On Demand Data Retrieval Web Service based Framework for Manufacturing Supply Chain Integration and Management Support

J. B. Carvalho
Department of Production and Systems
University of Minho
Braga, Portugal
joelbcarvalho@gmail.com

M. L. R. Varela
Department of Production and Systems
University of Minho
Guimarães, Portugal
leonilde@dps.uminho.pt

***Abstract*— Manufacturing management problems solving can benefit a lot from the use of web service technology. Moreover, it turns out to be possible to gain in agility and in flexibility for building a supply chain management system, which deals with complex and sequential flows of information. This is due to the central idea of web services technology for ensuring connectivity and interoperability of software components over the web.**

Having real-time information available at any time can increase information transparency, which in turn, makes it easier for manufacturing managers to identify potential risks. Therefore, in this paper we propose a framework for updating in real-time and on demand an enterprise data warehouse system based on a typical scheduling ETL (extraction, Transformation and Loading) tool. This on demand ETL tool, will allow managers to access updated valuable data through the use of a web service which will handle the update of each specific table of data for better supporting manufacturing decision making.

Keywords- Web Service; ETL; Data Warehouse; Decision Support System; RESTFull; ERP; Supply Chain

I. INTRODUCTION

Restrictions between web and desktop interaction are clouding, in one hand, the traditional PC desktop is now populated by widgets such as the Mac Dashboard, web fast-download apps such as Java Web Start or Adobe Air, and expanded browser functionality such as Chrome. On the other hand, computation and applications that were once part of the desktop are now being relocated to the web (example: word-processing with Google Docs), and various technologies enable web applications to function even when users have no connectivity to the internet (example: Google Gears and the offline mode of HTML5). The web becomes an extensive manufacturing resource source, and increases the variety and scale of manufacturing resources available. In modern years enterprise-level websites are having more attention by manufacturing companies. These enterprise websites and portals contain ample manufacturing resources and application service information, such as design knowledge, application services, manufacturing information, enterprise data and so on. Web-based enterprise databases and application platforms supply all kinds of manufacturing resources as well.

Through the 1990s numerous companies participated in large specialized information systems that sustained complex business functionalities, data analysis and reporting with sophisticated user interfaces. Businesses financed greatly on enterprise resource planning (ERP), supply chain management (SCM), and other organizational structures. Through the 1990s numerous companies participated in large specialized information systems that sustained complex business functionalities, data analysis and reporting with sophisticated user interfaces. Businesses financed greatly on enterprise resource planning (ERP), supply chain management (SCM), and other organizational structures. Information and knowledge distribution over customer supplier value chains and partners occurred through electronic data interchange (EDI) systems [1]. While these systems pleased significant and crucial necessities at the time of their introduction, they slowly grew to be information silos with little data sharing across systems. Even the EDI systems were specific to the interface and did not bring the flexibility needed by changing business processes. Despite the numerous shortcuts and service programs, businesses found these select information systems rapidly becoming out-of-date and preventive. To authorize information distribution through organizational departments and information systems, a new kind of enterprise and web technology have been developed which provided a phase for integration [1].

This directed to the e-business projects compromising specific applications like CRM (Customer Relationship Management), SCM, ERP, PMS (Project Management Software), BI (Business Intelligence), legacy integration, etc. In order to control and optimize the information generation and consumption of these diverse enterprise-wide applications it turns out to be important to blend these using middleware resulting an EAI (Enterprise application integration) architecture. Information users started having interaction with diverse devices like mobile devices, smartphones, computer terminals and hardware like RFID (Radio-frequency identification) rose as original devices to collect stock movement information, etc. The need was presently felt to integrate these diverse applications running on diverse platforms and devices together in a distinctive manner. This led to the arrival of web-based principles for exchange data using text-based format like XML by Internet protocols and commonly known as the web services [2].

Businesses should use web services to discover the ups and downs of demand, driving loyalty by providing customers with active service, to create a real-time bill of materials that is created from suppliers' data, to capture and act upon fast-changing information in order to gain operational efficiencies, like reducing inventory buffers or shortening the ship-to-cash cycle [3].

Over the last years, manufacturers have perceived a quick fall in demand and have been left with too much inventory, excess production capacity, and frozen capital assets. While existing tools use historical data to make conventions about future demand, a firm scans respond to demand more effectively by using web services to support:

- Order fulfillment.
- Custom pricing.
- Product provisioning.
- Provide inventory alerts.
- Help small customers to buy
- Measure supplier performance.
- Publish and execute orders
- Query and consolidate shipping capacity.
- Aggregate and schedule production needs.

Web services are self-contained, self-describing and Internet-based modular applications. Web services are the new standard that allows machine-to-machine communication via the Internet using together simple protocols and interfaces. An Extensible Markup Language (XML)-based service description is important for the explanation of a web service. It also enables to share an Universal Description, Discovery and Integration (UDDI), which is a centralized service directory for its service discovery [4]. Web services are invoked as streams of services and give universal access for any kind of platform (interoperability). This interoperability amongst systems is the basis for active creation of business partnerships. Having an extensive requirement, web services are perfect for business services that need to be completely dispersed and distributed over the Internet. Consequently, web services become a seamless technical platform for allowing collaboration among partners or business items in a supply chain [4].

This study assumes a supply chain that contains, suppliers, manufactures and clients. The system is composed by an old Electronic data interchange system (EDI) by which the clients associated to the system make various orders for various periods of time. The manufacturer has to make the order by the time it was ordered, making this supply chain a make-to-order type of manufacturing process and with pull-type operation.

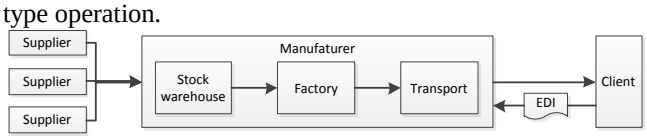

Figure 1. Make-to-Order (MTO) pull-type manufacturer schema.

The arrival and subsequent spread of e-business has rapidly and completely changed the styles of transaction employed in peer-to-peer (P2P) transactions, explicitly from off-line to online modalities. E-business has succeeded existing business models, mostly as the results of the simplicity and efficacy with which we can access and explore through huge quantities of knowledge and resources on the internet [5].

As Internet dispersal and use develops rapidly worldwide, and numerous new information and communication technologies are infinitely developed based on it, administrations in the private and public sectors are trying to exploit this inclination by creating large investments for the proposal, development, delivery and support of abundant types of e-services, such as e-business, e-banking, e-government and e-learning [6]. Due to continuous developments in Information and Communication Technologies and the fast nature of the business settings today, businesses make and deal with increasingly more data. Directors are frequently dazed with reports and information mixed out from a pack of business information systems such as Enterprise Resource Planning (ERP), scorecards, and business intelligence (BI) software that compete for the director's attention. This phenomenon is usually known as information overload. The problem is more worsened when reports are bad designed with low detail on how information is presented, which frequently divert than guide decision maker's attention. [7].

The information flow in this study is based in an old ERP system, which receives Electronic Data Interchange (EDI) orders, newer database (DB) systems for operational and business data transaction.

This study is based on a case study of an automotive seats manufacturing industry in Portugal, called Coindu S.A. Coindu specializes in the manufacturing of leather and other types of materials for car seat covers. Being the top textile producer, Coindu uses an EDI data for client orders, allowing it to build to order. When an order has arrived, an internal order for build is also created; this is only possible after analyzing all information about stock availability, work time window, lead time, logistics, etc. The company's decision managers have difficulties mainly in the right time data availability and fast information support system. Using a mature system for EDI client data interchange, the system cannot follow the fast pace needed for fast enough decision support system to be implemented (Figure 1).

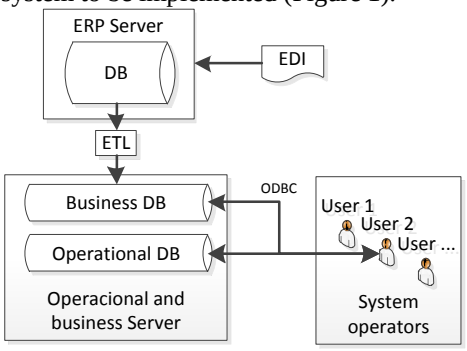

Figure 2. Information flow between users and system.

The schema shown above on Figure 1 represents the mature information flow, active in the company. Being a

very mature ERP system, the Extraction, Transformation and Loading (ETL) tool is slow to extract the data from the ERP DB making it an information bottleneck. Due to this fact the ETL tool will only work in low activity times such as during the night and only once. As a result, users can not have real time data information loaded in both the business and operational DB, therefore users typically make everyday decisions lacking important information that is always arriving to the ERP DB.

What we propose in our framework is a way to update in near real time the information required for a user. To do this our framework has to be able to be always updating, even during day time and also to be able to update specific tables requested from users. With this in mind, a different type of ETL tool was formulated and improved, by which it would be continuously and slowly updating the database and would have the ability to update on demand certain DB tables from our Data Warehouse (DW). We call this new ETL as, On Demand ETL (ODETL), as expressed on Figure 2.

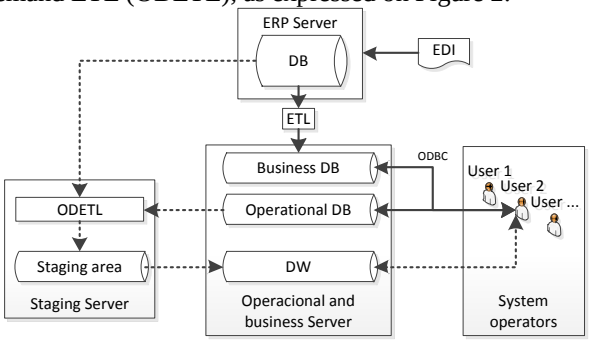

Figure 3. Proposed framework with new DW and ETL tool.

By using web services, we were able to able to create a service that would permit a user request an update and replying the expected time and position until update is completed. To show this information in an ordered manner, a monitoring dashboard was created for easy and comprehensible information.

Dashboards return information through visualization. Information visualization refers to the "use of interactive visual representations of abstract, nonphysical based data to amplify cognition" [11].

Visualization is efficient if the maximum volume of data is perceived in a minimum interval of time. Dashboards might offer a solution for the information overload problem by allowing an all-inclusive package for management, incorporating various ideas and applications such as strategy maps, scorecards, and BI into one solution. This study focuses in the way users can meet their data demand, in a fast way and always on time. Using web services as standard communication layer and a web server with connection to our ODETL and subsequent data stores, will permit using dashboarding methods, as a way for users to request and monitor their requests in real-time. This will allow our supply chain to be more flexible and fast which in turn will benefit and increase returns.

On the remaining sections of this paper, we will first further investigate the methodologies and technologies available in the market, as well as ERP software, web-services frameworks. Next we will present our proposed framework in more detail.

II. ERP VS WEB SERVICES

A business process is a group of connected, structured actions that produce a specific service or product for a specific customer or customers [8], as illustrated on Figure 4.

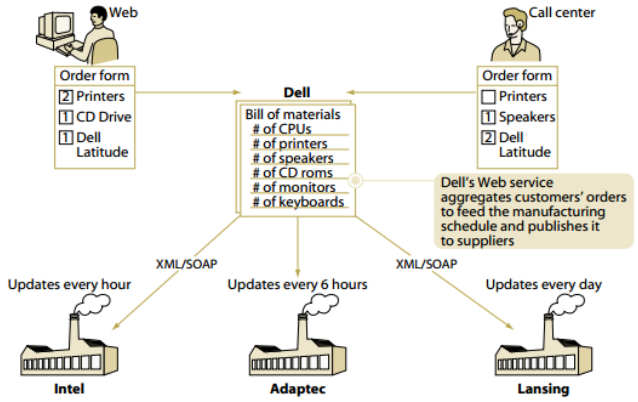

Figure 4. How Dell uses web services to publish its manufacturing schedule. Source: Forrester Research, Inc [8].

In this Internet-dominated business world, supply chain collaboration has become an essential means for partners in a chain to accomplish common goals that exceed a written contract agreement. Successful collaboration between partners helps them to achieve both strategic and operational advantages. At the operational level, collaboration brings visibility between both upstream and downstream partners, and also reduces the costs of inventory, premium services, and production scheduling. At the strategic level, collaboration enhances customer satisfaction through focused responses and also allows partners to manage their resources through a more flexible way.

Just like in kwon, Im & Lee (2011) we assume that changing customer demands cause high demand uncertainties and then the supply also result in uncertainties in lead-time and production capacity.

ERP system meanings have extended in range over the years. According to Davenport (2000), ERP systems are computer application packages that support numerous aspects of a company's information necessities by enabling the integration of company information into a central database. Others believe an ERP system is an enterprise-wide set of forecasting, planning, and scheduling tools, which links customers and suppliers into a complete supply chain [9].

ERP systems were designed as integrated standalone software systems including materials requirements planning, accounting, order entry, distribution and shop floor control functionalities. Alongside, ERP systems began gathering other supply chain related functions, such as demand forecasting, scheduling, warehousing, capacity requirements planning and logistics. Recently, two very important tendencies regarding ERP systems are being monitored, one relates to the addition of even more functionalities, such as

project management, content management, workflows, enterprise portals, customer relationship management, human resource management and knowledge management and the other is linked to the need to disintegrate large-scale ERP applications to autonomous, easily pluggable, reusable and loosely-coupled application components [9]. So, ERP systems may include autonomous parts from each other, which will operate like Lego bricks in an enterprise system in contrast to the previous closed non-modular architectures. As result, an enterprise will not have to obtain the entire enterprise software suite, but will be able to select each module even from diverse vendors and make a unique, cost-efficient and tailor-made solution [9]. Yet, these technologies and tools such as the EDI and the ERP systems often do not meet the necessities and requirements for handling construction supply chains, due to their high cost, inflexibility to modification, and absence of system extensibility [10].

Amongst the assumptions of the research, was that one big drawback of ERP systems, was the absence of Web-based modular and distinguishable ERP systems and absence of extended enterprise operation capabilities. In this setting, one drawback subcategory of the non-modularity of ERP systems is the absence of modular management of the supply chain [9].

In the context of ERP, web services offers two vital advantages: easiness of integration and cost drops through the hosted application model. Integration is a big source of spending for enterprises due to business software system complexity. Customers and outsourcing vendors might demand contact to information delivered to internal ERP users—like order status, inventory levels, and invoice data, without having ERP client software. This is where web services affect, allowing continuous data access to the authenticated users at the right time from everywhere without the need of specific software clients. With the ease of use of web services, integration can be attained with greater reliability, security, manageability, testing and effectiveness. Web services use object-oriented technology to blend data and programming elements in web service methods that can be retrieved by diverse applications. Web services allow proprietary applications to communicate over the web. Proprietary ERP applications and web services can use integration or other tools—such as SAP's Netweaver, HP's E-Speak toolsets; IBM's Dynamic e-business (infrastructure and software); and Sun ONE all of which enable data flow and communication across miscellaneous applications. Web-services combined with ERP deliver an integrated, multi-component application software platform perfect for performing multiple business functions.

Web services authorize the company to access information effectively. As this new technology advances, more vendors support web service solutions diversified with XML technologies. After that, web service broker hubs were introduced. A broker hub uses a portal that provides an interface for users so that they can locate, evaluate, subscribe to, and manage web services. Arising amounts of business software vendors are bringing web service broker hubs, such as SAP and Oracle. Microsoft with Navision is a vendor with a web service broker hub for ERP software users.

Lastly, web-based solutions are able to simply interoperate with the full supply chain entity, containing what is usually referred as Virtual Enterprise (VE) [9].

This idea can be understood through web-service methodologies such as SOA (service oriented architecture) and web-development techniques in general. These guidelines and developments can be drawn on the documentation of international enterprise software vendors like Oracle, SAP, PeopleSoft, or even in smaller ones like Exact Software Inc or Hyperion. Some vendors have developed different products in this philosophy such as the Oracle Fusion Middleware of Oracle [9].

Increasingly, companies are looking at SOA and their associated interfaces as an architectural plan and set of standards for addressing the integration requirements intricate in making multi-enterprise collaborative applications [12].

Impact of SOA adoption on electronic supply chain performance Lim and Wen identify case studies where SOA adoption led to cost savings and increased business efficiency [1]. Many large corporations have had successful implementations of web services in their e-commerce and electronic supply chain channels. For example, Motorola estimated that adopting web services based architecture led to savings of about €100,000 to €150,000. It was estimated that General Motors would be able to reduce operating costs by €1000 per vehicle by adopting web services [10, 12]. Thus, there is a broad agreement that SOA adoption leads to improvement in supply chain performance.

In order to effectively implement quality management, organizations that typically want to stand by the standards of quality management systems (such as ISO). Obedience with quality standards is important, since it guarantees that a precise company's products and services are gathering precise quality criteria. To accept each specific quality standard, numerous business processes, typically of bureaucratic nature, have to be carried out in order to correspond with the standard events that the enterprise really follows [9].

III. TECHNOLOGIES DESCRIPTIONS

Built on top of Java, JavaFX is designed to simplify the process of creating applications that can be deployed across devices ranging from cell phones to desktops, with little or no additional work being required to move your code between these different device types. JavaFX applications are written using JavaFX Script, a new and easy-to-use language that is introduced in the second part of this chapter [13]. The core of the platform is the JavaFX runtime library, which applications can use to build user interfaces, create animations, read data from RSS (Rich Site Summary), Atom feeds, play video and audio files, among other things. After a brief discussion of the features of the runtime library, we look at the development tools that you can use to build JavaFX applications, and then we examine the options available for packing and deploying your application.

The JavaFX platform consists of a compiler, a set of runtime libraries, and some developer tools, including plug-ins for the NetBeans and Eclipse integrated development environments (IDEs) that enable to develop JavaFX applications in a highly productive environment. One strength of JavaFX is that it runs on the Java platform, which means that an application written with JavaFX can make use of all the security and deployment features of Java and also has access to all the Java application programming interfaces (APIs) in addition to those provided by the JavaFX runtime itself.

Two Java frameworks have emerged to help with building RESTfull web services [14]. It implements concepts such as resources, representation, connector, and media type for any kind of RESTfull system, including web services [14].

SOAP, initially defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of web services in computer networks. It trusts on Extensible Markup Language (XML) as its message format and typically trusts on extra Application Layer protocols, most particularly Remote Procedure Call (RPC) and HTTP for message transmission. SOAP forms the basis layer of the web services protocol providing a rudimentary messaging framework with which abstract layers can be built on [8].

RESTfull web services have been recovering popularity, mainly with Internet companies. These also meet the W3C definition, and are often better integrated with HTTP than SOAP-based services. They do not require XML messages or WSDL (Web Service Definition Language) service-API definitions [8].

In simple terms, a RESTfull service should be accessible as a network-addressable resource that accepts a set of verbs defined by the implementer of that service. Those set of verbs usually map to HTTP methods (for example GET, POST, PUT and DELETE). This mapping is the common practice, but it is not mandatory. The following sequence diagram shows a typical RESTfull interaction where a client invokes a RESTfull service passing a verb and some content (like parameters) to be processed by that service.

The reply from the service could take one of different representations. XML is among those representations, but it could be any other format that makes sense to the service provider and consumer. For example if an Asynchronous JavaScript and XML (AJAX) application is what to communicate with your service it would make more sense to use JavaScript Object Notation (JSON) than XML. On the other hand, the example provided later on in this article uses raw binary format to exchange data between the client and the service to stress the fact that this format is arbitrary.

In the Restlet framework, both the client and server are components. Components communicate with each other through connectors. The most important classes in the framework are the abstract class Uniform and its concrete subclass Restlet, the subclasses of which are specialized classes such as Application, Filter, Finder, Router, and Route. Those subclasses work together to handle authentication, filtering, security, data transformation, and routing the incoming requests to the respective resources. The Resource class generates the presentation for the client.

IV. PROPOSED FRAMEWORK

As we said before, this framework tries to facilitate the updating of specific tables in an old ERP system, therefore increasing its agility and flexibility. The Figure 5 intends to schematize the global framework with all the system components.

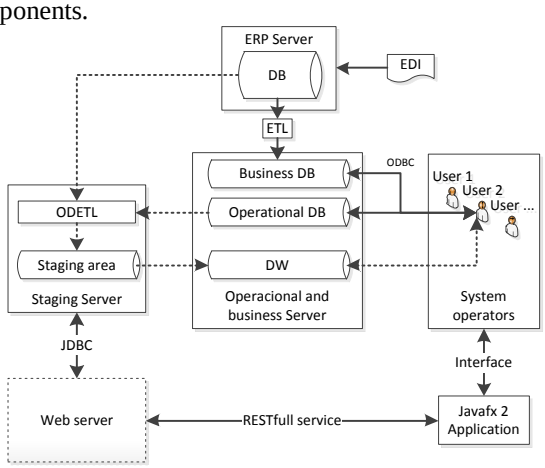

Figure 5. Global view of the proposed framework.

Figure 6 shows automated clients, which include Java and scripts of various languages Ajax, Flash, JavaFX, GWT, blogs, and wikis that run inside the browser and act as RESTfull web service consumers, which also belong to this group because they act in an automated fashion on behalf of the user. Automated web service clients send HTTP requests to the Resource Request Handler in the web tier. The stateless requests from the client contain the method information in the header, namely POST, GET, PUT, and DELETE, which will be mapped to corresponding operations of the resources in the Resource Request Handler. Each request contains all the necessary information including credentials for the Resource Request Handler to process the request.

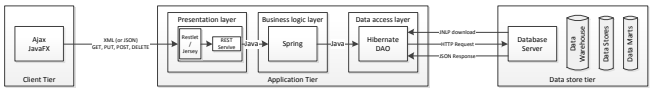

Figure 6. Diagram of the proposed multi-tiered web application environment.

V. CONCLUSIONS AND FUTURE WORK

The proposed framework brings a lot of indirect advantages, due to its main advantage, which in this case is the speed at obtaining valuable data at the right time. Knowing that changing hardware systems is typically very costly, the notion of small tweaks to the same old system, might bring almost the same efficiency as the newer one but at a fraction of its cost. Making use of the existing hardware and software, we tried to super charge the existing

framework in order to fully advantage the different types of data systems. The supported issue here is the availability of data at the right time for supporting manufacturing decision making along the supply chain. This increase in data speed retrieval will influence almost every department in the supply chain by speeding up their decision making capabilities. Making use of a different methodology for data retrieval, web services intermediate the communication between the user and the actual system in a much automated way. The proposed system might not be the most effective, but it is by far the most efficient for the present system.

Not being the only supply chain with this type of old system, we hope that our system might give another insight or way of achieving maximum performance from existing systems, just like this one, making use of the already available hardware and software, web services and open source software.

ACKNOWLEDGMENT

The authors wish to acknowledge the support of: 1) The Foundation for Science and Technology – FCT, under the scope of the financed Project on "Ubiquitous oriented embedded systems for globally distributed factories of manufacturing enterprises" - PTDC/EME-GIN/102143/2008, and 2) EUREKA, under the Project E!4177-Pro-Factory UES.

REFERENCES

[1] Vijay e Krishnan, M. S. 2007. Does SOA Improve the Supply Chain? An Empirical Analysis of the Impact of SOA Adoption on Electronic Supply Chain Performance. Kumar, Sanjeev, Dakshinamoorthy,. Proceedings of the 40th Hawaii International Conference on System Sciences.

[2] Mallick, Sudeep, et al., et al. Web services in the retail industry. Sadhana Vol. 30, Parts 2 & 3. April/June de 2005, pp. 159–177.

[3] Yates, Simon. The Web Services Payoff. s.l. : THE FORRESTER REPORT, 2001.

[4] Kwon, O., Im, G. P. e Lee, K. C. An agent-based web service approach for supply chain collaboration. Scientia Iranica. December de 2011, pp. 1545–1552.

[5] Park, JaeSung, Kim, JaeJon e Koh, Joon. Determinants of continuous usage intention in web analytics services. Electronic Commerce Research and Applications. January–February de 2010, pp. 61–72.

[6] Loukis, E. e Salagara, Pazalos A. Transforming e-services evaluation data into business analytics using value models. Electronic Commerce Research and Applications. March–April de 2012, pp. 129–141.

[7] Yigitbasioglu, Ogan M. e Velcu, Oana. A review of dashboards in performance management: Implications for design and research. International Journal of Accounting Information Systems. March de 2012, pp. 41–59.

[8] Teodoru, Silviu Florin. Business Process Management Integration Solution in Financial Sector. Informatica Economică vol. 13, no. 1. 2009, pp. 47-56.

[9] Tarantilis, C. D., Kiranoudis, C. T. e Theodorakopoulos, N. D. A Web-based ERP system for business services and supply chain management: Application to real-world process scheduling. European Journal of Operational Research 187 . 2008, pp. 1310–1326.

[10] Cheng, Jack C. P., et al., et al. A service oriented framework for construction supply chain integration. Automation in Construction. March de 2010, pp. 245–260.

[11] Wunsche, B. (2004). A survey, classification and analysis of perceptual concepts and their application for the effective visualization of complex information. Proceedings of the 2004 Australasian symposium on Information Visualisation, Australian Computer Society, 17–24.

[12] Yuen, K. (2010). Development of an enterprise decision platform: service-oriented architecture approach. *International Journal of Intelligent Information and Database Systems*, 156–17.

[13] http://www.oracle.com/technetwork/java/javafx/overview/index.html

[14] http://www.oracle.com/technetwork/articles/javase/index-137171.html