

Scalable Push Pull architecture for retrieving student information request and response with data wrappers

J.Saravanes¹, Dr.E.Ramaraj²,

¹ Assistant Professor, Department Of Computer Science,

MKU College, Madurai, Tamil Nadu, India

Saravanes¹@hotmail.com

² Professor, Department of Computer Science,

Alagappa University, Karaikudi Tamil Nadu, India

eramaraj@rediffmail.com

Abstract:

The concept of web services deals with request and response services for storing and retrieving data. As the data focus on dynamism, the strategy becomes highly impeccable to deal with the growing data volumes. The data service also retrieves the unwanted data that come along with the request. The strategy should satisfy the demand and balance the volume and provide exact result using the pull push architecture. The proposed work address the same that produce feasible result for student information gathering based on the information request by them and the response made in return. The work also address the data wrapper set that encapsulate specific information target based on the request.

Keyword: Pull-Push mechanism, Architecture, Data wrappers, Scalable web service.

1. Introduction

Web is a dynamic medium where the information's are scattered and widely available. It's very hard to balance the request and response as the data demand is not certain and limited. There are lags in the data delivery as the volume reaches the peak while making the request. Even after the lag the data received may not be certain and there may data insufficient or irrelevant in connection with the request made. The web service involved in this activity also maintains a decent search hits and miss. Hits leads to the data received and miss leads to failure. Even in hits there are irrelevant data which are also in the form of duplicate data that makes the job tougher. The concept of trace file involved in this concept that keeps the search intact to provide better result. The trace file works based on HTTP request retrieved from the search URL that are keyed in the address bar. The trace file also keeps an eye on the

traffic volume and the boundary where the data where sets. In order to set limit on data volumes prefetching can be done and such case will work when the data volume is high and hectic. The smooth execution of pull and push relies on the data distribution that provides the data for web services Mikko Pohja (2010). The following are the areas addressed in this work.

Pull mechanism in Data wrapper mode

This mode will activate when there is a need of resource access from the server. While accessing with this mechanism the server loads the exact data and that will be checked for consistency before doing it so. The throughput is evaluated and the exact data is identified. This initiative will reduce the inconsistent data that will be accessed while accessing the exact data. The concept of data wrapper also plays a vital role when the resources are accessed within the specified boundary. Generally the pull mechanism does resource request process. Aspect of dealing with data wrapper sets the request mode for consistent and reliable access of resource from the server.



Figure 1: Pull with data wrapper mode

Push mechanism in Data wrapper mode

This mode does response data processing from the server. This is an initiative for Pull mechanism which stores the data into the server. The data wrapper mode allocates the boundary and the data are loaded into it. While loading the

data, the reliability and consistency of the data sets were also checked and in parallel executes the process. Push mechanism stores the data into the server and push with data wrapper mode does it with consistency and reliable check. Hence the data that are to be stored and retrieved will involves dual checking process (i.e) both with pull and push mechanism.



Figure 2: Push with data wrapper mode

Scalable performance

The scalability is measured based on utility score that are achieved in respect with pull and push execution process. The utility score is the number of resources accessed within the specified time either using pull or push and the actual data sets retrieved. Hence the consistent performance of pull and push with reliable score gives positive utility score that was addressed in our earlier work. The time factors is calculated by the following

Reliable push query – Exact result achieved within specified time ensuring the time span never exceeds

Reliable pull query - Exact result achieved within specified time ensuring the time span never exceeds

Consistency score – Purely based on time factor as the time factor exceeds it is assumed that the reliable data can never be reached.

Reliable web services with Data wrapper

The reliability score depends on consistency score that can be achieved using Pull and Push by John Hagel, John Seely Brown (2008).

2. Review literature

In order to justify the proposed work the following are the areas referred as literature survey. The reliable resources can be resolved by the work of P. Mitra et.al (2002) that addresses the feature and non-feature selection of data in large data sets.

It’s our belief the data dynamism is the major pit fall that affects the reliability score and also the reliability is affected by latency measures addressed by G. Ruß et.al (2007). The reliability is also measured using replicated services were the data sets are duplicated by the web service. Such task will shrink the usage of web service and the services used before will be regressive and avoiding such factor will increase the reliable score H. Yu (2000).

The next aspect in this work relies on web server configuration based on the model used in this work. The work of Lars (2010) suggested the method for configuring web server for push based method and in this work the work is extended for both pull and push method.

The consistency check of the web services was addressed by Brian D. et. al (2000) that focus on web server response for push method. Combining the work of Lars and Brain the extension of this work uses the architecture for both push and pull mechanism.

The huge volume of data sets using wrapper sets feature was addressed in the work of Jinsong Leng et. al (2010) were the concept of data wrapper was insisted when the data sets volume is enormous.

The proposed works combines the work of data set wrapper using push and pull mechanism for evaluating consistency check which in turn impact on reliability.

3. Data set Wrapper class Architecture for Push and Pull

The architecture proposed in this work was shown in fig 3. The architecture defines certain roles for pull and push mechanisms V. Trecordi and G. Verticale (2000) for evaluating the consistency check and enforce reliability measures. For every transactions evaluated by the web services Thanisa Numnonda and Rattakorn Poonsuph (2012), both the push and pull mechanism are involved in consistency basis. The mechanism is entirely based on target data sets that are considered to be enormous. The time factor is set as unique for both the push and pull mechanism and the reliability is checked by this process. The time factor will not be deliberated when there is no definite data set volumes are defined. The utilization factors are also been bounded with these factors as it paves the way for consistency and reliability check.

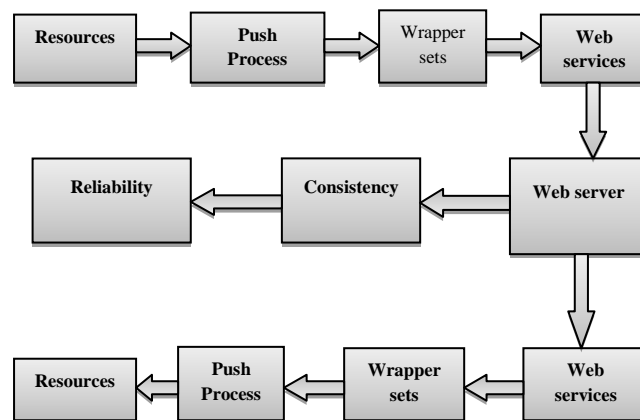


Figure 3: Web service architecture using Push-Pull with Data wrappers

In the fig 3 combining both push and pull in the web services for retrieving and storing data in the web server that are distributed makes the reliable and consistent measure based on the data looked upon in the web server. The proposed web server is distributed and justifies huge volume of data sets and despite being enormous the reliable and consistency score is achieved. The data set may be any kind either it may be health related data, student data base, web log information, web server accessed data and so on. In order to access the data the service has to be undergoing one complete cycle involving pull and push mechanism.

The reliable measure of the web services that uses pull and push mechanism can be calculated using

$$Pu_{req} = S(t) \tag{1}$$

Where Pu_{req} is the push factor and $S(t)$ is the server time frame. This was used in our earlier work for estimating push frequency in the server

$$Pu_{res} = S(t) \tag{2}$$

Pu_{res} is the response time of pull mechanism from the server.

The proposed work modifies the equation like

$$c(Pu_{req}) = Pu_{req} + DS_{req} \quad (3)$$

And DS_{req} is the DS data set for pull mechanism.

$$c(Pu_{res}) = Pu_{res} + DS_{res} \quad (4)$$

And DS_{res} is the DS data set for push mechanism.

The $c(Pu_{req}) + c(Pu_{res})$ gives the consistency score and the reliability is measure using

$$R(i) = c(Pu_{req}) + c(Pu_{res}) \quad (5)$$

4. Proposed work

The proposed work uses the student data base as the data set and that have been accessed from a live web server using services. Both the push and pull score is monitored based on the data entered, search made and result achieved.

The results are focused such that the student information's are feed into the system and the accessing of such information were done using request and response mode. In order to achieve feasible result the data set volumes is increased and its' subjected to

5. Evaluation and result

The figure 4 shows the plot analysis that shows the reply received towards the response made by the students. Figure 5 shows the representation of hits and failure of the request based on server access time. Failure increase when the server time is maximum to provide the result and for hit it's vice versa.

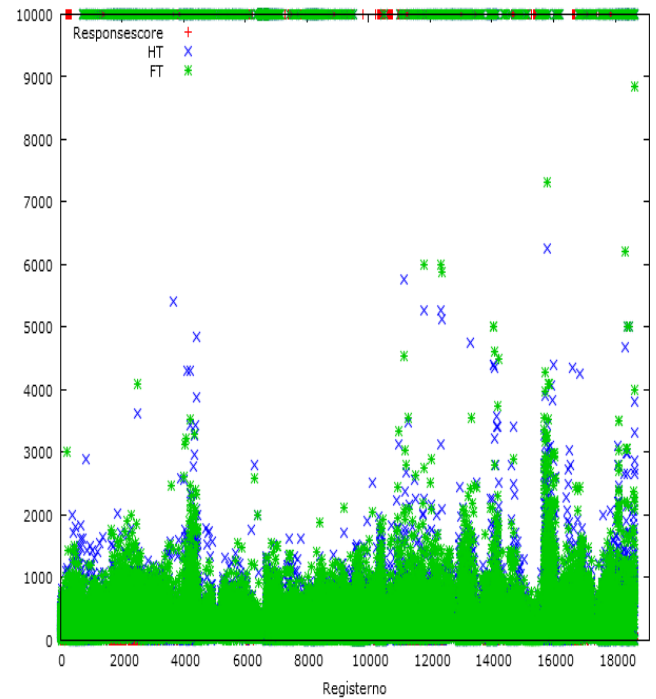


Figure 5: Hit and failure measure between request and response

Student based observations using consistent

Dependent variable: Registerno

	coefficient	std. error	t-ratio	p-value
Const	9713.36	44.5009	218.3	0.0000 ***
Response score	0.00892102	0.0166693	0.5352	0.5925
Consistency	-0.175726	0.0112685	-15.59	1.76e-054 ***
Mean dependent var	9321.000	S.D. dependent var		5381.338
Sum squared resid	5.30e+11	S.E. of regression		5332.836
R-squared	0.018050	Adjusted R-squared		0.017945
F(2, 18638)	171.3005	P-value(F)		1.91e-74
Log-likelihood	-186419.3	Akaike criterion		372844.5
Schwarz criterion	372868.0	Hannan-Quinn		372852.2

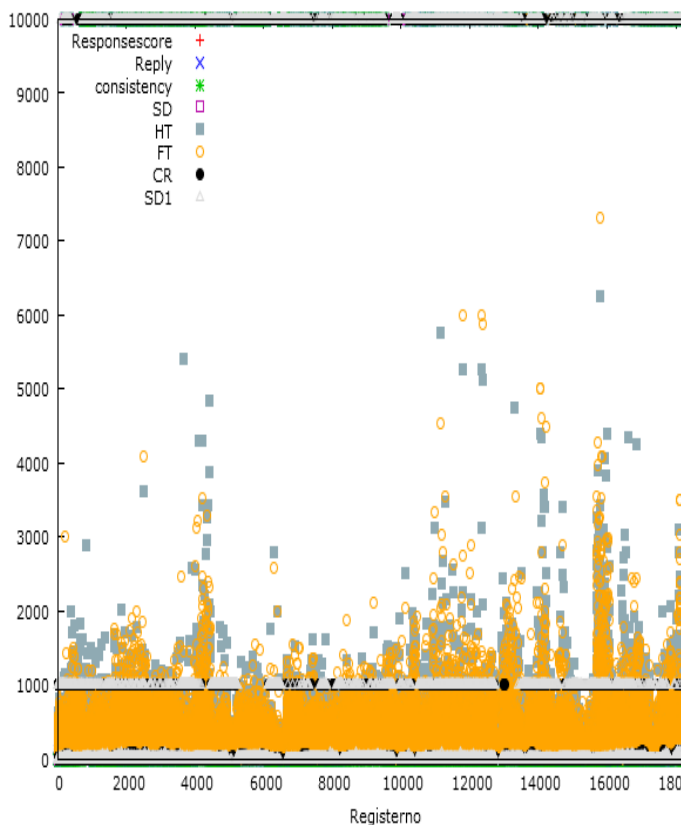


Figure 4: Plot analysis between student reply and response

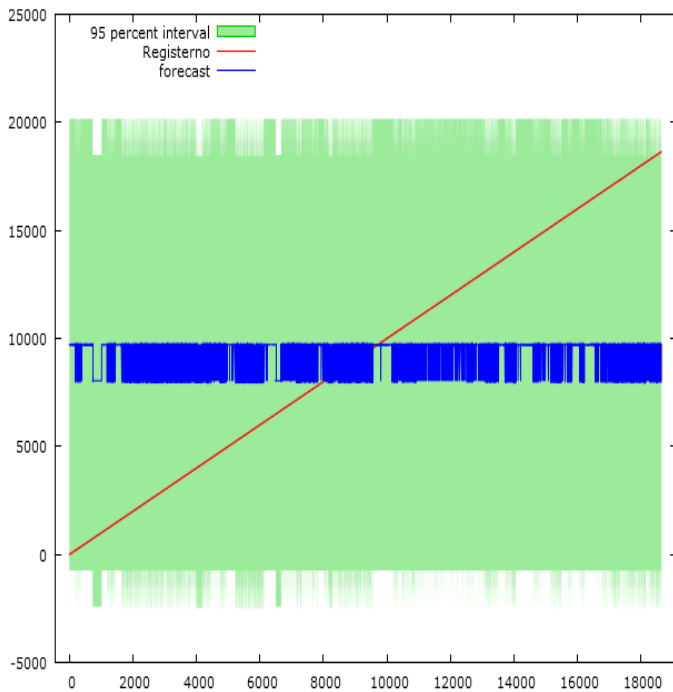


Figure 6. Forecast analysis for request and response

6. Conclusion and future work

The proposed work identifies the key areas for providing feasible results towards request and reply based on student information stored in the web service. The forecasting results shown in Figure 6 is the result is based on standard deviation score. The efforts have taken to maximize the hit rate and to minimize the failure rate.

The result also justifies the fact of achieving feasible result with enormous data set using data wrapper set. The extension of the work will focus on web orchestration for gathering information instead of using data wrappers.

Reference

- [1] P.Mitra, C.Murthy, and S. Pal. Unsupervised Feature Selection Using Feature Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:301–312, 2002.
- [2] G. Ruß and R. Kruse. Feature Selection for Wheat Yield Prediction. In *Research and Development in Intelligent Systems XXVI, Incorporating Applications and Innovations in Intelligent Systems XVII*, pages 465–478, 2010
- [3] H. Yu and A. Vahdat, Design and Evaluation of a Continuous Consistency Model for Replicated Services., *Procs. of OSDI*, October 2000.
- [4] Lars Brenna "Configuring Push-Based Web Services", Dept. of Computer Science University of Tromso, Norway, 2010
- [5] Brian D. Davison and Vincenzo Liberatore "Pushing Politely: Improving Web Responsiveness One Packet at a Time", *citeseerx.ist.psu.edu*, 2000

[6] Jinsong Leng, Craig Valli, Leisa Armstrong, "A Wrapper-based Feature Selection for Analysis of Large Data Sets", 2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010)

[7] Mikko Pohja , "Server Push for Web Applications via Instant Messaging", *Journal of Web Engineering*, Vol. 9, No. 3 (2010) 227–242

[8] V. Trecordi and G. Verticale. An architecture for effective push/pull web surfing. In 2000 IEEE International Conference on Communications, volume 2, pages 1159–1163, 2000.

[9] John Hagel, John Seely Brown, "From Push To Pull: Emerging Models For Mobilizing Resources" *Journal of Service Science – Third Quarter 2008 Volume 1, Number 1*

[10] Thanisa Numnonda, Rattakorn Poonsuph, "Inversion of Web Service Invocation using Publish/Subscribe", *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 1, No 3, January 2012



J.Saravanesh is working as a Assistant Professor, Department of Computer Science, Madurai Kamaraj University College, Madurai, Tamilnadu. He has acquired MCA, M, Phil in Computer Science and also holds P.G. Diploma in Information and Communication Laws and P.G.Diploma in Public Relations. He has over 6 years of experience and has published articles in international journals and has Presented Papers in National Conferences. His Area of interest is Data Engineering and his current work focus is Access Control Methods and their implications in secure data sharing environments .



Dr. E.Ramaraj is working as a Professor, Department of computer Science and Engineering, Alagappa University, Karaikudi. He has also served as the Director, Computer Centre, Alagappa University, Karaikudi, Tamil Nadu, India. He was also deputed as a Technical Advisor in Madurai Kamaraj University, Madurai, Tamilnadu for three years. He has acquired M.Sc (Mathematics). M.C.A., M.Phil. and PhD in Computer Science. He has more than 25 years teaching experience at Post Graduate Level and 15 years research experience in the field of Computer Science. His

contribution in the area of Data Mining, Speech recognition system and Software Engineering has been acknowledged across the e-

world. He has Guided more than 15 Research Scholars to attain PhD