Business Internet Consortium (BIC)

XML Convergence Workgroup

High-Level Conceptual Model for B2B Integration

Version: 1.0

28 September, 2001

CONFIDENTIAL
Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Legal Disclaimer

This document contains information that is confidential and proprietary to the Business Internet Consortium. The BUSINESS INTERNET CONSORTIUM name and design logo is a trademark of the Business Internet Consortium. All other trademarks and brands are the property of their respective owners.

Copyright

©2001 Business Internet Consortium. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. Printed in the United States of America

Trademarks

Table of Contents

Version History ..... **vii**

Preface ..... **ix**

1 Introduction **1**

1.1 Why We Need A Conceptual Architecture 2

1.2 Why We Need a Layered Architecture 3

1.3 Three Categories of Standards – a Gartner Group View 4

1.4 Basic Architectural Principles 6

2 Overall Model Description **7**

3 Conceptual Model Layers (Components) **10**

3.1 Backend Integration 11

3.2 Service-Oriented Architecture 12

3.3 Network Transport 13

3.4 Core XML Standards 14

3.5 Messaging 15

3.6 Repository 16

3.7 Registry Services 17

3.8 Business Content Format Definition 18

3.9 Universal Business Content 19

3.10 Specialized Business Content 20

3.11 Business Content Instance 21

3.12 Service Description Language 22

3.13 Process Description Language 23

3.14 Universal Business Process 24

3.15 Specialized Business Process 25

3.16 Business Process Instance 26

3.17 Trading Partner Agreement (TPA) 27

3.18 Security 28

3.19 Management 29

4 Next Steps **30**

Appendix A References **32**

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Appendix B Glossary 33

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Table of Figures

Figure 1. Functional Blocks of B2B Architecture 5
Figure 2. B2B Architecture Conceptual Model 8
Figure 3. Follow-up Work for the B2B Conceptual Model 31

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Version History

Version 0.1	16 August 2001	Jackson He and Pete Wenzel drafted based on BIC WG presentation
Version 0.2	25 August 2001	Jackson He made changes based on F2F meeting input
Version 0.3	05 September, 2001	Jackson He changed based on discussion with Pete Wenzel and Pat O'Sullivan
Version 0.4	13 September, 2001	Pete Wenzel and Steve Agritelley added more changes to the previous version
Version 0.5	18 September, 2001	Jackson He and Colin Hulme made changes based on input from WG meeting
Version 0.6	25 September, 2001	Tim Thomasma made changes and added more examples
Version 1.0	28 September, 2001	Final version

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Preface

Purpose of the Document

This document is to be used as a high-level architecture guideline for Business-to-Business (B2B) integration. It will serve as a framework to collect customer requirements and analyze current B2B implementations. The requirements and architectural gaps identified could then be fed into B2B standard bodies to improve future B2B standard definitions. It could also be used as a blueprint for B2B customers and vendors as they plan for implementations and develop roadmaps for future B2B systems.

Intended Audience

The target audience of this document is e-Business architects and business managers who are responsible for strategy and implementing B2B solutions; B2B standard bodies (W3C, OASIS, OAGI, etc.); B2B vendors and solution providers; and Members of other BIC workgroups.

Prerequisites

Readers are expected to have an understanding of the basic architectural concepts of B2B e-Commerce. This includes knowledge of Internet, component-based computing architecture and web services.

Scope of the Document

The focus of this document is on high-level components of both enabling technologies and business processes for B2B automation. This is not an architecture description for direct implementation, and does not address the details of the logical and physical models, or implementation details for each layer.

Structure of this Document

This document begins with a general introductory section which explains the conceptual model presented here, followed by more detailed descriptions of each component of the model. The descriptions follow the order of bottom to top, left to right.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

Acknowledgements

This document is based on a presentation that went though several iterations of review by the BIC XML Convergence Workgroup. Many thanks to the contributors from RosettaNet, Intel and Ford Motor Company. Also, our thanks for the contributions from other companies' (IBM, HP, Pennzoil, and SAP) members of the workgroup.

During the development of this white paper, the workgroup collected previous work regarding B2B integration architecture and web services frameworks from Gartner Group, IBM, Intel, AberdeenGroup, etc. All the reference materials are cited in the Reference Section of this document. We really appreciate the works from our colleagues in this area.

We also want to thank the following key authors and key reviewers to make this work possible.

Primary Authors	Key Contributors
Jackson He, Intel Corporation Pete Wenzel, RosettaNet on loan from SeeBeyond Tim Thomasma, Ford Motor Company	George Zagelow, IBM Dan Haller, HP Dawn Xu, Ford Motor Myron Menk, Pennzoil Sinisa Zimek, SAP Steve Agritelley, Intel Pat O'Sullivan, RosettaNet on loan from Intel Colin Hulme, Intel

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

1 Introduction

This document is primarily designed to show the architectural components needed in a Business-to-Business (B2B) environment. It shows the relationship of these components and examples of various standards that play the roles shown in this architecture. This is a high level view of B2B architecture aimed at delivering a framework for B2B standards convergence and interoperability.

This conceptual model described in this document is intended to be used as a vehicle to set a context and collect architectural requirements with B2B customers. The requirements could then be passed on to standards bodies and B2B vendors for standard and solution development. The conceptual model can also be used as a blueprint as B2B customers and vendors plan their implementation roadmap.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

1.1 Why We Need A Conceptual Architecture

B2B Automation Standardization is about interoperability of business content and message exchange between business systems of different enterprises, as well as the process automation associated with them. It requires many decisions at different levels. Without careful architectural thinking and planning, it is impossible to make right decisions that cover all the bases.

Because of the nature of complexity, it is unrealistic to have any single group or standard body make all the decisions. It requires collaboration of both vertical and horizontal standards bodies and industry leaders to work together to drive B2B standards definition and convergence. However, there is a need for a common view of what the basic components of B2B solutions should be and what are the common definitions of terminologies for these architectural elements. Therefore, we need to define a conceptual model that states the high level architectural elements, without getting into details of architectural definitions. Such a model will remain relatively stable over time as technologies, standards and implementation details evolve.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

1.2 Why We Need a Layered Architecture

As we stated in the previous sections, B2B solutions are very complex, involving business processes, business contents, and underline enabling technologies. By using a layered architecture and having the lower layers support and enable the upper layers, it is possible to divide a very complex problem into several less complex, more manageable sub-problems – a classic divide and conquer approach. Another advantage of the layered architecture is to allow different groups (standards bodies) to work on different layers at the same time, while remaining connected, which will shorten the time needed to solve the overall problem.

In the conceptual model described herein, the different elements of the B2B architecture can be represented as layers where one is built on top of the other; each layer supporting all of those above it. It makes sense then that the lower the layer the bigger the effect of deviation and duplication, therefore, the bigger the benefit for convergence. It is also interesting to note that since the lower layers are more technical and support more horizontal functionality, it is easier to seek commonalities in terms of basic technologies used. This results in greater opportunities for convergence. It is conceivable that we should drive standard convergence from the bottom up, i.e. from the Network Transport and Messaging layers up to more sophisticated business content and business process description layers.

We also realized that we may not be able to achieve convergence at all layers, especially the business content, business process, and backend integration layers. However, a broad agreement on convergence at the lower, enabling layers will make diversity on the top layers more effective and manageable. It also offers the potential of reuse and interoperability among the different business content and processes as we drive convergence to higher levels in the conceptual model.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

1.3 Three Categories of Standards – a Gartner Group View

There are different ways to categorize the different types of standards needed to conduct B2B transactions. We have adopted a view produced by Gartner Group (see References), which divides the conceptual architecture into three major sections:

- Top: Business Content and Process Standards – Meaning of information and processes
- Middle: Message and Associated Structure Standards – Syntax
- Base: Messaging Protocol and Tools Standards – Communication

A quote from the Gartner report defines the terms as follows:

Meaning of Information *– The relationship between values in the fields and the external world to which the data relates.*

Process Definitions *– The business rules, the definition of the roles of the parties involved, and the trigger events that provide the context for the exchange of information. Process definitions should cover the complete set of business events required to accomplish a business objective (e.g., placing an order would include steps such as sourcing, issuing a purchase order, receiving acknowledgments and dealing with changes) rather than just discrete steps (e.g., issuing a purchase order).*

Syntax *– The structure of the message, usually as a sequence of data fields.*

Communication Layer *– The mechanisms by which messages will be transported from party to party.*

Based on the categories described above, a high-level functional block diagram of the B2B architecture appears in Figure 1:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

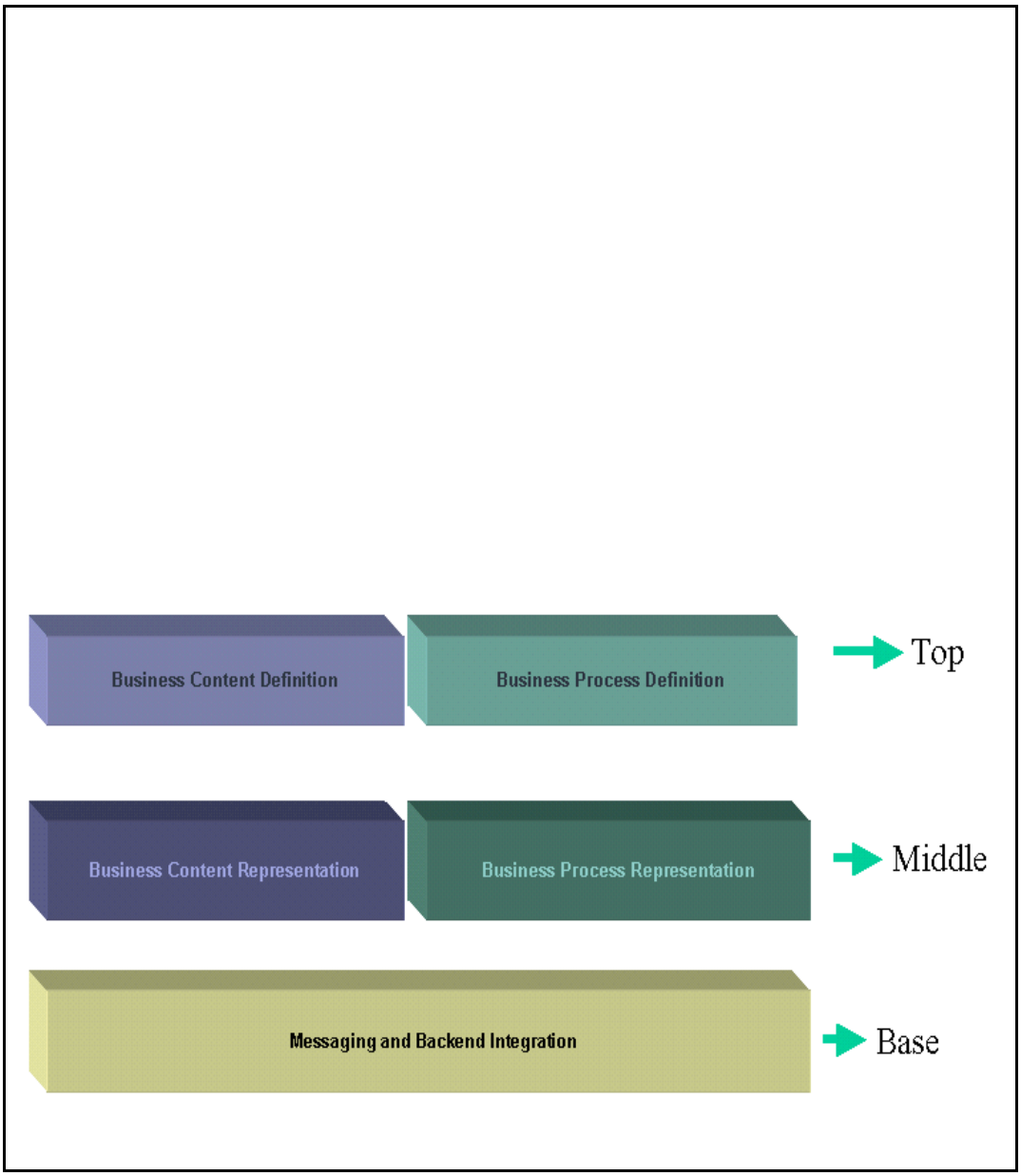

Figure 1. Figure 1. Functional Blocks of B2B Architecture

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

1.4 Basic Architectural Principles

Several basic architecture considerations (principles) were employed in developing this model:

- Openness: Open Standards, Open Platform.
- Layered specifications: Enabling technologies provide foundations to higher level functions and business logic; complex problems are divided into manageable layers, applying a divide-and-conquer methodology.
- Loosely coupled: Supports a loosely coupled approach to integrating trading partners' business applications, as well as B2B other components, connecting using messaging rather than programmable function calls.
- Extensibility: The architecture can grow over time to cover more business processes and more industries.
- Reuse: Reuse of business objects and patterns; reuse of technologies and architectural components, not reinventing the wheel.
- Self-describing: Each component has clearly defined interfaces to describe the services provided and methods for interactions. The description is publishable and searchable on a common registry.
- Dynamic discovery and binding: Services could be dynamically located on a common registry and composed into more complex services or transactional steps to serve particular business needs.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

2 Overall Model Description

Companies across all industries are realizing the fundamental benefits of using the Internet to integrate their supply chains in an automated fashion. The potential to reduce inventory, improve time-to-market, reduce transaction costs, and conduct business with a broader network of supply chain partners has direct and measurable benefits to a company's bottom line.

Because of the benefits that result from supply chain integration, companies are exploring open, XML-based standards that help remove the formidable barriers associated with developing a common business language and protocol for Internet-based collaboration, communication and commerce.

The conceptual model as shown in Figure 2 was developed with the input from many industry and technology organizations and respected thought leaders. We acknowledge that there may be differing views or alternative perspectives to the model. However, we believe that after many iterations of development by the BIC XML Convergence Workgroup, this conceptual model reflects the current implementations and future vision of B2B Automation. The model is supported by several case studies and has proven to be a relevant and true reflection of the basic components needed for B2B Automation. (Please see the case studies provided by this workgroup.)

Throughout the development of the conceptual model, a best effort has been made to be inclusive and to comprehend inputs from different aspects. We take into consideration the directions of each of the leading standards bodies, industry consortia and majors ISVs. The conceptual model is meant to describe a reconciled view of the basics for B2B Automation. We consider the conceptual model a living document that evolves with the changing faces of business and technology. On the other hand, we also believe this high-level, generic conceptual model will be relatively stable for a period of time as B2B automation technology and standards are developed.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

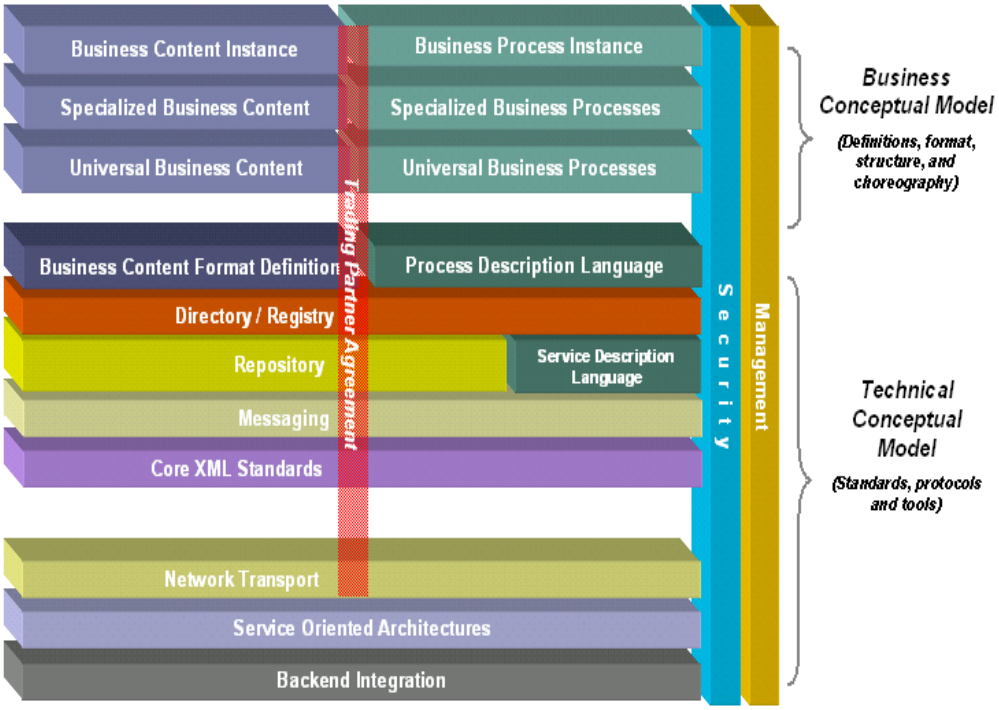

Figure 2. **Figure 2. B2B Architecture Conceptual Model**

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

This model is divided into two general sections: the Business Conceptual Model on the top and the Technical Conceptual Model at the bottom. The Business Conceptual Model is focused on the definition, structures and formats of business transactions, as well as the business processes that handle transactions. The Technical Conceptual Model provides the technical foundation that enables business document and process definition. The model is also divided into left and right halves: the left side represents the components that support business content (payload of business transactions), while the right side represents layers associated with business processes. The three vertically oriented layers (Trading Partner Agreement, Security, and Management) have implications across all, or nearly all, layers.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3 Conceptual Model Layers (Components)

The following sections describe each layer of the model, as well as the relationship of each layer to those around it.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.1 Backend Integration

Definition: Provides hooks into the backend enterprise systems through API or shared messaging bus. Includes functions like business logic processing and format transformation.

Relationship with other layers: This is the gateway to the backend ERP systems. It is developed using tools from the Service-Oriented Architecture layer and communicates with upper layers through the Network Transport and Messaging layers.

Due to the proprietary nature of ERP systems, there are fewer opportunities for standardization in this layer. However, XML provides a vehicle to have common adapters for popular ERP systems. As part of the private processes development, B2B system integrators need to work with ERP vendors to build seamless connections. From the B2B standards development point of view, it may be out of the scope. However, this layer is vital to developing end-to-end B2B solutions and is very critical to customers who want to implement their own solutions. One strategy is to work with ERP vendors like SAP, PeopleSoft, etc., so that they will adopt the principle of this conceptual model and develop backend integration solutions that match the conceptual model and are suitable for corresponding ERP customers.

Examples: Proprietary gateways and adapters to Enterprise Resource Planning (ERP) and database systems. No known open standards for interfaces or mappings.

Notes:

- Goal of this WG is to drive convergence and help customers deploy faster. Again, case studies that focus on different kinds of back-end integration will have important values for customers and ERP vendors.
-

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.2 Service-Oriented Architecture

Definition: Development platform for Web Services. This layer provides basic development standards and tools (Java, C#, J2EE, .NET, etc.) and related development environments. This layer also defines APIs that "glue" e-Business transaction systems with the backend ERP systems.

Relationship with other layers: This is the development environment that "glues" the e-Business transaction systems with the backend ERP systems. This layer will have native support of the messaging layer and enable APIs and gateways to talk to the upper layers through the Messaging layer. It will have native support of development tool standards used for developing APIs and web services.

Examples: J2EE (current) .NET(future)

Notes: Major ISVs comprise the primary stakeholders for this layer, who should be able to build competitive solutions based on XML standards.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.3 Network Transport

Definition: This layer addresses the basic messaging transport protocols needed to communicate on the Internet, messaging services that provide for asynchronous publish/subscribe, asynchronous message queuing, and synchronous request/reply. Additionally, it addresses how messages are placed on and off the transport bus . These standards specify mechanisms for transporting messages in a secure and reliable way.

Relationship with other layers: This layer provides the foundation for messages to get on the wire. The Message Packing and Routing based on the XML Core Standards (XML-based messaging layers) have to bind with the Network Transport layer.

Examples: HTTP, HTTPS, SSL, SMTP (current)

Notes: Major portions of this layer have been well established and used within the Internet community over the past several years.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.4 Core XML Standards

Definition: Basic XML protocols that are associated W3C standards for defining document types and for accessing the data within the documents. This syntax is used to express specifications in the layers above for defining the representation of business content and processes.

Relationship with other layers: This layer contains the basic standards needed to define business content formats. It provides the basis for business semantics definition, XML parsing and business process interaction.

Example: XML DTD, XSL/XSLT (current), XML Schema, Xforms (future)

Notes: W3C defines most of the core XML standards on this layer.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.5 Messaging

Definition: Standardized message and envelope structure and layout definitions, which have specific technical purposes. This layer addresses the need to record session and communication settings for message transport in order to enable coordination between parties in a business transaction, including parameters that control Reliable Messaging, Secured Messaging, etc.

Relationship with other layers: This layer is the foundation of communications amongst all the layers. It provides the lower-level message exchange support for the Service Description Language, Directory/Registry, Process Description Language layers. It also provides a base for Business Content Format Definitions layer.

Example: RosettaNet RNIF1.1, SOAP (current), ebXML TRP (future)

Notes: Session management and transaction coordination in a loosely-coupled environment will be necessary for multi-organizational business operations where the e-Business solutions may not be readily accessible at all times.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.6 Repository

Definition: Standardized repository services that specify the structure and access protocol and schemas for business content storage and retrieval, which includes the term, its constraints, its representations, etc.

Relationship with other layers: This layer provides standard-based services for storage and retrieval of entries at the Registry Services. It will provide a platform-independent way to store and retrieve business content format definition schema and business process description.

Example: RosettaNet Dictionary Repository (current), ebXML Reg/Rep (future)

Notes:

- There have not been standards defined for this layer. Repository is typically defined on ad hoc based and closely tied to database technologies used for implementation.
- The advantage of having an implementation platform independent repository standard for business content is that the higher layers of business content will have a standards way to store and retrieve business content structures once they are defined that is cross-platform, abstract from implementation details — something Additionally, it addresses in an ad hoc manner today and need to standardize.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.7 Registry Services

Definition: Specifies the structure and access protocol of registries and repositories that trading entities can access to discover each other's capabilities and services. It covers naming, directory, registry, privacy, authorization and identification services.

Relationship with other layers: This layer is used to publish and register business processes and services. Business processes that need to dynamically explore and discover available services or that publish services for other businesses to use will make use of the services specified in this layer. The Registry Services could be used to publish and discover both business content and business processes. The Registry Service keeps a list of the entries of entities and stores the objects in the Repository.

Example: UDDI (early pilot), ebXML Req/Rep (future)

Notes: Enables electronic discovery and configuration of business processes, or web services, between trading partners.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.8 Business Content Format Definition

Definition: Business Content includes everything that composes the payload of business transactions, which dictionary entries, composition of dictionary entries, special business documents, and attachments. Business Content Format Definition is the specification of the data structures, data types, constraints and code lists of all the items necessary to compose valid business content.

Relationship with other layers: This layer specifies the structure and semantics for particular business processes. It is built on top of the Core XML Format Standards and with knowledge of particular business processes required for business transactions on the right side (business process side) of the model. This layer also takes into consideration the schema required to store and retrieve content formation definition based on the services provided from the dictionary and repository layer.

Example:

- Current: RosettaNet Technical Dictionary Structure, RosettaNet Business Dictionary Structure, RosettaNet PIP Service Content, OAGI Business Object Document (current), ebXML Core Components (future)

Notes:

- Business Content is a broad term and could mean different things at different context of business transactions.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.9 Universal Business Content

Definition: Specifies business terminology and accepted values that may be universally used in business messages that support a broad range of industries, business models and locales; the vocabulary used to construct the business content of a message. This content covers many domains of discourse, such as product, materials management, finance, quality.

Relationship with other layers: The content of this layer is expressed in the structure defined by the Dictionary Structure layer. The content of this layer is used in Specified Dictionary Content and Business Content Instance.

Example: RosettaNet Business Dictionary, OAGIS, HR-XML, CBL, HL7, boleroXML, eBIS-XML, PDX (current), ebXML Core Components (future)

Notes:

- The distinction between "universal" and "specialized" is relative and could change over time and situation. The purpose of this distinction to drive reuse and achieve manageability and economies of scales. Our overall goal is to drive "universal" definitions whenever economically feasible. We understand that there are areas have to stay "specialized" to meet business needs. However, something "specialized" today could change into "universal" tomorrow. We should consistently evaluate the situation and turn "specialized" into "universal" whenever possible.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.10 Specialized Business Content

***Definition**:* Industry- or supply chain-specific technical lexicon (terms, properties, values, taxonomic structures) to be used to extend and specialize the Universal content to construct the content of an industry-specific business document.

Relationship with other layers: This layer specifies the content of specialized dictionary (vertical industry, particular business model, and locale specialization, etc.). The content of this layer will be used by the Business Content Instance.

Example: RosettaNet Technical Dictionary (current), Automotive Industry Action Group eAPQP XML (future)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.11 Business Content Instance

Definition: Describes the particular business content exchanged during a particular business transaction. An instance could contain recursive combinations of universal and specialized business content defined in the lower layers.

Relationship with other layers: This layer is an instantiation of Universal and specialized business content, plus necessary business context to conduct a business transaction. It is the actual payload of a business transaction.

Example: Purchase order from Arrow to Intel over RosettaNet (current)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.12 Service Description Language

Definition: This layer describes the tools and languages for service implementation and service interface, which is key to achieving loosely coupled architecture and reducing the amount of custom programming as well as the effort of integration between service requester and service provider.

Relationship with other layers: This layer provides the tools used to define basic service behavior and interface with other services. It provides a foundation for higher-level business process and process chorography definition at the Business Process Description layer.

Example: WSDL, WSEL (future)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.13 Process Description Language

Definition: Specifies the way in which any business process (whether Universal or Specific in nature) is recorded, such that is understood and executable in a repeatable fashion by a wide array of humans and/or applications.

Relationship with other layers: This layer builds on top of Service Description and Directory Services layers and provides tools to specify the semantics of business processes of both Universal and Specialized.

Example: UML (via XMI), XLANG (current), ebXML BPSS, BPML, WSFL(future)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.14 Universal Business Process

***Definition**:* Specifies business processes that are applicable to a broad range of businesses, regardless of the vertical industry or locale within which the business operates or of the specific characteristics of the business. These processes cover many domains of activity that businesses engage in, such as collaborative product development, request for quote, supply chain execution, purchasing, and manufacturing.

Relationship with other layers: This layer uses tools provided by the Process Description layer to describe the business process sequencing and choreography amongst processes that are Universal to all businesses or business domains.

Example: Invoicing process, Purchasing process, Base level Purchase Order (current)

Notes:

- The distinction between "universal" and "specialized" is relative and could change over time and situation. The purpose of this distinction to drive reuse and achieve manageability and economies of scales. Our overall goal is to drive "universal" definitions whenever economically feasible. We understand that there are areas that have to stay "specialized" to meet business needs. However, something "specialized" today could change into "universal" tomorrow. We should consistently evaluate the situation and turn "specialized" into "universal" whenever possible.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.15 Specialized Business Process

Definition: Specifies business processes that are not Universally applicable but instead are specific to a business operating within a specific industry or supply chain (such as Electronic Components, Pharmaceuticals, Automotive), and locales or business models:

- Simple processes that are unique to a business model (e.g., non-profit), or
- Higher-level composites or sequences of specified Universal Business Processes that are unique to a business model.
- Special business processes defined for particular locale and region

Relationship with other layers: This layer uses tools provided by the Process Description layer to describe the business process sequencing and choreography amongst processes to describe special processes needed to fit a particular supply chain, a business model, or a locale or region.

Example: (current) Purchase Order tax modules added to a (Universal) Base Purchase Order

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.16 Business Process Instance

***Definition**:* Particular instance of business processes for business transactions that could be a recursive and complex combination of Specialized and Universal Business Processes. This layer also defines binding of business processes and business content to complete particular business transactions

Relationship with other layers: This layer is an instantiation of Universal Business Process and Specialized Business Process to fulfill a particular business transaction.

Example: The process used to complete a Purchase Order between Arrow and Intel (current)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.17 Trading Partner Agreement (TPA)

Definition: Dynamic creation and management of relationships between partners. Profiles of trading partners' B2B infrastructure, protocols, contractual agreement for transactions.

Relationship with other layers: This layer spans multiple layers in the model. It contains the technology, business content (including structures), and business processes that a trading partner uses to conduct a particular business transaction. This layer touches all the layers from Network Transport to Business Content Instance and Business Process Instance, including the Security Layer. It is a signature (snapshot) of the preferred way the business partners want to trade.

Example: ebXML CPP/CPA (future)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.18 Security

Definition: This layer spans a wide range of abstractions from basic encryption, authentication and authorization on the Core XML layer, to non-repudiation and security policies in the business process layer. It includes both the technologies used to implement security functions and the policies that manage and apply the technologies.

Relationship with other layers: This layer spans all the layers that have security needs (all the way from Backend Integration to Business Content Instance and Business Process Instance). For dictionary definitions, there may not be security needs; however, dictionary authoring and publishing will need basic security functions to protect business integration.

Example: XML Signature, Digital Certificates (current)

Notes:

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

3.19 Management

Definition: This layer specifies system management tools and standards that can be used to discover the existence, availability and health of a B2B solution. In addition, the management tools should also be able to control and configure the components. Furthermore, this layer will manage the quality of services for the overall system and ensure that the level of services will not degrade over time.

Relationship with other layers: Like Security layer, this layer spans all the layers that have system management needs (all the way from Backend Integration to Business Content Instance and Business Process Instance) to monitor the health of the B2B system and adjust configurations to maintain overall system at optimal operation states.

Example: SNMP (current)

Notes:

- There has not been a system management standard for B2B systems defined. This will be a future feature that has yet defined.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

4 Next Steps

The purpose of this conceptual model is to provide a framework for B2B customers, vendors, and standard bodies alike to have a common view of what it takes to develop automated B2B solutions. However, the conceptual model alone is not directly used in solving customers' immediate B2B problems and in providing input to standard bodies.

As indicated in the figure below, we need to develop additional material to make the model truly useful. For example, case studies will be generated to demonstrate how the conceptual model is relevant to the reality of B2B integration implementations today, and how it reflects the vision and direction of future solutions. These case studies will also identify the gaps between specific, existing solutions and the vision the conceptual model reflects. These gaps must be addressed in order to increase the value of B2B Automation.

We will then collect use cases and customer requirements for what could be needed in the short-term solutions (next 6~12 months), and what needs to happen in the long-term solutions (12 months and beyond), in order to develop a migration roadmap that shows a path for achieving the promises of B2B Automation. The results of this follow-up work based on the conceptual model will be fed to B2B customers, vendors, and standard bodies as a guideline for future implementation planning and standard development. Figure 3 is a graphical representation of the different documents centered around this conceptual model and their intended audience. The BIC XML Convergence Workgroup will follow through with delivering the documents in the next few months.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

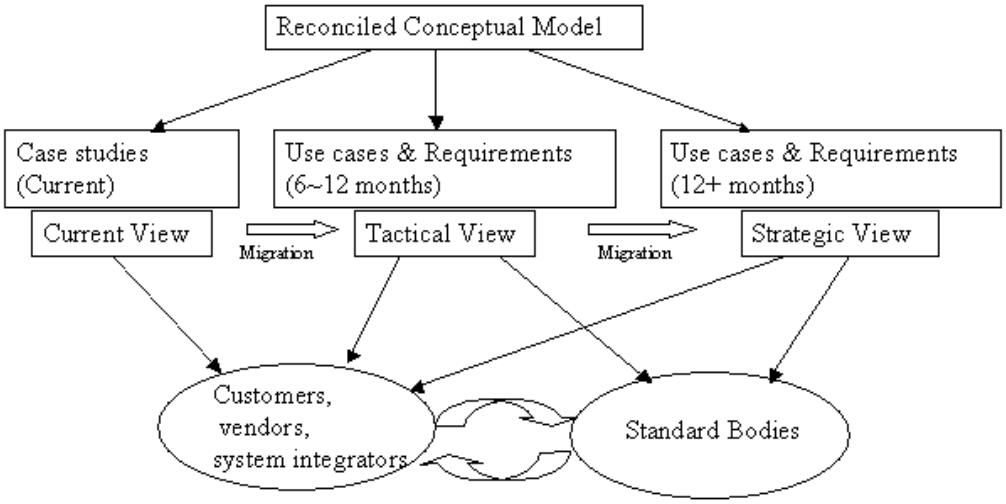

Figure 3. Figure 3. Follow-up Work for the B2B Conceptual Model

There are several key standard bodies we want to engage. Amongst them, the BIC Workgroup member companies already have established tie connections, such asW3C, OASIS, RosettaNet, OAGI, etc. In the coming months, we are going to present this whitepaper and the vision for convergence from this workgroup to demonstrate the value of such a conceptual model and increase BIC's visibility and credibility in driving XML-based standard convergence.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

APPENDIX A REFERENCES

- Heather Kreger, IBM Software Group, “Web Service Conceptual Architecture 1.0”, May 2001
- Gartner Group, "OAGI: Fostering Standards Evolution or Revolution?" A Gartner Advisory Research Note 14 December 2000
- Randy. E. Hall , “W3C Web Service Position Paper from Intel”, W3C Workshop, April 12, 2001
- AberdeenGroup, “Trading Community Management: The Fast Track to Positive e-Business ROI”, June 2001
- RosettaNet, “RosettaNet Architecture Conceptual Model”, July 2001

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

APPENDIX B GLOSSARY

asynchronous: Communication among distributed processes is said to be "asynchronous" when there is no expectation that the reply to a request comes within the time interval in which the communication session of the request is still "live." Compare with “synchronous.”

authorization: permission to access a protected resource, a service, or sensitive information. Sometimes confused with *authentication*, which is simply verification that a user is who he claims to be. One can be properly authenticated but not be authorized to access a protected resource, a service, or sensitive information.

DTD: a type of schema used to specify the structure and semantics of an XML document or message.

e-business: an enterprise that conducts many of its business functions through electronic means. The term also refers to businesses that operate on the Internet and offer goods, services, and information for sale via the Web. (from Jonar C. Nader, Prentice Hall’s Illustrated Dictionary of Computing, 3rd edition, 1998)

framework: a set of related architectural components.

guideline: a set or collection of specifications, sometimes including specific implementation advice.

implementation framework: guidelines for creating instances of related architectural components.

message: a properly packaged business action or business signal.

message choreography: the exchange of business actions and business signals required to complete a specific business activity.

non-repudiation: the ability of a message transfer system to provide unforgeable evidence that a specific action occurred. Three types of the non-repudiation services are most common: non-repudiation of origin, non-repudiation of submission, and non-repudiation of delivery. Non-repudiation of origin protects against any attempt by a message originator to deny sending a message. Non-repudiation of submission protects against any attempt by a message transfer agent to deny that a message was submitted for delivery. Non-repudiation of delivery protects against any attempt by a message recipient to deny receiving a message.

Payload: the Business Content plus any file attachments needed to complete a transaction.

protocol: a protocol is a formal set of rules and conventions that governs how computers exchange information over a network medium.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.

schema: a specification for the structure and semantics of some related data. One uses the schema to validate or otherwise understand a group of data. One type of schema is the XML-DTD.

standard: a set of clearly defined and agreed-upon conventions for specific programming interfaces that has been approved by a formally constituted standards-setting body.

structure: something composed of organized or interrelated elements; the manner in which the elements of something are organized or interrelated

syntax: the patterns of formation of sentences and phrases from words and the rules for the formation of grammatical sentences in a language.

trading partner: An organization or company that transacts business

Trading Partner Agreement (TPA): information exchanged between trading partners that describes certain mutually agreed upon execution parameters and service level expectations that will be used when conducting business between them.

valid XML document: An XML document is **valid** if it has an associated document type declaration and if the document complies with the constraints expressed in it. (From World Wide Web Consortium, *Extensible Markup Language (XML) 1.0: W3C Recommendation* 10-February-1998.)

well-formed XML document: An XML document that, taken as a whole, matches the XML production labeled "document," meets all the well-formedness constraints given in the XML specification, and each of the parsed entities which is referenced directly or indirectly within the document is well-formed. A well-formed document may also be "valid" if it meets additional criteria. (Adapted from World Wide Web Consortium, *Extensible Markup Language (XML) 1.0: W3C Recommendation* 10-February-1998.) (*See also* valid XML document.)

XML document: a data object made up of virtual storage units called entities, which contain either parsed or unparsed data. Parsed data is made up of characters, some of which form the character data in the document, and some of which form markup. Markup encodes a description of the document's storage layout and logical structure. (From www.w3.org/TR/PR-xml-971208) *See also* well-formed XML document; valid XML document.

Copyright © 2001 Business Internet Consortium.
Other names and brands are property of their respective owners.