Technical Factors for Implementing SOA-Based Business Intelligence Architecture: An Exploratory Study

Lee-Kwun Chan[1], William Yeoh[2,] Wou-Onn Choo[1] and Phooi-Yee Lau[1]

[1]Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Perak, Malaysia

[2]School of Information Systems, Deakin University, Victoria, Australia

_____

Abstract

Business intelligence (BI) architecture based on service-oriented architecture (SOA) concept enables enterprises to deploy agile and reliable BI applications. However, the key factors for implementing a SOA-based BI architecture from technical perspectives have not yet been systematically investigated. Most of the prior studies focus on organisational and managerial perspectives rather than technical factors. Therefore, this study explores the key technical factors that are most likely to have an impact on the implementation of a SOA-based BI architecture. This paper presents a conceptual model of BI architecture built on SOA concept. Drawing on academic and practitioner literature related to SOA and software architectural design, we propose fourteen key factors that may influence the implementation of a SOA-based BI architecture. This study bridges the gap between academic and practitioners.

Keywords: Business Intelligence, Service-Oriented Architecture, Technical Factors.

_____

Introduction

In recent years many enterprises are turning their effort towards implementing Business Intelligence (BI) systems to improve decision making process (Gartner, 2011). A typical BI system includes functions such as reporting, querying, multi-dimensional analysis, online analytical processing, forecasting and data-mining, (Sharda et al., 2010; Turban et al., 2011; Becerra-Fernandez et al., 2010; Larson, 2009). Often, a typical enterprise-level BI architecture constitutes a data warehouse as a common back-end layer. According to Inmon (1992), data warehouse with its analytics capabilities is a non-volatile; integrated; theme-oriented repository to support long term decision making process. More significantly, a distributed multi-tier architecture has been widely adopted as a modern enterprise-scale BI architecture (Davenport et al., 2007).

In the traditional approach, client-server architectures are tightly coupled. Any changes made to the client-side might require relevant changes to be implemented on the server-side as well and vice versa (Dennis et al., 2010; Ashrafi et al.,2009; Cognos Corporation,2008). However, in the ever evolving business environments, enterprises require an architecture that can easily include, remove or integrate additional services 'on the fly'. In view of this, architectures built on SOA principles can separate implementation components from the underlying infrastructure. The separation enables the architecture to support the flexibility, inter-changeability, resiliency, security and availability needed in a modern enterprise-level BI architecture (Dennis et al., 2010;

Ashrafi et al., 2009; IBM Global Technology Services, 2007).

A few academic researchers have proposed their business intelligence success frameworks from managerial perspective (Yeoh and Koronios, 2010; Arnott, 2008; Dinter et al., 2009). However, none of the past studies was carried out in the context of business intelligence architecture from technical perspectives, nor from SOA concept. The effectiveness of business intelligence architecture might be influenced by internal and external factors. Therefore, this study aims to investigate the key technical factors that influence the implementation of an enterprise-scale SOA-based BI architecture.

In the remainder of this paper, we first outline the typical BI architectures from major practitioners and propose a conceptual model of SOA-driven business intelligence architecture. Section three provides a matrix of factors studies in prior research and presents the analysis of factors derived from both academic and practitioner literature. We then propose and discuss a new set of key technical factors for the implementation of a SOA-based BI architecture in section four. The final section provides the conclusion and indicates future research on this subject.

A Conceptual Model of SOA-Driven Bi Architecture

Erl (2005) defines service-oriented architecture (SOA) as a group of well-defined services which can be merged, reused and communicated with each other over networks. The introduction of service-oriented architecture changes many business operations into reusable services which are accessible over a network on demand (Erl, 2005; IBM, 2007). Within SOA environment, one of the advantages is the individual services that can be accessed without knowing the underlying system platform (Kodali, 2005). Nickull (2005) argues that reusability and repurposing are the main reasons for adopting SOA in implementing enterprise-scale BI systems. SOA enables low-cost system development but with good system quality because it is able to provide a flexible and standardised architecture that supports data sharing and integration of various systems. Other advantages of SOA includes flexibility, responsiveness, reusability, ease of connection, development cost reduction and agility (Erl, 2005; Dinter, 2009; IBM Global Technology Services, 2007; García-González,2009).

One of the main reasons for a BI project failure is largely due to the selection of inappropriate BI tools that fails to meet the specified business requirements. Therefore, Ponniah (2001) recommends that enterprises should design the architecture first, only then select the tools to match the functions and services stipulated for the architectural components. According to Friedman et al. (2004), there is a limited awareness of the value of architecture in the BI context. This creates challenges in data quality for many enterprises and the "hidden" aspects such as reliability, scalability and flexibility of BI are not being emphasized.

Davenport el al. (2007) assert that developing a robust BI system is more than just gathering and storing huge amount of data since it involves facets such as data quality, business processes, incentives, skills, organisational cultures, and sponsorships. Therefore, a BI architecture should be considered first when scoping a BI solution so that a more appropriate BI solution can be developed in meeting the actual needs of an enterprise. The finding was supported by Friedman et al. (2004), whom state that architecture is the foundation of BI and paying attention to the architecture will ensure success in BI implementation. Watson et al. (2005) also emphasize that architecture has significant impact when users can easily access the business data, which in turn leads to improved decision-making capabilities. In order to meet the requirements of an emerging business environment, a BI architecture should be designed in a flexible and adaptable manner (Davenport et al. 2007). Invariably, an agile BI architecture design will be better off serving the needs of an ever-changing business environment.

According to Ponniah (2001), architecture is the combination of different unique components which provide a mean to store data, and deliver information to users. Architecture is a template that contains rules and functions used for serving business requirements. The various elements in architecture such as the standards, measurements, designs and other supporting techniques, aim to enable the smooth data flow from the source to the destination within a framework (Ponniah, 2001). A BI architecture that comprises of different systems, applications, and processes in an enterprise enables decision makers to access the valuable information for complex analytical processes. The BI architecture should be able to provide the correct information promptly in supporting the decision making processes. Hence, the information must be distributed via multiple channels including emails and page alerts spreadsheets, analytic queries, scorecards and dashboards (Davenport et al., 2007). In other words, BI architecture is anything that enables the transformation of data into useful information in decision-making process and to acquire evolving business advantages. Howson (2007) asserts that a fundamental architecture is the essential element for all BI system deployments. A lower level BI architecture can be formed by transactional systems and front-end tools. On the other hand, a higher level BI architecture deployment may consist of data marts, data warehouses, ETL (extract, transform and load) tools and BI front-end tools.

A study was separately performed on the BI architectures by two major BI vendors, namely, IBM and Microsoft. We find that both of the proposed architectures consist of five similar layers: data access/presentation layer, data analysis layer, data repository/data storage layer, data integration layer, and data source layer (IBM, 2004 and Microsoft, 2006). Moreover, Cognos (2008) provided a business intelligence architecture example which is built on service-oriented concept. However, Cognos's BI architecture deploys three distinct tiers to deliver BI capabilities; namely presentation tier that handles interaction with users over the network;

application tier to handle all BI processing with special-built services; and data tier that provides access to various data sources (Cognos Corporation, 2008). Nonetheless, although Cognos's BI architecture is built on SOA concept but it is insufficient to demonstrate architecture functional independence. Hence, a new SOA-based BI architecture is proposed which enables flexible deployment and integration of independent services. These can be assessed without knowing the underlying computing platforms. The benefits of having SOA include flexibility, responsiveness, reusability, ease of connection, cost reduction and agility (IBM Global Technology Services, 2007). In addition, we propose an additional separated layer which is called the communication tier to achieve functional independence. This layer solely describes about the communication aspect in term of open standards such as Simple Object Access Protocol (SOAP) and Extensible Markup Language (XML), Application Program Interfaces (APIs), Hypertext Transfer Protocol (HTTP) and decision services middleware like Web Services Description Language (WSDL) and Universal Description, Discovery and Integration (UDDI).

In view of the challenges and limited capabilities of existing SOA-based architecture, a six-layer SOA-based BI architecture conceptual model is proposed to deliver BI capabilities in the constantly evolving business environments. Figure 1 depicts conceptual SOA-based BI architecture. The details of the model's layers are described in the following sections.
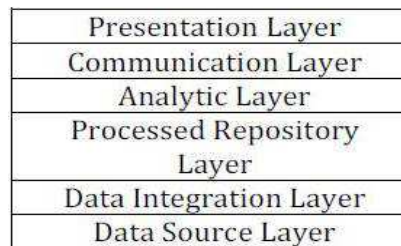
Figure 1. Conceptual SOA-Based BI Architecture

Presentation Layer: The top layer is the data presentation or data access layer which provides access to the users via various means such as host computers, portal, web service or even handheld mobile devices. Furthermore, users are able to view reports in various formats such as chart, table, report, dashboard, scorecard, and aggregated metrics. The reports are equipped with navigation capabilities like drilling, pivoting, drag-drop so that users can seamlessly analyse the data in any reports.

Communication Layer: Open interface is used by all services to communicate with each others and it should take place in the communication Layer. This can achieve location transparency among services. Open web standards, such as Simple Object Access Protocol (SOAP), Extensible Markup Language (XML) and Web Services Definition Language (WSDL) should be adopted for communication purpose.

Analytic Layer: Since a BI system contains huge volume of data, tools and techniques from data integration layer, data analysis layer is critical for managing, summarising, querying and analysing the data. It provides analytics functions such as ad-hoc query, reporting, modelling, embedded analytics, Online Analytical Processing (OLAP), data mining and collaboration. Data is organised into multidimensional cubes based on a dimensional model which enables summarising of information for analytical queries in an efficient manner. Data analysis leverage the information stored in data storage.

Processed Repository Layer: Data Storage is the end result of Data Source and Data Integration layers. Data Storage can be termed as Data Warehouse (DW)/Data Mart (DM). According to Microsoft architecture, it involves elements such as dimensional modelling, partitions and indexes. Meanwhile IBM proposes operational data stores, staging areas and metadata in this layer.

Data Integration Layer: BI often involves analysis of aggregated and integrated data from various operational systems. Data is extracted, cleaned, profiled, staged and filtered from the operational systems before it is loaded into a data warehouse. A data integration layer primarily consists of ETL process (extracting, transforming, and loading processes) (Howson, 2007).

Data Source Layer: Data are collected from various resources at the data source layer where it includes different sources, different platforms and different operating systems. It could be operational data which are generated by internal departments, unstructured data, informational data and external data. Operational data can be generated from internal operational systems as well as external enterprise systems such enterprise resource planning system, customer relationship management system and supply chain system. Unstructured data may also come from external sources in various data formats such as from suppliers, customers or any other information on the Internet.

Literature Review of Factors for SOA and Software Architectural Design

There is limited academic research into the technical factors that affect SOA-based BI architecture implementation. As depicted in Table 1, this research develops an aggregated matrix based on rigorous

examination of the academic and practitioner literature related to SOA, BI and software architectural design. O'Brien et al. (2007) proposed some factors for SOA implementation whereas Meier (2009) suggested several factors to be considered for architecture design. O'Brien et al. (2007) discussed the quality attributes for SOA but did not group the attributes into appropriate category. Meier (2009) put forward the quality attributes in four categories, namely system, runtime, design-time, and user qualities in an architecture

framework which demonstrates a simple way to organise the space. As shown in Table 1, O'Brien and Meier have different set of quality attributes for service-oriented architecture and architectural design respectively. On the other hand, the design factors identified by Dinter et al. (2009), i.e. excellence in BISOA, excellence in embedded BI, process orientation and excellence in data management, focus merely on managerial or organisational context but not technical factors for SOA-based BI architecture.

Table 1. An Analysis of Factors from Literature

	O'Brien et al. (2007)	Meier (2009)	Mulik et al.(2009)	Rotem-Gal-Oz et al.(2009)	Shim et al. (2008)	Dong et al.(2008)	Liu et al.(2007)
Interoperability	x	x	x				
Usability	x	x					
Agility		x					
Manageability		x					
Security	x	x	x			x	
Availability	x	x		x		x	
Reliability	x	x	x			x	x
Scalability	x	x					
Conceptual integrity		x					
Maintainability		x					
Supportability		x					
Testability		x		x			
Portability				x			
Effectiveness					x		
Understandability					x		
Flexibility					x		
Reusability		x			x		
Modifiability	x						

Key Technical Factors for SOA-Based Bi Architecture

Dennis et al. (2006) posit that there are four primary non-functional requirements that should be considered in the design of client-server architecture: operational requirements, performance requirements, security requirements and political requirements. In this study, it is found that most of technical factors for SOA-based BI

architecture fall in the category of operational requirements and performance requirements. Hence, security requirements and political requirements will not be considered in this study. As provided in Table 2, we propose a more comprehensive set of 14 key technical factors for implementing BI architecture with SOA concept. The factors within corresponding dimensions are discussed next.

Table 2. Suggested Factors That May Influence SOA-Based BI Architecture

Dimension	Factor	Description	Author(s)
Operational Dimension	Deployability	The extent to which deploying the BI system has to be a simple activity, as does making changes to the way information is deployed	O'Brien et al. (2007)
	Interoperability	The extent to which a BI system has to provide a single interface for all BI capabilities and existing infrastructure. Users can navigate scorecards, dashboards, or reports all in one product.	Mulik et al.(2009), Lakshmanan et al. (2009), O'Brien et al. (2007),Meier (2009)
	Commonality	The extent to which a BI system has to deliver a single and common view of the business ensuring the validity and maintaining data quality. A readily modified on demand business model can be created with data modeler.	New factor proposed by the authors of the present study
	Single open API	The extent to which BI architecture should integrate seamlessly with other application and systems such as planning and budgeting with a single application programming interface (API).	New factor proposed by the authors of the present study
	Openness	The extent to which BI will operate with other information system, broad access to enterprise data and data integration with existing applications, portals and security systems.	New factor proposed by the authors of the present study
	Usability	The extent to which underlying BI architecture must be shaped to allow different types of user from various work environment in retrieving information in a broad range of formats and technologies including mobile devices on demand.	Ashrafi et al.(2009), O'Brien et al. (2007), Meier (2009)
	Agility	The extent to which underlying architecture must be able to adapt the changes in enterprise business strategy.	Grigoriu(2006) Lewis(2009)
	Manageability	The extent to which administrator able to manage it effectively by identifying potential issues proactively before they become problems and maintaining optimal system performance.	Fraiß et al. (2009), Pedrinaci et al.(2009), Meier (2009)
	Security	The capability to ensure the system is protected from loss and unauthorized access to information as well as the possibility of a malicious code attack.	Ashrafi et al.(2009), Mulik et al.(2009), Dennis et al. (2010), O'Brien et al. (2007), Meier (2009), Dong et al.(2008)
Performance Dimension	Response Time	The extent to which system must performs its functions within a specific duration.	Ashrafi et al.(2009), Dennis et al. (2010), Mulik et al. (2009)
	Throughput	Throughput defines rate at which incoming requests are completed and it may be the number of transactions per second.	Dennis et al. (2010), Mulik et al.(2009)
	Availability	The extent to which a BI system has to operate on a 24x7 basis with the exception of scheduled maintenance.	Dennis et al. (2010), Fraiß et al. (2009), O'Brien et al. (2007), O'Brien et al. (2007), Dong et al.(2008), Rotem-Gal-Oz et al.(2009)
	Reliability	The extent to which as BI system has to grow in fulfillment of its functions and sophistication, the application must remain responsive.	Ashrafi et al. (2009), Dennis et al. (2010), Mulik et al.(2009), Fraiß et al. (2009), O'Brien et al. (2007), Meier (2009), Dong et al.(2008), Liu et l.(2007)
	Scalability	The extent to which as BI system must provide support of ten thousands of users and provide a support of terabytes of data across a global organisation in a linear fashion.	Ashrafi et al.(2009), Dennis et al. (2010), Fraiß et al. (2009), O'Brien et al. (2007)

Operational Dimension

Operational requirements specify the operating environment in which a BI system must perform over a period of time (Dennis et al.,2010; Ashrafi et al.,2009;Bennett et al.,2006). Referring to past studies shown in Table 2, we found that interoperability, usability, agility, manageability, and security are common factors in both SOA and software architecture. In view of the factors that could affect the efficiency of a BI system, we propose several factors such as commonality, single open API, openness in addition to SOA and software architecture common quality attributes.

Interoperability is one of the major benefits of SOA because it enables service interoperate seamlessly on different platforms. O'Brien et al. (2007) emphasize that the SOA architecture should be designed and evaluated with care in order to keep away from any drawbacks in performance since interoperability may cause performance overhead. Besides that, location transparency in SOA is able to improve system performance via availability of a system when services are available in multiple places and transparent to the users. Usability defines how well the application meets the user requirements. O'Brien et al. (2007) indicate that the nature of SOA in distributed computing has a significant impact on usability when user request to remote service providersis processed. According to Clements et al. (2002), modifiability is the ability for a system to make changes promptly and cost-effectively. This factor is semantically similar to agility. Both O'Brien and Meier agree that this is an important factor so that a software system can continually sustain in an ever changing business environment. Manageability is the extent of how easy it is to manage, monitor, and debug an application (Meier, 2009). Manageability is a run-time quality of the system. Meier (2009) identifies three key issues for manageability. These key issues are lack of information on monitoring, tracing, and diagnostic, lack of runtime configurability and lack of troubleshooting tools. O'Brien et al. (2007) further assert that security is a major concern for SOA and Web Services; architects should pay attention to factors that directly impact security. Dong et al. (2008) mention that a service-oriented system is able to support fault detection, authentication capabilities as well as automation of trust negotiation. Security is associated with three core areas, namely confidentiality, integrity and authentication in which can be addressed by a string security protocol such as Secure Socket Layer (SSL). However, distributed components in SOA can enhance scalability, manageability while cutting costs.

Performance Dimension

Performance requirements specify the response time and the input-output volume that the system can handle within a particular timeframe (Dennis et al., 2010; Ashrafi et al.,2009; Bennett et al.,2006). O'Brien et al. (2007) highlight that prior to a system implementation, the SOA architecture should be designed and evaluated because performance has negative impact on SOA in most cases. However, the key factors that contribute to performance issues mentioned by O'Brien et al. are more related to technical perspective such as the use of interaction protocol, the use Simple Object Access Protocol (SOAP) and the use of a standard messaging format via Extensible Markup Language (XML). In addition, availability, reliability, scalability are the common factors in both SOA and software architecture design leading to the success of a SOA-based system. We propose that response time, throughput, availability, reliability, scalability to be mapped into performance dimension. O'Brien et al. (2007) point out that availability is a concern for the success of a SOA from the user's and provider's perspectives. According to Meier (2009), failure of a physical tier such as the database server or application server may lead to failure of entire system. Availability defines the proportion of time that the system is functional by measuring the percentage of the total system downtime over a predefined period. Liu et al. (2007) assert that availability is the quality attribute of whether the web service in SOA is ready for

use. Availability may be affected by system and infrastructure fault, malicious code attacks, and system load (Meier,2009). The implementation of distributed transactions involve services that may be implemented in different languages and platforms, thus it is critical for a SOA-based system to operate correctly without failure (O'Brien et al., 2007). Meier (2009) defines reliability is the capability of a system will not fail to remain operational over a specified period. Availability can be affected by system errors, infrastructure problems, malicious code attacks, and system workload. O'Brien et al. (2007) emphasize that since Web services technology does not offer any inherent scalability feature, thus an increasing number of users or system size and volume without performance degradation can be a major issue in SOA. Meier (2009) also address that scalability is a critical issue because the system is supposed to be able to function well when there are changes to the load or demand.

Conclusion

This paper has presented a conceptual model of business intelligence architecture built on SOA concept. Drawing on academic and practitioner literature related to SOA and software architectural design, a set of key technical factors that may influence the implementation of a SOA-based BI architecture have been proposed and discussed. Commonality, openness and single open API factors have also been suggested in this paper and subject to further empirical study. Amongst the findings, the research indicates that the technical factors generally exist in two major dimensions composed of operational dimension and performance dimension. Although industry has placed more focus on the implementation of SOA-based BI architecture, the contemporary approach of offering flexible BI solutions, the literature review indicates little academic research on the key technical factors involved in successful SOA-based BI architecture implementation. Therefore, this research provides a fundamental reference point for further empirical research to shed light on where and how the technical factors influence the implementation of a SOA-based BI architecture.

References

Ashrafi, N. & Ashrafi, H. (2009). Object-oriented Systems Analysis and Design, *Pearson Education Inc.*,Upper Saddle River, New Jersey.

Becerra-Fernandez I. & Sabherwal R. (2010). Business Intelligence: Practices, Technologies, and Management, *John Wiley & Sons Inc.*

Bennett, S., Farmer, R. & McRobb, S. (2006). Object-oriented Systems Analysis and Design using UML, 3rd Edition. McGrall Hill.

Clements, P., Kazman, R. & Klein, M. (2002). 'Evaluating Software Architectures,' *Addison-Wesley,* Boston.

Cognos Corporation (2008). The Right Architecture for Business Intelligence: The Foundation of Effective Enterprise BI. *White Paper.*

Davenport, T. H. & Harris, J. G. (2007). Competing on Analytics: The New Science of Winning. Harvard Business School Press, 240.

Dennis A., Wixom B. H. & Roth R. M. (2010). System Analysis and Design. 4rd. *John Wiley & Sons,Inc.*

Dinter, B. & Stroh, F. (2009). Design Factors for Service-Oriented Architecture Applied to Analytical Information Systems: An Explorative Analysis. 17th European Conference on Information Systems ISBN: 978-88-6129-391-5Verona, Italy, 2687-2698.

Dong, J., Paul, R. A. & Zhang, L.- J. (2008). "High-Assurance Service-Oriented Architectures," *Computer,* 41, 27-28.

Erl, T. (2005). Service-Oriented Architecture: Concepts, Technology, and Design. *Prentice Hall.*

Fraiß, M. & Zinser E. (2009). Speed Up to the Next Level of Enterprise Architecture. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Friedman, T. & Strange, K. H. (2004). 'Architecture: The Foundation of Business Intelligence,' *Gartner Research*, AV-226453.

García-González, J. P., Gacitua-Decar, V. & Pahl, C. (2009). Service Registry: A Key Piece for Enhancing Reuse in SOA. The Architecture Journal:Service Orientation Today and Tomorrow. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Gartner Press Release, (2011). "Gartner Says Worldwide Business Intelligence, Analytics and Performance Management Software Market Surpassed the $10 Billion Mark in 2010," [Online], [Retrieved August 11,2011], http://www.gartner.com/it/page.jsp?id=16 42714

Grigoriu, A. (2006). An Enterprise Architecture Development Framework: The Business Case, Framework and Best Practices for Building Your Enterprise Architecture. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Howson, C. (2007). 'Successful Business Intelligence: Secrets to Making BI a Killer App,' *McGraw-Hill.*

IBM Corporation (2004). Introduction to the BI Architecture Framework and Methods. *IBM Software Group,* DB2 Data Management Software. White paper.

IBM Global Technology Services (2007). Infrastructure Considerations for Service-Oriented Architecture. White paper.

Inmon, W. H. (1992). Building the Data Warehouse.

Kodali, R. R. (2005). What is service-oriented architecture?. [Online], [Retrieved August 11,2011] http://www.javaworld.com/javaworld/jw-06-2005/jw-0613-soa.html

Lakshmanan, G. & Manish, P. (2009). How the Cloud Stretches the SOA Scope. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Larson, B. (2009). Delivering Business Intelligence with Microsoft SQL Server 2008, *McGraw-Hill/Osborne.*

Lewis, G. A. (2009). Is SOA Being Pushed Beyond Its Limits?. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Liu, Z., Gu, N. & Yang, G. (2007). A Reliability Evaluation Framework on Service Oriented Architecture. *IEEE.*1-4244-0971-3.

Meier, J. D. (2009). A Language for Software Architecture. The Architecture Journal TechEd Special Edition. *Mircosoft.* [Online], [Retrieved February 11,2012], http://www.architecturejournal.net

Microsoft Corporation (2006). 'Business Intelligence Guidelines Conceptual Framework,' White paper.

Microstrategy (2007). Enterprise Business Intelligence, Improving Corporate Performance through Integrated Reporting, Analysis, and Monitoring. *White paper.*

Mulik, S. & Godse, M. (2009). Design-Time Selection of Web Services for Composite Applications. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Nickull, D. (2005). 'Service Oriented Architecture Whitepaper,' Adobe Systems, Inc, pp.2. White paper.

O'Brien, L., Merson, P. & Bass, L. (2007). "Quality Attributes for Service-Oriented Architectures," International Workshop on Systems Development in SOA Environments SDSOA07 ICSE Workshops 2007, *IEEE Computer Society,* 3-3.

Pedrinaci, C., Simperl, E., Krummenacher, R. & Norton, B. (2009). Toward Web-Scale SOA. [Online], [Retrieved August 11,2011], http://www.architecturejournal.net

Ponniah, P. (2001). Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals. *John Wiley & Sons,* 127.

Purewal, K., Yang, L. & Grigg, A. (2009). "Quantitative Assessment of Quality Attributes in Systems Architecture Using Evidential Reasoning," In.

Rotem-Gal-Oz, A., Bruno, E. & Dahan, U. (2009). SOA Patterns.ISBN: 1-933988-26-6.

Shim, B., Choue, S., Kim, S. & Park, S. (2008). A Design Quality Model for Service-Oriented Architecture. 15th Asia-Pacific Software Engineering Conference. *IEEE.*

Turban, E., Sharda, R. & Delen D. (2010). Decision Support and Business Intelligence Systems, 9th Ed., *Prentice-Hall.*

Turban, E., Shard, R., Delen, D. & King, D. (2011). Business Intelligence: A Managerial Approach, Second Edition, *Pearson Prentice Hall,* Upper Saddle River, New Jersey.

Watson, H. J. & Ariyachandra, T. (2006). 'Benchmarks for BI and Data Warehousing Success,'

Wu, L., Barash, G. & Bartolini, C. (2007). "A Service-oriented Architecture for Business Intelligence," Proceedings of the IEEE International Conference on Service-Oriented Computing and Applications (SOCA '07), 279-285.

Yeoh, W. & Koronios, A. (2010). Critical Success Factors for Business Intelligence System, *Journal of Computer Information Systems,* 50(3), 23-32.