

Empirical Performance Metrics Study of Execution of Database Queries in Implementation of Web Services

¹Velmurugan, K. and ²M.A. Maluk Mohamed

¹Department of Computer Science and Engineering,
Anjalai Ammal Mahalingam Engineering College, Anna University, India

²Department of Computer Science and Engineering,
M.A.M College of Engineering, Anna University, India

Abstract: Problem statement: Web services are increasingly being deployed in business applications, due to its unique features such as flexibility, interoperability and other features. Most of the business applications involve extensive use of database operations for data management in back end. Further business applications demand very high level of performance from software solutions and it is a continual and never-ending process. This study focuses on measurement and analysis of performance metrics of database queries in implementation of web services. **Approach:** In this experimentation, web services are implemented in two popular and standard platforms and database queries are realized through all commercial and standard databases. Performance measurement is done by implementing a common sample application on each realization and using a pair of performance metrics, response time and packet count, that is, number of packets involved in communication between the layers of implementation. **Results and Conclusion:** This novel study summarizes the various performance aspects of databases in web services with a basic set of database queries in the back end and concludes with firm results on optimum performance offered by database in execution of database queries in realization of web services.

Key words: Performance metrics, web services, database support

INTRODUCTION

Nowadays web services are gaining tremendous growth due to its unique features such as flexibility, interoperability. And also web services facilitate for reliable distributed computing and grid computing applications. The web services technology, which is increasingly adopted by companies and research departments to achieve more efficient functions and higher profits, brings reusable components, highly aggregation, low coupling and platform independent to software architecture (Xie *et al.*, 2008). Further, Web services feature ubiquitous protocols, language-independence and standardized messaging (Pacifci *et al.*, 2005). Open Industry standards and simple implementation techniques makes web services usage widespread and grow at exponential rate. A Transaction is a unit of work composed of a set of operations on objects and transactions are fundamental to reliable distributed computing (Li *et al.*, 2004). Transaction based web services that demand real-time performance guarantees, such as on-line auctions, stock trading and real-time database servers generally require a higher level of

security and performance compared to normal services (Demir and Ghose, 2005). Concretely, Paypal's Web services, e-Bay Developer Program and Amazon web services are illustrative examples of web services being used in mission-critical, security-sensitive and truly large-scale applications (Wei *et al.*, 2008). These types of large-scale business applications deploy database in back end for maintenance of large volume of data. This leads to a lot of database queries being executed in the back end as part of realization of web services.

Presently business applications demand for two vital key non-functional requirements: Performance and Security. So, Performance and security enhancement at all stages of implementation of web services is inevitable and persistent efforts are necessary to achieve continual growth in performance and also in security. Further, Mission-critical applications based on web services require service agreements between consumers and providers of web services to succeed. Vital service factors include web service availability, performance and redundancy (Barclay *et al.*, 2006). In this study, an attempt is made to measure and analyze performance of execution of database queries in

Corresponding Author: Velmurugan, K., Department of Computer Science and Engineering,
Anjalai Ammal Mahalingam Engineering College, Anna University, Trichy, India

realization of web services, using some standard databases for queries and J2EE and .Net software platforms for web service application.

The motivation behind this study is that web services have become the backbone of distributed computing and grid computing applications. Moreover various standards and technologies involved in web services make web services interoperable and pervasive. In any application, the selection of particular technology is purely based on performance offered by the technology. The performance of web services, to a certain extent, also depends upon database used as back end in implementation of web services. Hence selection of database for web services depends upon performance criteria. In this study, standard set of queries are experimented with all the standard databases as back end of web services to analyze performance of databases in the context of web services. This study attempts to analyze the performance of web services with databases based on execution of some DML commands and performance metrics response time and packet count at the level of data link layer. In a nutshell, this study focuses on performance aspects of the databases by deploying them with J2EE and .Net, for realizing and consuming web services with database queries in back end.

Related works: Web services and databases have very high level of close association in implementing business applications using web services and so several works have been carried out in different aspects of web services and databases including performance analysis. The commit operation is the vital one in database hence few works (Li *et al.*, 2004; Younas *et al.*, 2006) concentrated on commit protocol. In the context of Quality of Support (QoS) for web service transaction, QoS based commit protocol was presented to enable the QoS integration in web service transactions (Li *et al.*, 2004). The study (Younas *et al.*, 2006) investigates performance of transaction commit protocols in the context of composite web services and introduces new commit protocol to improve performance of composite web services transaction. Some research efforts (Gao *et al.*, 2005; Paul *et al.*, 2007) were made to enhance performance based on database transactions and ACID property of transactions. The experimental results from (Gao *et al.*, 2005) show that by slightly relaxing consistency within individual distributed objects, the application realizes both high availability and excellent performance. The study (Paul *et al.*, 2007) concentrated on maintaining an acceptable level of service in web services without compromising traditional ACID properties for transactions.

Several new attempts were made to measure and analyze performance of web services and databases.

The study (Gounaris *et al.*, 2008) propose a robust control theoretical solution to the problem of optimizing the data transfer in queries over WSS, by continuously tuning at runtime the block size and tracking the optimum point. As part of reverse engineering, web services were deployed to allow remote access and management of legacy databases apart from discovery of functionality (Guzman *et al.*, 2006).

The study (Xie *et al.*, 2008) proposes a configurable web service performance testing framework which contains client module, application module and database table and allows testers to configure transaction characteristics to confirm performance metrics. A middleware infrastructure was presented in (Zarras *et al.*, 2005) for the development of virtual databases in pervasive computing, by replacing traditional treatment of databases. Two new web service architectures were proposed for distributed search (Ivanova, 2004). Pacifici *et al.* (2005) present an architecture and prototype implementation of a performance management system for cluster-based web services.

A new approach (Jiang *et al.*, 2005) is presented for creating XML views from databases and so applications over web services may exchange data without knowing database structure. Wang *et al.* (2007) proposes an autonomic, workload-aware approach to generate relational scheme for XML files according to dynamic application characteristics or user-specified workload. Some other works concentrated on experiments on applying web services on other areas or incorporating new concepts in web services. In study (Zhang *et al.*, 2007) experiments were carried out by deploying several geospatial web services on top of popular spatial databases and Geographical Information Systems (GIS) to study performance of geospatial web services. A database management system was developed to provide users with on-line web services to build and manage their database to perform parallel data mining on internet and performance study was done based on toolkits (Liang *et al.*, 2006). Datla and Goseva-Popstojanova (2005) focus on measurement-based performance analysis of an e-commerce application which uses web-service components to execute business operations.

Table 1: Deployment environment

System	Hardware	Software
Web service server,database server, Client	Intel Core 2 Duo Processor E6550 2.33 GHz, 4 MB L2 Cache, 1.33 Mhz FSB, 1 GB RAM	Windows XP SP3., Windows 2003 IIS 6.0 MS visual studio 2005 Net beans 6.0 Glass Fish v2

Table 2: Database driver details

Modules/Database	Access	Oracle	SQL Server	DB2	MySQL
.NET	OleDb	Oracle client	SQL client	IBM.DB2.DB2	MySqlClient
JAVA	JDBC-ODBC	Ojdbc	JDBC-SQL	JDBC-DB2	JDBC-MySQL

Ma and Bartos (2006) addresses deployment scenarios for optimizing web-based transactions and future optimization of web transactions. The study (Tran and Tsuji, 2008) presents ranking algorithm for web services based on QoS parameters using fuzzy logic. The previous study of authors (Velmurugan and Mohamed, 2009) experimentally measure and analyze performance of software platforms for realizing web services and it provides base for this study. The study (Gupta *et al.*, 2007) uses semantic web concepts in web services matchmaking and presents performance analysis and quantifies gains due to knowledge acquisition algorithm. The study (Liu *et al.*, 2007) presents a SLA-based web services guarantee model to improve web services runtime environment, based on the idea of differentiating web services. Stantchev and Malek (2008) evaluates replication possibilities for web services at the operating system level and how they affect web service performance.

This study is novel in the aspect of analyzing performance of execution of database queries in the context of web services, when compared to the works discussed above. The metrics used to analyze performance also unique in the aspect how they are applied, that is, metrics data are collected on executing a basic set of DML commands in web service application.

Experimentation:

Experimental setup: This experimentation is carried out by developing an inventory control application for maintaining stock. The business logic is developed using J2EE and .net based web services. Clients access the web services through a web site and in turn web services uses different databases in back end. The application is developed with same features in all aspects in both the technologies in order to compare the performance. The same type of functionalities are supported, database with same fields and same queries are used for experimentation. Performance measurement is done on web services with DML commands using the metrics response time and packet count. The reason for using these two metrics is response time is the key performance factor involved in any real-time application and packet count reveals size of communication overhead that takes place between the layers of implementation. The size of communication overhead plays a vital role in internet due to the fact that network traffic and cost of communication is directly proportional to the size.

Deployment environment: Deployment environment consists of two machines, one for web services provider and other for database server. Clients access the web service in turn web service access database for data access. These systems are connected by means of 100 MB Ethernet LAN. The configurations of the systems stated above are listed in Table 1. Wire shark, network protocol analyzer, is used for analyzing performance based on network traffic involved in communication between different layers of implementation for materializing web services with database in back end.

Databases deployed for experimentation: All the standard and commercial databases were used for experimentation and list is given below.

Microsoft Access 2003, Microsoft SQL Server 2005, My SQL 5.1, IBM DB2 v9.5 Oracle 9i

The database drivers deployed for the above databases are provided in Table 2.

Metrics used for performance measurement: The performance is measured and analyzed through the following metrics:

- Response Time (RT): time spent between requests originated from client, reaches the web service server, web service access database for data and response being delivered to Client
- Packet Count (PC): Number of packets involved in communication between the layers of implementation

Results of experimentation: The values mentioned here are arrived by calculating statistical average from fifteen trials so that it reflects closely correct value. To find the overall performance of database, statistical average is calculated for all the average values of operations of each database, because it is tedious to arrive at conclusion just by looking at one operation alone and it is not the correct method to judge performance of database by considering only one operation. Moreover, considering a complete database as a whole, it should cover all basic DML commands and not just one. Figure 1-16 illustrates the metrics values obtained for different DML operations. All the time values are in milliseconds.

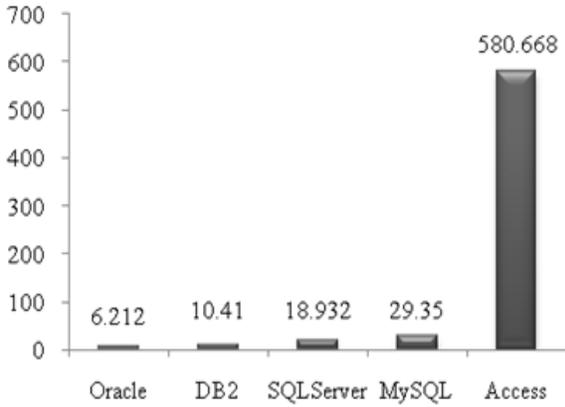


Fig. 1: Response time for Insert Operation

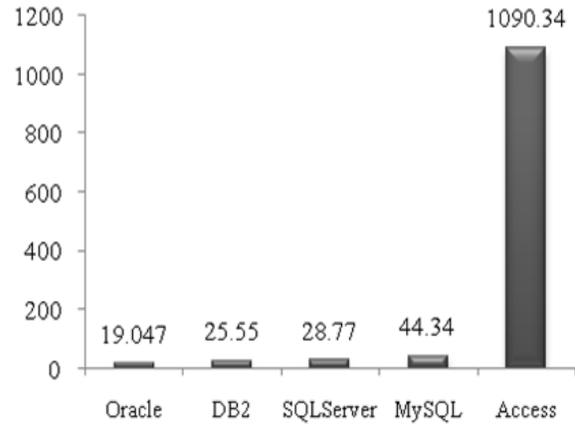


Fig. 4: Response time for update operation

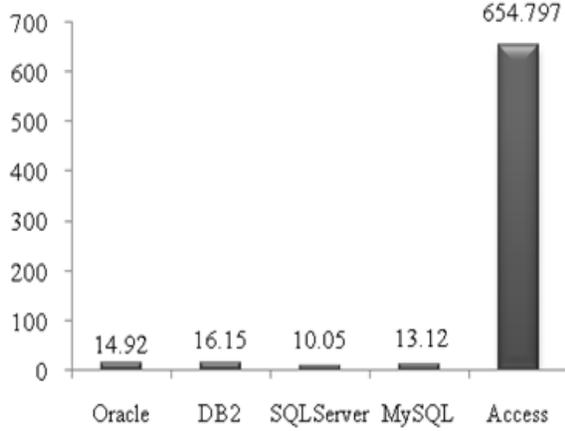


Fig. 2: Response time for select with parameter operation

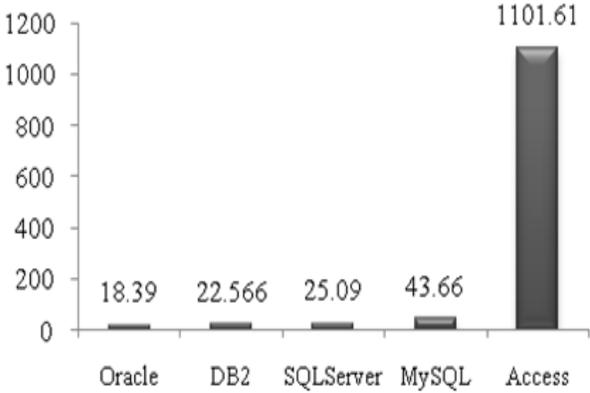


Fig. 5: Response time for delete with parameter operation

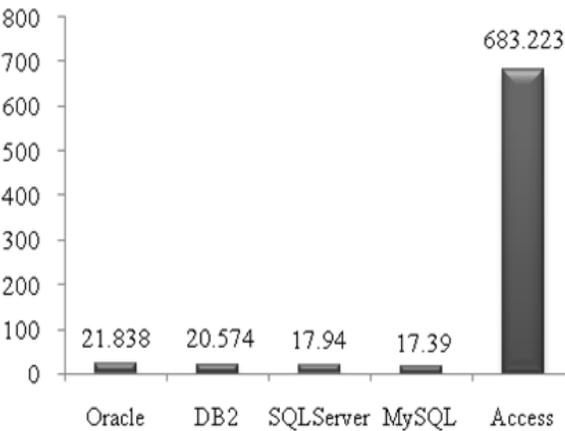


Fig. 3: Response time for Select without Parameter Operation

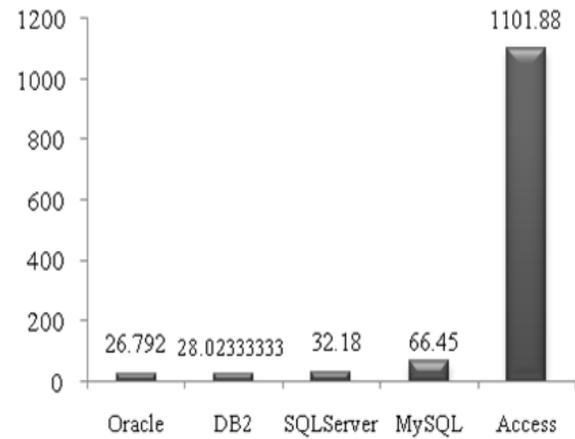


Fig. 6: Response time for delete without parameter operation

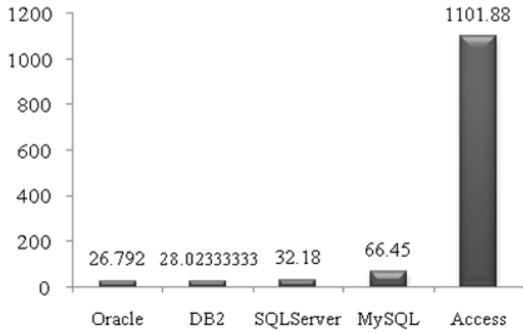


Fig. 7: Total average response time for .net

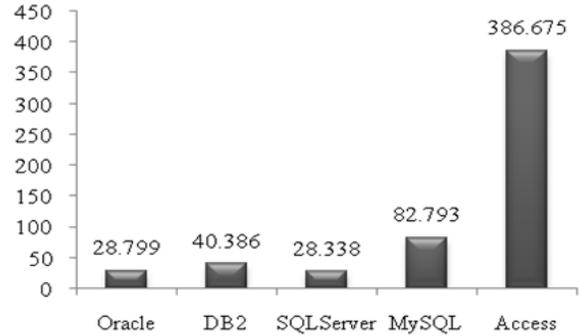


Fig. 11: Response time for update operation

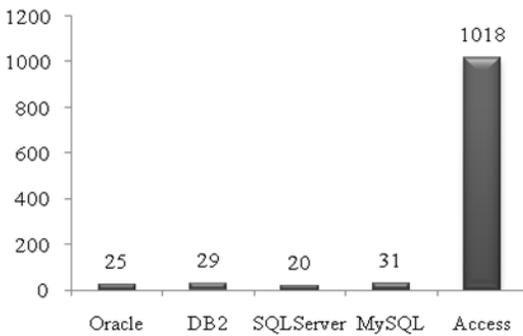


Fig. 8: Average packet count for .NET module

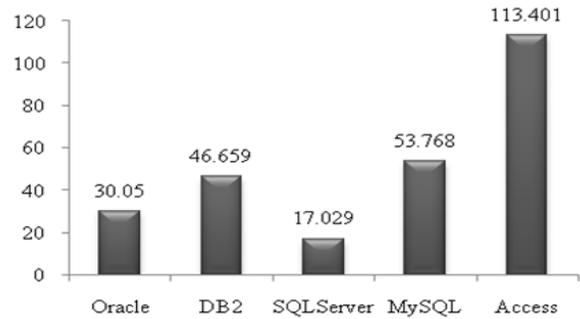


Fig. 12: Response time for select without parameter operation

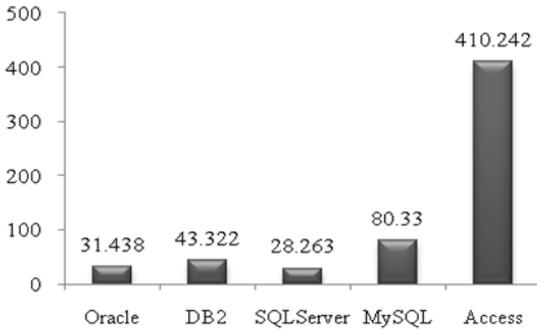


Fig. 9: Response time for insert operation

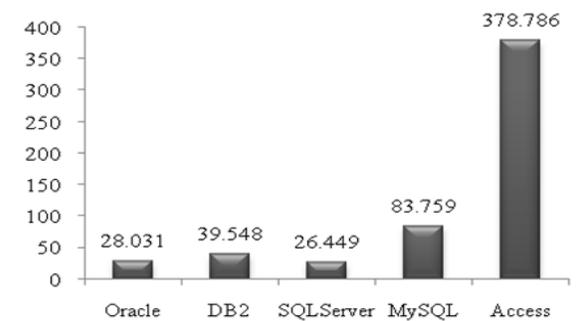


Fig. 13: Response time for delete with parameter operation

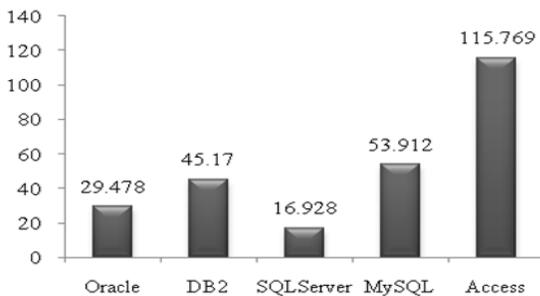


Fig. 10: Response time for Select with Parameter Operation

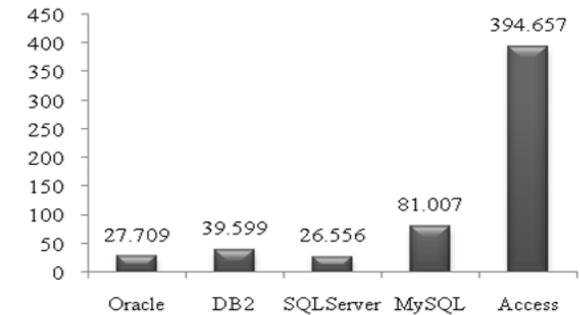


Fig. 14: Response time for delete without parameter operation

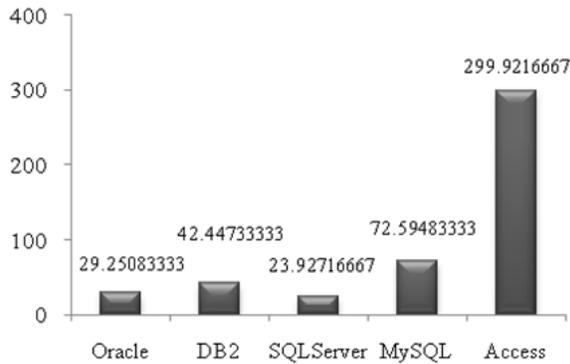


Fig. 15: Average response time for Java

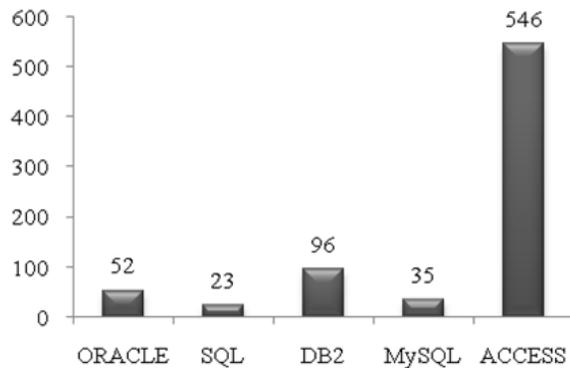


Fig. 16: Average packet count for java module

CONCLUSION

Conclusions and future study: The performance measurement is viewed in two different dimensions that is response time and packet count. The primary aim of any real time system with web services, is minimum response time and then with minimum packet count. Hence, in total, the experimentations revealed that in case of .Net module, oracle provides better response time and SQL server provides response with lesser packet count which is very slightly higher than oracle. In case of J2EE module, SQL server provides optimum performance with lesser response time and also with minimum packet count for communication compared to other realizations.

This study may be extended in several ways and some of them are described below. Cross-platform approach may be applied in the same experiment discussed above. The same setup may be evaluated by more metrics and new QoS parameters may be added to further evaluate the system. Database performance tuning methods may be incorporated in each realization to enhance performance of web services. Different database drivers may be used to establish connection

with database to evaluate performance in depth. This experiment is done based on a basic set of DML commands but more complex features of database may be taken into account for further analysis.

ACKNOWLEDGEMENT

The researchers express thanks to Mr. P. Balasubramanian, Mr. R. Chinnakannan, Mr. T.M. Nandhakumar, Mr. R. Raghunath for their invaluable assistance in this study.

REFERENCES

- Barclay, T., J. Gray, S. Ekblad, E. Strand and J. Richter, 2006. Designing and building terraservice. J. Mag. 10: 16-25. DOI: 10.1109/MIC.2006.95
- Datla, V. and K. Goseva-Popstojanova, 2005. Measurement-based performance analysis of e-commerce applications with Web services components. Proceedings of the IEEE International Conference on e-Business Engineering, Oct. 12-18, IEEE Xplore Press, Beijing, pp: 305-314. DOI: 10.1109/ICEBE.2005.85
- Demir, O. and K. Ghose, 2005. Improving transaction server performance under heavy loads with differentiated service and active network interfaces. Proceedings of the 4th IEEE International Symposium on Network Computing and Applications, Jul. 27-29, IEEE Xplore Press, Cambridge, MA., pp: 251-254. DOI: 10.1109/NCA.2005.26
- Gao, L., M. Dahlin, A. Nayate, J. Zheng and A. Lyengar, 2005. Improving availability and performance with application-specific data replication. IEEE Trans. Knowl. Data Eng., 17: 106-120. DOI: 10.1109/TKDE.2005.10
- Gounaris, A., C. Yfoulis, R. Sakellariou and M.D. Dikaiakos, 2008. Robust runtime optimization of data transfer in queries over web services. Proceedings of the IEEE 24th International Conference on Data Engineering, Apr. 7-12, IEEE Xplore Press, Cancun, pp: 596-605. DOI: 10.1109/ICDE.2008.4497468
- Gupta, C., B. Rajdeep, R. Michael, G. Madhusudhan and M. Weiyi, 2007. Improving performance of web services query matchmaking with automated knowledge acquisition. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Nov. 2-5, IEEE Xplore Press, Fremont, CA., pp: 559-563. DOI: 10.1109/WI.2007.101

- Guzman, I.G.R.D., M. Polo and M. Piattini, 2006. Obtaining web services from relational databases. Proceedings of the Conference on Software Maintenance and Reengineering, Mar. 22-24, IEEE Xplore Press, Bari, pp: 310-314. DOI: 10.1109/CSMR.2006.38
- Ivanova, E., 2004. Web service architectures for distributed search in databases. Proceedings of the 2nd International IEEE Conference Intelligent Systems, Jun. 22-24, IEEE Xplore Press, pp: 572-575. DOI: 10.1109/IS.2004.1344814
- Jiang, H., A. Lo, T. Ozyer and R. Alhajj, 2005. XML views based approach for Web services. Proceedings of the IEEE International Conference on Information Reuse and Integration, IEEE Xplore Press, pp: 458-463. DOI: 10.1109/IRI-05.2005.1506516
- Li, J., W. Tang and X. Wang, 2004. Adding QoS to web service transaction management. Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, (CCIS' 04), IEEE Xplore Press, Singapore, pp: 1308-1313. DOI: 10.1109/ICIS.2004.1460781
- Liang, T.Y., S.Y. Wang and L.I. Chang, 2006. A web-based database management system supporting parallel data mining service on PC clusters. Proceedings of the 1st International Conference on Innovative Computing, Information and Control, Aug. 30-Sept. 1, IEEE Xplore Press, Beijing, pp: 507-510. DOI: 10.1109/ICICIC.2006.224
- Liu, C., Z. Li, J. Chen, X. Lin and D. Ma, 2007. A service level agreement-based web services performance. Proceedings of the 16th International Conference on Computer Communications and Networks, Aug. 13-16, IEEE Xplore Press, Honolulu, HI, pp: 1279-1284. DOI: 10.1109/ICCCN.2007.4317997
- Ma, K.J. and R. Bartos, 2006. Analysis of transport optimization techniques. Proceedings of the International Conference on Web Services, Sept. 18-22, IEEE Xplore Press, Chicago, IL., pp: 611-620. DOI: 10.1109/ICWS.2006.30
- Pacifici, G., M. Spreitzer, A.N. Tantawi and A. Youssef, 2005. Performance management for cluster-based web services. IEEE J. Selected Areas Commun., 23: 2333-2343. DOI: 10.1109/JSAC.2005.857208
- Paul, D., F. Henskens and M. Hannaford, 2007. Isolation and web services transactions. Proceedings of the 8th International Conference on Parallel and Distributed Computing, Applications and Technologies, Dec. 3-6, IEEE Xplore Press, Adelaide, SA., pp: 181-182. DOI: 10.1109/PDCAT.2007.29
- Stantchev, V. and M. Malek, 2008. Addressing web service performance by replication at the operating system level. Proceedings of the 3rd International Conference on Internet and Web Applications and Services, Jun. 8-13, IEEE Xplore Press, Athens, pp: 696-701. DOI: 10.1109/ICIW.2008.113
- Tran, V.X. and H. Tsuji, 2008. QoS based ranking for web services: Fuzzy approaches. Proceedings of the 4th International Conference on Next Generation Web Services Practices, Oct. 20-22, IEEE Xplore Press, Seoul, pp: 77-82. DOI: 10.1109/NWeSP.2008.41
- Velmurugan, K. and M.A.M. Mohamed, 2009. An empirical performance metrics measurement and analysis of software platforms for implementation of web services. Proceedings of the 8th IEEE/ACIS International Conference on Information Science, Jun. 1-3, IEEE Xplore Press, Shanghai, pp: 914-920. DOI: 10.1109/ICIS.2009.194
- Wang, X., A. Zhou and G. Liu, 2007. Efficient service data management in dynamic environment. Proceedings of the IEEE International Conference on Web services, Jul. 9-13, IEEE Xplore Press, Salt Lake City, UT., pp: 1040-1047. DOI: 10.1109/ICWS.2007.82
- Wei, J., L. Singaravelu and C. Pu, 2008. A secure information flow architecture for web service platforms. IEEE Trans. Services Comput., 1: 75-86. DOI: 10.1109/TSC.2008.10
- Xie, J., X. Ye, B. Li and F. Xie, 2008. A configurable web service performance testing framework. Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications, Sept. 25-27, IEEE Xplore Press, Dalian, pp: 312-319. DOI: 10.1109/HPCC.2008.53
- Younas, M., K.M. Chao, C.C. Lo and Y. Li, 2006. An efficient transaction commit protocol for composite web services. Proceedings of the 20th International Conference on Advanced Information Networking and Applications, Apr. 18-20, IEEE Xplore Press, Vienna, pp: 591-596. DOI: 10.1109/AINA.2006.84
- Zarras, A., P. Vassiliadis and E. Pitoura, 2005. Query management over ad-hoc communities of Web services. Proceedings of the International Conference on Pervasive Services, Jul. 11-14, IEEE Xplore Press, pp: 261-270. DOI: 10.1109/PERSER.2005.1506419
- Zhang, J., D.D. Pennington and W.K. Michener, 2007. Performance evaluations of geospatial web services composition and invocation. Proceedings of the IEEE International Conference Web Services, Jul. 9-13, IEEE Xplore Press, Salt Lake City, UT., pp: 1128-1135. DOI: 10.1109/ICWS.2007.201