Cost Model and Adaptive Scheme for Publish/Subscribe Systems on Mobile Grid Environments

Sangyoon Oh[1,2], Sangmi Lee Pallickara[2], Sunghoon Ko[1],
Jai-Hoon Kim[1,3], and Geoffrey Fox[1,2]

[1] Community Grids Computing Laboratory, Indiana University, Bloomington, IN. USA
{ohsangy, leesangm, suko, jaikim, gcf}@indiana.edu
[2] Department of Computer Science, Indiana University, Bloomington, IN. USA
[3] Graduate School of Information and Communications, Ajou University, Suwon, S. Korea
jaikim@ajou.ac.kr

Abstract. Publish/subscribe model is appropriate in many push based data dissemination applications. This paper presents cost model for publish/subscribe systems, analyze its performance, and compare to other interaction-based models such as the client-server model and the polling model. Based on the cost analysis, we have proposed an adaptive model which can dynamically select an appropriate model for each client independently.

1 Introduction

Publish/subscribe system [1] have been widely used in many applications [2], [3]. Publish/subscribe system consists of publishers (ES: Event Source), servers (EBS: Event Brokering System), and subscribers (ED: Event Displayer). After a publisher publishes data (events) asynchronously to a server, the server disseminates the data (events) to subscribers which registered their interest on the server. Thus publish/subscribe model is appropriate in many applications such as data dissemination services, information sharing, service discovery, etc. Fig. 1 depicts system configurations.

In this paper, we present cost model for publish/subscribe systems, analyze its performance, and compare to other interaction based models such as a client-server model and a polling models. We can estimate performance and adopt publish/subscribe systems effectively by using our proposed cost model and analysis of publish/subscribe systems. Based on the cost analysis, we propose adaptive model which can dynamically select an appropriate model for each client independently. We believe the adaptive scheme we introduce here is very useful for the mobile and ubiquitous services where characteristics of device and networks are diverse and dynamically changing. We also experimentally measured and compared performance of publish/subscribe model to client/server model

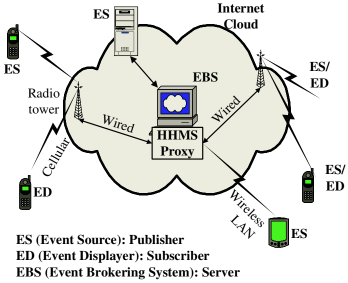

ES
Internet
Cloud
ES
EBS
ES/
ED
Radio
tower
Wired
Wired
HHMS
Proxy
Cellular
ED
ES/
ED
Wireless
LAN
ES
ES (Event Source): Publisher
ED (Event Displayer): Subscriber
EBS (Event Brokering System): Server

Fig. 1. Pub/Sub System Configurations

V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3516, pp. 275 – 278, 2005.

on our test bed including mobile device and NaradaBrokering [4] (our publish/subscribe based message brokering system) to verify correctness of our model on the real systems. Our cost analysis model is simple but accordant with experimental results.

2 Cost Model

System Models

To evaluate the cost model for different systems, we assume following system parameters to analyze cost: α, publish rate of event; β, subscriber's access rate of published events or request rate of client in the client/server models; $c_{ps}(\alpha)$, publish/subscribe cost per event, $c_{pub} + c_{sub}$; $c_{rr}(\beta)$, cost per request and reply; $c_{poll}(\alpha,T)$, cost of periodic publish or polling; $c_{delay}(\alpha,T)$, cost of delaying publish; $s(n)$, effect of sharing among n subscribers; t_{ps}, time delay for publish/subscribe, $t_{pub} + t_{sub}$; t_{rr}, time delay for request and reply; $t_{poll}(\alpha, T)$, time delay for periodic publish.

Cost Analysis

In this analysis, we analyze cost of three different models without any failure of communication link or node. We consider (1) conceptual total cost (e.g., the number of message, amount of message, or time delay) per unit time for each model, (2) cost for each access by client (or subscriber), (3) time delay for access after subscriber's (or client's) intention, and (4) time delay between event occurrence and notification to subscriber (or recognition by client). Cost can be the number of message, amount of message, or time delay. Table 2 shows the summary of the cost for each model analyzed in this paper. Please refer to [5] for detailed analysis.

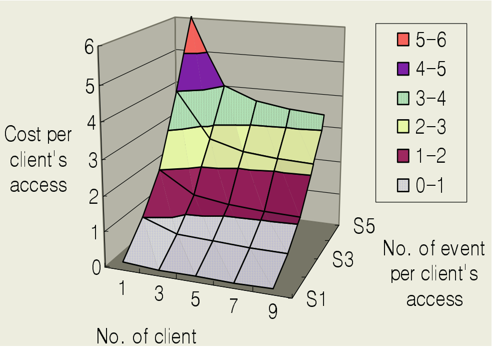
Table 1. The cost of the selected model

Model	Publish/Subscribe	Request/Reply	Polling
conceptual total cost per time unit	$\alpha\,(c_{pub} + n\,s(n)c_{sub})$	$\beta\,n\,c_{rr}.$	$(c_{poll}(\alpha,T) + c_{delay}(\alpha,T))\,/\,T$
cost for each access	$\frac{\alpha}{\beta}\,(\frac{c_{pub}}{n} + c_{sub})$	c_{rr}	$c_{poll}(\alpha,T) + c_{delay}(\alpha,T)$
time delay between intention and access	0	t_{rr}	$T/2$
time delay between event occurrence and notification/recognition (access)	$t_{ps} = t_{pub} + t_{sub}$ $(t_{ps} = t_{pub} + t_{sub} + \frac{1}{\beta}\,)$	$\frac{1}{2\beta}$	$T/2$

Adaptive Scheme

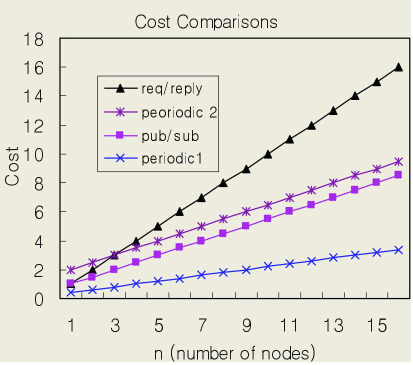
Adaptive scheme can choose an appropriate model among publish/subscribe and request/reply models. Each client node can select its own model independently (hybrid model) and change its model during its service (dynamic model).

In this paper, we consider cost per each client's access as a cost metric. During a period of time, the average number of events occurred per client's access is measured for each client. At the end of the period, the average cost for each client's access is computed using the analysis in section 2, which is $\frac{\alpha}{\beta}(\frac{c_{pub}}{n}+c_{sub})$, where $\frac{\alpha}{\beta}$ is average number of event occurred per client's access and n is the number of subscriber. In our adaptive scheme, average number of event and the number of subscriber are obtained experimentally during the execution of application. At the end of the period, the model that is expected to require less cost than the other model during the following period is selected independently for each client. Fig.2 shows that publish/subscribe model is appropriate when the number of client is large and/or the number of event per client's access is small.



(c_{pub}=1 and c_{sub}=1)

Fig. 2. Cost per client's access of publish / subscribe model



(α = 0.5, s(n)=1, c_{ps} = 2, and c_{rr} = 2; $c_{pub}(\alpha,T)$= c_{pub}, $c_{sub}(\alpha,T)$= c_{sub}, and $c_{delay}(\alpha,T)$= 0 for periodic1; $c_{pub}(\alpha,T)$= $\alpha T c_{pub}$, $c_{sub}(\alpha,T)$= $\alpha T c_{sub}$, $c_{delay}(\alpha,T)$= $2\alpha T c_{dealy}$ for periodic2)

Fig. 3. Communication cost by varying number of clients

3 Performance Comparisons

Parametric Analysis

In this section, we describe performance comparisons by parametric analysis. We set system parameters as shown in Table 2. Fig.3 shows performance comparisons between publish/subscribe, request/reply, and polling systems. Since publish/subscriber system disseminates data via server instead of individually for each client, it requires less cost than request/reply system. As the number of client node increases, the cost gap between two systems increases. Periodic polling system saves cost by transferring data once per period when delay cost is negligible. However, cost increases as delay cost increases. Polling system is viable approach when data delay is allowed and cost is negligible.

Table 2. Parameters

Param.	Values
α, β	0.5
c_{ps}, c_{rr}	2
c_{pub}, c_{sub}	1
$c_{poll}(\alpha, T)$	1 or αT
$c_{delay}(\alpha, T)$	0, T, or αT
$s(n)$	$1/n$ - 1
t_{ps}, t_{rr}	1
t_{proc}	1 or 5
$t_{poll}(\alpha, T)$	1, T, or αT

Experimental Results

The performance framework consists of NaradaBrokering System and Handheld Message Service (HHMS) [6]. The framework executes test applications – written in J2ME MIPD 2.0 and J2SE. We wrote two applications for two different communication paradigms – pub/sub and Remote Procedure Call (RPC). A client of the first application (ED) echoes message back to the sender (ES). With them, we experimented to perform the test for a cost of message. By varying a size of message, we measured RTT of different size messages. The result is shown in Fig.4. A client of the second application replies ACK back to the sender to measure RTT of different number of mobile clients. The second test was executed with mobile client emulators that come with Sun Microsystems' J2ME WirelessToolkit. This is a limited configuration, but it is still enough to exemplify the analysis we've made in Section 2. We define the data transition time of publish/subscribe and RPC as RTT/2 and RTT respectively from the semantics of each messaging scheme. Experiments are performed on mobile devices – a Samsung SPH-I300 phone and a Treo 600 with Sprint PCS Vision service and Linux machine.

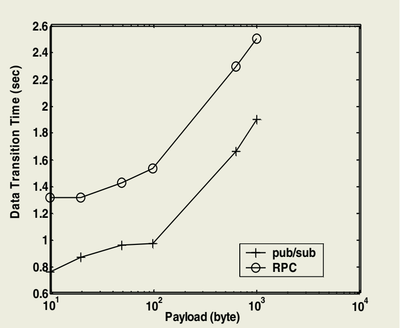


Fig. 4. Latency by Payload

4 Conclusions

We presented cost analysis model for publish/subscribe systems. Based on the cost analysis, we proposed an adaptive scheme which can dynamically select an appropriate model for each client independently. Experimental results (delay time) from our test bed were quite similar to our cost analysis models, which verifies that our cost model is useful to select proper model and to design adaptive schemes.

References

1. P. Eugster, P. Felber, R. Guerraoui, and A. Kermarrec, "The Many Faces of Publish/Subscribe", ACM Computing Surveys, vol. 35, no. 2, pp. 114-131, Jun. 2003.
2. A. Rowstron, A. Kermarrec, M. Castro, and P. Druschel, "SCRIBE: The design of a large-scale event notification infrastructure", Networked Group Communication, 2001.
3. A. Uyar, S. Pallickara and G. Fox, "Audio Video Conferencing in Distributed Brokering Systems", Proc. of the International Conf. on Communications in Computing, June 2003.
4. S. Pallickara and G. C. Fox, "NaradaBrokering: A Middleware Framework and Architecture for Enabling Durable Peer-to-Peer Grids", Proc. of ACM/IFIP/USENIX International Middleware Conference Middleware, pp 41-61, 2003.
5. S Oh, S. Lee Pallickara, S. Ko, J. Kim, G. Fox, "Cost Model and Adaptive Scheme for Publish/ Subscribe Systems on Mobile Environments", Community Grids Lab Technical Report, Dec. 2004. (http://grids.ucs.indiana.edu/ptliupages/publications/)
6. S. Oh, G. C. Fox, S. Ko, "GMSME: An Architecture for Heterogeneous Collaboration with Mobile Devices", Proc. of the Fifth IEEE/IFIP MWCN 2003, Singapore, Oct. 2003.