

適応型オンラインストレージにおける 切断時データ同期方式

A Method of Synchronizing Disconnected Data in the Adaptive On-line Storage System

井上 知洋 中村 元紀 久保田 稔†

Tomohiro Inoue Motonori Nakamura Minoru Kubota

1 はじめに

ユビキタスネットワーク社会の実現が期待されて久しいが、現在の一般ユーザのネットワーク利用形態は依然として限定されたものにとどまっている。つまり作業用端末として個人のノート PC など特定の端末を携帯し、情報参照やメール送信のためにネットワークへ接続するという利用形態である。このような利用形態の場合、ユーザは固定の端末に縛られておりネットワークがユビキタス化する恩恵を十分に受けられない。我々は、遍在するネットワーク上に自由にデータを保存できるストレージの構築を目指している。本稿では、ユビキタスネットワーク環境における既存のストレージシステムの問題を明らかにし、それらを解決する適応型オンラインストレージの提案および評価を行った。

2 現状の課題

2.1 将来のネットワーク環境

将来のネットワークは、Home Network や Cellular Network, Hot Spot など様々な形態のネットワークが基幹ネットワークを介して連結された、不均一性の強い動的なネットワークになると考えられる。今後、Adhoc Network や PAN (Personal Area Network) など無線リンクで構成されるネットワークが普及するにつれ、ネットワークトポロジの動的な変化や Network Mobility [1] への対応が求められる。この環境ではクライアント端末が単独で移動するだけではなく、主に無線リンクで構成されるネットワークそのものが移動し、他のネットワークとの接続方法が逐次変化する。例えば飛行機内の端末によって構成されるネットワークのように、ローカルネットワークがインターネットから継続的に孤立するようなケースが考えられる。しかし、そのような孤立したネットワーク内においてもユーザはインターネットとの接続時と同様な環境で作業を継続できることが望ましい。

2.2 既存ストレージの課題

分散ファイルシステム

広域ネットワーク環境でも実用的なネットワークファイルシステムの実現のため、AFS や Coda [3] などの分散ファイルシステムが開発されてきた。これらのシステムではクライアント端末上にファイルのキャッシュを配置し高速なファイルアクセスを可能にしている。

特に Coda ではクライアント端末がネットワークから切断された場合にもキャッシュファイルへの更新を行うこと

が可能な切断時操作をサポートしている。この場合、切断時にキャッシュファイル上で更新されたデータはファイルサーバとの接続復帰時にオリジナルファイルへと同期される。Coda は更新が衝突した場合には人手で更新の整合性をとるという楽観的な同期のポリシーを採用している。

想定するネットワークでの課題

既存のストレージシステムの構成を前節で述べたような動的なネットワーク環境に適用することを考える。分散ファイルシステムではファイルサーバが全てのクライアント端末のキャッシュデータを管理し、キャッシュ間の整合性を維持する責任を負っている。このファイルサーバは固定のノードであるため、ストレージを利用している複数のクライアントを含むネットワークがファイルサーバとの接続を失った場合、キャッシュデータ間の一貫性を管理することができない。

Coda のように切断時操作を行うことは可能であるが、この操作はあくまでも各々のクライアント端末単位で行われ、ある端末は自端末以外で行われた更新結果を参照することができない。そのため、図 1 で示すようにそれぞれの端末上のキャッシュデータがお互いに独立に更新されてゆき、バージョンの異なるデータが更新を行った端末の数だけ生まれることになる。結果として、ファイルサーバとの接続が復帰してキャッシュデータの同期が行われたときに更新内容が衝突する可能性が高くなってしまふ。更新の衝突が発生した場合には人手によって整合性を回復する作業が必要となるため、このようなキャッシュファイルのバージョン数の増加は更新衝突回復コストの増大を招き、ストレージとしての可用性を大幅に低下させる。

3 適応型オンラインストレージ

3.1 サーバレスなデータ複製管理

前節の問題は、データキャッシュの一貫性制御の仕組みが固定のファイルサーバに強く依存しているため、たとえクライアント端末同士が孤立したネットワーク内で接続されていたとしてもキャッシュ間の整合性をとることができないことが原因となっている。そこで我々は、孤立したネットワーク内においてもデータ複製間の整合性をとることを可能とする、サーバレスなオンラインストレージシステムを提案している。

3.2 データ管理方式の概要

図 3 に本方式の概要を示す。(1) データを新たに作成した端末に、管理トークンを持つ「コアデータ」が作られる。(2) コアデータ周辺の端末、あるいはコアデータを利用した端末にはキャッシュデータとなる「複製データ」が作ら

† 日本電信電話 (株) NTT 未来ねっと研究所,
NTT Network Innovation Laboratories, NTT Corporation

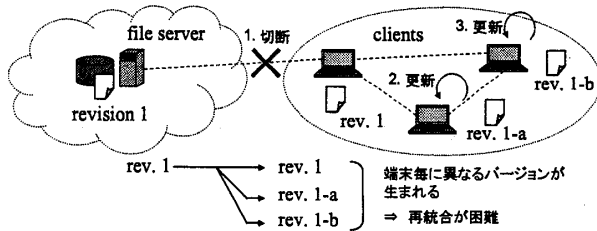


図1 既存分散ファイルシステムの切断時操作

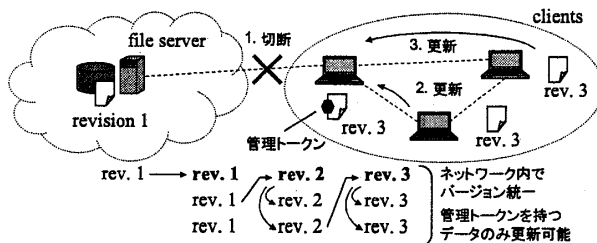


図2 適応型オンラインストレージによる複製データ管理

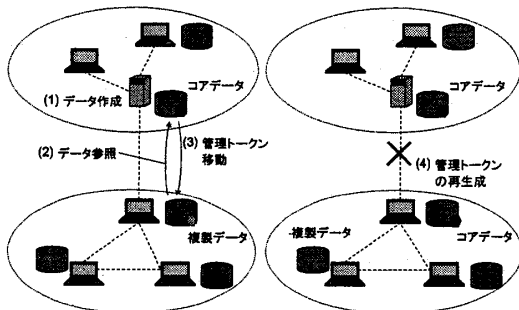


図3 データ管理方式の概要

れる。複製データはデータの参照のみが可能とし、データの更新はオリジナルであるコアデータのみが受け付ける。(3) 管理トークンが移動することにより任意の複製データはコアデータに変わることができる。(4) ネットワーク分断によってコアデータが失われた場合には、いずれかの複製データをコアデータに変更することにより、前述のデータ管理を継続する。

3.3 効果

本データ管理方式では、コアデータにアクセス可能な範囲では常に複製データ間の一貫性制御が可能となる。このためネットワーク内でのデータのバージョンは常に一つとなり、クライアント端末毎に多数のバージョンが発生してしまう問題は解決される。

ネットワークの分断が起きた場合においても、孤立したそれぞれのネットワーク内で複製データ間の整合性は維持される。この結果、同時に存在するデータのバージョン数は最大でも分割されたネットワーク数に限られる。

4 評価

全ネットワーク内の更新回数 N_u に対して、分割されたネットワーク数を $N_n (=2, 3, 4)$ とした場合、実際に生じるバージョン数の期待値を図4に示す。また、 N_n 個のバージョン（それぞれのバージョンは $N_w = N_u / N_n$ 個の更新履歴を持つとする）を再統合する際に必要なコストを $K N_w N_n C_2$

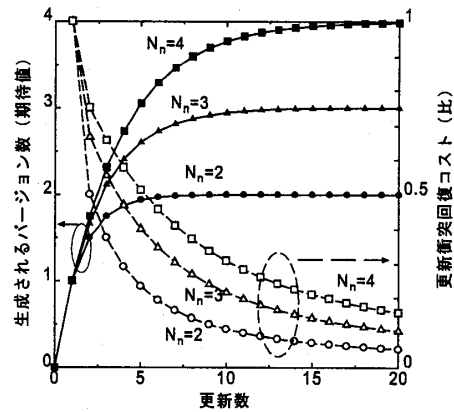


図4 更新回数とバージョン数、衝突回復コストの評価

(K は比例定数) として評価し、本方式を用いた場合と端末単位の切断時操作を行った場合のコストの比をプロットした。分割されたネットワーク数に対し、2 倍以上の回数の更新がある場合更新衝突回復コストは $1/2$ 以下となり、本方式によって更新衝突コスト削減が可能であることがわかる。

5 関連研究

バージョン管理システムの CVS は、広域ネットワーク環境での協調的なファイルの更新や同期を可能としているため、オンラインストレージとしての側面を持つ。ミラーリングによるレポジトリの分散配置も可能であるが、その場合でもオリジナルとなるレポジトリは固定であり、複製レポジトリの生成も手動で行わなければならない。本方式では複製データの自動的な生成を行うことによってストレージとしての利便性を高めている。また、オリジナルとなるデータの位置を適宜変更することが可能なため、トポロジの変化する動的なネットワーク環境での可用性が高い。

P2P 型のファイル共有サービスは広域分散環境でのデータ共有を可能としているが、ほとんどのサービスではファイルの配布を行っているのみであり、広域配布されたデータの更新を行う操作は基本的に考慮されていない。本方式ではサーバレスな構成をとりながらデータ管理を行う端末を明示することによって、容易にデータの更新が可能なストレージシステムを構築できる。

6 おわりに

本稿では、動的なネットワーク環境における既存の分散ファイルシステムの課題を明らかにし、それらを解決する適応型オンラインストレージのデータ管理方式を提案した。その上で、更新衝突回数について既存ストレージと比較し、衝突回復コストが減少することを確認した。

参考文献

- [1] H. Soliman and M. Pettersson. "Mobile networks (monet) problem statement and scope", *Internet Draft, IETF*, February 2002.
- [2] 中村元紀 他, "アドホックネットワーク環境のためのデータ管理方式", 信学総大, 2002.
- [3] <http://www.coda.cs.cmu.edu/>