

大規模分散環境におけるレプリカ探索方式の提案

孫 静涛[†] 田胡和哉[‡]

東京工科大学[†] 東京工科大学[‡]

1 はじめに

近年, クラウドコンピューティングが広く普及している。サーバ上にあるファイルの可用性を高めるため、レプリカファイルの探索とレプリカの配置というレプリケーション管理機構は不可欠となっている。

しかし、レプリカ間のコンシステンシー問題を生じてしまうため、Primary Backup Replication[1]や Active Replication[1]といった各種レプリケーションモデルが提案されてきた。

本稿では、レプリカファイルに「Write once」原則で、レプリカに頻繁な読み書き込みを避ける方式を提案する。本提案のレプリカは通常のファイルの複製と違い、Peer ノード探索を行っていた後に、Peer ノードの履歴情報そのものをレプリカに記録させ、レプリカ探索の効率及びシステムの耐障害性を向上させることを目的としている。

2 関連研究

2.1 Peer-to-Peer の構造

大規模分散環境を構築するため、Peer-to-Peer の構造について、広く研究が行われている。

Peer-to-Peer においては、メッセージの転送方式により、構造化オーバーレイネットワークと非構造化オーバーレイネットワークの2種類に分類されている。非構造化オーバーレイネットワーク方式を用いてファイル共有システムの代表例は Gnutella や Winny などがあり、構造化オーバーレイネットワークにおいては、分散ハッシュテーブル (Distributed Hash Table, DHT) や SkipGraph[2]などがある。

2.2 分散ハッシュテーブル

大規模分散ネットワークアーキテクチャはもちろん、メカニズム上に探索自体の効率化を改善するため Chord[3]、OceanStore [4]、Kademlia[5]など DHT アルゴリズムに関する研究が2002年頃多く発表されている。

しかし、DHT に基づいた方法は、地理的に離れたファイルに対しては不向きである。これらの従来研究に関する検討を行った上で、大規模分散ネットワーク環境において、地理的に離れたファイルに対しても、短時間、高効率、強適応性のレプリケーション管理機構を提案する。

3 提案手法

本研究では、JXSE[6] プロトコルと OpenChord[7] アルゴリズムが用いる。提案方式では、探索した履歴を記録するため、以下の方法をとる。

1. 一回目探索を終わった時点で、Chord にファイル識別子を Key にして、Peer ノードの ID を Value として登録する。登録先は必ず Peer ノードが属する PeerGroupLink である。
2. 二回目から、レプリカバージョン管理システムに問い合わせして、レプリカに保存された Peer ノードの記録から相関ノードの情報を取得し、探索を行う。さらに、3層構造しているオーバーレイネットワーク基盤に沿って、各層ごとに請求されたファイルの探索を行う。

この方法に従って、通常の Peer-to-Peer ネットワークのノード探索手順と違い、探索履歴から探索を行うことでレプリカ探索時間を大幅に減らすことが期待できる。本研究のレプリカはバージョン番号とファイルネームによって、作られたものと定義する。

4 実現方法

本研究では、以下のステップで実装を行っている。

1. JXSE プロトコルと Chord アルゴリズムを基づいた大規模 Peer-to-Peer 構造化オーバーレイネットワークシステムとして構築す

A New Approach Toward Replica Searching for Large-Scale Distributed Systems

[†]「Jingtao Sun・kingxihe27@gmail」

[‡]「Kazuya Tago・ktago.gm@gmail.com」

「Computer Science, Tokyo University of Technology」

る。

2. この JXSE 基盤を3層に階層化する。上から下まで WorldLayer、CountryLayer、SchoolLayer に定義し、各層ごとにファイルの探索を行う。それによって、ファイル探索の時間、効率と適応性が高まっていく。
3. SchoolLayer 層から一気に WorldLayer に通信を行われるため、モバイルエージェント AgentSpace[8]API を使い、通信レイテンシの遅延を下げないように実装する。

4.1 提案方式の構成図

提案方式の構成図を図1に示す。

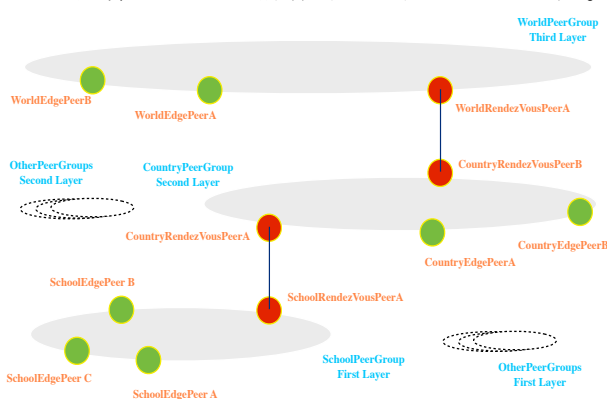


図1 本システム構成図

4.2 提案方式説明

1. ファイルは SchoolLayer 層で探索を始め、JXSE プロトコルによって、ファイルの PeerID 番号とファイルの拡張子を記録し、レプリカとして、保存しておく。
2. 同じファイルを再び探索請求がされた場合に、ファイルの請求側 PeerID とファイルの提供側 PeerID には、パイププロトコルによってパッケージ通信が始まる。

ファイルの請求側に新しいレプリカを作っている場合、ファイルの提供側にも新たなレプリカファイルを作る。旧レプリカファイル内容と新たな追加された内容をまとめて、新しいレプリカバージョンとして保存しておく。

5 試行

本提案方式においては、SchoolEdgeLayerA からファイル請求が訪ねに来ること及び 937kb のファイルを実験対象としてものである。

通信レイテンシの遅延状況を図2に示す。実験結果によって、本提案が通信レイテンシを大幅に下げることがわかる。

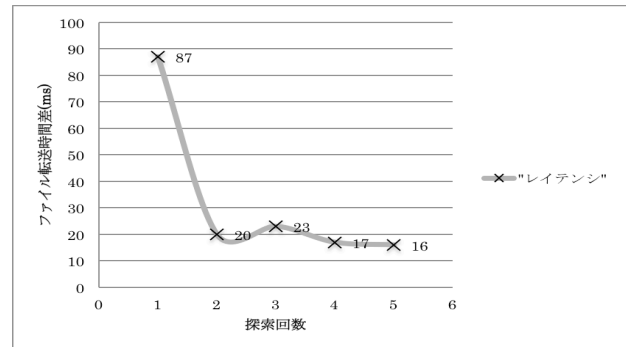


図2 レプリカ探索をかかった時間

6 応用

本学と八王子市内の数社と連携で作った「インテリジェント・エコターミナル」(以下 IeT) という太陽光発電サーバに本レプリカ探索方式を実装する予定である。大災害が発生後、クラウドサーバ上に保存されているデータの損害を最小限にするため、本方式の効果が期待されている。

7 まとめ

分散システム上で、効率の良いレプリカ探索を利用したファイル共有システムについて提案した。これから、本システムの使用により、レプリカに保っているデータがさらに巨大化になることが考えられる。今後、それを解決するためにグーグルの Spanner のようにウィクショナリーデータ保存手法に加え、実装と評価実験を行う予定である。

参考文献

- [1]GeorgeCoulouris, JeanDollimore, TimKindberg, andGerdonBlair. DistributedSystemsConcepts and Design, FifthEdition. Addison Wesley. 2012
- [2]James Aspnes, Gauri Shah. Skip Graphs. Volume3Issue4, November2007. ArticleNo. 37. ACM
- [3]IonStoica, RobertMorris, DavidKarger, M. Frans Kaashoek, ect.Chord:A Scalable Peer-to-PeerLookupServiceforInternetApplications. October2001, ACM
- [4]<http://oceanstore.cs.berkeley.edu/>
- [5]Petar Maymounkov, ect Kademlia:A Peer-to-PeerInformationSystemBasedontheXORMetric. Peer-to-PeerSystems, 2002-Springer.
- [6]<http://jxse.kenai.com/>
- [7]<http://open-chord.sourceforge.net/>
- [8]<http://research.nii.ac.jp/~ichiro/agent/agentspace.html>