplicationCertificateRef tp:certId="CompanyA_AppCert"/>
      <tp:ServiceBinding>
        <tp:Service>bpid:icann:rosettanet.org:3A4$2.0</tp:Service>
        <tp:CanSend>
          <tp:ThisPartyActionBinding
            tp:id="companyA_ABID1"
            tp:action="Purchase Order Request Action"
            tp:packageId="CompanyA_RequestPackage">
            <tp:BusinessTransactionCharacteristics
              tp:isNonRepudiationRequired="true"
              tp:isNonRepudiationReceiptRequired="true"
              tp:isConfidential="transient"
              tp:isAuthenticated="persistent"
              tp:isTamperProof="persistent"
              tp:isAuthorizationRequired="true"
              tp:timeToAcknowledgeReceipt="PT2H"
              tp:timeToPerform="P1D"/>
            <tp:ActionContext
              tp:binaryCollaboration="Request Purchase Order"
              tp:businessTransactionActivity="Request Purchase Order"
              tp:requestOrResponseAction="Purchase Order Request Action"/>
            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
          </tp:ThisPartyActionBinding>
        </tp:CanSend>
      </tp:CanSend>
      <tp:CanSend>
        <tp:ThisPartyActionBinding
          tp:id="companyA_ABID2"
          tp:action="ReceiptAcknowledgement"
          tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
          <tp:BusinessTransactionCharacteristics
```

```

        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
            tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
<!-- The next binding uses a synchronous delivery channel -->
<tp:CanSend>
    <tp:ThisPartyActionBinding
        tp:id="companyA_ABID6"
        tp:action="Purchase Order Request Action"
        tp:packageId="CompanyA_RequestPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
            tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT5M"
        tp:timeToPerform="PT5M"/>
    <tp>ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Request Action"/>
    <tp:ChannelId>syncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:CanReceive>
    <tp:ThisPartyActionBinding
        tp:id="companyA_ABID7"
        tp:action="Purchase Order Confirmation Action"
        tp:packageId="CompanyA_SyncReplyPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT5M"/>
    <tp>ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
    <tp:ChannelId>syncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
    <tp:ThisPartyActionBinding
        tp:id="companyA_ABID8"
        tp:action="Exception"
        tp:packageId="CompanyA_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>syncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
    <!-- COMMON -->
</tp:CanReceive>
    <simpleType name="statusValue.type">
</tp:CanSend>
    <restriction base="NMTOKEN">
<tp:CanReceive>
    <enumeration value="agreed"/>
    <tp:ThisPartyActionBinding
        <enumeration value="signed"/>
        tp:id="companyA_ABID3"
        <enumeration value="proposed"/>
        tp:action="Purchase Order Confirmation Action"
        </restriction>
        tp:packageId="CompanyA_ResponsePackage">
    </simpleType>
    <tp:BusinessTransactionCharacteristics
        <simpleType name="endpointType.type">
        tp:isNonRepudiationRequired="true"
        <restriction base="NMTOKEN">
        tp:isNonRepudiationReceiptRequired="true"
        <enumeration value="login"/>
        tp:isConfidential="transient"
        <enumeration value="request"/>
        tp:isAuthenticated="persistent"
        <enumeration value="response"/>
        tp:isTamperProof="persistent"
        <enumeration value="error"/>
    </tp:CanReceive>
    <tp:ThisPartyActionBinding

```



```

tp:id="companyA_ABID4"
tp:action="ReceiptAcknowledgment"
tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
<tp:BusinessTransactionCharacteristics
  tp:isNonRepudiationRequired="true"
  tp:isNonRepudiationReceiptRequired="true"
  tp:isConfidential="transient"
  tp:isAuthenticated="persistent" tp:isTamperProof="persistent"
  tp:isAuthorizationRequired="true"/>
  <tp:ChannelId>asyncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID5"
    tp:action="Exception"
    tp:packageId="CompanyA_ExceptionPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanReceive>
</tp:ServiceBinding>
</tp:CollaborationRole>
<!-- Certificates used by the "Buyer" company -->
<tp:Certificate tp:certId="CompanyA_AppCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_AppCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_SigningCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_SigningCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_EncryptionCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_EncryptionCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_ServerCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_ServerCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_ClientCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_ClientCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA1">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA1_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA2">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA2_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA3">
  <ds:KeyInfo>
    tp:isAuthorizationRequired="true"
    tp:timeToAcknowledgeReceipt="PT2H"/> </restriction>
  <tp:ActionContext </simpleType>
    tp:binaryCollaboration="Request Purchase Order" <simpleType name="non-empty-string">
    tp:businessTransactionActivity="Request Purchase Order" <restriction base="string">
    tp:requestOrResponseAction="Purchase Order Confirmation Action"/> <minLength
value="1"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId> </restriction>
  </tp:ThisPartyActionBinding> </simpleType>
  <tp:CanReceive> <restriction base="NMOKEN">
    <ds:KeyName>TrustedRootCertA3_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>

```

TCVN ISO/TS 15000-1 : 2007

```
<tp:Certificate tp:certId="TrustedRootCertA4">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA4_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA5">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA5_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:SecurityDetails tp:securityId="CompanyA_TransportSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA1"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA2"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA4"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<tp:SecurityDetails tp:securityId="CompanyA_MessageSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA3"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA5"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<!-- An asynchronous delivery channel -->
<tp:DeliveryChannel
  tp:channelId="asyncChannelA1"
  tp:transportId="transportA2"
  tp:docExchangeId="docExchangeA1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="none"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"/>
</tp:DeliveryChannel>
<!-- A synchronous delivery channel -->
<tp:DeliveryChannel
  tp:channelId="syncChannelA1"
  tp:transportId="transportA1"
  tp:docExchangeId="docExchangeA1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="signalsAndResponse"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"/>
</tp:DeliveryChannel>
<tp:Transport tp:transportId="transportA1">
  <tp:TransportSender>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:AccessAuthentication>digest</tp:AccessAuthentication>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyA_ClientCert"/>
      <tp:ServerSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportClientSecurity>
  </tp:TransportSender>
  <tp:TransportReceiver>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:AccessAuthentication>digest</tp:AccessAuthentication>
    <tp:Endpoint
      tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler/sync"
      tp:type="allPurpose"/>
    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>
      <tp:ClientSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
<tp:Transport tp:transportId="transportA2">
  <tp:TransportSender>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:AccessAuthentication>digest</tp:AccessAuthentication>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyA_ClientCert"/>
      <tp:ServerSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportClientSecurity>
```

```

</tp:TransportSender>
<tp:TransportReceiver>
  <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
  <tp:AccessAuthentication>basic</tp:AccessAuthentication>
  <tp:AccessAuthentication>digest</tp:AccessAuthentication>
  <tp:Endpoint
    tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler/sync"
    tp:type="allPurpose"/>
  <tp:TransportServerSecurity>
    <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
    <tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>
    <tp:ClientSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
  </tp:TransportServerSecurity>
</tp:TransportReceiver>
</tp:Transport>
<tp:DocExchange tp:docExchangeId="docExchangeA1">
  <tp:ebXMLSenderBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:SenderNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningCertificateRef tp:certId="CompanyA_SigningCert"/>
    </tp:SenderNonRepudiation>
    <tp:SenderDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
    </tp:SenderDigitalEnvelope>
  </tp:ebXMLSenderBinding>
  <tp:ebXMLReceiverBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:ReceiverNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
    </tp:ReceiverNonRepudiation>
    <tp:ReceiverDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionCertificateRef tp:certId="CompanyA_EncryptionCert"/>
    </tp:ReceiverDigitalEnvelope>
  </tp:ebXMLReceiverBinding>
</tp:DocExchange>
</tp:PartyInfo>
<!-- SimplePart corresponding to the SOAP Envelope -->
<tp:SimplePart
  tp:id="CompanyA_MsgHdr"
  tp:mimetype="text/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
<tp:SimplePart
  tp:id="CompanyA_ReceiptAcknowledgment"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
    tp:version="2.0">
    http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to an Exception business signal -->
<tp:SimplePart
  tp:id="CompanyA_Exception"
  tp:mimetype="application/xml">

```

TCVN ISO/TS 15000-1 : 2007

```
<tp:NamespaceSupported
  tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
  tp:version="2.0">
  http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
</tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a request action -->
<tp:SimplePart
  tp:id="CompanyA_Request"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd"
    tp:version="2.0">
    http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a response action -->
<tp:SimplePart
  tp:id="CompanyA_Response"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PurchaseOrderConfirmation.xsd"
    tp:version="2.0">
    http://www.rosettanet.org/schemas/PIP3A4PurchaseOrderConfirmation.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- An ebXML message with a SOAP Envelope only -->
<tp:Packaging tp:id="CompanyA_MshSignalPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_MshSignal"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a request action payload -->
<tp:Packaging tp:id="CompanyA_RequestPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_RequestMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyA_Request"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a response action payload -->
<tp:Packaging tp:id="CompanyA_ResponsePackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_ResponseMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyA_Response"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a Receipt Acknowledgment signal, plus a business response,
  or an ebXML message with an Exception signal -->
<tp:Packaging tp:id="CompanyA_SyncReplyPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_SignalAndResponseMsg"
      tp:mimetype="multipart/related"
```

```

        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
        <tp:Constituent tp:idref="CompanyA_ReceiptAcknowledgment"/>
        <tp:Constituent tp:idref="CompanyA_Response"/>
    </tp:Composite>
</tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a ReceiptAcknowledgment payload -->
<tp:Packaging tp:id="CompanyA_ReceiptAcknowledgmentPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyA_ReceiptAcknowledgmentMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
            <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
            <tp:Constituent tp:idref="CompanyA_ReceiptAcknowledgment"/>
        </tp:Composite>
    </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
<tp:Packaging tp:id="CompanyA_ExceptionPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyA_ExceptionMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
            <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
            <tp:Constituent tp:idref="CompanyA_Exception"/>
        </tp:Composite>
    </tp:CompositeList>
</tp:Packaging>
<tp:Comment xml:lang="en-US">Buyer's Collaboration Protocol Profile</tp:Comment>
</tp:CollaborationProtocolProfile>

```

cpp-example-companyB-2_0b.xml:

```

<?xml version="1.0"?>
<!-- Copyright UN/CEFACT and OASIS, 2001. All Rights Reserved. -->
<tp:CollaborationProtocolProfile
    xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
        cpp-cpa-2_0.xsd"
    tp:cppid="uri:companyB-cpp"
    tp:version="2_0b">
    <!-- Party info for CompanyB-->
    <tp:PartyInfo
        tp:partyName="CompanyB"
        tp:defaultMshChannelId="asyncChannelB1"
        tp:defaultMshPackageId="CompanyB_MshSignalPackage">
        <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</tp:PartyId>
        <tp:PartyRef xlink:type="simple" xlink:href="http://CompanyB.com/about.html"/>
        <tp:CollaborationRole>
            <tp:ProcessSpecification
                tp:version="2.0"
                tp:name="PIP3A4RequestPurchaseOrder"
                xlink:type="simple" xlink:href="http://www.rosettanet.org/processes/3A4.xml"
                tp:uuid="urn:icann:rosettanet.org:bpid:3A4$2.0"/>
            <tp:Role
                tp:name="Seller"
                xlink:type="simple"
                xlink:href="http://www.rosettanet.org/processes/3A4.xml#seller"/>
            <tp:ApplicationCertificateRef tp:certId="CompanyB_AppCert"/>
            <tp:ServiceBinding>
                <tp:Service>bpid:icann:rosettanet.org:3A4$2.0</tp:Service>
                <tp:CanSend>
                    <tp:ThisPartyActionBinding
                        tp:id="companyB_ABID1"
                        tp:action="Purchase Order Confirmation Action"
                        tp:packageId="CompanyB_ResponsePackage">
                    </tp:BusinessTransactionCharacteristics

```

```

        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"/>
    <tp:ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
    <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
<tp:CanSend>
    <tp:ThisPartyActionBinding
        tp:id="companyB_ABID2"
        tp:action="ReceiptAcknowledgement"
        tp:packageId="CompanyB_ReceiptAcknowledgmentPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
<tp:CanSend>
    <tp:ThisPartyActionBinding
        tp:id="companyB_ABID3"
        tp:action="Exception"
        tp:packageId="CompanyB_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
<tp:CanReceive>
    <tp:ThisPartyActionBinding
        tp:id="companyB_ABID4"
        tp:action="Purchase Order Request Action"
        tp:packageId="CompanyB_RequestPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"
        tp:timeToPerform="P1D"/>
    <tp:ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Request Action"/>
    <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
    <tp:ThisPartyActionBinding
        tp:id="companyB_ABID5" tp:action="ReceiptAcknowledgment"
        tp:packageId="CompanyB_ReceiptAcknowledgmentPackage">
    <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanReceive>
<!-- The next binding uses a synchronous delivery channel -->

```

```

<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID6"
    tp:action="Purchase Order Request Action"
    tp:packageId="CompanyB_SyncReplyPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"
      tp:timeToAcknowledgeReceipt="PT5M"
      tp:timeToPerform="PT5M"/>
    <tp:ActionContext
      tp:binaryCollaboration="Request Purchase Order"
      tp:businessTransactionActivity="Request Purchase Order"
      tp:requestOrResponseAction="Purchase Order Request Action"/>
    <tp:ChannelId>syncChannelB1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanReceive>
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID7"
    tp:action="Purchase Order Confirmation Action"
    tp:packageId="CompanyB_ResponsePackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"
      tp:timeToAcknowledgeReceipt="PT5M"/>
    <tp:ActionContext
      tp:binaryCollaboration="Request Purchase Order"
      tp:businessTransactionActivity="Request Purchase Order"
      tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
    <tp:ChannelId>syncChannelB1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanSend>
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID8"
    tp:action="Exception"
    tp:packageId="CompanyB_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>syncChannelB1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanSend>
</tp:CanSend>
</tp:CanReceive>
</tp:ServiceBinding>
</tp:CollaborationRole>
<!-- Certificates used by the "Seller" company -->
<tp:Certificate tp:certId="CompanyB_AppCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_AppCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_SigningCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_Signingcert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_EncryptionCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_EncryptionCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_ServerCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_ServerCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_ClientCert">

```

```

    <ds:KeyInfo>
      <ds:KeyName>CompanyB_ClientCert_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:Certificate tp:certId="TrustedRootCertB4">
    <ds:KeyInfo>
      <ds:KeyName>TrustedRootCertB4_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:Certificate tp:certId="TrustedRootCertB5">
    <ds:KeyInfo>
      <ds:KeyName>TrustedRootCertB5_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:Certificate tp:certId="TrustedRootCertB6">
    <ds:KeyInfo>
      <ds:KeyName>TrustedRootCertB6_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:Certificate tp:certId="TrustedRootCertB7">
    <ds:KeyInfo>
      <ds:KeyName>TrustedRootCertB7_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:Certificate tp:certId="TrustedRootCertB8">
    <ds:KeyInfo>
      <ds:KeyName>TrustedRootCertB8_Key</ds:KeyName>
    </ds:KeyInfo>
  </tp:Certificate>
  <tp:SecurityDetails tp:securityId="CompanyB_TransportSecurity">
    <tp:TrustAnchors>
      <tp:AnchorCertificateRef tp:certId="TrustedRootCertB5"/>
      <tp:AnchorCertificateRef tp:certId="TrustedRootCertB6"/>
      <tp:AnchorCertificateRef tp:certId="TrustedRootCertB4"/>
    </tp:TrustAnchors>
  </tp:SecurityDetails>
  <tp:SecurityDetails tp:securityId="CompanyB_MessageSecurity">
    <tp:TrustAnchors>
      <tp:AnchorCertificateRef tp:certId="TrustedRootCertB8"/>
      <tp:AnchorCertificateRef tp:certId="TrustedRootCertB7"/>
    </tp:TrustAnchors>
  </tp:SecurityDetails>
  <!-- An asynchronous delivery channel -->
  <tp:DeliveryChannel
    tp:channelId="asyncChannelB1"
    tp:transportId="transportB1"
    tp:docExchangeId="docExchangeB1">
    <tp:MessagingCharacteristics
      tp:syncReplyMode="none"
      tp:ackRequested="always"
      tp:ackSignatureRequested="always"
      tp:duplicateElimination="always"/>
  </tp:DeliveryChannel>
  <!-- A synchronous delivery channel -->
  <tp:DeliveryChannel
    tp:channelId="syncChannelB1"
    tp:transportId="transportB2"
    tp:docExchangeId="docExchangeB1">
    <tp:MessagingCharacteristics
      tp:syncReplyMode="signalsAndResponse"
      tp:ackRequested="always"
      tp:ackSignatureRequested="always"
      tp:duplicateElimination="always"/>
  </tp:DeliveryChannel>
  <tp:Transport tp:transportId="transportB1">
    <tp:TransportSender>
      <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
      <tp:AccessAuthentication>basic</tp:AccessAuthentication>
      <tp:TransportClientSecurity>
        <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
        <tp:ClientCertificateRef tp:certId="CompanyB_ClientCert"/>
        <tp:ServerSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
      </tp:TransportClientSecurity>
    </tp:TransportSender>
    <tp:TransportReceiver>
      <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
      <tp:AccessAuthentication>basic</tp:AccessAuthentication>
      <tp:Endpoint
        tp:uri="https://www.CompanyB.com/servlets/ebxmlhandler/sync"
        tp:type="allPurpose"/>
    </tp:TransportReceiver>
  </tp:Transport>

```



```

    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyB_ServerCert"/>
      <tp:ClientSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
<tp:Transport tp:transportId="transportB2">
  <tp:TransportSender>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyB_ClientCert"/>
      <tp:ServerSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
    </tp:TransportClientSecurity>
  </tp:TransportSender>
  <tp:TransportReceiver>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:Endpoint
      tp:uri="https://www.CompanyB.com/servlets/ebxmlhandler/async"
      tp:type="allPurpose"/>
    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyB_ServerCert"/>
      <tp:ClientSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
<tp:DocExchange tp:docExchangeId="docExchangeB1">
  <tp:ebXMLSenderBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:SenderNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningCertificateRef tp:certId="CompanyB_SigningCert"/>
    </tp:SenderNonRepudiation>
    <tp:SenderDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionSecurityDetailsRef tp:securityId="CompanyB_MessageSecurity"/>
    </tp:SenderDigitalEnvelope>
  </tp:ebXMLSenderBinding>
  <tp:ebXMLReceiverBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:ReceiverNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningSecurityDetailsRef tp:securityId="CompanyB_MessageSecurity"/>
    </tp:ReceiverNonRepudiation>
    <tp:ReceiverDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionCertificateRef tp:certId="CompanyB_EncryptionCert"/>
    </tp:ReceiverDigitalEnvelope>
  </tp:ebXMLReceiverBinding>
</tp:DocExchange>
</tp:PartyInfo>
<!-- SimplePart corresponding to the SOAP Envelope -->
<tp:SimplePart
  tp:id="CompanyB_MsgHdr"
  tp:mimetype="text/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/draft-msg-header-05.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/draft-msg-header-05.xsd
  </tp:NamespaceSupported>

```

TCVN ISO/TS 15000-1 : 2007

```
</tp:SimplePart>
<!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
<tp:SimplePart
  tp:id="CompanyB_ReceiptAcknowledgment"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
    tp:version="2.0">
    http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to an Exception business signal -->
<tp:SimplePart
  tp:id="CompanyB_Exception"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/draft-msg-header-05.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/draft-msg-header-05.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a request action -->
<tp:SimplePart
  tp:id="CompanyB_Request"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd"
    tp:version="2.0">
    http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a response action -->
<tp:SimplePart
  tp:id="CompanyB_Response"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PurchaseOrderConfirmation.xsd.xsd"
    tp:version="2.0">
    http://www.rosettanet.org/schemas/PIP3A4PurchaseOrderConfirmation.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- An ebXML message with a SOAP Envelope only -->
<tp:Packaging tp:id="CompanyB_MshSignalPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyB_MshSignal"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a request action payload -->
<tp:Packaging tp:id="CompanyB_RequestPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="RequestMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyB_Request"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a response action payload -->
<tp:Packaging tp:id="CompanyB_ResponsePackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyB_ResponseMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
```

```

        <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
        <tp:Constituent tp:idref="CompanyB_Response"/>
    </tp:Composite>
</tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a Receipt Acknowledgment payload -->
<tp:Packaging tp:id="CompanyB_ReceiptAcknowledgmentPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyB_ReceiptAcknowledgmentMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
                <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
                <tp:Constituent tp:idref="CompanyB_ReceiptAcknowledgment"/>
            </tp:Composite>
        </tp:CompositeList>
    </tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
<tp:Packaging tp:id="CompanyB_ExceptionPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyB_ExceptionMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
                <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
                <tp:Constituent tp:idref="CompanyB_Exception"/>
            </tp:Composite>
        </tp:CompositeList>
    </tp:Packaging>
<!-- An ebXML message with a Receipt Acknowledgment signal, plus a business response,
or an ebXML message with an Exception signal -->
<tp:Packaging tp:id="CompanyB_SyncReplyPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyB_SignalAndResponseMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
                <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
                <tp:Constituent tp:idref="CompanyB_ReceiptAcknowledgment"/>
                <tp:Constituent tp:idref="CompanyB_Response"/>
            </tp:Composite>
        </tp:CompositeList>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyB_SyncExceptionMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
                <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
                <tp:Constituent tp:idref="CompanyB_Exception"/>
            </tp:Composite>
        </tp:CompositeList>
    </tp:Packaging>
    <tp:Comment xml:lang="en-US">Seller's Collaboration Protocol Profile</tp:Comment>
</tp:CollaborationProtocolProfile>

```

Phụ lục B

(tham khảo)

Ví dụ về tài liệu CPA

Ví dụ trong Phụ lục này là để được phân tích cú pháp với một Giản đồ phân tích cú pháp XML. Giản đồ đó sẵn có dưới dạng một tệp ASCII tại

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0b.xsd

Ví dụ này có thể được phân tích cú pháp cùng với XSD sẵn có tại

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-example-2_0b.xml

```
<?xml version="1.0"?>
<!-- Copyright UN/CEFACT and OASIS, 2001. All Rights Reserved. -->
<tp:CollaborationProtocolAgreement
  xmlns:tp="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
    cpp-cpa-2_0.xsd"
  tp:cpaId="uri:companyA-and-companyB-cpa" tp:version="2_0b">
  <tp:Status tp:value="proposed"/>
  <tp:Start>2001-05-20T07:21:00Z</tp:Start>
  <tp:End>2002-05-20T07:21:00Z</tp:End>
  <tp:ConversationConstraints tp:invocationLimit="100" tp:concurrentConversations="10"/>
  <!-- Party info for CompanyA -->
  <tp:PartyInfo
    tp:partyName="CompanyA"
    tp:defaultMshChannelId="asyncChannelA1"
    tp:defaultMshPackageId="CompanyA_MshSignalPackage">
    <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">123456789</tp:PartyId>
    <tp:PartyRef xlink:href="http://CompanyA.com/about.html"/>
    <tp:CollaborationRole>
      <tp:ProcessSpecification
        tp:version="2.0"
        tp:name="PIP3A4RequestPurchaseOrder"
        xlink:type="simple"
        xlink:href="http://www.rosettanet.org/processes/3A4.xml"
        tp:uid="urn:icann:rosettanet.org:bpid:3A4$2.0"/>
      <tp:Role
        tp:name="Buyer"
        xlink:type="simple"
        xlink:href="http://www.rosettanet.org/processes/3A4.xml#Buyer"/>
      <tp:ApplicationCertificateRef tp:certId="CompanyA_AppCert"/>
      <tp:ServiceBinding>
        <tp:Service>bpid:icann:rosettanet.org:3A4$2.0</tp:Service>
        <tp:CanSend>
          <tp:ThisPartyActionBinding
            tp:id="companyA_ABID1"
            tp:action="Purchase Order Request Action"
            tp:packageId="CompanyA_RequestPackage">
            <tp:BusinessTransactionCharacteristics
              tp:isNonRepudiationRequired="true"
              tp:isNonRepudiationReceiptRequired="true"
              tp:isConfidential="transient"
              tp:isAuthenticated="persistent"
              tp:isTamperProof="persistent"
              tp:isAuthorizationRequired="true"
              tp:timeToAcknowledgeReceipt="PT2H"
              tp:timeToPerform="P1D"/>
            <tp>ActionContext
              tp:binaryCollaboration="Request Purchase Order"
              tp:businessTransactionActivity="Request Purchase Order"
              tp:requestOrResponseAction="Purchase Order Request Action"/>
            <tp:ChannelId>asyncChannelA1</tp:ChannelId>
          </tp:ThisPartyActionBinding>
          <tp:OtherPartyActionBinding>companyB_ABID4</tp:OtherPartyActionBinding>
        </tp:CanSend>
      </tp:CanSend>
    </tp:PartyInfo>
  </tp:CollaborationProtocolAgreement>
```

```

<tp:ThisPartyActionBinding
  tp:id="companyA_ABID2"
  tp:action="ReceiptAcknowledgement"
  tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"/>
  <tp:ChannelId>asyncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:OtherPartyActionBinding>companyB_ABID5</tp:OtherPartyActionBinding>
</tp:CanSend>
<!-- The next binding uses a synchronous delivery channel -->
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID6"
    tp:action="Purchase Order Request Action"
    tp:packageId="CompanyA_RequestPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"
      tp:timeToAcknowledgeReceipt="PT5M"
      tp:timeToPerform="PT5M"/>
    <tp:ActionContext
      tp:binaryCollaboration="Request Purchase Order"
      tp:businessTransactionActivity="Request Purchase Order"
      tp:requestOrResponseAction="Purchase Order Request Action"/>
    <tp:ChannelId>syncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
  <tp:OtherPartyActionBinding>companyB_ABID6</tp:OtherPartyActionBinding>
  <tp:CanReceive>
    <tp:ThisPartyActionBinding
      tp:id="companyA_ABID7"
      tp:action="Purchase Order Confirmation Action"
      tp:packageId="CompanyA_SyncReplyPackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT5M"/>
      <tp:ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
      <tp:ChannelId>syncChannelA1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyB_ABID7</tp:OtherPartyActionBinding>
  </tp:CanReceive>
  <tp:CanReceive>
    <tp:ThisPartyActionBinding
      tp:id="companyA_ABID8"
      tp:action="Exception"
      tp:packageId="CompanyA_ExceptionPackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
      <tp:ChannelId>syncChannelA1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyB_ABID8</tp:OtherPartyActionBinding>
  </tp:CanReceive>
</tp:CanSend>
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID3"
    tp:action="Purchase Order Confirmation Action"
    tp:packageId="CompanyA_ResponsePackage">

```

```

    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"
      tp:timeToAcknowledgeReceipt="PT2H"/>
    <tp:ActionContext
      tp:binaryCollaboration="Request Purchase Order"
      tp:businessTransactionActivity="Request Purchase Order"
      tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
  <tp:OtherPartyActionBinding>companyB_ABID1</tp:OtherPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID4"
    tp:action="ReceiptAcknowledgment"
    tp:packageId="CompanyA_ReceiptAcknowledgmentPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
  <tp:OtherPartyActionBinding>companyB_ABID2</tp:OtherPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID5"
    tp:action="Exception"
    tp:packageId="CompanyA_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"/>
    <tp:ChannelId>asyncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
  <tp:OtherPartyActionBinding>companyB_ABID3</tp:OtherPartyActionBinding>
</tp:CanReceive>
</tp:ServiceBinding>
</tp:CollaborationRole>
<!-- Certificates used by the "Buyer" company -->
<tp:Certificate tp:certId="CompanyA_AppCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_AppCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_SigningCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_SigningCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_EncryptionCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_EncryptionCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_ServerCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_ServerCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyA_ClientCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyA_ClientCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA1">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA1_Key </ds:KeyName>
  </ds:KeyInfo>

```

```

</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA2">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA2_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA3">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA3_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA4">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA4_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertA5">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertA5_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:SecurityDetails tp:securityId="CompanyA_TransportSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA1"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA2"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA4"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<tp:SecurityDetails tp:securityId="CompanyA_MessageSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA3"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertA5"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<tp:DeliveryChannel
  tp:channelId="asyncChannelA1"
  tp:transportId="transportA1"
  tp:docExchangeId="docExchangeA1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="none"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"/>
</tp:DeliveryChannel>
<tp:DeliveryChannel
  tp:channelId="syncChannelA1"
  tp:transportId="transportA2"
  tp:docExchangeId="docExchangeA1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="signalsAndResponse"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"/>
</tp:DeliveryChannel>
<tp:Transport tp:transportId="transportA1">
  <tp:TransportSender>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyA_ClientCert"/>
      <tp:ServerSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportClientSecurity>
  </tp:TransportSender>
  <tp:TransportReceiver>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:Endpoint
      tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler/async"
      tp:type="allPurpose"/>
    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>
      <tp:ClientSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
<tp:Transport tp:transportId="transportA2">
  <tp:TransportSender>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>

```

```

    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:TransportClientSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ClientCertificateRef tp:certId="CompanyA_ClientCert"/>
      <tp:ServerSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportClientSecurity>
  </tp:TransportSender>
  <tp:TransportReceiver>
    <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
    <tp:AccessAuthentication>basic</tp:AccessAuthentication>
    <tp:Endpoint
      tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler/sync"
      tp:type="allPurpose"/>
    <tp:TransportServerSecurity>
      <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
      <tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>
      <tp:ClientSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>
    </tp:TransportServerSecurity>
  </tp:TransportReceiver>
</tp:Transport>
<tp:DocExchange tp:docExchangeId="docExchangeA1">
  <tp:ebXMLSenderBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:SenderNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningCertificateRef tp:certId="CompanyA_SigningCert"/>
    </tp:SenderNonRepudiation>
    <tp:SenderDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
    </tp:SenderDigitalEnvelope>
  </tp:ebXMLSenderBinding>
  <tp:ebXMLReceiverBinding tp:version="2.0">
    <tp:ReliableMessaging>
      <tp:Retries>3</tp:Retries>
      <tp:RetryInterval>PT2H</tp:RetryInterval>
      <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
    </tp:ReliableMessaging>
    <tp:PersistDuration>P1D</tp:PersistDuration>
    <tp:ReceiverNonRepudiation>
      <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
      <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
      <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
      <tp:SigningSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
    </tp:ReceiverNonRepudiation>
    <tp:ReceiverDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionCertificateRef tp:certId="CompanyA_EncryptionCert"/>
    </tp:ReceiverDigitalEnvelope>
  </tp:ebXMLReceiverBinding>
</tp:DocExchange>
</tp:PartyInfo>
<!-- Party info for CompanyB -->
<tp:PartyInfo
  tp:partyName="CompanyB"
  tp:defaultMshChannelId="asyncChannelB1"
  tp:defaultMshPackageId="CompanyB_MshSignalPackage">
  <tp:PartyId tp:type="urn:oasis:names:tc:ebxml-cppa:partyid-type:duns">987654321</tp:PartyId>
  <tp:PartyRef xlink:type="simple" xlink:href="http://CompanyB.com/about.html"/>
  <tp:CollaborationRole>
    <tp:ProcessSpecification
      tp:version="2.0"
      tp:name="PIP3A4RequestPurchaseOrder"
      xlink:type="simple"
      xlink:href="http://www.rosettanet.org/processes/3A4.xml"
      tp:uuid="urn:icann:rosettanet.org:bpid:3A4$2.0"/>
    <tp:Role
      tp:name="Seller"
      xlink:type="simple"
      xlink:href="http://www.rosettanet.org/processes/3A4.xml#seller"/>
  <tp:ApplicationCertificateRef tp:certId="CompanyB_AppCert"/>

```



```

<tp:ServiceBinding>
  <tp:Service>bpid:icann:rosettanet.org:3A4$2.0</tp:Service>
  <tp:CanSend>
    <tp:ThisPartyActionBinding
      tp:id="companyB_ABID1"
      tp:action="Purchase Order Confirmation Action"
      tp:packageId="CompanyB_ResponsePackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"/>
      <tp>ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
      <tp:ChannelId>asyncChannelB1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyA_ABID3</tp:OtherPartyActionBinding>
  </tp:CanSend>
  <tp:CanSend>
    <tp:ThisPartyActionBinding
      tp:id="companyB_ABID2"
      tp:action="ReceiptAcknowledgement"
      tp:packageId="CompanyB_ReceiptAcknowledgmentPackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
      <tp:ChannelId>asyncChannelB1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyA_ABID4</tp:OtherPartyActionBinding>
  </tp:CanSend>
  <tp:CanSend>
    <tp:ThisPartyActionBinding
      tp:id="companyB_ABID3"
      tp:action="Exception"
      tp:packageId="CompanyB_ExceptionPackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"/>
      <tp:ChannelId>asyncChannelB1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyA_ABID5</tp:OtherPartyActionBinding>
  </tp:CanSend>
  <tp:CanReceive>
    <tp:ThisPartyActionBinding
      tp:id="companyB_ABID4"
      tp:action="Purchase Order Request Action"
      tp:packageId="CompanyB_RequestPackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT2H"
        tp:timeToPerform="P1D"/>
      <tp>ActionContext
        tp:binaryCollaboration="Request Purchase Order"
        tp:businessTransactionActivity="Request Purchase Order"
        tp:requestOrResponseAction="Purchase Order Request Action"/>
      <tp:ChannelId>asyncChannelB1</tp:ChannelId>
    </tp:ThisPartyActionBinding>
    <tp:OtherPartyActionBinding>companyA_ABID1</tp:OtherPartyActionBinding>
  </tp:CanReceive>
  <tp:CanReceive>
    <tp:ThisPartyActionBinding

```

TCVN ISO/TS 15000-1 : 2007

```
tp:id="companyB_ABID5"
tp:action="ReceiptAcknowledgment"
tp:packageId="CompanyB_ReceiptAcknowledgmentPackage">
<tp:BusinessTransactionCharacteristics
  tp:isNonRepudiationRequired="true"
  tp:isNonRepudiationReceiptRequired="true"
  tp:isConfidential="transient"
  tp:isAuthenticated="persistent"
  tp:isTamperProof="persistent"
  tp:isAuthorizationRequired="true"/>
  <tp:ChannelId>asyncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:OtherPartyActionBinding>companyA_ABID2</tp:OtherPartyActionBinding>
</tp:CanReceive>
<!-- The next binding uses a synchronous delivery channel -->
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID6"
    tp:action="Purchase Order Request Action"
    tp:packageId="CompanyB_RequestPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"
    tp:timeToAcknowledgeReceipt="PT5M"
    tp:timeToPerform="PT5M"/>
  <tp:ActionContext
    tp:binaryCollaboration="Request Purchase Order"
    tp:businessTransactionActivity="Request Purchase Order"
    tp:requestOrResponseAction="Purchase Order Request Action"/>
  <tp:ChannelId>syncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:OtherPartyActionBinding>companyA_ABID6</tp:OtherPartyActionBinding>
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID7"
    tp:action="Purchase Order Confirmation Action"
    tp:packageId="CompanyB_SyncReplyPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"
    tp:timeToAcknowledgeReceipt="PT5M"/>
  <tp:ActionContext
    tp:binaryCollaboration="Request Purchase Order"
    tp:businessTransactionActivity="Request Purchase Order"
    tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
  <tp:ChannelId>syncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:OtherPartyActionBinding>companyA_ABID7</tp:OtherPartyActionBinding>
</tp:CanSend>
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID8"
    tp:action="Exception"
    tp:packageId="CompanyB_ExceptionPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"/>
  <tp:ChannelId>syncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
<tp:OtherPartyActionBinding>companyA_ABID8</tp:OtherPartyActionBinding>
</tp:CanSend>
</tp:CanReceive>
</tp:ServiceBinding>
</tp:CollaborationRole>
<!-- Certificates used by the "Seller" company -->
<tp:Certificate tp:certId="CompanyB_AppCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_AppCert_Key</ds:KeyName>
```

```

    </ds:KeyInfo>
  </tp:Certificate>
<tp:Certificate tp:certId="CompanyB_SigningCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_Signingcert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_EncryptionCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_EncryptionCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_ServerCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_ServerCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="CompanyB_ClientCert">
  <ds:KeyInfo>
    <ds:KeyName>CompanyB_ClientCert_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertB4">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertB4_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertB5">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertB5_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertB6">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertB6_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertB7">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertB7_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:Certificate tp:certId="TrustedRootCertB8">
  <ds:KeyInfo>
    <ds:KeyName>TrustedRootCertB8_Key</ds:KeyName>
  </ds:KeyInfo>
</tp:Certificate>
<tp:SecurityDetails tp:securityId="CompanyB_TransportSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertB5"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertB6"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertB4"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<tp:SecurityDetails tp:securityId="CompanyB_MessageSecurity">
  <tp:TrustAnchors>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertB8"/>
    <tp:AnchorCertificateRef tp:certId="TrustedRootCertB7"/>
  </tp:TrustAnchors>
</tp:SecurityDetails>
<!-- An asynchronous delivery channel -->
<tp:DeliveryChannel
  tp:channelId="asyncChannelB1"
  tp:transportId="transportB1"
  tp:docExchangeId="docExchangeB1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="none"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"
    tp:duplicateElimination="always"/>
</tp:DeliveryChannel>
<!-- A synchronous delivery channel -->
<tp:DeliveryChannel
  tp:channelId="syncChannelB1"
  tp:transportId="transportB2"
  tp:docExchangeId="docExchangeB1">
  <tp:MessagingCharacteristics
    tp:syncReplyMode="signalsAndResponse"
    tp:ackRequested="always"
    tp:ackSignatureRequested="always"

```

```

        tp:duplicateElimination="always"/>
    </tp:DeliveryChannel>
    <tp:Transport tp:transportId="transportB1">
        <tp:TransportSender>
            <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
            <tp:AccessAuthentication>basic</tp:AccessAuthentication>
            <tp:TransportClientSecurity>
                <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
                <tp:ClientCertificateRef tp:certId="CompanyB_ClientCert"/>
                <tp:ServerSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
            </tp:TransportClientSecurity>
        </tp:TransportSender>
        <tp:TransportReceiver>
            <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
            <tp:AccessAuthentication>basic</tp:AccessAuthentication>
            <tp:Endpoint
                tp:uri="https://www.CompanyB.com/servlets/ebxmlhandler/async"
                tp:type="allPurpose"/>
            <tp:TransportServerSecurity>
                <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
                <tp:ServerCertificateRef tp:certId="CompanyB_ServerCert"/>
                <tp:ClientSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
            </tp:TransportServerSecurity>
        </tp:TransportReceiver>
    </tp:Transport>
    <tp:Transport tp:transportId="transportB2">
        <tp:TransportSender>
            <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
            <tp:AccessAuthentication>basic</tp:AccessAuthentication>
            <tp:TransportClientSecurity>
                <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
                <tp:ClientCertificateRef tp:certId="CompanyB_ClientCert"/>
                <tp:ServerSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
            </tp:TransportClientSecurity>
        </tp:TransportSender>
        <tp:TransportReceiver>
            <tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>
            <tp:AccessAuthentication>basic</tp:AccessAuthentication>
            <tp:Endpoint
                tp:uri="https://www.CompanyB.com/servlets/ebxmlhandler/sync"
                tp:type="allPurpose"/>
            <tp:TransportServerSecurity>
                <tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>
                <tp:ServerCertificateRef tp:certId="CompanyB_ServerCert"/>
                <tp:ClientSecurityDetailsRef tp:securityId="CompanyB_TransportSecurity"/>
            </tp:TransportServerSecurity>
        </tp:TransportReceiver>
    </tp:Transport>
    <tp:DocExchange tp:docExchangeId="docExchangeB1">
        <tp:ebXMLSenderBinding tp:version="2.0">
            <tp:ReliableMessaging>
                <tp:Retries>3</tp:Retries>
                <tp:RetryInterval>PT2H</tp:RetryInterval>
                <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
            </tp:ReliableMessaging>
            <tp:PersistDuration>P1D</tp:PersistDuration>
            <tp:SenderNonRepudiation>
                <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
                <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
                <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
                <tp:SigningCertificateRef tp:certId="CompanyB_SigningCert"/>
            </tp:SenderNonRepudiation>
            <tp:SenderDigitalEnvelope>
                <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
                <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
                <tp:EncryptionSecurityDetailsRef tp:securityId="CompanyB_MessageSecurity"/>
            </tp:SenderDigitalEnvelope>
        </tp:ebXMLSenderBinding>
        <tp:ebXMLReceiverBinding tp:version="2.0">
            <tp:ReliableMessaging>
                <tp:Retries>3</tp:Retries>
                <tp:RetryInterval>PT2H</tp:RetryInterval>
                <tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
            </tp:ReliableMessaging>
            <tp:PersistDuration>P1D</tp:PersistDuration>
            <tp:ReceiverNonRepudiation>
                <tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#</tp:NonRepudiationProtocol>
                <tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1</tp:HashFunction>
                <tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1</tp:SignatureAlgorithm>
                <tp:SigningSecurityDetailsRef tp:securityId="CompanyB_MessageSecurity"/>
            </tp:ReceiverNonRepudiation>
        </tp:ebXMLReceiverBinding>
    </tp:DocExchange>

```

```

    </tp:ReceiverNonRepudiation>
    <tp:ReceiverDigitalEnvelope>
      <tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
      <tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
      <tp:EncryptionCertificateRef tp:certId="CompanyB_EncryptionCert"/>
    </tp:ReceiverDigitalEnvelope>
  </tp:ebXMLReceiverBinding>
</tp:DocExchange>
</tp:PartyInfo>
<!-- SimplePart corresponding to the SOAP Envelope -->
<tp:SimplePart
  tp:id="CompanyA_MsgHdr"
  tp:mimetype="text/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<tp:SimplePart
  tp:id="CompanyB_MsgHdr"
  tp:mimetype="text/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a Receipt Acknowledgment business signal -->
<tp:SimplePart
  tp:id="CompanyA_ReceiptAcknowledgment"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
    tp:version="2.0">http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<tp:SimplePart
  tp:id="CompanyB_ReceiptAcknowledgment"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd"
    tp:version="2.0">
    http://www.ebxml.org/bpss/ReceiptAcknowledgment.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to an Exception business signal -->
<tp:SimplePart
  tp:id="CompanyA_Exception"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<tp:SimplePart
  tp:id="CompanyB_Exception"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
    tp:version="2.0">
    http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<!-- SimplePart corresponding to a request action -->
<tp:SimplePart
  tp:id="CompanyA_Request"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd"
    tp:version="1.0">
    http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd
  </tp:NamespaceSupported>
</tp:SimplePart>
<tp:SimplePart
  tp:id="CompanyB_Request"
  tp:mimetype="application/xml">
  <tp:NamespaceSupported
    tp:location="http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd"

```

TCVN ISO/TS 15000-1 : 2007

```
    tp:version="1.0">
      http://www.rosettanet.org/schemas/PIP3A4RequestPurchaseOrder.xsd
    </tp:NamespaceSupported>
  </tp:SimplePart>
  <!-- SimplePart corresponding to a response action -->
  <tp:SimplePart
    tp:id="CompanyA_Response"
    tp:mimetype="application/xml">
    <tp:NamespaceSupported
      tp:location="http://www.rosettanet.org/schemas/PurchaseOrderConfirmation.xsd"
      tp:version="1.0">
      http://www.rosettanet.org/schemas/PIP3A4PurchaseOrderConfirmation.xsd
    </tp:NamespaceSupported>
  </tp:SimplePart>
  <tp:SimplePart
    tp:id="CompanyB_Response"
    tp:mimetype="application/xml">
    <tp:NamespaceSupported
      tp:location="http://www.rosettanet.org/schemas/PurchaseOrderConfirmation.xsd"
      tp:version="1.0">
      http://www.rosettanet.org/schemas/PIP3A4PurchaseOrderConfirmation.xsd
    </tp:NamespaceSupported>
  </tp:SimplePart>
  <!-- An ebXML message with a SOAP Envelope only -->
  <tp:Packaging
    tp:id="CompanyA_MshSignalPackage">
    <tp:ProcessingCapabilities
      tp:parse="true"
      tp:generate="true"/>
    <tp:CompositeList>
      <tp:Composite
        tp:id="CompanyA_MshSignal"
        tp:mimetype="multipart/related"
        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      </tp:Composite>
    </tp:CompositeList>
  </tp:Packaging>
  <tp:Packaging
    tp:id="CompanyB_MshSignalPackage">
    <tp:ProcessingCapabilities
      tp:parse="true"
      tp:generate="true"/>
    <tp:CompositeList>
      <tp:Composite
        tp:id="CompanyB_MshSignal"
        tp:mimetype="multipart/related"
        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
      </tp:Composite>
    </tp:CompositeList>
  </tp:Packaging>
  <!-- An ebXML message with a SOAP Envelope plus a request action payload -->
  <tp:Packaging tp:id="CompanyA_RequestPackage">
    <tp:ProcessingCapabilities
      tp:parse="true"
      tp:generate="true"/>
    <tp:CompositeList>
      <tp:Composite
        tp:id="CompanyA_RequestMsg"
        tp:mimetype="multipart/related"
        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
        <tp:Constituent tp:idref="CompanyA_Request"/>
      </tp:Composite>
    </tp:CompositeList>
  </tp:Packaging>
  <tp:Packaging tp:id="CompanyB_RequestPackage">
    <tp:ProcessingCapabilities
      tp:parse="true"
      tp:generate="true"/>
    <tp:CompositeList>
      <tp:Composite
        tp:id="CompanyB_RequestMsg"
        tp:mimetype="multipart/related"
        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
        <tp:Constituent tp:idref="CompanyB_Request"/>
      </tp:Composite>
    </tp:CompositeList>
  </tp:Packaging>
```

```

</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a response action payload -->
<tp:Packaging tp:id="CompanyA_ResponsePackage">
  <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_ResponseMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyA_Response"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<tp:Packaging tp:id="CompanyB_ResponsePackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyB_ResponseMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyB_Response"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus a Receipt Acknowledgment payload -->
<tp:Packaging tp:id="CompanyA_ReceiptAcknowledgmentPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_ReceiptAcknowledgmentMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyA_ReceiptAcknowledgment"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<tp:Packaging tp:id="CompanyB_ReceiptAcknowledgmentPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyB_ReceiptAcknowledgmentMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyB_ReceiptAcknowledgment"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a SOAP Envelope plus an Exception payload -->
<tp:Packaging tp:id="CompanyA_ExceptionPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyA_ExceptionMsg"
      tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml">
      <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
      <tp:Constituent tp:idref="CompanyA_Exception"/>
    </tp:Composite>
  </tp:CompositeList>
</tp:Packaging>
<tp:Packaging tp:id="CompanyB_ExceptionPackage">
  <tp:ProcessingCapabilities
    tp:parse="true"
    tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite
      tp:id="CompanyB_ExceptionMsg"
      tp:mimetype="multipart/related"

```

TCVN ISO/TS 15000-1 : 2007

```
        tp:mimeparameters="type=text/xml">
        <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
        <tp:Constituent tp:idref="CompanyB_Exception"/>
    </tp:Composite>
</tp:CompositeList>
</tp:Packaging>
<!-- An ebXML message with a Receipt Acknowledgment signal, plus a business response,
      or an ebXML message with an Exception signal -->
<tp:Packaging tp:id="CompanyA_SyncReplyPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyA_SignalAndResponseMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
            <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
            <tp:Constituent tp:idref="CompanyA_ReceiptAcknowledgment"/>
            <tp:Constituent tp:idref="CompanyA_Response"/>
        </tp:Composite>
    </tp:CompositeList>
</tp:Packaging>
<tp:Packaging tp:id="CompanyB_SyncReplyPackage">
    <tp:ProcessingCapabilities
        tp:parse="true"
        tp:generate="true"/>
    <tp:CompositeList>
        <tp:Composite
            tp:id="CompanyB_SignalAndResponseMsg"
            tp:mimetype="multipart/related"
            tp:mimeparameters="type=text/xml">
            <tp:Constituent tp:idref="CompanyB_MsgHdr"/>
            <tp:Constituent tp:idref="CompanyB_ReceiptAcknowledgment"/>
            <tp:Constituent tp:idref="CompanyB_Response"/>
        </tp:Composite>
    </tp:CompositeList>
</tp:Packaging>
    <tp:Comment xml:lang="en-US">buy/sell agreement between CompanyA.com and CompanyB.com</tp:Comment>
</tp:CollaborationProtocolAgreement>
```


Phụ lục C

(tham khảo)

Quy định *quá trình kinh doanh* tương ứng định nghĩa CPA và CPP đầy đủ

Quy định *quá trình kinh doanh* này được tham chiếu bởi CPP và CPA trong Phụ lục A và Phụ lục B được tái tạo ở đây. Tài liệu này sẵn có dưới dạng một tệp ASCII tại:

http://www.oasis-open.org/committees/ebxml-cppa/schema/bpss-example-2_0a.xml

Giản đồ mà tài liệu ví dụ phù hợp sẵn có dưới dạng một tệp ASCII tại: <http://www.oasis-open.org/committees/ebxml-cppa/schema/ebBPSS1.04.xsd>

```
<?xml version="1.0" encoding="UTF-8"?>
<ProcessSpecification
  xmlns="http://www.ebxml.org/BusinessProcess"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ebxml.org/BusinessProcess ebBPSS1.04.xsd"
  name="PIP3A4RequestPurchaseOrder"
  uuid="urn:icann:rosettanet.org:bpid:3A4$2.0"
  version="R02.00">
  <Documentation>
    This PIP enables a buyer to issue a purchase order and obtain a quick response
    from the provider that acknowledges which of the purchase order product line
    items are accepted, rejected, or pending.
  </Documentation>
  <!--Purchase order Request Document-->
  <BusinessDocument
    name="Purchase Order Request"
    nameID="Pip3A4PurchaseOrderRequest"
    specificationLocation="PurchaseOrderRequest.xsd">
    <Documentation>
      The document is an XSD file that specifies the rules for creating the XML
      document for the business action of requesting a purchase order.
    </Documentation>
  </BusinessDocument>
  <BusinessDocument
    name="Purchase Order Confirmation"
    nameID="Pip3A4PurchaseOrderConfirmation"
    specificationLocation="PurchaseOrderConfirmation.xsd">
    <Documentation>
      The document is an XSD file that specifies the rules for creating the XML
      document for the business action of making a purchase order confirmation.
    </Documentation>
  </BusinessDocument>
  <BusinessTransaction
    name="Request Purchase Order"
    nameID="RequestPurchaseOrder_BT">
    <RequestingBusinessActivity
      name="Purchase Order Request Action"
      nameID="PurchaseOrderRequestAction"
      isAuthorizationRequired="true"
      isIntelligibleCheckRequired="true"
      isNonRepudiationReceiptRequired="true"
      isNonRepudiationRequired="true"
      timeToAcknowledgeReceipt="P0Y0M0DT2H0M0S">
      <DocumentEnvelope
        businessDocument="Purchase Order Request"
        businessDocumentIDRef="Pip3A4PurchaseOrderRequest"
        isAuthenticated="persistent"
        isConfidential="transient"
        isTamperProof="persistent"/>
    </RequestingBusinessActivity>
    <RespondingBusinessActivity
      name="Purchase Order Confirmation Action"
      nameID="PurchaseOrderConfirmationAction"
      isAuthorizationRequired="true"
      isIntelligibleCheckRequired="true"
      isNonRepudiationReceiptRequired="true"
      isNonRepudiationRequired="true"
      timeToAcknowledgeReceipt="P0Y0M0DT2H0M0S">
```

TCVN ISO/TS 15000-1 : 2007

```
<DocumentEnvelope
  businessDocument="Purchase Order Confirmation"
  businessDocumentIDRef="Pip3A4PurchaseOrderConfirmation"
  isAuthenticated="persistent"
  isConfidential="transient"
  isPositiveResponse="true"
  isTamperProof="persistent"/>
</RespondingBusinessActivity>
</BusinessTransaction>
<BinaryCollaboration
  name="Request Purchase Order"
  nameID="RequestPurchaseOrder_BC"
  initiatingRole="BuyerId">
  <Role
    name="Buyer"
    nameID="BuyerId"/>
  <Role
    name="Seller"
    nameID="SellerId"/>
  <Start toBusinessState="Request Purchase Order"/>
  <BusinessTransactionActivity
    name="Request Purchase Order"
    nameID="RequestPurchaseOrder_BTA"
    businessTransaction="Request Purchase Order"
    businessTransactionIDRef="RequestPurchaseOrder_BT"
    fromRole="Buyer" fromRoleIDRef="BuyerId"
    toRole="Seller" toRoleIDRef="SellerId"
    isLegallyBinding="true"
    timeToPerform="P0Y0M0DT24H0M0S"
    isConcurrent="false"/>
  <Success
    fromBusinessState="Request Purchase Order"
    conditionGuard="Success"/>
  <Failure
    fromBusinessState="Request Purchase Order"
    conditionGuard="BusinessFailure"/>
  <Transition
    fromBusinessState="Request Purchase Order"
    toBusinessState="Request Purchase Order"/>
</BinaryCollaboration
> </ProcessSpecification>
```

Phụ lục D

(quy chuẩn)

W3C Tài liệu về giản đồ XML tương ứng định nghĩa CPA và CPP đầy đủ

Tài liệu về giản đồ XML này sẵn có dưới dạng một tệp ASCII tại:

http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0b.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- This is the schema that corresponds to the version 2.0 CPP/A spec -->
<!-- Some parsers may require explicit declaration of 'xmlns:xml="http://www.w3.org/XML/1998/namespace"'.
      In that case, a copy of this schema augmented with the above declaration should be cached and used
      for the purpose of schema validation for CPPs and CPAs. -->
<schema
  targetNamespace="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
  xmlns:tns="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  elementFormDefault="qualified"
  attributeFormDefault="qualified" version="2_0b">
  <import
    namespace="http://www.w3.org/1999/xlink"
    schemaLocation="http://www.oasis-open.org/committees/ebxml-msg/schema/xlink.xsd"/>
  <import
    namespace="http://www.w3.org/2000/09/xmldsig#"
    schemaLocation="http://www.w3.org/TR/xmldsig-core/xmldsig-core-schema.xsd"/>
  <import
    namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/03/xml.xsd"/>
  <attributeGroup name="pkg.grp">
    <attribute ref="tns:id" use="required"/>
    <attribute name="mimetype" type="tns:non-empty-string" use="required"/>
    <attribute name="mimeparameters" type="tns:non-empty-string"/>
  </attributeGroup>
  <attributeGroup name="xlink.grp">
    <attribute ref="xlink:type" fixed="simple"/>
    <attribute ref="xlink:href" use="required"/>
  </attributeGroup>
  <element name="CollaborationProtocolAgreement">
    <complexType>
      <sequence>
        <element ref="tns:Status"/>
        <element ref="tns:Start"/>
        <element ref="tns:End"/>
        <element ref="tns:ConversationConstraints" minOccurs="0"/>
        <element ref="tns:PartyInfo" minOccurs="2" maxOccurs="2"/>
        <element ref="tns:SimplePart" maxOccurs="unbounded"/>
        <element ref="tns:Packaging" maxOccurs="unbounded"/>
        <element ref="tns:Signature" minOccurs="0"/>
        <element ref="tns:Comment" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="cpaid" type="tns:non-empty-string" use="required"/>
      <attribute ref="tns:version" use="required"/>
    </complexType>
  </element>
  <element name="Signature">
    <complexType>
      <sequence>
        <element ref="ds:Signature" maxOccurs="3"/>
      </sequence>
    </complexType>
  </element>
  <element name="CollaborationProtocolProfile">
    <complexType>
      <sequence>
        <element ref="tns:PartyInfo" maxOccurs="unbounded"/>
        <element ref="tns:SimplePart" maxOccurs="unbounded"/>
        <element ref="tns:Packaging" maxOccurs="unbounded"/>
        <element ref="tns:Signature" minOccurs="0"/>
        <element ref="tns:Comment" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="cppid" type="tns:non-empty-string" use="required"/>
      <attribute ref="tns:version" use="required"/>
    </complexType>
  </element>

```

TCVN ISO/TS 15000-1 : 2007

```
</complexType>
</element>
<element name="ProcessSpecification">
  <complexType>
    <sequence>
      <element ref="ds:Reference" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="name" type="tns:non-empty-string" use="required"/>
    <attribute ref="tns:version" use="required"/>
    <attributeGroup ref="tns:xlink.grp"/>
    <attribute name="uuid" type="anyURI"/>
  </complexType>
</element>
<element name="Service" type="tns:service.type"/>
<element name="Protocol" type="tns:protocol.type"/>
<element name="SendingProtocol" type="tns:protocol.type"/>
<element name="ReceivingProtocol" type="tns:protocol.type"/>
<element name="OverrideMshActionBinding">
  <complexType>
    <attribute name="action" type="tns:non-empty-string" use="required"/>
    <attribute name="channelId" type="IDREF" use="required"/>
  </complexType>
</element>
<element name="ChannelId" type="IDREF"/>
<complexType name="ActionBinding.type">
  <sequence>
    <element ref="tns:BusinessTransactionCharacteristics"/>
    <element ref="tns:ActionContext" minOccurs="0"/>
    <element ref="tns:ChannelId" maxOccurs="unbounded"/>
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="id" type="ID" use="required"/>
  <attribute name="action" type="tns:non-empty-string" use="required"/>
  <attribute name="packageId" type="IDREF" use="required"/>
  <attribute ref="xlink:href" use="optional"/>
  <attribute ref="xlink:type" fixed="simple"/>
</complexType>
<element name="ActionContext">
  <complexType>
    <sequence>
      <element ref="tns:CollaborationActivity" minOccurs="0"/>
      <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="binaryCollaboration" type="tns:non-empty-string" use="required"/>
    <attribute name="businessTransactionActivity" type="tns:non-empty-string" use="required"/>
    <attribute name="requestOrResponseAction" type="tns:non-empty-string" use="required"/>
  </complexType>
</element>
<element name="CollaborationActivity">
  <complexType>
    <sequence>
      <element ref="tns:CollaborationActivity" minOccurs="0"/>
    </sequence>
    <attribute name="name" type="tns:non-empty-string"/>
  </complexType>
</element>
<element name="CollaborationRole">
  <complexType>
    <sequence>
      <element ref="tns:ProcessSpecification"/>
      <element ref="tns:Role"/>
      <element name="ApplicationCertificateRef" type="tns:CertificateRef.type" minOccurs="0"
maxOccurs="unbounded"/>
      <element name="ApplicationSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
      <element ref="tns:ServiceBinding"/>
    </sequence>
  </complexType>
</element>
<element name="PartyInfo">
  <complexType>
    <sequence>
      <element ref="tns:PartyId" maxOccurs="unbounded"/>
      <element ref="tns:PartyRef" maxOccurs="unbounded"/>
      <element ref="tns:CollaborationRole" maxOccurs="unbounded"/>
      <element ref="tns:Certificate" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="tns:SecurityDetails" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="tns:DeliveryChannel" maxOccurs="unbounded"/>
      <element ref="tns:Transport" maxOccurs="unbounded"/>
      <element ref="tns:DocExchange" maxOccurs="unbounded"/>
      <element ref="tns:OverrideMshActionBinding" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

```

    </sequence>
    <attribute name="partyName" type="tns:non-empty-string" use="required"/>
    <attribute name="defaultMshChannelId" type="IDREF" use="required"/>
    <attribute name="defaultMshPackageId" type="IDREF" use="required"/>
  </complexType>
</element>
<element name="PartyId">
  <complexType>
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute name="type" type="anyURI"/>
      </extension>
    </simpleContent>
  </complexType>
</element>
<element name="PartyRef">
  <complexType>
    <sequence>
      </sequence>
      <attributeGroup ref="tns:xlink.grp"/>
      <attribute name="type" type="anyURI"/>
      <attribute name="schemaLocation" type="anyURI"/>
    </complexType>
</element>
<element name="DeliveryChannel">
  <complexType>
    <sequence>
      <element ref="tns:MessagingCharacteristics"/>
    </sequence>
    <attribute name="channelId" type="ID" use="required"/>
    <attribute name="transportId" type="IDREF" use="required"/>
    <attribute name="docExchangeId" type="IDREF" use="required"/>
  </complexType>
</element>
<element name="Transport">
  <complexType>
    <sequence>
      <element ref="tns:TransportSender" minOccurs="0"/>
      <element ref="tns:TransportReceiver" minOccurs="0"/>
    </sequence>
    <attribute name="transportId" type="ID" use="required"/>
  </complexType>
</element>
<element name="AccessAuthentication" type="tns:accessAuthentication.type"/>
<element name="TransportSender">
  <complexType>
    <sequence>
      <element name="TransportProtocol" type="tns:protocol.type"/>
      <element ref="tns:AccessAuthentication" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="tns:TransportClientSecurity" minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="TransportReceiver">
  <complexType>
    <sequence>
      <element name="TransportProtocol" type="tns:protocol.type"/>
      <element ref="tns:AccessAuthentication" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="tns:Endpoint" maxOccurs="unbounded"/>
      <element ref="tns:TransportServerSecurity" minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="Endpoint">
  <complexType>
    <attribute name="uri" type="anyURI" use="required"/>
    <attribute name="type" type="tns:endpointType.type" default="allPurpose"/>
  </complexType>
</element>
<element name="TransportClientSecurity">
  <complexType>
    <sequence>
      <element name="TransportSecurityProtocol" type="tns:protocol.type"/>
      <element name="ClientCertificateRef" type="tns:CertificateRef.type" minOccurs="0"/>
      <element name="ServerSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
      <element ref="tns:EncryptionAlgorithm" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="TransportServerSecurity">

```

```

<complexType>
  <sequence>
    <element name="TransportSecurityProtocol" type="tns:protocol.type"/>
    <element name="ServerCertificateRef" type="tns:CertificateRef.type"/>
    <element name="ClientSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
    <element ref="tns:EncryptionAlgorithm" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>
</element>
<element name="Certificate">
  <complexType>
    <sequence>
      <element ref="ds:KeyInfo"/>
    </sequence>
    <attribute name="certId" type="ID" use="required"/>
  </complexType>
</element>
<element name="DocExchange">
  <complexType>
    <sequence>
      <element ref="tns:ebXMLSenderBinding" minOccurs="0"/>
      <element ref="tns:ebXMLReceiverBinding" minOccurs="0"/>
    </sequence>
    <attribute name="docExchangeId" type="ID" use="required"/>
  </complexType>
</element>
<element name="ReliableMessaging">
  <complexType>
    <sequence>
      <element name="Retries" type="integer" minOccurs="0"/>
      <element name="RetryInterval" type="duration" minOccurs="0"/>
      <element name="MessageOrderSemantics" type="tns:messageOrderSemantics.type"/>
    </sequence>
  </complexType>
</element>
<element name="PersistDuration" type="duration"/>
<element name="SenderNonRepudiation">
  <complexType>
    <sequence>
      <element name="NonRepudiationProtocol" type="tns:protocol.type"/>
      <element ref="tns:HashFunction"/>
      <element ref="tns:SignatureAlgorithm" maxOccurs="unbounded"/>
      <element name="SigningCertificateRef" type="tns:CertificateRef.type"/>
    </sequence>
  </complexType>
</element>
<element name="ReceiverNonRepudiation">
  <complexType>
    <sequence>
      <element name="NonRepudiationProtocol" type="tns:protocol.type"/>
      <element ref="tns:HashFunction"/>
      <element ref="tns:SignatureAlgorithm" maxOccurs="unbounded"/>
      <element name="SigningSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="HashFunction" type="tns:non-empty-string"/>
<element name="EncryptionAlgorithm">
  <complexType>
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute name="minimumStrength" type="integer"/>
        <attribute name="oid" type="tns:non-empty-string"/>
        <attribute name="w3c" type="tns:non-empty-string"/>
        <attribute name="enumerationType" type="tns:non-empty-string"/>
      </extension>
    </simpleContent>
  </complexType>
</element>
<element name="SignatureAlgorithm">
  <complexType>
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute name="oid" type="tns:non-empty-string"/>
        <attribute name="w3c" type="tns:non-empty-string"/>
        <attribute name="enumerationType" type="tns:non-empty-string"/>
      </extension>
    </simpleContent>
  </complexType>
</element>

```

```

<element name="SenderDigitalEnvelope">
  <complexType>
    <sequence>
      <element name="DigitalEnvelopeProtocol" type="tns:protocol.type"/>
      <element ref="tns:EncryptionAlgorithm" maxOccurs="unbounded"/>
      <element name="EncryptionSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="ReceiverDigitalEnvelope">
  <complexType>
    <sequence>
      <element name="DigitalEnvelopeProtocol" type="tns:protocol.type"/>
      <element ref="tns:EncryptionAlgorithm" maxOccurs="unbounded"/>
      <element name="EncryptionCertificateRef" type="tns:CertificateRef.type"/>
    </sequence>
  </complexType>
</element>
<element name="ebXMLSenderBinding">
  <complexType>
    <sequence>
      <element ref="tns:ReliableMessaging" minOccurs="0"/>
      <element ref="tns:PersistDuration" minOccurs="0"/>
      <element ref="tns:SenderNonRepudiation" minOccurs="0"/>
      <element ref="tns:SenderDigitalEnvelope" minOccurs="0"/>
      <element ref="tns:NamespaceSupported" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute ref="tns:version" use="required"/>
  </complexType>
</element>
<element name="ebXMLReceiverBinding">
  <complexType>
    <sequence>
      <element ref="tns:ReliableMessaging" minOccurs="0"/>
      <element ref="tns:PersistDuration" minOccurs="0"/>
      <element ref="tns:ReceiverNonRepudiation" minOccurs="0"/>
      <element ref="tns:ReceiverDigitalEnvelope" minOccurs="0"/>
      <element ref="tns:NamespaceSupported" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute ref="tns:version" use="required"/>
  </complexType>
</element>
<element name="NamespaceSupported">
  <complexType>
    <simpleContent>
      <extension base="anyURI">
        <attribute name="location" type="anyURI" use="required"/>
        <attribute ref="tns:version"/>
      </extension>
    </simpleContent>
  </complexType>
</element>
<element name="BusinessTransactionCharacteristics">
  <complexType>
    <attribute name="isNonRepudiationRequired" type="boolean"/>
    <attribute name="isNonRepudiationReceiptRequired" type="boolean"/>
    <attribute name="isConfidential" type="tns:persistenceLevel.type"/>
    <attribute name="isAuthenticated" type="tns:persistenceLevel.type"/>
    <attribute name="isTamperProof" type="tns:persistenceLevel.type"/>
    <attribute name="isAuthorizationRequired" type="boolean"/>
    <attribute name="isIntelligibleCheckRequired" type="boolean"/>
    <attribute name="timeToAcknowledgeReceipt" type="duration"/>
    <attribute name="timeToAcknowledgeAcceptance" type="duration"/>
    <attribute name="timeToPerform" type="duration"/>
    <attribute name="retryCount" type="integer"/>
  </complexType>
</element>
<element name="MessagingCharacteristics">
  <complexType>
    <attribute ref="tns:syncReplyMode" default="none"/>
    <attribute name="ackRequested" type="tns:perMessageCharacteristics.type" default="perMessage"/>
    <attribute name="ackSignatureRequested" type="tns:perMessageCharacteristics.type" default="perMessage"/>
    <attribute name="duplicateElimination" type="tns:perMessageCharacteristics.type" default="perMessage"/>
    <attribute name="actor" type="tns:actor.type"/>
  </complexType>
</element>
<element name="ServiceBinding">
  <complexType>

```

```

    <sequence>
      <element ref="tns:Service"/>
      <element ref="tns:CanSend" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="tns:CanReceive" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="CanSend">
  <complexType>
    <sequence>
      <element name="ThisPartyActionBinding" type="tns:ActionBinding.type"/>
      <element name="OtherPartyActionBinding" type="IDREF" minOccurs="0"/>
      <element ref="tns:CanReceive" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="CanReceive">
  <complexType>
    <sequence>
      <element name="ThisPartyActionBinding" type="tns:ActionBinding.type"/>
      <element name="OtherPartyActionBinding" type="IDREF" minOccurs="0"/>
      <element ref="tns:CanSend" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="Status">
  <complexType>
    <attribute name="value" type="tns:statusValue.type" use="required"/>
  </complexType>
</element>
<element name="Start" type="dateTime"/>
<element name="End" type="dateTime"/>
<element name="Type" type="tns:non-empty-string"/>
<element name="ConversationConstraints">
  <complexType>
    <attribute name="invocationLimit" type="int"/>
    <attribute name="concurrentConversations" type="int"/>
  </complexType>
</element>
<element name="Role">
  <complexType>
    <attribute name="name" type="tns:non-empty-string" use="required"/>
    <attributeGroup ref="tns:xlink.grp"/>
  </complexType>
</element>
<element name="SignatureTransforms">
  <complexType>
    <sequence>
      <element ref="ds:Transform" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="EncryptionTransforms">
  <complexType>
    <sequence>
      <element ref="ds:Transform" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="Constituent">
  <complexType>
    <enumeration value="allPurpose"/>
    <sequence>
      <element ref="tns:SignatureTransforms" minOccurs="0"/>
      <element ref="tns:EncryptionTransforms" minOccurs="0"/>
    </sequence>
    <attribute ref="tns:idref" use="required"/>
    <attribute name="excludedFromSignature" type="boolean" default="false"/>
    <attribute name="minOccurs" type="nonNegativeInteger"/>
    <attribute name="maxOccurs" type="nonNegativeInteger"/>
  </complexType>
</element>
<element name="Packaging">
  <complexType>
    <sequence>
      <element name="ProcessingCapabilities">
        <complexType>
          <attribute name="parse" type="boolean" use="required"/>
          <attribute name="generate" type="boolean" use="required"/>
        </complexType>
      </element>
    </sequence>
  </complexType>

```



```

</element>
<element name="CompositeList" maxOccurs="unbounded">
  <complexType>
    <choice maxOccurs="unbounded">
      <element name="Encapsulation">
        <complexType>
          <sequence>
            <element ref="tns:Constituent"/>
          </sequence>
          <attributeGroup ref="tns:pkg.grp"/>
        </complexType>
      </element>
      <element name="Composite">
        <complexType>
          <sequence>
            <element ref="tns:Constituent" maxOccurs="unbounded"/>
          </sequence>
          <attributeGroup ref="tns:pkg.grp"/>
        </complexType>
      </element>
    </choice>
  </complexType>
</element>
</sequence>
<attribute ref="tns:id" use="required"/>
</complexType>
</element>
<element name="Comment">
  <complexType>
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute ref="xml:lang"/>
      </extension>
    </simpleContent>
  </complexType>
</element>
<element name="SimplePart">
  <complexType>
    <simpleType name="syncReplyMode.type">
      <enumeration value="mshSignalsOnly"/>
      <enumeration value="responseOnly"/>
      <enumeration value="signalsAndResponse"/>
      <enumeration value="signalsOnly"/>
      <enumeration value="none"/>
    </restriction>
  </simpleType>
  <complexType name="service.type">
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute name="type" type="tns:non-empty-string"/>
      </extension>
    </simpleContent>
  </complexType>
  <complexType name="protocol.type">
    <simpleContent>
      <extension base="tns:non-empty-string">
        <attribute ref="tns:version"/>
      </extension>
    </simpleContent>
  </complexType>
  <attribute name="idref" type="IDREF"/>
  <attribute name="id" type="ID"/>
  <attribute name="version" type="tns:non-empty-string"/>
  <attribute name="syncReplyMode" type="tns:syncReplyMode.type"/>
  <complexType name="SecurityPolicy.type"/>
  <complexType name="CertificateRef.type">
    <attribute name="certId" type="IDREF" use="required"/>
  </complexType>
  <simpleType name="perMessageCharacteristics.type">
    <restriction base="NMTOKEN">
      <enumeration value="always"/>
      <enumeration value="never"/>
      <enumeration value="perMessage"/>
    </restriction>
  </simpleType>
  <simpleType name="actor.type">
    <restriction base="NMTOKEN">
      <enumeration value="urn:oasis:names:tc:ebxml-msg:actor:nextMSH"/>
      <enumeration value="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"/>
    </restriction>
  </simpleType>

```

TCVN ISO/TS 15000-1 : 2007

```
</simpleType>
<simpleType name="messageOrderSemantics.type">
  <restriction base="Name">
    <enumeration value="Guaranteed"/>
    <enumeration value="NotGuaranteed"/>
  </restriction>
</simpleType>
<complexType name="SecurityDetailsRef.type">
  <attribute name="securityId" type="IDREF" use="required"/>
</complexType>
<simpleType name="persistenceLevel.type">
  <restriction base="Name">
    <enumeration value="none"/>
    <enumeration value="transient"/>
    <enumeration value="persistent"/>
    <enumeration value="transient-and-persistent"/>
  </restriction>
</simpleType>
<element name="SecurityDetailsRef" type="tns:SecurityDetailsRef.type"/>
<element name="SecurityDetails">
  <complexType>
    <sequence>
      <element ref="tns:TrustAnchors" minOccurs="0"/>
      <element ref="tns:SecurityPolicy" minOccurs="0"/>
    </sequence>
    <attribute name="securityId" type="ID" use="required"/>
  </complexType>
</element>
<element name="TrustAnchors">
  <complexType>
    <sequence>
      <element name="AnchorCertificateRef" type="tns:CertificateRef.type" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<element name="SecurityPolicy">
  <complexType>
    <sequence>
    </sequence>
  </complexType>
</element>
<simpleType name="accessAuthentication.type">
  <restriction base="NMTOKEN">
    <enumeration value="basic"/>
    <enumeration value="digest"/>
  </restriction>
</simpleType>
</schema>
```

Phụ lục E

(tham khảo)

Thành phần cấu tạo của CPA

E.1 Các đề xuất đối với việc thiết kế các thủ tục tính toán

Việc kiểm tra nhanh các gián đồ phân tử mức cao, **CollaborationProtocolProfile** (CPP) và **CollaborationProtocolAgreement** (CPA), cho thấy rằng một CPA có thể được xem như là kết quả của việc kết hợp các phần tử **PartyInfo** (thông tin bên tham gia) được tìm thấy trong cấu tạo CPPs và sau đó tích hợp các phần tử **PartyInfo** (thông tin bên tham gia) này với các phần tử CPA cùng loại khác, được xem như các phần tử **PartyInfo** (thông tin bên tham gia) đang làm ảnh hưởng nên chu kỳ giá trị CPA.

Việc kết hợp các CPP vào các CPA là cách duy nhất trong các cách kết hợp để các đối tác kinh doanh có thể đưa ra được một đề nghị hoặc một bản phác thảo CPA. Một bản phác thảo CPA cũng có thể được tạo thành từ một khuôn mẫu CPA. Một khuôn mẫu CPA trình bày việc thực hiện đề nghị của một bên trong *quá trình kinh doanh* là sử dụng các giá trị trình giữ chỗ cho việc nhận dạng các khía cạnh của phía bên kia, như phần tử **PartyId** (ID bên tham gia) hoặc TransportEndpoint. Để tạo thành một CPA từ khuôn mẫu CPA, các giá trị trình giữ chỗ được thay thế bởi các giá trị thực tế của đối tác kinh doanh kia. Giá trị thực tế có thể tự được rút ra từ CPP của đối tác kinh doanh kia nếu như CPP này là có hiệu lực hoặc có thể thu được từ việc quản trị viên thực hiện chức năng nhập dữ liệu.

Chúng ta cho là mục đích của bản phác thảo CPA là để chỉ ra việc sử dụng các tiềm năng của họ như các dữ liệu được đưa vào một quá trình thỏa thuận CPA trong trường hợp một bản phác thảo CPA được thẩm tra để thích hợp với cả hai bên, được sửa đổi cho đến khi đạt được một CPA thích hợp hoặc phát hiện ra những cái không thể thực hiện được cho đến khi một bên (hoặc cả hai) có được những năng lực phần mềm bổ sung. Các giao thức và thủ tục thỏa thuận trên hiện nay đã được thiết kế, các yêu cầu của nó đã được định rõ và kết quả của văn bản kỹ thuật này phải có hiệu lực cho việc phát hành chúng. Nói chung, một bản phác thảo CPA sẽ tạo thành một đề nghị về việc liên kết toàn diện một *quá trình kinh doanh* để truyền ra việc thực hiện, trong khi đó việc thỏa thuận sẽ được sử dụng để đưa ra các giá trị chi tiết cho những tham số để mang lại một thỏa thuận cuối cùng. Một tài liệu hướng dẫn cụ thể, NegotiationDescriptorDocument (tài liệu mô tả thương lượng), đưa ra hai trọng tâm là việc các tham số nào có thể được thỏa thuận cũng như việc sắp xếp hoặc thiết lập các giá trị có thể chấp nhận được cho các tham số này.

Mục tiêu của phần còn lại trong phụ lục này là để nhận dạng và mô tả các nhiệm vụ cơ bản cho việc lắp ráp các thủ tục điện toán của bản phác thảo CPA để đạt được mục đích đặt ra. Trong khi đó, văn bản kỹ thuật không có tính quy chuẩn là cung cấp được một thuật toán cho việc tạo thành CPA, cung cấp được một vài hướng dẫn cho người thực hiện.

TCVN ISO/TS 15000-1 : 2007

Các thông tin này có thể giúp đỡ cho người thực hiện phần mềm trong việc thiết kế một hệ thống phần mềm tự động hoá một phần và một phần tương tác có ích cho việc định cấu hình BusinessCollaboration (hợp tác kinh doanh) để hoàn thành tốt các mức độ thao tác.

Trước khi liệt kê và mô tả các nhiệm vụ cơ bản, phần còn lại trong phụ lục này đề cập đến hai lý do cơ bản quan trọng tại sao chúng ta lại tập trung vào các phần nhiệm vụ cụ thể cho việc hình thành CPA hơn là việc cố gắng để cung cấp thuật toán cho việc hình thành CPA. Các lý do này cung cấp một số gợi ý để thực hiện một trong các biện pháp mà họ có thể tùy chỉnh cách tiếp cận của mình để phác thảo các CPA từ các CPP.

E.1.1 Tính biến thiên trong dữ liệu vào

Ưu tiên của người sử dụng là cung cấp một nguồn biến thiên trong các dữ liệu vào tới quá trình hình thành CPA. Chúng ta hãy giả định rằng trong mục này mỗi bên đã cấu tạo sẵn CPP để trở thành những người cộng tác tiềm năng. Thông thường, một bên sẽ có lời đề nghị cộng tác kinh doanh (được chỉ rõ trong tài liệu ProcessSpecification) để thực hiện với người cộng tác mong đợi. Vì thông tin đưa vào máy tính thông thường sẽ được tập trung vào sự ưu tiên của người sử dụng về công việc cộng tác kinh doanh được mong đợi để thêm vào các CPP một cách chính xác.

Công cụ để tạo thành một CPA có thể được truy cập đến phần thông tin của người sử dụng mà thông tin này không được thông báo trong CPP, các thông tin đó có thể đóng góp để hình thành CPA. Một người sử dụng có thể chỉ có một cách lựa chọn thông báo khả năng của hệ thống đó để cho thấy khả năng này là không bị phản đối. Ví dụ, một người sử dụng có thể chỉ thông báo HTTP và bỏ qua FTP, thậm chí ngay cả khi có khả năng đang sử dụng FTP. Lý do của việc bỏ qua FTP có thể liên quan đến việc quản lý các trương mục, các thư mục và mật khẩu của người sử dụng cho các phiên FTP. Mặc dù không thông báo khả năng FTP, phần mềm cấu hình có thể sử dụng kiến thức ngầm về khả năng FTP để tạo thành một CPA với người cộng tác được mong đợi, người cộng tác này ngẫu nhiên chỉ có khả năng FTP cho việc thực hiện một đề nghị cộng tác kinh doanh. Nói cách khác, trong trường hợp này lợi ích kinh doanh có thể quan trọng hơn những điều khoản phản đối. Cả hai kiến thức ngầm và việc chi tiết các trương mục thông tin ưu tiên chính là tính biến thiên của dữ liệu vào trong quá trình tạo thành CPA.

E.1.2 Tính chính xác của biến số trong việc đánh giá các thỏa thuận đưa ra

Điều kiện cho đầu ra của một CPA là tạo ra được hai CPP có mức độ và phạm vi thao tác khác nhau. Nói cách khác, khi một giải pháp tối ưu nhằm để thỏa mãn mọi yêu cầu ở các mức độ khác nhau và các ràng buộc thêm khác là không tồn tại, một bên có thể đưa ra một CPA nhằm thỏa mãn đủ các yêu cầu để thực hiện tốt công việc. Dữ liệu vào của người sử dụng có thể xác định được thực hiện tốt công việc là gì và vì vậy có thể được thay đổi như là việc lựa chọn cấu hình của người sử dụng để biểu lộ sự ưu tiên. Trong thực tế, các sự thỏa hiệp có thể được thực hiện trong sự bảo mật, thông điệp và trong mức độ của ký hiệu và thông điệp báo nhận là xác thực và một số vấn đề khác để có thể tìm thấy một vài điều kiện có thể chấp nhận cho công việc kinh doanh.

Một CPA có thể hỗ trợ một cấu hình có đầy đủ thao tác để đạt được một sự thỏa thuận trong tất cả các mức độ kỹ thuật cần thiết cho việc cộng tác kinh doanh. Trong từng trường hợp, việc phù hợp với khả năng sẽ được tìm thấy trong tất cả các mức độ kỹ thuật có liên quan.

Tuy nhiên, ở đó có thể có các cấu hình về thao tác được chấp nhận tồn tại trong một CPA mà không liên quan gì đến tất cả các bên trong việc phù hợp một quá trình cộng tác kinh doanh. Các thiếu sót có thể tồn tại trong cả các gói, việc bảo mật, chuyển tín hiệu, trong thông điệp và các vùng đáng tin cậy khác và có thể tồn tại ngay cả trong những hệ thống vẫn đang truyền tải các dữ liệu kinh doanh và trong điều kiện đặc biệt có thể tận dụng để sử dụng các ngoại lệ này. Như trong trường hợp, một CPA chỉ có thể phản ánh các điều khoản định cấu hình hoặc chỉ cho phép người sử dụng lờ đi các thiếu sót trong các cấu hình. Một hệ thống không có đủ khả năng để tập trung cho việc cộng tác kinh doanh, cần phải được hỗ trợ một khả năng cụ thể để không từ chối thực hiện các đề nghị nhận được, nhưng có thể vẫn được chấp nhận cho các lý do kinh doanh. Một hệ thống có thể không cần sử dụng tất cả các quá trình hỗ trợ cần thiết, ví dụ, SOAP với các phần đính kèm vẫn còn có thể giải quyết được nhiều phần việc theo quá trình tiến hành công việc mua bán “nhiều phần/ hỗn hợp” và cho phép việc cộng tác kinh doanh được diễn ra. Trên thực tế, nếu không có đủ khả năng có thể truyền tải dữ liệu và khả năng có thể cung cấp dữ liệu liên quan đến việc cộng tác kinh doanh thì có thể việc sắp xếp một vài các tính năng một cách không tạm thời hoặc rõ ràng sẽ mang lại lợi ích kinh doanh tốt hơn cả. Tình hình thao tác từng phần này có thể thỉnh thoảng vẫn còn xảy ra và vì vậy làm ảnh hưởng đến việc trình bày rõ ràng, chính xác một thuật toán sạch cho việc quyết định thế nào là đủ thao tác.

E.2 Các phần nhiệm vụ hình thành CPA

Theo quan điểm kỹ thuật, một CPA cung cấp sự liên kết giữa các văn bản kỹ thuật Business Colaboration (như việc các văn bản kỹ thuật này được định rõ bên trong các tài liệu tham chiếu của ProcessSpecification) với các dịch vụ và giao thức được sử dụng để thực hiện các văn bản kỹ thuật đó. Việc thực hiện được diễn ra với các mức độ khác nhau và đòi hỏi các dịch vụ khác nhau theo từng mức độ trên. Một CPA đạt được một sự liên kết cộng tác có đầy đủ các thao tác có thể được coi như đang đạt được thao tác từ ứng dụng đến việc tích hợp ứng dụng. Nhiều CPA có thể không đạt mục đích này tuy nhiên vừa có thể được sử dụng vừa có thể được chấp nhận cho việc cộng tác giữa các bên. Tất nhiên, nếu không có thể tìm ra được khả năng truyền tải dữ liệu phù hợp, một CPA không thể cung cấp nhiều các biện pháp cho việc tích hợp các thao tác. Tương tự như vậy, từng phần của CPA có thể được bỏ lại cho việc thực hiện các hoạt động quan trọng của hệ thống trước khi thực hiện tốt việc hoàn thành quá trình từ ứng dụng đến việc tích hợp ứng dụng. Tuy nhiên, việc tích hợp các phần nói trên có thể đủ cho việc cộng tác được diễn ra và được hưởng phần từ việc tăng các mức độ tự động hoá.

Trong thực tế, quá trình hình thành CPA có thể đem lại một CPA đầy đủ, một kết quả không như mong đợi, một danh sách các thiếu sót để làm chậm lại một cuộc hội thoại với người sử dụng hoặc thậm chí có thể một CPA chỉ thực hiện vừa đủ tốt việc thao tác từng phần cho các đối tác kinh doanh.

Vì cả hai việc có khả năng phù hợp và các thao tác có thể đều là các vấn đề quan trọng, cho nên nhiệm vụ hình thành CPA chính là việc tìm ra các khả năng so khớp với từng mức độ và dịch vụ khác nhau. Sau đây, chúng ta bắt đầu mô tả các nhiệm vụ rất quan trọng cho việc hình thành CPA.

E.3 Việc hình thành CPA từ các CPP: Nội dung nhiệm vụ

Để đơn giản hoá việc thảo luận, sau đây hãy thừa nhận rằng chúng ta đang xem xét các nhiệm vụ trước mắt bằng một tác nhân phần mềm khi:

1. đã biết người cộng tác được mong đợi và *CPP* của người cộng tác đã được truy lục;
2. **processSpecification** giữa chúng ta và người cộng tác được mong đợi là đã được lựa chọn;
3. trong tác nhân phần mềm để vận hành việc cộng tác kinh doanh (giới hạn **BilaryCollaborations**), **Service, Action** và các phân tử **Role** (vai trò) cụ thể là đã biết và;
4. cuối cùng, là biết được các khả năng đã thông báo trong *CPP*.

Để cho chi tiết hơn, có thể mở rộng các thảo luận về việc sử dụng ví dụ “3A4” ebBPSS và các CPP của công ty A và B, các vấn đề này được tìm thấy đầy đủ trong các phụ lục của tài liệu này và cũng có thể tìm được trong trang Web của Ủy ban kỹ thuật OASIS ebXML CPPA. Để tuyệt đối, có thể cho rằng thông tin về các khả năng là đã được giới hạn để tìm thấy những gì trong các tác nhân của CPP và trong CPP của đối tác được mong đợi của ta. Giả định rằng quan điểm của công ty A để lắp ráp một bản phác thảo CPA. Chú ý rằng, điều này là không bảo đảm cho việc các bản phác thảo CPA tương tự có thể được đưa ra từ các quan điểm khác nhau theo một trình tự giống nhau.

Nói chung, nhiệm vụ cơ bản gồm có việc tìm ra các sự so khớp giữa khả năng và khả năng của người cộng tác được mong đợi trong các mức độ khác nhau của các ngăn xếp giao thức cộng tác và về việc cung cấp các dịch vụ theo các mức độ khác nhau nói trên. Do không cần thiết phải mô tả một cách chi tiết, các ngăn xếp này ít nhất cũng được nhận ra bằng một mức độ ứng dụng và mức độ chuyển thông điệp. Mức độ ứng dụng bị ảnh hưởng bởi một văn bản kỹ thuật chảy trong *quá trình kinh doanh* như ebBPSS. Mức độ chuyển thông điệp sẽ gồm có một số thủ tục và các lựa chọn những giao thức chuyển giao liên quan, lựa chọn việc bảo vệ, kết hợp các bộ phận và các mẫu thông điệp liên quan (bao gồm cả những loại thông điệp báo nhận khác nhau, những thông điệp sai và các vấn đề tương tự khác).

Trong thực tế, thông thường việc lắp ráp các nhiệm vụ vào trong quá trình có sử dụng máy điện toán là có thể được làm theo hướng thực hiện theo một trình tự đích xác. Toàn bộ trình tự này sẽ phản ánh cấu trúc tuyệt đối của CPA: Cam kết đầu tiên là bảo đảm các nhiệm vụ được nói trên là một sự so khớp đối với quá trình cộng tác kinh doanh. Không thể tìm ra được rằng các đối tác đều có thể tham gia thành công trong **ProcessSpecification** giống nhau, đó là chi tiết không quan trọng trong việc thi hành hết các lựa chọn cho việc thực hiện. Sau đó, xem xét các việc phù hợp trong phạm vi các phần hợp thành việc liên kết mà đã được thông báo trong quá trình cộng tác kinh doanh, kiểm tra trước các việc phù hợp không thể thiếu nhất (Transport – Có liên quan) và tiếp tục kiểm tra các lớp phản ánh việc tích hợp các thao tác trong các gói, việc bảo mật, các mẫu ký hiệu và mẫu giao thức, v..v.

Với các mô tả chung cơ bản nhưng ngắn gọn nói trên, chúng ta hãy đi đến xem xét các nhiệm vụ cơ bản một cách hết sức chi tiết.

E.4 Nhiệm vụ tạo phù hợp quá trình cộng tác kinh doanh

Công ty A đã có thông báo trong phạm vi CPP của công ty ít nhất một phần tử **PartyInfo** (thông tin bên tham gia). Để được các kết quả hiện hành, điểm trọng tâm ban đầu quan trọng nhất là nằm trong tất cả các phần tử **PartyInfo** (thông tin bên tham gia) cùng loại với đường dẫn **/CollaborationProtocolProfile/PartyInfo/Role**. Từng phần tử thuộc loại này có một phần tử con là **ProcessSpecification**. Nhiệm vụ so khớp ban đầu (chắc chắn rằng việc xem xét một cách cẩn thận hơn được xem như là một nhiệm vụ lọc) là lựa chọn các nút nơi mà **ProcessSpecification** có ích cho chúng ta trong việc xây dựng một CPA. Việc kiểm tra các giá trị thuộc tính **name** (tên), **xlink:href** hoặc **uuid** cho phép chúng ta lựa chọn bằng cách so sánh các trị số. Giá trị cuối cùng cho việc phù hợp các văn bản kỹ thuật của quá trình ebBPSS là giá trị được tìm thấy trong thuộc tính **ProcessSpecification/@uuid**.

E.4.1 Phù hợp ProcessSpecification/Role (quy định quá trình/vai trò) và Actions (hành động)

Lọc và lựa chọn ban đầu

Về bản chất, nhiệm vụ đầu tiên là tìm ra được hai nhóm nút **CollaborationRole** trong phạm vi và trong các tài liệu CPP của người cộng tác của chúng ta, mà tại các nút này các **ProcessSpecification** là đồng nhất và đủ khả năng để giá trị lợi ích được sinh ra lớn hơn. Nói cách khác, là chúng ta phải có **CollaborationRole** với **ProcessSpecification/@name='PIP3A4RequestPurchaseOrder'**. Nhiệm vụ này là thuận lợi nhưng không cần thiết để sử dụng thuộc tính **name** (tên) trong việc thực hiện lựa chọn này.

Tiếp theo, chúng ta tiếp tục lọc các nhóm nút nói trên. Chúng ta đưa ra được giá trị phần tử **Role** (vai trò) cho việc tham gia trong **ProcessSpecification**. Đối với công ty A, phần tử **Role** (vai trò) này có thuộc tính **name** (tên) với giá trị 'Buyer - Người mua'. Bởi vì, ở đây chúng ta chỉ tính toán đến BinaryCollaboration trong thuật ngữ ebBPSS (hoặc tương đương theo các ngôn ngữ lập trình chảy khác nhau), do đó, chúng ta chỉ có thể đạt được lợi ích khi trong các nhóm nút **CollaborationRole** trong phạm vi CPP của người cộng tác có một giá trị **Role** đủ khả năng 'Seller - Người bán'. Vì vậy, chúng ta cho rằng chúng ta đã thu hẹp điểm trọng tâm đến các nhóm nút **CollaborationRole** trong CPP của công ty A với **Role/@name='Buyer'** và trong các nhóm nút **CollaborationRole** của công ty B với **Role/@name='Seller'**.

Để có thêm các sự cộng tác thông thường, như trong các **MultiPartyCollaboration** của ebBPSS, chúng ta cần phải biết danh sách các chức năng có thể dùng được trong phạm vi của quá trình và theo dõi từng **CollaborationRole**, lựa chọn các giá trị **Role** phù hợp đúng cách thức với những người tham gia. ở đây, chúng ta không thảo luận là nhiệm vụ so khớp/lọc cho quá trình cộng tác có gồm nhiều hơn hai chức năng hay không, như các CPA của các bên liên quan là không nằm trong phạm vi phiên bản 2.0 của văn bản kỹ thuật này.

E.5 Tiến hành phù hợp

Sau khi lọc các **CollaborationRole** với **ProcessSpecification** được yêu cầu, chúng ta phải tìm ra được một **CollaborationRole** trong CPP nơi mà chúng ta đóng vai chức năng người mua và một **CollaborationRole** trong CPP của công ty B - Người cộng tác được mong đợi - Đóng vai chức năng người bán.

Nhiệm vụ tiếp theo là định vị việc liên kết các ứng viên cụ thể liên quan đến việc hình thành CPA. Đó là việc liên kết dịch vụ và các hành động. Để đơn giản, chúng ta coi các nhiệm vụ so khớp chi tiết như việc đưa ra một tiêu chuẩn cho trường hợp cộng tác bao gồm một hành động yêu cầu, tiếp theo là một hành động đáp lại yêu cầu. Về phiên bản 2.0 của văn bản kỹ thuật này, đa số các nhiệm vụ so khớp sẽ bao hàm việc phù hợp các phần hợp thành có liên quan của các phần tử **ThisPartyActionBinding** (quy định hoạt động bên tham gia này) trong CPP theo **CollaborationRole/ServiceBinding/CanSend/** và theo **CollaborationRole/ServiceBinding/CanReceive**.

E.5.1 Hành động phù hợp và lựa chọn tương quan các PackageID (id gói) và ChannelId (id kênh truyền)

Trong các CPP, theo mỗi phần tử **CollaborationRole** (vai trò hợp tác)/**ServiceBinding/CanSend/** và **CollaborationRole/ServiceBinding/CanReceive** là các danh sách **ThisPartyActionBinding**. Đối với các khuôn mẫu hợp tác yêu cầu - Đáp lại yêu cầu, chúng ta sẽ đạt được lợi ích trong các việc phù hợp sau:

1. trong việc liên kết **CanSend / ThisPartyActionBinding** của bên yêu cầu với **CanReceive / ThisPartyActionBinding** của bên đáp lại yêu cầu cho hành động yêu cầu và;
2. trong việc liên kết **CanSend / ThisPartyActionBinding** của bên yêu cầu với **CanReceive / ThisPartyActionBinding** của bên đáp lại yêu cầu cho hành động đáp lại yêu cầu.

Các việc liên kết tương quan nói trên cho ta các tham chiếu để chi tiết các phần hợp thành, mà các phần hợp thành này là cần thiết để so khớp cho một thỏa thuận đầy đủ các thao tác giữa các phần. Trường hợp 1 gắn liền với việc yêu cầu. Trường hợp 2 gắn liền với việc đáp lại yêu cầu. Ví dụ, đối với công ty A, có thể tìm theo **CanSend**:

```
<tp:ThisPartyActionBinding tp:action="Purchase Order Request Action"
tp:packageId="CompanyA_RequestPackage">
  <tp:BusinessTransactionCharacteristics ... />
  <tp:ActionContext tp:binaryCollaboration="Request Purchase Order"
tp:businessTransactionActivity="Request Purchase Order"
tp:requestOrResponseAction="Purchase Order Request Action"/>
  <tp:ChannelId>asyncChannelA1</tp:ChannelId>
</tp:ThisPartyActionBinding>
```

Tương quan với trường hợp này, đối với công ty B, có thể tìm theo **CanReceive**:

```
<tp:ThisPartyActionBinding tp:action="Purchase Order Request Action"
tp:packageId="CompanyB_RequestPackage">
  <tp:BusinessTransactionCharacteristics ... />
  <tp:ActionContext tp:binaryCollaboration="Request Purchase Order"
```



```
tp:businessTransactionActivity="Request Purchase Order" tp:requestOrResponseAction="Purchase Order Request Action"/>
```

```
<tp:ChannelId>asyncChannelB1</tp:ChannelId>
```

```
</tp:ThisPartyActionBinding>
```

Thông thường, sự tương quan của các phần tử (khi chúng ta đang đề cập đến BPSS **BinaryCollaborations** hoặc tương đương trong các cách trình bày khác) có thể được dựa vào trạng thái bằng nhau của các giá trị **action** (hoặc **requestOrResponseAction**). Để chi tiết hơn nữa sự tương quan của các phần tử, có thể tận dụng việc kiểm tra và so sánh một cách chi tiết hơn các giá trị trong các phần tử con **ActionContext** của hai phần tử **CanSend** và **CanReceive** có liên quan.

Như đã nói ở trên, chúng ta đã minh họa được việc phù hợp của **CanSend** và **CanReceive** cho các liên kết không đồng bộ. Tất cả các liên kết **CanSend** là các liên kết cùng theo một phần tử **ServiceBinding**, các liên kết này là không đồng bộ và tận dụng các kết nối TCP khác nhau để các khía cạnh của **CanSend** được bắt đầu trong một cổng nghe TCP. Để trình bày chi tiết việc liên kết cho hoạt động gửi đồng bộ, khi một quy ước được chấp nhận mà nhờ đó phần tử **CanReceive** cho một người gửi được đặt theo Phần tử **CanSend** (có thể gửi) của nó.

Việc liên kết đó được minh họa như sau:

```
<tp:CanSend>
```

```
<tp:ThisPartyActionBinding
```

```
tp:id="companyA_ABID6"
```

```
tp:action="Purchase Order Request Action"
```

```
tp:packageId="CompanyA_RequestPackage">
```

```
<tp:BusinessTransactionCharacteristics
```

```
tp:isNonRepudiationRequired="true"
```

```
tp:isNonRepudiationReceiptRequired="true"
```

```
tp:isConfidential="transient"
```

```
tp:isAuthenticated="persistent"
```

```
tp:isTamperProof="persistent"
```

```
tp:isAuthorizationRequired="true"
```

```
tp:timeToAcknowledgeReceipt="PT2H"
```

```
tp:timeToPerform="P1D"/>
```

```
<tp:ActionContext
```

```
tp:binaryCollaboration="Request Purchase Order"
```

```
tp:businessTransactionActivity="Request Purchase Order"
```

```
tp:requestOrResponseAction="Purchase Order Request Action"/>
```

```
<tp:ChannelId>syncChannelA1</tp:ChannelId>
```

```
</tp:ThisPartyActionBinding>
```

```
<tp:CanReceive>
```

```
<tp:ThisPartyActionBinding
```

```
tp:id="companyA_ABID7"
```

```
tp:action="Purchase Order Confirmation Action"
```

```
tp:packageId="CompanyA_SyncReplyPackage">
```

```
<tp:BusinessTransactionCharacteristics
```

```
tp:isNonRepudiationRequired="true"
```

```
tp:isNonRepudiationReceiptRequired="true"
```

```
tp:isConfidential="transient"
```

```
tp:isAuthenticated="persistent"
```

```
tp:isTamperProof="persistent"
```

```
tp:isAuthorizationRequired="true"
```

```
tp:timeToAcknowledgeReceipt="PT2H"
```

```
tp:timeToPerform="P1D"/>
```

```
<tp:ActionContext
```

```
tp:binaryCollaboration="Request Purchase Order"
```

```
tp:businessTransactionActivity="Request Purchase Order"
```

```

    tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
<tp:ChannelId>syncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanReceive>
<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyA_ABID8"
    tp:action="Exception"
    tp:packageId="CompanyA_ExceptionPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent"
      tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true"
      tp:timeToAcknowledgeReceipt="PT2H"
      tp:timeToPerform="P1D"/>
    <tp:ChannelId>syncChannelA1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
</tp:CanReceive>
</tp:CanSend>

```

Sự phụ thuộc này cũng sẽ được mang sang phía bên nhận đồng bộ, một trong một số các Phần tử **CanSend** (có thể gửi) theo phần tử **CanReceive** được sử dụng để mô tả việc nhận một yêu cầu ban đầu. Một minh họa việc liên kết đồng bộ của công ty B như sau::

```

<tp:CanReceive>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID8"
    tp:action="Purchase Order Request Action"
    tp:packageId="CompanyB_SyncReplyPackage">
    <tp:BusinessTransactionCharacteristics
      tp:isNonRepudiationRequired="true"
      tp:isNonRepudiationReceiptRequired="true"
      tp:isConfidential="transient"
      tp:isAuthenticated="persistent" tp:isTamperProof="persistent"
      tp:isAuthorizationRequired="true" tp:timeToAcknowledgeReceipt="PT5M"
      tp:timeToPerform="PT5M"/>
    <tp>ActionContext
      tp:binaryCollaboration="Request Purchase Order"
      tp:businessTransactionActivity="Request Purchase Order"
      tp:requestOrResponseAction="Purchase Order Request Action"/>
    <tp:ChannelId>syncChannelB1</tp:ChannelId>
  </tp:ThisPartyActionBinding>
  <tp:CanSend>
    <tp:ThisPartyActionBinding
      tp:id="companyB_ABID6"
      tp:action="Purchase Order Confirmation Action"
      tp:packageId="CompanyB_ResponsePackage">
      <tp:BusinessTransactionCharacteristics
        tp:isNonRepudiationRequired="true"
        tp:isNonRepudiationReceiptRequired="true"
        tp:isConfidential="transient"
        tp:isAuthenticated="persistent"
        tp:isTamperProof="persistent"
        tp:isAuthorizationRequired="true"
        tp:timeToAcknowledgeReceipt="PT5M"
        tp:timeToPerform="PT5M"/>
    </tp:ThisPartyActionBinding>
  </tp:CanSend>

```

```

<tp:ActionContext
  tp:binaryCollaboration="Request Purchase Order"
  tp:businessTransactionActivity="Request Purchase Order"
  tp:requestOrResponseAction="Purchase Order Confirmation Action"/>
<tp:ChannelId>syncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
<tp:CanSend>
  <tp:ThisPartyActionBinding
    tp:id="companyB_ABID7"
    tp:action="Exception"
    tp:packageId="CompanyB_ExceptionPackage">
  <tp:BusinessTransactionCharacteristics
    tp:isNonRepudiationRequired="true"
    tp:isNonRepudiationReceiptRequired="true"
    tp:isConfidential="transient"
    tp:isAuthenticated="persistent"
    tp:isTamperProof="persistent"
    tp:isAuthorizationRequired="true"
    tp:timeToAcknowledgeReceipt="PT5M"
    tp:timeToPerform="PT5M"/>
  <tp:ChannelId>syncChannelB1</tp:ChannelId>
</tp:ThisPartyActionBinding>
</tp:CanSend>
</tp:CanReceive>

```

E.5.2 Phù hợp và kiểm tra các chi tiết của DeliveryChannel (Kênh truyền)

Cho đến nay, đa số công việc phù hợp là nhằm bảo đảm để tìm ra hai liên kết hành động tương quan và vì vậy việc phù hợp được thực hiện chức năng như một cái máy lọc. Tuy nhiên, ngay khi có hai liên kết hành động tương quan, công việc gồm có việc kiểm tra sự so khớp các kích thước khác nhau của truyền tải thao tác, bảo mật quá trình truyền tải, khả năng tương thích PKI cho các nhiệm vụ khác nhau, thỏa thuận về các nét đặc trưng của thông điệp (thông điệp xác thực, việc bao số, lời báo nhận được ký hiệu (tối thiểu để xác nhận việc không từ chối)), nguồn gốc của việc không từ chối, các chi tiết đóng gói.

Khi mà có các liên kết hành động, IDREFs cung cấp các tham chiếu cho các phần cấu thành cơ bản của việc so sánh. Ví dụ, khi so sánh các chi tiết gói, IDREFS yêu cầu là được tìm ra theo **CanSend/ThisPartyActionBinding/@packageId** và theo **CanReceive/ThisPartyActionBinding/@packageId** trong phạm vi CPP khác. Đối với yêu cầu "hành động yêu cầu đặt mua" của công ty A, **IDREF** gói được tìm trong:

```
tp:packageId="CompanyA_RequestPackage"
```

và giá trị **IDREF** này liên quan đến:

```

<tp:Packaging tp:id="CompanyA_RequestPackage">
  <tp:ProcessingCapabilities tp:parse="true" tp:generate="true"/>
  <tp:CompositeList>
    <tp:Composite tp:id="CompanyA_RequestMsg" tp:mimetype="multipart/related"
      tp:mimeparameters="type=text/xml;">
  <tp:Constituent tp:idref="CompanyA_MsgHdr"/>
  <tp:Constituent tp:idref="CompanyA_Request"/>
  </tp:Composite>
</tp:CompositeList>

```

</tp:Packaging>

Đối với yêu cầu “hành động yêu cầu đặt mua” của công ty A, IDREF của kênh phát được tìm ra trong:

<tp:ChannelId>asyncChannelA1</tp:ChannelId>

và giá trị IDREF này dựa vào phần tử ID đó, cụ thể là:

<tp:DeliveryChannel tp:channelId="asyncChannelA1" tp:transportId="transportA1"

tp:docExchangeId="docExchangeA1">

<tp:MessagingCharacteristics

tp:syncReplyMode="none"

tp:ackRequested="always"

tp:ackSignatureRequested="always"

tp:duplicateElimination="always"/>

</tp:DeliveryChannel>

Hai tham chiếu quyết định còn lại cho việc hiểu biết quá trình liên kết là **DeliveryChannel/@transportId** và **DeliveryChannel/@docExchangeId** được tìm ra trong các thuộc tính của **DeliveryChannel**.

Ví dụ, đối với công ty A, chúng ta tìm **transportID**="transportA1" và **docExchangeId**="docExchangeA1" là các **IDREF** về thông tin liên kết tiếp theo với **DeliveryChannel**, "asyncChannelA1". Việc giải quyết các tham chiếu nói trên, có thể thu được:

<tp:Transport tp:transportId="transportA1">

<tp:TransportSender>

<tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>

<tp:TransportClientSecurity>

<tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>

<ClientCertificateRef tp:certId="CompanyA_ClientCert"/>

<tp:ServerSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>

</tp:TransportClientSecurity>

</tp:TransportSender>

<tp:TransportReceiver>

<tp:TransportProtocol tp:version="1.1">HTTP</tp:TransportProtocol>

<tp:Endpoint tp:uri="https://www.CompanyA.com/servlets/ebxmlhandler/async" tp:type="allPurpose"/>

<tp:TransportServerSecurity>

<tp:TransportSecurityProtocol tp:version="3.0">SSL</tp:TransportSecurityProtocol>

<tp:ServerCertificateRef tp:certId="CompanyA_ServerCert"/>

<tp:ClientSecurityDetailsRef tp:securityId="CompanyA_TransportSecurity"/>

</tp:TransportServerSecurity>

</tp:TransportReceiver>

</tp:Transport>

Đối với transportID "transportA1" và

<tp:DocExchange tp:docExchangeId="docExchangeA1">

<tp:ebXMLSenderBinding tp:version="2.0">

<tp:ReliableMessaging>

<tp:Retries>3</tp:Retries>

<tp:RetryInterval>PT2H</tp:RetryInterval>

<tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>

</tp:ReliableMessaging>

<tp:PersistDuration>P1D</tp:PersistDuration>

<tp:SenderNonRepudiation>

<tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#

</tp:NonRepudiationProtocol>

<tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1

</tp:HashFunction>

<tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1

</tp:SignatureAlgorithm>

<tp:SigningCertificateRef tp:certId="CompanyA_SigningCert"/>

```

</tp:SenderNonRepudiation>
<tp:SenderDigitalEnvelope>
<tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
<tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
<tp:EncryptionSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
</tp:SenderDigitalEnvelope>
</tp:ebXMLSenderBinding>
<tp:ebXMLReceiverBinding tp:version="2.0">
<tp:ReliableMessaging>
<tp:Retries>3</tp:Retries>
<tp:RetryInterval>PT2H</tp:RetryInterval>
<tp:MessageOrderSemantics>Guaranteed</tp:MessageOrderSemantics>
</tp:ReliableMessaging>
<tp:PersistDuration>P1D</tp:PersistDuration>
<tp:ReceiverNonRepudiation>
<tp:NonRepudiationProtocol>http://www.w3.org/2000/09/xmldsig#
</tp:NonRepudiationProtocol>
<tp:HashFunction>http://www.w3.org/2000/09/xmldsig#sha1
</tp:HashFunction>
<tp:SignatureAlgorithm>http://www.w3.org/2000/09/xmldsig#dsa-sha1
</tp:SignatureAlgorithm>
<tp:SigningSecurityDetailsRef tp:securityId="CompanyA_MessageSecurity"/>
</tp:ReceiverNonRepudiation>
<tp:ReceiverDigitalEnvelope>
<tp:DigitalEnvelopeProtocol tp:version="2.0">S/MIME</tp:DigitalEnvelopeProtocol>
<tp:EncryptionAlgorithm>DES-CBC</tp:EncryptionAlgorithm>
<tp:EncryptionCertificateRef tp:certId="CompanyA_EncryptionCert"/>
</tp:ReceiverDigitalEnvelope>
</tp:ebXMLReceiverBinding>
</tp:DocExchange>

```

Đối với docExchangeId, docExchangeA1.

E.5.2.1 Việc phù hợp Transport (Truyền tải)

Đầu tiên, việc phù hợp **Transport** là bao hàm việc phù hợp các khả năng **Transport/TransportSender/TransportProtocol** của người yêu cầu với các khả năng **Transport/TransportReceiver/TransportProtocol** được tìm ra theo người cộng tác nhận được yêu cầu trên. Cũng như một vài so khớp có thể tồn tại, các so khớp này có thể được sử dụng trong việc định hình một bản phác thảo và có thể cung cấp các khía cạnh khác để có thể so khớp được một cách hoàn toàn. Mỗi CPP được thừa nhận để có được danh sách các giao thức truyền tải ưa thích đầu tiên (như việc xác định rõ danh sách các liên kết bằng cách tham chiếu phần tử **Transport**), nhưng các tác động khác nhau có thể dẫn đến việc phụ thuộc vào các CPP là được sử dụng đầu tiên cho quá trình tìm kiếm các sự so khớp. Nói chung, các cách giải quyết khác nhau là bỏ sót một giai đoạn nhất định của thỏa thuận CPA, tiếp theo là việc đề nghị của một bản phác thảo CPA. Thỏa thuận có thể được thực hiện bởi các hành động rõ ràng của người sử dụng, nhưng có thể được xảy ra để thích hợp với việc tăng mức độ tự động hoá.

Thứ hai, việc phù hợp sự truyền tải bao hàm việc phù hợp các khả năng **TransportSender/TransportProtocol** về việc trả lời người cộng tác với các khả năng **TransportReceiver/**

TransportProtocol được tìm ra theo người cộng tác nhận được câu trả lời trên, việc phù hợp này là đặc trưng cho người cộng tác đã gửi một yêu cầu.

Cũng như một vài so khớp có thể tồn tại, các so khớp này có thể được sử dụng trong việc định hình một bản phác thảo. Tuy nhiên, trong trường hợp cụ thể có thể không cần thiết thực hiện việc phù hợp thứ hai này trong **TransportProtocol**. Nếu chúng ta đang sử dụng HTTP hoặc một vài giao thức hỗ trợ cho việc trả lời một cách đồng bộ và **DeliveryChannel** có một kết quả **MessagingCharacteristics** có thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) với một giá trị "signalsAndResponse", tiếp theo đó mọi thứ trở lại một cách đồng bộ và không cần thiết để so khớp trong **TransportProtocol** cho **DeliveryChannel** yêu cầu.

Nếu **TransportSecurity** là hiện hữu, tiếp theo đó có thể được kiểm tra thêm vào. Đầu tiên, **TransportSender/TransportClientSecurity/TransportSecurityProtocol** phải tương thích với **TransportReceiver/TransportServerSecurity/TransportSecurityProtocol**. Thứ hai, nếu hoặc phần tử **TransportSender/TransportClientSecurity/CilentSecurityDetailsRef** hoặc phần tử **TransportSender/TransportClientSecurity/ServerSecurityDetailsRef** là hiện hữu và IDREF tham chiếu một phần tử bao gồm một vài **AnchorCertificateRef**, tiếp theo đó một cơ hội tồn tại để kiểm tra sự tin cậy vào việc chứng nhận PKI của một bên được sử dụng trong **TransportSecurityProtocol**. Ví dụ, bằng cách giải quyết giá trị IDREF trong **TransportSender/TransportClientSecurity/CilentSecurityDetailsRef@certId**, có thể thu được sự chứng nhận của máy khách về việc xác nhận các đề nghị là đúng. Bằng cách giải quyết các IDREF từ **AnchorCertificateRef**, có thể xác định được hay không rằng sự chứng nhận của máy khách về các đề nghị sẽ là "chain to a trusted root" theo các PKI của phía bên máy server (máy chủ). Trong hiệu lực của sự chứng nhận của máy server (máy chủ), các nhận xét tương tự áp dụng để kiểm tra được tìm ra bằng cách giải quyết **TransportReceiver/TransportServerSecurity/ServerCertificateRef**. Sự chứng nhận của máy server (máy chủ) nói trên khi có thể kiểm tra ngược lại các nguồn tin cậy CA sẽ được tìm ra bằng cách giải quyết **TransportSender/TransportClientSecurity/ServerSecurityDetailsRef@securityId** và việc tìm ra các chứng nhận CA (hoặc các chuỗi chứng nhận CA) trong các phần tử **KeyInfo** theo phần tử **Certificate** (chứng chỉ) thu được bằng cách giải quyết IDREF được tìm ra trong **AnchorCertificateRef@certId**.

Khi mà các so khớp tồn tại cho các phần hợp thành **Transport** tương quan, chúng ta sẽ tìm ra được một giải pháp thao tác trong mức độ truyền tải. Nếu không, không có CPA nào sẽ có thể dùng được và một kẻ hở sẽ được nhận dạng ra rằng nó cần thiết phải được sửa chữa bằng bất kể các thủ tục xử lý ngoại lệ nào. Tiếp theo, chúng ta hãy tính toán đến các khả năng khác cần thiết để so khớp các giải pháp thao tác giữa các phần một cách tập trung nhất.

E.5.2.2 Việc kiểm tra BusinessTransactionCharacteristics (Đặc điểm giao dịch kinh doanh), **DeliveryChannel** (Kênh truyền) và **MessagingCharacteristics** (Đặc điểm kênh truyền)

Theo mỗi liên kết hành động tương quan, một phần tử con của **DeliveryChannel**, **MessagingCharacteristics** là có một vài các thuộc tính quan trọng cho các nhiệm vụ hình thành CPA.

Các thuộc tính có các hàm ý rộng hơn là **syncReplyMode**, **ackRequested** và **ackSignatureRequested**; đối với các thuộc tính **DuplicateElimination** (loại trừ sao chép lại) và **actor**, tính tương thích tồn tại khi các thuộc tính là được tìm ra theo **CanSend** và **CanReceive** **DeliveryChannels** có các giá trị giống nhau. Như tên của phần tử gợi ý tất cả các tính năng của các **DeliveryChannel** gắn liền với lớp thông điệp.

Mặt khác, **BusinessTransactionCharacteristics** được tìm ra theo **ThisPartyActionBinding** bao gồm các thuộc tính phản ánh sự đa dạng của các tính năng gắn liền với đề nghị bảo mật và các đặc tính của công việc kinh doanh là được thực hiện phù hợp với các **DeliveryChannel**. Các đặc tính này có thể được hàm ý trong những khả năng mà các khả năng này là cần thiết trong việc chi tiết hơn nữa các phần cấu thành của các phần tử **DeliveryChannel** (kênh truyền), như trong phần tử **Packaging** (đóng gói). Khi đang sử dụng các văn bản kỹ thuật của quá trình BPSS, các đặc tính nói trên có thể được định rõ trong phạm vi công việc kinh doanh. Tuy nhiên, các đặc tính của phần tử **BusinessTransactionCharacteristics** (đặc điểm giao dịch kinh doanh) là sẽ có tác dụng trong việc thực hiện đầy đủ **BusinessTransaction** và có thể quan trọng hơn việc định rõ các giá trị đã được tìm ra trong các văn bản kỹ thuật của chương trình BPSS. Bởi vì, các đặc tính là gồm nhiều loại khác nhau, do đó, việc trình bày chi tiết quá trình thực hiện đầy đủ các đặc tính có thể được trải rộng ra các phần tử khác đã được tham chiếu trong phạm vi của các phần tử **DeliveryChannel**.

Các thuộc tính nói trên áp dụng cho hoặc một kênh phát yêu cầu hoặc một kênh phát đáp lại yêu cầu, nhưng có thể có tác động đến hoặc người gửi hoặc người nhận (hoặc cả hai) trong cùng một kênh. Mặt khác, các thuộc tính này ảnh hưởng đến các tin báo nhận, ví dụ, việc mô tả mối tương quan của các phần tử **DeliveryChannel** (kênh truyền) bằng cách định rõ cách hoạt động chính là để mô tả rõ nội dung của thông điệp đáp lại.

Đa số việc kiểm tra cơ bản sự tương thích của một số thuộc tính hoặc trong **MessagingCharacteristics** hoặc trong **BusinessTransactionCharacteristics** là kiểm tra các thuộc tính tương ứng với **DeliveryChannel** của bên người gửi được tham chiếu bởi **CanSend/ThisPartyActionBinding/ChannelId** và tương ứng với **DeliveryChannel** của bên nhận được tham chiếu bởi **CanReceive/ThisPartyActionBinding/ChannelId**. Nếu các thuộc tính là không tương ứng và tất cả các liên kết của cả hai bên sẽ được xem xét, thì một bản phác thảo CPA sẽ trình bày một sự thỏa hiệp để thiết lập một vài chức năng phổ biến được trình bày bởi các thuộc tính.

Trong các thảo luận dưới đây, chúng ta sẽ tính toán đến một số thuộc tính trong hai phần tử **Characteristic** và liên kết chúng lại để thêm vào các chi tiết cơ bản cho quá trình thực hiện. Một trong các thuộc tính đó là **Packaging**.

Ở một mức độ cao, thỏa thuận cơ bản trong gói là vấn đề tương thích của gói được tạo ra trong bên gửi với gói được phân tách trong bên nhận. Vì thế, việc kiểm tra gói cơ bản là việc kiểm tra tính tương thích

TCVN ISO/TS 15000-1 : 2007

gói theo Phần tử **CanSend** (có thể gửi) của hành động người gửi với gói theo phần tử **CanReceive** của hành động tương tự bên người nhận.

Về hiệu quả, việc trình bày các khả năng của gói phân tách/điều khiển có thể sử dụng cả hai là các ký tự đại diện và bản sao chép và các khả năng này cũng có thể cần thiết để mở nhanh khuôn thức dữ liệu được sử dụng trong bên tạo ra. Ví dụ, việc tính toán đến *SimplePart*:

```
<tp:SimplePart tp:id="IWild" tp:mimetype="*/*/"/>
```

Bằng các giá trị *mimetype* đại diện, chúng ta trình bày khả năng về việc chấp nhận bất cứ dữ liệu nào và sẽ so khớp mọi kiểu MIME cụ thể. Ngoài ra, việc tính toán đến một *Constituent* xuất hiện trong phạm vi một *Composite*:

```
<tp:Constituent tp:idref="MsgHdr"/>  
  <tp:Constituent minOccurs="0"  
    maxOccurs="10" tp:idref="IWild"/>
```

Chú thích trên đáp ứng cho việc bắt giữ lại khả năng điều khiển mọi phần thân MIME bất kỳ nào trong phạm vi **Composite** đã được định rõ. Một khả năng gói được xem như là việc có thể so khớp một cách rõ ràng cụ thể hơn các lược đồ gói đã được tạo ra, cũng như việc phù hợp đúng với một lược đồ tổng quát.

Việc kiểm tra chắc chắn hơn nữa là rất cần thiết cho các lựa chọn gói phức tạp gắn liền với **syncReplyMode**. Điều này được thảo luận như sau.

syncReplyMode Packaging (đóng gói) là quan trọng trong việc định rõ các khả năng và chi tiết thực hiện. Nhưng vì mức độ của ký hiệu kinh doanh có thể được đòi hỏi, cho nên các liên kết hành động khác cần thiết phải được kiểm tra cộng với các liên kết đã được lựa chọn cho hành động yêu cầu và trả lời. Ngoài ra, các giá

syncReplyMode chính là giá trị để chỉ ra các phần nào của một thông điệp phải được trở lại trong mục Reply (trả lời) của truyền tải thao tác đồng bộ, như HTTP. (ở đây, chúng ta sử dụng thuật ngữ “đồng bộ” có nghĩa là “giống như kết nối TCP”, đó là một cách sử dụng thuật ngữ này). Chúng ta không làm việc định rõ bất cứ việc chờ đợi, thông báo hoặc cách hành động ngăn chặn nào trong các quá trình hoặc các chuỗi đã được đòi hỏi, tuy vậy có thể đoán chừng việc này là một số hành động sử dụng máy điện toán để duy trì trạng thái kết nối và quan trọng hơn TCP và các lớp ổ cắm.

Các quá trình thực hiện hợp lý gắn liền với các giá trị khác nhau của các **syncReplyMode** là số đồng, nhưng có thể thử để ít nhất cũng chỉ ra được các nhân tố chính.

Cũng được xem xét như trên, phần tử trị của **TransportReceive/Endpoint/@type** cần thiết phải được kiểm tra khi cung cấp các bản phác thảo CPA.

- đầu tiên, chúng ta hãy bắt đầu với các trường hợp sau: các hành động đáp lại yêu cầu, các ký hiệu bộ quản lý dịch vụ thông điệp và ký hiệu kinh doanh ngược trở lại trong một vài phối hợp giữa việc hỏi âm đồng bộ với các thông điệp đồng bộ khác. Những việc phối hợp khác nhau này sẽ được thảo luận đối với các phần tử của **syncReplyMode**: “mshSignalsOnly”, “signalsOnly”, “responseOnly” và “signalsAndResponse”;

- theo như quy ước, việc trình bày sự hồi âm đồng bộ phụ thuộc vào các Phần tử **CanSend** (có thể gửi) hoặc **CanReceive** theo các phần tử **CanReceive** (có thể nhận) hoặc **CanSend** (có thể gửi) đã trình bày các khả năng liên kết yêu cầu ban đầu.

Để trình bày đồng bộ các yêu cầu, việc hồi âm hoặc các ký hiệu, thì các Phần tử **CanSend** (có thể gửi) hoặc **CanReceive** là cùng loại và phụ thuộc trực tiếp vào **ServiceBinding**. Ví thế, cả hai khả năng đồng bộ và không đồng bộ có thể được tập hợp lại dưới một **ServiceBinding** trong một CPP, nhưng có thể vẫn phân biệt được một cách rõ ràng. Nói chung, việc tăng dần sự phụ thuộc có thể chỉ ra các mẫu thoại phức tạp hơn hành động yêu cầu và đáp lại yêu cầu. Tuy nhiên, không có nhiều trường hợp sử dụng phổ biến chức năng nói trên trong thời gian viết văn bản này.

mshSignalsOnly

Cả hai **DeliveryChannel** (kênh truyền) của người gửi yêu cầu (được tham chiếu bởi **CanSend/ThisPartyActionBinding/ChannelId**) và **DeliveryChannel** (kênh truyền) của người nhận yêu cầu (được tham chiếu bởi **CanReceive/ThisPartyActionBinding/ChannelId**) đều phải có giá trị **MessagingCharacteristics/@syncReplyMode** là **mshSignalsOnly**.

Khi một bên có thể nhận ra rõ ràng một **DeliveryChannel** cho đường bao SOAP với sự phụ thuộc vào các Phần tử **CanSend** (có thể gửi) hoặc **CanReceive** và các liên kết thích ứng với chúng, điều trên được bỏ qua đối với phần mềm thông điệp ebXML. Điều đó được coi như là mỗi bên có thể xử lý việc hồi âm đồng bộ được xây dựng phù hợp với thông điệp ebXML. Ở đây, kỹ thuật trình bày **DeliveryChannel** cung cấp một trình giữ chỗ cho việc bắt giữ lại các giao thức ký hiệu thông điệp khác.

Hiện nay, kể cả trong đường bao SOAP, lời báo nhận và lời báo nhận đã được ký hiệu cùng với các lỗi là các tín hiệu MSH chính. Nếu công ty A thiết lập **syncReplyMode** là **mshSignalsOnly**, tiếp theo **CanReceive/ThisPartyActionBinding/@packageld** tương quan của công ty B phải gồm có một ổ **CanSend/ThisPartyActionBinding/@packageld** cho một thông điệp mà không cần bất cứ payload hay tín hiệu kinh doanh nào. Thêm vào đó, **CanSend/ThisPartyActionBinding/@packageld** của hành động đáp lại yêu cầu của công ty B phải giải quyết dạng mẫu gói (và có thể các thành phần khác) mà hành động trả lời ngược lại là không đồng bộ. Tính tương thích của các phần tử **DeliveryChannel** (kênh truyền) có thể được kiểm tra, như việc kiểm tra khả năng của công ty A có thể nhận vùng mang thông tin trả lời, vùng mang thông tin ký hiệu hoặc các bó hành động đáp lại yêu cầu với các ký hiệu có được theo như việc định rõ bằng các dạng mẫu gói được tham chiếu bởi các giá trị thuộc tính **packageld** (id của gói) của các phần tử **ThisPartyActionBindings** có liên quan hay không.

SignalsOnly

Cả hai **DeliveryChannel** của người gửi yêu cầu (được tham chiếu bằng **CanSend/ThisPartyActionBinding/ChannelId** của nó) và **DeliveryChannel** của người nhận yêu cầu (được tham chiếu bằng **CanReceive/ThisPartyActionBinding/ChannelId**) đều phải có giá trị **MessagingCharacteristics/@syncReplyMode** là **SignalsOnly**.

Nếu Công ty A thiết lập **syncReplyMode** cho đến *signalsOnly*, tiếp theo, thuộc phần tử **CanReceive** tương quan của công ty B, phải thiết lập được một ở **CanSend/ThisPartyActionBinding** của giá trị thuộc các thuộc tính **packages** để giải quyết một dạng mẫu gói thích hợp cho các ký hiệu.

Để **CanSend/ThisPartyActionBinding/@packageld** kết hợp được với hành động đáp lại yêu cầu mức độ kinh doanh của công ty B, giá trị **IDREF** của thuộc tính phải giải quyết một dạng mẫu gói có khả năng trở lại payloads và bỏ qua các ký hiệu kinh doanh. Phần tử **CanSend** (có thể gửi) sẽ là một phần tử con trực tiếp của **ServiceBinding**, là một sự sắp đặt việc trình bày các ký tự không đồng bộ của nó. Bên đưa ra yêu cầu khởi tạo sẽ cần thiết phải có một **CanReceive/ThisPartyActionBinding** tương thích với bên trả lời và là phần tử con trực tiếp của phần tử **ServiceBinding**.

Việc sử dụng các phần tử phụ **CanSend** và **CanReceive** có thể có ích nếu các chi tiết **DeliveryChannel** cho các ký hiệu ngoại lệ khác với việc định rõ yêu cầu và đáp lại yêu cầu. Ví dụ, các liên kết ký hiệu có thể khác bởi việc bỏ qua **ackRequested** hoặc có thể là một trong các tính năng bảo mật (đường bao số hoặc xác nhận việc không từ chối) được sử dụng cho hành động yêu cầu và đáp lại yêu cầu. Đúng như với các cách kiểm tra khác trong hành động yêu cầu và đáp lại yêu cầu, có thể kiểm tra sự tương thích trong **Packaging**, **DocExchange**, **MessagingCharacteristics** hoặc **BusinessTransactionCharacteristics** khác với trong việc phụ thuộc tương quan vào **CanSend** và **CanReceiveDeliveryChannels**.

responseOnly

Cả hai **DeliveryChannel** của người gửi yêu cầu (được tham chiếu bằng **CanSend/ThisPartyActionBinding/Channelld**) và **DeliveryChannel** của người nhận yêu cầu (được tham chiếu bằng **CanReceive/ThisPartyActionBinding/Channelld**) đều phải có giá trị **MessagingCharacteristics/@syncReplyMode** của **responseOnly**.

Nếu công ty A thiết lập **syncReplyMode** là *responseOnly*, **CanSend/ThisPartyActionBinding/@packageld** cho việc đáp lại yêu cầu của công ty B phải giải quyết một dạng mẫu gói có khả năng trở lại các vùng mang thông tin, nhưng bỏ qua các ký hiệu kinh doanh. Phần tử **CanSend/ThisPartyActionBinding** sẽ bao gồm một phần tử con của phần tử **CanReceive**, do đó, người đáp lại yêu cầu có thể chỉ ra rằng đó là một hành động đáp lại một cách đồng bộ.

Ở đó phải là một phương pháp độc lập để trở lại các ký hiệu lỗi về mức độ kinh doanh. Vì vậy, ở đó phải là một **ThisPartyActionBinding** cho bất cứ việc payload ký hiệu đã được thông báo nào và các việc liên kết này phải trong mức độ của phần tử con trực tiếp của **ServiceBinding** để trình bày sự không đồng bộ của chúng.

Nó là không có đủ khả năng để cho **ReceiptAcknowledgment** và các ký hiệu tương tự sẽ được sử dụng khi mà một hành động đáp lại yêu cầu được trở lại một cách đồng bộ. Động cơ thúc đẩy để sử dụng các ký hiệu nói trên là để chỉ ra một cách rõ ràng các tiến trình tiếp theo và động cơ này sẽ trở nên yếu dần khi một hành động đáp lại yêu cầu được trở lại một cách trực tiếp.

Đối với *responseOnly*, bao gồm việc phụ thuộc **CanSend/ThisPartyActionBinding** và **CanReceive/ThisPartyActionBinding**, có nghĩa là có thể kiểm tra sự tương thích trong **Packaging**, **DocExchange**, **MessagingCharacteristics** hoặc **BusinessTransactionCharacteristics**.

Ở đây, các thuộc tính **syncReplyMode** (phương thức trả lời đồng bộ) và **ackRequested** phải được tính toán đến một cách cẩn thận, bởi vì, ở đây, một giá trị *mshSignalsOnly* được hiểu là vòng quay tương tự của thông điệp đồng bộ sẽ cần thiết để duy trì cho việc kết nối tương tự. Nhân đây, khi các phần tử **Transport** (truyền tải) được tham chiếu theo các liên kết phụ thuộc, thì ở đó không cần thiết bất cứ phần tử **Endpoint** (điểm cuối) nào. Nếu ở đó có các phần tử **Endpoint** (điểm cuối) thì có thể lờ chúng đi. **signalsAndResponse**

Cả hai **DeliveryChannel** của người gửi yêu cầu (được tham chiếu bằng **CanSend/ThisPartyActionBinding/ChannelId**) và **DeliveryChannel** của người nhận yêu cầu (được tham chiếu bằng **CanReceive/ThisPartyActionBinding/ChannelId**) đều phải có giá trị **MessagingCharacteristics/@syncReplyMode** của **signalsAndResponse**.

Nếu công ty A thiết lập **syncReplyMode** là **signalsAndResponse**, **CanSend/ThisPartyActionBinding** cho việc đáp lại yêu cầu của công ty B phải phụ thuộc vào phần tử **CanReceive** của công ty này. Dạng mẫu gói đã được tham chiếu phải có khả năng trả lại payloads và các ký hiệu được bó lại với nhau. Nếu không có các liên kết không đồng bộ tồn tại trong các ký hiệu lỗi, thì chỉ có **DeliveryChannel** được xác định là đồng ý với tất cả các khía cạnh của việc trao đổi thông điệp trong công việc kinh doanh. Tuy nhiên, thông thường việc này là có khả năng cung cấp một liên kết không đồng bộ để gửi các ký hiệu ngoại lệ.

AckRequested and ackSignatureRequested

Việc kiểm tra các thuộc tính **ackRequested** (báo nhận được yêu cầu) và **ackSignatureRequested** trong phạm vi các **DeliveryChannel** tương quan (là tương quan bởi vì đã được tham chiếu theo các Phần tử **CanSend** (có thể gửi) và **CanReceive** của một hành động) là chính để nhận ra rằng các giá trị của các thuộc tính tương ứng là giống nhau.

Tuy nhiên, ở đó là một vài tương tác của các thuộc tính nói trên với các mục thông tin khác mà cần thiết để được tính đến.

Việc sử dụng chủ yếu thuộc tính **ackRequested** (báo nhận được yêu cầu) là trong phạm vi các cấu hình thông điệp xác thực. Nếu các thông điệp xác thực là được định cấu hình, thì tiếp theo việc kiểm tra thỏa thuận trong các phần tử **ReliableChannel** tương quan như việc tìm ra theo **DocExchange/ebXMLSenderBinding** và **DocExchange/ebXMLReceiveBinding** là hợp lệ. Cũng vậy, giá trị của thuộc tính **DuplicateElimination** (loại trừ sao chép lại) của **MessagingCharacteristics** cũng phải được kiểm tra cho việc thỏa thuận. Các bản phác thảo CPA có thể được tạo thành bởi các giá trị chuẩn trực có tính toán mà các giá trị này là không cân bằng cùng với một vài các kích thước. Việc đánh giá thấp có thể đưa ra các CPA mà hầu hết là có khả năng để đạt được việc chấp nhận; vì vậy, ví dụ

nếu **duplicateElimication** là không đúng trong phía bên nhận, việc chuẩn trực không đúng trong phía bên gửi thì hầu hết là vấn đề khả năng cung cấp được một bản phác thảo thành công.

Chức năng thêm vào của **ackSignatureRequested** là cung cấp một việc thực hiện không đầy đủ cho dịch vụ không từ chối việc nhận. Việc kiểm tra cơ bản là như nhau đối với các giá trị thuộc tính, nhưng ngoài ra có thể phải kiểm tra bắt buộc đối với cách kiểm tra và chuẩn trực. Nếu không có ký hiệu về khả năng thực hiện không từ chối việc nhận là được tìm ra theo **ServiceBinding**, tiếp theo có một giá trị thường xuyên cho **ackSignatureRequested** đề nghị việc trực chuẩn các thuộc tính **BusinessTransactionCharacteristics**, thì **isNonRepudiationReceiptRequired** là đúng. Tuy nhiên, nếu việc đó được thực hiện, thì phải cẩn thận thi hành việc kiểm tra rằng **isIntelligibleCheckRequired** của thuộc tính **BusinessTransactionCharacteristics** là không đúng. Đó là lý do vì sao việc thực hiện thông điệp chỉ có thể được thỏa thuận với việc xác nhận theo hướng có thể nhận được một dòng bit (và tiếp tục có thể dùng được cho quá trình xa hơn nữa). Điều đó là không an toàn cho bất cứ việc kiểm tra ngữ nghĩa hoặc cú pháp trong dữ liệu đã được thực hiện nào.

E.5.2.3 Kiểm tra DocExchange (Trao đổi tài liệu) đối với **BusinessTransactionCharacteristics** (Các đặc điểm giao dịch kinh doanh)

Khi mà hầu hết việc sử dụng các CPP và CPA với thông điệp ebXML là có khả năng sớm được triển khai, thì sẽ tồn tại một cơ hội để kiểm tra việc thỏa thuận dựa trên các thuộc tính **BusinessTransactionCharacteristics**:

Dưới đây là ba thuộc tính cần thiết để có được các giá trị ngang nhau trong các liên kết của hoặc một yêu cầu hoặc một việc đáp lại yêu cầu. Loại trừ việc một công cụ để sinh ra CPA đã được đưa ra một cách tinh xảo để có thể kiểm soát sự cố kết các giá trị được chọn dùng ở đây với các giá trị cho các tham số của thông điệp xác thực, việc không cần thảo luận xa hơn nữa có thể sẽ cung cấp được trong phụ lục này việc tồn tại các liên kết **ReceiptAcknowledgment** hoặc **AcceptanceAcknowledgment** thích hợp với cấu hình bên trong **syncReplyMode** dựa trên các "deadline-hạn chót".

```
<attribute name="timeToAcknowledgeReceipt" type="duration"/>
<attribute name="timeToAcknowledgeAcceptance" type="duration"/>
<attribute name="timeToPerform" type="duration"/>
```

Việc giữ nguyên các thuộc tính bao hàm việc đưa ra một con số bảo mật và là trọng điểm cho việc giữ nguyên thảo luận về các thuộc tính của **BusinessTransactionCharacteristics**:

```
<attribute name="isNonRepudiationRequired" type="boolean"/>
<attribute name="isNonRepudiationReceiptRequired" type="boolean"/>
<attribute name="isIntelligibleCheckRequired" type="boolean"/>
<attribute name="isAuthenticated" type="tns:persistenceLevel.type"/>
<attribute name="isTamperProof" type="tns:persistenceLevel.type"/>
<attribute name="isAuthorizationRequired" type="boolean"/>
<attribute name="isConfidential" type="tns:persistenceLevel.type"/>
```

Ở đây, việc kiểm tra cơ bản là kiểm tra các **DeliveryChannel** tương quan, kiểm tra các thuộc tính tương ứng có các giá trị tương đương. Mặt khác, việc kiểm tra một vài sự tương tác giữa các khía cạnh với các phần của **DeliveryChannel** sẽ thúc đẩy việc thực hiện một vài việc kiểm tra thêm nữa.

Trước đây, khi thảo luận phần tử **ackSignatureRequested** của thuộc tính **MessagingCharacteristics**, cần phải lưu ý rằng việc thực hiện đầy đủ thông điệp sẽ không hỗ trợ được nhiều cho việc giữ **isNonRepudiationReceiptRequired** đúng và việc cung cấp được thuộc tính **isIntelligibleCheckRequired** (yêu cầu kiểm tra khả năng hiểu được) là sai.

Khi cả hai là đúng, thì tiếp theo phải tồn tại một ký hiệu kinh doanh với các giá trị **Packaging** và **DeliveryChannel** tương thích. Nếu ký hiệu được mô tả một cách độc lập trong phạm vi các Phần tử **CanSend** (có thể gửi) và **CanReceive** không đồng bộ, thì việc biết được tên ký hiệu (như **ReceiptAcknowledgment**) có thể hỗ trợ một cách tương đối đơn giản cho hoạt động tìm kiếm và kiểm tra. Tuy nhiên, nếu truyền tải đồng bộ là phức tạp, thì một số quá trình lọc việc sử dụng **syncReplyModes** có thể cần thiết để tìm ra một sự hỗ trợ cơ bản cho việc thực hiện đầy đủ dịch vụ không từ chối việc nhận.

Khi mà việc xác nhận không từ chối được thực hiện đầy đủ bằng ký hiệu kinh doanh, tiếp theo việc kiểm tra dựa trên hiệu lực của sự chứng nhận ký hiệu có thể bao hàm **CollaborationRole/ApplicationCertificateRef** và **CollaborationRole/ApplicationSecurityDetailsRef**, việc này cung cấp được một tham chiếu để phần tử **SecurityDetails** (chi tiết an ninh) chứa được một danh sách **TrustAnchors** (mấu neo chứng thực). Sự chứng nhận từ bên phát ký hiệu **ReceiptAcknowledgment** sẽ được kiểm tra ngược lại với các sự chứng nhận có liên quan bằng **AnchorCertificate** dưới **TrustAnchors**.

Đôi khi, ký hiệu kinh doanh sẽ được truyền như một phần của thông điệp. Điều này vẫn đúng vì chính bản thân thông điệp có thể vẫn được gửi đi thông qua một MSH và MSH cũng có thể ra ký hiệu cho thông điệp sử dụng sự chứng nhận mà sự chứng nhận này được tìm ra bằng cách giải quyết **IDREF** mà được tìm ra tại **DocExchange/ebXMLSenderBinding/SenderNonRepudiation/SigningCertificateRef@certId**.

Nếu thành phần của một phần mềm cá biệt có thể thực hiện đầy đủ cả hai chức năng MSH và chức năng bảo mật mức độ kinh doanh, thì có thể việc chứng nhận tương tự nhau sẽ được chỉ rõ bởi **ApplicationCertificateRef** và **SigningCertificateRef/@certId**. Nói cách khác, sự khác biệt giữa ký hiệu mức độ MSH và ký hiệu mức độ ứng dụng là một logic và không thể tương ứng với các ranh giới giữa các thành phần của phần mềm. Bởi vì, chữ ký của MSH là hơn thông điệp, chữ ký của thông điệp có thể hơn chữ ký của mức độ ứng dụng. Mặc dù, điều này có thể không cần thiết cho một vài cấu hình hệ thống, nhưng các giao thức phải cần đến cả hai chữ ký này để tồn tại trên các miền khác nhau.

Việc thiếu khả năng để làm cho sự chứng nhận có hiệu lực có thể không ngăn cản việc hình thành một bản phác thảo CPA. Đầu tiên, việc chứng nhận ký hiệu của người gửi có thể là một chứng nhận tự ký hiệu. Vì vậy, cần một tham chiếu cho chứng nhận tự ký hiệu để có thể được thêm vào danh sách **TrustAnchors/AnchorCertificateRef** của người nhận. Việc đề nghị này có nghĩa là đề nghị để được đồng ý một mô hình uỷ thác trực tiếp, hơn là một mô hình có thứ bậc bao gồm các quyền chứng nhận cụ thể. Thứ hai, ngoài ra có thể thực hiện một đề nghị để thêm vào một gốc uỷ thác bằng việc xem lại một cách thích hợp các **TrustAnchors**.

TCVN ISO/TS 15000-1 : 2007

Khi mà việc xác nhận không từ chối được thực hiện đầy đủ bằng lớp thông điệp, thì việc kiểm tra dựa trên PKI được thực hiện bằng việc sử dụng các phần tử dưới **DocExchange**.

isNonRepudiationRequired

isAuthenticated

isAuthorizationRequired

isTamperProof

Các giải thuật về sự xác nhận là đúng, sự cho phép, việc không từ chối và bản chất việc bảo đảm chống lưc lợi có thể là rất khác biệt nhau như các khái niệm về mức độ kinh doanh, song việc thực hiện các nhân tố nói trên hướng về việc sử dụng các kỹ thuật, công nghệ tương tự. Trên thực tế, việc ngăn ngừa sự lưc lợi là không được thực hiện đúng. Để thay thế, là phải cung cấp được các điều kiện để phát hiện ra sự lưc lợi (hoặc một vài sự cất xén ngẫu nhiên) đã xảy ra. Tương tự như vậy, thông thường việc thực hiện cho phép là được cung cấp bởi việc thực hiện điều khiển truy cập (việc cho phép hoặc ngăn chặn người sử dụng trong vai trò sử dụng tài nguyên) và bản trình bày của một mã thông báo hoặc giấy uỷ nhiệm để có thể được truy cập, việc cho phép có thể bao hàm việc xác nhận là đúng như bước đầu! Việc không từ chối có thể dựa vào tất cả các chức năng liên trước vào thông tin tiếp tục dùng cho việc cung cấp các chứng cứ có cơ sở về nguồn gốc trong thời gian gần đây.

Khi mà việc kiểm tra ***isNonRepudiationRequired*** có thể được thiết lập đúng cho cả hai bên hay không, thì việc kiểm tra chứng nhận ký hiệu có đúng hay không sẽ có hiệu lực đối với người nhận.

Tham chiếu IDREF cho việc chứng nhận ký hiệu được tìm ra trong ***DocExchange/ebXMLSenderBinding/SenderNonRepudiation/SigningCertificateRef/@certId***. Các chứng nhận đã được tham chiếu phải kiểm tra được tính hiệu lực đối với các neo uỷ thác thu được từ các phần tử ***TrustAnchors/AnchorCertificate*** theo phần tử ***SecurityDetails*** (chi tiết an ninh) được tham chiếu bởi IDREF tại ***DocExchange/ebXMLReceiverBinding/ReceiverNonRepudiation/SigningSecurityDetailsRef/@securityId*** như lưu ý ở trên, việc thiếu khả năng để làm cho việc chứng nhận có hiệu lực là không ngăn cản việc xây dựng một bản phác thảo CPA. Hoặc là tự chứng nhận ký hiệu hoặc là các neo uỷ thác mới có thể được thêm vào để sắp cho thành hàng các mô hình uỷ thác thuộc một bên với việc chứng nhận của phía bên kia.

Để kiểm tra thêm các thao tác giữa các phần phụ thuộc của PKI, thì có thể thực hiện việc kiểm tra dựa trên tính tương thích của các giá trị trong các thuộc tính khác trong ***DocExchange/ebXMLReceiverBinding/ReceiverNonRepudiation*** và trong ***DocExchange/ebXMLSenderBinding/SenderNonRepudiation***. Các giá trị ***NonRepudiationProtocol***, ***HashFunction*** và ***SignatureAlgorithm*** có thể tương thích thậm chí ngay cả khi không ngang bằng nhau nếu kiến thức của các yêu cầu giao thức cho phép rút đến giá trị thực hiện bắt buộc. Vì vậy, ở đây các giá trị có thể được tìm ra ngang nhau, được sắp thành hàng hoặc được dàn xếp để đạt được một sự thỏa thuận.

Nếu **isNonRepudiationRequired** là đúng, thì **isAuthenticated** và **isTamperProof** cũng phải là đúng.

Bởi vì, ở đó trong việc thực hiện **isNonRepudiationRequired** bằng các điều kiện của chữ ký dạng số, thì cả hai việc xác nhận là đúng (đối với việc kết hợp đồng nhất với chứng nhận ký hiệu) và việc phát hiện ra sự lục lọi (đối với mớ lộn xộn mật mã của chữ ký) đều cũng sẽ được thực hiện. Các điều ngược lại không cần thiết phải đúng bởi vì việc xác nhận là đúng và việc phát hiện ra sự lục lọi có thể được hoàn thành mà không cần thông tin lưu trữ cần thiết để hỗ trợ sự khẳng định là không từ chối.

isConfidential

Thuộc tính **isConfidential** (bảo mật) chỉ ra các đặc tính được sắp xếp khác nhau giữa các mức độ của các ngăn xếp ứng dụng đến ứng dụng gửi/nhận.

isConfidential có các giá trị “none-không có”, “transient-tạm thời”, “persistent-liên tục” và “transient-and-persistent-tạm thời và liên tục” có thể chấp nhận được. Các giá trị liên tục hoặc tạm thời - và - liên tục chỉ ra rằng một vài chức năng phong bì là hiện hữu; một giá trị tạm thời chỉ ra sự cần mật là được áp dụng ở lớp hoặc dưới lớp chuyển giao.

Thông điệp ebXML phiên bản 2.0 không có một sự thực hiện đầy đủ chính thức cho các đường bao số và có liên quan tới văn bản kỹ thuật của sự mật hoá XML tương lai như phương hướng được mong đợi cho chức năng này. Tuy nhiên, hiện nay văn bản kỹ thuật của sự mật hoá XML chính là thư giới thiệu người cộng tác và thích hợp đối với việc thực hiện sơ bộ.

Trong phạm vi CPA, **DocExchange/ebXMLSenderBinding/SenderDigitalEnvelope** và **DocExchange/ebXMLReceiverBinding/ReceiverDigitalEnvelope** có thể cung cấp các chi tiết cấu hình gắn liền với việc bảo mật phù hợp với [XMLENC]. Việc sử dụng sự mật hoá XML thông thường cũng có thể lộ ra trong giá trị của **DigitalEnvelopeProtocol** và có thể cũng lộ ra trong phạm vi một phần tử **NamespaceSupported** (tên miền được hỗ trợ) bên trong **Packaging**.

Hiện nay, [ebMS] chỉ ra được duy nhất một phương hướng cuối cùng để sử dụng sự mật hoá XML, nhưng không uỷ thác được bất cứ giao thức đường bao số nào. đường bao số có thể được làm ở “application level-mức độ ứng dụng” và có thể lộ ra dưới các kiểu MIME trong phạm vi phần tử **Packaging** (đóng gói). Việc phù hợp PKI có thể sử dụng các chứng nhận được cung cấp trong **ApplicationCertificateRef** và **ApplicationSecurityDetailsRef**. Nếu các giao thức khác là sẽ được sử dụng thì nó sẽ là an toàn nhất để mở rộng nội dung theo **DocExchange**, thí dụ, **XXXSenderBinding** và **XXXReceiverBinding** và theo khuôn mẫu nội dung ebXML làm theo **DocExchange**. Các phiên bản tương lai của văn bản kỹ thuật này dự kiến sẽ làm cho việc mở rộng ở trên dễ hơn về mặt ngữ nghĩa để sử dụng thao tác giữa các phần; hiện nay, việc mở rộng này phải là một sự mở rộng nhiều phía trong phạm vi một vài cộng đồng thương mại.

Việc có thể kiểm tra được **isConfidential** hay không có thể là thiết lập được việc “liên tục” hoặc “tạm thời - và - liên tục” cho cả hai bên, việc có kiểm tra được sự chứng nhận chuyển đổi khoá hay không sẽ được tính đến như hiệu lực về phía người gửi. Tham chiếu IDREF cho phần tử **SecurityDetails** (chi tiết an ninh) là được tìm ra trong

DocExchange/ebXMLSenderBinding/SenderDigitalEnvelope/EncryptionSecurityDetailsRef/@securityId.

Các chứng nhận neo uỷ thác thu được từ các phần tử ***TrustAnchors/AnchorCertificateRef*** thuộc phần tử ***SecurityDetails*** (chi tiết an ninh) sẽ được sử dụng để kiểm tra sự chứng nhận của bên người gửi được tham chiếu bởi ***DocExchange/ebXMLReceiverBinding/ReceiverDigitalEnvelope/EncryptionCertificateRef/@certId*** có hiệu lực hay không.

Như lưu ý trước đây, việc thiếu khả năng để làm cho việc chứng nhận có hiệu lực là không ngăn cản việc xây dựng một bản phác thảo CPA. Hoặc là tự chứng nhận ký hiệu hoặc là các neo uỷ thác mới có thể được thêm vào để sắp cho thành hàng các mô hình uỷ thác thuộc một bên với việc chứng nhận của phía bên kia.

Để thêm vào việc kiểm tra có liên quan đến CPI và việc sắp thành hàng, thì các phần tử ***EncryptionAlgorithm*** (thuật toán mã hóa) và ***DigitalEnvelopeProtocol*** phải được kiểm tra về tính bình đẳng (hoặc tính tương thích) và nếu không tương thích hoặc bằng nhau thì việc sắp thành hàng các giá trị sẽ có hiệu lực đối với phiên bản ban đầu của một CPA đã được đề nghị. Việc ưu tiên và việc sắp thành hàng các phần tử nói trên có thể được hoàn thành trong một giai đoạn thỏa thuận sau.

Cuối cùng, có thể rằng đường bao số của một bên sẽ được làm mô hình để sử dụng ***DocExchange/ebXMLSenderBinding/SenderDigitalEnvelope*** hay ***DocExchange/ebXMLReceiverBinding/ReceiverDigitalEnvelope***, trong khi phía bên kia sử dụng duy nhất ***Packaging*** để chỉ ra việc sử dụng, ví dụ, các đường bao số S/MIME, bởi vì nó đã nhận một payload phong bì từ một ứng dụng. Như là trong trường hợp, việc kiểm tra hiệu lực của sự chứng nhận có thể phải phụ thuộc vào việc kiểm tra sự chứng nhận được mô tả bằng ***DocExchange/ebXMLReceiverBinding/ReceiverDigitalEnvelope/EncryptionCertificateRef/@certId*** có hiệu lực ngược lại với ***TrustAnchors*** đã được tìm ra bằng cách giải quyết ***CollaborationRole/ApplicationSecurityDetailsRef*** không. Sự phức tạp này được phát sinh có khả năng là do chức năng đường bao số có thể đã được trải ra khắp các phần khác biệt của nhiệm vụ trong việc cài đặt phần mềm khác nhau.

E.6 Tạo CPA: Các chi tiết kỹ thuật

Khi việc lắp ráp một bản phác thảo CPA từ việc phù hợp các phần của hai phần tử ***PartyInfo*** (thông tin bên tham gia) thuộc CPP, thì một vài ràng buộc thêm vào cần thiết phải được tuân theo.

Đầu tiên, như đã được đề cập trong mục 9.11.1, phần mềm cho việc đưa ra bản phác thảo CPA là cần thiết để bảo đảm rằng các giá trị ID trong một CPP là khác biệt trong CPP khác để không có tham chiếu IDREF nào xung đột khi các CPP được kết hợp. Các giá trị ID dưới đây là các đối tượng tiềm năng dẫn đến xung đột:

Certificates

SecurityDetails

SimplePart

*Packaging**DocExchange**Transport**DeliveryChannel**ThisPartyActionBinding*

Đó là các phần tử và việc xác định các loại phức hợp bao gồm các IDREF. Ngoài ra, một vài phần tử có các thuộc tính cùng với các giá trị IDREF. Đó là:

*PartyInfo**ActionBinding.type**ThisPartyActionBinding**OtherPartyActionBinding**OverrideMSHActionBinding**ChannelId**DeliveryChannel**Constituent**CertificateRef.type**AnchorCertificateRef**ApplicationCertificateRef**ClientCertificateRef**ServerCertificateRef**SigningCertificateRef**EncryptionCertificateRef**CertificateRef**SecurityDetailsRef.type*

Thứ hai, khi mà thông tin liên kết **CanSend** và *CanReceiver* được tìm ra để so khớp (ngang bằng nhau, phù hợp với hoặc tương thích với) thông tin liên kết theo các Phần tử **CanSend** (có thể gửi) và **CanReceiver** của phía bên kia, thì các tham chiếu IDREF cho **OtherPartyActionBinding** là được điền vào trong CPA.

Thứ ba, đối với các CPA đã được ký hiệu, người thực hiện được báo cho biết để xem lại mục 9.9.1.1 khi sử dụng [XMLDSIG] cho phương pháp kỹ thuật ký. Một CPA được đề nghị không cần thiết phải có một chữ ký.

Thứ tư, khi mà một CPA được soạn từ hai CPP, xem mục 8.8 trong trường hợp nó được nói rõ rằng các phần tử **Comment** (chú giải) từ cả hai CPP phải được tính đến trong CPA trừ khi đã chấp nhận cách khác.

TCVN ISO/TS 15000-1 : 2007

Thứ năm, các việc kiểm tra khác nhau về hiệu lực của CPA có thể được kiểm soát dựa trên bản phác thảo CPA, nhưng các việc kiểm tra này là then chốt hơn đối với việc một CPA được thỏa thuận là sẽ được triển khai và nhập vào trong các thành phần của phần mềm dùng trong thời gian chạy.

1. kết thúc: Các chứng nhận được sử dụng trong việc ký hiệu một CPA có thể được kiểm tra để xác nhận rằng chúng chưa hết hiệu lực trước khi CPA hết hiệu lực, như việc đưa ra phần tử *End*;
2. kết thúc sự chứng nhận: nếu sự tồn tại của một CPA vượt quá sự tồn tại của sự chứng nhận được chấp nhận để sử dụng trong việc ký hiệu, chuyển đổi khoá hoặc các chức năng bảo mật khác, thì sau đó nó sẽ thích hợp để thực hiện **ds:KeyInfo** có liên quan đến các chứng nhận, hơn là để tính đến chúng trong phạm vi phần tử bằng trị số;
3. các tham chiếu quy định quá trình có thể được kiểm tra phù hợp với các điều khoản của mục 8.4.4 và các tiểu khu của nó.

Cuối cùng, một CPA có các phần tử khác nhau, các giá trị của các phần tử này là không đặc trưng cho việc bắt nguồn từ một trong các CPP (và có thể cần thiết kiểm tra khi sử dụng một khuôn mẫu CPA như là nền tảng cho một bản phác thảo CPA). Phần tử **Status**, **Start**, **End** và có thể là **ConversationConstraints** cần thiết được thêm vào các thuộc tính:

CollaborationProtocolAgreement/@cpaid,

CollaborationProtocolAgreement/@version,

CollaborationProtocolAgreement/Status@value,

CollaborationProtocolAgreement/ConversationConstrain@invocationLimit và

CollaborationProtocolAgreement/ConversationConstraint@concurrentConversations

có thể cũng cung cấp các giá trị khi cần thiết.

Phụ lục F

(quy chuẩn)

Quan hệ tương đương giữa CPA và các tham số truyền thông điệp ebXML

Bảng sau đây chỉ ra quan hệ tương ứng giữa các phần tử được sử dụng trong tiêu đề thông điệp dịch vụ truyền thông điệp ebXML và các bản sao của chúng trong CPA đó.

Thuộc tính/Phần tử Tiêu đề thông điệp	Thuộc tính/Phần tử CPA tương ứng
Phần tử PartyId (ID bên tham gia)	Phần tử PartyId (ID bên tham gia); nếu nhiều phần tử PartyId (ID bên tham gia) xuất hiện thuộc cùng phần tử PartyInfo (thông tin bên tham gia) trong CPA đó, tất cả trong các CPA đó BẮT BUỘC được chứa trong tiêu đề thông điệp đó
Phần tử Role (vai trò)	Phần tử Role (vai trò)
Phần tử CPAId (id CPA)	Thuộc tính cpaid (id của CPA) trong phần tử CollaborationProtocolAgreement (thỏa thuận giao thức hợp tác)
Phần tử ConversationID (id hội thoại)	Không tương đương; Nên được tạo bởi phần mềm trên giao diện dịch vụ thông điệp (MSI)
Phần tử Service (dịch vụ)	Phần tử Service (dịch vụ)
Phần tử Action (hành động)	Thuộc tính action (hành động) trong phần tử ThisPartyActionBinding (quy định hoạt động bên tham gia này)
Phần tử TimeToLive (thời gian tồn tại)	được tính toán như tổng của Timestamp (trong tiêu đề thông điệp) + PersistDuration (thuộc DocExchange/ebXMLReceiverBinding)
Phần tử MessageId (id thông điệp)	Không tương đương; được tạo ra bởi mỗi thông điệp MSH
Phần tử Timestamp (tem thời gian)	Không tương đương; được tạo ra bởi mỗi thông điệp MSH
Phần tử RefToMessageId (tham chiếu tới id thông điệp)	Không tương đương; thường được chuyển qua bởi ứng dụng khi có thể áp dụng; Nên được sử dụng để tương quan các thông điệp phản hồi với các thông điệp yêu cầu
Phần tử SyncReply (trả lời đồng bộ)	Thuộc tính syncReplyMode (phương thức trả lời đồng bộ) trong phần tử MessagingCharacteristics ; bao gồm phần tử SyncReply (trả lời đồng bộ) nếu và chỉ nếu thuộc tính syncReplyMode (phương thức trả lời đồng bộ) khác "none"
Phần tử DuplicateElimination (loại trừ sao chép lại)	Thuộc tính DuplicateElimination (loại trừ sao chép lại) trong phần tử MessagingCharacteristics ; bao gồm phần tử DuplicateElimination (loại trừ sao chép lại) nếu thuộc tính DuplicateElimination (loại trừ sao chép lại) thuộc MessagingCharacteristics được thiết lập là "always" hoặc nếu nó được thiết lập là "perMessage" và ứng dụng chỉ tới MSH để loại trừ trùng lặp được yêu cầu

Phần tử Manifest (bản kê)	Phần tử Packaging (đóng gói); Mỗi phần tử Reference thuộc Manifest Nên tương đương với một SimplePart được tham chiếu từ một trong phần tử CompositeLists thuộc Packaging
Thuộc tính xlink:role trong phần tử Reference (tham chiếu)	Thuộc tính xlink:role trong phần tử SimplePart (thành phần đơn giản)
Phần tử AckRequested (báo nhận được yêu cầu)	Thuộc tính ackRequested (báo nhận được yêu cầu) trong phần tử MessagingCharacteristics ; một phần tử AckRequested (báo nhận được yêu cầu) được chứa trong tiêu đề SOAP Nếu thuộc tính ackRequested (báo nhận được yêu cầu) được thiết lập là "always"; Nếu nó được thiết lập là "perMessage", đầu vào được chuyển tới MSI là để được sử dụng để xác định nếu một phần tử AckRequested (báo nhận được yêu cầu) cần được bao gồm; tương tự như vậy, thuộc tính được ký thuộc AckRequested sẽ được thiết lập một cách thích hợp dựa trên cơ sở thuộc tính ackSignatureRequested (báo nhận ký được yêu cầu) và có thể được xác định đầu bởi vào chuyển đến MSI đó.
Phần tử MessageOrder (thứ tự thông điệp)	Thuộc tính messageOrderSemantics trong phần tử ReliableMessaging ; phần tử MessageOrder (thứ tự thông điệp) có mặt nếu phần tử AckRequested (báo nhận được yêu cầu) có mặt và nếu thuộc tính messageOrderSemantics trong phần tử ReliableMessaging (truyền thông điệp tin cậy) được thiết lập là "Guaranteed"
Phần tử ds:Signature	ds:Signature có mặt trong tiêu đề SOAP nếu thuộc tính isNonRepudiationRequired (yêu cầu không từ chối) trong phần tử BusinessTransactionCharacteristics được thiết lập là "true"; các tham số liên quan để lập cấu trúc chữ ký có thể đạt được từ phần tử SenderNonRepudiation (không từ chối bên gửi) và ReceiverNonRepudiations

Bảng sau đây chỉ ra các tham số hàm ẩn được thực hiện bởi dịch vụ truyền thông điệp ebXML không được chứa trong tiêu đề thông điệp và cách thức các tham số đó có thể được đạt được từ CPA.

Các tham số truyền thông điệp hàm ẩn	Thuộc tính/Phần tử CPA tương ứng
Retries (không trong tiêu đề thông điệp) nhưng được sử dụng quản lý việc truyền thông điệp xác thực hành vi của người gửi	Phần tử Retries (thuộc phần tử ReliableMessaging)
RetryInterval (không trong tiêu đề thông điệp) nhưng được sử dụng quản lý truyền thông điệp xác thực hành vi của người gửi	Phần tử RetryInterval (thuộc phần tử ReliableMessaging)
PersistDuration (không trong tiêu đề thông điệp) nhưng được sử dụng quản lý truyền thông điệp xác thực hành vi của người nhận	Phần tử PersistDuration (khoảng thời gian lâu dài) (thuộc phần tử ebXMLReceiverBinding)

Endpoint (không trong *tiêu đề thông điệp*) nhưng được sử dụng đối với việc gửi thông điệp SOAP

Phần tử **Endpoint** (điểm cuối) (thuộc **TransportReceiver**); kiểu thông điệp được gửi BẮT BUỘC được chuyển tới MSI đó; một **endpoint** thích hợp sau đó có thể được lựa chọn từ giữa các **Endpoint** được bao gồm thuộc phần tử **TransportReceiver** (truyền tải bên nhận)

Sử dụng **Service & Action** để xác định **DeliveryChannel** tương ứng

DeliveryChannel (kênh truyền)

Sử dụng **ReceiverDigitalEnvelope** để xác định thuật toán mật mã hóa và khóa

ReceiverDigitalEnvelope (đường bao số của bên nhận)

Sử dụng **SenderNonRepudiation** để xác định (các) chứng chỉ ký và **ReceiverNonRepudiation** để xác định các mẫu neo chứng thực và chính sách an ninh để áp dụng cho việc ký chứng chỉ

SenderNonRepudiation (không từ chối của bên gửi) và **ReceiverNonRepudiation** (không từ chối của bên nhận)

Use **Packaging** để xác định cách các bộ chứa thiết bị mang phải được gói gọn. Cũng sử dụng **Packaging** để xác định cách một **SimplePart** riêng phải được rút gọn và kiểm tra tính hợp lệ so với giản đồ của nó

Packaging (đóng gói)

Sử dụng **TransportClientSecurity** và **TransportServerSecurity** để xác định các chứng chỉ được sử dụng bởi máy server (máy chủ) và ứng dụng client (khách) đối với các mục đích xác thực

TransportClientSecurity (truyền tải an ninh bên client) và **TransportServerSecurity** (truyền tải an ninh bên server)

Sử dụng **DeliveryChannel** (kênh truyền) được xác định bởi **defaultMshChannelId** (id kênh MSH mặc định) đối với các thông điệp mức MSH độc lập như Xác thực, Báo lỗi, StatusRequest, StatusResponse, Ping, Pong, trừ khi ghi đè bởi **OverrideMshActionBinding**

Thuộc tính **defaultMshChannelId** trong phần tử **PartyInfo** (thông tin bên tham gia) và **OverrideMshActionBinding** (quy định hành động ghi đè MSH)

Phụ lục G

Bảng liệt kê các thuật ngữ

<u>Thuật ngữ</u>	<u>Định nghĩa</u>
THỎA THUẬN	Một sự sắp xếp công việc giữa hai bên tham gia để quy định trước các điều kiện chịu tác động mà hai bên tham gia sẽ trao đổi (Các thuật ngữ về hàng gửi, các thuật ngữ về thanh toán, các giao thức về hồ sơ hợp tác, v..v.) một thỏa thuận không không quy định hàm ý các cam kết kinh tế.
ỨNG DỤNG	Phần mềm trên mức MSH để thực hiện một dịch vụ bằng việc xử lý một hoặc nhiều thông điệp trong các trao đổi tài liệu được kết hợp với dịch vụ đó.
CẤP PHÉP	Một quyền hoặc một sự cho phép được chấp nhận cho một thực thể hệ thống để truy cập một nguồn hệ thống.
HOẠT ĐỘNG KINH DOANH	hoạt động kinh doanh được sử dụng để biểu diễn trạng thái của <i>quá trình kinh doanh</i> của một trong các bên tham gia. Đối với trường hợp người yêu cầu là trạng thái gửi yêu cầu, trạng thái đợi phản hồi hoặc trạng thái nhận.
HỒ SƠ HỢP TÁC KINH DOANH	Một hoạt động được tạo ra giữa hai hoặc nhiều bên tham gia đối với mục đích đạt được một kết quả đã quy định.
TÀI LIỆU KINH DOANH	Tập các thành phần thông tin được trao đổi như bộ phận của một hoạt động kinh doanh.
BÊN THAM GIA KINH DOANH	Một thực thể tham gia vào <i>giao dịch kinh doanh</i> với các bên tham gia kinh doanh khác.
QUÁ TRÌNH KINH DOANH	Phương tiện mà một hoặc nhiều các hoạt động được hoàn tất trong thao tác thực tiễn kinh doanh.
LƯỢC ĐỒ QUY ĐỊNH QUÁ TRÌNH KINH DOANH	Xác định tập các phần tử cần thiết để quy định các khía cạnh về thời gian chạy và các tham số về cấu hình để điều khiển các hệ thống của các bên tham gia được sử dụng trong hồ sơ hợp tác. Mục đích của giản đồ quy định BP là cung cấp cầu nối giữa lập mô hình <i>quá trình kinh doanh</i> điện tử và quy định các thành phần phần mềm trong kinh doanh điện tử.
GIAO DỊCH KINH DOANH	Một <i>giao dịch kinh doanh</i> là một khối đơn vị logic của kinh doanh được tạo ra bởi hai hoặc nhiều bên tham gia để tạo ra một trạng thái lỗi hoặc thành công có thể tính toán được. Cộng đồng, các bên tham gia và quá trình, tất cả có thể xác định và trạng thái tự bảo đảm trước khi <i>giao dịch kinh doanh</i> và trong một giao dịch mới có thể xác định, trạng thái tự bảo đảm sau <i>giao dịch kinh doanh</i> đó. Nói cách khác, nếu vẫn đang 'đợi' phản ứng và phản hồi của bên tham gia kinh doanh, <i>giao dịch kinh doanh</i> chưa hoàn thành.
BÊN CLIENT	Phần mềm để khởi tạo một kết nối với máy server (máy chủ).
HỒ SƠ HỢP TÁC	Hai hoặc nhiều bên tham gia làm việc cùng nhau thuộc một tập xác định các quy tắc.

GIAO THỨC VỀ HỒ SƠ HỢP TÁC	Giao thức để định nghĩa cho một quá trình hợp tác: 1. trình tự, tính phụ thuộc và ngữ nghĩa của các tài liệu được trao đổi giữa các bên tham gia để để tiến hành quá trình hợp tác và 2. Các khả năng truyền thông điệp được sử dụng khi gửi các tài liệu giữa các bên tham gia đó. Chú ý rằng một Quá trình hợp tác có thể có nhiều hơn một Giao thức về hồ sơ hợp tác bởi nó có thể được thực hiện.
THỎA THUẬN VỀ GIAO THỨC VỀ HỒ SƠ HỢP TÁC (CPA)	Thông tin được thỏa thuận giữa hai (hoặc nhiều) bên tham gia để bao gồm hoặc mô tả Giao thức về hồ sơ hợp tác cụ thể đã thỏa thuận để sử dụng. một CPA chỉ ra điều mà các bên tham gia "sẽ" thực hiện khi tiến hành một Quá trình hợp tác. một CPA có thể biểu diễn bởi một tài liệu.
SƠ LƯỢC VỀ GIAO THỨC VỀ HỒ SƠ HỢP TÁC (CPP)	Thông tin về một bên tham gia có thể được sử dụng để mô tả một hoặc nhiều Quá trình hợp tác và giao thức về hồ sơ hợp tác kèm theo mà bên tham gia đó hỗ trợ. Một CPP chỉ ra điều mà một bên tham gia "có thể" thực hiện để tiến hành một Quá trình hợp tác. Một CPP có thể biểu diễn bởi một tài liệu. trong khi theo logic, một CPP là tài liệu đơn, trong thực tế, CPP có thể là một tập các tài liệu được liên kết để diễn tả các khía cạnh biến đổi của các khả năng đó. Một CPP không là một thỏa thuận. Nó trình bày khả năng của một bên tham gia.
QUÁ TRÌNH HỢP TÁC	Một quá trình được chia sẻ bởi hai bên tham gia cùng tham gia công việc để để tiến hành một quá trình. Quá trình hợp tác có thể được xác định bởi một mô hình về hồ sơ hợp tác ebXML.
TƯƠNG THÍCH	Sự hoàn thành của một sản phẩm, quá trình hoặc dịch vụ của toàn bộ các yêu cầu được quy định; sự tham gia của một việc thực hiện các yêu cầu đó của một hoặc nhiều tiêu chuẩn hoặc quy định cụ thể.
CHỮ KÝ DẠNG SỐ	Một mã số có thể được gắn vào một thông điệp được truyền dạng điện tử để xác định người gửi duy nhất
TÀI LIỆU	Một tài liệu là dữ liệu nào đó có thể được trình bày trong một dạng thức số.
TRAO ĐỔI TÀI LIỆU	Một sự trao đổi các tài liệu giữa hai bên tham gia.
MẬT MÃ HÓA	Biến đổi mật mã dữ liệu (được gọi là "văn bản được mã hóa đơn giản") sang một dạng (được gọi là "văn bản mật mã hóa") để che đậy nghĩa gốc của dữ liệu để ngăn ngừa nó khỏi sự nhận biết hoặc sử dụng. Nếu biến đổi đó là thuận nghịch, quá trình nghịch đảo tương ứng được gọi "giải mật mã hóa", đó là một phép biến đổi để lưu trữ dữ liệu được mật mã sang trạng thái gốc của nó.
NGÔN NGỮ ĐÁNH DẤU MỞ RỘNG	XML được thiết kế để cho phép việc trao đổi thông tin (dữ liệu) giữa các ứng dụng và nguồn dữ liệu khác nhau trên World Wide Web và đã được tiêu chuẩn hóa bởi W3C.
SỰ THI HÀNH	An sự thi hành là sự thực hiện một quy định. Nó có thể là một sản phẩm phần mềm, hệ thống hoặc chương trình.
THÔNG ĐIỆP	Sự vận động của một tài liệu từ một bên tham gia tới một bên tham gia khác.
TIÊU ĐỀ THÔNG ĐIỆP	Quy định cấu trúc và thành phần cấu tạo của thông tin đó cần thiết đối với một dịch vụ truyền thông điệp ebXML để khởi tạo thành công hoặc xử l một thông điệp theo ebXML.
CÁC KHẢ NĂNG TRUYỀN THÔNG ĐIỆP	Tập các khả năng để hỗ trợ việc trao đổi các tài liệu giữa các bên tham gia. Các ví dụ là giao thức truyền thông và các tham số của nó, các định nghĩa an ninh và các đặc tính chung của việc gửi và nhận các thông điệp.

TCVN ISO/TS 15000-1 : 2007

DỊCH VỤ TRUYỀN THÔNG ĐIỆN	Một khung cơ cấu cho phép hoạt động cùng nhau, việc trao đổi các thông điệp an toàn và tin cậy giữa các bên tham gia thương mại.
ĐÓNG GÓI	Một cơ chế mục đích chung để tổ chức các phần tử vào các nhóm. Các gói có thể được lồng nhau trong các gói khác.
BÊN THAM GIA	Một bên tham gia là một thực thể như một công ty, phòng ban, tổ chức hoặc cá nhân tạo, gửi, nhận hoặc tiếp các tài liệu.
QUÁ TRÌNH PHÁT HIỆN BÊN THAM GIA	Quá trình hợp tác bởi một bên tham gia có thể phát hiện thông tin CPP về các bên tham gia khác.
THIẾT BỊ MANG	Một phần của dữ liệu/thông tin không là bộ phận của thiết bị mang ebXML.
BỘ CHỨA THIẾT BỊ MANG	Bộ chứa được sử dụng bao quanh thiết bị mang thực của một thông điệp ebXML. Nếu một thiết bị mang có mặt, bộ chứa thiết bị mang bao gồm một phần tiêu đề MIME (Thiết bị mang bao quanh ebXML đó) và một phần nội dung (chính thiết bị mang).
PHONG BÌ THIẾT BỊ MANG	Các tiêu đề MIME cụ thể được kết hợp với một phần MIME.
NGƯỜI NHẬN	Bên nhận một thông điệp.
ĐĂNG KÝ	Một cơ chế nhờ đó các mục và siêu dữ liệu của kho liên quan về chúng có thể được đăng ký như một con trỏ tới vị trí của chúng và toàn bộ siêu dữ liệu của chúng, có thể đạt được một kết quả của một câu truy vấn.
BÊN YÊU CẦU	Người khởi tạo một <i>giao dịch kinh doanh</i> .
BÊN ĐÁP ỨNG	Bên đối tác của người khởi tạo trong một <i>giao dịch kinh doanh</i> .
VAI TRÒ	Hành vi cụ thể được đặt tên của một thực thể tham gia trong một ngữ cảnh riêng. một vai trò có thể là tĩnh (như là; một kết thúc liên kết) hoặc động (như là; một vai trò về hồ sơ hợp tác).
CHÍNH SÁCH AN NINH	Một tập các quy tắc và thực tiễn để quy định hoặc điều chỉnh cách thức hệ thống hoặc tổ chức cung cấp các dịch vụ an ninh để bảo vệ nguồn hệ thống nhạy cảm.
NGƯỜI GỬI	Người khởi tạo của một thông điệp.
MÁY CHỦ	Phần mềm để chấp nhận một kết nối được khởi tạo bởi một bên Client
ĐỊNH DANH DUY NHẤT	Khái niệm trừu tượng của việc sử dụng một cơ chế và quá trình tiêu chuẩn để ấn định một trình tự các mã chữ-số vào các hạng mục trong sổ đăng ký ebXML, bao gồm: các thành phần chính, các thực thể thông tin tập hợp và <i>quá trình kinh doanh</i> .
ĐỊNH DANH DUY NHẤT THỐNG NHẤT (UUID)	Một định danh để truy cập duy nhất về cả thời gian và không gian, đối với không gian của tất cả UUID. một UUID có thể được sử dụng cho nhiều mục đích, từ việc gắn thẻ các đối tượng với một khoảng thời gian rất ngắn, để xác thực việc định danh các đối tượng rất dài qua một mạng.