Hybrid Peer-to-Peer Architecture in Support of Supply Chain Visibility

Zhijie Li 1, Haowan Wu1, Alvaro Esperanca1,

Jeffrey Tazelaar2, John Wassick2, Zina Ben Miled1

1 Electrical and Computer Engineering Department, Purdue School of Engineering and Technology, IUPUI

2The Dow Chemical Company

Abstract

???? 2 to 3 paragraphs that you can add once you finish the paper

1. Introduction

Why is this needed – what challenges are you trying to address …..

Move from transactional system to event based system

What is value added network what is electronic data interchange

a real-time information layer is needed to overcome these limitations

xml vs json

xml good for document exchange not suitable for the amount of traffic involved in this project

json lightweight more scalable. Object oriented .This is particularly important when the granularity of the data exchanged moves from the container or pallet level to the individual product level.

(1 page)

1. Related work

2 pages – related work – other systems that are similar (what is the same and what is better)

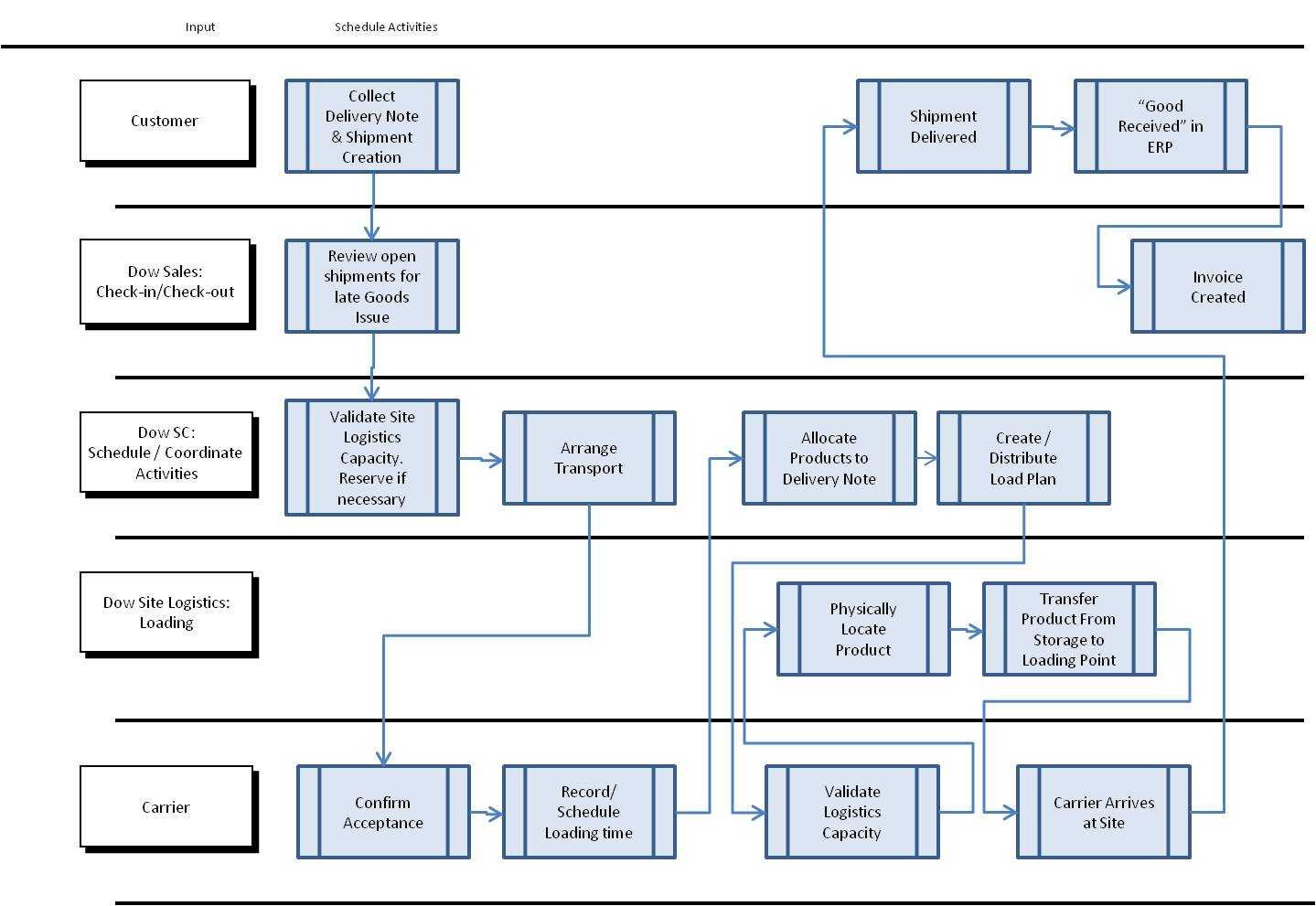
Part 1 – exiting SC systems and why is your approach better

Example: Ariba Business Network from SAP, GT Nexus, Elemica, E2Open, etc. – You need to be familiar with these

Part 2 - Sensor Aggregation & IOT (ThingWorx, Siemens – Team Center - Google Cloud Platform , ….

Part 3 – Peer to Peer systems (e.g. Napster, and others – Eric we covered these in class ) how does your approach outperform others and how is it similar

### *Figure 1: Physical Distribution Complexities*



Part 4 – dynamic user centric P2P network or hybrid P2P networks (e.g. paper that I gave you Fortes – Figuiredo and others)

1. Physical Distribution

What is physical distribution

1. Solution Overview

### 

**Figure 1.** this figure is too simple and does not add any information – start with the one that you drew on the board in SL 113

The proposed physical distribution system is based a hybrid peer-to-peer architecture that is customized dynamically for each stakeholder. The SCV (give a name – so we do not called the proposed system throughout) is based on a collection of purpose-centric customized networks that can be configured dynamically on the fly. This is a departure from the traditional transaction based EDI exchange systems. SCV allows stakeholders to share information related to a given shipment in real time.

4.1 Roles and Configurations

There are three main stakeholders for a given shipment. These roles consists of Customer, Carrier, Supplier. Obviously a given entity may assume more than one role over several shipments. For instance, a carrier in one shipment can also be a customer in other shipment. However, for a given shipment these roles are in general distinct.

4.2 Modules

Describe each module in your system and its functionality – also include a detailed diagram that shows all the modules –

The overview of the model is shown in Figure ????. The main modules include the Index Server, …., …..

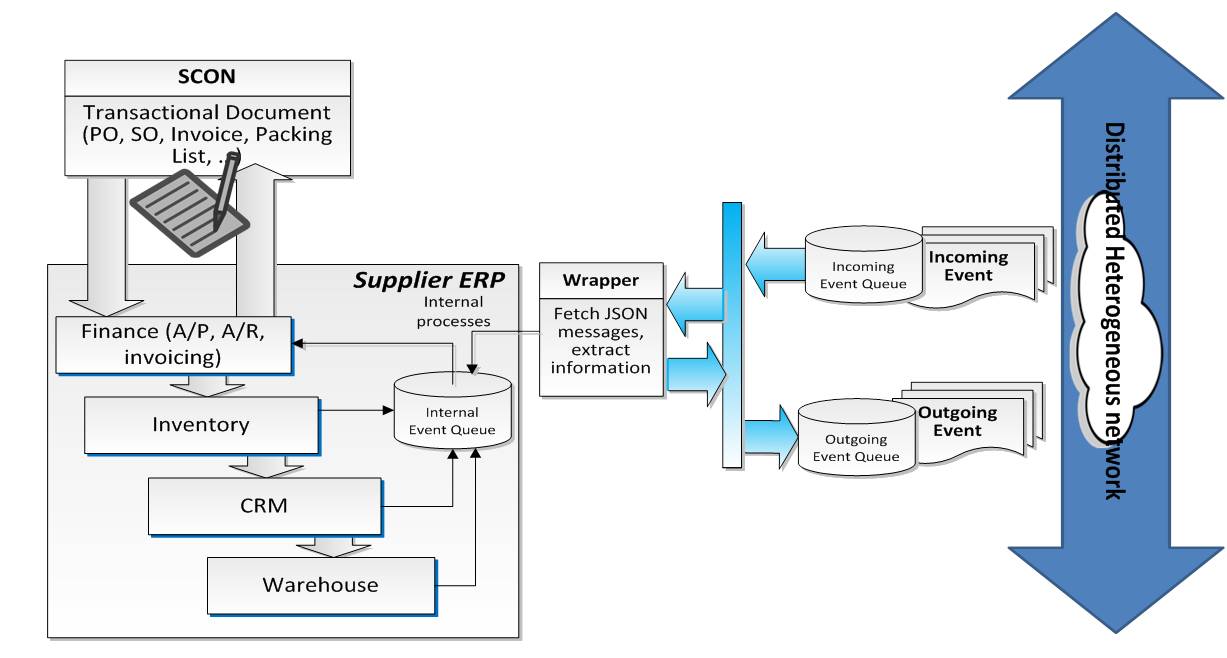
1. Index Server

Index server acts as a yellow page. It holds all clients’ information such as IP address, clients’ names. When a client wants to communication with others, it will first query index server for the information of the target client. How ? what is the process flow ? why do you have to maintain an index server ? Client will update their IP address regularly. How and why can’t you maintain static IP – mobile client – etc…

Each client will act as customer or carrier or supplier. The communication procedure is shown on the following picture.

### 

**Figure 2.** same comment as for Figure 1



This model is based on the assumption that the order information has already been initialized by each company’s ERP (enterprise resource planning) and the order information is already in our MongoDB. The order information is shown in appendix. The communication is initialized by the supplier. The supplier will establish a TCP connection with customer and carrier. It will update the shipment information to carrier and customer when a new signal ???? arrives. The signal is sent by some integrated sensor network such as bar code scanner. Our software will broadcast to all parties that involved in the shipment, once it receives the signal. It will keep update the status until carrier picks up. The carrier will receive GPS signal from the track and it will use the same mechanism to broadcast to other parties. The customer will be required to verify the status of the shipment and manually submit the final acceptance of the shipment once the order is delivered.

Client queries index server about others’ IP address according to the information contained in Order struct which will be explained later in this paper. There are few advantage of adding one index server in to the system. Firstly it can reduce the amount of network traffic. one client has to send IP address update message to all clients’ in the network if every client stores each other’s IP address in its own database, instead of doing this, we have index server to handle all the client information related communication, so client only have to send one message to index server to update its’ information. Secondly this structure can help protect the privacy of each company’s information. Company’s information will only be accessible by index server and the parties that are involved in one transaction. Other parties can only access the information while a shipment is taking place. The information will not be stored in any place other than index server.

Peer to peer

This system uses peer to peer architecture for the communication between clients. Peer to peer architecture can reduce network traffic , since communication are between clients, there is no server interfere in between the communication. Using peer to peer architecture can avoid single point failure, even one client goes down the communication will continue. Once the client resumes online it will receive the newest information from other clients. In addition it can reduce data storage cost. Since the data is stored locally in every client. There is no need to establish a centralized server storage. I

1. Client Application

Client side software uses a 3 tier architecture. The first layer is presentation layer. This layer is generally known as user interface which displays the information user need to know. Second layer is application layer. It is the layer that connects presentation layer and data layer. User can use presentation layer to send commands to application layer, application layer will handle all kinds of requests such as confirming the delivery of a shipment. The third layer is the data layer. It holds all information of the client such as order/shipment information. Using this three layer design can first separate each individual part clearly and makes it easy to modify each part. The change of one layer does not affect the others. Security is enhanced by this model as well. The only way to access data layer is through application layer. This can ensure the safety of the data.

You need to add a detailed diagram – and explain each layer in details – block diagram and description

When I am done reading the paper I should have a blue print of the system that you are proposing

**4.3 Data exchange template**

How is data exchanged and what is its format – content

**4.4 Example Scenario**

1. Conclusion

The proposed platfo …..

References