Available online at http://www.bretj.com

International Journal of Current Engineering Sciences

RESEARCH ARTICLE

ISSN: 2320-9046

International Journal of Current Engineering Sciences- Vol.4, Issue, 2, pp. 1-3, February, 2015

A FAULT TOLERANT MOBILE AGENT SYSTEM

[1]Punithavathi, R and [2]Duraiswamy, K

[1]Department of Information Technology, Vivekananda Institute of Engineering & Technology For Women, Tiruchengode, Namakkal(DT), Tamil Nadu, India
[2] Department of Computer Science and Engineering, KS Rangasamy, College of Technology, Tiruchengode, Tamilnadu India

A R T I C L E I N F O

Article History

Received 8th, January, 2015
Received in revised form 20th, January, 2015
Accepted 10th, February, 2015
Published online 28th, February, 2015

Key words:

Client-server architecture, mobile agent paradigm, mobile computing environment, fault tolerant, mobile agent

A B S T R A C T

Mobile agent paradigm is an efficient approach for mobile computing environment based applications such as Mobile banking system. Though, client server paradigm is used in many emerging applications, a shift of paradigm is needed to create service infrastructures, capable of providing high quality services and utilizing the network resource efficiently. In the mobile agent system, the absence of execution of mobile agent can be considered as fault. In order to make the mobile agent system as a fault tolerant, the system has to continue to provide the normal operation when the failure of one or more mobile agent is not performing the assigned task due to loss. For a reliable mobile agent application like mobile banking, it must be fault tolerant. A modified volunteer based coordinator selection fault tolerant algorithm is proposed.

INTRODUCTION

A mobile agent (or simply agent) is a self-contained software element responsible for executing a programmatic process, which is capable of autonomously migrating through a network. An agent migrates in a distributed environment between logical “places” referred her to as agencies. When an agent migrates, its execution is suspended at the original agency, the agent is transported (i.e. program code, data, execution state and control information) to another agency in the distributed environment, and, after being re-instantiated at the new agency, the agent resumes execution Raimundo *et al* (2001).

The IEEE defines fault tolerance as (IEEE 1983):

1. The ability of a system or component to continue normal operation despite the presence of hardware or software fault. or
2. The number of faults a system or component can withstand before normal operation is impaired. or
3. Pertaining to the study of errors, faults, and failures, and methods for enabling systems to continue normal operation in the presence of fault.

In the mobile agent system, the absence of execution of mobile agent can be considered as fault. In order to make the mobile agent system as a fault tolerant, the system has to continue to provide the normal operation when the failure of one or more mobile agent is not performing the assigned task due to loss. For a reliable mobile agent application like mobile banking, it must be fault tolerant.

The basic idea of the volunteer algorithm is explained with the following steps:

1. if an agent is not currently committed to execute any plan, it will try to find a plan in which it can play a role
2. the agent will find out whether that role has been assigned to any other agents,
3. If that role has not been assigned to any other agent, the agent will volunteer to play that role and send a message to everyone else in the team Yu Zhang and Jianwen Yin (2006).

The Extended Volunteer Algorithm (XVA) (Marikkannu 2011) based on the volunteer algorithm is valid only inside the Context-Aware Environment. The additionally assigned roles are discarded once the mobile agent gets out of the environment. So the Mobile Agent, when it moves out of the environment can carry only the data that is concerned with its basic role. For applications such as mobile banking where the mobile host is not fixed and also mobile banking transaction must be exactly once, XVA is not suitable.

The mobile group concept was introduced to provide synchronization among a group of mobile agents Silva and Macedo (2000). A mobile group is an extension of the traditional concept of a process group that can directly

Corresponding author:* **R. Punithavathi
Department of Information Technology, Vivekananda Institute of Engineering & Technology For Women, Tiruchengode, Namakkal(DT), Tamil Nadu, India 637205

support migrating processes as members of the group. With mobile groups, a migrating process has the ability to change its location in the distributed environment while belonging to a group. Mobile groups also provide message delivery guarantees and a sort of virtual synchrony. However, mobile groups provide these guarantees despite the mobility of their members. Furthermore, they make process mobility not only visible for the group, but also consistently ordered with other group actions (such as process crashes, joins, and leaves). Due to the guarantees that they provide, mobile groups represent a suitable abstraction for the development of reliable agent-based applications Flavio Assis Silva and Raimundo Macedo (2001).

In a mobile group, agent communicates with each other by exchanging messages which are multicasted to the whole group. In order to preserve consistency among the state of group members, the group communication protocols guarantee certain properties such as atomic delivery (either all processes deliver a message or no one delivers it), message ordering guarantees (e.g., causal and total), and a sort of virtual synchrony where modifications on the group membership (caused by events such as process crashes and joins, etc.) are consistently ordered with respect to message delivery.

In mobile groups, a agent can migrate while belonging to the group. The mobile group system recognizes the movement action of the agent and considers movement as a single group operation.

A concept for coordinating groups of agents is described, but which is not fault tolerant Baumann and Radouniklis (1997). Approaches for reliable message delivery to mobile agents are proposed, but failures are not tolerated Murphy and Picco (1999). In another approach failures are considered, but the approach supports only unicast (peer-to-peer communication) Ranganathan *et al* (2000). A concept of mobile group is discussed with only fixed host, it does not deal with the mobile group for the mobile devices Flavio Assis Silva and Raimundo Macedo (2001).

MATERIALS AND METHODS

Fault tolerant algorithm

The set of agents used in mobile banking will form a mobile group. Each agent represents a process of the group like authentication, balance checking, bill payment, fund transfer etc. If any agent fails, the coordinator create new agents to do the job of failed ones. Also, since mobile groups provide a consistent view of process locations, when an agent decides to visit a new location, it will not move to locations it knows were already visited by another agent of the group. If the coordinator fails, the other agents in the mobile group will eventually know about this. To select the new coordinator, the agent with the greatest process id in the group concept is used Flavio Assis Silva and Raimundo Macedo (2001). This concept may not always work properly. A suitable coordinator algorithm is needed to make the system more reliable.

A new coordinator can be elected by applying volunteer algorithm on the set of operational agents. The agency informs the failure of the coordinator to the agent directory. To select the new coordinator in case of failure, a modified volunteer based coordinator selection fault tolerant algorithm is proposed.

Volunteer Based Coordinator Selection Fault Tolerant Algorithm

```
Step 1: The agent directory sends a ready_to_accept message to all the agents
        In the mobile process group.
Step 2: If any agent responds with I_am_ready message within a specific time
        Interval t then
Set True to the ready_to_work message for that agent
Else
Set False to the ready_to_work message for agents not responded within time interval t
Step 3: If no_of_roles < 2 with low priority and ready_to_work = True then
3a. Assign the additional role as coordinator
3b. Add 1 to the no of _roles
3c. reset the value of ready_to_work
Step 4: If there is no agent with the above criteria then
Create new agent and assign it as the new coordinator
```

Table 1 Fault Tolerant Coordinator Selection At Various Intervals

Number of user requests	After 10 ms	After 25 ms	After 50 ms	After 100 ms
1	213	194	185	172
5	270	253	241	233
10	292	268	257	243
15	303	287	279	264
20	312	298	285	277
25	324	312	305	394
30	388	374	362	352

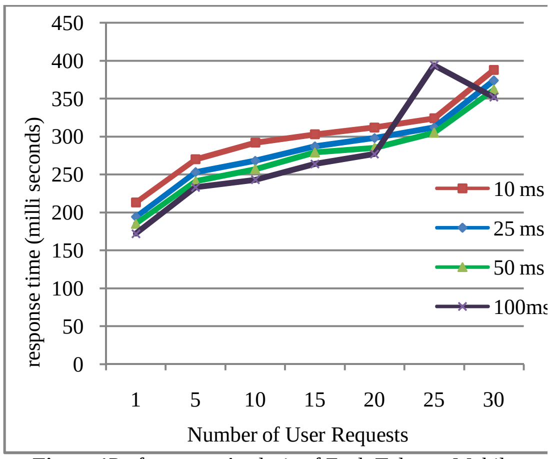

Figure 1 Performance Analysis of Fault Tolerant Mobile Banking System

RESULTS AND DISCUSSION

Fault tolerant trade-off analysis

It discusses about a trade-off analysis in order to determine the coordinator selection intervals satisfying a response time goal with the minimum possible cost

Panagiotis Katsaros *et al* (2007). As shown in Table 3.1, a fault tolerant coordinator is selected after every 10 ms, 25 ms, 50 ms and 100 ms and the respective response time of the mobile banking system is given. This analysis has to be considered for other applications and also for other environment

As shown in the Figure 3.1, if the fault tolerant coordinator is selected after every 100 ms, the cost associated with the overhead is minimal. So that the minimum response time goal is achieved. But when the coordinator is failed in between, the cost associated for making the system to be fault tolerant is high. If the fault tolerant coordinator is selected after every 10 ms, the cost associated is high. The response time is also high. Based on the fault tolerant trade-off analysis, the interval for the selection of fault tolerant coordinator has to be decided. Using this interval value, the performance can be increased or decreased.

CONCLUSION AND FUTURE WORK

In the fault tolerant mobile banking system based on mobile group, the coordinator is selected at various intervals like after every 10 ms, 25 ms, 50 ms and 100 ms. A fault tolerant trade off analysis is used to select the interval. So that the response time goal is achieved.

References

Assis Silva, F.M. and Macedo, R.J.A. “Reliability Requirements in Mobile Agent Systems” in Proceedings of the second Workshop on Tests and Fault Tolerance, Curitiba, Brazil, July 2000.

Baumann, J. and Radouniklis, N. “Agent Groups in Mobile Agent Systems” in Proceedings of the Distributed Applications and Interoperable Systems, Chapman & Hall, 1997.

Flavio M. Assis Silva and Raimundo J.A. Macedo, “Reliable Communication for Mobile Agents with Mobile Groups”, in the Proceedings of the Workshop on Software Engineering and Mobility (co-located with IEEE/ACM ICSE 2001). Toronto, Ontario, Canada. May 13-14, 2001.

IEEE: IEEE Standard Glossary of Software Engineering Terminology, IEEE Std, pp. 729-1983, 1983.

Marikkannu, P., Adri Jovin, J.J. and Purusothaman, T. “Fault-Tolerant Adaptive Mobile Agent System using Dynamic Role based Access Control”, International Journal of Computer Applications (0975 – 8887) Vol. 20, No.2, 2011.

Murphy, A. and Picco, G.P. “Reliable Communication for Highly Mobile Agents” in Proceedings of the Joint Symposium on Agent Systems and Applications / Mobile Agents, 1999.

Panagiotis Katsaros, Lefteris Angelis and Constantine Lazos, “Performance and effectiveness trade-off for checkpointing in fault-tolerant distributed systems” Research Articles. Concurrency and Computation: Practice & Experience Vol. 19, Issue 1, 2007.

Raimundo Jose de Araujo Macedo and Flavio Morais de Assis Silva, “Mobile Groups” Proceedings of the 19th Brazilian Symposium on Computer Networks – SBRC 2001, Florianópolis, SC, Brazil. 2001.

Ranganathan, M., Bednarek, M. and Montgomery, D. “A Reliable Message Delivery Protocol for Mobile Agents” in Proceedings of the Joint Symposium on Agent Systems and Applications / Mobile Agents, 2000.

Yu Zhang and Jianwen Yin, “A Role-Based Modeling for Agent Teams” in Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and its applications 2006.

How to cite this article:

Punithavathi, R and Duraiswamy, K., A Fault Tolerant Mobile Agent System. International Journal of Current Engineering Sciences- Vol.4, Issue, 2, pp. 1-3, February, 2015

****