Controlling of Web Services Transactions Using Transaction Scheduling Algorithm

Cherry Chit, Latt Latt Htwe

University of Computer Studies, Yangon, Myanmar

chit.cherrychit@gmail.com, llhtwe@gmail.com

Abstract

Web services based business processes are generally composed of invocation to internal business processes outsourced by loosely coupled providers [4]. A key requirement for the success of web service-based integration of business application is to ensure a correct and reliable execution of the composed process with regards to partner's transactional requirements. The system is implemented the web service transaction using transaction scheduling algorithm. Its aim is to reach an agreement between the client and all participating providers on what transaction processing times have to be expected, accepted, and guaranteed. This enables service consumers to find a set of best suited providers fitting their deadlines [1]. The system implements to control web service transaction that applied on online job seeking system.

1. Introduction

Web service-based business processes are generally composed of invocations to internal business processes outsourced by loosely coupled providers. Web Services will transform the web from a collection of information into a distributed computational device. The standard interfaces of web service technology do not fully support transaction management. Web service-based transactions are valuable and assigned deadline first, the transaction might be committed in a successful termination before this deadlines. So web service providers have to control their local resources and ensure their consistency and integrity [2]. In conventional scheduler algorithm, time constraints for committing the whole transaction or the individual services' constraints are ignored when scheduling decisions are made. Web services perform functions, which can be anything from simple requests to complicated business processes. Once a Web service is deployed, other applications (and other Web services) can discover and invoke the deployed service. So Web service based transactions may lead to more complex and concurrent execution. Transaction management is major impact on Web service [3].

2. Background Theory

2.1. Web Technologies

The web once solely a repository for text images, is evolving into a provider of services, information providing service such as flight reservation service, temperature sensors, and a variety of e-commerce and business to business applications. Web technologies include web server, browser, Domain Name services etc.. E-commerce application use web technology and web services composed with UDDI, SOAP, and XML messaging [7][10].

2.2. Web Service

The term "Web Services" is generally used to describe a collection of protocols and standards that are used to facilitate interoperability between applications [2]. In other way, A Web service is an interface that describes a collection of operations that are network accessible through standardized XML messaging. A Web service performs a specific task or a set of tasks. The web service specification represents an open standard for distributed service oriented architectures, already impacting a broad range of commerce and industry. Web service-based

business processes are long running processes and generally composed of invocations to internal business processes outsourced by loosely coupled providers [4]. They are self-contained, self describing, modular applications that can be published, located, and invoked across the Web. Web Services can be defined as software objects that can be assembled over the Internet using standard protocols to perform functions or execute business processes. Business services can be completely decentralized and distributed over the Internet and accessed by a wide variety of communications devices. Businesses are changed from traditional processes into the form of web applications or web service. Then the Internet will become a global common platform where organizations and individuals communicate with each other to carry out various commercial activities and to provide Value added services. And so Web services are rapidly growth.

2.3. Important of Transaction scheduling

Transactions have long been considered mainly in the context of tightly couple database system. As enterprise information system more and more distributed [9]. Short running transactions such as Web Service Atomic transaction are typically implement using the two phase commit protocol. And so transaction scheduling is very important for web services because services may be time out or conflict if transaction schedule is not used in the system. Example of transaction conflicts as shown in Figure 1.

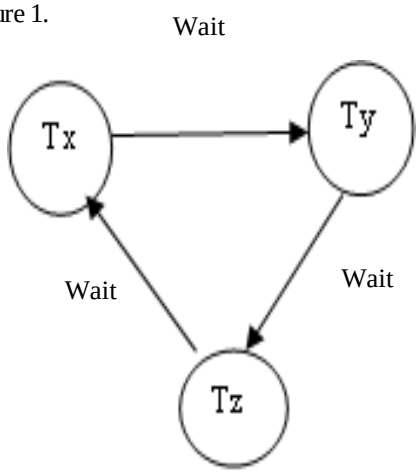

Figure 1.Conflict Transactions

Tx is an input to Ty, Ty is input to Tz, Tz is input to Tx. So, it is dependent transaction, It is lead to conflict and block transaction. When web service1 and web service2 invoke T1 and T2 concurrently, conflict may occur because of accessing the same resource.

2.4. Transaction Scheduling Algorithm

Extensible Transaction Scheduling Algorithm

Coordinator _ Process _ Execute _ Web Service Transaction

Begin

Input: T=set transaction task,

Dc=Client deadline for transaction Commit

Do

Collect Web Service Providers from UDDI;

For each provider P_i

Call Lock & Requisition Process;

Accept L & R from every provider;

Call Provider Selection Algorithm;

Construct execution plan;

Execute Actual Process;

End

Figure 2.Transaction Scheduling Algorithm

Provider _ Process _ Lock & Requisition Time

Begin

Input:

L=expected lock time; R= release time; Scheduler Q, L=0, R=0, maximum expectable time m=0, T_k where k is Position of transaction T, expected time E

Do

If k=0 then

Lk=E+0; Rk=m;

Else

Lk=Rk-1; Rk=Lk+1;

End If

Return L, R;

End

Figure 3.Provider Process Lock & Requisition Time

Provider _ Selection _ Algorithm

Begin

Input:

Dc=Deadline, F= set of collected offers; Set $F_i(P,L,R)$=list of collected offers order by L where i is position of offers; P=Reference for Provider; L=Lock Acquisition Time; R=Release Time

Do

L_{max}=max($F_v[0](L)$; // L_{max}= expected time for the last lock

Plan[v]=$F_v[0]$; // v=position

For each ti in T

Do

While ($F_i[0](L) <= L_{max}$ && Plan[i] isEmpty)Do

If ($F_i[0](R) > L_{max}$) then Plan[i]=$F_i[0]$;

Else $F_i = F_i - F_i[0]$;

End While

If Plan[i] isEmpty then

$F_v=F_v-F_v[0]$;

Plan[v]=$F_v[0]$;

While (L_{max}<Dc)

If ($Fi[0](R) > L_{max}$) then

Plan[v+1]=$F_i[0]$;

Else $F_i=F_i-F_i[0]$;

End While

End if

End For

Return Plan[];

End

Figure 4.Provider Selection Algorithm

3. Implementation of the System

Job Seeking System processes based on web service and web technologies that manage users request information and online job searching.

The system is implemented using web sites which use multiple web services to control their transaction processes. The system includes job request, provide job list to users, request confirmation processes. Many job providers (company lists) are stored in the agency database. The system has five distinct processes such as Login, Search, Job Request, Confirm Process and Admin portion. System admin must prepare or update company data, job vacancy and other system information. When users enter the system, he can search available job information including no of post, required qualification, required date etc. But to request job for a specific company he must login first. Otherwise he must create account to join the system. After requesting the post, system admin check users' quality and other requirement. If requirement check is valid then the requested users will be in the confirm lists in order to send form to specific company. To implement web services transactions, the system is issued that includes job agency A and job agency B. They invoke job request service for available confirmation. Agency A must call request web service to confirm user request information and Agency B must also too to prevent blocking occurrence or other transaction errors.

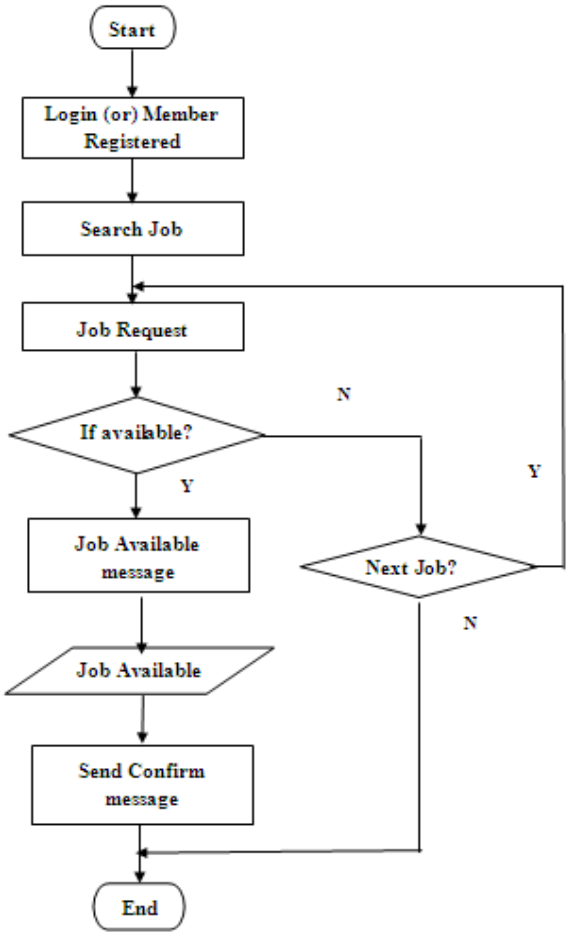

Figure 5.Flow Diagram for Job Seeking System

In the system service requesters are the users who want to get any jobs and service providers are web sites of Job Seeking System. In the UDDI registry service providers, web services and offers of transactions that required web services are stored. The task of coordinator is to manage web service transactions and to coordinate requested providers and services. In the figure, requester are job seekers and the arrows from Web site to WS (web Services) indicate the request link.

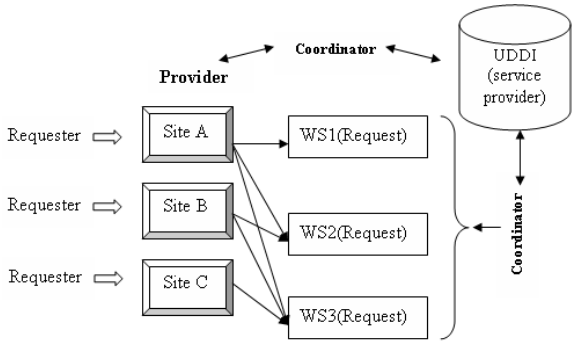

Figure 6.Service Providers and UDDI

3.1. Non-blocking Scheduling for Web Services Transactions

Web services transactions based on concurrency. When concurrent access to same resources such as same job position at same location, more probability for completing dependency among transactions may be occur. So any transactions will be block at commit time when waiting other transaction. In the system job request users can receive their requested result to commit before deadline and no blocking by using web service transaction scheduling. Scenario about the system shows in Figure 7.

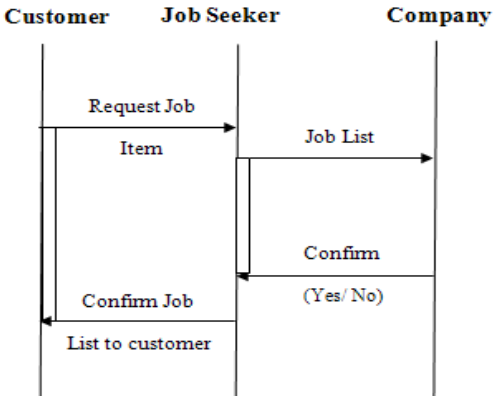

Figure 7.On-line order request transactions

3.2. Transaction Management in UDDI

The Web service-based transactional process is composed at design time by specifying abstract services and predefined deadlines. The scheduling can be then conducted by the transaction manager of the client's application i.e. the Transaction Coordinator. Overall process is shown in Figure 8.

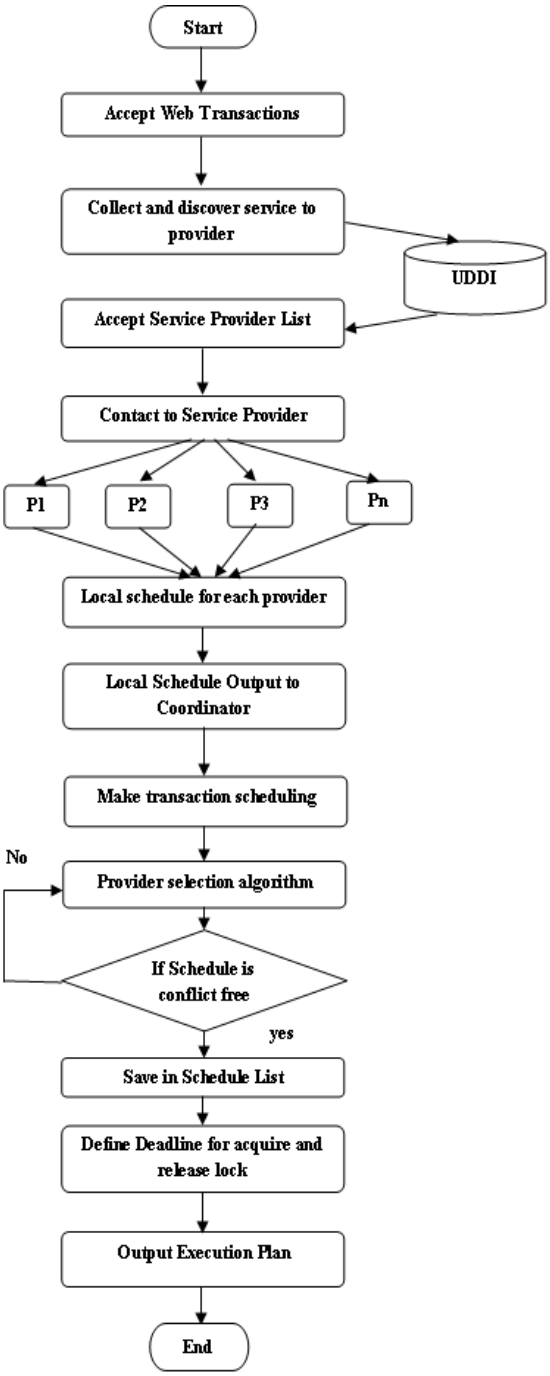

Figure 8.Overview Design for Web Service Transaction Scheduling

The Coordinator takes the abstract transactional processes as input and produces as output a concrete execution plan (i.e. a concrete service implementation for each abstract service) that does not suffer the problem of long waiting times. The

coordinator requests local schedule from all participant providers. In order to get final execution plan, the transaction scheduling algorithm invokes provider selection algorithm. Given all information from local schedules at the different sites, the provider selection algorithm is used to efficiently find a correct and possibly optimal combination of web service providers such that no timing conflicts between their local schedules occur, no transaction blocking is needed. Provider selection Algorithm is the most challenging part in transaction scheduling algorithm. The provider selection algorithm is used to efficiently find a correct and possibly optimal combination of Web service providers.

4. Related Work

Conventional ACID transactions are not well suited to Web Services [3].

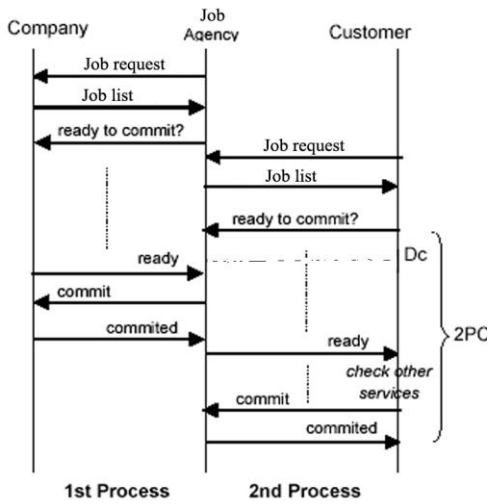

Figure 9.Two Interdependent business processes

A web services based transaction T is a partially ordered Set on tasks T. There is a completion dependency between transactions T1 and T2 invoke same web services [8]. Conventional transactional scheduling must obey ACID properties that do not flexible for web service transactions. Because business activities many composite web services may be long running and long waiting processes that might be commit independently of each other without having to wait for the root transaction to commit Web-services-based business transactions are long-running process, blocked transactions which have some deadlines are likely to abort because of the missed deadlines. The higher the degree of concurrency in the system, the higher the number of aborted transactions because of missed deadlines.

5. Conclusion

The system applies web service transactions using transaction scheduling algorithm for concurrency control to support the time characteristics of the transactions in Web service environment [5][6]. It takes into account the deadline constraints of both the consumers and the providers of Web services while scheduling the execution of global transaction at every participating site. This system is beneficial for both the service consumer and service provider. Transaction scheduling algorithm controls the system of web service transactions.

6. Further Extension

The system based on web application which includes search services. But Search Engine cannot available. More Admin control and search are further extension for other studies. And so the growth of Internet, computer telephony, electronic banking, interactive media services, mobile communication and related technologies have paved the way for new global network the Internet and telephone network emerge. This system can be extended to Internet with e-banking services.

References

[1] Eric Newcomer, "Understanding Web Services XML, WSDL, SOAP and UDDI"

[2] Implementing database transaction with michrosoft.net "Web services description language." W3C.

[3] Implementing database transaction with michrosoft.net http://msdn.moscrosoft.com

[4] McGregor, C. and Schiefer, J. (2003) 'A framework for analyzing and measuring business performance with web services', Presented at E-Commerce, 2003. CEC 2003. IEEE

[5] http://www.w3.org/TR/wsdl#introduction

[6] http://www.w3.org/TR/2002/WD.WS-arch-scenarios-20020730

[7] http://www.uddi.org/specification.html

[8]. Paul Anderson, What is Web Ideas, technologies and implications for education

[9]. O'REILLY, T. 2005b. *Web 2.0: Compact Definition.* O'Reilly Radar (blog), 1st

[10] Web Services Agreement Specification (WS Agreement September 2005, published

http://www.gridforum.org/Public_Comment_Docs/Documents