

A Voucher-Integrated Trading Model for C2B and C2C E-Commerce System Development

Makoto Iguchi, Masayuki Terada, Yoshitaka Nakamura, and Ko Fujimura

NTT Information Sharing Platform Laboratories, NTT Corporation

Abstract: This paper presents a trading model that aims to cover a wide variety of C2B and C2C business schemes. The proposed model uses a digital data entity called voucher and a voucher trading system (VTS) to construct a versatile trading framework. The trading framework handles the delivery/payment phases of trading transactions, and this makes it possible to construct various C2B and C2C e-commerce systems by simply combining the suitable matching logic implementations on our unified trading framework.

Key words: E-commerce, Consumer-to-Consumer (C2C) trading, Consumer-to-Business (C2B) trading, Voucher, Voucher Trading System (VTS)

1. INTRODUCTION

The introduction of the Internet has changed the way consumers handle commerce transactions. By providing an interactive communication medium, the Internet lets consumers become more deeply involved in commerce transactions than ever before. Compared to traditional commerce flows in which suppliers play the leading role, e-commerce flows let consumers take the lead. Consumer-initiated business schemes, namely Consumer-to-Consumer (C2C) trading schemes and Consumer-to-Business (C2B) trading schemes, have been taking off over the past few years and are expected to thrive in the next few years; trading schemes in the e-commerce field include auctions, reverse-auctions, and group purchasing.

Many such business schemes require consumers and suppliers to interact in a complex manner, and thus a tool for constructing truly supportive e-commerce systems will soon become essential. Many EC construction tools are currently available for implementing Web-based e-commerce systems

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35617-4_48](https://doi.org/10.1007/978-0-387-35617-4_48)

J. L. Monteiro et al. (eds.), *Towards the Knowledge Society*

© IFIP International Federation for Information Processing 2003

such as INTERSHOP Enfinity (Intershop Enfinity White Paper), IBM Websphere Commerce Suite, and Microsoft Commerce Server. These tools make it relatively easy to build e-commerce systems that allow integrated matching/payment/delivery transactions to be realized on the server and provide “look and feel” GUIs, usually for Web browsing, on the client side. However, the tools are basically designed for constructing supplier-initiated trading systems, or were designed for specific consumer-initiated trading systems. Thus, developers have to construct systems from scratch if they want to create consumer-initiated trading e-commerce systems.

In this paper, we present a trading model with a versatile trading framework that can be used to realize a wide range of C2B and C2C e-commerce systems. Section 2 overviews our trading model. Section 3 discusses our trading model in detail by examining the technical structures of the model; examples are given of implementing C2B and C2C e-commerce systems on our trading model. We introduce a prototype implementation of our model in Section 4. Related works are discussed in Section 5.

2. OVERVIEW OF THE VOUCHER-INTEGRATED TRADING MODEL

There exists a great number of C2B and C2C business schemes. A close appraisal at each of the matching, payment, and delivery phases, reveals that the diversities of C2B and C2C business schemes can be categorized as follows:

- Diversity of matching logics:
Taking just auction logic as an example, the varieties include open bid auctions (English auction and Dutch auction) and sealed bid auctions (first price auction and second price auction).
- Diversity of trading objects:
The merchandise may be tangible goods, intangible goods, or the provision of services.
- Diversity of delivery and payment methods:
The delivery may be achieved by courier services (if the object is tangible) or by downloading via the Internet (if the object is digital). Payment may be achieved by credit card clearance, money orders, or electronic money.
- Diversity of trading agreement/contract:
Some schemes require the participants to define the terms of compensation and terms of responsibilities to cover the case when the atomicity of the delivery and payment fails.

Our goal is to design a trading model that can cover a wide range of business schemes that exhibit these diversities. The key to this model is to our perception that the basic business scheme has two phases: a matching phase and a delivery/payment phase. The negotiation to establish the trade, such as which objects are to be traded, which participants are to be engaged in trading, and how trading is to be conducted, is made in the matching phase. The delivery/payment phase conducts the exchange of designated objects settled at the matching phase.

Let us assume that the following two procedures exist:

- 1) A procedure to create a digital data (voucher) that is capable of representing a wide variety of merchandise (tangible, intangible, or the provision of services) and monetary values: This procedure guarantees the conversion of the voucher to the corresponding object.
- 2) A procedure to conduct a fair exchange of such vouchers: Here, a fair exchange means that voucher exchange is performed so as to achieve an “all-or-none” result; the exchange is conducted in such a way that at the end of the exchange, both or neither of the participants have received the vouchers expected. This procedure guarantees that the exchange is conducted if and only if a user shows an intention to engage in the exchange.

The first procedure lets us handle the diversities of trading objects and delivery/payment methods. The delivery of merchandise now becomes the transfer of a voucher, representing the merchandise, from the supplier to the consumer over the Internet. In the same way, payment now becomes the transfer of a voucher, representing the positive evidence of payment, from the consumer to the supplier over the Internet. This allows various delivery and payment methods to be modelled as the transfer of the vouchers over the Internet.

The second procedure let us handle the diversity of trading contracts/agreements. If such an exchange exists, the atomicity of delivery and payment transactions can then be realized by the fair exchange of one voucher, representing the merchandise, for another voucher, representing the monetary value. Thus, once the procedure is realized, the agreement only has to define which vouchers are to be exchanged. The exchange of vouchers itself is guaranteed by the exchange protocol, so complex contracts defining terms of compensation and responsibility in the event of failure of delivery and/or payment transactions are no longer necessary.

Figure 1 shows the trading model that can be achieved by integrating the procedures described above. The model separates the matching phase from the delivery/payment phase. In the matching phase, the participants are matched following the appropriate market coordination requirements given by the intended business scheme. An agreement stating the result of the

matching process (basically defining which vouchers are to be exchanged between which participants) is generated at the end of the matching phase. The delivery/payment phase conducts a fair exchange of the vouchers as defined by the agreement that represents the participants' intention.

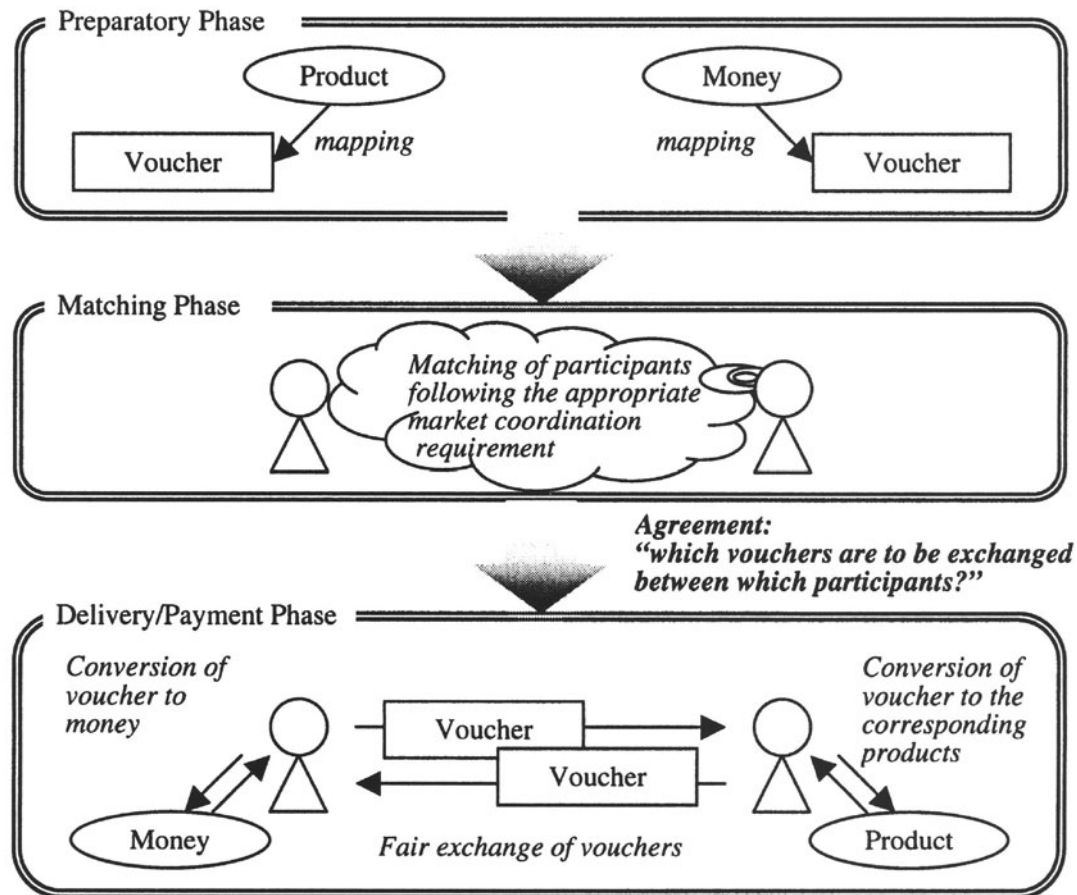


Figure 1: Overview of voucher-integrated trading model

In the model, the delivery/payment phase acts as a unified trading framework for C2B and C2C trading. Once this framework is implemented, developers can build C2B and C2C e-commerce systems by combining the desired matching phase implementations on the framework.

3. THE BUILDING BLOCKS AND THE DETAILED DISCUSSIONS

After having overviewed the proposed trading model and the requirements that should be solved, we now elucidate the trading model in

more detail by introducing the technical structures of the model and by showing some implementation options.

3.1 Building blocks of the voucher-integrated trading model

Figure 2 shows the building blocks of our trading model. In the figure, the modules surrounded by the dotted line form a trading framework.

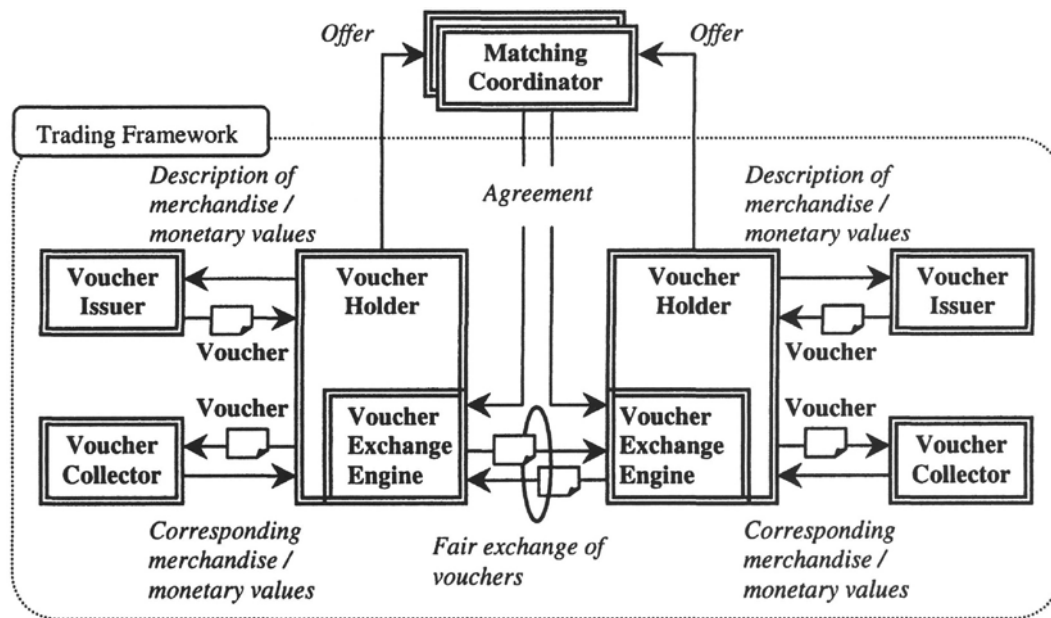


Figure 2: Architecture of the voucher-integrated trading model

Each building block is explained as follows:

Voucher:

A voucher is a digital representation of a right to claim goods or services. A voucher is composed of a promise (a description of the corresponding merchandise or monetary value) and issuer information (a description of the voucher issuer). The conversion of a voucher into the merchandise or monetary value described in the promise is to be guaranteed by the issuer specified by the issuer information.

Voucher Holder:

A Voucher Holder stores vouchers. Each participant possesses at least one Voucher Holder. The Voucher Holder can be implemented in various ways, such as a smartcard or a database located on a server.

Voucher Issuer:

A Voucher Issuer creates the corresponding voucher upon receiving information of the merchandise or monetary value from participants as input. The voucher generated by the Voucher Issuer has the issuer information that specifies the Voucher Issuer himself. Since each voucher issuer guarantees the conversion of vouchers into the corresponding merchandise and services, Voucher Issuers should only create vouchers if and only if he agrees to guarantee such conversion. Because of this reason, the Voucher Issuer is to be maintained by the appropriate entity. For instance, courier service providers should maintain a Voucher Issuer if the Voucher Issuer is to create vouchers representing claim tags. (In this case, the Voucher Issuer received the actual merchandise itself instead of its description upon issuing the voucher).

Matching Coordinator:

The Matching Coordinator receives offers from Voucher Holders. An offer describes how the Voucher Holder intends to engage in the trading. An offer contains the following information:

- Conditions of voucher(s) to be given to the trading partner's Voucher Holder.
- Conditions of voucher(s) to be taken from the trading partner's Voucher Holder.
- Other conditions, such as the expiration date of the trade [optional].

Taking offers from Voucher Holders as input, the Matching Coordinator matches the offers according to dedicated matching logic and forms an agreement if possible.

An agreement contains the following information:

- Vouchers to be exchanged.
- Voucher Holders taking part in the exchange.
- Other conditions, such as the expiration date of the trade [optional].

The agreement is sent to the appropriate Voucher Exchange Engine to commence the exchange of the vouchers. It is desirable to create the agreement in machine-readable form such as XML documents.

The Matching Coordinator is a customization point in our trading model. Developers can construct the Matching Coordinator with the proper matching logic that suits their business requirements. Several examples of the Matching Coordinator are shown in Section 3.3.

Voucher Exchange Engine:

Upon receiving the agreement from the Matching Coordinator, the appropriate Voucher Exchange Engines perform the fair exchange of vouchers according to the conditions specified in the agreement. Prior to the exchange, each Voucher Exchange Engine checks to see if the agreement reflects the description of the participant's offer, that is, the intention of the participant.

Voucher Collector:

A Voucher Collector converts the vouchers into the corresponding merchandise and monetary values. The Voucher Collector can be maintained by the same entity that maintains the Voucher Issuer or by a different entity. If the entities differ, the Voucher Collector should only process the vouchers issued by the Voucher Issuers trusted by the Voucher Collector; This verification allows the Voucher Collector to be ensured that the conversion of the voucher to the corresponding merchandise and monetary values will be conducted safely. Examples of a Voucher Collector include a courier service that accepts a delivery order upon receiving a voucher, and amusement park gate systems that punch the vouchers issued by the trustworthy Voucher Issuer (e.g. ticket agency) upon entering the park.

Note than in Figure 2, the Voucher Collector to the right handles a voucher issued by the Voucher Issuer to the left, and vice versa.

3.2 Requirements

For the trading models in Figure 2 to be effective, the following requirements should be met:

- The forgery, alteration, or reproduction of vouchers should be prevented.
- The exchange of vouchers should be conducted while ensuring fairness.
- The voucher promise descriptions should be flexible enough to cover an adequate range of goods and services.

The concepts of vouchers and VTS (Voucher Trading System) (Fujimura, et al, 2001) can be used to fulfil the first requirement. VTS is a technology that is currently being actively discussed in the IETF trade WG (Fujimura, 2001 and Terada, Fujimura, 2002), and we are developing VTS specifications in IETF so as to establish voucher trading as the infrastructure of E-commerce. VTS provides a means to circulate issuer-generated vouchers among users while preventing the forgery, alteration, or reproduction of vouchers. Some of the protocols that can be applied to achieve this security level include FlexToken (Terada et al, 2000), ticket-Token (Matsuyama et al, 1999), and SOX (Howland).

The various protocols proposed for realizing the fair exchange of electronic data (Zhou, 2001 and Lacoste et al, 2000 and Asokan, 1998) can be applied to achieve the second requirement. One approach is to apply fair exchange techniques with an offline Trusted Third Party (TTP). By applying these types of protocols to the VTS, the fair exchange of vouchers in a decentralized environment can be realized without active TTP intervention. An alternative is to apply fair exchange protocols with an online/inline TTP. Though these types of protocols require the active intervention of the TTP in every voucher exchange transaction, the approach may be still suitable for services within a centralized environment (e.g. services in which vouchers are stored in a central server).

The third requirement can be solved by defining the promise in XML. Several methods for defining diverse types of goods and services in XML have been proposed (Fujimura et al, 2002), and they can be applied for defining vouchers.

3.3 Implementation Flexibilities

As shown in Figure 2, our trading model isolates the Matching Coordinator from the voucher-driven trading framework. This enables developers to integrate their desired matching logic into the Matching Coordinator, and so develop the e-commerce systems that suit their requirements. We demonstrate this by describing several business scenarios:

- 1) “Goods for sale” scenario:
Participant X (seller) sets a fixed price (e.g. \$100) on the merchandise and advertises his/her intention to sell the merchandise at this price. Later, participant Y (buyer) accepts this offer which initiates the trade.
- 2) “Goods wanted” scenario:
Participant X (buyer) sets a fixed price (e.g. \$100) on the merchandise and advertises his/her intention to buy the merchandise at this price. Later, participant Y (seller) accepts this offer which initiates the trade.
- 3) “Limit ordering” scenario:
Participant X indicates an intention to sell the merchandise at a specified price (\$100). Participant Y indicates his/her intention to buy the merchandise at the specified price (\$100). Later, their intentions are matched.
- 4) “Barter trading” scenario:
Participant X and Y each indicate their desire to exchange merchandise. The actual trade is initiated if two participants have requests that “fit”.
- 5) Auction scenario:
Participant X (seller) sets a minimum price (e.g. \$50) on the merchandise. Buyers bid on it, and the one who bids the maximum price

(e.g. \$100), Participant Y, commences trading with Participant X.

6) Reverse-auction scenario:

Participant X (buyer) sets a maximum price (e.g. \$200) on merchandise. Sellers bid for it, and the one who bids the minimum price (e.g. s\$100), Participant Y, commences trading with the buyer.

Preparatory phase:

In the preparatory phase, participants prepare vouchers using the Voucher Issuer.

For all scenarios, a seller triggers the issuance of a voucher representing the merchandise for sale by sending their descriptions to the Voucher Issuer. Likewise, a buyer triggers the issuance of a voucher representing the monetary values to be paid (\$100 worth of money in this case). Let us call the voucher representing the merchandise *Voucher A*, and the voucher representing \$100 *Voucher B*.

For all scenarios, the vouchers issued are stored in each participant's Voucher Holder.

Matching phase:

In the matching phase, the Matching Coordinator receives offers from participants and matches the offers according to the appropriate coordination requirements. The procedures used to achieve this coordination vary according to the business schemes.

“Goods for sale” & “Goods wanted” scenario:

In the “goods for sale” scenario, the Matching Coordinator receives an offer containing the following information from Participant X:

“I will give Voucher A in exchange for Voucher B.”

The offer is then advertised by the Matching Coordinator. After Participant Y finds and accepts the offer, the following agreement is made and sent to the Voucher Exchange Engine:

“Participant X will give Voucher A to Participant Y in exchange for taking Voucher B from Participant Y.”

In the “goods wanted” scenario, the same procedures are followed, except that *Voucher A* and *Voucher B* in the offer and the agreement are switched.

“Limit ordering” & “Barter trading” scenario:

In the “Limit Ordering” scenario, both Participant X and Y send offers to the Matching Coordinator. The contents of the offers are as follows, respectively:

“I will give Voucher A in exchange for Voucher B.”

“I will give Voucher B in exchange for Voucher A.”

The Matching Coordinator matches these offers and creates the agreement:

“Participant X will give Voucher A to Participant Y in exchange for taking Voucher B from Participant Y.”

In the “Barter Trading” scenario, the same procedure is followed, except that *Voucher B* in the offers and the agreement is replaced with *Voucher A*, a voucher representing merchandise to be provided from Participant Y to Participant X.

“Auction” & “Reverse auction” scenario:

In the auction scenario, the following offer is sent from Participant X:

“I will give Voucher A in exchange for taking a Voucher that represents \$50 or higher worth of money.”

An offer sent from Participant Y is as follows:

“I will give Voucher B in exchange for taking Voucher A.”

The Matching Coordinator, after conducting an auction and recognizing Participant Y as a winner, creates the agreement:

“Participant X will give Voucher A to Participant Y in exchange for taking Voucher B from Participant Y.”

In the reverse auction scenario, the following offer is sent from Participant X:

“I will give a Voucher that represents \$200 or smaller amount of money in exchange for taking Voucher A.”

An offer sent from Participant Y is as follows:

“I will give Voucher A in exchange for taking Voucher B.”

The Matching Coordinator, after conducting an auction and recognizing Participant Y as a winner, creates the agreement:

“Participant X will give Voucher B to Participant Y in exchange for taking Voucher A from Participant Y.”

The offers and agreements generated in each scenario are summarized in Table 1. Note that for all scenarios, the role of the Matching Coordinator is fixed; the Matching Coordinator receives offers from participants, matches the offers, and creates an agreement accordingly for all scenarios. It is how

the Matching Coordinator performs this offer matching that makes each scenario unique.

The agreement is a simple description defining vouchers to be exchanged and participants engaged in the exchange, so it can be easily defined as an XML document. An example of the agreement in the “goods for sales” scenario is shown in Section 4.2.

Scenario	Offer from Participant X	Offer from Participant Y	Agreement formed by Coordinator
(1)	<i>GIVE: Voucher A</i> <i>TAKE: Voucher B</i>	<i>(Accept Participant X's offer)</i>	<i>X -> Y: Voucher A</i> <i>Y -> X: Voucher B</i>
(2)	<i>GIVE: Voucher B</i> <i>TAKE: Voucher A</i>	<i>(Accept Participant X's offer)</i>	<i>X -> Y: Voucher B</i> <i>Y -> X: Voucher A</i>
(3)	<i>GIVE: Voucher A</i> <i>TAKE: Voucher B</i>	<i>GIVE: Voucher B</i> <i>TAKE: Voucher A</i>	<i>X -> Y: Voucher A</i> <i>Y -> X: Voucher B</i>
(4)	<i>GIVE: Voucher A</i> <i>TAKE: Voucher A'</i>	<i>GIVE: Voucher A'</i> <i>TAKE: Voucher A</i>	<i>X -> Y: Voucher A</i> <i>Y -> X: Voucher A'</i>
(5)	<i>GIVE: Voucher A</i> <i>TAKE: Voucher (\$50 or higher)</i>	<i>GIVE: Voucher B</i> <i>TAKE: Voucher A</i>	<i>X -> Y: Voucher A</i> <i>Y -> X: Voucher B</i>
(6)	<i>GIVE: Voucher (\$200 or lower)</i> <i>TAKE: Voucher A</i>	<i>GIVE: Voucher A</i> <i>TAKE: Voucher B</i>	<i>X -> Y: Voucher B</i> <i>Y -> A: Voucher A</i>

Table 1: Offers and agreements for each business scenario

Delivery/payment phase:

After receiving the agreement from the Matching Coordinator, the Voucher Exchange Engine confirms the fulfillment of the conditions defined in the agreement, and the correspondence of the agreement to the offer created by the Voucher Holder. Once confirmed, the Engine conducts the exchange of designated vouchers between the appropriate Voucher Holders. The exchange is achieved while ensuring fairness, so developers do not have to consider restoration from unfair events; e.g. the supplier receives a voucher while the consumer does not. The exchange is conducted as defined in the agreement, thereby providing a versatile trading framework for all scenarios.

After the exchange of the vouchers is finished, the vouchers are sent to the Voucher Collector. The Voucher Collector confirms the validity of the vouchers and the issuer of the vouchers. The Voucher Collector provides corresponding merchandise or monetary values to the participants if the vouchers are valid and issued by the trustworthy issuers.

4. PROTOTYPE IMPLEMENTATION OF THE VOUCHER-INTEGRATED TRADING MODEL

To prove the feasibility of our trading model, we implemented a prototype system.

4.1 Overview of the prototype

The prototype system, implemented using Java, provides a GUI interface to participants and consists of the following building blocks:

Voucher Issuer, Voucher Holder, and Voucher Collector:

These building blocks are implemented so as to comply with the VTS proposed in the IETF trade Working Group (Terada et al, 2002). The prototype uses XML Voucher (Fujimura et al, 2002) and FlexToken (Terada et al, 2000) as its underlying technologies. XML Voucher is used for defining vouchers, and FlexToken is used to realize the functions (e.g. issuing, transferring, and redemption of vouchers) required for implementing the voucher-integrated trading framework.

Voucher Exchange Engine:

The prototype realizes the fair exchange of vouchers by using an escrow technique. Prior to initiating an exchange of vouchers, both participants upload their vouchers to remote Voucher Holders on the escrow server. Once both participants have entered the vouchers defined in the agreement, the vouchers are swapped on the server. Later, each participant downloads the swapped vouchers to his/her Voucher holder.

Matching Coordinator:

The Matching Coordinator used in our prototype supports the “goods for sales” and “goods wanted” scenario that are described in Section 3.4.

4.2 Screenshots

Figure 3 is a screenshot of our prototype system that demonstrates the “goods for sales” scenario; Participant Y is confirming and accepting an offer (Participant Y is to buy a concert ticket for the price of 6,000 yen).

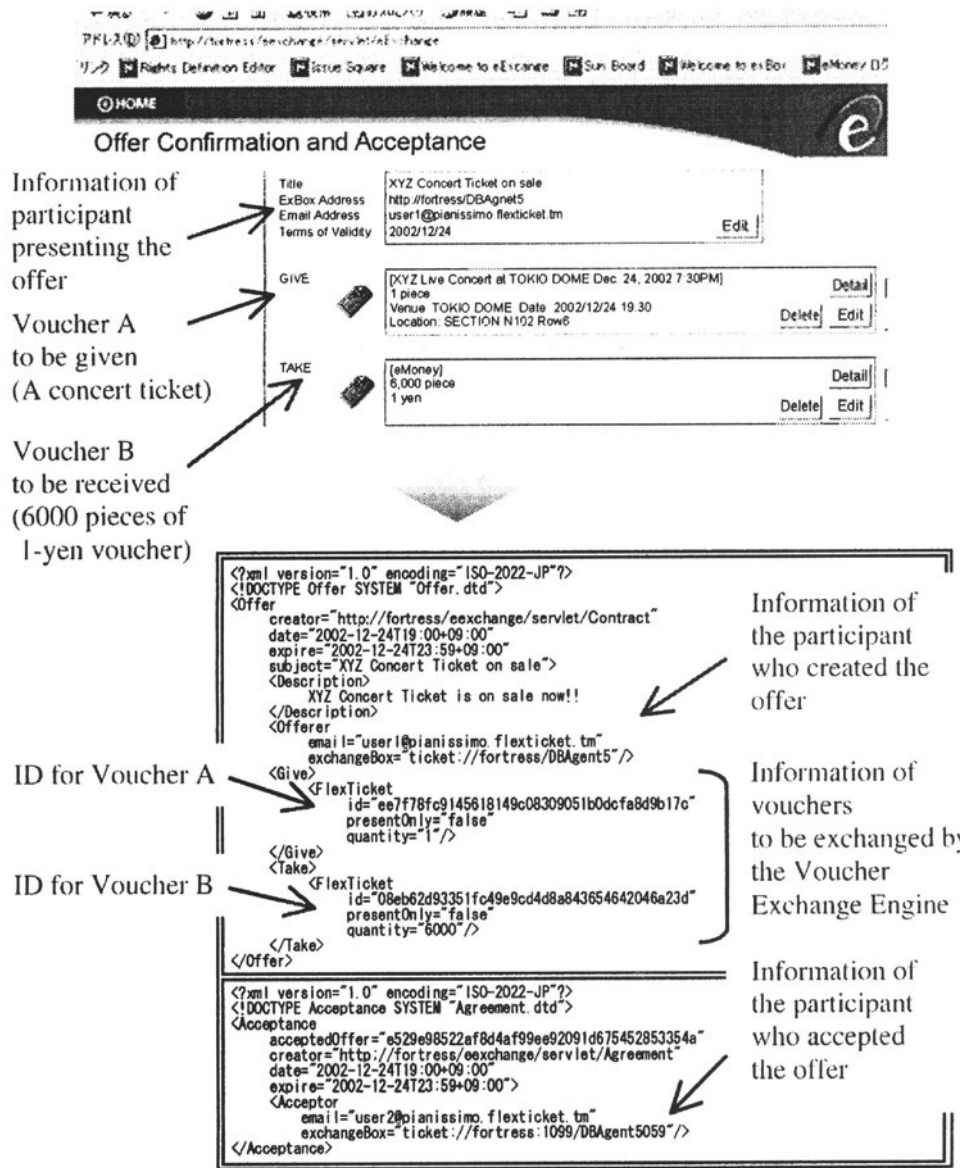


Figure 3. Screenshot of our prototype

The XML documents in Figure 3 represents the corresponding agreement formed upon offer acceptance. The agreement is passed to the Voucher Exchange Engine which executes a fair exchange of vouchers as described in the agreement.

5. RELATED WORKS

The model proposed by (Ketchpel et al, 1997) sets an external shopping controller that controls business workflow. The state information maintained on the shopping controller, together with the Shopping Transaction Record

(STR) exchanged among participants, realizes a highly flexible e-commerce framework to which external payment and delivery mechanisms can be added easily. However, the model requires the exchange of STR for each transaction conducted. This incurs excessive communication costs and delays. Moreover, the model does not explicitly provide security.

WebXice (Wombacher et al, 2001) was designed to provide an e-commerce architecture that allows the maximum reuse of legacy infrastructures such as payment and authentication mechanisms. It extends the ICE protocol (Webber et al), a communication protocol for exchanging information between information providers and consumers, to realize flexible e-commerce systems with a thin-client implementation approach. The basic strategy taken in the model is to provide efficient message logging so that consumers can collect and manage non-refutable proofs, but how the collected proofs would be actually used in the case of disputes was not discussed. The target of (Wombacher et al, 2001) was the trading of information goods, such as news, software, or reports, so the model has limited coverage.

SEMPER (Secure Electronic Marketplace for Europe) (Lacoste et al, 2000) is an open architecture for secure electronic commerce. The architecture focuses on providing secure trading through the use of fair exchange protocols while not sacrificing trading practicality as a whole. The study describes extensive research on how to achieve the fair exchange of electronic data. We believe that some protocols, an optimistic fair exchange protocol for example, are effective in realizing our payment/delivery phase in a distributed environment without requiring the active intervention of a TTP (e.g. an escrow server).

6. CONCLUSION

This paper proposed a trading model suitable for implementing the widest possible variety of consumer-initiated C2B and C2C businesses schemes. The idea of mapping diverse goods and monetary values onto vouchers, and the idea of simplifying diverse payment/delivery schemes into the fair exchange of corresponding vouchers, are used to form a versatile trading framework. Once this framework is built, developers can construct C2B and C2C e-commerce systems simply by combining their matching modules on this framework. This flexibility of our trading model was shown by giving several business scenarios as examples and by showing that these scenarios can be successfully implemented using our model.

The security of voucher trading in the model is guaranteed by adopting the FlexToken technology. This technology ensures that vouchers cannot be altered, forged, or reproduced during their circulation. (We did not examine

security issues in detail in this paper, see (Terada et al, 2000) for more information.) The fair exchange of vouchers can be achieved by adopting various fair exchange protocols. The fair voucher exchange in distributed environment can be realized by adopting fair exchange protocols with offline TTP, while the fair exchange in centralized environment can be realized by adopting fair exchange protocols with inline/online TTP.

7. REFERENCES

- Intershop Enfinity White Paper*, <http://www.intershop.com/pdf/products/enfinity/enfinity.pdf>.
IBM Websphere Commerce Suite,
<http://www.ibm.com/software/webservers/commerce/servers/index.html>.
Microsoft Commerce Server, <http://www.microsoft.com/commerceserver/>.
- K. Fujimura and M. Terada, "Trading Among Untrusted Partners via Voucher Trading System," 1st IFIP Conference on e-Commerce, e-Business, and e-Government (IFIP I3E), 2001.
- K. Fujimura, "Requirements for Genetic Voucher Trading," IETF Internet Draft, <http://www.ietf.org/internet-drafts/draft-ietf-trade-drt-requirements-04.txt>, 2002.
- M. Terada, and K. Fujimura, "Voucher Trading System Application Programming Interface (VTS-API)," IETF Internet Draft, <http://www.ietf.org/internet-drafts/draft-ietf-trade-voucher-vtsapi-02.txt>, 2002.
- M. Terada, H. Kuno, M. Hanadate, and K. Fujimura, "Copy Prevention Scheme for Right Trading Infrastructure," 4th Smart Card Research and Advanced Application Conference (CARDIS), 2000.
- K. Matsuyama and K. Fujimura, "Distributed Digital-Ticket Management for Rights Trading System," 1st ACM Conference on Electronic Commerce, pp. 110-118, 1999.
- G. Howland, "Developing of an Open and Flexible Payment System," Systemics Ltd., <http://www.systemics.com/docs/sox/overview.html>.
- J. Zhou, "Non-repudiation in Electronic Commerce," Artech House, Inc., 2001.
- G. Lacoste, B. Pfitzmann, M. Steiner, and M. Waidner (ed), "SEMPER: Secure Electronic Marketplace for Europe," Springer-Verlag, Berlin, 2000.
- N. Asokan, "Fairness in electronic commerce," PhD thesis, University of Waterloo, May 1998.
- K. Fujimura and M. Terada, "XML Voucher: Generic Voucher Language," IETF Internet Draft, <http://www.ietf.org/internet-drafts/draft-ietf-trade-voucher-lang-03.txt>, 2002.
- S. Ketchpel, H. Garcia-Molina, and A. Paepcke, "Shopping Models: A Flexible Architecture for Information Commerce," In Proc. of 2nd ACM International Conference on Digital Libraries, 1997.
- A. Wombacher, P. Kostaki, and K. Aberer, "webXice: an Infrastructure for Information Commerce on the WWW," In Proc. of HICSS, IEEE Press, 2001.
- N. Webber, C. O'Connell, B. Hunt, R. Levine, L. Popkin, and G. Larose, "The Information and Content Exchange (ICE) Protocol," <http://www.w3.org/TR/NOTE-ice/>.