マルチメディア通信と分散処理　102－25
（２００１．３．２２）

自律分散DBシステムにおけるフォールトトレランス技術

廣瀬雅人，平井健治，森欣司

東京工業大学 情報工学科
〒152-8552 東京都 目黒区 大岡山 2-12-1
TEL: 03-5734-2510, FAX: 03-5734-2510
E-mail: {piro, khirai}@mori.cs.titech.ac.jp , mori@cs.titech.ac.jp

あらまし：インターネットに代表される情報通信技術の発達に伴い情報サービスは多様化した。その結果エンドユーザの嗜好も多様化し、企業は関連異業種企業と情報を共有し合うことで経営の効率化をはかり、熾烈な競争に勝ち残ろうとしている。関連異業種企業システム間提携の一例であるSCMにみられるDB統合においては、リアルタイム性、柔軟性、フォールトトレランス性が、時々刻々と変化する状況下で同時に実現されることが求められており、これを実現するDBシステムとして、自律分散DBシステムが考案されている。本稿では、この自律分散DBシステムにおけるフォールトトレランス技術に焦点を当て、自律障害検知・回復技術の提案を行った。また、シミュレーションによりその有効性を示した。

Fault Tolerance Technique in Autonomous Decentralized Database System

Masato Hirose, Kenji Hirai, Kinji Mori

Dept. of Computer Science, Tokyo Institute of Technology
2-12-1 Ookayama, Meguro, Tokyo, 152-8552, Japan
TEL: 03-5734-2510, FAX: 03-5734-2510
E-mail: {piro, khirai}@mori.cs.titech.ac.jp , mori@cs.titech.ac.jp

Abstract:Information services and the users' requirements have been diversified as the development of the information technology. Under this social situation, companies would like to share the informaiton among related disparate industries. Thus the DB-integration is highly needed. It is required for DB system to realize real-time property, flexibility, and fault-tolerance under evolving situation. In order to achieve these properties, Autonomous Decentralized Database System (ADDS) has proposed. This paper is focused on fault-tolerance of the ADDS, and proposes the autonmous fault-detection technique and fault-recovery technique. Lastly effectiveness of the technique is shown by simulation.

1　はじめに

IT技術が急速に進歩している今日、インターネットを始めとする情報通信技術の発達により、社会は急速に変化している。技術の進歩に伴い情報サービスは多様化し、その結果一般ユーザは、インターネットを利用して、自分にあったサービスを自分の都合のよい時に受けることができるなど、エンドユーザ

の嗜好も益々多様化してきている。このような状況下の中、各企業は関連する異業種企業と情報を共有し合うことで、経営の効率化をはかり熾烈な競争に勝ち残ろうとしている。

異業種企業システム間提携の一例として、小売・流通・メーカの情報システムを統合させた、サプライチェーンマネージメント（SCM）がある。サプライチェーンマネージメントにみられる大規模異業種間システム統合においては、各企業はリアルタイムな在庫量の把握（余剰在庫の削減）と、受注更新処理の迅速さ（顧客要求に対する機械損失回避）という相反する要求を実現しなければならず、システムにはリアルタイム性が求められている。またこれと同時に、提携企業先の動的な変化や、ニーズの変化に対し容易に適応できる柔軟性、障害発生時にもシステムを停止させることなく常時サービス提供が行えるフォールトトレランス性も要求されている。その結果、これらのシステムニーズを、常時変動する環境下で同時に実現することが望まれている。

これらのニーズを実現させるデータベースシステムとして、自律分散データベースシステムが考案されている。自律分散ＤＢシステムは、自律性を持つサイトとモバイルエージェント（MA）から構成されている。各サイトは Allowable Volume という自律更新権限を保持しており、ＭＡがサイト間を自律的に巡回しこの Allowable Volume をリアルタイムに調整している。

本論文では、この自律分散データベースシステムにおけるフォールトトレランスに焦点を当て、自律障害検知・回復技術の提案を行った。提案したフォールトトレランス技術により、自律分散データベースシステムにおいて、ＭＡのダウンやサイトのダウンといった障害発生時においても、その自律的な検知・回復技術によりシステム全体を停止させることなく、自律的にモバイルエージェントや AllowableVolume を復旧させることができる。また、単一箇所の障害がシステム全体に波及することもなく、各ノードは連携可能な範囲内で自律的に稼動し続けることが可能となる。

さらに、提案したフォールトトレランス技術の有効性を示す為、シミュレーションにより分散データベースシステム内のノード間の平均連携率を求め、従来の分散データベースシステム（集中系）と比較を行った。その結果、システム規模が大きい場合でも、自律分散データベースシステムでは各ノードが自律している為、高いノード間連携率を得ることがでる。従って、より高いフォールトトレランス性を実現することが可能であるを示した。

2　背景とニーズ

2.1　背景

情報通信技術が急速に発達し、これまでにない様々なサービスが実現され得る現代において、サービスを受ける側であるユーザーのニーズは益々多様化している。その結果、企業を取り巻く環境はかつてない程、急速に変化しており、各企業は関連する異業種企業と提携することで、情報を共有し合い、常時変動するユーザーニーズに柔軟に対応し、激しい競争に打ち勝たねばならない。

異業種企業のシステム間提携の一例として、小売・流通・メーカーの情報システムを連携させるサプライチェーンマネージメント（SCM）がある。従来は、これらの企業間でデータベースシステムの統合は行われておらず、物流にかかわるそれぞれの会社は独自に他会社の要求量や売れ行きを判断することで、業務を遂行していた。こういった状況下では、各企業はリアルタイムに他会社の要求量や売れ行き、ユーザーニーズを把握することはできず、その結果企業は余剰在庫を抱え、販売の機会損失を発生させてしまうという、好ましくない状況に陥る事態であった。そこで、各企業のデーターベースシステムを統合し、相互の情報を共有し合いユーザーニーズをリアルタイムに把握することで、状況に応じた的確な生産計画を立て、余剰在庫削減の実現を図り、また販売においても受注から納入までの期間短縮が販売の機会損失削減を可能にするという意図を達成させているのがサプライチェーンマネージメントである。

2.2　ニーズ

このように、関連する異業種企業間での情報の共有に対するニーズ、つまりデーターベースシステム統合のニーズは、近年非常に高まってきている。

サプライチェーンマネージメントにみられるデー

タベースシステム統合においては、リアルタイムな在庫量の把握（余剰在庫削減）、納期保証の為の受注更新処理の迅速さ（顧客要求に対する機会損失の減少）という、相反する異種のニーズが存在する。これらの要求を実現させる為、システムにはリアルタイム性が求めらてれる。

またこれと同時に、提携企業先の動的な変化や、あらゆるニーズの変化に対し容易に対応できる柔軟性、障害発生時にもシステムを停止させることなく常時サービスが提供できるフォールトトレランスト性も要求されている。その結果、これらのシステムニーズを常時変動する環境下で同時に実現することが求められているのである。

ユーザニーズ	システムニーズ
顧客要求の機会損失減少 在庫維持費減少	リアルタイム性
提供サービス毎による 動的な提携先の変更	柔軟性
常時サービスの提供 / 利用	フォールトトレランス性

表 1: ユーザニーズとシステムニーズ

2.3 課題

初めから予測可能な環境下にあるシステム構築を想定した場合、システムに自律性は必要なく、従来の集中的管理手法によるデータベースシステムで問題は起こらない。しかし先に述べた通り、昨今のように、環境が時々刻々と変化し、かつ予測不可能な状況下において、これらのニーズを満足させる為には、システム内の全ての構成要素が自律性をもち、マスターの存在なしに、自律的に判断できることが不可欠となってくる。

そこで、上述（表1）のシステムニーズを実現させるデータベースシステムとして、自律分散データベースシステム (Autonomous Decentralized Datebase System, ADDS) が考案されている。次章では自律分散データベースシステムについて詳細に説明する。

3 自律分散DBシステム

3.1 コンセプト

自律分散データベースシステム (Autonomous Decentralized Database System, ADDS) は、次のコンセプトを基に考案された分散 DB システムであり、自律したサイトとモバイルエージェントから構成されている。

1. 全てのサイトは自律更新権限を有する
2. いかなるサイトもシステム全体の情報を把握しない
3. 自律したサイトの集合によりシステムを形成する

3.2 アーキテクチャー

図1に自律分散データベースシステムの基本構造を示す。

自律分散データベースにおいて、各サイトは Allowable Volume を保持している。Allowable Volume とは、自サイトの自律的な更新権限を意味する許容量であり、各サイトは、ユーザーから引当要求が来た際、その要求量が自サイトの保持する Allowable Volume の範囲内であれば、他のサイトと協調をはかることなく、サイトは自律的に更新処理行うことができるということを許容する属性である。つまり、各サイトはこの Allowable Volume を保持することで、自律性を保持している（**自律更新技術**）。

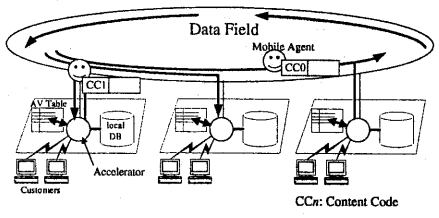

図 1: 自律分散データベースシステム

図1に示す通り、各サイトはこの Allowable Volume の管理テーブルと隣接サイト管理テーブル、ローカルデータベース、アクセラレータと呼ばれる

管理モジュールから構成されている。自律分散データベースにおいて、自サイトの動作の管理・他サイトとの協調は、このアクセラレータにより実現されている。

自律更新技術により、自律分散データベースでは、各サイト間でのデータの疎一貫性を許容している。この点が、従来のデータベースの概念と大きく異なる点である。

各サイトがAllowable Volumeという自律更新権限を持つことで、保持しているAllowable Volume内での更新要求に対してはリアルタイム性が得られ、結果、販売の機会損失は起こらない。しかし、各サイトで必要とするAllowable Volumeが保持されていない、もしくは保持できない場合、当然リアルタイム性は得られず、販売の機会損失という事態を招いてしまう。

一般に大規模システムでは、各サイトの状況はダイナミックに変化する。こういった時事刻々と変化する状況において、サイトは十分なだけのAllowable Volumeを保持している場合もあれば、当然そうでない場合も存在する。つまり、各サイトはあらゆる状況下においてAllowable Volumeの調整を柔軟にできることが求められる。自律分散データベースでは、自律性を持ったモバイルエージェントがサイト間を巡回、各サイトの要求を把握しAllowable Volumeを調整することで、常に変動する状況下でも更新要求に対しリアルタイム性を実現し、機会損失を回避している。MAが、自律的にサイト間を巡回して各サイトと交渉を行い、Allowable Volumeを調整することで、結果的に自律したサイト間で協調がとられている（**自律協調技術**）。
このように各サイトの状況に応じ、絶えず情報の疎一貫性を図れることで、リアルタイム性や柔軟性が実現されているのである。

ここで、MAがどのようにしてサイト間を移動しているかを簡単に説明する。各サイトは、隣接するサイトの局所情報を記録した隣接サイト管理テーブルを保持している。（図1）MAは、サイトを訪れた際、サイトの持つこの管理テーブルを見て次に訪れるサイトを自律的に判断し、サイト間を移動している（モバイルエージェント移動技術）。つまり、サイトもMAもシステム全体のサイトのアドレスといったグローバルな情報を持つことはなく、各サイトは、隣接するサイトのローカルな位置情報のみを保持し、MAはその位置情報に基づいてサイト間を巡回しているのである。

このように、自律分散データベースシステムは、自律性を持つサイトとモバイルエージェントから構成されており、システム内に（サイト間どうしはもちろん、サイト・MA間においても）マスタという概念は存在しない。各サイトは、Allowable Volumeという自律更新権限を持ち、MAがサイト間を自律的に巡回しそのAllowable Volumeをリアルタイムに調整する。こうして、自律分散データベースシステムでは、各サイトの持つ異なるニーズを共存させ、あらゆる状況に柔軟に適応することができるのである。

4 自律障害検知・回復技術

本章では、自律分散データベースシステムにおけるフォールトトレランス技術として、自律障害検知・回復技術を提案する。

4.1 自律障害検知技術

3章で述べた自律分散データベースシステムにおいて、想定される障害としては、

- MAのダウン
- Siteのダウン
- MAとSiteのダウン

ある。しかし、どのような障害が発生した場合であっても、モバイルエージェントはサイト間を巡回し続けることができない。従って、各サイトはMAが自サイトに一定時間来ない場合、タイムアウトで自律的に障害を検知する（この時システム内にどのような障害が発生しているのかは不明である）。

4.2 自律障害回復技術

次に障害を検知したサイトが自律的にその障害を回復させる、自律障害回復技術を提案する。

- MAの障害発生に対する回復技術

タイムアウトで障害を検知したサイトは、モバイルエージェントを再生する。この時、作り出すMAには、自サイトのアドレスをＩＤとして付加する。複数のサイトが障害を検知しMAを作り出した場合、システム内には複数のMAが存在してしまうことになるが、各サイトがMAのＩＤと自サイトのアドレスの大小関係の比較を行い、MAのＩＤが自サイトのアドレスより大きかった場合、そのMAを強制的にダウンさせることで、巡回し続けるMAの数を１つにすることが可能となる。従って、システム内にMAのダウンが発生した場合でも、サイトの自律的判断により、システムを停止させることなくMAを再生することが可能であり、MAのダウンにより、システムダウンという事態は起こらない。

- Site（SiteとMA）の障害発生に対する回復技術

検知した障害がMAのダウンであった場合上述の回復技術で成功となるが、サイトの障害発生時、サイトが何回モバイルエージェントを再生したとしてもMAが巡回してくることはなく、再びタイムアウトとなり障害を検知してしまうことになる。そこで、サイトはある一定回数（ｎ回）MAを再生しても、タイムアウトで障害を検知してしまう場合（サイトの障害発生時）、隣接サイトに対し試験メッセージを送信する。（図 2）その結果、試験メッセージに

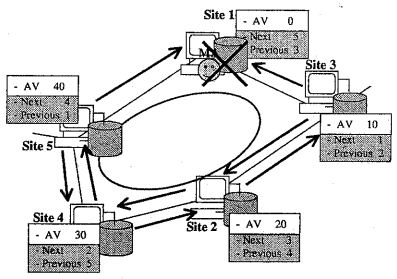

図 2: Site の障害発生時

対する返答があったか否かにより、ダウンしているサイトの隣接サイト（図では、Site3,Site5）は、ダウンサイト（図では、Site1）を検知することができる。従って、次回からは Site3,Site5 は巡回してきたMAをダウンサイトに送り出すことはなく、MAをループバックさせることができ、結果的にシステムは自律的に Site5,Site4,Site2,Site3 で連携をとり、稼動し続けることが可能となる。

つまりこの回復技術により、システム内にダウンしたサイトが存在した場合でも、システムは可能な範囲内で自律的に連携をとり稼動し続けることができる。また、図のようにシステム内で複数のサイト

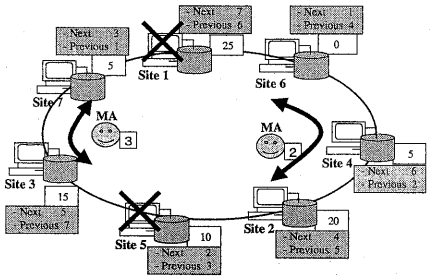

図 3: Site の障害発生時（複数）

が不稼動になり、ネットワーク的に分断された状況になっても、同様にして可能な範囲内で自律的に連携をとり稼動し続けることが可能である。

5 シミュレーション

本研究の有効性を検証すため、シュミレーションを行った。シュミレーションでは、本研究で提案したフォールトトレランス技術を取り入れた自律分散データベースシステム（Autonomous Decentralized DataBase System, ADDS）と、システム内にマスターサイトが存在する従来の集中型分散データベースシステムに対し, 各ノードのダウン確率をパラメタにとり、システム内のノード間の平均連携率を求め、比較を行った。

フォールトトレランスの評価として、分散データベースシステムがサイト数（ｎ）から構成されている時、ノードのダウン確率をパラメタとし（非マスターサイト：α、マスターサイト：β）、システムを構成しているノード間の平均連携率を求め、評価を行った。

このとき、マスターノードのダウン確率は、単独では非マスターノードより１０倍低いものの、自身が管理する非マスターノード数が増えると共にダウン確率は増加するとの仮定 ($\beta = 0.1\alpha * \gamma^{n-1}(\gamma > 1)$) を設けた。

5.1 シュミレーション結果

以下に、シュミレーション結果を示す。横軸はノード数、縦軸は平均ノード間連携率である。

- **Case:1**

非マスターノードのダウン確立：α	0.01
マスターノードのダウン確立：β	$0.001 * \gamma^{n-1}$
γ	1.1

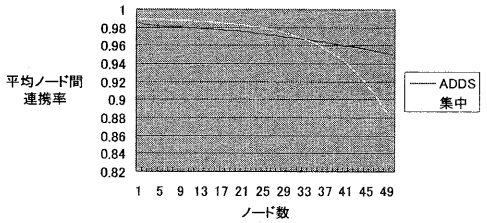

図 4: Simulation Case:1

グラフの交点：ノード数 33

5.2 考察

- ノード数が小さい（システム規模が小さい）場合

従来の集中型分散データベースシステムでも高いノード間連携率が得られる。

- ノード数が大きい（システム規模が大きい）場合

提案フォールトトレランス技術を取り入れた自律分散データベースシステムでは、障害発生時においても各ノードは可能な範囲内で自律的に連携をとり稼動し続けることが可能となる。従って、システム規模が大きい場合でも高いノード間連携率を得ることができ、より高いフォールトトレランスを実現している。

6 まとめ

インターネットに代表される情報通信技術の発達に伴い情報サービスは多様化した。その結果エンドユーザの嗜好も多様化し、企業は関連異業種企業と情報を共有し合うことで熾烈な競争に勝ち残ろうとしている。その一例である SCM にみられる DB 統合においては、リアルタイム性、柔軟性、フォールトトレランス性が時々刻々と変化する状況下で同時に実現されることが求められており、これを実現する DB システムとして、自律分散 DB システムが考案されている。本稿では、この自律分散 DB システムにおけるフォールトトレランス技術に焦点を当て、自律障害検知・回復技術の提案を行なった。この提案技術により ADDS で障害発生時においても、システムを停止させることなく MA を自律的に回復させることができる。また各ノードは可能な範囲内で自律的に連携し続けることができる。従って従来の集中型分散 DB に比べ、システム規模が大きい場合でも高いノード間連携率を得ることができ、高いフォールトトレランス性を実現することが可能となる。

参考文献

[1] K.Mori, et.al:"Autonomous Decentralized Software Structure and Its Application", IEEE, pp.1056-1063, Nov.1986

[2] K.Mori, et.al:"Autonomous Decentralized File System and Its Application", IEEE, Workshop on Future Trends of Distributed Computing Systems, pp.262-268, April, 1992

[3] K.Mori, et.al:"機能信頼度に基づくシステム分割の評価", 測定自動制御学会論文集, Vol.28, No.2, pp.273-280, Feb.1992

[4] K.Mori:"Autonomous Decentralized Systems: Concepts, Data Field Airchtecture and Future Trends", Proc. of Autonomous Decentralized Systems(ISAD 93), IEEE, Kawasaki, Japan, pp.28-34, April, 1993

[5] 花村 英郎:"異質なニーズを満たす分散データベースの自律的一貫性実現技術に関する研究", 修士論文, 東京工業大学大学院 理工学研究科, 2000

[6] 平井 健治:"アシュアランス性実現の為の自律的分散データベース情報一貫性管理技術に関する研究", 修士論文, 東京工業大学大学院 理工学研究科, 2001