Information visualization for business applications

Ramesh Gopalakrishnan

Follow this and additional works at: https://scholarworks.rit.edu/theses

Recommended Citation

This Thesis is brought to you for free and open access by RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.
Rochester Institute of Technology
B. Thomas Golisano College
of
Computing and Information Sciences
Master of Science in Information Technology

Thesis Approval Form

Student Name: Ramesh Gopalakrishnan

Thesis Title: Information Visualization for Business Applications

Thesis Committee

Name | Signature | Date
--- | --- | ---
Prof. Elouise Oyzon, M.S. (RIT, NY) | Elouise R. Oyzon 3/1/08 | Chair
Jonathan Schull | 3/13/08 | Committee Member

Dr. Jonathan Schull, Ph.D. (RIT, NY)

Dr. Tao Lin, Ph.D. (SAP Research, CA)

Tao Lin 03/07/08

Ramesh Gopalakrishnan 1 Winter 2007
Thesis Reproduction Permission Form

Rochester Institute of Technology

B. Thomas Golisano College of Computing and Information Sciences

Master of Science in Information Technology

Information Visualization for Business Applications

I, Ramesh Gopalakrishnan, hereby grant permission to the Wallace Library of the Rochester Institute of Technology to reproduce my thesis in whole or in part. Any reproduction must not be for commercial use or profit.

Date: 03/07/2008

Signature of Author: Ramesh Gopalakrishnan
# TABLE OF CONTENTS

1. Abstract.................................................................6
2. Hypotheses.............................................................7
3. Introduction...........................................................8
4. Introduction to Supply Chain Management.........................9
5. Supply Chain Process................................................10
6. Complexity of Supply Chain Data..................................11
   6.1. Distribution Network Configuration.........................11
   6.2. Information Flow...............................................11
   6.3. Inventory Management........................................11
   6.4. Manufacturing Flow Management............................11
   6.5. Sales Order Fulfillment Process............................11
7. Use Cases for an SCM Application................................12
8. Case Study on Manufacturers Supply Chain Application..........13
9. Research Study 1: Microsoft Dynamics SCM.................................................14
10. Research Study 2: SAP SCM..........................................21
11. Research Implementation..........................................23
   11.1. Introduction..................................................23
       11.1.1. Technical Problems Addressed........................23
       11.1.2. Innovations for this Implementation..................25
       11.1.3. Related Work...........................................25
   11.2. Scenario: Real Time Manufactory Monitoring..............27
       11.2.1. Global Supply Chain Mapping Layer..................27
       11.2.2. Manufacturer Shop Floor...............................35
       11.2.3. Assembly Line............................................40
   11.3. Architecture: Web-based Information System for Manufacturer.................44
       11.4.1. Introduction.............................................44
       11.4.2. Technologies..............................................45
       11.4.3. Dynamic Visualization Framework.....................49
       11.4.4. MVC Design Pattern....................................50
12. Discussions........................................................51
13. Future Roadmap......................................................54
14. Conclusion..........................................................55
15. Selected Code from MVC........................................56
16. Supplemental Storage Media......................................60
17. References..........................................................61
1. Abstract:

Business applications are generating ever-increasing amounts of data, and more and more of this data are in real-time data. Data mining is a standard feature in most data management system for retrieving complex information but the challenge is about how to represent these information in an effective way for better analysis and decision making. With this rich information, the challenge still remains to derive the most business value from ever-increasing amount of available information. Information visualization can significantly solve this problem for handling such complex and high volumes of data to be represented in business applications. Through this research, an attempt will be made to solve this problem by prototyping an interactive data driven business application through visualization. The key focus of this research work will be given in the “Presentation Tier” of a business application for visually representing the information in an aesthetic manner for better understanding.
2. Hypotheses:

2.1. Alternative Hypothesis \((H_1)\): Developing a Business Application based on Information Visualization has an effect on understanding the complex information and decision making.
3. Introduction:

The world is about business. Business application is a vast domain in Information Technology from a simple online flight reservation system to a complicated global supply-chain management application. But the problem that remains in common for all the applications is to better understand the information and get a clear visibility on how things are related with each other. This research aims at solving a complicated problem and will investigate on how effectively we can build these applications with visualization as a primary tool. With these objectives in consideration, this case study will be a prototype development of a supply-chain application which is the most complex application for the business people that deals with enormous amount of information exchange. A thorough research will be made on an existing supply-chain application used in the current business world and investigate on how we can enhance this application through visualization for better visibility. Problems involved in understanding a flat text based application for business analysis will also be researched. A web-based information system will be developed to demonstrate the results of this case study.
4. Introduction to Supply Chain Management (SCM):

A supply chain, logistics network, or supply network management is a coordinated system of organizations, people, activities, information and resources involved in moving a product or providing service in physical or virtual manner from supplier to customer. Supply chain activities transform raw materials and components into a finished product that is delivered to the end customer.

“Supply chain management (SCM) is the oversight of materials, information, and finances as they move in a process from supplier to manufacturer to wholesaler to retailer to consumer. Supply chain management involves coordinating and integrating these flows both within and among companies. It is said that the ultimate goal of any effective supply chain management system is to reduce inventory.”

[ http://searchcio.techtarget.com/sDefinition ]
5. Supply Chain Process:

A typical product flow involved in a supply chain process is illustrated below:

```
Suppliers
  ↓
Manufacturers
  ↓
Distribution Centers
  ↓
Retailers
  ↓
Consumers
```

For example, if a consumer wants to purchase a refrigerator, the entire process begins with a supplier. A supplier is the one who manufactures the parts or raw materials like cabinets, cooling parts, condenser etc. Once all the required parts are obtained, the process moves to the manufacturer. A manufacturer is the one who assembles all the individual parts together in his assembly line to come up with a unique product, which in this example is a refrigerator. After the product is successfully assembled in the manufacturer's assembly line, the product is moved to a distribution center or a warehouse. The product is stored in the distribution center for a few days or months depending on the retailer's requirement. Once the retailer requests for a particular refrigerator model, the item is shipped to the requested Retailer. Eventually the consumer obtains his requested model. This is how a product flows from a supplier to a customer in the supply chain process for any industry. In certain cases, this process could also be bi-directional.
6. Complexity of Supply Chain Data:

During the above mentioned product flow, enormous amounts of information or data are generated in each individual process. For example, the following could be a small list of process that might occur when a product moves to a Manufacturer: New sales order generated, check item availability, check parts availability, request supplier, product assembled, pick, pack, and ship to the distribution center. Each of these processes will further generate data based on the date, time and description of the process. These massive amounts of supply chain data are very crucial for the business people to better understand them and make decisions. They should be organized and made visible to overcome the following fundamental problems of developing a supply chain management application:

6.1. Distribution Network Configuration: Number and locations of suppliers, production facilities, distribution centers, warehouses and customers.

6.2. Information Flow: Integrate systems and processes through the supply chain to share valuable information, including demand signals, forecasts, simulating manufacturer’s shop floor and logistics etc.

6.3. Inventory Management: Quantity and location of inventory from the global suppliers including raw materials, work-in process and finished goods.

6.4. Manufacturing Flow Management: A process that includes all activities necessary to move products through the plants and to obtain, implement and manage manufacturing flexibility in the supply chain.

6.5. Sales Order Fulfillment Process: All activities necessary to define customer requirement and to design a process to meet customer’s request.
7. Use Cases for an SCM Application:

The following set of major use case criteria will be considered for this research study. These criteria will be applied on the Microsoft SCM and SAP SCM application by conducting a case study to analyze the problems in executing these cases successfully.

7.1. Synchronous Behavior: Synchronous behavior use case talks about the technology used in the client side for a faster response time so that the user can execute his/her tasks faster.

7.2. Information Drill Down: Information drill down can be referred as getting detailed information on an interested item and further drilling it down to get more details on the selected item for better visibility and decision making.

7.3. Real-time Simulation: Real-time simulation in this use case refers to the real time activities that occur in a manufacturer’s shop floor and how managers are able to understand those activities.

7.4. Logistics Management: Logistics management deals with all the activities involved in understanding logistics related tasks for transporting/receiving the goods to a supplier, retailer or a manufacturer.

7.5. Inventory Management: Inventory management deals with the visibility of the list of items available in stock with each supplier and also within the company’s warehouse.

7.6. Supply Chain Relation: Supply chain relation specifically deals with the data associated with the relation between a supplier, a manufacturer, and a retailer in a product flow.

7.7. Accessibility: Accessibility talks about the Section 508 issues and how those guidelines are followed in this application.
8. Case Study on Manufacturers Supply Chain Application:

This thesis will focus on a case study of an existing supply chain application used by a manufacturer. As mentioned earlier, a manufacturer is one of several other components involved in the supply chain process. A manufacturer’s application is chosen for the case study because it acts as a middleware between a supplier and a retailer. That being said, enormous amounts of information exchange could take place in this area. Through this study, problems involved in using an existing supply chain application will be discussed in detail. A research will be conducted to overcome these issues through visualization and the results will be demonstrated in a web application.
9. Research Study 1: Microsoft Dynamics SCM

Microsoft Dynamics SCM (Supply Chain Management) application is one popular business software used among many of Microsoft's business customers. It offers global manufacturers a business management solution that provides integrated, transactional, real-time information in an open standards format that can be easily shared across the entire supply chain. This is a desktop application which provides access to detailed inventory information from a central location, has easy to use stock analysis tools and flexible reporting capabilities to enable the creation of personalized reporting to meet customer demands. This application helps manufacturers quickly turn data into decision-driving information for responding rapidly to change and fuel customer satisfaction.

The case study for this thesis will emphasize on the technology and the user experience of this supply chain application. Figure shown below is a screenshot of Microsoft Dynamics SCM application. Its user interface has a similar look and feel of Outlook Express mail client which is quite familiar to Microsoft's huge customer base.
9.1. Problem Statement:

The following are some of the major problems that were identified in using this application:

9.1.1. Desktop Application:

One of the first prominent features that will be noticed in Microsoft Dynamics SCM is the platform in which it operates the system. It is a traditional desktop application which runs on a client’s computer locally. It is connected to a centralized Windows Server platform for information exchange based on the client’s request. It uses HTTP protocol to communicate between the server and the client.

A major drawback for developing a SCM application over the desktop could be the maintainability related issues. For an example, if the company decides to make some changes in certain UI features of the client application, it requires installing an update over the internet or locally. This installation needs to be done in the entire client’s system which runs this application. This could be a huge problem in terms of maintainability if the company has massive number of clients.

9.1.2. Synchronous Behavior:

Synchronous behavior is a client server interactivity model as illustrated in the figure below. In this model, the client gets a response from the server side only when the user makes a request. Therefore, when a new message arrives in the server side, it will not be immediately notified to the client unless and until the user makes a request.
Microsoft Dynamics SCM follows this type of synchronous behavior which makes the client unaware of a new alert received in the server side. However, it checks for alerts after every one minute or user defined time interval as it does in Outlook Express mail client. This would still miss the instant notification of an alert to the client. For example, if an item shortage is discovered in the warehouse, it should be immediately alerted in the application for taking the required action. This could be a very crucial step in the inventory management of supply chain process.

9.1.3. **Information Drill-down:**

As shown in the figure below, huge chunks of information related to the supply chain data are represented either in the form of tables or textually.
Information drill down could be one common activity by the business people for getting detailed information on an area they are interested in. Representing such information in the form of tables and flat text makes it difficult to understand the relationship between the data.

9.1.4. **Real-time Simulation:**

In this application, there is no real time simulation involved to simulate the actual shop floor of a manufacturer. However, it does provide data in the form of alerts when an exception occurs in the shop floor. This may not be the efficient way to develop an application for a manufacturer. The operator who is controlling the assembly line would be totally unaware of the process and the exceptions that might occur while assembling a product. Without providing the real-time simulation of a manufacturer’s assembly line, a supply chain application doesn’t get 100% visibility.
9.1.5. Web Services:

In this application, web services acts as a key communicating platform between the client and the server. Information is in multiple layers and connected cross layers. Data related to the supply chain process is stored in multiple servers in multiple locations and they are accessed by the clients all over the company network in the form of web service. However, it doesn’t utilize the full potential of a web service. It doesn’t use any kind of web mash-up for accessing a third party service. When dealing with a global supply chain process, using a third party mapping tool as a web service could be a very efficient way to map the information on top of their application.

9.1.6. Logistics Management:

Logistics Management is part of the supply chain process which plans, implements and controls the efficient, effective forward and reverse flow and storage of goods, services and related information between the point of origin and the point of consumption in order to meet customer’s requirement. It involves the integration of information, transportation, inventory, warehousing, material handling, and packaging. A logistic manager who would use this application will take the responsibility of geographical repositioning of raw materials, work in process, and finished inventories where required at the lowest cost possible.

In order to accomplish these goals, the supply chain application should provide all possible information related to the geographical location of its suppliers and determining the shortest path for transporting the raw materials, products to or from a manufacturer. For example, let us assume a retailer requesting for a particular refrigerator model to a manufacturer and the manufacturer does not have the requested model in stock. So, the manufacturer decides to assemble the model and still he does not have all the required raw
materials. Hence, the logistic manager's task now is to determine the nearest supplier for getting the raw materials within a very short period of time in order to satisfy his retailer's request. But, this application does not provide any such feature related to the above mentioned logistics problem for making the right decision on the right time.

9.1.7. Visibility on Inventory Management:

Inventory is a list of goods and materials or those goods and materials themselves, held available in stock by a retailer, manufacturer or a supplier. Inventory are held in order to manage and hide from the customer the fact that manufacture/supply delay is longer than delivery delay, and also to ease the effect of imperfections in the manufacturing process that lower production efficiencies if production capacity stands idle for lack of materials. An inventory is a detailed list of items in stock.

The above figure is a screenshot of the application when requested to list the availability of an item in stock. This description does not provide an efficient way to compare the inventory with other suppliers. Each description is
opened in a separate window which makes it difficult to analyze and compare the
different data sets and thereby impacting on the decision making.

9.1.8. **Supply Chain Relation:**

Supply chain relation helps in understanding the product flow from the
source of production to the market place and identifying the relation between
each flow. For example, when a sales order ID is typed in the application, the
application should clearly demonstrate the product flow of that particular order
from a retailer to a supplier along with their relation. When the information is
further drilled down, each process should give more details of the flow with
respect to time, date and description of the flow. This functionality is not
implemented in this application in an efficient way yet it gives the details of
individual flow separately.

9.1.9. **Section 508 Issues:**

"Section 508 of the Rehabilitation Act requires that when Federal
agencies develop, procure, maintain, or use electronic and
information technology, they shall ensure that this technology allows:

a. Federal employees with disabilities to have access to and use of
information and data that is comparable to that by Federal
employees who are not individuals with disabilities, unless an
undue burden would be imposed on the agency, and
b. Individuals with disabilities, who are members of the public
seeking information or services from a Federal agency, to have
access to and use of information and data that is comparable to
that provided to the public who are not individuals with
disabilities."

[ http://www.section508.gov ]

Accessibility guidelines are completely not followed in this application
and thereby ignoring the Section 508 Rehabilitation act.
10. Research Study 2: SAP SCM

SAP has pioneered in the enterprise business application development for more than thirty years and today they are the number one software providers for business applications. SAP SCM allows the managers to model the existing supply chain; set goals; and forecast, optimize, and schedule time, materials, and other resources. It enables the enterprise to maximize return on assets and ensure a profitable match of supply and demand.

SAP SCM features a ready-to-use business processes, event management, performance management and other key processes. It is a seamless integration through the technology foundation of SAP NetWeaver. SAP NetWeaver is based on Enterprise Service Oriented Architecture (SOA) that combines enterprise applications with Web Services and open technologies – enabling truly adaptive business.

10.1. Problem Statement:

SAP SCM has a history of success stories from its satisfied customers. However, SAP applications in general are highly critiqued by the business people for its bad user interfaces. In its SCM user interface, it lacks in one key area:

10.1.1. Real-time Event Notification:

Business applications are about event notifications. SAP SCM does the best in its world to alert the events to the managers. However, SAP SCM does not have a dynamic, intuitive reporting engine to understand the complex data. For example, in its manufacturer based supply chain application it produces enormous amount of data and yet, its reporting engine reports the events in old-fashioned tables or in the form of alerts. Notifying the key events is one of the main tasks for the managers to be performed through an SCM application which is not very well handled in this application.
10.1.2. Information Drill Down:

As mentioned earlier, SAP applications in general are highly critiqued for its bad user interfaces. Figure shown below is the user interface of an SAP SCM application.

Just from the view of this user interface, it looks like an excel spreadsheet. Almost every known information is represented in the form of tables and making the information drill down task even more difficult. Yet another important critique about this UI is the representation of tables with both horizontal and vertical scroll bars. Almost every table in the application has a horizontal scroll bar which has a big impact on the user experience.

10.1.3. Real-time Simulation:

Similar to the Microsoft SCM application, this application doesn’t have any sophisticated feature for simulating the real time activities going on in a manufacturer’s shop floor. It doesn’t have any kind of simulating engine to simulate the events and processes that occur in the shop floor.
11. Research Implementation:

This research implementation aims at solving the problems discussed above by focusing on developing a supply chain application emphasizing on client side interactivity and information visualization. The alternative hypothesis will be considered for developing the business application based on information visualization for better understanding the complex information and decision making.

11.1. Introduction:

11.1.1. Technical problems addressed:

Technology is the master piece of this research implementation. A variety of different web 2.0 technologies are implemented for developing this prototype. This application will be a show case of all the major components of web 2.0 technologies. One primary reason to focus on this buzzword is because there is no supply chain application that exists in the current business world which fully utilizes the power of a Rich Internet Application (RIA).

As learnt from Microsoft Dynamics SCM, it centered all activity around a client-server architecture with a thin client. Through this system all processing is done on the server, and the client is used only to display the static information. The biggest drawback with this system is that all interaction with the application must pass through the server, which requires data to be sent to the server, the server to respond and the result to be displayed on the client side. In order to solve this problem and get a real-time client side interactivity, a client side technology (AS2) is used in the simulation engine for sending/receiving streaming data through client/server sockets. RIA based supply chain application can circumvent this slow and synchronous loop for many user interactions.

The technical problems that are addressed through this research implementation are as follows:
11.1.1.1. **Richer**: This richer functionality of the application may include anything that can be implemented in the technology used on the client side, including drag and drop, using a slider to change data, calculations performed only by the client and which do not need to be sent back to the server, for example, validating the fields of a new sales order.

11.1.1.2. **More Responsive**: The interface of this application is typically much more responsive than those of a standard desktop application which must always interact with a remote server. The most sophisticated examples of RIA’s exhibit a look and feel approaching that of a desktop environment. Using a client engine can also produce other performance benefits.

11.1.1.3. **Client/Server Balance**: The demand for client and server computing resources is better balanced, so that the web server need not be the workhorse that it is with a traditional web application. This frees server resources, allowing the same server hardware to handle more client sessions concurrently.

11.1.1.4. **Asynchronous Communication**: The client engine can interact with the server asynchronously. That is, without waiting for the user to perform an interface action like clicking on a button or link. This option allows the application to move data between the client and the server without making the user wait.

11.1.1.5. **Network Efficiency**: The network traffic may also be significantly reduced because an application specific client engine can be more intelligent than a standard web browser when deciding what data needs to be exchanged with servers. This can speed up individual requests or responses because less data is being transferred for each interaction, and overall network load is reduced. However, use of asynchronous perfecting techniques can neutralize or even reverse this potential benefit. Because
the code cannot anticipate exactly what every user will do next, it is common for such techniques to download extra data, not all of which is actually needed to many or all clients.

11.1.2. Implementation Innovations:

Innovations for this implementation focus on developing a business application for monitoring the real-time activities on a manufacturer’s shop floor. There is no scalable supply chain application that exists in the current business world to monitor and simulate the exceptions that occur in a manufacturer’s shop floor.

SAP AG is the world’s largest business software company which pioneers in Supply Chain Management applications. SAP’s SCM application monitors all the key events that occur in the shop floor but its reporting engine is overwhelmed with data. Their reporting engine simply alerts out the exceptions in a textual form. This snippet of information is not intuitive to use and it is not an efficient way to understand the exceptions occurred. Hence, this research implementation aims at solving the above mentioned problem by attempting to simulate the events with the overwhelmed data. This Thesis work is done in collaboration with SAP AG to address this problem by experimenting with the data from their Supply Chain Application. This real-time manufactory monitoring is the major innovation that has been implemented for this Thesis.

11.1.3. Related Work:

There were some attempts made by other organizations/institutions to solve the similar supply chain problem through visualization.

11.1.3.1. MIT Tangible Media Group:

MIT’s Tangible Media group has developed a physical sensible table which helps the managers to physically construct and interact with
models on how products flow between their business, suppliers and customers. Managers can physically modify this model on the table to send information and receive real-time feedback on the simulation as how the information sent has changed the actual process.

[http://tangible.media.mit.edu/projects/scvis/]

This is yet another innovative approach by MIT to solve the similar problem proposed in this Thesis. However, the idea is still in the research phase and it needs to address some major technical challenges to put up a functional prototype from the actual supply chain data.

11.1.3.2. PMC:

PM Corporation provides simulation services and software’s for proposed enterprise systems to improve productivity. They maintain a wealthy portfolio of 3000 successful simulation projects. However, this type of service is more of a static animation to demonstrate how a system would function in the real world and they are not data centric. They help managers to understand & analyze the supply chain cost & performance.

[ http://www.pmcorp.com/Industries/discrete_event_simulation.asp ]
11.2. Scenario: Real-time Manufactory Monitoring

This supply chain application that was researched and developed for this Thesis consists of three different intuitive layers in the presentation tier. A global supply chain mapping layer to keep track of the global events and the product flow. A manufacturer shop floor to simulate the real time events that occur in a shop floor. A detailed level assembly line layer to simulate the assembly process of a product.

11.2.1. Global Supply Chain Mapping Layer:

The global supply chain layer is built on top of the Yahoo! Maps API and it is implemented with the following functionalities in order to overcome the issues learned from the case study.
11.2.1.1. Global Locator:

A global locater helps in identifying the physical locality of all the suppliers, manufacturers and retailers associated with a company's supply chain process. This feature helps the managers to physically identify where their nearest supplier is located in the global map. By utilizing the web service provided by Yahoo! Maps they could also figure out the exact distance between each supplier to further optimize their transaction time. In this way it helps the managers to make better decisions by visualizing these supply chain data.

As shown in the figure below, all the markers plotted with the color orange are the real-time location of the manufacturers and blue ones are the suppliers and the grey markers are the retailers located in different parts of the globe. Further location related information can be drilled down by clicking on the markers itself. These markers are plotted by feeding in their address information into the application. If the address is not available, markers can also be plotted by knowing the latitude and longitudinal co-ordinates of the location.
11.2.1.2. Parts Tracker:

Tracking the availability of parts with all the suppliers is a key functionality in the inventory management. As learnt from Microsoft SCM, inventory management is represented in a tabular form and each supplier data are represented in a different window. In order to address this issue, data associated with the inventory management are directly displayed on the map overlay of this application. The user doesn’t have to worry about comparing different tables and make a decision. Through this overlay, inventory management can be compared and analyzed from different suppliers. Information can be further drilled down by choosing the interested supplier.

In the figure below, numbers at the left 101, 102, 103, etc., are the ID’s associated with the parts and the numbers at the right are the available numbers in stock of that individual part. It is represented with an interactive bar graph which gives a visual overview of the available item. All these information are displayed in the real-time location of the suppliers.
11.2.1.3. Shipment Tracker:

The shipment tracker feature helps in analyzing all the transactions involved during the transportation of items from one place to another. This feature gives all the information regarding when an item was shipped, where the item is shipped to, time stamp of the origin and the destination. If there were to be a problem during the transit, application will automatically alert the manager through its asynchronous behavior.

In the figure below, it has been assumed that an item is shipped from Stanford University, Palo Alto to SAP Labs, Palo Alto. The details of the origin and the destination can be viewed by clicking on the respective markers. At the intersection of Page Mill and Junipero Blvd, an exception has occurred during the transit which immediately alerts the logistic manager with a red marker. When the red marker is clicked, complete details regarding the exception are displayed to the manager including the description of the exception.
11.2.1.4. **Interactive 3D Bar Graph:**

The 3D bar graph component has the functionality of any other bar graph in exception to its three dimensional representation and drag and drop utility. This component gives a visual overview of the parts available with all the suppliers within the selected countries.

As shown in the figure below, when the user types the ID of an interested part, it comes up with a 3D bar graph which shows the percentage of parts available with all the suppliers in USA, Germany and China. This feature helps in comparing the parts availability of suppliers from different countries. An individual bar can be dragged and dropped in any area of the map overlay. In this way it gives a greater visibility on the physical locality of each country and the associated graph overlaid on top of it. The color purple in the graph refers to the percentage of items which were already purchased or floating in the supply chain process and the color green represent the percentage of items available in the inventory.
11.2.1.5. Interactive 3D Pie Chart:

The interactive 3D pie chart helps in analyzing the availability of an item from a supplier, manufacturer, distribution center, retailer and purchased. This data visualization is tied with a database and thereby making the pie chart dynamic for all the available parts in the inventory.

As shown in the figure, an item with an ID 102 is searched in the database. As a result, it displays the percentage and number of items available from five different sources. Each individual slice of the pie chart can be dragged to a separate segment (orange) as shown in the figure. The thickness of each slice is directly dependent on the availability of items. As shown in the visualization, the supplier (red) has the maximum number of selected item and hence its thickness is greater than the rest of the slices. These slices can also be rotated in 360 degree to see the variation in the thickness of each individual. In this way this interactive 3D pie chart is more intuitive to use and to understand the data as opposed to Microsoft’s SCM application.
11.2.1.6. Visualizing Business Objects:

Business process visualization is an interactive bubble visualization overlaid on top of the global map as shown in the figure and also as a separate component. This visualization gives an overview of entire process that occurred in the supply chain process.

As shown in the figure, red bubble represents all the process occurred within a retailer and then the process moves on to the orange bubble for manufacturers and finally to the purple bubble for suppliers. Surrounding smaller bubbles are called the business objects which gives a snapshot of individual process. When these business objects are dragged and dropped within the circular area of the UI at the left, users will get a complete description of the selected object. The description will include the name of the object, date and time of the transaction and any comments available in the database. In the example shown below, “Sales Order Generation” object of retailer is dropped in to the UI for further information drill down. These details can be obtained by dropping any node of the visualization into the circular area.
Business process visualization can also function as a separate component as opposed to overlaying the data on a map. The only advantage of using the map overlay is to get the real-time location of the suppliers, manufacturers and retailers. If the user is concerned only about getting the visibility on business process, this component shown below can be used.

11.2.1.7. Visualizing Supply Chain Relation:

Supply chain relation visualization helps in identifying the relation between each unique product flow process. This visualization helps in keeping track of how a company is related to its supplier or a retailer. In the figure shown below, it visualizes two supply chain relations: one represented in orange and another in white. From the orange lines, a manufacturer in China named "Meina Li" is the principal manufacturer associated with all the retailers in USA which are pointed
with the grey markers. In the same way from the white lines, “Fannie Mae” is a manufacture in Rochester, NY associated with two other retailers in London and Madras, India.

11.2.2. Manufacturer Shop Floor:

Manufacturer shop floor is the second layer of the application which simulates the movement of all physical objects of an industrial landscape. User will be able to navigate to this layer of the application by clicking the marker of any manufacturer (orange) displayed in the global layer. In this way a business manager can keep track of all the events and exceptions that occur during the assembly process. For this research implementation, simulation of one manufacturer is particularly considered.

This layer is implemented with the following functionalities in order to overcome the issues learned from the case study.
11.2.2.1. Real-time simulation:

This functionality will help in simulating the movement of all physical objects on the assembly line of a manufacturer. The key objective of this functionality is to give a greater visibility on the physical objects and alert all possible exceptions to the manager for decision making.

Figure shown below is the top view of a manufacturer's shop floor and for this research implementation, a refrigerator manufacturing company is considered. The assembly process of the refrigerator starts from the assembly line A1 and the cabinet moves to different assembly lines (A2, A5, A8) based on the model of the refrigerator that needs to be assembled. At each stage, different parts of the refrigerator are put together. This movement is completely tracked and simulated in the application for a real time visibility on physical objects.

The data for this prototype are obtained from a manually fed XML file. In order to make this application for a real world implementation, data in the future will be generated from an RFID reader by scanning the
RFID tags affixed with each individual item on the assembly line. In this way, every single item in the assembly line can be tracked based on timestamp and their current locality. The current locality of an item can be determined by the ID’s of each RFID reader points located at the junction of every assembly line. Data from the reader will be sent to the application asynchronously and thereby delivering a real-time simulation.

This real-time simulation of an assembly process is entirely unique and considered as a major innovation for this research and there is no supply chain application that exist in the current business world to attempt this type of simulation.

11.2.2.2. Instant Alert:

Instant alert functionality helps in notifying the managers if any exception were to happen in the assembly line. For example, in a refrigerator assembling process when a wrong part is attached, it is considered as an exception. In order to prevent that item to continue further with the assembly process, the operator should be immediately notified about the exception that has occurred. Through this visualization, alerts are instantly displayed on the simulation. Once the operator receives the notification, required action is taken and thereby handling the exception.

These exceptions can be triggered by the RFID reader by comparing the EPC code in the reader and in the tag. When there is a mismatch, exception is sent to the application. This functionality helps in understanding the exception in a much more realistic way as opposed to getting alerts through textual information.
11.2.2.3. Customizable Assembly Routes:

Customizable assembly routes are one of the unique features of the simulation. During the assembly process of a refrigerator, items don’t always follow a single path to complete the process. They travel in different paths based on the model and the requirements of the refrigerator. In order to simulate this real world process, the assembly route should be configurable in the application. This application has the ability to accomplish this goal by providing the required path through an external XML file. As a result, the object in the simulation follows the specified path and can be changed at any time.

As show in the figure, this research implementation consists of three configurable paths path 1(A2), path 2(A3, A6, A2) and path 3(A5, A8, A2). Path 1 always follows the straight assembly line A2 for the assembly process and path 2 changes its track from A3 to A6, A6 to A2 and A2 to the storage area. These paths are fed through the external XML file or the database to accomplish the simulation.
11.2.2.4. RFID Reader Points:

RFID reader points are affixed at the junction of each assembly line. These readers scan the RFID tags affixed with every physical object floating in the assembly line to make sure that the items are moving in the right track. If the item is in the wrong assembly line, it should be immediately notified to the operator. In order to perform this task successfully, RFID readers should always be active. If there is a problem with the reader, it should be immediately notified in the application.

This application handles this task in a very sophisticated manner through its real-time simulation by displaying an interactive alert box near each RFID reader. The status will display as success or failure for each reader. In the figure shown below, alert boxes are displayed with a success status message for each reader which confirms the operator that everything goes fine with the assembly process. In this way, the operator will be able to get a real-time visibility of a manufacturer’s landscape.
11.2.3. **Assembly Line:**

Assembly line layer is the third and the inner most layer of the application which gives a detailed level monitoring of an individual assembly line. The user will be able to view this layer when he/she drills down the map from the manufacturer’s shop floor layer. This layer gives detail level information about the parts that are being assembled, event management, instant alert, and a snapshot of inventory management. Assembly line layer is implemented with the following functionalities in order to overcome the issues learned from the case study.

11.2.3.1. **Detailed Assembly Process:**

This functionality gives a greater visibility on each and every individual stage of the assembly process. For this research implementation, 8 different stages are considered for assembling a refrigerator. In the figure shown below, the assembly process starts with stage one which is "Installing the cabinets" and then moves on to stage 2 and vice versa. User will be able to keep track of all the exceptions and events that occur in each one of these stages to ensure that the right parts are being assembled.

This layer also comes with a snapshot of items available in the inventory for assembling a refrigerator. In the figure shown below, at the bottom of the application a simple UI which gives a quick preview of list of all the items and its availability in the storage area and smart shelves. At the right, it also has the retailer sales order information for manually comparing the items that are being assembled based on the retailers requirement.
11.2.3.2. Event Management:

Event management functionality helps in visualizing all possible exceptions that might occur during the assembly process. In the figure shown below, an exception has occurred which has been notified in the application with a tortoise icon. This alert implies that this particular item is moving slowly over the assembly line and it requires the operator's...
attention. These alerts are instantly notified through the asynchronous behavior of the application for better decision making.

11.2.3.3. Visibility on Assembled Parts:

The assembly process is all about putting the right parts or the raw materials together to come up with a unique product. Hence, the visibility on assembled parts is the key for successfully executing the process. From the previous functionality it was clear how this application was capable of instantly alerting the exception and notifying it to the operator. After the alert notification, next task is to figure out what could have caused the problem. Visibility on parts functionality helps in identifying the problem that has caused the exception.

In the figure show below, an exception is marked in red as “Problem: parts mismatch” which notifies the operator that a wrong part is assembled on a product. In order to identify the cause of this problem, user then clicks on the alert. This comes with a complete list of parts that
are used to assemble that particular product which has caused the exception. In this way, the user will be able to compare this list of parts with the retailer’s sales order information to ensure that they have the same set of parts. If not, that could have caused the parts mismatch exception.

11.2.3.4. AJAX – Notification Inbox:

AJAX notification Inbox is the only module of this application developed in AJAX. This widget acts as a quick messaging service which notifies some key events to the user. In the figure shown below, a message has been sent to the user with an alert type of ‘severe’. This notifies that an item is out of stock in the inventory and that it requires an immediate attention by the user. User will be able to get further details
on that alert with respect to the date and time by opening the message.

Transaction suggest feature is quite similar to that of the Yahoo! search assist for suggesting the transactions. When the users wants to view the details of a particular sales order, everything begins with a transaction ID. When the user types the first number of a transaction ID, this AJAX widget pulls the list of all transactions from the database that starts with the typed number and suggests it to the user. This search can also be performed by typing the first few characters of a retailer. This functionality helps the user to complete the task in a much shorter time and thereby resulting in a better user experience.

11.4. Architecture: Web-based Information System for Manufacturer

11.4.1. Introduction:

This Supply Chain Management application that was researched and developed is a browser dependent application which includes the major components of web 2.0 technologies. In order to reinforce the visibility of global supply chain process, most of the applications functionality was built on top of an existing mapping application. For this research implementation, Yahoo Maps API was used as a web service for mapping data in the real-time locations through geo co-ordinates. Since the application deals with real-time information, asynchronous update of data is reinforced and thereby making it a rich client application. AJAX and Flash were used together to accomplish this and the results were demonstrated in the web application. Entire visualization part of the application was developed in vector...
graphics. Aesthetics has played a very big role in the visualization for making it intuitive to use and better understanding the complex information.

11.4.2. Technologies:

11.4.2.1. Client Side Technologies:

11.4.2.1.1. Flash Authoring Tool:

This prototype has most of its visualization functionalities built with Adobe’s Flash Professional 8. Flash was primarily chosen for this research implementation because of its capability to generate vector graphics and it is widely accepted in almost every browser client. From the Adobe’s statistics below, it proves that 98.8% of global internet users have Flash plug-in installed in their browser client and thereby making a Flash application universally accessible.

[ http://www.adobe.com/products/player_census/flashplayer/ ]

The second major reason for choosing Flash was about its tremendous capability to interact with server side technologies
through an XML data interchanging format. Due to this, Flash can communicate with just about any server side technology by returning an XML output. The same principle applies for getting/setting information from a database.

W3C’s SVG was also considered initially due to its capability to generate vector graphics and also it is getting very popular among the developers. However, SVG is not fully supported even by the modern browsers. As of now from the above statistics, SVG is supported only by 9.9% of the global internet audience.

11.4.2.1.2. Object-Oriented ActionScript:

ActionScript is the primary programming language for the Flash authoring tool. Before Flash version 5, ActionScript was more of a scripting language and later that version, it came out in Object Oriented programming language which resembled a lot like Java. Due to its resemblance of Java, Flash was widely accepted among many developers. For this research implementation, all the animation is dynamically controlled through ActionScript. Due to this applications scalability, none of the animation is based on timeline and everything is controlled through the code. Generic ActionScript custom classes are written based on good Object Oriented coding principles.

11.4.2.1.3. AJAX/JavaScript:

AJAX/JavaScript is used in this application for implementing all the features that are textually represented. Notification inbox of this application is a good example of that which is completely based on AJAX. This technology is approached in order to make all the textual information accessible.
11.4.2.1.4. **XHTML:**

XHTML is used in this application with all possible semantic mark up. More importance is given on the semantic markup in order to deal with the Section 508 issue which was completely ignored in the Microsoft’s SCM application.

11.4.2.1.5. **XML:**

XML acts as a primary data interchanging format for both Flash and AJAX. XML was chosen because it is very well supported by both client side technologies as well as server side technologies and thereby acting as a valid middle tier.

11.4.2.1.6. **CSS:**

CSS is used only in the assembly line layer of the application where AJAX is used. The basic idea of CSS was to support the Section 508 issue and make the application accessible to everyone. CSS helps in achieving this goal by clearly separating the presentation of the content from the structure of the content.

11.4.2.2. **Server Side Technologies - JSP & Java Servlets:**

Java Servlets is used as a primary server side technology in order to reinforce the business logic and database interaction involved in the supply chain process. Since, this project was done in correlation with SAP Labs, it was an industry standard recommendation to develop the entire logic tier of the application in Java. Java is also preferred due to its Object Oriented principles for good code maintainability and scalability.

JSP is mostly used in the presentation tier of the application for combining the server side variables and HTML together. JSP also plays a major role in generating the output from the server side logic.
and the database in an XML format for communicating with the Flash environment.

11.4.2.3. Database - SQL Server 2005:

SQL Server database is used in this application as part of the industry standard recommendation. SQL Server is also widely popular among the enterprise level business application development.

11.4.2.4. Application Server - Sun Java Application Server:

Sun Java Application Server acts as a web server for hosting the entire application in a server environment. There were no specific reasons for choosing this web server as it comes pre-installed with the Sun Java Studio Creator IDE.

11.4.2.5. IDE - Sun Java Studio Creator:

Sun Java Studio Creator IDE was preferred as it is a proprietary IDE of Sun Microsystems itself and thereby getting the in-house effect of Java programming language.

11.4.2.6. Web Services - Yahoo! Maps API:

Yahoo! Maps API plays a major role in the global layer of the application. Pretty much every functionality in the global layer is built on top of Yahoo! Maps. Yahoo! Maps is preferred over Google Maps due to its API preferences. Yahoo! has its API opened up for both Flash and AJAX/JavaScript as opposed to Google which can be utilized only through AJAX/JavaScript. Since this research implementation is emphasized primarily on visualization and vector graphics, Flash API is preferred and thereby choosing Yahoo! Maps. The user interface developed for this application is customized to blend with the look and feel of Yahoo’s mapping interface as everything was part of the Flash family.
11.4.3. Dynamic Visualization Framework:

This three layered supply chain application was built by creating a generic visualization framework. Since this prototype emphasize particularly on the presentation tier, it was even more important to develop the application based on a framework for connecting it with a back end system. For this research implementation, this application is not tied with SAP NextWeaver as illustrated in the framework shown below. NextWeaver is SAP’s integrated technology platform and is the technical foundation for all SAP applications. As per the industry requirement, the ultimate goal of this research implementation is to tie this application with NextWeaver for getting the real time supply chain data.

In order to accomplish this goal, the application is coded by developing a generic visualization framework so that it might be easier to connect with NextWeaver. Custom ActionScript classes were created by clearly separating the classes based on the functionality of individual components.
11.4.4. Model-View-Controller (MVC) Design Pattern:

Model-View-Controller is an architectural design pattern followed in Software Engineering. In a complex application like this that deals with enormous amount of data exchange to the user, the developer often wishes to separate data and user interface concerns, so that changes to the user interface will not affect data handling, and that the data can be reorganized without changing the user interface. The Model-View-Controller solves this problem by decoupling data access and business logic from data presentation and user interaction, by introducing an intermediate component called the controller.

For this research implementation, Model-View-Controller design pattern is implemented and perfectly utilized by clearly separating the classes associated with model from the view or the user interface. Most part of the model are developed with Java Servlets for data access and business logic. View is utilized through JSP, XHTML, ActionScript, AJAX and XML. Controller and the classes associated with event handling are partially done in ActionScript and the rest in JSP.
12. Discussions:

The most important factor for developing any application is to do the user study and understand their goals and objectives to use that application. It is very important to understand why a user should be using that particular application. Once the goals and objectives are identified, the application should be developed in a way that users are actually able to accomplish those tasks in an effective way.

In this thesis, a use case is considered initially for determining the goals and objectives of a user to use a SCM application. Based on those objectives, a case study is conducted on Microsoft SCM and SAP SCM. Problems were identified in using those textual applications. The primary goal of this research implementation was to solve those problems identified through information visualization.

Through this research implementation, it was quite clear that visualizing business data has a significant amount of improvement in the visibility of information. It has also been proved that utilizing the real power of web 2.0 technologies has a greater impact in completing the users tasks in a very efficient way. Simulating the manufacturer’s shop floor is considered as one of the major

![Diagram of Supply Chain Network](3 Layered Supply Chain Network)
innovation for this research implementation. As illustrated in the above figure, the primary goal of the simulation is to bridge the gap between the physical layer and the alerting layer of supply chain process. Physical layer is the layer, which has all the physical objects that are moving around the manufacturer’s shop floor, and the alerting layer is where we have the three-layered supply chain application for alerting the user. The middle tier, information layer has all the streaming data that are being generated by the physical layer. The problem that has been addressed through this simulation was to connect the gap between these layers and get a thorough visibility on the supply chain process.

Web 2.0 technologies however has its own advantages and disadvantages when compared to the traditional web 1.0 technologies. Through this application, advantages have been utilized to its full potential. However, one of the major disadvantages that could be considered with these rich Internet applications is about the amount of server side load that could be fed. Considering a single client making asynchronous call to the server after every second could not be much of a problem. However, if hundreds and thousands of clients are accessing the same application at the same time with an asynchronous call from each one of those clients could have a huge amount of server load. This will have a significant impact on the response time and thereby requiring a distributed computing approach to share the process load.

Secondly, usage of Flash in this application has a great advantage over generating vector graphics and thereby successfully executing the simulation process. However, Flash has its own downside when it comes to the performance related issues. All the Flash based applications renders the output in the SWF (Shockwave Flash) format which runs within the browser environment as an ActiveX Control (IE) or as an embedded file (Gecko). This acts like a virtual environment running in parallel with the browser rendering engine. As a result, SWF has some
performance related issues by slowing down its processing speed and this applies only when it deals with enormous amount of data exchange. This was one of the major reasons for the recent switch over of Yahoo! Maps from Flash to AJAX. For this prototype, this issue wasn’t much of a problem as the data were very limited and manually fed. However, when this application deals with real-time supply chain data, there might be some performance related issues.

Other than these minor technological challenges, this research implementation has clearly demonstrated that developing a business application based on information visualization has an impact on understanding the complex information.
13. Future Roadmap:

The biggest challenge that is yet to be considered in this functional prototype is to connect this application with SAP’s NextWeaver. As mentioned earlier, NextWeaver is SAP’s integrated technology platform and is the technical foundation for all SAP applications which includes SAP’s SCM application. This research implementation was more of a practical demonstration of the ideas expressed in this thesis. The data used in this research are manually fed into the application to demonstrate the results. As a future road map, this application should be tied with the NextWeaver to experiment with the results when dealing with real supply chain data.
14. Conclusion:

Based on the case study of two major supply chain applications, I have learnt the problems involved in using those applications for understanding the complex information. Those applications had the capability to utilize the full power of client-server computing as opposed to the usage of modern cloud computing. Problems involved in understanding the complex information through those text based applications were also determined. A supply chain application is developed based on information visualization and web 2.0 technologies to solve the problems identified through the case study. From the results demonstrated above, it was quite clear that developing a business application based on information visualization with the combined power of web 2.0 technologies has a significant impact on understanding the complex information.

Alternative hypothesis says that:

“Developing a Business Application based on Information Visualization has an effect on understanding the complex information and decision making.”

Hence, through this thesis alternative hypothesis has been proved and accepted.
15. Selected Code from MVC:

This section has a selected class from each of the different components of Model-View-Controller design pattern to demonstrate the code standards followed in this project.

15.1. Model: (BizProcess.java)

```java
/*
 * BizProcess.java
 * @author Ramesh Gopalakrishnan
 */
package BizProcess;

import java.io.*;
import javax.servlet.ServletException;
import javax.servlet.http.*;
import java.sql.*;
import java.lang.*;
public class BizProcess
extends HttpServlet {
  public void doGet(HttpServletRequest request,
HttpServletResponse response)
throws ServletException,
IOException {
  response.setContentType("text/xml");
  PrintWriter out = response.getWriter();
  try {
    //JDBC-ODBC Driver- Opening a Database Connection
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
  } catch (ClassNotFoundException clsex) {
    out.println(clsex);
  }
  try {
    Connection con = DriverManager.getConnection("jdbc:odbc:biz", "sa", "Tendulkar1");
    String typedId = request.getParameter("typedId");
    String requester = request.getParameter("requester");
    String result = "No Results Found !";
    String sql = "SELECT * FROM [scm].[dbo].["] + requester + "];
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery(sql);
    out.println("<businessProcess>");
    while( rs.next() ) {
      String orderId = rs.getString("orderId");
      if(orderId.equals(typedId)) {
        String process1 = rs.getString("process1");
        String process2 = rs.getString("process2");
        String process3 = rs.getString("process3");
        String process4 = rs.getString("process4");
        String name = rs.getString("name");
        String address = rs.getString("address");
        result = "Results Found !";
        out.println("<process1>" + process1 + "</process1>" );
        out.println("<process2>" + process2 + "</process2>" );
        out.println("<process3>" + process3 + "</process3>" );
        out.println("<process4>" + process4 + "</process4>" );
        out.println("<processes>" );
        out.println("<address>" + address + "</address>" );
```

Ramesh Gopalakrishnan
15.2. View: (AssemblyLine.as)

//Assembly Line
import mx.utils.Delegate;

class AssemblyLine extends MovieClip {
    private var whichLoc:MovieClip;
    private var simData:XML;
    private var timer:Number;

    public function AssemblyLine( mcLoc:MovieClip ) {
        this.whichLoc = mcLoc;
        this.simData = new XML();
        this.simData.ignoreWhite = true;
        this.simData.onLoad = Delegate.create( this, onSimDataLoad );
        this.simData.load( "../../XML/Landscape/assembly.xml" );
    }

    private function onSimDataLoad( isLoaded:Boolean ) : Void {
        if( isLoaded ) {
            this.simulate();
        } else {
            trace("Error loading XML file");
        }
    }

    private function simulate() : Void {
        var locate:Object = new Object();
        var pathsNode:XMLNode = this.simData.firstChild.childNodes[0];
        var cabinetsNode:XMLNode = this.simData.firstChild.childNodes[1];
        var pathId:String;
        var cabinetId:String;

        for ( var i:Number = 0; i<this.getTotalCabinets(); i++ ) {
            for( var j:Number = 0; j<this.getTotalPaths(); j++ ) {
                if( this.getPath( cabinetsNode.childNodes[i] ) ==
                    pathsNode.childNodes[j].firstChild.firstChild.nodeValue ) {
                    pathId =
                    pathsNode.childNodes[j].firstChild.firstChild.nodeValue;
                    cabinetId =
                    cabinetsNode.childNodes[i].firstChild.firstChild.nodeValue;
                    locate._x = Number( pathsNode.childNodes[j].childNodes[1].firstChild.firstChild.nodeValue );
                    locate._y = Number( pathsNode.childNodes[j].childNodes[1].lastChild.firstChild.nodeValue );
                    if ( !eval("this.whichLoc.cabinet"+cabinetId+"_mc" ) ) {
                        } } }
            } }
        } }
this.whichLoc.attachMovie( "path" + pathId, "cabinet" + cabinetId + ":mc", Number(cabinetId), locate );

private function getPath( cabinetNode:XMLNode ) : Number {
    return Number( cabinetNode.childNodes[1].firstChild.nodeValue );
}

private function getTotalCabinets() : Number {
    return this.simData.firstChild.childNodes[1].childNodes.length;
}

private function getTotalPaths() : Number {
    return this.simData.firstChild.childNodes[0].childNodes.length;
}

15.3. Controller: (Cabinet.as)

//Cabinets
class Cabinet extends MovieClip {

    public var speed:Number;
    public var sapBoxLevel:Number;
    public var run:Boolean;

    public function Cabinet() {
        sapBoxLevel = Math.round(Math.random()*10000);
        speed = 2; //Speed of the Cabinet in Pixels (2 px)
        _y = 66;
        run = true;
    }

    private function moveMe() {
        _x += speed;
    }

    private function stopMe() {
        _x = _x;
    }

    private function checkWrap() {
        if ( _x >= Stage.width * 3.2 ) {
            this.createSAPbox();
            this.remove();
        }
    }

    private function checkOnAssembly() {
        //Put the Cabinets back in the assembly line
        if ( _y <= 80 & _y >= 50 & run==false ) {
            _y = 66;
            this.moveToMe();
        }
    }

    private function closeDetails() {
        if (_x == 620 || _x == 1340 || _x == 2300) {
            this.gotoAndPlay( "close" );
        }
    }

Ramesh Gopalakrishnan 58 Winter 2007
private function createSAPbox() {
    _level0.assembly_mc.cabinetTunnel_mc.attachMovie( "SapBox",
    "sapBox"+sapBoxLevel, sapBoxLevel );
    eval("_level0.assembly_mc.cabinetTunnel_mc.sapBox" + sapBoxLevel
    ).cacheAsBitmap = true;
}

private function onRelease() {
    run = false;
    this.stopDrag();
}

private function onReleaseOutside() {
    run = false;
    this.stopDrag();
}

private function remover() {
    this.removeMovieClip();
}

private function onEnterFrame() {
    if ( run == true ) {
        this.moveMe();
    } else {
        this.stopMe();
    }
    this.checkOnAssembly();
    this.checkWrap();
    this.closeDetails();
}
16. Supplemental Storage Media:

The end product of this research is this Thesis document itself and a deliverable storage media (CD) which includes the complete source code (250 MB) of the web application.
17. References:

17.3. Norman, Donald. The Design of Everyday Things
17.4. Bogaard, Daniel S., Vullo, Ronald P. and Casciolio, Christopher D. SVG for Education Simulations
17.5. Gershon, Nauhman, Eick, Stephen G. and Card, Stuart. Design
17.6. Harris, Robert L. Information Graphics 1996
17.7. Tegarden, DavidP. Business Information Visualization
17.8. Wong, Chien Y., Agarwal, Vivek and McFarlene, Duncan. The Intelligent Product Driven Supply Chain
17.9. Keim, Daniel A. Information Visualization and Visual Data Mining
17.10. Lerner, Reuven. At the Forge: Creating Mashups
17.11. North, Stephen. Topological Fisheye Views for Visualizing Large Graphs
17.13. Joshi, Yogesh V. Information Visibility and its Effect in Supply Chain Dynamics
17.15. Yoshicmichi, Tanida, Yasuki, Nakayama. Atlas of Visualization III
17.18. Williams, John R. An RFID Network Simulator.
17.20. Lo, Joyce. Securing Pharmaceutical Supply Chain.


17.30. Heer, Jeffrey and Agrawala, Maneesh. Software Design Patterns for Information Visualization.

17.31. Heer, Jeffrey. Socializing Visualization.


17.33. Agrawala, Maneesh, Phan, Doantam, Heiser, Julie, Haymaker, John, Klinger, Jeff, Hanrahan, Pat and Tversky, Barbara. Designing Effective Step-By-Step Assembly Instructions.