Academic Global Publications
ABN 77 114 568 115

**International Technology Management Review**
**Volume 1 Number 2 (November) 2008**
**<**http://www.academicglobalpublications.com/itmr/>

# A Review of Emerging Technology Trends in E-Commerce

**Dr. Qingxiong Ma**
Department of Computer Information Systems
University of Central Missouri
Dockery 300B, Warrensburg, MO 64093
Email: qma@ucmo.edu
Phone: 1-660-543-8637

## Abstract

**Title:** A Review of Emerging Technology Trends in E-Commerce

**Keywords**: Emerging Technology Trends, Application Service, Web Service, Grid Service, Service-Oriented Architecture, E-commerce

**Category of paper:** Review paper

**Purpose of the research:** After several years of growth, E-commerce has become mainstream and matured. However, the new challenges require that tomorrow's E-commerce system move beyond the basic functions such as a static website with electronic catalogue and/or a shopping cart towards an intelligent, dynamic, and secure commerce system. This research project attempts to explore the emerging technologies that can solve these issues such as systems integration and information sharing. Specifically, the nascent technologies introduced include application services, Web Services, grid computing and their combinations. The benefits and perils of each technology as well as the potential factors that influence businesses to adopt them are also discussed.

**Methodology:** Qualitative

**Implications for practices:** The description and discussion of the emerging technology trends will have implications for businesses in making strategic decisions and thus increase their business sustainability and competitiveness.

**Value of the paper:** This paper will be of interest to business students, academics, and business planners and decision makers as well.

**Number of pages:** 15

**Number of figures:** 2

**Section headings:** Abstract, Introduction, Application service, Web Services, Grid computing, Service-oriented architecture, Grid service, Conclusion

Academic Global Publications
ABN 77 114 568 115

**International Technology Management Review**
**Volume 1 Number 2 (November) 2008**
<http://www.academicglobalpublications.com/itmr/>

# A Review of Emerging Technology Trends in E-Commerce

## Abstract

After several years of growth, ecommerce has become mainstream and matured. However, the new challenges require that tomorrow's ecommerce system move beyond the basic functions such as a static website with electronic catalog and/or a shopping cart towards an intelligent, dynamic, and secure commerce system. Businesses are expecting to streamline their business processes and have an efficient information system so that they can share information and maximize business investments and opportunities. This research project attempts to explore the emerging technologies that can solve these issues such as systems integration and information sharing. Specifically, the nascent technologies introduced include application services, web services, grid computing and their combinations. The potential factors that influence businesses to adopt ecommerce are discussed. Knowledge of these new technology trends will definitely help businesses in making strategic decisions and thus increase their business sustainability and competitiveness.

## Introduction

E-commerce is no longer an infant now. It has reached toddler-hood (Johnson 2005). According to the statistics of Forrester Research, worldwide total E-commerce was $6,790 billion in 2004. Research and analysis firm Jupiter Research (Business Wire Jan. 17, 2007) has forecasted that US online retail sales will grow by 16 percent in 2007 to reach $116 billion. Over the next five years US online retail will grow at a CAGR of 11 percent reaching $171 billion in 2011. No doubt, these figures imply that E-commerce has become mainstream and matured.

However, researchers and business practitioners noticed that many challenges and problems inhibit the development of E-commerce and hurt businesses' revenue. First, the customer's confidence is low, which leads to low conversion rate. Conversion rate refers to how many visitors turn into actual buyers. A recent Neilsen/NetRatings survey found that the average sales conversion rate among the top 100 retailers is 4.9 percent (Yen 2005). For many start-up businesses, this rate is less than 1 percent. Second, security is an important issue in E-commerce. The survey report from Javelin Strategy & Research Survey (2007) indicated that the US total fraud amount is $55.7 billion and the average fraud amount per victim is $6,278 in 2006. In fact, the growth of computer attacks and incidences is exponential. Third, the operation cost of running E-commerce is high. In today's E-commerce, information sharing and capturing are becoming the bottleneck of information flow. Business needs to automate business processes in stores and across the enterprise (Maranville and Brown 2006) and share information with their customers, suppliers and distributors. For example, according to BIGresearch (2005), 75% of surveyed consumers said they regularly or occasionally go online to research before making an in-store purchase, retailers should respond with multichannel marketing and integration. Those who are not integrating their in-store marketing with online initiatives risk losing incremental sales opportunities.

To gain competitive competence, businesses have to meet the challenges and solve E-commerce problems. In order to increase customer satisfaction and productivity at a lower cost, many measures can be taken. Besides using promotion activities (such as free shipping, gifts, and discounts) many businesses are turning to differentiation and innovation to attract and keep customers by providing purchase decision support (Ma and Souren 2003), providing a secure, customer-centric, more interactive shopping experience as well after-sale services such as a return policy. They are attempting to integrate their systems by increasing spending on supply-chain improvements and better store systems and planning applications. In 2006, hardware investments will likely focus on core POS, payment systems, mobile devices, and kiosks to replace aging infrastructure and enable lower operating and maintenance costs, improved store associate performance, and greater consumer satisfaction (Sullivan 2005).

One successful business case in business process restructuring, system integration, and information sharing is FedEx Kinko. They used to have disparate technologies, standalone systems, and unique customer data canters. This business process led to duplicated data entry, long credit authorization, difficult labour sharing and pricing controls. After they integrated the business process, they are using integrated applications with single shared database. Consequently, they improved labour efficiency, reduced costs, increased flexibility in pricing and accuracy in customer information capturing, optimized workforce sharing, and improved service quality. In general, on the front-end, they improved customer service and increase quality by implementing customer order entry, customer identification, and automated pricing. In the back office, they lowered credit rates, increased labour efficiencies, and removed data entry redundancies. Enterprise-wide, they use a centralized repository to share information and business intelligence to help the decision process.

This paper is focused on discussing some of the major emerging technologies and how they can provide opportunities for businesses and help them solve E-commerce problems. Specifically, the nascent technologies that will be introduced include application services, Web Services, grid computing and their combinations. Knowledge of these new technology trends and factors will definitely help businesses in making strategic decisions and thus increase their business sustainability and competitiveness.

## Application services

Currently, information technologies/information systems are considered more as services than products. Especially in client/server architecture, the term "service" is used to indicate the ability to make a remote call. With web and Internet technology, software can be hosted remotely on an application server and can be called a service by a client using a web browser. With this concept of information service, business integration needs, and distributed computing technology, the way software applications are designed, architected, delivered and consumed is changing. Information service has a hierarchical structure that consists of several layers. It can either be software applications or components of applications. Among the service-oriented computing technologies, a*pplication services* were introduced a few years ago.

An application service provider (ASP) is a supplier of application services. Application service provision represents an e-business model of supplying and consuming applications over computer networks. Applications provided by an ASP may be as simple as single software or as complex as an enterprise resource planning (ERP) system. An ASP assumes responsibility for buying, hosting, and maintaining a software application on its own facilities, publishes its user interfaces over computer networks, and provides its clients with shared access to the published user interfaces. The client organizations, on the other hand, subscribe and receive the application services through the Internet or a dedicated network connection as an alternative to hosting the same application in-house.

The ASP model essentially allows organizations to hand over the responsibility of systems deployment or its execution to an outside vendor while still satisfying self-information needs. It reduces the complexity associated with the traditional make-or-buy model while allowing effective control of the deployment costs and risks. On the one hand, an ASP can amortize expenditures over its entire client base, enabling it to improve quality of services, security, and risk reduction measures that individual clients may find cost-prohibitive. On the other hand, client organizations do not incur the costs associated with traditional software implementation, including software license fees, hardware investment, and staffing and training of system administration personnel. They avoid nightly data backup, monthly software updates, loss of data because of local hard drive or server failures, and the need to contract with a technical support group. By eliminating the need to manage hardware, software, information, and personnel, they can focus on their core businesses and free up resources for mission critical applications. By eliminating the need to evaluate, purchase, deploy, and test hardware and software, applications can be up and running in a matter of weeks, instead of months or even years. One successful example of an ASP is Salesforce.com. They provide tools on customer relationship management.

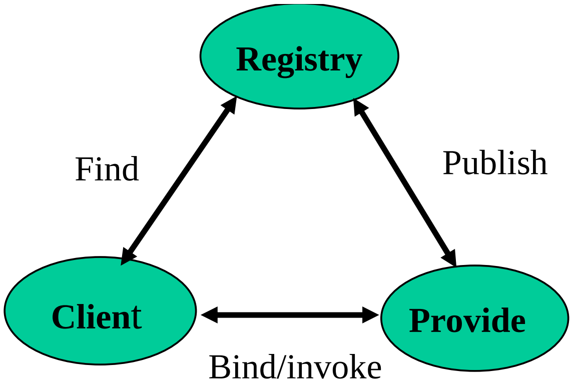
However, customers have generally been dissatisfied with ASP offerings, and many have seen their ASPs go out of business (Susarla, Barua & Whinston 2003; Wendy, Currie, Philip & Graham 2007). This is not just because the ASPs were not economically viable, but also because the service model concepts were still fairly new to the software world, and the ASP model itself had some substantive practical issues that hindered its widespread acceptance. First, the ASP model did not meet the need for information integration, especially for data exchange between business partners. It is not uncommon to find business partners running different applications. The increased complexity of integration with other applications was perceived as a weakness of the ASP model. Second, many companies had concerns about trust and security in the ASP model. They were reluctant to put their vital data on a remote server run by unknown providers over the open Internet. Third, many businesses lost the flexibility to control the application because of the centralized and standardized management in ASP model.

As the outsourcing trend in capital and labour continues, software application outsourcing across the Internet will become a major wave too. Although the ASP model had a difficult start, the fundamental concepts of virtualization and ‘on-demand rent’ give it a compelling advantage. For enterprise software firms, those who do not see software as a service (SaaS) may not survive in an increasingly competitive market (Miranda 2005).

## Web Services

Web Services are the current most promising technology and have been getting a lot of attention since 2003. Different from ASP, where the whole systems or applications are provided as a service, Web Services provide components for use. These components do not have user interfaces. Instead, they have system or application programming interface (API). A service has a network-addressable interface. Because Web Services are small and independent components of applications, representing business functions or business services and accessible by another program remotely, Web Services stress interoperability and may be dynamically discovered and used. To illustrate how a Web Service works, a typical Web Service model is shown in Figure 1.

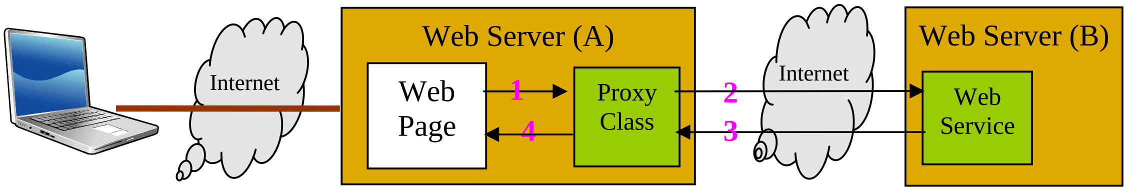
Figure 1 Generic Web Service Model



In this model, there are four different standards involved. When a Web Service is created, it is registered through UDDI (Universal Description, Discovery and Integration), so that it can be easily discovered by users. The UDDI governs service publication and discovery. Each registered Web Service has a description, written in the WSDL (Web Service Definition Language), regarding the location of the service and what kind of operations (or methods) it has and how it can be used (like the Yellow Pages). All messages are written in XML (Extensible Markup Language)-based text, which follows SOAP (Simple Object Access Protocol) coding and formatting specifications, instead of cryptic binary strings. This mechanism enables Web Services to communicate with other applications that may be developed in different programming languages and reside on different platforms. Thus, Web Services are autonomous, platform-independent computational elements that can be described, published, discovered, orchestrated and programmed using standard protocols for the purpose of building networks of collaborating applications distributed within and across organizational boundaries.

When a web application calls the Web Service, it does not communicate to the Web Service directly. Instead, the communication is accomplished through a proxy class on the local web server as shown in Figure 2.

Figure 2 The Role of Proxy Class



Generally, there are four steps for a Web Service to complete a transaction:

1) The Web page on Web Server A instantiates an instance of the proxy class.
2) The proxy class marshals the parameter list and makes an HTTP request to the Web Service residing on Web Server B
3) The Web Service unmarshals the incoming parameters, runs the method, and marshals the output parameters. These are sent back in an HTTP response.
4) The proxy class unmarshals the return parameters and passes back the result to the Web page.

With marginal costs, each organization can expose some or all functionalities of its internal legacy (open or proprietary) systems to Web Services. These services can be as simple as scheduling appointments, validating credit cards, browsing product catalogues, or submitting invoices. They can also be as complex as the functions carried out by an entire supply chain, a customer relations management system, or ERP (enterprise resource planning) applications. These services hide their internal complexities, such as data types and business logics, from their users, but expose their APIs using the WSDL and their locations using UDDI protocols. Since every service complies with a single set of Web Services standards, there is no need for custom bridges to accommodate different computing platforms. Instead, organizations can exchange data by directly invoking data exchange services. For example, WebserviceX.NET provides Web Services in seven areas: business and commerce, communications, standards and lookup data, graphics and multimedia, and utilities. The top Web Services are: stock quote, currency converter, global weather, translation etc. According to their web site, over 2 million Web Services transactions were completed every day.

Web services technology is emerging as a promising infrastructure to support loosely coupled, Internet-based applications that are distributed, heterogeneous and dynamic. It has been envisioned as an important trend in application development and integration. Due to the fact that Web services use the XML-based SOAP protocol and do not depend on any operating system they are ideal for the integration of even completely inhomogeneous systems.  Thus, Web services technology provides a standards-based, process-centric framework for achieving the sharing of distributed heterogeneous applications. It allows pre-built applications/application components wrapped as web services to interact with each other through standardized interfaces and form larger application systems.

While Web Services technology provides a promising foundation for developing distributed applications for e-business, to realize the benefits of Web Services, many issues need to be addressed. For example, interactions among business organizations need to follow the policies, regulations, security and other business rules of the organizations: How to integrate business event and rule management with the Web Services? How to bridge the gap between the available Web Service infrastructure and current understanding of the Web Service business context? Since the business domains Web Service covered is wide and problems may be complex, how to utilize the Web Service effectively can be another issue. How to implement additional features are required to make this paradigm truly useful in the real world? Last, a major drawback of existing Web Service approaches is the missing security conventions.

To address these issues, many researchers are working on topics related to Web Services. Some of the topics include Web Services specifications and enhancements, Web Services discovery and integration, Web Services security, Web Services standards and formalizations, Web Services Modelling, Web Services-oriented Software Engineering, Web Services-oriented software testing, Web Services-based applications and solutions, Web Services realizations, and semantic Web Services.

## Grid computing

Grid computing is the next logical step in distributed networking. Just as an electric power grid delivers energy on demand, grid computing dynamically manages network, computing and storage resources to automatically support business processes across the IT infrastructure. Although it has been popular tool in academia for many years, it has recently become more popular in industries (Carlino, Gore, Venturini & Warner 2006). With grid computing, an organization can transform its distributed and difficult-to-manage systems into a large virtual computer that can be set loose on problems and processes too complex for a single computer to handle efficiently. Grid computing is a new IT architecture, but is rapidly growing in popularity. Forrester Research (Gillett 2004) reports that 37% of enterprises are piloting, rolling out or have implemented some form of grid computing. IDC calls grid computing the fifth generation of computing, after client-server and multi-tier. It is considered one of the key emerging technologies that will likely form the foundation of a new wave in IT.

Grid technology is evolving rapidly. Standards, frameworks, implementations, and applications are changing on a constant basis. Because it is an emerging technology, grid computing can mean different things to different people. To define the specifications for grid computing, the Global Grid Forum (GGF) was formed. From the service perspective, grid computing allows you to unite pools of servers, storage systems, and networks into a single large system so you can deliver the power of multiple-systems resources to a single user point for a specific purpose. To a user, datafile, or an application, the system appears to be a single enormous virtual computing system. The systems linked in a grid might be in the same room, or distributed around the world. They might be running different operating systems on many hardware platforms. They might even be owned by different organizations. Regardless of the depth of a grid's resources, all the grid user experiences are the processing resources of a very large virtual computer.

The main resources grid computing is designed to give access to include, but are not limited to:

- Computing/processing power.
- Data storage/networked file systems.
- Communications and bandwidth.
- Application software.

These information resources are shared based upon their availability, capability, and cost as well as the user's quality of service (QoS) requirements.

The advantages of grid computing include: 1) reduced total cost of ownership (TCO); 2) aggregated and improved efficiency of computing, data, and storage resources; 3) the creation of virtual organizations for applications and data sharing.

Currently major software developers and providers are all offering solutions to grid computing. IBM grid solutions can bring benefits to businesses in five key business and technology areas: business analytics, engineering and design, enterprise optimization, government development, and research and development. Oracle grid technology comprises the following products: 1) Oracle Database 10g and Real Application Clusters that coordinate the use of large numbers of servers and storage acting as one self-managing grid for the highest quality of service on low-cost, modular hardware; 2) Oracle Fusion Middleware enables enterprises to streamline and optimize business and IT operations, increase the accuracy and timeliness of business decisions, and secure information and drive compliance; 3) Enterprise Manager with Oracle Grid Control provides a single, integrated interface for administering and monitoring applications and systems in an Oracle grid. Similarly, the grid solution from Sun Microsystems is designed to help IT organizations architect and deploy appropriate grid computing infrastructure that contributes directly to their key business goals. The solution combines services, Reference Architectures, Sun Fire V60x Compute Grid, storage and software needed for setting up a grid environment faster, easier and at a lower cost than competitors. Microsoft has also outlined its vision of grid computing. Windows Compute Cluster Server 2003 is designed to provide a solution that makes computation easily accessible, consumable and fully integrated into the rest of the enterprise infrastructure as well as more affordable to mainstream users. Microsoft is also working with grid computing software vendors such as Platform. In the open source world, developers contributed their solution, called Globus Toolkit, which has become a de facto standard for grid computing and transformed the way on-line resources are shared across organizations.

Despite the significant benefits from grid computing, such as cost savings and increased performance, many technical and non-technical barriers hinder the adoption of grid computing. First, the current software licensing model does not fit into grid computing. At present, software developers and vendors charge customers according to the computer processor that runs the application – in other words, one application, one computer, one price. They have not been able, or willing, to relax this per-CPU software licensing model to allow grid to flourish (Wailgum 2005). Other barriers to the creation of a solid grid market include a lack of standards and the high cost of reconfiguring standard client/server applications to work on grid. Besides, there are non-technical obstacles too. GRIDtoday (2005) reported that internal politics create significant barriers to widespread Grid adoption. The top non-technical obstacles (ranked "high" or "very high") to Grid computing include:

- Perceived loss of control or access to resources (44%);
- Perceived risks associated with enterprise-wide deployment (40%);
- Perceived loss or reduction of budget dollars (33%).

Like other technology, it will take time for businesses to understand and adopt Grid computing. In order to gain internal support, people need to be educated within all levels of an organization – from the IT manager to the CEO – about the business benefits of Grid computing.

**Service-oriented architecture**

Automating, integrating, and maintaining business processes in a dynamic and cost-effective manner has proven to be a difficult endeavour for even the most technically sophisticated organizations. Service Oriented Architecture (SOA) is a new application development and integration methodology that addresses these issues effectively. Technically, SOA is based on XML and Web Services technologies and has been incorporated into Business Process Management and Enterprise Application Integration (BPM/EAI) platforms.

It is important to understand that Web-Services is not the same as SOA. Web-Services is a collection of technologies, including XML, SOAP, WSDL, and UDDI, which lets you build programming solutions for specific messaging and application integration problems. On the other hand, SOA is a software architecture in software engineering. The software architecture of a system consists of the large-grained structures of the software. It describes the components of the system and how those components interact at a high level. The interactions between components are called connectors. The configuration of components and connectors provide both a structural and a behavioural view of the system. It does not require Web-Services. Within a business environment, an SOA might be something like "an application architecture within which all functions are defined as independent services with well-defined invokable interfaces which can be called in defined sequences to form business processes" (IBM Web Services Architecture team). Within the SOA architecture, an application is no longer an opaque, procedural implementation mechanism. Instead, it is an orchestrated sequence of messaging, routing, processing, and transformation events capable of processing the exposed declarative properties of rich (XML) documents.

SOA provides several salient advantages. First, organizations can extend the SOA beyond the boundaries of the enterprise to link to suppliers and customers, and offer collaboration with outside systems via the same architecture. They can leverage SOA to establish a single connection to each vendor or customer organization rather than building point-to-point connections. If one organization were to change how a certain function is processed internally, as long as the published programming interface does not change, the rest of the system is not affected. Thus, the cost of entry and exit will be greatly reduced. Second, SOA simplifies operation and maintenance process. Traditionally, every new application or process change leads to hundreds of custom bridges, whereas SOA requires just one integration point for each. New systems and platforms can be plugged into a software bus to be immediately integrated with all the other systems on the same bus. SOA can also be used to integrate middleware islands and thus leverage existing EAI investments (Themistocleous & Chen 2004).

Experts from IBM summarize the benefits of deploying a SOA (Channabasavaiah et al. 2003) as follows:

- Leverage existing assets – A business service can be constructed as an aggregation of existing components, using a suitable SOA framework, and made available to the enterprise. Using this new service only requires knowing its interface and name. The service's internals are hidden from the outside world, as well as the complexities of the data flow through the components that make up the service. This component anonymity lets organizations leverage current investments, building services from a conglomeration of components built on different machines, running different operating systems, and developed in different programming languages. Legacy systems can be encapsulated and accessed via Web Service interfaces.

- Infrastructure, a commodity – Infrastructure development and deployment will become more consistent across all the different enterprise applications. Existing components, newly-developed components, and components purchased from vendors can be consolidated within a well-defined SOA framework. Such an aggregation of components will be deployed as services on the existing infrastructure, resulting in the underlying infrastructure beginning to be considered more as a commodity element.
- Faster time-to-market – Organizational Web Services libraries will become the core asset for organizations adapting the SOA framework. Building and deploying services with these Web Services libraries will reduce the time-to-market dramatically, as new initiatives reuse existing services and components, thus reducing design, development, testing, and deployment time.
- Reduced cost – As business demands evolve and new requirements are introduced, the cost of enhancing and creating new services by adapting the SOA framework and the services library, for both existing and new applications, is greatly reduced. The learning curve for the development team is reduced as well.
- Risk mitigation – Reusing existing components reduces the risk of introducing new failures into the process of enhancing or creating new business services.
- Continuous Business Process improvement – An SOA allows a clear representation of process flows identified by the order of the components used in a particular business service. This provides an ideal environment for monitoring business operations. Process modelling is reflected in the business service. Process manipulation is achieved by reorganizing the pieces in a pattern (components that constitute a business service).
- Process-centric architecture – The existing architecture models and practices tend to be program-centric. Applications are developed for the programmer's convenience. Often, process knowledge is spread between components. The application is much like a black box, with no granularity available outside it. Reuse requires copying code, incorporating shared libraries, or inheriting objects. In a process-centric architecture, the application is developed for the process. The process is decomposed into a series of steps, each representing a business service. In effect, each service or component functions as a sub-application. These sub-applications are chained together to create a process flow capable of satisfying the business need. This granularity lets processes leverage and reuse each sub-application throughout the organization.

According to Channabasavaiah et al. (2003), to realize the benefits of SOA, five management issues must be resolved: 1) security management that handles validation or authorization of the request, encryption, and decryption; 2) deployment management that allow services to be moved around the network for performance, availability, or other reasons; 3) logging management for auditing, metering, and tracking; 4) dynamic rerouting for fail over or load balancing; and 5) maintenance management.

The market for SOAs will dominate the distributed computing arena by 2010. The total SOA Implementation Framework market opportunity will go from $4.4 billion in 2005 to $43 billion by 2010 (ZapThink 2004). Also the big winners from the shift to service orientation will be large vendors who are able to leverage the innovation of the smaller players to build fully functional SOA Implementation Frameworks.

In addition to the ‘typical’ SOA mentioned above, there are also alternatives such as peer-to-peer (P2P) infrastructure. P2P systems are highly dynamic, allowing for ad hoc addition and removal of data sources. Instead of a fixed, global schema, P2P architectures couple the peers directly and map their local schemas via domain relations and coordination formulas

(Bernstein et al. 2002) or peer mappings (Halevy & Tatarinov 2004). An application directly sends a query to a single peer using its local schema, which may access other peers to answer the query. P2P opens a whole new range of business opportunities. The best-known instance of this today is Napster.

## Grid service

Standardization is the key to the realization of the vision of Grid computing. By standardization, the diverse components that make up a modern computing environment can be discovered, accessed, allocated, monitored, accounted for, billed for, etc. and, in general, managed as a single virtual system –even when provided by different vendors and/or operated by different organizations. To this end, based on advantages of Web Services technologies, the Grid community designed the Open Grid Services Architecture (OGSA), which offers an extensible set of services that virtual organizations can aggregate in various ways (Comito et al. 2005). With grid services, services can be deployed on the grid. The organization can move service to a new server in the event of system failure.

The fundamental goal of the OGSA is to provide a well-defined set of basic interfaces for the development of interoperable Grid systems and applications. In OGSA, every resource (e.g. computer, storage, and program) is represented by a service, i.e. a network-enabled entity that provides some capability through the exchange of messages. Web Services address only persistent service discovery and invocation, while OGSA supports transient service instances created and destroyed dynamically. Hence a Grid service is a potentially transient Web Service based on Grid protocols expressed by WSDL. Grid services maintain an internal state to distinguish one instance of a service from another that provides the same interface.

Enabling collaborative computing, grid services can serve as an effective market mechanism for distributing Web Services (Liu & Ma 2003). A service grid may act as an intermediary between service providers and service consumers and break a typical many-to-many business (between providers and consumers) into simple one-to-many relationships. It buys Web Services from the providers and then sells the services to consumers. Then consuming Web Services becomes as easy as watching TV programs from a cable network or obtaining electricity from a power grid; requesting and delivering Web Services becomes as easy as plugging an appliance into the grid. At the same time, those infinitesimal service providers do not have to incur prohibitive expenses to advertise and run their businesses. They can focus on their core business – developing and upgrading Web Services, and then plug the services into the grid to sell.

## Conclusion

In this paper we discussed ASP, Web Services, grid computing, OSA, and grid service. All these new technologies share the same vision: managing complexity, enabling common languages through open industry standards and improving your business through virtualization and automation of IT-based services. They are not exclusive to each other. Rather, they are complementary. First, ASP usually offers large, complete applications with little customization. They are used by common users. Web Services are small components designed to solve special business problems. Although they have published programming interfaces that have better interoperability, to integrate the service into applications requires advanced programming skills.

Which technology the business should adopt or purchase depends on many factors. Besides the technical factors discussed in the paper, there are also many non-technical factors. Generally, the most significant of such factors are: business leaders' support, budget, privacy and security. In E-commerce, security is important to build trust and compliance with law. However, the credit card companies have not come together to agree on a standard approach to security, and that could mean slower adoption of anti-fraud tools.

## Acknowledgements

The author would like to thank the Institute for Entrepreneurial Studies & Development at the University of Central Missouri for financial support.

## References

BIGresearch 2005, ‘More Consumers Researching Online Before Buying in the Store,’ *Market Wire*, COLUMBUS, OH -- 3/1/2005, http://www.bigresearch.com/news/big030105.htm

Bernstein, P., Giunchiglia, F., Kementsietsidis, A., Mylopoulos, J., Serafini, L., and Zaihrayeu, I. 2002, ‘Data management for peer-to-peer computing: A vision,’ In *Proceedings of the Workshop on the Web and Databases* (WebDB'02).

Business Wire 2007, ‘JupiterResearch Forecasts That Off-line Retail Sales Influenced by Online Research Will Reach One Trillion Dollars by 2011,’-- 1/17/2007. http://findarticles.com/p/articles/mi_m0EIN/is_2007_Jan_17/ai_n17117211

Carlino, U., Gore, B., Venturini, G., and Warner D. 2006, ‘Network Implications of Enterprise Grid computing,’ *Business Communications Review Magazine*, Sept. 18, 2006

Channabasavaiah, K., Holley, K., and Tuggle, E. Jr. 2003, ‘Migrating to a service-oriented architecture,’ *IBM.com*. http://www-128.ibm.com/developerworks/webservices/library/ws-migratesoa2/

Comito, C., Talia, D. and Trunfio, P. 2005, ‘Grid services: principles, implementations and use', *Int. J. Web and Grid Services*, Vol. 1, No. 1, pp.48-68.

Gillett, F. 2004 ‘Grids gets big, but the term is confusing,’ *Trends*, May 18, 2004.

GRIDToday 2005 ‘Study: Politics is Key Barrier to Grid Computing,’ *Daily News and Information for The Global Grid Community* / March 17, 2003: Vol. 2 No. 11 http://www.gridtoday.com/03/0317/101173.html

Halevy, A., and Tarrinov, I. 2004, ‘Efficient query reformulation in peer data management systems,’ in Proceedings of the 2004 ACM SIGMOD, *International Conference on Management of Data*, Paris, France. Pp: 539 - 550

IBM Web Services Architecture team, 2000 ‘Web Services architecture Overview - The next stage of evolution for e-business,’ *IBM.com*. Accessed at http://www.ibm.com/developerworks/library/w-ovr/

Monahan, M. T. 2007 ‘2006 Identity Fraud Survey Report-Identity Fraud is Dropping, Continued Vigilance Necessary.’ *Javelin Strategy and Research*.

Johnson, C. 2005, ‘The State of Retailing Online,’ *Forrester*, http://nrfweb.nrf.com/Attachments.asp?id=7007

Liu, L., and Ma, Q. 2003, ‘Emerging e-Business Technologies for Electronic Medical Records,’ *International Journal of Healthcare Technology and Management*, 3/4/5, pp.157-178.

Ma, Q. and Souren, P. 2003, ‘Web-based Consumer Decision Support System,’ *Proceedings of Americas Conference on Information Systems 2003*.Tampa, FL.

Maranville, R. and Brown, C. 2006, ‘Automating Business Processes in Stores and Across the Enterprise,’ *NRF Annual Convention 2005 Handouts*. http://www.360commerce.com/data/docs/FedEx Kinkos Automates Enterprisewide Business Processes.pdf

Miranda, C. 2005, ‘Buying and Selling Software as a Service: A View from the Trenches,’ A*spnews.com*. www.aspnews.com/strategies/article.php/3554361

Oracle, Oracle Grid Computing (an Oracle Business White Paper), February, 2005.

Services architecture Overview - The next stage of evolution for e-business”
Stansberry, M. Enterprise Grid Alliance addresses obstacles, SearchDataC enter. com. http://searchdatacenter.techtarget.com/cina/0,289202,sid80 gci1132040.00.html

Sullivan, I. 2005, ‘Retailers Buying Into IT,’ *Information week*--December 01, 2005

Susarla, A. Barua, A. and Whinston, A. 2003, ’Understanding the Service Component of Application Service Provision: An Empirical Analysis of Satisfaction with ASP Services,’ *MIS Quarterly*. 27, 1 (March 2003), pp91-123.

Themistocleous, M., and Chen, H. 2004, ‘Investigating the integration of SMEs' information systems: an exploratory case study,’ *International Journal of Information Technology and Management*, 3(2/3/4), pp. 208-234.

Wailgum, T. 2005, ‘The State of Retailing Online by Carrie Johnson principal analyst,’ *Forrester Research*.

Wendy L. C., Philip, J., and Graham, W. 2007, ‘Evaluating Application Service Provisioning Using System Dynamics Methodology,’ *British Journal of Management* 18 (2), pp172–191.

Yen, T. 2005, ‘Online Retail Report Card: Highest Conversion Rates Amongst Non-Brick-And-Mortar Retailers,’ *Nielsen/Netratings* Megaview Online Retail. Accessed at http://www.nielsen-netratings.com/pr/pr_050407.pdf

Zapthink 2004, ‘Service Orientation Market Trends Report Predicting the Future ofXML & Web Service.’