氏　　名	MOHAMMAD REZA SELIM
博士の専攻分野の名称	博士（学術）
学位記号番号	博理工甲第703号
学位授与年月日	平成20年9月19日
学位授与の条件	学位規則第4条第1項該当
学位論文題目	A Peer-to-Peer Network Based Middleware for Large-Scale Persistent Computing Systems （大規模永続計算システムのためのピアツーピアネットワークに基づくミドルウェア）
論文審査委員	委員長　教授　程　京徳 委　員　教授　吉田　紀彦 委　員　教授　池口　徹 委　員　准教授　山田　敏規

論文の内容の要旨

Background and Problem:

Many reliable systems need to run continuously even when they are being upgraded, maintained or reconfigured. Such systems were defined as Persistent Computing Systems (PCSs). The Soft System Bus (SSB) was proposed to function as a middleware of PCSs. The main requirements of SSB are that it must be asynchronous with data preservation facilities and it must support runtime upgradeability, maintainability and re-configurability. Although high level requirements of SSB were identified, the design and implementation of SSB has not been done before.

PCSs are necessary not only for small scale enterprises but also for large or even Internet scale enterprises. Large-scale PCSs has many promising applications including building a simple messaging system running in hundreds of sites of a large enterprise. We are mainly interested in large-scale systems. The main goal of designing such systems as PCSs is to get continuous services even when a disaster destroys one or more sites or when one or more nodes are being upgraded or maintained.

Traditional middleware based systems can rarely provide such services. They lack dynamism; if the middleware node(s) of a site fails the applications running in that site neither get service nor can connect to any other site.

Purpose and Objectives:

Our purpose is to design and evaluate an SSB (i.e., middleware) that can be used for large-scale PCSs. To achieve our goal we need to identify the additional requirements of SSB to work as a middleware for large-

scale PCSs, to propose a design which solves those requirements and to implement a PCS based on the proposed design in a simulated environment and finally to evaluate the design.

Approach and Solution:

A detailed analysis is done to elicit the requirements that need to be solved for a middleware for large-scale PCSs. We then investigate if any existing middleware is sufficient to satisfy those requirements. We argue that among three types of middlewares: synchronous, publish-subscribe and point-to-point Message Queuing Middleware (MQM), the last one is sufficiently strong to be used as middleware for PCSs. However, as we will show, MQM lacks dynamism and it has some limitations which make it costly in term of resource requirements for large scale systems.

To be usable for large-scale PCSs, we, therefore, propose a design of a middleware which is very dynamic. We used a simple mechanism to build the middleware: a structured peer-to-peer network. As it is built on peer-to-peer based overlay network, it provides a good routing efficiency and fault tolerance in large/Internet-scale systems. As various nodes and components of the systems can easily be removed or added, it provides a good environment for runtime (but incremental) upgradeability and maintainability. As we describe, there are a number of unique features of our middleware. It never loss any data although it ensures ordered delivery between any two components of the system which are connected asynchronously. If a node fails, its states never need to be recovered from persistent storage, therefore eliminating the need for expensive database servers, DBMS software and highly available persistent storage devices. It is not necessary to take special arrangement for disaster recovery provided that the system contains sufficient nodes distributed in geographic areas.

We have built a Chord p2p based simulator of a PCS based on our proposed middleware in order for evaluating our design. From our collected data so far, we show that in an ideal case it is possible to achieve a four nine availability ensuring ordered delivery and no loss of data. The maximum achievable (average) channel utilization about 90%.

Contributions:

Our work has following contributions:

1. We have identified and analyzed the requirements of SSB for large-scale PCSs.
2. We have investigated the traditional midlewares and shown that they lack dynamisms. Therefore they can not guarantee continuous services without using huge resources. For these reason existing middlewares are not appropriate for large-scale PCSs.
3. We have proposed a new design of SSB which can satisfy the requirements of large-scale PCSs.
4. In addition to satisfy the requirements its costs as little as one-fifth of a cluster based deployment of traditional middleware.
5. For the first time, we have used structured p2p network as a reliable point-to-point middleware. We also have proposed algorithms for lossless ordered data delivery in such a middleware.
6. We have evaluated our design with respective to the requirements of large-scale SSB and in some cases compared it with existing middlewares by using a simulator. The simulated behaviors imply that our middleware can fulfill the requirements and can provide high availability with reasonable performance.

Future Work:

We would like to compare our works with that of MQM from several perspective like routing efficiency, throughput, resource requirements etc. Therefore, we have to simulate the MQM and then compare with our middleware.

Our basic approach will be improved with respect to efficiency. We hope that using a small cache in every node can increase the routing efficiency. A Pastry based implementation of our simulation can minimize the inefficiency caused due to random distribution of nodes and components even if they are in the same locality. We also want to find an approach to distribute the load evenly among the nodes of the middleware.

論文の審査結果の要旨

ユビキタスコンピューティング(ubiquitous computing)や先行的コンピューティング(anticipatory computing)やサービスコンピューティング(services computing)などのような計算パラダイムを確立するためには、いつでも止まらずに動作し、計算・サービスを続けるシステムを実現しなければならない。永続計算システム(persistent computing system)とは、上記のような背景の下に提案された、外部環境からの計算・サービス要求に連続的に反応し、保守や更新のときも、故障が生じたときも、攻撃を受けたときも、止まらずに動作し、計算・サービスを続けるシステムである。永続計算システムを実現するための基盤としては、ソフトシステムバス(soft system bus)というミドルウェアが提案されている。しかし、最近までに、永続計算システムやソフトシステムバスに関する研究は、概念的、原理的、方法論的なものが多く、実践的、実験的な研究が殆どなかった。従って、実用的な観点に基づいた、永続計算システムやソフトシステムバスの実現に関する研究調査は、重要な課題になっている。

本論文は、大規模永続計算システムを構築するために、ピアツーピアネットワークに基づいてソフトシステムバスを実現する方法について、著者の実践的、実験的研究を通じて得た成果をまとめたものであり、5章から構成されている。

まず、第1章では、永続計算システムとソフトシステムバスの概念を簡単に紹介し、高度情報化社会におけるそれらの重要性、およびそれらの研究開発現状を説明し、本研究の位置付けを明確にしたうえ、本研究の目的が大規模永続計算システムを構築するために必要不可欠なソフトシステムバスの設計とその評価であることを示した。また、本研究における基本的考え、用いた技法、得られた成果、および本論文の構成について簡単に説明した。

第2章では、大規模永続計算システムを構築するためのソフトシステムバスに対して詳細な要求分析を行って得た結果を述べ、永続性、可用性、信頼性、安全性の各側面からソフトシステムバスが満足しなければならない要求を洗い出し明確に列挙した。そして、従来のミドルウェア（同期的なミドルウェアCORBA、Java RMI、MS DCOM、および非同期的なミドルウェアHermes、TIBCO Rendezvous、Siena、Java Messaging Services (JMS)、Rebeca、Meghdoot、IBM MQSeries/WebSphere、Active MQ、Oracle Advanced Queuing、Microsoft Message Queuing (MSMQ)）に対して比較研究を行った結果として、計算・サービスの永続性を最初から考慮に入れなかった従来の様々なミドルウェアの何れもソフトシステムバスに対する要求を満足できないという結論を述べた。

第3章では、第2章で列挙したソフトシステムバスに対する要求を満たす新しいミドルウェアとして、ピアツーピアネットワークに基づいたミドルウェアの設計について述べた。著者の基本的な考えは、ピアツーピアネットワークにおいてはノード（ピア）間の連結を動的に変えやすいという性質を利用し、ソフトシステムバスにおけるデータ・指令基地局をピアツーピアネットワークにおけるノードとして、また、データ・指令基地局間の通信チャンネルをピアツーピアネットワークとして実現することである。新しいミドルウェアのルーティングプロトコルは、ピアツーピアネットワークにおいてよく使われている、分散ハッシュテーブルを実現するChordアルゴリズムを用いた。これにより、ソフトシステムバスにおけるあるデータ・指令基地局に何らかの問題が生じる時に、そのデータ・指令基地局に繋がっている機能部品を簡単に別のデー

タ・指令基地局に繋ぎ替えることができる。また、データ・指令の複数コピーを別々のデータ・指令基地局に保存させることにより、データ・指令の紛失に対応することができる。

第４章では、新たに設計したミドルウェアのプロトタイプ（数十個のデータ・指令基地局と数百個の機能部品を持つシミュレータ）を作成し、それを使って実験を行い、実用の観点からその可用性、拡張性、効率性、コストを評価した結果、新しいミドルウェアは十分実用的なものであることを示した。

最後に、第５章では、本研究で得た成果をまとめ、残された研究課題を示した。

なお、本論文で述べた主な内容は、既に５編の論文としてまとめられ、学術論文誌や査読付きの国際会議論文集において公表されている。

以上のように、本論文は、大規模永続計算システムを構築するために、ソフトシステムバスを実際に実現する初めての方法として、ピアツーピアネットワークに基づいたミドルウェアを提案し、その設計思想、実現技法、実験よる評価結果を述べ、実用的なソフトシステムバスを実現することができることを示した。これらの研究成果は、情報工学分野にとって新しい知見と結果を示し貢献するものである。従って、当学位論文審査委員会は、本論文が、博士（学術）の学位を授与するに十分値するものと判定した。