Diversified SCM Standard for the Japanese Retail Industry

Koichi Hayashi Naoki Koguro Reki Murakami

UL Systems, Inc.
Triton Square, Tower X 14F, 1-8-10, Harumi, Chuo-ku,
Tokyo 104-6014, Japan
+81-3-6220-1400

{koichi.hayashi, naoki.koguro, reki.murakami}@ulsystems.co.jp

ABSTRACT

In this paper, we present the concept of a *diversified SCM (supply chain management) standard* and *distributed hub architecture* which were used in B2B experiments for the Japanese retail industry. The conventional concept of B2B standards develops a single ideal set of business transactions to be supported. In contrast, our concept allows a wide range of diverse business transaction patterns necessary for industry supply chains. An industry develops a standard SCM model that partitions the whole supply chain into several transaction segments, each of which provides alternative business transaction patterns. For B2B collaboration, companies must agree on a *collaboration configuration*, which chooses the transaction alternatives from each segment. To support the development of a B2B system that executes an agreed collaboration, we introduce an SOA (service oriented architecture) based pattern called a *distributed hub architecture*. As a hub of B2B collaboration, it includes a complete set of services that can process every possible business transaction included in a standard SCM model. However, it does not function as a centralized service that coordinates participants. Instead, it is deployed on every participant and executes the assigned part of the supply chain collaboratively with other distributed hubs. Based on this concept, we analyzed actual business transactions in the Japanese retail industry and developed a standard SCM model, which represents more than a thousand possible transaction patterns. Based on the model, we developed an experimental system for the Japanese retail industry. The demonstration experiment involved major players in the industry including one of the largest general merchandise stores, one of the largest wholesalers, and major manufacturers in Japan.

Categories and Subject Descriptors

K.4.4 [**Computers and Society**]: Electronic Commerce – *Distributed commercial transactions, Electronic data interchange (EDI)*; H.3.5 [**Information Storage and Retrieval**]: Online Information Services – *Web-based services*

General Terms

Design, Experimentation, Languages, Standardization

Keywords

SOA (service oriented architecture), Web Services, business process management, B2B collaboration, supply chain management, standardization, ebXML, retail industry

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

1. INTRODUCTION

The global spread of the Internet has promoted the innovation of commercial transactions called B2B (Business-to-Business). Typified by RosettaNet, B2B commerce covers inter-corporate business communication using an XML-based protocol, while B2C (Business-to-Customer) commerce refers to Internet shopping in a Web browser. Although RosettaNet has demonstrated a promising future of B2B, B2B standards have not spread as rapidly as expected, except in some sectors, such as the IT industry. Many industries are still looking for a convincing business benefit in return for such new IT investment.

The main purpose of B2B collaboration is an optimization of SCM (supply chain management) that includes external partners. Companies involved in a supply chain must make coordinated efforts for delivery of high value-added services to consumers. B2B integration is a means for exchanging information about orders, inventory, and delivery among companies. As B2B protocols use common Internet technologies, such as HTTP and XML, a company can introduce a B2B system with flexible and sophisticated functions at lower cost than conventional Electronic data interchange (EDI).

Industry standards play a key role in the success of B2B collaboration. Up-to-date industry standards specify inter-corporate processes for message exchange, sometimes called *public flows*, as well as message formats to be exchanged. For example, RosettaNet defines more than 100 PIPs (Partner Interface Processes), which are message exchange processes used in the IT industry [10]. ebXML standards further provide BPSS (Business Process Specification Schema) as a generic language to define standard business transactions for each industry [6].

Although the benefit has been clearly demonstrated and the tools for standardization are ready to use, many industries are still hesitating to adopt a standard SCM based on B2B collaboration. In reality, the most efforts for standardization in an industry are spent to reach an agreement among key players in there rather than to resolve technical issues to accomplish it's business requirements. By its nature, SCM functions as designed only when most of the players in the industry accept the standard. Conventionally, a B2B standard means a single ideal set of business transactions that should be supported by every participant.

On the other hand, adoption of a standard implies rebuilding of the business transactions with partners and paying costs for a new system that conforms to the standard. As each player has their own business interests, it is difficult to define an ideal SCM standard acceptable for a wide range of players. The Japanese retail industry, for example, involves various types of participants, such as supermarkets, drugstores, home centers, wholesalers, distributors, manufacturers, agricultural cooperatives, *etc*. Thus, a wide variety of business transactions are used in this industry.

An SCM standard acceptable to various players should not require drastic changes to existing business transactions between partners. It should also minimize the cost of development of the new B2B collaboration systems that support the standard.

In this paper, we propose a *diversified SCM standard*, a standardization concept, and *distributed hub*, an architecture pattern for developing B2B collaboration systems that support the standard SCM model. The diversified SCM standard can represent a wide variety of business transactions patterns. As it allows companies to select the patterns most similar to those that they currently use, it does not result in drastic changes in business transactions. The distributed hub architecture, which is based on the SOA (service oriented architecture) concept, enables the implementation of a B2B system conforming to a diversified standard at minimal cost. By representing the business transaction patterns as service combinations, the necessary pattern for executing the collaborations can be configured as service combinations.

Here, we further present the diversified SCM standard that we developed for real supply chains in the Japanese retail industry and results of the demonstration experiment using an experimental system based on the distributed hub architecture.

The remainder of this paper is organized as follows. Section 2 briefly reviews related work. Section 3 illustrates the concept of the diversified SCM standard. Section 4 describes the distributed hub architecture. Section 5 shows the results of application to supply chains for the Japanese retail industry. Section 6 addresses future work. Section 7 concludes the paper.

2. RELATED WORK

This section presents a brief review of related work.

2.1 Standardization Using XML

XML is the key technology for standardization of B2B collaboration. Many industries have developed standards using XML. However, most define only XML-based message formats and do not define the business transaction.

RosettaNet is the first and most successful activity that defines business transactions for the IT industry. It developed standard business transactions as forms of PIPs covering a wide range of processes used in the industry. The scope of PIPs includes the transaction segments with which this paper is concerned. PIPs are classified into several clusters, each of which is divided into various segments. For example, cluster 3 is concerned with “Order Management.” This cluster involves ordering as segment 3A, distribution as segment 3B, and settlement as segment 3C. However, PIPs do not allow much diversity as RosettaNet basically defines single monolithic processes. Moreover, PIPs represent separate processes instead of defining a whole supply chain. Therefore, to support a long supply chain from ordering to payment, companies need to gather associated PIPs and coordinate them on their own, *i.e.*, in a non-standardized way[5].

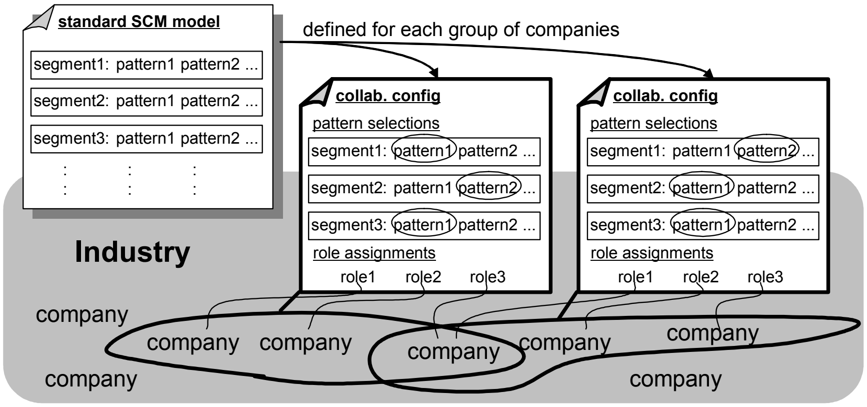

standard SCM model
segment1: pattern1 pattern2 ...
segment2: pattern1 pattern2 ...
segment3: pattern1 pattern2 ...
defined for each group of companies
collab. config
pattern selections
segment1: pattern1 pattern2 ...
segment2: pattern1 pattern2 ...
segment3: pattern1 pattern2 ...
role assignments
role1 role2 role3
collab. config
pattern selections
segment1: pattern1 pattern2 ...
segment2: pattern1 pattern2 ...
segment3: pattern1 pattern2 ...
role assignments
role1 role2 role3
Industry
company
company
company
company
company
company
company
company

Figure 1. Diversified SCM standard.

While RosettaNet is concerned with a specific industry, ebXML provides generic tools to define industry standards. It includes various specifications from messaging to coordination. MS (message service) is a messaging protocol via the Internet. BPSS (business process specification schema) provides the generic syntax for describing business transactions. CPPA (collaboration partner profile and agreement) provides a way to make agreements for business transactions with business partners. Some commercial B2B servers support these standards. Although they are useful tools for describing determined SCM specifications, they are not concerned with how to determine SCM standards.

2.2 SOA Based Process Management

SOA is attracting a great deal of interest from enterprises seeking the fundamental principles to improve their IT infrastructures. SOA is a concept that composes services distributed in a network to define a new service. Although the concept itself is not new in the context of distributed computation, the set of Internet technologies, such as XML and Web Services, increases its value and possible utility.

In the context of SOA, B2B collaboration is recognized as a coordination of services across companies. Therefore, BPEL4WS [3], which is a standard means for coordinating services, can support B2B collaborations. Use of BPEL4WS has been intensively investigated [2, 8]. It is applied to various types of service compositions from intra-corporate application integration [7] to inter-corporate negotiation [9]. BPEL4WS can directly describe the invocation sequence of necessary services, while BPSS merely describes interfaces between companies. However, BPEL4WS is typically used in a centralized service that orchestrates distributed services, like an intra-corporate EAI server. It is often unacceptable for all of a company’s business transactions to depend on a central server shared with other industry players, including their competitors. Although the distributed coordination model for BPEL4WS has been studied in detail, difficulties still remain [4].

3. DIVERSIFIED SCM STANDARD

This section describes the concept of a diversified SCM standard, which is designed to represent a wide variety of public flows included in industry supply chains.

The essence of the concept is the determination of a specific standard public flow in two steps using a *standard SCM model* and *collaboration configuration*. Figure 1 shows an overview of this idea. A standard SCM model, which is developed by an industry, provides possible alternatives of public flows used in the industry. A collaboration configuration, which is a set of chosen alternatives agreed between partners, defines a specific public flow used for collaborations with the partners.

3.1 Standard SCM Model

A standard SCM model is developed for an industry to represent a variety of public flows used in the industry supply chains. It defines the roles and collaborations among them. An SCM model is partitioned into more than one *transaction segment*. Each transaction segment has one or more alternative *transaction patterns*.

Figure 2 shows a simplified example of a standard SCM model using a BPMN [1] like notation. It can be described using XML as shown in Figure 3. This model specifies four roles: buyer, seller, shipper and receiver. The whole supply chain is partitioned into three transaction segments: ordering, distribution, and settlement. Each transaction segment has two alternative transaction patterns. Therefore, by simple arithmetic this model represents a total of 8 (=2×2×2) possible public flows.

A transaction pattern defines business processes and transitions among them. It specifies input and output messages and roles for each business process. Each business process has a specific procedure to be executed by the assigned role. The associated processes are activated on receipt of an input message. An activated process invokes the specified procedure with the input information and creates output messages. These output messages further activate other processes. This message-driven invocation sequence forms a chain of business processes.

For example, pattern 1 of the ordering segment defines collaboration between a buyer and a seller. Processes associated with each role are placed in a box called pool. Each pool is divided into two lanes, *i.e.,* “system” for processes of incorporate systems and “public flow”. The public flow lanes include three

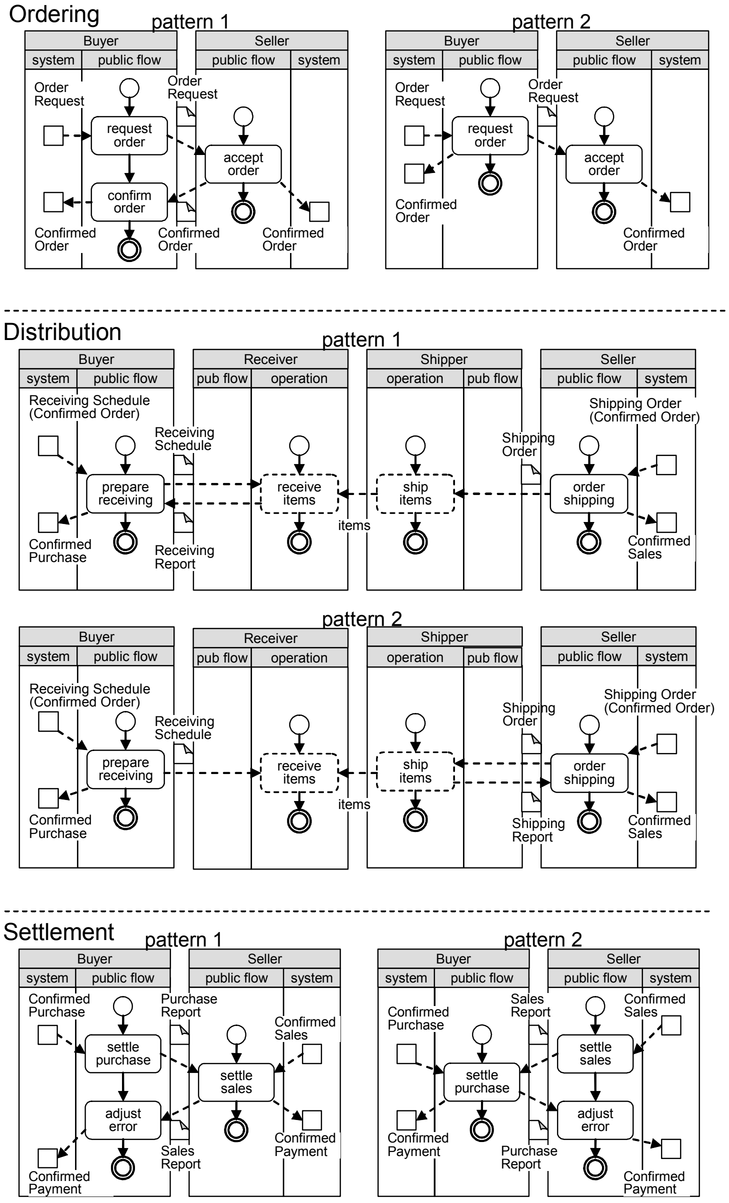


Figure 2. Standard SCM model.

```
<segment name="Ordering">
  <pattern id="1">
    <process name="request order" role="Buyer">
      <input>
        <message type="Order Request" from="system"/>
      </input>
      <output>
        <message type="Order Request" to="Seller"/>
      </output>
    </process>
    <process name="accept order" role="Seller">
      <input>
        <message type="Order Request" from="Buyer"/>
      </input>
      <output>
        <message type="Confirmed Order" to="Buyer"/>
        <message type="Confirmed Order" to="system"/>
      </output>
    </process>
      ...   ...   ...
 </pattern>
  <pattern id="2">
    <process name="request order" role="Buyer">
      <input>
        <message type="Order Request" from="system"/>
      </input>
      <output>
        <message type="Order Request" to="Seller"/>
        <message type="Confirmed Order" to="system"/>
      </output>
    </process>
    ...   ...   ...
  </pattern>
</segment>
<segment name="Distribution">
  <pattern id="1">
    <process name="prepare receiving" role="Buyer">
      <input>
        <message type="Receiving Schedule" from="system"/>
        <message type="Receiving Report" from="Receiver"/>
      </input>
      <output>
        <message type="Receiving Schedule" to="Receiver"/>
        <message type="Confirmed Purchase" to="system"/>
      </output>
    </process>
    ...   ...   ...
  </pattern>
 ...   ...   ...
</segment>
...   ...   ...
```

Figure 3. Description of standard SCM model.

processes: "request order", "accept order", and "confirm order". Processes included in the system lanes are omitted since they can be hidden by common message exchange interfaces. Solid arrows represent the sequence of the processes. Dashed arrows represent message or datagram transmission. Processes in public flows exchange XML messages with processes of the other public flows. They may use other proprietary data models to exchange with processes in incorporate systems. In this pattern, "Order Request" data activates the "request order" process of the buyer. It invokes the ordering procedure to send "Order Request" message to the seller. This message activates "accept order" process of the seller. It invokes a reply procedure to send "Confirmed Order" message to the buyer. This message further activates "confirm order" process of the buyer. Table 1 shows input and output required by each segment. Note that they appear in the system lanes of each pattern and show the common interfaces with intra-corporate systems.

Pattern 2 of this segment omits "confirm order" process from pattern 1. It implies that a buyer can proceed to the shipping process without receiving order acknowledgement. Partners may choose the preferred pattern considering the properties of their business transactions, such as credit of partners and types of product. If a buyer orders commodity process foods and they are stably supplied by global food manufacturers, this process might be omissible. However, if a buyer orders agricultural products to a small farm, omission of the process might cause a shortage in stock.

Related business processes are not always linked with explicit message transmission. For example, pattern 1 (and also pattern 2) of the distribution segment does not have an explicit message exchange between "ship items" process and "receive items" process. They are not part of public flows and implicitly connected by physical operations for transferring products.

3.2 Collaboration Configuration

A collaboration configuration defines a specific public flow that partner companies should execute. The public flow is determined by choosing the transaction pattern from the standard SCM model.

A collaboration configuration includes role assignments and pattern selections. A role assignment binds each role to a specific company. Pattern selections are a list of patterns from which the partners agree to choose for each transaction segment. Each company in this collaboration executes the processes associated with the assigned role.

Table 1. Segment input and output.

Segment	Input (from)	Output (to)
Ordering	Order Request (BU)	Confirmed Order (BU) Confirmed Order (SE)
Distribution	Receiving Schedule (BU) Shipping Order (SE)	Confirmed Purchase (BU) Confirmed Sales (SE)
Settlement	Confirmed Purchase (BU) Confirmed Sales (SE)	Confirmed Payment (BU) Confirmed Payment (SE)

BU: Buyer, SE: Seller

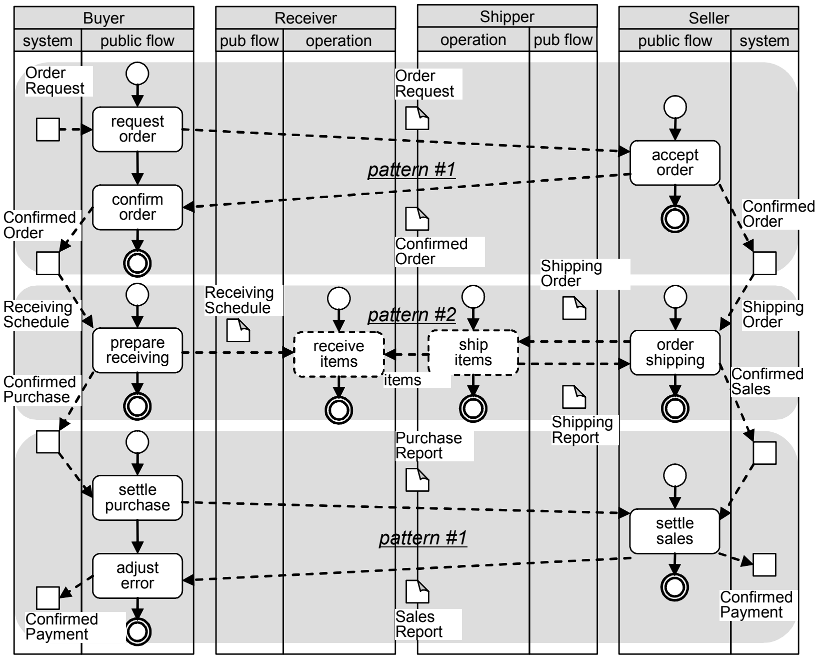


Figure 4. Determination of public flow.

Figure 4 shows an example of the collaboration configuration. By choosing pattern 1 from the ordering segment, pattern 2 from the distribution segment, and pattern 1 from the settlement segment, a specific public flow is defined.

4. DISTRIBUTED HUB ARCHITECTURE

This section introduces the SOA based architecture to simplify the development of B2B systems that support the diversified SCM standard.

4.1 Complexity Caused by Diversity

To join the collaboration based on the standard SCM model, each partner must prepare a system that executes the public flow specified by the collaboration configuration. For use of the standard to spread, it is necessary to clarify how to develop systems that conform to the standard.

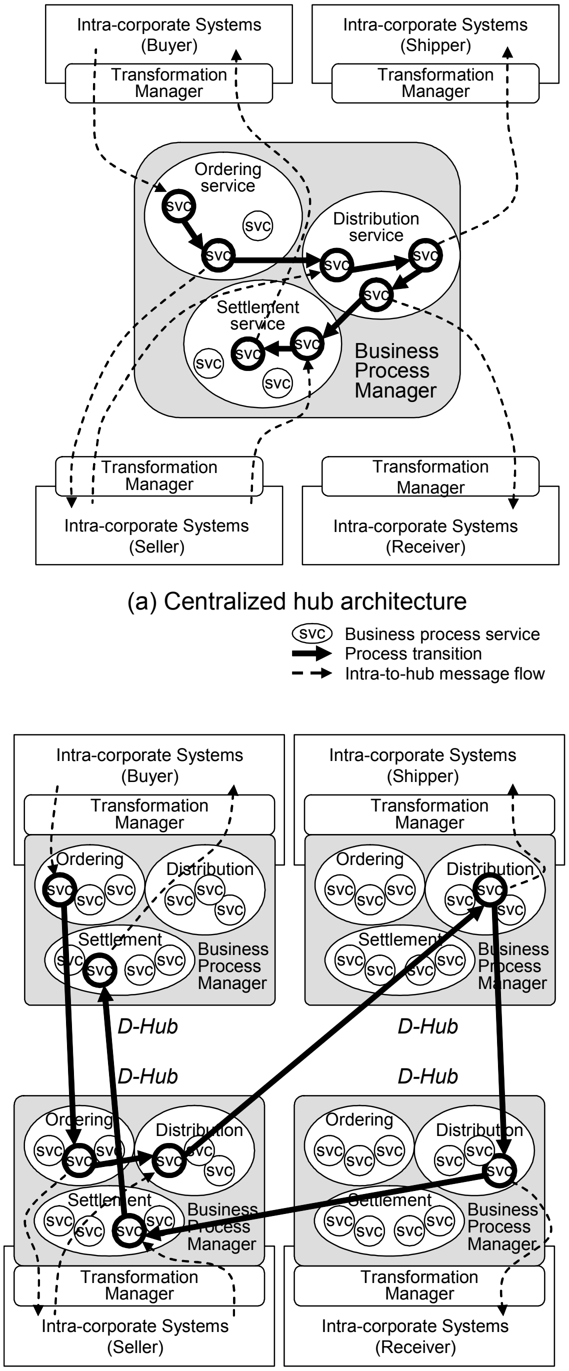
A standard SCM model may represent a large number of possible patterns, perhaps more than a thousand patterns. Moreover, each partner may have more than one collaboration configuration each of which tailored to partners and products. Therefore, each partner must support a large number of public flows. It is not easy for every player to implement such mechanisms.

4.2 Centralized Hub

Business processes defined in the transaction pattern are message-driven and loosely linked *via* input and output messages. They are naturally recognized as services in the context of SOA. Each transaction patterns can be represented as service combinations.

An intuitive architecture to realize the service combinations is the building of a centralized server that coordinates services according to the collaboration configurations as shown in Figure 5(a). As a hub, the centralized server holds all services associated with processes and transactions defined in the standard SCM model. The coordination can be described using languages, such as BPEL4WS. What each player must do is converted from their proprietary message formats to standard messages and sent to the central server.

Although this solution is technically feasible, the first and hardest issues to be resolved prior to starting B2B collaborations are who is responsible for both the costs and risks of running such a centralized server.


Intra-corporate Systems (Buyer)
Intra-corporate Systems (Shipper)
Transformation Manager
Ordering service
Distribution service
Settlement service
svc
Business Process Manager
Intra-corporate Systems (Seller)
Intra-corporate Systems (Receiver)
svc Business process service
Process transition
Intra-to-hub message flow
Ordering
Distribution
Settlement
D-Hub

(a) Centralized hub architecture

(b) Distributed hub architecture

Figure 5. Distributed hub architecture.

4.3 Distributed Hub

Here, we propose architecture named the distributed hub, which deploys distributed services to all partners. As a hub, the distributed service holds all services associated with processes and transactions defined in the standard SCM model. However, it does not function as a centralized service that coordinates participants. Instead, it is deployed on every participant and executes the assigned part of the supply chain collaboratively with other distributed hubs. Figure 5(b) shows the concepts of the distributed hub.

A distributed hub (*d-hub*) consists of two parts: transformation manager and business process manager. The transformation manager converts corporate proprietary message formats to standard message formats (typically defined using XML), and *visa versa*. The business process manager executes the transaction patterns chosen in the collaboration configurations.

A d-hub mediates intra-corporate systems and public flows. Each process included in a transaction is composed as a sequence of inbound services, which communicate with intra-corporate systems, and outbound services, which communicate with partners.

A d-hub is implementation-independent. It may use a native server that is developed to directly execute the transactions defined in the collaboration configuration. It may also use the generic business process management engine that executes standard languages, such as BPEL4WS.

5. EMPIRICAL STUDY ON THE JAPANESE RETAIL INDUSTRY

This section shows a standard SCM model for real supply chains in the Japanese retail industry and results of demonstration experiment using an experimental system based on the distributed hub architecture.

5.1 Standard SCM Model

5.1.1 Roles

For the standard SCM model from ordering to settlement, we found four roles: buyer, seller, shipper, and receiver.

A buyer buys products and pays the cost. Buyers include retailers, such as supermarkets, convenience stores, and drugstores.

A seller sells products and is paid the cost. Sellers include suppliers, such as manufacturers, wholesalers, and agricultural cooperatives.

A receiver receives products from a seller. Receivers include retailers' shops, wholesaler's DCs (distribution centers), and retailers' TCs (transfer centers). DCs and TCs are repositories that deliver products. They differ in that DCs have an operation for holding stock, while TCs do not.

A shipper ships products to a buyer. Shippers include retailers' TCs, wholesalers' DCs, and shared DCs, which are distributors that treat products of more than one retailer.

5.1.2 Segments

The standard SCM model is partitioned into 4 segments: ordering, distribution, purchase and sales settlement, and account settlement. The distribution segment is further partitioned into 6 segments:

Table 2. Segment definitions for the Japanese retail industry.

Segment		Descriptions	Input (from)	Output (to)	Patterns
Ordering		A buyer and a seller confirm order information, including items, amounts and delivery dates.	Order Request (BU)	Confirmed Order (BU) Confirmed Order (SE)	#1: Acceptance required #2: Acceptance not required #3: Recommended by Seller #4: None
Distribution	Shipping preparation	A shipper prepares shipping according to the shipping order.	Shipping Order (BU or SE)	Shipping Order (SH)	#1: Informed by Buyer #2: Informed by Seller #3: None
	Receiving preparation	A receiver prepares inspection of received items according to a receiving schedule.	Receiving Schedule (BU, SE, or SH)	Receiving Schedule (RE)	#1: Informed by Buyer #2: Informed by Seller #3: Informed by Shipper #4: None
	Shipping	A shipper ships the ordered items.	N/A [not public activity]	N/A [not public activity]	Picking patterns are irrelevant to public flow.
	Receiving	A receiver receives the ordered items and inspects them.	N/A [not public activity]	N/A [not public activity]	Inspection patterns are irrelevant to public flow.
	Confirming sales	A seller records the sales of the shipped items.	Shipping Report (SH or RE)	Confirmed Sales (SE)	#1: Shipping basis #2: Receiving basis #3: None
	Confirming purchase	A buyer records the purchase of the received items.	Receiving Report (SH or RE)	Confirmed Purchase (BU)	#1: Shipping basis #2: Receiving basis #3: None
Purchase and sales settlement		A seller and a buyer settle debts and credits by checking purchase and sales of daily transaction.	Confirmed Purchase (BU) Confirmed Sales (SE)	Confirmed Payment (BU) Confirmed Payment (SE)	#1: Purchase informed #2: Sales informed #3: Both #4: None
Account settlement		A seller and a buyer settle amounts of payment for transactions within a time frame.	Confirmed Payment (BU) Confirmed Payment (SE)	Confirmed Remittance (BU) Confirmed Invoice (SE)	#1: Remittance advice #2: Invoice #3: Both #4: None

BU: Buyer, RE: Receiver, SH: Shipper, SE: Seller

shipping preparation, receiving preparation, shipping, receiving, confirming sales, and confirming purchase.

Table 2 shows the descriptions of segments and patterns defined in each segment. As the specific public flows are determined by choosing a pattern from each segment, simple arithmetic indicates that there are thousands possible public flows.

5.1.3 Examples of Public Flows

Figure 7 and 8 show examples of public flows that were determined based on two different collaboration configurations. Both models are actually used in the Japanese retail industry.

In both cases, a retailer and a wholesaler play the roles of buyer and seller, respectively. The model in Figure 7 uses a wholesaler's DC as a shipper to deliver products to a retailer's TC as a receiver. In this model, to avoid delay caused by message exchange between DC and TC, the processes for reporting sales and purchases proceed independently. The model in Figure 8 uses a shared DC as a shipper to deliver products to a retailer's shop as a receiver. In this model, the retailer entrusts inspection of products to the shared DC, who directly distributes the products to each retailer's shop.

5.2 Experimental System

We have developed an experimental system based on a distributed hub architecture for applying to the real B2B environments. Figure 6 shows the common B2B system architecture for each player in collaboration. It consists of a business module (BM), data mapping framework, and ebXML communication server. They make up the framework for B2B communication necessary for the development of the diversified supply chain. Intra-corporate systems and adapters on the data mapping framework should be developed to fit the specific requirements of each company.

The business module is the central piece that controls message exchange according to the SCM standards as a d-hub. It deploys services associated with the process defined in the transaction patterns. Each service executes receiving and sending messages according to the collaboration configurations agreed with the partners. The business module plays a role in mediating intra-corporate systems and partners with the services.

To communicate with partners, the business module exchanges messages to/from the ebXML communication server. The J-XML format, which is an XML-based standard developed for the Japanese retail industry, is adopted as the format for message exchange with partners. The ebXML communication server sends/receives messages to/from the external partners using ebXML MS.

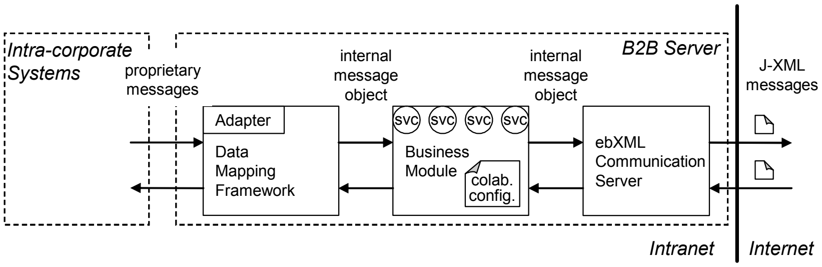


Figure 6. Experimental system architecture.

The business module communicates with existing intra-corporate systems *via* a data mapping framework. The data mapping framework has generic mapping functions to transform proprietary data formats from/to message objects used in the business module. The data mapping framework has an adapter architecture, which uses adapters designed to satisfy specific requirements for mapping proprietary corporate data.

5.3 Results of Application

The standard SCM model prepared showed that it had sufficient diversity to cover a wide variety of public flows used in the Japanese retail industry. It is a revised version of the model that was presented and approved in the standardization committee that is organized by Ministry of Economy, Trade and Industry and consists of major companies in the Japanese retail industry.

As a demonstration experiment, we applied the systems to a limited but real part of supply chains in two months. The experiment involved 8 companies including one of the largest general merchandise stores, one of the largest wholesalers, and major food manufacturers in Japan. As data, we used real everyday data exchanged by these players. The category of data was limited to groceries and did not include fresh foods. The data were processed in parallel with the existing EDI message exchange to avoid confusion in the real operation.

So far, the experimental system has shown sufficient abilities to exchange the data necessary for business transactions. Even though XML messages require more network bandwidth and processing power than the current fixed-length binary records, the transmission rate based on the XML message exchange via the Internet in the experiment exceeded the rate of the existing EDI system. Meanwhile, the range of message size rather than the size itself raised a new design requirement for time-out control. The message varied in size from a few hundred K byes to a hundred M bytes. Therefore usual small size messages were processed within a few second while large size messages required more than 10 minutes. To cope with this variety, time-out parameters should be dynamically determined in proportion to message sizes instead of using a fixed value.

In the experiment we developed supply chains based on the TC model that the retailer uses. Through the development of the system, we proved that the distributed hub architecture was feasible for conforming to a diversified standard. Due to the concept of SOA, the system encapsulated the complexity of public flows into services that provide a set of interfaces for intra-corporate systems. To date, as advantages of SOA or Web Services, sharing or finding services *via* the network has been emphasized. In contrast, the distributed hub architecture delivers the same services to every participant. This approach is useful for spreading sophisticated standard rapidly.

Our next step is to shift to full-scale operation from the current experiment phase. In full-scale operation, the number of partners will increase to more than 1000 and the categories of products will be widened to include fresh foods.

6. FUTURE WORK

This section discusses the issues remaining as future work.

6.1 Binding to Web Service Standards

For acceleration of the spread of industry standards, the definition of bindings to major standards becomes inevitable

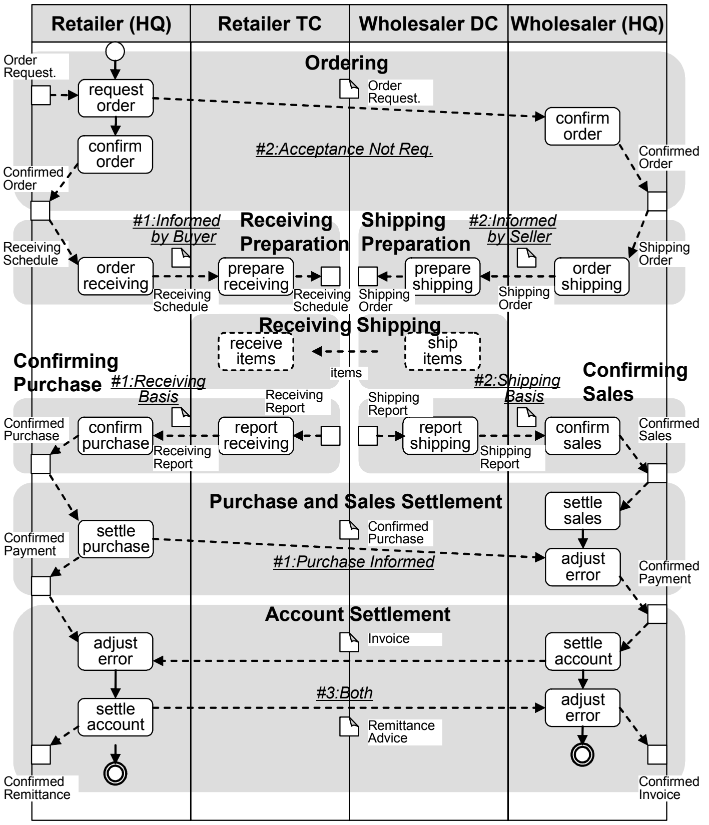


Figure 7. TC model.

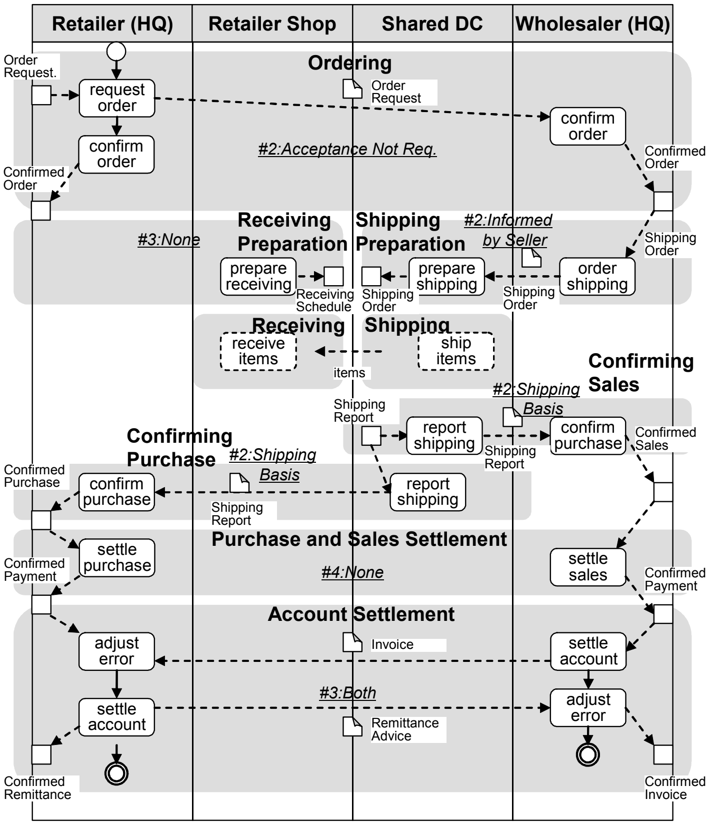


Figure 8. Shared DC model.

even if the proprietary XML descriptions are more concise. As mentioned above, we can define binding of the service invocation sequence for business transactions to BPEL4WS. We can also define bindings to ebXML BPSS, and CPPA. These bindings can lower the threshold for development of systems that conform to the diversified standard. By generating binding descriptions from the standard SCM model and a collaboration agreement, each company can use a generic EAI server that supports Web Service or ebXML standards.

6.2 Verification of Public Flows

As shown in the case analysis, a standard SCM model allows large numbers of possible public flows. Although the number of possible patterns may be increased, not all of the possible public flows make sense—some stop in the middle of the transaction, some form loops, and some are semantically useless even if topologically normal. It is a challenge to provide a mechanism that can automatically verify the collaboration configuration and show suggestions regarding which patterns should be used.

7. CONCLUSIONS

Although B2B collaboration has attracted a great deal of attention, B2B standards have not spread as rapidly as expected. Many industries are still hesitating to adopt standards for B2B collaboration as an adoption of the standard implies rebuilding of the business transactions with partners and paying costs for a new system to conform to the standard.

Essentially, a standard narrows the variation in each player's choice to reduce complexities and increases interoperability. Here, we presented an approach that widens the variation of player's choice but encapsulates the complexity using the SOA concept. We introduced a concept of a diversified SCM standard and distributed hub architecture for the development of B2B systems. This allows the development of SCM standards that are acceptable to various players with no drastic changes to the existing business transactions and minimizing the cost of development of new B2B collaboration systems.

Based on this concept, we analyzed the real business transactions used in the Japanese retail industry and developed an experimental system for applying to real supply chains. The results of the analysis proved that the diversified SCM standard could represent a sufficient variety of business transactions necessary for the industry. Through a demonstration experiment the system based on the distributed hub architecture could encapsulate the complexity of the diversified standard.

8. REFERENCES

[1] Business Process Modeling Notation Version 1.0, http://www.bpmn.org/.

[2] Benatallah, B., Dumas, M., Fauvet, M.-C. Rabhi, F. A., and Sheng, Q.Z., Overview of Some Patterns for Architecting and Managing Composite Web Services. in SIGecom Exchanges, vol. 3, no. 3, (August 2002), 9-16.

[3] Business Process Execution Language for Web Services Version 1.1. http://www-128.ibm.com/developerworks/library/ws-bpel/.

[4] Chafle, G., Chandra, S., and Mann, V. Decentralized Orchestration of Composite Web Services. in WWW2004 (New York, May 2004), 134-143.

[5] Damodaran, S. B2B Integration over the Internet with XML - RosettaNet Success and Challenges. in WWW2004 (New York, May 2004), 188-195.

[6] ebXML Technical Specifications, http://ebxml.org/specs/#technical_specifications.

[7] Huang, Y., and Chung, J.-Y. A Web Services-Based Framework for Business Integration Solutions. Electronic Commerce Research and Applications, vol. 2, issue 1 (2003), 15-26.

[8] Khalaf, R., Mukhi, N, and Weerawarana, S. Service-Oriented Composition in BPEL4WS. in WWW2003 (Budapest May 2003).

[9] Kim, J. B., Segev, A. Patankar, A., and Cho, M. G. Web Services and BPEL4WS for Dynamic eBusiness Negotiation Processes. in the 1st International Conference on Web Services (ICWS '03), (Las Vegas, 2003).

[10] RosettaNet Standards, http://www.rosettanet.org/standards.