Integrating Radio Frequency Identification (RFID) data with Electronic Data Interchange (EDI) business processes

by

Yan Chen

Bachelor of Electrical Engineering, National University of Singapore
Master of Business and Administration, National University of Singapore

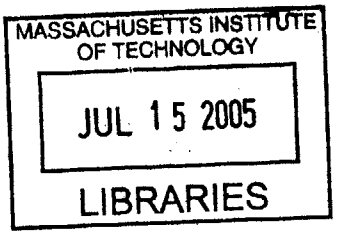
Submitted to the Engineering Systems Division in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Logistics

at the

Massachusetts Institute of Technology

June 2005



Signature of Author .....
.....
gineering Systems Division
May 6, 2005

Certified by .....
.....
David L. Brock
Principle Research Scientist, MIT
Thesis Supervisor

Accepted by .....
.....
Yossi Sheffi
Professor of Civil and Environmental Engineering
Professor of Engineering Systems
Director, MIT Center for Transportation and Logistics

BARKER

Integrating Radio Frequency Identification (RFID) data with Electronic Data Interchange (EDI) business processes

by

Yan Chen

Submitted to the Engineering Systems Division
on 6th May 2005 in Partial Fulfillment of the
Requirements for the Degree of
Master of Engineering in Logistics

Abstract

Radio Frequency Identification (RFID) technology, an important component in the enterprise IT infrastructure, must be integrated into the legacy IT system. This thesis studies how RFID technology can be integrated into the existing Electronic Data Interchange (EDI) infrastructure, particularly how RFID can be used in the current EDI exchange process to accelerate the receiving process. After detailed review of current receiving process and relevant data specification, the author finds it possible to replace the current manual receiving process by RFID enabled automatic receiving and reconciliation process. A middleware is proposed to implement this approach.

Thesis Supervisor: David L. Brock
Title: Principle Research Scientist, MIT

Acknowledgements

I would like to thank my thesis supervisor Dr. David L. Brock at MIT Auto-ID lab for the time and effort he has devoted in guiding me in the writing of this thesis. His guidance, resource and insight made this thesis project a fun and fulfilling learning experience.

I would like to thank Stephen Miles, Edmund W. Shuster and Tatsuya Inaba at MIT Auto-ID lab for their consistent support and advice.

Dedication

This thesis is dedicated to my parents (Chen Xueji and Pan Wenqing), my sister (Chen Li) and Celena Yew. Without their continuous support, I would not have accomplished my program at MIT.

Biographical Note

Yan Chen is currently a candidate for Master of Engineering in Logistics at MIT. Prior to MIT, he worked as a Senior Engineer in Agilent Technologies, Singapore. His work at Agilent involved developing Agilent's latest generation handheld Gigabit Ethernet network tester. He has experiences in product development, technical marketing, procurement and manufacturing. Prior to Agilent Technologies, Yan Chen worked as a Design Engineer for Pixelmetrix Corporation developing world's first real-time Digital Video Broadcasting operation monitoring equipment. He completed his MBA and bachelor degree in Electrical Engineering from National University of Singapore. Yan Chen is an active member of IEEE Communication Society, and he is also a member of APICS.

Table of Contents

List of Tables

List of Figures

1 Introduction

This thesis describes the integration of Radio Frequency Identification (RFID) data with existing business process transactions. It shows how RFID can be integrated into existing Electronic Data Interchange (EDI) transactions in the automatic reconciliation process. This thesis answers one of many important questions related to the management of RFID data.

Radio Frequency Identification (RFID) is a technology that promises to increase visibility in the supply chain. By tagging products, companies are able to visually track product movement along the supply chain. With such information, companies can make better decisions on how much to manufacture, how much to put into inventory and how much to ship to the customer.

Various industries are deploying RFID, including retail, fashion, semiconductor, food and drug industry. Wal-Mart, the largest retailer chain in the world, has set 2005 as the deadline for its top 100 suppliers to use RFID tags on all shipment to its distribution center [1]. The Department of Defense (DoD) mandated RFID tagging to take effect on Jan 1, 2005 for selected distribution locations and for specific classes of goods. The tagging will expand to a wider range of goods and distribution sites in the next few years [2]. Finally, the Food and Drug Administration (FDA) is promoting RFID tags for tracking prescription drugs [3].

With so many industry mandates, companies dealing with Wal-Mart, DoD and FDA are forced to implement RFID systems in the near future. Implementing RFID systems and managing RFID data are challenging tasks.

This thesis focuses on integrating RFID technologies with existing business process applications.

This chapter explains the research problem and its motivation, reviews the relevant literature, and details the research methodology. It ends with a description of the thesis organization.

1.1 Research problem and Motivation

Prior to the introduction of RFID, companies have installed Enterprise Resource Planning (ERP) software and communicate business events using Electronics Data Interchange (EDI). Now with RFID, companies are struggling to integrate the technology and its associated software applications into the current enterprise IT infrastructure.

How RFID technology and its associated software applications can be integrated into the current enterprise IT infrastructure? This thesis is motivated by this question. Specifically, this thesis chooses the standard protocol for data exchange between applications, which is EDI, as an example. This thesis examines EDI and describes how the current business-to-business transaction process enabled by EDI, should be modified with the introduction of RFID technologies.

1.2 Literature Review

With the steady adoption of the RFID technology by both the private and public entities, research associated with RFID technology went up tremendously. However, most of the researchers treat the RFID system as a stand-alone system, and there are few papers which discuss the problem of integrating RFID technology into an existing IT infrastructure, and there are fewer which discuss the EDI infrastructure.

Several articles provide insight on integration issues. Yoon, Duncan, Robin, Christian and Laxmiprasad presents guidelines on integrating the Auto-ID data into the Manufacturing business information system [4]. Timothy had written some particularly interesting papers, which highlights the processes for shipping and receiving, and presented ways to incorporate RFID data into such process [5, 6]. Issues about how RFID technology offers more intelligence to ERP system have been explored by Edmund and David [7, 8]. SAP had published approaches to integrate Auto-ID data into its own ERP system [9, 10].

1.3 Research Methodology

I plan to address the question by firstly defining the problem in a more specific way. Within the EDI transactions, I am interested to find out how RFID technology can enable automatic re-conciliation and payment process. I plan to study the EDI protocol of ANSI X12 810 (Invoice), 856 (Ship Notice) and 820 (Payment), as well as the RFID related standard, particularly Physical Markup Language (PML) representation of the read events. Then, I will use EPCglobal PML version 1.0 as the reference to study the PML representation of such read events. Finally, I propose a simple algorithm to automate the re-conciliation and payment process.

My goal is to create an algorithm, which is an efficient procedure, for automating the reconciliation of receiving and the subsequent payment process.

1.4 Organization

This thesis consists of seven chapters and one appendix. Chapter 2 introduces basic RFID technology, and Chapter 3 presents basic EDI technology. Chapter 4 discusses the business transactions, such as Automatic Shipment Notice (ASN) and automatic reconciliation. Chapter 5 presents a simple algorithm to address the problem of automating these functions. Chapter 6 points out some areas for further research, and Chapter 7 conclude the thesis.

Appendix 1 presents some EDI message protocols in great detail.

2 RFID

2.1 Introduction of RFID

RFID is a means of identifying an object using radio frequency transmission. The technology can be used to identify, track, or detect a wide variety of objects. Communication takes place between a reader and a transponder (Silicon Chip connected to an antenna) often called a tag. Tags can either be active (powered by battery) or passive (powered by the reader field).

2.1.1 Reader and Tag communication

In a typical system, tags are attached to objects. Each tag has certain amount of internal memory in which it stores information about the object, in particular an object identifier such as the Electronic Product Code (EPC). When these tags pass through a reader, they transmit information back to the reader, thereby identifying the object. The data then is filtered and routed to the backend IT systems. Savant ([11], [12]) serves this purpose, which act as buffers between the RFID front-end and the IT backend.

A reader emits a signal at the selected frequency band, such as 860 - 960MHz for UHF ([13], [14]) or 13.56MHz for HF [15]. Any corresponding tag in the vicinity of the reader will detect the signal and use the energy from it to wake up and supply operating power to its internal circuits. Once the Tag has decoded the signal as valid, it replies to the reader, and indicates its presence.

If many tags are present then they will all reply at the same time, which at the reader end is seen as a signal collision and an indication of multiple tags. The reader manages this problem by using an anti-collision algorithm [16] designed to allow tags to be sorted and individually selected. Once a tag is selected, the reader is able to perform a number of operations such as read the tags identifier number, or write information to it (in the case of a read/write tag).

2.1.2 RFID Tag

Every object to be identified in an RFID system will need to have a tag attached to it. RFID tags are designed and manufactured using some of the most advanced and smallest geometry silicon processes available [17]. In terms of computational power, RFID tags are quite dumb, containing only basic logic and state machines capable of decoding simple instructions. Most RFID tags contain a certain amount of non-volatile memory (NVM) in order to store data. RFID tags are classified into 5 classes [19]:

- CLASS 0/I – Read Only, Factory programmed

These are the simplest type of tags, where the data is written only once into the tag during manufacturing. The memory is then disabled from any further updates.

- CLASS II – Write Once Read Only, Factory or User programmed

In this case the tag is manufactured with no data written into the memory. Data can then either be written by the tag manufacturer or by the user for one time. Following this no further writes are allowed and the tag can only be read.

- CLASS III – Read Write

This is the most flexible type of tag, where users have access to read and write data into the tags memory.

- CLASS VI – Read Write with on board sensors

These tags contain on-board sensors for recording parameters like temperature, pressure, and motion, which can be recorded by writing into the tags memory. As sensor readings must be taken in the absence of a reader, the tags are either semi-passive or active.

- CLASS V – Read Write with integrated transmitters.

These are like miniature radio devices which can communicate with other tags and devices without the presence of a reader. This means that they are completely active with their own battery power source.

2.2 EPC – Electronic Product Code

In 1999 Auto-ID Center was established at Massachusetts Institute of Technology (MIT) in US. The center consisted of six leading universities, including MIT, University of Cambridge, University of Adelaide, Keio University, Fudan University and University of St. Gallen. Auto-ID

Center had a common vision as "Internet of Things" [20]. Together with a number of leading companies, the Center developed the idea of a unique electronic identifier code called the Electronic Product Code (EPC) [18]. The EPC is similar in concept to the Universal Product Code (UPC) used in barcodes today. Having just a simple code of up to 256 bits would lead to smaller chip size, and hence lower tag costs, which is recognized as the key factor for wide spread adoption of RFID in the supply chain.

While the potential benefits of RFID tags had been identified long before, what was stopping the adoption of this technology was the cost of tags. The Auto-ID Center recognized that in order to solve this problem, tags needed to be as simple as possible, and act as pointers to information held on the Internet. This leads to the idea of the EPC [21].

The Auto-ID Center officially closed on October, 2003. The Center had completed its work and transferred its technology to EPCglobal [22], which will administer and develop EPC standards going forward. The center was renamed as MIT Auto-ID Lab [51] after the technology transfer, and it continued to refine the technology.

2.2.1 EPC Code - General

The code is similar to the UPC used in bar codes, and ranges from 64 bits to 256 bits with 4 distinct fields, including version number, domain manager number, objective class number and serial number. Each number is encoded in its distinct partition of the bit sequence. What sets the EPC apart from bar codes is its serial number, which distinguishes the uniqueness of an item and tracks it through the supply chain.

There are three general categories of EPC code:

- 64-bit EPC, known as EPC-64 [23]
- 96-bit EPC, known as EPC-96 [18]
- 256-bit EPC, known as EPC-256 [24]

The structure of three EPC variances is shown in Table 3.1 as follows:

		Version	Domain Manager	Object Class	Serial Number
EPC-64	TYPE I	2	21	17	24
	TYPE II	2	15	13	34
	TYPE III	2	26	13	23
EPC-96	TYPE I	8	28	24	36
EPC-256	TYPE I	8	32	56	192
	TYPE II	8	64	56	128
	TYPE III	8	128	56	64

Table 2.1 Structure difference of EPC variances (Source: [18], [23], [24])

As Table 2.1 indicates, the structural difference between EPC versions is the number of bits that are available to encode numbers in each of the other three partitions. The domain manager number controls the allocation of all object class numbers and serial numbers for that particular domain manager number [48].

2.2.2 EPC-96

Structure of EPC-96 is shown in Figure 2.1:

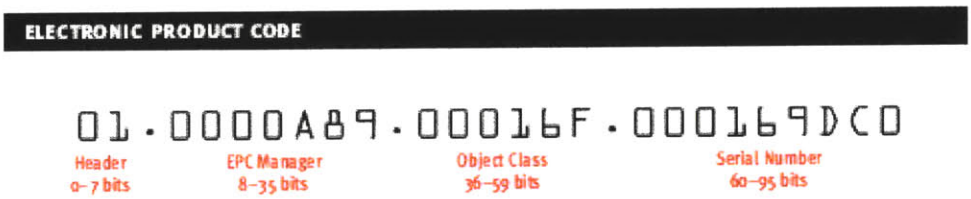


Figure 2.1 EPC-96 Code Structure [18]

- Header (0- 7) bits

The Header is 8 bits, and defines the length of the code. In this case 01 indicates an EPC type 1 number which is 96 bits in length. The EPC length ranges from 64 [23] to 256 bits [24].

- EPC manager (8- 35) bits

It typically contains the manufacturer of the Product an EPC tag is attached to

- Object Class (36-59) bits

It refers to the exact type of product in the same way as a Stock Keeping Unit (SKU)

- Serial Number (60 – 95) bits

It provides a unique identifier for up to 2^{36} products

2.2.3 EPC Network Infrastructure (Pre-EPCglobal)

2.2.3.1 Network Components

The EPC network infrastructure originally developed by Auto-ID Center consists of several key components:

- **Readers**

Reader detects and communicates with the tags when tags enter read range. Protocol on how readers communicate with host software, particularly Savant, has been defined by Auto-ID center [41].

- **Savant**

Savant [42] serves as a software buffer which sits between the RFID readers, and the servers storing the product information. It allows companies to process relatively unstructured tag data taken from many RFID readers, and direct it to the appropriate information systems.
Savants are able to perform many different operations, such as monitor the RFID reader devices, manage false reads, cache data and finally query an Object Naming Service (ONS).

- **Object Name Service (ONS)**

ONS [25] matches the EPC code to information about the product via a querying mechanism similar to the Domain Naming system (DNS) used in the Internet, which is already proven technology capable of handling the volume of data expected in an EPC RFID system. The ONS server provides the Internet Protocol (IP) address of a Physical Markup Language (PML) Server that stores information relevant to the EPC.

- **ONS Local Cache**

ONS Local Cache stores the frequently-asked or recently asked contents, so as to reduce the need to query the global ONS for each object. The local cache can manage lookup of private internal EPC for asset tracking. The registration function within local cache registers EPC with global ONS system, and it works with a dynamic ONS system to enable private tracking and collaboration within the supply chain for each unique object.

- **EPC Information Service (EPC IS)**

The EPC IS [30] presents EPC Network related data in PML format to requesting parties. Various data sources may be available through EPC IS, such as tag read data collected from Savant, instance level data (date of manufacture and expiry data) and object class level data (product catalog information). When requested, EPC IS pulls data from various sources within

an organization, translating data into PML format and present to the requesting parties. EPC IS used to be called PML Server [49].

2.2.3.2 Data format

The syntax and semantics of data exchanged among network components are represented by the data standards. As described in Section 2.2.1, EPC identifies a physical object. PML represents the information of objects in common and standardized Extensible Markup Language (XML) vocabularies.

- **Physical Markup Language (PML)**

PML ([26], [27]) is based on the widely used and accepted XML. PML is designed to store any relevant information about a product, such as location, telemetry, composition and etc. Auto-ID center defined PML Core [50] which is the core part of the PML representation.

2.2.3.3 Network Architecture

EPC Architecture can be implemented both within the enterprise as well as across the enterprises. Figure 2.2 shows the EPC Architecture within the enterprise, while Figure 2.3 shows the EPC Architecture across the enterprises.

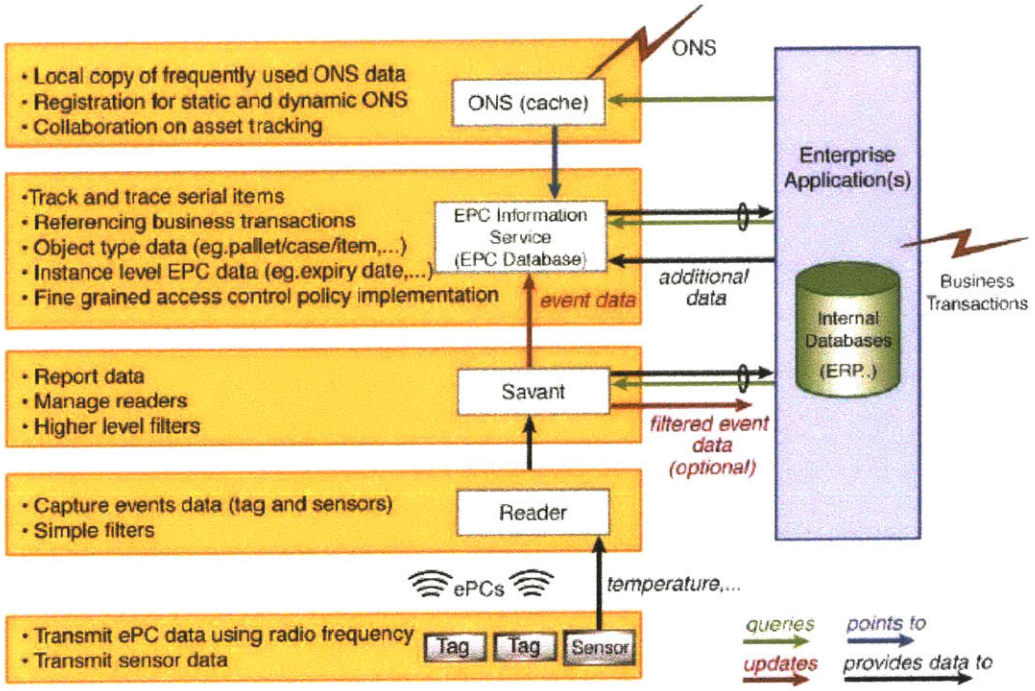


Figure 2.2 EPC architecture within enterprise [11]

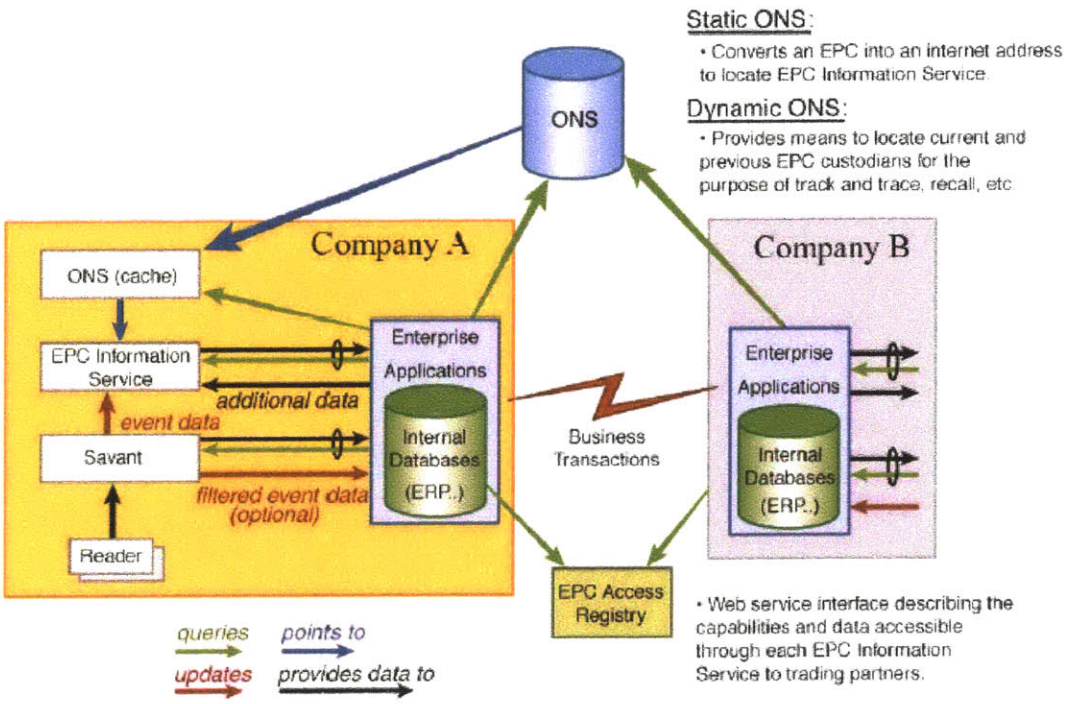


Figure 2.3 EPC architecture across enterprises [11]

One key component to enable both two implementation architecture is ONS.

As Figure 2.3 indicated, Static ONS provides an Internet address where information associated with a manufacturer is stored; while Dynamic ONS records a sequence of records where a product can be trace through its movement. ONS Local Cache, as presented by both Figure 2.2 and 2.3, keeps a local version of the records obtained from the global ONS server, so as to reduce the need for a local EPC system to query the global ONS server.

2.2.4 EPC Infrastructure (EPCglobal)

Since 2003 Oct, EPCglobal inherited the technologies developed by Auto-ID center and continued commercialization of the technology. There are several changes in terms of EPC Infrastructure after the transition:

2.2.4.1 Application Level Events (ALE)

ALE [28] is EPCglobal official version to replace the Savant defined by Auto-ID center. It clearly defines the functionalities of the software which sit between RFID reader and the data applications. It defines web-service interfaces for clients to access the filtering layer reports

through well-defined XML schema. It gives clients freedom on how to define the boundaries of the collection and reporting period, how to filter and group objects, and where to send the report.

2.2.4.2 EPC Information Service (EPC IS)

EPC IS [43] standardizes a client side web-service interface to allow applications to query and provide updates to networked database which store EPC-related data.

2.2.4.3 EPC IS Discovery Service

EPC IS Discovery Service aims to provide serial-level track and trace across the supply chain, and it is still under development. The registry within Discovery Service is updated by each custodian (within supply chain) on handover, which make serial-level EPC lookup possible. The revised EPC architecture looks like the following:

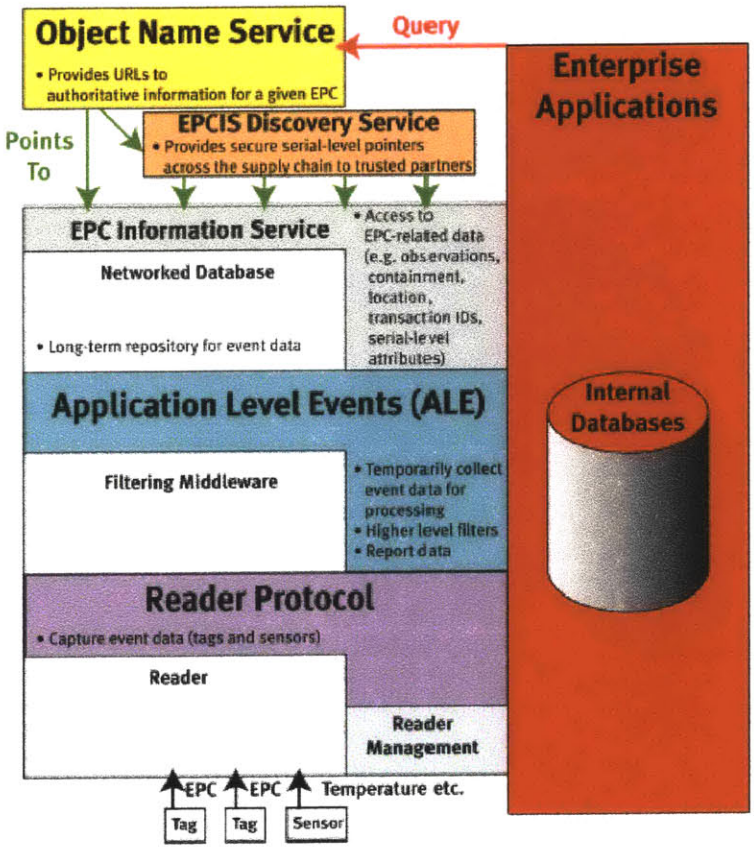


Figure 2.4 Revised EPC Network architecture [29]

3 EDI

EDI exchanges business documents between computers in a pre-defined standard format. It avoids the human intervention of reading and processing information between trade partners by establishing a standard data transmission protocol. Therefore, it eliminates the processing delays and errors due to the document transfers and data re-entries.

Business partners use EDI to transmit documents electronically, such as invoices, purchase orders, receipts and shipping documents. Financial document, such as payment order, can be transmitted through EDI as well.

3.1 Components of EDI system

Like any other communication process, EDI involves senders, receivers, language, content and medium. In EDI, the senders and receivers are called trading partners, and X12 and EDIFACT standards supply a common language for formatting the content of messages. Software tools called translators enable trading partners to converse in a standard language. Networking facilities such as the Internet or a commercial Value-added network (VAN) supplies the messaging medium.

A typical setup of EDI system is shown in Figure 5.1 as follows:

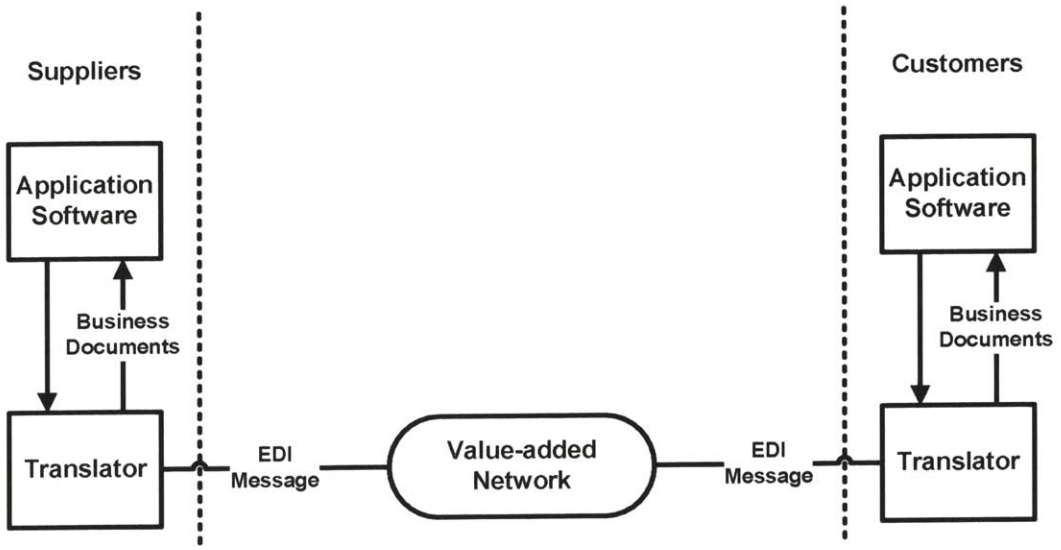


Figure 3.1 EDI System

- Trading Partners

Trading partners are any combination of organizations who are involved in a business transaction. A typical example of trading partners is Customers and Suppliers.

- Business Documents

A business document is a legal document defining the transaction conducted between trading partners. Trading partners define the legal boundaries for such transaction, and are bound by terms and conditions of such documents. Various types of business documents exist, such as Purchase Order, Purchase Order Acknowledgement, Purchase Order Change, Invoice and Payment Order.

- EDI Messages

The need for a common standard is obvious. Standardization of message structure allows representation that can be easily processed by computers independent from the application software.

Today, there are two major EDI message standards available

1) ANSI ASC X12 Standard

American National Standards Institute (ANSI) [31] formed a committee called the Accredited Standard Committee (ASC) in 1979, which developed ANSI ASC X12 standard [32]. ANSI ASC X12, normally known as X12 standard, defines the structure of the data, what documents are transmitted electronically and what information to be included in each document.

The X12 standard defines a set of documents, known as Transaction Set [33]. Each transaction set represents a business transaction, for example, the Transaction Set 810 refers to the invoice and Transaction Set 856 refers to the ship notice.

2) United Nations EDIFACT Standard

Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT) is a set of international standards provided by the United Nation to enable the electronics data exchanges between countries. It is a combination of the ASC X12 standards and the Trade Data Interchange (TDI) standards developed in United Kingdom and used throughout Europe.

This thesis uses ASC X12 standard exclusively for discussion.

- Application Programs

Application programs generate and process data in business documents; typically they are part of ERP suite.

- Translators

The translator is responsible for mapping application data to the EDI standard format and vice versa. It separates the application software's focus, which is business logic, from EDI standard message structure.

- Value-added Networks

VAN is the communication network which provides a medium to transmit and receive EDI transactions between trading partners.

3.2 X12 standard overview

A good introduction about the EDI syntax and terminology can be found in [35].
There are three components to the X12 standard, namely (1) transaction sets, (2) segments and (3) data elements, each of which will be reviewed in this section.

1. Transaction sets

A transaction set is equivalent to a business document. Each business document that has been adopted by X12 has a corresponding transaction set. Table 3.1 highlights some of the popular transaction sets, some of which will be covered extensively in this thesis.

Transaction Set	**Business Document**
810	Invoice
820	Payment Order
832	Sales Catalog
850	Purchase Order
855	Purchase Order acknowledgment
856	Ship Notice

Table 3.1 Selected X12 standards (Source: combined sources)

Transaction Sets start with a Transaction Set Header segment (ST) and end with a Transaction Set Trailer segment (SE). A specific segment may occur in several positions within a transaction set, representing different information in each different position.

2. Segments

Each transaction set is made up of multiple data segments. Appendix 1 illustrates the Transaction Set 810, 850 and 856. The sample code from 810 in Appendix 1 is as follows:

ST*810*0001
BIG*19981231*10429*****FB
N1*BT*The Scheduling Coordinator, Inc
N3*53241 Hamilton Dr
N4*Palo Alto*CA*95622*US
N1*RE*Bank of America- (Mkt and GMC)
N3*1850 Gateway Boulevard
N4*Concord*CA*94520*US
REF*11*1233626208
REF*01*121000358
ITD*03*****19990107
DTM*150*19980930
DTM*151*19981031
IT1*1*1*EA*19406.14**TP*0101
PID*X****Day-Ahead Spinning Reserve due ISO
IT1*2*1*EA*1764.26**TP*0102
PID*X****Day-Ahead Non-Spinning Reserve due ISO
IT1*3*1*EA*9774.42**TP*0103
PID*X****Day-Ahead AGC/Regulation due ISO
TDS*21351663
CTT*16
SE*48*0001

Each line represents a data segment. Each segment contains information that is needed to make a complete transaction. The asterisks "*" represent separators between data elements.

For example, the N4 segment is Geographic Location segment to specify the geographic place of the named party;

N4*Concord*CA*94520*US

N4 segment consists of four data elements: (1) the city, (2) state, (3) postal code and (4) country.

Segments generally start with a two or three character segment tag which identifies the segment. Data elements are separated by a delimiter character known as an element separator, and end with a different delimiter character known as the segment terminator. Elements which are not assigned values in a particular instance of a segment are represented by consecutive delimiters, and such trailing delimiters are not transmitted. For example, if a segment XYX has five elements and in a particular transmission only the second has a value, it is represented as XYZ**123<CR>, where "*" is the element separator and <CR> is the segment terminator.

The segment may be assigned as Mandatory, Optional, or Conditional. In N4 example, it is an optional segment.

Sometimes, segments may form a loop, which is a set of related segments in a Transaction Set. Segments are grouped together in this way to conveniently represent a block of related information. The loop within above sample code is as follows:

IT1*1*1*EA*19406.14**TP*0101
PID*X****Day-Ahead Spinning Reserve due ISO
IT1*2*1*EA*1764.26**TP*0102
PID*X****Day-Ahead Non-Spinning Reserve due ISO
IT1*3*1*EA*9774.42**TP*0103
PID*X****Day-Ahead AGC/Regulation due ISO

3. Data elements

Individual data elements are contained within each segment, and the combination of individual data elements make up a segment. For example, N4 segment consists of City Name, State Code, Postal Code and country code.

X12 defines the name of the data element, element usage (mandatory, optional or conditional), its minimum or maximum length and any pre-defined codes associated with this data element.

Data elements may have the following types: N (Numeric with implied decimal point, signed), ID (Identifier, a coded value, usually alphanumeric), AN (String, alphanumeric), DT (Date, YYMMDD) or TM (Time, HHMM). As in N4 example, data element named as "City Name" is an optional field with AN (Alphanumeric) data type, and its minimum length is 2 while its maximum length is 30.

3.3 EAN/UCC Code and GTIN

Various X12 transaction sets use European Article Number (EAN) /Uniform Code Council (UCC) [36] Code to identify the products during the transaction, including Baseline Item Data segment within Invoice transaction set (810), Baseline Item Data segment within Purchase Order transaction set (850) and Item Identification segment within Ship Notice transaction set (856).

There are four variances of EAN/UCC code, including original Universal Product Code (UPC, also known as UCC-12), EAN/UCC-8, EAN/UCC-13 (UPC's European variance) and EAN/UCC-14. These four codes are part of the Global Trade Item Number (GTIN) family of code structure.

GTIN is represented as 14 digits by right justifying and zero filling left, and acts as a superset to include 8 digits, 12 digits, 13 digits and 14 digits code structures of EAN/UCC as shown in Table 3.2.

GTIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
EAN/UCC-8	0	0	0	0	0	0	1	2	3	4	5	6	7	8
UCC-12(UPC)	0	0	1	2	3	4	5	6	7	8	9	10	11	12
EAN/UCC-13	0	1	2	3	4	5	6	7	8	9	10	11	12	13
EAN/UCC-14	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Table 3.2 GTIN code mapping (Source: combined sources)

Among the four EAN/UCC code structures, UCC-12 (UPC) is the most widely used version. UPC is developed by UCC, and it is the Bar Code which has been deployed widely in retailer chains. UPC consists of 12 digits, which are organized into 4 partitions [37]. The first digit indicates the numbering system. The second to sixth digits are the Manufacturer Identification number. The seventh to eleventh digits are the Item number which is maintained by Manufacturers, to make sure unique identification of the products the manufacturers produce. Last digit is just a validation check code.

Figure 3.2 UPC Code illustrated [18]

4 Business Transaction

4.1 Typical Business related transactions

During a typical buying and selling process, trading partners exchange business documents to record the exchange of goods and services. EDI helps to improve the speed and efficiency in such exchanges. Various parties are interested in exchange of business documents, such as customers, suppliers, carriers, financial institutes, insurance institutes and government agencies.

Customers may ask suppliers to provide price catalogs and quotes. They may place Purchase Order to supplier, and expect supplier to acknowledge their orders and provide delivery schedules. They may query the suppliers for the status of the orders. Sometime they may change or cancel their orders. They require suppliers to notify them about the shipment. They acknowledge the suppliers that the goods have been received, and sometimes return the goods damaged in transit. When the goods have been delivered, customers expect to receive the invoices.

Carriers transport goods from suppliers to customers. Normally, the shipper requests carrier pickup. The carrier responds to shipper with a pickup date. The carrier informs the receiving party of the shipment, and then the receiver tells the carrier where to unload the goods. The shipper may request to track the status of shipment, and if so, the carrier has to inform the shipper of the status of shipment. Once delivered, the carrier bills the shipper with an invoice, and carrier then receives payment from the shipper.

Financial institutions act as agents for payers. In a typical buying and selling process, banks receive payment authorization instructing them to transfer funds from a customer's account to a supplier's account. Sometimes the customer may request the bank for account information, and the bank responds to the request.

Sometimes insurance institutions are involved in the buying and selling process, to pay claims for items damaged during shipment. The customer establishes an insurance policy with the insurance company. The customer then informs the insurer of the goods being transported. Depending on the shipment outcome, the insurer may process the claim for goods being damaged in transit and inform the customer about the settlement.

Government regulation bodies, at federal, state or local level, may also involve in some transaction. Customs departments are involved in shipping and receiving goods across national board. Typically the shipper provides Customs with detail records of cargos in shipment. The shipper obtains either a cargo release and clearance information, or a rejection notice from the customs.

4.2 Technology enabler

Due to complexities indicated in Section 4.1, business communities rely heavily on technology to automate those transactions. The whole e-Business industry is built upon to support such transactions. With the proliferation of Internet, more technology standards based on Internet are established. EDI used to play an important role in such business transaction, but increasingly Internet based technologies, such as Electronic Business XML (ebXML) [47] and Universal Business Language (UBL) [46], are taking the lead.

4.3 Business transaction using EDI

EDI has been implemented for over 20 years, and it has automated a lot of business transactions. For example, several ANSI X12 messages are exchanged between the buyer and supplier to finish a procurement process.

Figure 4.1 captures several key processes of Procurement using EDI messages. These messages are exchanged in a sequence, beginning with the Sale Catalog.

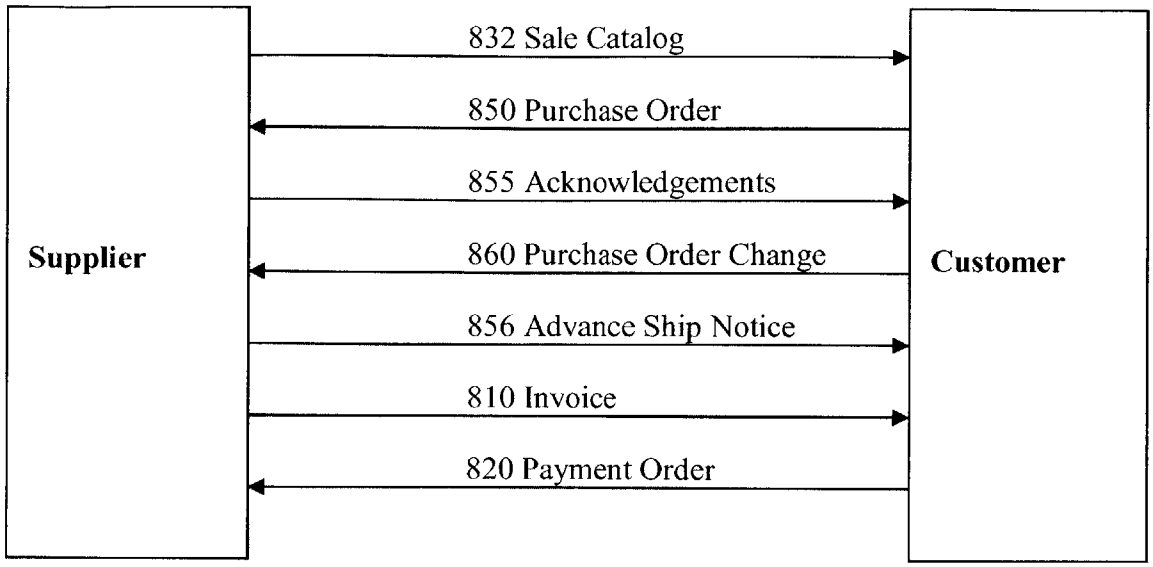


Figure 4.1 Procurement Process using EDI message

EDI 832 conveys product, pricing and packaging information, the typical data which one will find in product catalogs. Suppliers normally create and transmit EDI 832 documents to the customers as soon as products are added or updated within its product catalog.

Once the customer makes a purchase decision, he sends EDI 850 (Purchase Order) to the supplier. EDI 850 translates the typical paper based Purchase Order into the EDI format.

Upon receiving the EDI 850 message, the supplier replies the customer with an EDI 855 message, which conveys the supplier's confirmation of receipt of the Purchase Order for materials and quantity described within EDI 850.

Before the supplier sends the products to the customer, the customer can modify the Purchase Order by sending the supplier an EDI 860 message.

The supplier transmits an EDI 856 Advanced Shipping Notice (ASN) to shipment recipient after shipment leaves supplier's loading dock. ASN helps to expedite the receiving process on the customer side.

Once the customer receives the products, and re-conciliates his order, the supplier sends the EDI 810 invoice to customer. Upon receiving the invoice, customer instructs the financial institute with EDI 820 Payment order to pay the supplier accordingly.

5 Integrating RFID with EDI

The current reconciliation process involves people to physically count the products received and manually verify the products received with the Purchase Order. The deployment of RFID readers in the receiving dock can help automating this process. Middleware is required to convert the EPC information (which contains Serialized GTIN code) into UPC code, so that the question about "what is received" can be answered. Also the middleware needs to record the amount received and compare this with the Purchase Order to make sure the quantity received is the same as indicated in the Purchase Order.

5.1 Current receiving process

Although EDI has automated many steps as indicated in Figure 4.1 in Chapter 4, human beings are still involved in physical counting products and reconciling the Purchase Order.
Figure 5.1 depicts the automatic reconciliation through a manual counting process.

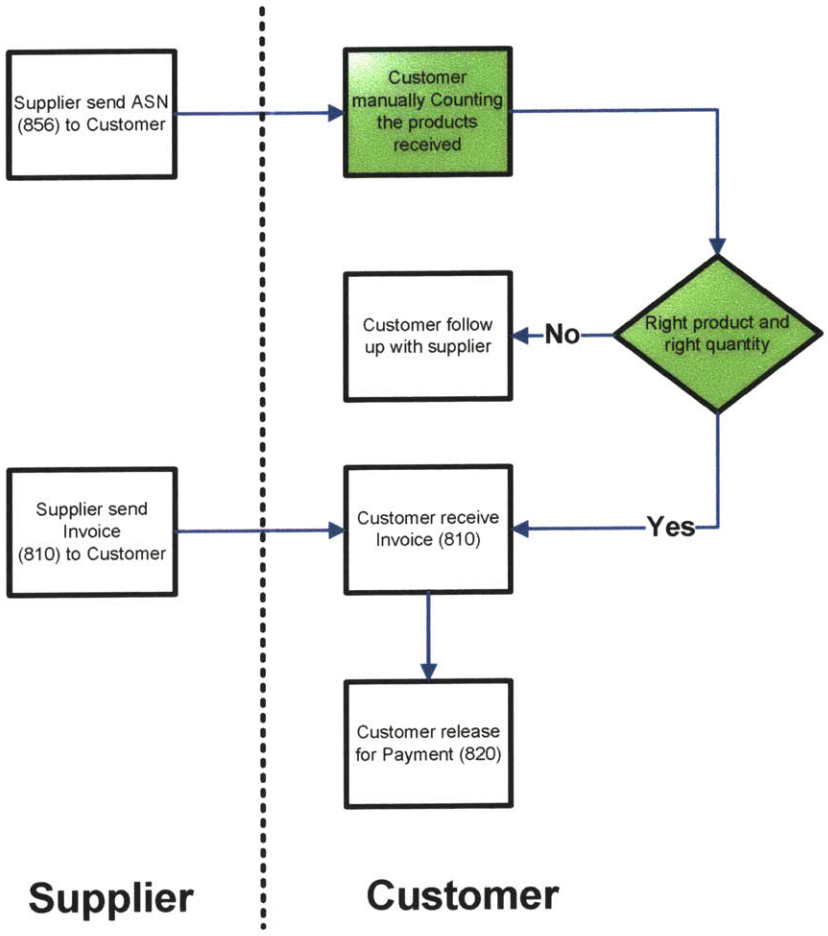


Figure 5.1 Manual Reconciliation

RFID technology can potentially automate the re-conciliate process, particularly in counting and verification area as illustrated in Figure 5.1.

5.2 Research Focus

To integrate RFID with EDI in an Automatic Reconciliation Process, the Ship Notice (856) and PML representation of the RFID reader event should be studied carefully.

5.2.1 Ship Notice (856)

Appendix 1, Section 3 is the detailed message structure for Ship Notice (856).

There are three segments in 856 that are particularly interesting in this application:

1) Segment PRF: Purchase Order Reference

   It provides a reference to a specific Purchase Order.

2) Segment LIN: Item identification

It describes what items have been ordered in the Purchase Order, and it can be represented by UPC [36].

3) Segment SN1: Item Detail

It describes the items' quantities.

The combination of the PRF, LIN and SN1 segment is powerful. It is the message the supplier gives the customer, identifying which Purchase Order the particular shipment is intended to fulfill, the amount and types of product in the shipment.

5.2.2 EPC Architecture

As described in Chapter 2, an RFID reader scans the tag and passes the events to the ALE. ALE filters and aggregates the read events to EPC IS (as shown in Figure 2.4). EPC IS presents the relevant information in the PML format.

5.3 Key Assumptions

Several assumptions are needed to make this application possible:

1) EPC IS can be configured to present individual read event.
2) Item level read event is represented in the PML format on EPC IS.
3) Purchase Order is associated with the EPC architecture through some ERP middleware.

5.4 Data Format

To implement the algorithm, the data format of UPC used in 856 and data format of PML used in EPC architecture need to be studied.

A typical PML representation from EPC IS looks like the following:

```
<pmlcore: Sensor>
        <pmluid:ID>urn:epc:1:4.16.36</pmluid:ID>
        <pmlcore:Observation>
                <pmlcore:DateTime>2005-04-03T13:04:34-06:00</pmlcore:DateTime>
                <pmlcore:Tag>
                        <pmluid:ID>urn:epc:id:sgtin:0000932.001fdc.000000000</pmluid:ID>
                </pmlcore:Tag>
                <pmlcore:Tag>
                        <pmluid:ID> urn:epc:id:sgtin: 0000932.001fdc.000000001</pmluid:ID>
                </pmlcore:Tag>
        </pmlcore:Observation>
</pmlcore:Sensor>
```

Notice that the EPC code embedded into the RFID tag is captured as follows:

```
<pmluid:ID> urn:epc:id:sgtin: 0000932.001fdc.000000000</pmluid:ID>
```

Uniform Resource Identifier (URI) is used in above sample. URI represents EPC in an independent way so that application software can manipulate. It decouples the application logic from the particular way EPC was represented within RFID tags.

There are several identification types which URI can represent, one of which is Serialized Global Trade Identification Number (SGTIN). SGTIN is the serialized version of GTIN, which has a serial number appended to the normal GTIN format.

The URI format for SGTIN is [38]:

urn:epc:id:sgtin:CompanyPrefix.ItemReference.SerialNumber

Chapter 2 has already illustrated how EPC Code is constructed. As indicated by [39], UPC (UCC-12, which is one of the GTIN representation as described in Section 3.3) code can be converted into EPC code easily, by converting the decimal number of the Manufacturer ID Number and Item number in UPC (UCC-12) code into Hexadecimal format, and stuffing them into EPC's EPC Manager and Object Class section respectively.

Figure 5.2 Map UPC (UCC-12) code into EPC [40]

5.5 Solution

Clearly, the reconciliation can be automated as long as there is software that extracts the EPC code from EPC IS, converts it into the UPC code, and correlates it with the Ship Notice (856) message. Figure 5.4 shows how the traditional manual process will be modified.

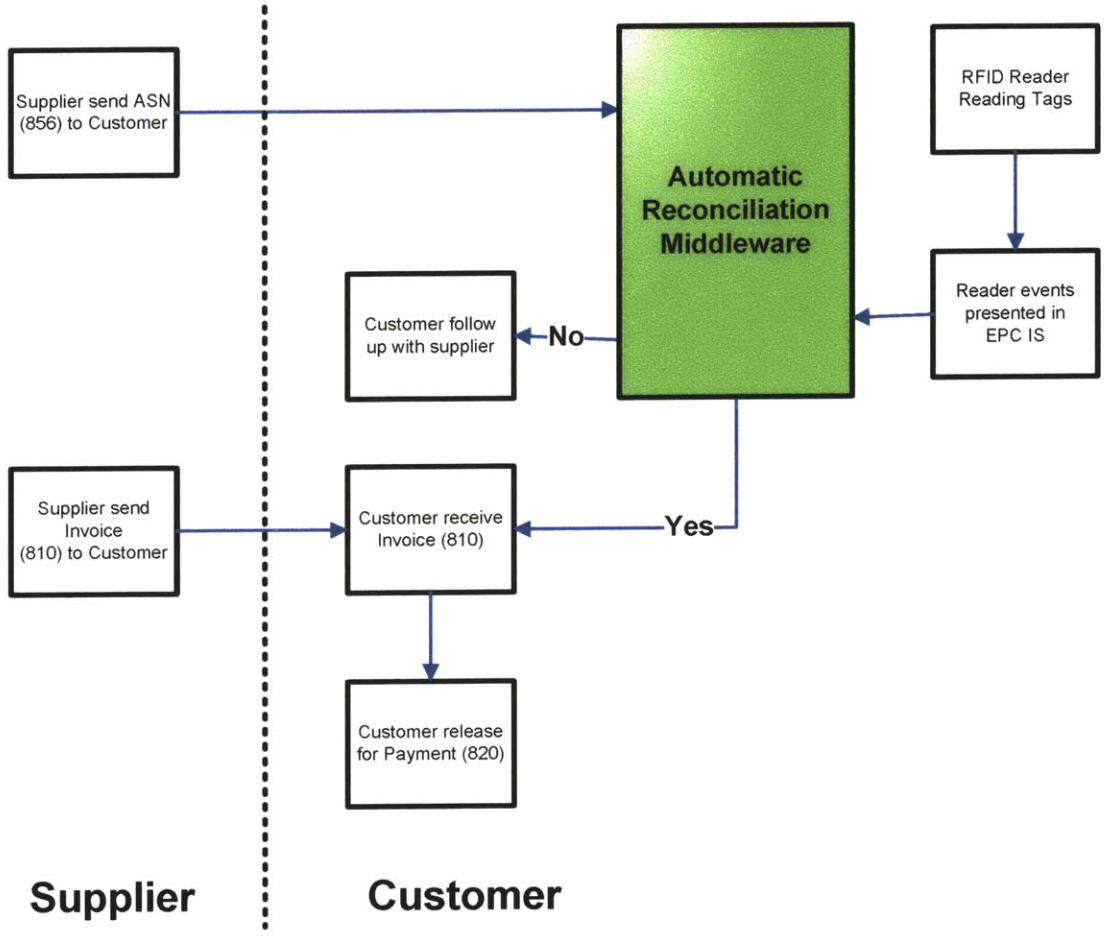


Figure 5.3 Automatic Reconciliation

As indicated in Figure 5.3, the middleware enabled by RFID will replace the tradition manual process shown in Figure 5.1. Figure 5.4 explains the detail implementation within the middleware.

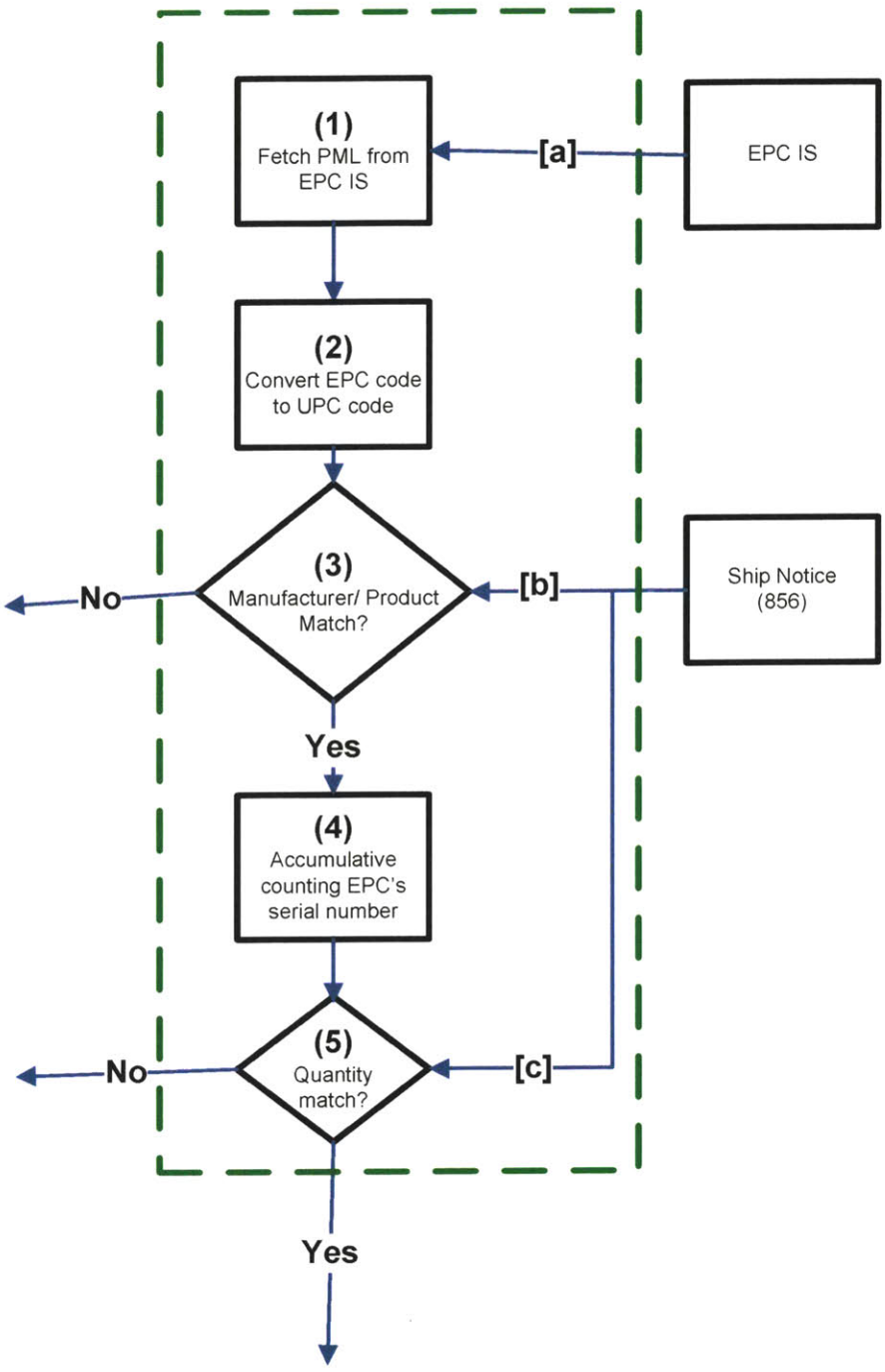


Figure 5.4 Automatic reconciliation Algorithm

Step (1)

The middleware fetches PML representation from EPC IS for the reader events during the receiving process. The PML representation from EPC IS through interface [a] (as shown in Figure 5.4) is as follows:

```
<pmlcore: Sensor>
        <pmluid:ID>urn:epc:1:4.16.36</pmluid:ID>
        <pmlcore:Observation>
                <pmlcore:DateTime>2005-04-03T13:04:34-06:00</pmlcore:DateTime>
                <pmlcore:Tag>
                        <pmluid:ID>urn:epc:id:sgtin:0000932.001fdc.000000000</pmluid:ID>
                </pmlcore:Tag>
                <pmlcore:Tag>
                        <pmluid:ID> urn:epc:id:sgtin: 0000932.001fdc.000000001</pmluid:ID>
                </pmlcore:Tag>
        </pmlcore:Observation>
</pmlcore:Sensor>
```

Step (2)

SGTIN is included in the PML representation as above, and SGTIN is a special type of EPC code which incorporates GTIN with serial numbers. Therefore the middleware can convert this PML representation of EPC code into GTIN representation.

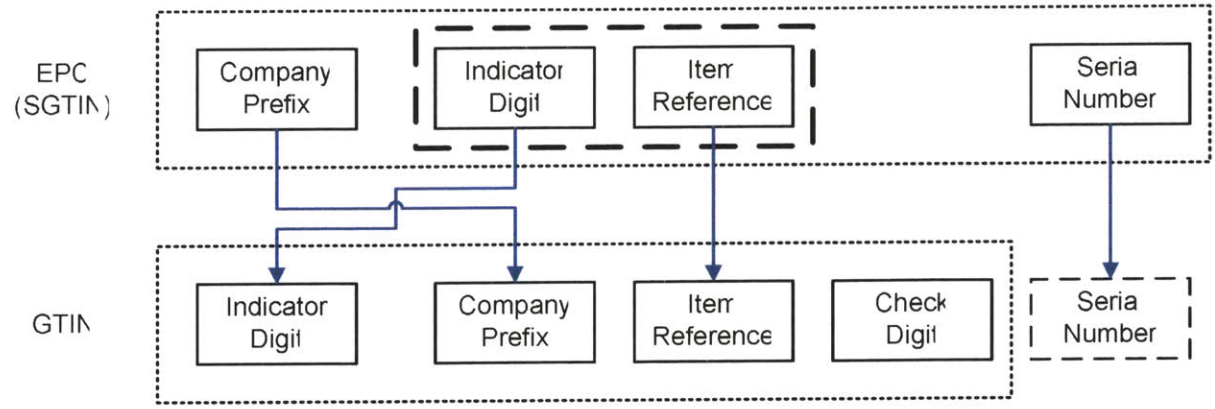


Figure 5.5 SGTIN and GTIN conversion [44]

As Section 5.4 has already discussed, PML representation of EPC code has the following format:

urn:epc:id:sgtin:CompanyPrefix.ItemReference.SerialNumber

Take Step (1) as an example, that is:

urn:epc:id:sgtin:0000932.001fdc.000000000

EPC code is represented in Hexadecimal format, while GTIN is in Decimal format. Therefore 0000932 (Hexadecimal) = 2354 (Decimal), and 01fdc (Hexadecimal) = 8156 (Decimal). The first digit of ItemReference ("001fdc") is "0" and it becomes the first digit of GTIN code.

As Section 3.3 has discussed, UPC code is 12 digits representation of GTIN. Therefore, the converted UPC code becomes:

0|02354|08156|5

Note that both "2354" and "8156" have been left appended with a "0" to make up 12 digits format. The check sum "5" is appended as the last digit.

Step (3)

Ship notice (856) provides the middleware with details about what have been shipped through interface [b] (as shown in Figure 5.4). Segment LIN (Item Description) within 856 looks like this:

LIN*001*UP*002354081565

That means the supplier informs customer that products with UPC code 002354081565 are going to be shipped.

The middleware can compare the GTIN (which includes UPC code) obtained in Step (2) and compare with UPC code specified by the EDI 856 message.

If the RFID reader reads the correct product code as the EDI 856 specifies, customer is receiving the right products and the receiving and reconciliation process continues.

If the recovered GTIN code is not match with the EDI 856 UPC code, customer should contact supplier for clarification.

Step (4)

The middleware will reach this step only if the customer is receiving the right products. Now the middleware needs to help customer to make sure they receive the right quantity.

Ship notice (856) also provides the middleware with details about how many have been shipped. Segment SN1 (Item Details) within 856 looks like this:

SN1**20*CA

That means the supplier informs the customer that 20 cases are going to be shipped.

PML representation includes the serial number of each read events, therefore the middleware can accumulate the serial number.

Step (5)

The middleware then compares the accumulated count with the number specified by the EDI 856 through interface [c] (as shown in Figure 5.4), to decide whether the quantity received is the same as the ship notice specified.

If the accumulated count is the same as the EDI 856 specifies, customer is receiving the right quantity. The customer completes its reconciliation process. Once it receives the invoice from supplier, it can release the payment.

If the accumulated count is different from the EDI 856 UPC code, the customer should contact supplier for clarification.

6 Future Research

For over 20 years EDI has played a key role in facilitating global electronic trade. However its adoption is mainly among Fortune 1000 companies, partially due to the following reasons:

- EDI is an expensive technology requiring high implementation and maintenance cost
- Only large organizations with long-term and high volume trade can afford such system
- Each manufacturer has its own implementation guideline, which make deployment a point-to-point solution between trading partners. Therefore EDI can not scale to support global trade which involves multiple trading partners.

6.1 ebXML

Electronic Business XML (ebXML) is envisioned as an open framework for global e-commerce, which is compatible with EDI and which is slowly replacing EDI [47]. ebXML defines business processes and associated core components in XML [52], and has a mechanism for registering and storing such business processes through a shared Registry [53, 54]. It also has mechanisms for describing Trading Partner Capabilities and Trading Partner Agreements [55]. In a nutshell, it standardizes the methodology/process for modeling real world business processes and translates these processes into XML representation.

Figure 6.1 illustrates how ebXML works from a high-level technical architecture view.

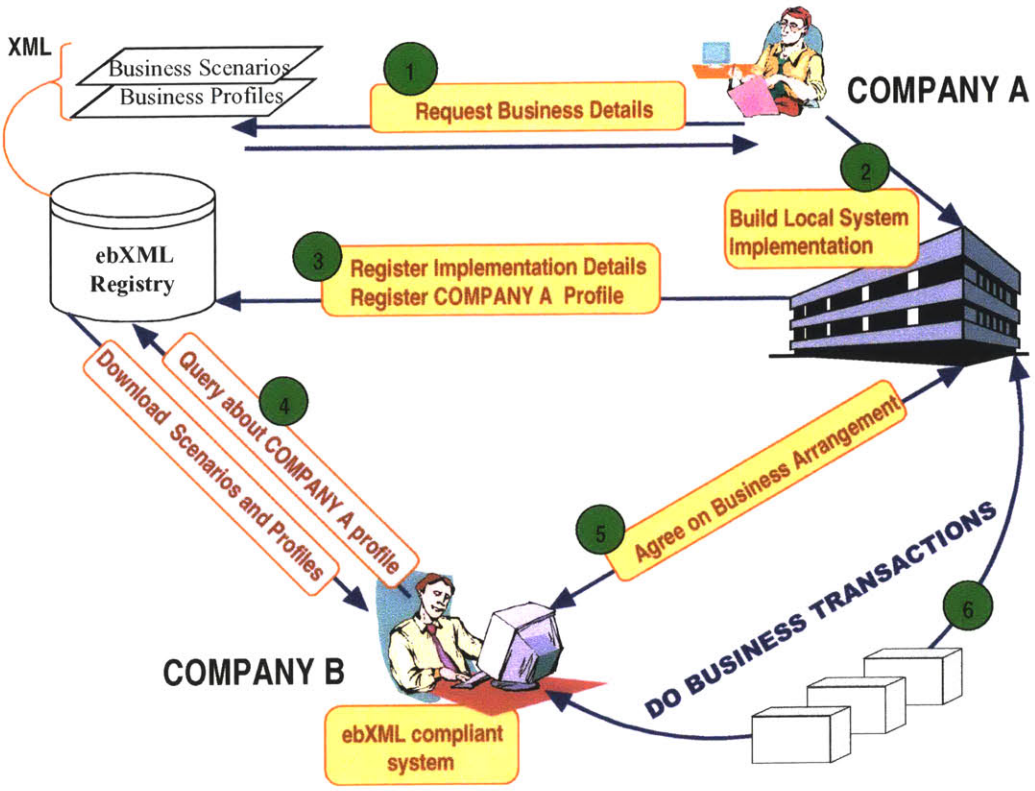


Figure 6.1 Scenario: two companies conduct e-commerce using ebXML [56]

One valuable direction for further research is to study how RFID data can be integrated into an ebXML driven business process. Since ebXML standardization is still under development, such research will help to shape the ebXML standard and make it easier to integrate with RFID technology.

6.2 Other areas for research

Integration of RFID data with the EDI business processes is still an interesting research topic, because EDI has the major role in electronic business transactions today. The following questions remain to be answered:

- Whether current EDI message protocols need to be modified to ease the integration of RFID and EDI
- Whether RFID can be included into the current EDI message protocol

Research on above topics can help accelerating the adoption of RFID technology into existing IT infrastructure.

7 Conclusion

With the further deployment of RFID technology, there will be more challenges and questions on how to integrate the RFID technology with existing IT systems.

This thesis presents one approach whereby RFID technology can integrate with EDI transactions to automate the receiving process. A middleware is proposed to implement this approach. By implementing this middleware, a firm can improve its re-conciliation efficiency and become less reliant on human intervention.

The author hopes this thesis can encourage more research on this important topic and help to accelerate the integration of RFID technology with the existing IT infrastructure, such as EDI.

References

[1] Lamont Wood. (2004, October) *Retail's Biggest RFID Project.* Chain Store Age, p. 11A.

[2] US Department of Defence. (2004, July) *US DoD RFID Policy Memo.* Washington DC.

[3] Jonathan Collions. (2004, November 15) *FDA Clears Way for RFID Tagging*, RFID Journal.

[4] Yoon Chang, Duncan McFarlane, Robin Koh, Christian Floerkmeier, Laxmiprasad Putta. (2002, November 1) *Methodologies for Integrating Auto-ID Data with existing Manufacturing Business Information Systems,* White Paper CAM-AUTOID-WH-009. The Auto-ID Center, Institute for Manufacturing, University of Cambridge, Cambridge. http://www.autoidcenter.org/research.asp

[5] Timothy P. Milne. (2002, November 1) *Auto-ID Business Use-Case Framework (A-Biz), Background,* Technical Memo MIT-AUTOID-TM-009. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[6] Timothy P. Milne. (2002, November 1) *Auto-ID Business Use-Case Framework (A-Biz), Dispatch Advice Use-Case,* Technical Memo MIT-AUTOID-TM-010. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[7] Edmund W. Shuster, David L. Brock. (2004, April) *Creating an Intelligent Infrastructure for ERP Systems, The role of RFID Technology.* Working Paper, APICS

[8] Edmund W. Shuster, David L. Brock, Stuart J. Allen, Pinaki Kar. (2005). *Enabling ERP through Auto-ID Technology.* Stanford University Press

[9] Christof Bornhovd, Tao Lin, Stephan Haller, Joachim Schaper. (2004). *Integrating Automatic Data Acquisition with Business Process, Experiences with SAP's Auto-ID Infrastructure.* Proceedings of the 30th VLDB Conference, Toronto Canada

[10] Christof Bornhovd, Tao Lin, Stephan Haller, Joachim Schaper. (2005). *Integrating Smart Items with Business Processes, An Experience Report.* Proceedings of the 38th Hawaii International Conference on System Sciences

[11] Oat Systems & MIT Auto-ID Center. (2002, February 1) *The Savant Version 0.1 (Alpha),* Technical Report MIT-AUTOID-TM-003. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[12] Amit Goval. (2003, April) *Savant Guide,* Technical Report MIT-AUTOID-TR-015. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[13] Auto-ID Center. (2003, June 1) *860MHz-930MHz Class 0 Radio Frequency Identification Tag Protocol Specification, Candidate Recommendation, Version 1.0.0,* Technical Report MIT-AUTOID-TR-016. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[14] Auto-ID Center. (2002, November 14) *860MHz-960MHz Class 1 Radio Frequency Identification Tag, Radio Frequency & Logical Communication Interface Specification, Recommended Standard, Version 1.0.0,* Technical Report MIT-AUTOID-TR-007. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[15] Auto-ID Center. (2003, February 1) *13.56 MHz ISM Band Class 1 Radio Frequency Identification Tag Interface Specification: Recommended Standard, Version 1.0.0,* Technical Report MIT-AUTOID-TR-011. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[16] Feng Zhou, Dawei Jin, Chenling Huang, Hao Min. (2003, October 1) *Optimize the Power Consumption of Passive Electronic Tags for Anti-Collision Schemes,* White Paper FUD-AUTOID-WH-001. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[17] Sanjay Sarma. (2001, November 1) *Towards the 5c Tag,* White Paper MIT-AUTOID-WH-006. The Auto-ID Center, MIT. Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[18] David L. Brock. (2001, Jane 1) *The Electronic Product Code (EPC), A Naming Scheme for Physical Objects,* White Paper MIT-AUTOID-WH-002. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[19] Sanjay Sarma, Daniel W. Engels. (2003, June 1) *On the Future of RFID Tags and Protocols,* Technical Report MIT-AUTOID-TR-018. The Auto-ID Center, MIT, Cambridge, Massachusetts.
http://www.autoidcenter.org/research.asp

[20] http://web.mit.edu/auto-id/www/aboutthelabs.htm

[21] David L. Brock. (2003, June 1) *The Electronic Product Code (EPC) as a Meta Code,* White Paper MIT-AUTOID-WH-020. The Auto-ID Center, MIT, Cambridge, Massachusetts.

http://www.autoidcenter.org/research.asp

[22] http://www.epcglobalinc.org/

[23] David L. Brock. (2001, November 1) *The Compact Electronic Product Code, A 64-bit Representation of the Electronic Product Code,* White Paper MIT-AUTOID-WH-008. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[24] David L. Brock. (2003, February 1) *EPC-256: The 256-bit Electronic Product Code Representation,* Technical Report MIT-AUTOID-TR-010. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[25] Oat Systems & MIT Auto-ID Center. (2002, February 1) *The Object Name Service, Version 0.5 (Beta),* Technical Manual MIT-AUTOID-TM-004. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[26] David L. Brock. (2001, February) *The Physical Markup Language, A Universal Language for Physical Objects,* White Paper MIT-AUTOID-WH-003. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[27] David L. Brock. (2001, June 1) *The Physical Markup Language, Core Components: Time and Place,* White Paper MIT-AUTOID-WH-005. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[28] EPCglobal. (2004, October) *The Application Level Events (ALE) Specification, Version 1.0, EPCglobal Last Call Working Draft Version of 14 Oct 2004,* EPCglobal.

[29] Mark Harrison. *EPC Information Service (EPCIS),* Cambridge Auto-ID Lab, Institute of Manufacturing, University of Cambridge.

[30] Mark Harrison. (2003, October 1) *EPC Information Servic, Data Model and Queries,* White Paper CAM-AUTOID-WH-025. Cambridge Auto-ID Lab, Institute of Manufacturing, University of Cambridge. http://www.autoidcenter.org/research.asp

[31] http://www.ansi.org/

[32] http://www.x12.org/

[33] http://www.edi-service.com/ansix12c.htm

[34] Robert L Scheier. (2003, January 20). *Internet EDI grows up.* Computerworld, Framingham. Vol. 37, Issue 3; p. 38

[35] Edward Cannon. (1993). *EDI Guide: a step by step approach.* Van Nostrand Reinhold.

[36] Uniform Code Council, http://www.uc-council.org/

[37] Uniform Code Council (UCC). (1986, January) *Uniform Product Code (UPC), Symbol Specification Manual* http://www.uc-council.org/reflib/01302/d36-t.htm

[38] EPCglobal. (2003, September 15) *PML Core Specification 1.0.* EPCglobal.

[39] David L. Brock. (2003, June 1) *EPC Tag Data Specification,* White Paper MIT-AUTOID-WH-025. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[40] David L. Brock. (2001, November 1) *Integrating the Electronic Product Code (EPC) and the Global Trade Item Number (GTIN),* White Paper MIT-AUTOID-WH-004. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[41] Auto-ID Center. (2003, September 5) *Auto-ID Reader Protocol 1.0, Working Draft Version of 5 September 2003.* The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.epcglobalinc.org/standards_technology/specifications.html

[42] Auto-ID Center. (2003, September 1) *Auto-ID Savant Specification 1.0, Version of 1 September 2003.* The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.epcglobalinc.org/standards_technology/specifications.html

[43] EPCglobal. (2005, March 8) *EPC Information Services (EPCIS) Version 1.0 Specification, Working Draft Version of 8 March 2005.* EPCglobal.

[44] EPCglobal. (2004, April 1) *EPC Tag Data Standard Version 1.1 Rev.1.24.* EPCglobal.

[45] EPCglobal. (2004, April 15) *EPCglobal Object Name Service (ONS) 1.0.* EPCglobal.

[46] http://www.oasis-open.org/committees/ubl

[47] http://ww.ebxml.org

[48] Daniel Engels. (2003, February 1) *The Use of the Electronic Product Code,* Technical Report, MIT-AUTOID-TR-009. The Auto-ID Center, MIT, Cambridge, Massachusetts. http://www.autoidcenter.org/research.asp

[49] Mark Harrison, Humberto Moran, James Brusey, Duncan McFarlane. (2003, September 1) *PML Server Developments,* White Paper, CAM-AUTOID-WH-015. The Auto-ID

Center, Institute for Manufacturing, University of Cambridge, Cambridge. http://www.autoidcenter.org/research.asp

[50] Christian Floerkemeier, Dipan Anarkat, Ted Osinski, Mark Harrison. (2004, January 1). *PML Core Specification 1.0,* White Paper, STG-AUTOID-WH-005. The Auto-ID Center, University of St. Gallen, Institute of Technology Management, Switzerland. http://www.autoidcenter.org/research.asp

[51] http://web.mit.edu/auto-id/www/index.htm

[52] OASIS. (2001, May 10) *Core Component Overview v1.05.* http://www.ebxml.org/specs/ccOVER.pdf

[53] OASIS. (2001, December 18) *Registry Information Model v2.0.* http://www.ebxml.org/specs/ebrim2.pdf

[54] OASIS. (2001, December 6) *Registry Services Specification v2.0.* http://www.ebxml.org/specs/ebrs2.pdf

[55] OASIS. (2002, September 23) *Collaboration-Protocol Profile and Agreement Specification v2.0.* http://www.ebxml.org/specs/ebcpp-2.0.pdf

[56] OASIS. (2001, February 16) *ebXML Technical Architecture Specification v1.04.* http://www.ebxml.org/specs/ebTA.pdf

Glossary

ALE	Application Level Events
ANSI	American National Standards Institute
ASC	Accredited Standard Committee
ASN	Automatic Shipment Notice
DNS	Domain naming system
DoD	Department of Defense
EAN	European Article Number
ebXML	Electronic Business Extensible Markup Language
EDI	Electronics Data Interchange
EDIFACT	Electronic Data Interchange for Administration, Commerce and Transport
EPC	Electronic Product Code
EPC ISEPC	Information Service
ERP	Enterprise Resource Planning
FDA	Food and Drug Administration
GTIN	Global Trade Item Number
IP	Internet Protocol
MIT	Massachusetts Institute of Technology
NVM	Non volatile Memory
ONS	Object Naming Service
PML	Physical Markup Language
RFID	Radio Frequency Identification
SGTIN	Serialized Global Trade Identification Number
SKU	Stock Keeping Unit
TDI	Trade Data Interchange
UBL	Universal Business Language
UCC	Uniform Code Council
UPC	Universal Product Code
URI	Uniform Resource Identifier
VAN	Value-added network
XML	Extensible Markup Language

Appendix

Appendix 1 Selected EDI Message details

M	Mandatory
O	Optional
C	Conditional (the use of this segment is predicted on the existence of another element within the segment)
AN	Alphanumeric
DT	Date (YYYYMMDD)
ID	Identifier
Nn	Numeric with implied decimal
TM	Time

Section 1. 810 – Invoice

a) Segment

Segment	ST – Transaction Set Header		
Level	Heading		
Usage	Mandatory		
Purpose	To indicate the start of a transaction set and to assign a control number		
Example	ST*810*000000592		
Ref. ID	**Name**	**Feature**	**Comments**
ST01	Transaction Set ID	M ID 03/03	'810'
ST02	Transaction Set	M AN 04/09	A unique number assigned to each transaction set within a functional group. This number must match the value in SE02.

Segment	BIG – Beginning Segment for Invoice		
Level	Heading		
Usage	Mandatory		
Purpose	To indicate the beginning of an inventory advice transaction set and to transmit identifying numbers and dates		
Example	BIG*19991002*10580*****FB		
Ref. ID	**Name**	**Feature**	**Comments**
BIG01	Invoice Date	M DT 08/08	Statement date
BIG02	Invoice number	M AN 01/22	Unique invoice number
BIG03	Purchase Order Date	O DT 08/08	Not used
BIG04	Purse Order Number	O AN 01/22	Not used
BIG05	Release Number	O AN 01/30	Not used
BIG06	Change Order sequence number	O AN 01/08	Not used
BIG07	Transaction Type	O ID 02/02	"PB" – Partial Bill "FB" – Final Bill "N6" – Notice of settlement

Segment	N1 – Name		
Level	Header		
Usage	Conditional		
Purpose	To identify a party by type of organization, name and code		
Example	N1*BT*ACME & Co*ZZ*9876151 N1*RE*Bank of America		
Ref. ID	**Name**	**Feature**	**Comments**
N101	Entity ID code	M ID 02/03	"RE" – Remit to

			"BT" – Bill to
N102	Name	C AN 01/60	Free form name
N103	ID Code qualifier	C ID 01/02	"ZZ"
N104	ID Code	O ID 02/80	Customer reference number

Segment	N2 – Additional Name Information		
Level	Header		
Usage	Optional		
Purpose	To specify additional names or those longer than 35 characters in length		
Example	N2*%John Johnson & Sons		
Ref. ID	**Name**	**Feature**	**Comments**
N201	Name	M AN 1/60	Free form name

Segment	N3 – Address Information		
Level	Header		
Usage	Optional		
Purpose	To specify the location of the named party		
Example	N3*1850 Gateway Boulevard		
Ref. ID	**Name**	**Feature**	**Comments**
N301	Address	M AN 01/55	Free form address
N302	Address	O AN 01/55	Free form address

Segment	N4 – Geographic Location		
Level	Header		
Usage	Optional		
Purpose	To specify the geographic place of the named party		
Example	N4*BAKERSPIELD*CA*93306*US		
Ref. ID	**Name**	**Feature**	**Comments**
N401	City Name	O AN 02/30	
N402	State Code	O ID 02/02	
N403	Postal Code	O ID 03/15	
N404	Country Code	O ID 02/03	
N405	Location qualifier	C ID 01/02	Not used
N406	Location ID	O AN 01/30	Not used

Segment	ITD – Terms of sale/Deferred Terms of sale		
Level	Header		
Usage	Optional		
Purpose	To specify terms of sale		
Example	ITD*03*****19990102		
Ref. ID	**Name**	**Feature**	**Comments**

ITD01	Terms Type code	O ID 02/02	"03" – Fixed Date
ITD02	Terms basis data code	O ID 01/02	Not used
ITD03	Terms discount percent	O R 01/06	Not used
ITD04	Terms discount due date	C DT 08/08	Not used
ITD05	Terms discount days due	C NO 01/03	Not used
ITD06	Terms net due date	O DT 08/08	YYYYMMDD
ITD07-15			Not used

Segment	DTM – Date/Time reference		
Level	Header		
Usage	Optional		
Purpose	To specify pertinent dates and times		
Example	DTM*150*19990101***		
Ref. ID	**Name**	**Feature**	**Comments**
DTM01	Date/Time qualifier	M AN 03/03	"150" – Service period start "151" – Service period end
DTM02	Date	C DT 08/08	Date (YYYYMMDD)
DTM03	Time	C TM 04/08	Not used
DTM04	Time code	O ID 02/02	Not used
DTM05	Date Time period	C ID 02/03	Not used
DTM06	Date Time period	C AN 01/35	Not used

Segment	IT1 – Baseline Item Data		
Level	Detail		
Usage	Optional		
Purpose	To specify the basic and most frequently used line item data for the invoice and related transactions		
Example	IT1*1*1*EA*125**TP*0351		
Ref. ID	**Name**	**Feature**	**Comments**
IT101	Assigned ID	O AN 01/20	Line item count
IT102	Quantity Invoiced	C R 01/10	Number of units invoiced (always a whole number)
IT103	Unit of measure code	C ID 02/02	"EA"
IT104	Unit Price	C R 01/17	Used to show settlement charge. Credits are prefixed with a minus sign.
IT105	Basic unit price code	O ID 02/02	Not used.

IT106	***Product /Service ID qualifier***	*C ID 02/02*	***"BP" – Buyer's part number "UP" – U.P.C Consumer Package Code***
IT107	Product/Service IE	O AN 01/48	Settlement charge code
IT108-125			Not used.

Segment	PID – Product / Item description		
Level	Detail		
Usage	Optional		
Purpose	To describe a product or process in coded or freeform format		
Example	PID*X****Monthly Grid Management Charge due ISO		
Ref. ID	**Name**	**Feature**	**Comments**
PID01	Item description type	M ID 01/01	"X" – Semi-structured (code and text) "F" – Free form
PID02-04			Not used.
PID05	Description	C AN 01/80	Settlement charge description
PID06-09			Not used.

Segment	SAC – Service, Promotion, Allowance, or Charge information		
Level	Details		
Usage	Options		
Purpose	To request or identify a service, promotion, allowance, or charge; to specify the amount or percentage for the service, promotion allowance, or charge.		
Example	SAC*A*A400*248*******09***5409 ND ALLOWANCE		
Ref. ID	**Name**	**Feature**	**Comments**
SAC01	Allowance or charge indicator	M ID 01/01	"A" – Allowance "C" – Charge
SAC02	Service, Promotion, Allowance or Charge code	O ID 04/04	"A400" – Allowance non-performance "D2400" – Freight
SAC03-04			Not used
SAC05	Amount	O N2 01/15	Monetary amount
SAC06-11			Not used
SAC12	Allowance or charge method of handling case	O ID 02/02	"06" – Charge to be paid by customer "09" – Allowance to be issued by vendor
SAC13-14			Not used

SAC15	Description	O AN 01/80	A free-form description to clarity the related data elements and their contents

Segment	TDS – Carrier details (Routing sequence / transit time)		
Level	Summary		
Usage	Mandatory		
Purpose	To specify the total invoice discounts and amounts		
Example	TDS*48510		
Ref. ID	**Name**	**Feature**	**Comments**
TDS01	Total invoice amount	M N2 01/15	Total balance due
TDS02	Amount subject to terms discount	O N2 01/15	Not used
TDS03	Discounted amount due	O N2 01/15	Not used
TDS04	Terms discount amount	O N2 01/15	Not used

Segment	CAD – Carrier Detail		
Level	Summary		
Usage	Optional		
Purpose	To specify transportation details for the transaction		
Example	CAD****FDE*FEDERAL**BM*123456789		
Ref. ID	**Name**	**Feature**	**Comments**
CAD01-03			Not used
CAD04	Standard Carrier alpha code	O ID 02/04	
CAD05	Routing	O AN 01/35	Free form description of the routing or requested routing for shipment, or the originating carrier's identity
CAD06			Not used
CAD07	Reference identification qualifier	O ID 02/03	"BM" – Bill of landing number
CAD08	Reference identification	O AN 01/30	Reference information as defined for a particular Transaction Set or as specified by the Reference Identification Qualifier

Segment	ISS – Invoice shipment summary
Level	Summary
Usage	Optional

Purpose	To specify summary details of total items shipped in terms of quantity, weight, and volume		
Example	ISS*1*PC*2*PG		
Ref. ID	**Name**	**Feature**	**Comments**
ISS01	***Number of units shipped***	***O R 01/10***	***Numeric value of units shipped in manufacturer's shipping units for a line item or transaction set***
ISS02	Unit or basis for measurement code	O ID 02/02	"PC" – Piece
ISS03	Weight	O R 01/10	
ISS04	Unit or basis for measurement code	O ID 02/02	"PG" – Pounds Gross

Segment	CTT – Transaction Total		
Level	Summary		
Usage	Optional		
Purpose	To transmit a hash total for a specific element in the transaction set		
Example	CTT*1		
Ref. ID	**Name**	**Feature**	**Comments**
CTT01	Number of line items	M NO 01/06	Count of line items
CTT02	Hash totals	O R 01/10	Not used
CTT03	Weight	C R 01/10	Not used
CTT04	Unit of Measurement code	C ID 02/02	Not used
CTT05	Volume	C R 01/08	Not used
CTT06	Unit of Measurement code	C ID 02/02	Not used
CTT07	Description	O AN 01/80	Not used

Segment	SE – Transaction Set Trailer		
Level	Summary		
Usage	Mandatory		
Purpose	To indicate the end of the transaction set and provide the count of the transmitted segments including the Transaction Set Header (ST) and Trailer (SE) segments.		
Example	SE*23*000000592		
Ref. ID	**Name**	**Feature**	**Comments**
SE01	Number of included segments	M NO 01/10	Total number of segments in transaction set including ST and SE
SE02	Transaction set control number	M AN 04/09	Same as corresponding ST02

b) Sample code

ST*810*0001
BIG*19981231*10429*****FB
N1*BT*The Scheduling Coordinator, Inc
N3*53241 Hamilton Dr
N4*Palo Alto*CA*95622*US
N1*RE*Bank of America- (Mkt and GMC)
N3*1850 Gateway Boulevard
N4*Concord*CA*94520*US
REF*11*1233626208
REF*01*121000358
ITD*03*****19990107
DTM*150*19980930
DTM*151*19981031
IT1*1*1*EA*19406.14**TP*0101
PID*X****Day-Ahead Spinning Reserve due ISO
IT1*2*1*EA*1764.26**TP*0102
PID*X****Day-Ahead Non-Spinning Reserve due ISO
IT1*3*1*EA*9774.42**TP*0103
PID*X****Day-Ahead AGC/Regulation due ISO
TDS*21351663
CTT*16
SE*48*0001

Section 2. 850 – Purchase Order

a) Segment

Segment	ST – Transaction Set Header		
Level	Heading		
Usage	Mandatory		
Purpose	To indicate the start of a transaction set and to assign a control number		
Example	ST*850*2344		
Ref. ID	**Name**	**Feature**	**Comments**
ST01	Transaction Set Identifier Code	M ID 03/03	'850'
ST02	Transaction set control number	M AN 04/09	Unique number assigned by the originator for a transaction set

Segment	BEG – Beginning segment for Purchase Order		
Level	Heading		
Usage	Mandatory		
Purpose	To indicate the beginning of the Purchase Order Transaction Set and transmit identifying numbers and dates		
Example	BEG*00*SA*BYX12345**19991201		
Ref. ID	**Name**	**Feature**	**Comments**
BEG01	Transaction Set Purpose code	M ID 02/02	"00" – Original "06" – Confirmation
BEG02	Purchase Order Type code	M ID 02/02	"CF" – Confirmation "NE" – New Order "SA" – Stand-alone order
BEG03	Purchase Order Number	M AN 01/22	Identifying number for Purchase Order assigned by the purchaser
BEG04	Release Number	O AN 01/30	Identifying a release against a Purchase Order previously placed by the parties involved in the transaction
BEG05	Date	M DT 08/08	YYYYMMDD
BEG06	Contract Number	O AN 01/30	Contract number

Segment	REF – Reference identification		
Level	Heading		
Usage	Optional		
Purpose	To specify identifying information		
Example	REF*DP*678901		
Ref. ID	**Name**	**Feature**	**Comments**
REF01	Reference Identification Qualifier	M ID 02/03	"DP" – Department Number "IA" – Internal vendor number "PD" – Promotion/Deal number

REF02	Reference Identification	O AN 01/30	Reference information as defined for a particular transaction set

Segment	PER – Administrative Communications Contract		
Level	Heading		
Usage	Optional		
Purpose	To identify a person or office to whom administrative communication should be directed		
Example	PER*BD*JANE SMITHE*TE*6517853691		
Ref. ID	**Name**	**Feature**	**Comments**
PER01	Contact Function Code	M ID 02/02	"BD" – Buyer Name or Department "CW" – Confirmed with "RE" – Receiving contact
REF02	Name	O AN 01/60	Free form name
REF03	Communication Number Qualifier	C ID 02/02	"EM" – Electronic mail "FX" – Fax "TE" – Telephone
REF03	Communication Number	C AN 01/80	

Segment	FOB – F.O.B related instruction		
Level	Heading		
Usage	Optional		
Purpose	To specify transportation instructions relating to shipment		
Example	FOB*CC		
Ref. ID	**Name**	**Feature**	**Comments**
FOB01	Shipment Method of payment	M ID 02/02	"CC" – Collect "CF" – Collect, Freight credited bank to customer "PC" – Prepaid but charged to customer "PP" – Prepaid (by seller) "PU" – Pickup
FOB02	Location Qualifier	C ID 01/02	"DE" – Destination (shipping) "OR" – Origin (shipping point)
FOB03	Description	O AN 01/80	Free-form

Segment	CSH – Sales Requirements		
Level	Heading		
Usage	Optional		
Purpose	To specify general conditions or requirement of the sale		
Example	CSH*N		
Ref. ID	**Name**	**Feature**	**Comments**

CSH01	Sales Requirement Code	O ID 01/02	"N" – No back order "SC" – Ship complete "SP" – Ship partial, balance cancel "Y" – Back order if out of stock

Segment	SAC – Service, Promotion, Allowance, or Charge information		
Level	Heading		
Usage	Optional		
Purpose	To request or identify a service, promotion, allowance or charge; to specify the amount or percentage for the service, promotion, allowance, or charge		
Example	SAC*A******5*****02		
Ref. ID	**Name**	**Feature**	**Comments**
SAC01	Allowance or charge indicator	M ID 01/01	"A" – Allowance "C" – Charge
SAC02-06			Not used
SAC07	Percent	C R 01/06	
SAC08-11			Not used
SAC12	Allowance or charge method of handling code		"02" – Off invoice

Segment	ITD – Terms of sale / Deferred Terms of sale		
Level	Heading		
Usage	Optional		
Purpose	To specify terms of sale		
Example	ITD*08*3*2**10**30		
Ref. ID	**Name**	**Feature**	**Comments**
ITD01	Terms Type Code	O ID 02/02	"08" – Basic discount offered
ITD02	Terms basis data code	O ID 01/02	"3" – Invoice date
ITD03	Terms discount percent	O R 01/06	
ITD04			Not used
ITD05	Terms discount days due	C N0 01/03	
ITD06			Not used
ITD07	Terms net days	O N0 01/03	

Segment	DTM – Data / Time reference
Level	Heading
Usage	Optional

Purpose	To specify pertinent dates and times		
Example	DTM*002*19991205		
Ref. ID	**Name**	**Feature**	**Comments**
DTM01	Data/Time qualifier	M ID 03/03	"002" – Delivery requested
DTM02	Date	C DT 08/08	YYYYMMDD

Segment	TD5 – Carrier Details (Routing Sequence / Transit Time)		
Level	Heading		
Usage	Optional		
Purpose	To specify the carrier and sequence of routing and provide transit time information		
Example	TD5*O***AE		
Ref. ID	**Name**	**Feature**	**Comments**
TD501	Routing sequence code	O ID 01/02	"O" – Original carrier
TD502-03			Not used
TD504	Transportation method/type code	C ID 01/02	"AE" – Air express

Segment	MAN – Marks and Numbers		
Level	Heading		
Usage	Optional		
Purpose	To indicate identifying marks and numbers for shipping containers		
Example	MAN*R*REQ#:12346		
Ref. ID	**Name**	**Feature**	**Comments**
MAN01	Marks and Numbers Qualifier	M ID 01/02	"R" – Originator assigned
MAN02	Marks and Numbers	M AN 01/48	

Segment	N1 – Name		
Level	Heading		
Usage	Optional		
Purpose	To identify a party by type of organization, name and code		
Example	N1*ST**1*987654321		
Ref. ID	**Name**	**Feature**	**Comments**
N101	Entity identifier code	M ID 02/03	"BS" – Bill and ship to "BT" – Bill to party "ST" – Ship to
N102	Name	C AN 01/60	Free form
N103	Identification code	C ID 01/02	

	qualifier		
N104	Identification Code	C AN 02/80	

Segment	N2 – Additional Name Information		
Level	Heading		
Usage	Optional		
Purpose	To specify additional names or those longer than 35 characters in length		
Example	N2*C/O ANDERSON CORP		
Ref. ID	**Name**	**Feature**	**Comments**
N201	Name	M AN 01/60	Free form
N202	Name	O AN 01/60	Free form

Segment	N3 – Address information		
Level	Heading		
Usage	Optional		
Purpose	To specify the location of the named party		
Example	N3*1513 E. 5 AVE		
Ref. ID	**Name**	**Feature**	**Comments**
N301	Address Information	M AN 01/55	Free form
N302	Address Information	O AN 01/55	Free form

Segment	N4 – Geographic Location		
Level	Heading		
Usage	Optional		
Purpose	To specify the geographic place of the named party		
Example	N4*NEW YORK*NY*012345811		
Ref. ID	**Name**	**Feature**	**Comments**
N401	City Name	O AN 02/30	Free form
N402	State or Province Code	O ID 02/02	State code
N403	Postal Code	O ID 03/15	
N404	Country Code	O ID 02/03	

Segment	PO1 – Baseline Item Data		
Level	Detail		
Usage	Mandatory		
Purpose	To specify basic and most frequently used line item data		
Example	PO1*1*800*EA*1.5**UK*00012223451231*IN*22345		
Ref. ID	**Name**	**Feature**	**Comments**

PO101	Assigned Identification	O AN 01/20	Alphanumeric characters assigned for differentiation within a transaction set
PO102	Quantity Ordered	C R 01/15	
PO103	Unit or Basis for Measurement Code	O ID 02/02	"BX" – Box "CA" – Case "EA" – Each
PO104	Unit Price	C R 01/17	
PO105	Basis of Unit Price Code	O ID 02/02	"PE" – Price each
PO106	***Product/Service ID qualifier***	***C ID 02/02***	***"UA" – UPC/EAN Case Code*** ***"UD" – UPC/EAN Consumer Package Code*** ***"UE" – UPC/EAN Module Code*** ***"UI" – UPC Consumer Package Code*** ***"UK" – UPC/EAN Shipping Container Code***
PO107	Product/Service ID	C AN 01/48	Identification number for a product or service

Segment	PID – Product Item Description		
Level	Detail		
Usage	Optional		
Purpose	To describe a product or process in coded or free-form format		
Example	PID*F****471 PLSTC TP WHITE ½ X 36 YD BULK		
Ref. ID	**Name**	**Feature**	**Comments**
PID01	Item description type	M ID 01/01	"F" – Free form
PID02-04			Not used
PID05	Description	C AN 01/80	Free form

Segment	REF – Reference Identification		
Level	Detail		
Usage	Option		
Purpose	To specify identifying information		
Example	REF*CT*XZ234-A		
Ref. ID	**Name**	**Feature**	**Comments**
RFF01	Reference identification qualifier	M ID 02/03	"CT" – Contract Number "PR" – Price Quote Number "WS" – Warehouse storage location number
RFE02	Reference identification	C AN 01/30	Reference information as defined for a particular Transaction Set

Segment	SDQ – Destination Quantity		
Level	Detail		
Usage	Optional		
Purpose	To specify destination and quantity detail		
Example	SDQ*EA*92*69283*100*83115*50*77721*100		
Ref. ID	**Name**	**Feature**	**Comments**
SDQ01	Unit or Basis for measurement code	M ID 02/02	"CA" – Case "EA" – Each "PK" – Package
SDQ02	Identification Code Qualifier	O ID 01/02	"92" – Assigned by Buyer
SDQ03	Identification Code	M AN 02/80	
SDQ04	Quantity	M R 01/15	

Segment	SCH – Line Item schedule		
Level	Detail		
Usage	Optional		
Purpose	To specify the data for scheduling a specific line-item		
Example	SCH*50*EA***002*19991215		
Ref. ID	**Name**	**Feature**	**Comments**
SCH01	Quantity	M R 01/15	
SCH02	Unit or Basis Measurement Code	M ID 02/02	"CA" – Case "EA" – Each "PK" – Package
SCH03-04			Not used
SCH05	Date/Time qualifier	M ID 03/03	"002" – Delivery requested
SCH06	Date	M DT 08/08	YYYYMMDD

Segment	CTT – Transaction Totals		
Level	Summary		
Usage	Optional		
Purpose	To transmit a hash total for a specific element in the transaction set		
Example	CTT*3		
Ref. ID	**Name**	**Feature**	**Comments**
CTT01	Number of Line Items	M N0 01/06	Total number of line items in the transaction set

Segment	AMT – Monetary Amount
Level	Summary
Usage	Optional

Purpose	To indicate the total monetary amount		
Example	AMT*TT*241.74		
Ref. ID	**Name**	**Feature**	**Comments**
AMT01	Amount Qualifier Code	M ID 01/03	"TT" – Total Transaction amount
AMT02	Monetary Amount	M R 01/08	

Segment	SE – Transaction Set Trailer		
Level	Summary		
Usage	Mandatory		
Purpose	To indicate the end of the transaction set and provide the count of the transmitted segments (including the beginning (ST) and ending (SE) segments)		
Example	SE*10*2344		
Ref. ID	**Name**	**Feature**	**Comments**
SE01	Number of included segments	M N0 01/10	Total number of segments included in a transaction set including ST and SE segments
SE02	Transaction Set Control Number	M AN 04/09	Identifying control number that must be unique within the transaction set functional group assigned by the originator for a transaction set

b) Sample code

ST*850*4462
BEG*00*SA*ABC99876**19991001
N1*BS**1*992345123
PO1*1*800*EA*1.5**UI*01222345123*IN*22345
SCH*200*EA***002*19991015
SCH*400*EA***002*19991115
SCH*200*EA***002*19991215
PO1*2*100*CA*22.34**UI*01222366678*IN*55556
SCH*50*EA***002*19991015
SCH*50*EA***002*19991215
PO1*3*24*EA*22.95**UI*01222377123*IN*66667-1
SCH*24*EA***002*19991015
CTT*3
SE*14*4462

Section 3. 856 – Ship Notice

a) Segment

Segment	ST – Transaction Set Header		
Level	Heading		
Usage	Mandatory		
Purpose	To indicate the start of a transaction set and to assign a control number		
Example	ST*856*000006359		
Ref. ID	**Name**	**Feature**	**Comments**
ST01	Transaction Set Identifier Code	M ID 03/03	'856' – Ship Notice
ST02	Transaction Set Control Number	M AN 04/09	Identifying control number that must be unique within the transaction set functional group assigned by the originator for a transaction set

Segment	BSN – Beginning Segment for ship notice		
Level	Heading		
Usage	Mandatory		
Purpose	To transmit identifying numbers, dates, and other basic data relating to the transaction set		
Example	BSN*00*AX13131*19991209*1241		
Ref. ID	**Name**	**Feature**	**Comments**
BSN01	Transaction Set Purpose Code	M ID 02/02	"00" – Original
BSN02	Shipment Identification	M AN 02/30	A unique control number assigned by the original shipper to identify a specific shipment
BSN03	Date	M DT 08/08	YYYYMMDD
BSN04	Time	M TM 04/08	HHMM

Segment	HL – Hierarchical Level – Shipment Level		
Level	Detail		
Usage	Mandatory		
Purpose	To identify dependencies among and the content of hierarchically related groups of data segments		
Example	HL*1**S		
Ref. ID	**Name**	**Feature**	**Comments**
HL01	Hierarchical ID Number	M AN 01/12	A unique number assigned by the sender to identify a particular data segment in a hierarchical structure
HL02			Not used
HL03	Hierarchical Level	M ID 01/02	"S" – Shipment

	Code		

Segment	TD1 – Carrier Details (Quantity and Weight)		
Level	Detail		
Usage	Optional		
Purpose	To specify the transportation details relative to commodity, weight and quantity		
Example	TD1**25*****79*LB		
Ref. ID	**Name**	**Feature**	**Comments**
TD101			Not used
TD102	Lading quantity	C N0 01/07	Number of units of the lading commodity
TD103-06			Not used
TD107	Weight	C R 01/10	
TD108	Unit or Basis for measurement code	C ID 02/02	"LB" – Pound

Segment	TD5 – Carrier details (routing sequence / transit time)		
Level	Detail		
Usage	Optional		
Purpose	To specify the carrier and sequence of routing and provide transit time information		
Example	TD5**2PRES*LT*PRESTON TRUCKING COMPANY		
Ref. ID	**Name**	**Feature**	**Comments**
TD501			Not used
TD502	Identification Code qualifier	C ID 01/02	"2" – Standard Carrier Alpha Code (SCAC)
TD503	Identification Code	C AN 02/80	
TD504	Transportation Method/Type Code	C ID 01/02	"LT" _ Less than trailer load (LTL)
TD505	Routing	C AN 01/35	Free form description of the routing or requested routing for shipment

Segment	REF – Reference identification		
Level	Detail		
Usage	Optional		
Purpose	To specify identifying information		
Example	REF*CN*516646432		
Ref. ID	**Name**	**Feature**	**Comments**
REF01	Reference identification	M ID 02/03	"BM" – Bill of lading number "CN" – Carrier's reference number

	qualifier		
REF02	Reference identification	C AN 01/30	

Segment	DTM – Date / Time reference		
Level	Detail		
Usage	Optional		
Purpose	To specify pertinent date and times		
Example	DTM*011*19991209		
Ref. ID	**Name**	**Feature**	**Comments**
DTM01	Date/Time Qualifier	M ID 03/03	"011" – Shipped
DTM02	Date	C DT 08/08	YYYYMMDD

Segment	N1 – Name		
Level	Detail		
Usage	Optional		
Purpose	To identify a party by type of organization, name and code		
Example	N1*ST**92*0024		
Ref. ID	**Name**	**Feature**	**Comments**
N101	Entity Identifier code	M ID 02/03	"SF" – Ship from "ST" – Ship to
N102	Name	C AN 01/60	
N103	Identification Code Qualifier	C ID 01/02	"91" – Assigned by Seller
N104	Identification code	X AN 02/80	Code identifying a party or other code

Segment	N3 – Address information		
Level	Detail		
Usage	Optional		
Purpose	To specify the location of the named party		
Example	N3*200 N. MAIN STREET		
Ref. ID	**Name**	**Feature**	**Comments**
N301	Address Information	M AN 01/55	

Segment	N4 – Geographic Location		
Level	Detail		
Usage	Optional		
Purpose	To specify the geographic place of the named party		
Example	N4*ST. Paul*MN*551441000		
Ref. ID	**Name**	**Feature**	**Comments**

N401	City Name	O AN 02/30	
N402	State Code	O ID 02/02	
N403	Postal Code	O ID 03/15	

Segment	HL – Hierarchical Level – Order Level		
Level	Detail		
Usage	Mandatory		
Purpose	To identify dependencies among and the content of hierarchically related groups of data segments		
Example	HL*2*1*O		
Ref. ID	**Name**	**Feature**	**Comments**
HL01	Hierarchical ID number	M AN 01/12	
HL02	Hierarchical Parent ID number	O AN 01/12	
HL03	Hierarchical Level Code	M ID 01/02	"O" – Order

Segment	PRF – Purchase Order Reference		
Level	Detail		
Usage	Optional		
Purpose	To provide reference to a specific Purchase Order		
Example	PRF*123456***19991207		
Ref. ID	**Name**	**Feature**	**Comments**
PRF01	***Purchase Order Number***	***M AN 01/22***	
PRF02-03			Not used
PRF04	Date	O DT 08/08	YYYYMMDD

Segment	REF – Reference Identification		
Level	Detail		
Usage	Optional		
Purpose	To specify identifying information		
Example	REF*IV*EM00001		
Ref. ID	**Name**	**Feature**	**Comments**
REF01	Reference Identification Qualifier	M ID 02/03	"IV" – Seller's invoice number
REF02	Reference Identification	X AN 01/30	

Segment	HL – Hierarchical Level – Tare Level		
Level	Detail		
Usage	Mandatory		
Purpose	To identify dependencies among and the content of hierarchically related groups of data segments		
Example	HL*3*2*T		
Ref. ID	**Name**	**Feature**	**Comments**
HL01	Hierarchical ID Number	M AN 01/12	
HL02	Hierarchical Parent ID number	O AN 01/12	
HL03	Hierarchical Level code	M ID 01/02	"T" – Shipping Tare

Segment	MAN – Marks and Numbers		
Level	Detail		
Usage	Optional		
Purpose	To indicate identifying marks and numbers for shipping containers		
Example	MAN*GM*00100212004800015035		
Ref. ID	**Name**	**Feature**	**Comments**
MAN01	Marks and Numbers Qualifier	M ID 01/02	"GM"- SSCC-18 and Application Identifier
MAN02	Marks and Numbers	M AN 01/48	

Segment	HL – Hierarchical Level – Item Level		
Level	Detail		
Usage	Mandatory		
Purpose	To identify dependencies among and the content of hierarchically related groups of data segments		
Example	HL*4*3*I		
Ref. ID	**Name**	**Feature**	**Comments**
HL01	Hierarchical ID number	M AN 01/12	
HL02	Hierarchical Parent ID Number	O AN 01/12	
HL03	Hierarchical Level code	M ID 01/02	"I" – Item

Segment	LIN – Item identification
Level	Detail
Usage	Optional
Purpose	To specify basic item identification

Example	LIN*001*UP*021200002137*BP*22345		
Ref. ID	**Name**	**Feature**	**Comments**
LIN01	Assigned Identification	O AN 01/20	
LIN02	***Product/Service ID qualifier***	***M ID 02/02***	***"UI" – UPC Consumer Package Code*** ***"UK" – UPC/EAN Shipping Container code*** ***"SK" – Stock keeping unit (SKU)*** ***"UP" – UPC Consumer Packeage Code (1-5-5-1 format)***

Segment	SN1 – Item Detail (shipment)		
Level	Detail		
Usage	Optional		
Purpose	To specify line-item detail relative to shipment		
Example	SN1**20*CA		
Ref. ID	**Name**	**Feature**	**Comments**
SN101			Not used
SN102	Number of units shipped	M R 01/10	
SN103	***Unit or basis for measurement code***	***M ID 02/02***	***"CA" – Case*** ***"EA" – Each*** ***"PK" – Package***

Segment	CTT – Transaction Totals		
Level	Summary		
Usage	Optional		
Purpose	To transmit a hash total for a specific element in the transaction set		
Example	CTT*5		
Ref. ID	**Name**	**Feature**	**Comments**
CTT01	Number of line items	M N0 01/06	

Segment	SE – Transaction Set Trailer		
Level	Summary		
Usage	Mandatory		
Purpose	To indicate the end of the transaction set and provide the count of the transmitted segments (including the beginning (ST) and ending (SE) segments)		
Example	SE*33*000006359		
Ref. ID	**Name**	**Feature**	**Comments**

SE01	Number of included segments	M N0 01/10	Total number of segments included in a transaction set including ST and SE segments
SE02	Transaction set control number	M AN 04/09	Identifying control number that must be unique within the transaction set functional group assigned by the originator for a transaction set

b) Sample code

ST*856*000006359
BSN*00*OAND754272*19991209*1241
HL*1**S
TD1**25*****79*LB
TD5**2*PRES*LT*PRESTON TRUCKING COMPANY
REF*BM*OAND754272
REF*CN*516646432
DTM*011*19991209
N1*SF*3M – ONTARIO
N3*5151 PHILADELPHIA
N4*ONTARIO*CA*917612814
N1*ST**92*0024
HL*2*1*O
PRF*123456***19991207
REF*IV*EM00001
HL*3*2*T
MAN*GM*00100212004800015035
HL*4*3*I
LIN*001*UP*021200002137*BP*22345
SN1**15*CA
HL*5*1*O
PRF*123699***19991208
REF*IV*EM00025
HL*6*5*T
MAN*GM*00100212004654892603
HL*7*6*I
LIN*002*UP*021200010729*BP*10795
SN1**27*EA
HL*8*6*I
LIN*003*UP*021200010736*BP*43666
SN1**10*RL
CTT*8
SE*33*000006359