Mobile Agents for reliable distributed computing

Waleed Wagealla, Taha Osman

Intelligent Simulation and Modelling Group
Department of Computing
The Nottingham Trent University
Butron Street
Nottingham NG1 4BU, UK
Waleed.wagealla@ntu.ac.uk

Key words:
Mobile Agents, Fault-Tolerance, Error detection, Distributed system

Abstract

Previous research work in the ISM (Intelligent Simulation & Modelling) group focused on the reliability of fixed-network distributed systems [1]. The new research direction aims to extend the fault-tolerance research to embrace mobile computing systems, in particular mobile-agents systems [2].

Recently there was an explosion of distributed networked computing, boosted by the reduction in the cost of computing systems, network devices, and the global adoption of the Internet [3]. The Internet created a technological opportunity for the development of large-scale distributed systems by providing solid ground for establishing global networked applications. The Mobile Agents technology is best suited to serve such applications.

Mobile agents are hardware independent software entities that can autonomously roam the network on behalf of a user. Agents can access network resources more efficiently because they move to their network location rather than transferring multiple requests and responses across busy network link. Another advantage of agents is that they need not to be in continuous contact with the originator node, therefore the agent is not affected by sudden loss of connection, and can continue its task even if the user's computer powers down or disconnects from the network.

Hence, the agent distributed processing model accommodates great possibilities for developing dynamically growing distributed systems and are also suitable for partially connected computing [4]. This new paradigm is helped to shift from static to dynamic distributed systems.

While there is a significant research activity in the area of mobile computing, some very important problems have not been solved [5]. A lot of research effort concentrates on providing the basic system support for migration, communication, and security of the platform underlying the agent systems, but few fully addressed the reliability of agent systems and applications.

The previous reliability research has focused on fixed distributed systems, but the emergence of platform-independent mobile-code technologies meant that the network of hosts for distributed applications could grow dynamically. This complicates the procedure of fault detection and error recovery since there is a set of bookkeeping operations required for agent tracking and locating available distributed resources. It must be stressed that the probability of failure of agent-based distributed systems is higher because agents carry their execution state with them as they migrate, consequently a transmission error can result in execution failure as well as inter-process communication failure. A reliable agent-based system has become an important issue since mobile agents used wide spread in different types of applications.

The main objective of our research is to introduce a fault-tolerant wrapper that guarantees the reliable execution of mobile agents - based applications in the presence of fault that might occur in the underlying hardware environment. Two points are considered under our research umbrella, as well as granting reliable agents also providing reliable hosting machines; considering network partitioning, code failure, and agent crashes. Which is a unique research direction to look after agents and the environments in which they working.

The whole view of our research is implementing a fully-featured fault tolerant agent-based package, which its includes error detection mechanism and recovery procedures. The starting point was error detection of agent technology. Error detection is an issue that is frequently overlooked in agent-based

systems. We designed an algorithm for the detecting errors in agent-based distributed system, which is continued on previous research results in our research group concerning agent reliability [6].

We recognised the challenges posed by the dynamically evolving networks in which the agents mobile code lives. In contrast with fixed distributed hosts, dynamic networks require extra bookkeeping tasks for agent tracking and locating available resources. Moreover, there is the problem of agent failure during migration. We proposed to divide the agents distributed platform into spaces, allowing the seamless addition of new sub-networks – possibly with different topology and hardware architecture (see Fig. 1 and Fig. 2). Each space was assigned a server that manages all error detection operations for the nodes and agents in at the micro (space) level, thus preventing a total breakdown of the system in case of network partitioning. This also allows the central detection manager to perform a co-ordinating role for agent migrations between spaces (macro level) and prevents it from becoming a bottleneck for network detection messages between agents.

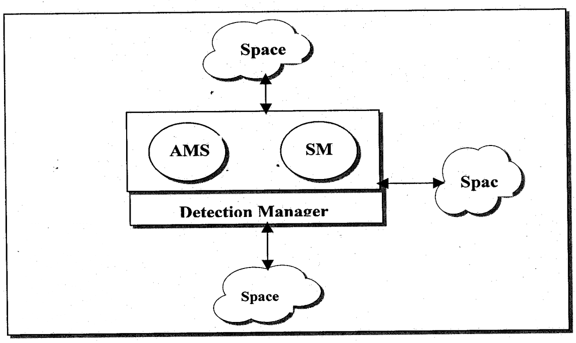

Fig. 1 The Space architecture

We adopted a hierarchical approach to designing the error-detection mechanism. The top of the pyramid is the detection manager, that co-ordinate the overall detection process for the spaces subscribed to our system. Each space contains a space server that looks after managing error detection of the space's agents and nodes. Finally we consider detecting faults at the lowest level - nodes and agents.

We mostly used messaging techniques for fault detection since for simple operations (operational status checking, liveliness reporting). Inter-agent (inter-process) messages are more efficient than using object migrations and reduces the detection transaction commit overhead.

By reaching error detection mechanism after a comprehensive study, evaluation and investigation means that we half-finished from our tolerant agent based package. The next stages will be carrying-out the performance of the designed algorithms; checking the overhead of applying our tolerant wrapper to agent-based real time systems.

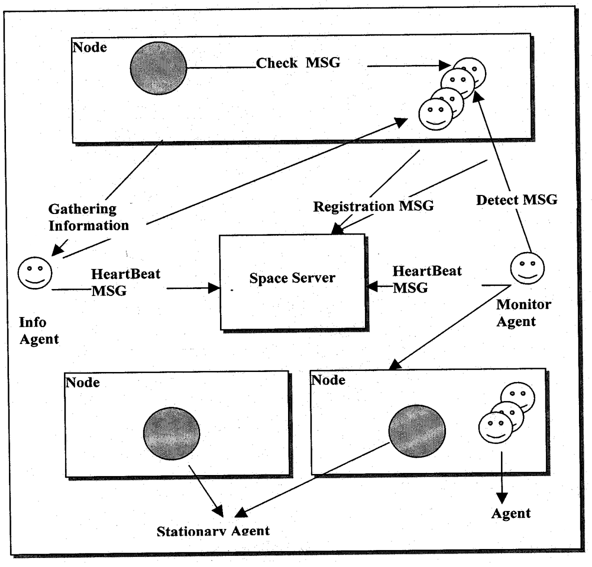

Fig 2. The Space Level

The core of the next stage is to complete the research by designing recovery methods, so as to be integrated with error detection mechanisms. Finally, comprehensive performance study and evaluation to the whole system need to be carried-out. This is necessary not only to prove the feasibility of the solution, but also to give the application developer a scale by which the reliability/performance ratio can be balanced.

References

1. Osman T., FADI: Afaullt Environment For Distributed Processing Systems, PhD thesis, The Nottingham Trent University, 1998.
2. Osman T, Bargiela A, " Fault-Tolerance for Mobile Agent Systems and Applications" Workshop 2000. Agent-based Simulation, pp. 211-221, Passau, Germany.
3. Chess D, Harrison C, Kershenbaum A. "Mobile agents: are they a good idea? " Mobile Object Systems. Towards the Programmable Internet. Second International Workshop, MOS 1996. , pp. 25-47. Berlin, Germany
4. General Magic, "Mobile Agents white paper", http://www.genmagic.com
5. Aridor Y, Oshima M. "Infrastructure for mobile agents: requirements and design". Mobile Agents Second International Workshop, 1998, pp. 38-49, Germany.
6. Wagealla W, Osman T, Bargiela A., Error detection algorithm for agent-based distributed applications. Submitted to a conference.