大規模仮想ディスクを用いたWebストレージ

チャイ エリアント　　上原 稔

東洋大学大学院工学研究科情報システム専攻

今日、大量のデータが氾濫している。そのデータを蓄積することが1つの課題になっている。現在の普及品の HDD は安価であるにもかかわらず、ストレージシステムは高価である。そこで、我々は安価な普及品とソフトウェアだけで大規模ストレージを構築するために VLSD(Virtual Large-Scale Disk)ツールキットを開発した。また、これを用いて PC の空き容量を集積し、大規模ストレージを試作した。従来の VLSD は組織内で大規模ストレージを構築することを目的としていた。しかし、より大きなストレージを構築するには1つの組織では困難である。そこで、インターネットを経由して大規模ストレージを構築するために、Web ストレージを開発した。本論文では、試作した Web ストレージの実装と評価について述べる。

A Web Storage using Virtual Large-Scale Disks

Erianto Chai　　Minoru Uehara

Toyo University Graduate School, Dept. of Open Information System

Today, massive data is flooding. One of major issues is to store massive data. Recently, although commodity HDD is very cheep, appliance storage system is very expensive. So, we developed VLSD (Virtual Large-Scale Disk) toolkit in order to construct a large-scale storage using only cheep commodity hardware and software. And we also developed a prototype of large-scale storage system by collecting free disk spaces of PCs using VLSD. Conventional VLSD aims to develop a large-scale storage in local network. However, it is hard to develop larger scale storage than conventional storage if only local resources are available. So, we have developed a prototype of Web storage in order to realize larger scale storage via Internet. In this paper, we describe the Web storage and evaluate it.

1　はじめに

近年、ストレージサービスに対する要求が高まっている。

従来のアプリケーションでは、データを共有するためにデータベースを用いていた。しかし、近年 Web サービスが普及するにつれ、SOA ベースのアプリケーションが増えてきた。これらのアプリケーションでは、異なるベンダーから提供されるサービスを組み合わせるマッシュアップが行われる。このようなアプリケーションでは、インターフェースを Web サービスで統合するため、ストレージサービスを利用する必要がある。ストレージの Web サービスは Amazon などが提供している。

YouTube やニコニコ動画などの動画投稿サイトはマスコミュニケーションを補完するジャーナリズムとして定着すると予想される。USA では、大統領選にも利用されている。このようなサイトでは、多くの動画を保管するために大容量のストレージを必要としている。

ライフログでは、人のあらゆる活動を記録する。そのためには一人当たり数 TB 以上の容量を必要とする。

企業は内部統制のためにログを保管する必要がある。企業の所有する全 PC から集められたログは膨大である。これを効率的に管理するため、情報ライフサイクル管理に基づく 3 階層ストレージが使用される。

このようなストレージサービスに対する要求は HDD 技術の進歩を促した。その結果、安価で大容量なディスクが入手可能となった。しかし、アプライアンス系のファイルサーバで使用している HDD はその普及品より 10 倍高価である。このように現在のストレージのコストは適切でない。ストレージが高価な理由は専用ハードウェアにある。普及品のハードウェアと専用ソフトウェアによって問題を解決できる。

我々は、大規模ストレージを構築するためのツールキット VLSD(Virtual Large-Scale Disk)を開発した[1][2]。VLSD は 100 % pure Java であるため、プラットフォームに依存しない。我々は VLSD を用いて 500 台の PC からそれぞれ空き容量 170GB を集め 70TB のストレージを構築するシステムを試作した。このシステムでは RAID66(2 階層の RAID6)を用いて十分な MTTF を実現している。

試作した 70TB ストレージを実用的に運用するにはセキュリティが課題となる。我々は、VLSD にセキュリティ機能を追加した[3]。

従来の VLSD は組織内で運用することを前提としていた。しかし、組織内ではストレージ容量も限られる。ストレージを効果的に活用するには組織間で共有することも1つの選択肢である。また、組織内で運用する場合にも、モバイルで外部からアクセスしたいことがある。このような場合は、いずれも FW を超えたアクセスが必要になる。現在のところ FW を超えることが可能な唯一のプロトコルは HTTP と考えられる。そこで、HTTP によるストレージアクセスを実現する必要がある。我々は、HTTP でアクセス可能なストレージを Web ストレージと呼び、これを実現する機能を VLSD に追加する。

本論文では、VLSD を用いた Web ストレージの実現法とその性能を評価し、問題点を明確にする。

本論文の構成は以下の通りである。第 2 章では関連研究について述べる。第 3 章では VLSD について述べる。第 4 章では VLSD における Web ストレージについて述べる。第 5 章では評価する。最後に結論を述べる。

2 関連研究

2.1 オンラインストレージ

ここでは、オンラインストレージサービスについてまとめる。

Wikipedia では、オンラインストレージを「サービスビューローのサーバの空き領域をインターネット経由で貸し出し、ユーザがそこにデータを保存できるようにするオンラインサービスのこと」と定義している。

この種のサービスでは、Web 画面でアップロードとダウンロードを行う。ダウンロードではファイル共有のための URL を生成する。この種のサービスでは、アプリケーションから直接ファイルをアクセスすることはできない。また、1回でアップロードできるファイルサイズやファイル数、全ファイルのトータルサイズなどに上限があることが多い。また、無料の場合、保存期間、ダウンロード回数などにも制限がある場合がある。多くは広告を主な収入源としている。

MyYahoo ブリーフケースは、無料で利用可能なオンラインストレージサービスである。トータルサイズ 300MB、ファイルサイズは 5MB である。これらの容量制限により現実的な利用には適さない。フォルダは階層的に作成できる。

apidshare.com(旧 rapidshare.de)は無料で利用可能なオンラインストレージサービスである。ただし、有料のサービスもある。無料版では、総容量 1000 テラバイト、ファイル数制限なし、ファイルサイズ 100MB、トータルサイズ制限なしで利用できる。また、無料版ではファイルは 45 日間しか維持されないが、有料版では永遠に維持される。なお、無料版ではダウンロードが 90 分に 1 回に制限される。有料版では、ダウンロードを並行して実施することができる。

Yousendit は無料で利用可能なオンラインストレージ及びメールサービスである。無料で 100MB、有料で 2GB までのファイルを転送することができる。ファイルは無料版で 7 日間、有料版で 14 日間維持される。無料で 100 回、有料で 200 回～500 回ダウンロードすることができる。有料版では、複数のファイルを同時に送信することができたり、送信したファイルを自動的にバックアップしたり、送信したファイルにパスワードをかけることができたりする。

megaupload.com は無料で利用可能なオンラインストレージサービスである。ファイルサイズ 500MB、トータルサイズ 50GB である。ファイルは 90 日間維持される。

MediaFire(http://mediafire.com/)は、無料で利用可能なオンラインストレージサービスである。完全無料で、ファイルサイズ 100MB、アップロードも無制限、登録も不要である。複数ファイルの同時アップロード、ダウンロードにも対応している。ファイルもフォルダも日本語に対応している。

PipeBytes(http://www.pipebytes.com/)は、無料で利用可能なオンラインストレージサービスであるが、他のオンラインストレージサービスと異なり、一時的な利用に特化している。具体的には、アップロードファイルの容量を無制限にし、1回のみダウンロード可能としている。

WebDAV を用いて比較的容易にオンラインストレージを構築できる。WebDAV は Windows で仮想的なフォルダとしてサポートされているため、ドラッグ＆ドロップで直感的に操作できる。しかし、基本的にはダウンロードとアップロードで実現されるため、大容量ファイルの操作には向かない。また、サーバにおけるファイルのアクセス権が限られるため、同じフォルダのファイルを異なるユーザで排他的に共有できない。

FreeDAV(http://www.freedav.com/) は WebDAV で構築されたオンラインストレージサービスである。WebDAV の欠点を補うため外部プログラムによりサービスを実現している。無料で 100MB まで利用できる。他のサービスに比べると最大容量が小さい。また、WebDAV の原理から個々のファイルも比較的小さいものに向いている。

Amazon S3 は、Amazon Web Service(AWS)の Simple Storage Service(S3)である。最大ファイルサイズは 5GB、ファイルの容量に応じて以下のように課金される。

$0.15/GB/month(storage)
$0.20/GB(transfer)

AWS の他サービス（例えば EC2）との間の転送には課金されないので、AWS で統合されたシステムでは安価なストレージを構築できる。例

えば、70TBのストレージを実現するために必要な年間経常経費は$0.15*70000*12=$126000≒1千万円となる。この試算によれば VLSD の導入経費は S3 の経常経費より安い。

S3 には SOAP と REST の 2 種類の API がある。REST では Range GET に対応している。Range ヘッダーを指定すれば、ファイルの任意のブロックを読み取ることができる。これによりダウンロードの中断および再開を制御できる。ただし、PUT では部分書き換えできない。

オンラインストレージは転送単位によってブロック転送方式とファイル転送方式に分類される。それによって使い勝手が大きく変わる。一般的に、ブロック転送方式はファイルの一部であるブロック単位で転送できるが、オンラインでなければ使えない。ファイル転送方式は、オフラインでも使える可能性があり、複数の修正をまとめて更新する。しかし、変更しない箇所もダウンロードする必要がある。ファイル転送方式はファイルのサイズが大きくなると実用的でなくなる。ダウンロードに要する時間が長くなり応答性が著しく低下する。巨大ファイルを扱う大規模ストレージではファイル転送方式は使えない。既存の Web ベースのオンラインストレージサービスは基本的にファイル転送方式である。

研究レベルでは、ブロック転送方式を実現したオンラインストレージも存在する。文献[4]では、HTTP-FUSE-cloop を読み書き可能に拡張し、Web ストレージを実現している。オーバーヘッドも小さい。汎用性を優先したためか、Range アクセスを用いていないため、ディスクはブロックごとに分割され、サーバ上に置かれる。そのため、必ずしも管理が容易とはいえない。

2.2 Web サービス

ここでは、オンラインストレージを実現するための実装技術として、Web サービスについてまとめる。

典型的な Web サービスは SOAP, WSDL, UDDI などで実現される。小規模な Web サービスは SOAP だけで開発されることもある。SOAP のメッセージは XML 文書であるので、SOAP は XML RPC の一種と言える。

一般的に SOAP の処理は重い。XML への変換など形式的な手続きに費やされるからである。そこで、REST(Representational State Transfer)と呼ばれる手法が注目されている。SOAP は XML RPC であるから、SOAP による Web サービスはメソッド中心に設計される。それに対して、REST は資源を中心に設計され、各資源が持つメソッドは HTTP のメソッドと直接的に対応する。適切な資源の概念を選択すれば、REST の方が SOAP より容易に Web サービスを実現できる。

Web サービスに基づくオンラインストレージには Amazon S3 などがある。Amazon S3 は SOAP と REST の 2 種類の API をサポートしている。

3 VLSD

VLSD[3]は大規模ストレージを構築するための 100% pure Java ツールキットである。ここでは、VLSD[3]およびそれを用いたストレージの概要について述べる。

3.1 VLSD のクラス

ここで、VLSD ツールキットのクラスについて説明する。

- NBDServer

NBD サーバのクラス。NBD サーバはクライアントで動作し、空き容量を仮想ディスクファイルとしてストレージシステムに提供する。クライアントの OS は Windows または Linux である。NBD サーバは Java で実装されているためプラットフォームに依存しない。Linux と Windows の両方で動作する。また、クライアントには複数のディスクが接続されていたり、FAT32 が使われていたりすることがある。FAT32 ではそのサイズが 2GB 以上のファイルを作成できない。これらのような場合、120GB の仮想ディスクを単一ファイルとして作成することはできないので、後述の RAID0 または JBOD と組み合わせることで複数のファイルを束ねて仮想ディスクを実現することができる。

- DiskServer

ディスクサーバのインターフェースである。

- DiskServerImpl

ディスクサーバのインプリメンテーションであって、RMI による遠隔ディスクを提供する。

- Disk

仮想ディスクが備えるべきインターフェースである。

- AbstractDisk

抽象的な仮想ディスクのクラスであって、下位クラスで使う定数やメソッドを定める。

- DiskArray

複数ディスクからなるディスク・ラッパーの基底クラスで、簡単な RAID1 を実装している。

- RAID

RAID の基底クラスで、簡単な RAID1 の実装を DiskArray から引き継いでいる。

- SingleDisk

単一ディスクからなるディスク・ラッパーの基底クラスである。

- PagedDisk

ページ単位でアクセスするディスクで、任意のディスクのラッパーとなる。ラップされたデ

ィスクはページ単位でしか read/write されない。ページ単位の端数は無視される。

- VariableDisk

可変容量ディスクであって、事前に資源を割り当てず、必要に応じて動的に資源を確保する。作成時に指定したサイズを超えて資源を使用することはないが、論理的なディスクサイズより大きくなることがある。

- RemoteDisk

ディスクサーバをアクセスする遠隔ディスクである。

- RAID0

RAID0 仮想ディスクのクラス。RAID0 は容量を増やすために使われる。後述の JBOD とはストライピングを行う点が異なる。若干性能はよいが、容量は最小サイズのディスクに合わされる。例えば、100GB, 120GB, 160GB を連結しても 100GB×3 にしかならない。純粋に容量増を目的とする場合 JBOD を用いたほうがよい。逆に、RAID0 は性能向上が期待できる場合がある。一部のファイルシステムでは i-node を管理するスーパーブロックが集中して配置される。このようなファイルシステムでは、規模が大きくなると特定のディスクにアクセスが集中する。このような場合、ストライピングは負荷を分散する効果がある。RAID0 はバイト単位でストライピングする。

- RAID5

RAID5 仮想ディスクのクラス。パリティを各ディスクに分散して格納する。パリティを格納するディスクはブロックごとに異なる。

- RAID6

RAID6 の実装で、GF テーブルの生成、ブロック単位ストライピングと分散パリティが行われる。同時に 2 つまでの故障に耐える。本実装では P+Q 方式を採用している。最大 256 台のディスクを用いて RAID6 を構成できる。

- JBOD

JBOD 仮想ディスクのクラス。RAID0 と同様に冗長性がなく、容量増のために用いられる。ストライピングを行わないため容量は単純に総和となる。例えば、100GB, 120GB, 160GB を連結すると 380GB になる。RAID0 には負荷分散の効果があると述べたが、JBOD には負荷を集中させる効果がある。ある程度の規模まではキャッシュが有効に働くため、RAID0 より性能がよくなる可能性がある。

3.2 VLSD を用いた大規模ストレージの試作

VLSD は大規模ストレージ構築のためのツールキットであり、Java によるソフトウェア RAID 実装と NBD 実装を含む。VLSD は 100% pure Java であり、Java が動作するプラットフォームの上なら VLSD も動作する。そのため Windows や Linux が混在する環境に適している。

VLSD を用いると OS に制約されることなく NBD デバイスと RAID を自由に組み合わせることができる。最低限必要な NBD デバイスはファイルサーバの 1 つである。

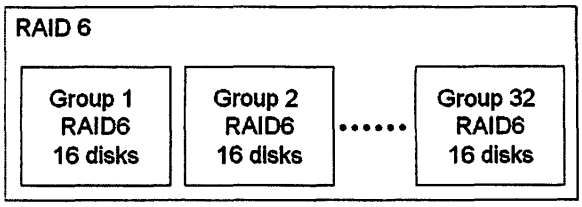

図 1　RAID66 の構成図

本研究ではディスクレベルで空き容量を連結して 1 つの 70TB ストレージを試作した。システムはディスクレベルなので、部分ディスクサイズを越えるファイルの保存も可能である。本研究は 512 台のディスク（1 ディスク=170GB）を 32 グループにして RAID66 を構築する(図 1 に示す)。

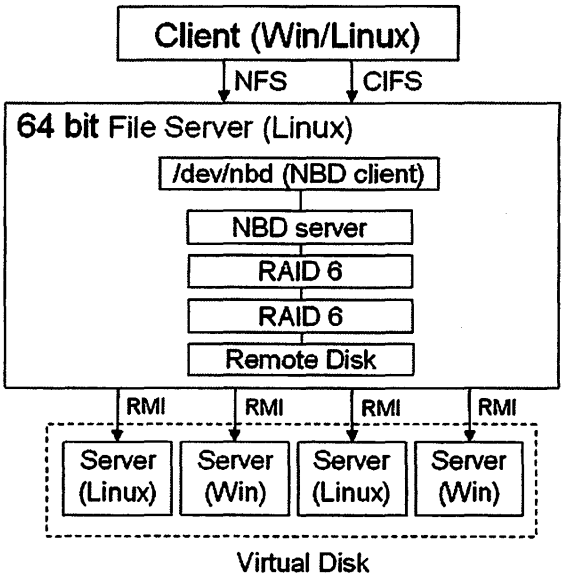

図 2　システム構成図

図 2 に示すように本システムは 64 ビットファイルサーバとディスクサーバがある。ディスクサーバの Linux や Windows などの OS からなる仮想ディスクはディスクの読み込みや書き込みを Java の RMI で機能を提供する。ファイルサーバの方は用意されたディスクに接続して RAID66 を構築する。RAID66 とは、2 階層の RAID6 である。NBD Server はその RAID66 を利用して NBD Client からのアクセスを待つ。そして、NBD Client の起動をした後、XFS でフォーマットする。Windows のクライアントは Samba を介してそのファイルサーバをアクセスする（Linux の場合は NFS）。

NBD プロトコルはセキュリティに欠けるため、ネットワーク上で運用するのは危険である。しかし、我々の方式では 1 台のサーバ内のプロセス間通信として NBD を用いているため安全に運用できる。実際の C/S 間通信はセキュリティを考慮した RMI に基づくプロトコルで実現される。

4 Web ストレージ

ここでは、我々が開発した Web ストレージの概要及び仕様について述べる。

4.1 概要

Web ストレージのシステム概要について述べる。本論文における Web ストレージとは、HTTP でブロック単位にアクセス可能なネットワークディスクである。図 3 に概要を示す。

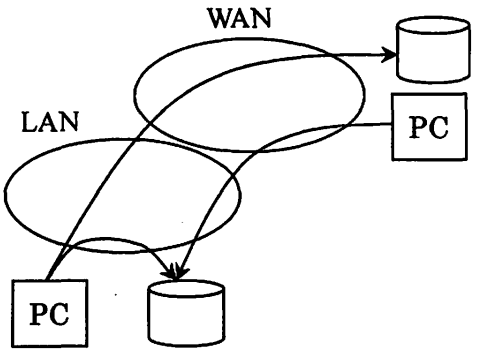

図 3　Web ストレージ

従来の VLSD は LAN 内の資源を有効に活用するために NBD および RMI を用いて NBDServer および RemoteDisk を実現した。しかし、これらは独自のプロトコル(NBD, RMI)を用いるため、FW を超えるアクセスは困難である。一組織のストレージ資源は限られているが、複数組織のストレージ資源を束ねることができれば、より効率的な利用が可能となる。それには Web ストレージが必要となる。

また、モバイルなど外部 PC から組織内 LAN のストレージにアクセスしたいこともある。そのような用途にも Web ストレージが必要である。

一般的に外部から FW を超えて内部にアクセスするには SSH が使われる。あるいは Web サーバを経由する場合、HTTP が使われる。逆に内部から FW を超えて外部にアクセスするときには HTTP しか許されないことがある。言い換えれば HTTP だけは許さざるを得ない。それゆえ、HTTP でアクセス可能なオンラインストレージ Web サービスが必要となる。

我々の Web ストレージはオンラインストレージの Web サービスである。その意味では S3 に近い。しかし、読み書きともブロックレベルのアクセスを許す。また、VLSD は完全にユーザレベルで動作するため、文献[4]のように OS にカーネルレベルの変更を加える必要もない。ただし、VLSD へトラップするために軽量 OS を仮想マシンで動作させる。この方式は VLSD に依存しない。VLSD と IntegraTUM WebDisk(Java 版 Samba)などを組み合わせることも可能である。しかし、現状では WebDisk は VLSD に対応していない。また、その場合には Java によるファイルシステム層が必要となる。

4.2 SOAP による Web ストレージ

我々は、まず SOAP に基づく Web サービスとして Web ストレージを実現した。ここでは、SOAP ベースの API について述べる。この API では、VLSD の Disk インターフェースに近い仕様を定めた。

public byte[] read(long pos, int len)
　pos の位置から len バイトを読み取り、実際に読み取ったバイト分の配列を返す。

public void write(long pos, byte[] b)
　pos の位置から b の内容をすべて書き込む。サイズを超えたらエラーとなる。

次に、このサービスを使うクラスについて述べる。

SOAPDiskServer
　このクラスは Web ディスクのサーバとなる。RemoteDisk に対する DiskServer に該当する。データは XML に変換されて送信される。

SOAPDisk
　このクラスは Web ディスクのクライアントとなる。RemoteDisk に該当する。データは XML に変換されて送信される。利用法は他の VSLD ディスクと変わらない。

これらのクラスでは WSO2WSAS[5]というフレームワークを用いてスタブを生成した。

4.3 REST による Web ストレージ

我々は、次に REST に基づく Web サービスとして Web ストレージを実現した。

GET url?pos=P&len=L
　パラメータ pos、len を伴い GET でアクセスするとディスクの pos から len[Byte]までの範囲のバイナリデータを返す。

PUT url?pos=P&len=L
　パラメータ pos、len を伴い PUT でアクセスするとディスクの pos から len[Byte]までの範囲に本文のバイナリデータを書き込む。本文のサイズと len が異なるときには本文のサイズを用いる。

次に、このサービスを使うクラスについて述べる。

WebDiskServer

このクラスは Web ディスクのサーバとなる。RemoteDisk に対する DiskServer に該当する。データはバイナリのまま変換されずに送信される。

WebDisk

このクラスは Web ディスクのクライアントとなる。RemoteDisk に該当する。データはバイナリのまま変換されずに送信される。利用法は他の VSLD ディスクと変わらない。

5 評価

ここでは、Web ストレージの性能を評価する。

5.1 SOAP による Web ストレージの評価

ここでは、SOAP による Web ストレージの性能を評価する。同容量の FileDisk との相対性能によって評価する。FileDisk は VLSD の中で最も高速な仮想ディスクである。

本評価は、AMD Athlon(tm) 64 X2 Dual Core 3800+、メモリ 4GB、Windows XP Professional x64 で行われた。

1KB のデータを読み書きした評価結果を表 1 に示す。この結果から SOAP 方式は非常に性能が悪いことが分かる。その理由はバイナリデータを XML に変換するためである。無駄なフォーマット変換がおこなわれている。

表 1 SOAP による Web ストレージの性能評価

Disk	Small Read[ms]	Small Write[ms]
FileDisk	0.017925	0.0824
SOAPDisk	22.658	30.782
相対性能	0.08%	0.27%

5.2 REST による Web ストレージの評価

ここでは、REST による Web ストレージの性能を評価する。同容量の FileDisk との相対性能によって評価する。FileDisk は VLSD の中で最も高速な仮想ディスクである。

本評価は、AMD Athlon(tm) 64 X2 Dual Core 3800+、メモリ 4GB、Windows XP Professional x64 で行われた。

1KB のデータを読み書きした評価結果を表 2 に示す。この結果から REST 方式の性能は FileDisk とそん色ないことが分かる。その理由はバイナリデータを直接送受信するため、無駄なフォーマット変換がおこなわれないからである。

表 2 REST による Web ストレージの性能評価

Disk	Small Read[ms]	Small Write[ms]
FileDisk	0.017925	0.0824
RESTDisk	8.844	2.936
相対性能	0.20%	2.81%

6 認証

WebDisk を安全に使うためにはユーザ認証が必要である。クライアントがディスクサーバにログインするために以下のヘッダーをディスクデータと共に送る。

【ユーザ ID の長さ】【デジタル署名の長さ】【ユーザ ID】【デジタル署名】【ディスクデータ】

デジタル署名はディスクサーバの秘密鍵から生成される。認証はデジタル署名と公開鍵によって判断される。

7 まとめ

本論文では、VLSD を用いて Web ストレージを構築する方法について述べた。我々の Web ストレージは HTTP によりアクセス可能な遠隔ブロックデバイスである。これにより FW の内外から大規模ストレージをアクセス可能となる。我々は SOAP と REST の 2 方式で実装し、その評価を行った。Web ストレージには REST が適するとの結論を得た。

謝辞

本研究は科研費基盤 (C)「PC グリッドによる高信頼・高効率な分散仮想ストレージの研究 (19500066)」により援助されています。

参考文献

[1] Chai Erianto, Minoru Uehara, Hideki Mori, Nobuyoshi Sato: "Virtual Large-Scale Disk System for PC-Room", LNCS 4658, Network-Based Information Systems, pp.476-485, (2007.9.3-4)

[2] Chai Erianto, Minoru Uehara, Hideki Mori: "Performance Evaluation at Failure in a Large-Scale Virtual Disk", DPSWS2007 (2007.10)

[3] Minoru Uehara: "Security Framework in a Virtual Large-Scale Disk System", In Proc. of IEEE 10th International Workshop on Multimedia Network Systems and Applications (MNSA2008), pp.30-35, (2008.6.20)

[4] 野尻祐亮, 並木美太郎: "ユビキタス計算機環境のためのオンラインストレージによる分散仮想ディスクシステムの開発", 情報処理学会第 69 回全国大会, 6K-2, pp.133-134, (2007.3)

[5] Web Services Application Server by WSO2, http://wso2.com/products/wsas/