社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－DSM－44 (1)
2007／3／9

SRBを利用した分散ファイルシステムの構築

飯田好美，佐々木節，鈴木聡，八代茂夫
高エネルギー加速器研究機構 計算科学センター

概要

高エネルギー加速器研究機構(KEK)で行われている実験では世界中にある共同研究機関とデータを共有する必要がある。多量の実験データをKEKから各機関に配布するのみではなく、モンテカルロ事象を分担して生成し、共有している。しかし、各機関で使用しているストレージの種類やファイルシステムは異なっているため共同研究者達は煩雑なアカウントの管理やファイルの管理をしなければならない。そこで、多種多様なストレージへの単一的なインターフェイスと論理的な名前空間を提供する論理分散ファイルであるSRBを導入してデータ共有の簡素化を図る。

Construction of distributed file system with SRB

Yoshimi Iida, Takashi Sasaki, Satoshi Suzuki and Shigeo Yashiro
Computing Research Center, High Energy Accelerator Research Organization

Abstract

In the experiments at High Energy Accelerator Research Organization (KEK), large amount of data are shared by the collaborating institutes world wide. However, because of differences of the file systems used and the account management systems at each institute, it is not easy to integrate them.

We introduced the SDSC Storage Resource Broker (SRB) that is a logical distributed file system to provide a uniform interface for connecting heterogeneous data storage resources and providing a single global logical namespace for simplification of data sharing.

1. はじめに

高エネルギー加速器研究機構(KEK)で行われている実験のデータなどはファイアウォールで守られたネットワーク内のストレージシステムに保存されている。世界中にいる共同研究者たちはそのデータにアクセスするためにDMZを経由して内部ネットワークに接続し、それぞれの研究機関でデータの解析を行う。解析されたデータはそれぞれの機関のストレージに保管されるため、それを利用する共同研究者は今度はそこのシステムにアクセスする必要がある。そのため、共同研究者たちはお互いのシステムにアクセスするためのアカウントやパスワードをいくつも持ち、どこの機関のどのシステムにどのようなデータがあるかを把握しなければならない。また、システムを管理する側も共同研究者の数だけアカウントを管理する必要があり、アカウン

トが増えればその分パスワードの不正取得などの危険も増える。

ここでは、これらのデータ共有の煩雑さを解消するために、The SDSC Storage Resource Broker (SRB)[1]による論理的分散システムの構築を行い、その成果を報告する。

2. SRB 分散ファイルシステム

SRB は San Diego Supercomputer Center(SDSC)で開発されたグローバル論理名とファイル階層をユーザに提供する論理的分散ファイルシステムである。これはネットワークに接続された多種多様なデータストレージに対して一様なインターフェイスを提供するミドルウェアで、複数のサイトに分散されたデータファイルを一つのツリー構造として管理することができる。SRB が扱うストレージリソースは NFS や RAID のような Unix ファイルシステムはもちろん、Oracle や DB2、Illustra のようなデータベースシステム、HPSS(High Performance Storage System)[2]のような HSM システム、テープライブラリなどである。また、ユーザインターフェイスとしては Unix コマンドに似たコマンドライン、C 言語や Java の API、Windows 用の GUI、Web アプリケーションなどが用意されている。

SRB は Metadata CATalog(MCAT)サービス、SRB サーバ、SRB クライアント(アプリケーション)の 3 つの要素から構成されており(図 1)、それぞれがネットワークで接続されている。

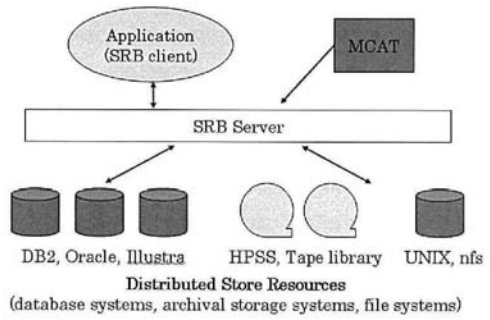

図 1 SRB アーキテクチャ

MCAT には SRB で管理されているユーザやリソースなどに関するメタデータが保管されており、SRB サーバからの問い合わせに応じて論理名と物理的属性のマッピングやメタデータの作成、更新を行う。クライアントには SRB サーバへ要求を行ったり、SRB サーバからの返答を受けたりするための API セットが提供されている。SRB サーバはクライアントの要求に応じて、MCAT と通信し、クライアントの代わりに様々なタイプのストレージシステムへの入出力を行う。

SRB は並列転送をサポートしており、ファイルサイズの大きなデータや高遅延環境でのデータ転送時にとても有効である。また、ファイルサイズの小さなデータを多く転送する際、バルク転送やコンテナーを使うことで複数のファイルを tarball のようにまとめて転送できるため、HPSS などのように小さなファイルの書き込みに適さないストレージにも効率ファイルを書き込むことが出来る。また、すでにストレージに存在する SRB に登録されていないデータファイルに関しても、ファイルをコピーすることなく SRB から扱えるようにメタデータを登録することが可能である。また SRB では、MCAT を連携し、相互に参照することで Zone 間でのデータやストレージの共有が可能になる。Zone とは 1 つの MCAT に管理されている SRB サーバのまとまりのことである。

3. KEK での SRB テスト環境

KEK では、SRB 導入のテスト環境として SRB サーバを 3 台、SRB クライアントを 1 台設置し、SRB リソースとして 3 種類のリソースを登録した(図 2)。

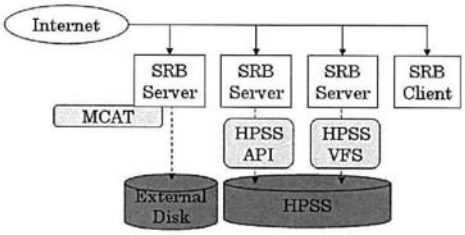

図 2 SRB 構成図

SRB サーバのうち 1 台には PostgreSQL を

ベースとした MCAT をインストールし、Fiber-Channel で接続された 1TB のディスクを SRB のリソースとして登録した。もう 1 台の SRB サーバには SRB が直接 HPSS クライアント API ライブラリを使う方法で HPSS エリアを SRB のリソースとして登録した。最後の 1 台には 2006 年の春から KEK で導入した新しい HPSS への接続方法、VFS サービスを利用したリソースとして SRB に登録した。

4. HPSS API を使用した接続

KEK では、米国 DoE 関連の研究機関が共同して開発し、IBM 社がサポートしている、高速転送をサポートした大容量ファイルシステムである HPSS を利用している。HPSS は、pftp コマンド、分散ファイルシステムによるアクセスに加え、クライアント API を備え、ユーザのソフトウェアとの親和性を高めている。

SRB がサポートしている HPSS クライアント API を使用しての接続には、SRB のコンパイル時にファイルの編集を行う必要がある。その際、HPSS のクライアント(つまり SRB サーバ)が HPSS へ接続するときに使用する認証方法を選ばなければならない。SRB がサポートしていた認証方法は、DCE(Distributed Computing Environment)[3]認証と NO_DCE 認証の 2 通りだけであった。DCE のセキュリティ・サービスを使用するとユーザとパスワードを中央管理でき、クライアントは暗号化されたチケットを取得し、それを使用して認証を受けることができる。NO_DCE 認証では Kerberos ライブラリや Solaris 用に開発された DCEless ライブラリを使用することになり、厳密には認証が行われない。

KEK では SRB サーバとして Linux を使用し、HPSS の認証方法として Kerberos 認証を使用していた。また、KEK の DCE サーバは Kerberos 認証と DCE 認証の両方を認識でき、HPSS クライアントから来た Kerberos 認証を DCE 認証に変換して HPSS へ接続していたため、HPSS からは DCE 認証で接続に来ていると認識していた。そのため、SRB のソースコードを含むいくつかのファイルの編集を行う必要があった。

SRB を最初に導入した時点では HPSS へ Linux から接続するという設定は全くなかったため、修正する箇所も非常に多かった。しかし、修正などを行うたびに SRB の開発元である SDSC にそのコードを伝えていたため、バージョンアップのたびにソースコードが改良され、現在のバージョンでは Linux からの接続のためのパラメータなどが加えられており、修正が必要な点は KEK の特殊な認証方式の部分のみになった。

しかし、KEK では 2006 年の春に新システムへ入れ替えを行なった際、HPSS を最新のバージョン 6.2 に変更した。SRB では HPSS のサポートはバージョン 5.x までになっており、バージョン 6.2 とはメタデータ管理とトランザクション監視、Low-level API、メタデータ構造体などが変わっている。そのため、そのままでは HPSS 上のファイルの状態を正しく取得することが出来なかった。そこで、今回はその部分についてもソースコードの修正を行った。

まず、最初に行ったのは mk.config ファイルの編集である。このファイルは SRB が HPSS 接続をサポートするために必ず編集しなければいけないファイルである。通常はこのファイル中でコメントアウトされた HPSS パラメータと使用する認証のパラメータを有効にし、必要なライブラリがあるディレクトリのパスを指定すればよい。KEK では HPSS、NO_DCE パラメータを有効にし、Kerberos ライブラリと HPSS のライブラリ、ヘッダーがあるディレクトリパスを指定した。

次に行ったのは Makefile の編集である。ここでは HPSS クライアント API を呼ぶアプリケーションをリンクするために必要なオプションについて追加、修正を行った。

次に srbServMisc.c ソースコードの編集を行った。ここでは、HPSS v6.2 で使用されなくなった hpss_SetAuthType()関数をコメントアウトした。

最後に hpssFileDvr.c ソースコードの

hpssStat に関する部分を修正し、HPSS 上のファイル情報を取得できるようにした。

これらの修正により KEK では SRB から HPSS への接続が可能になった。しかし、現在のところ、HPSS への並列転送は出来ないでいる。これも HPSS のバージョンにより正しいファイルサイズを取得できないことが原因であり、現在それに関するソースコードの修正を行っている。

5. HPSS VFS サービスを使用した接続

KEK では HPSS への接続方法として HPSS クライアント API を使用する方法の他に、VFS を使用する方法を導入した。VFS はユーザからのファイル操作に関する要求を受け付け、それを HPSS の API に変換し HPSS エリアに接続する。ユーザからは VFS での API の変換作業は見えないため、単にローカルディスクにあるファイルにアクセスするような意識でファイル操作が可能になる。これにより、SRB からもローカルディスクとしてストレージリソースを登録可能になるため、HPSS 接続に必要なファイルの編集が必要ない。

しかし現在まだ VFS の安定稼動に入っていないため、詳細な動作確認などは行っていない。

6. MCAT の連携

KEK では SRB を使ったデータ共有の検証として、オーストラリア, 韓国、台湾、中国、ポーランド、日本の 6 カ国にある Belle 実験[4]の共同研究機関と相互連携した(図 3)。

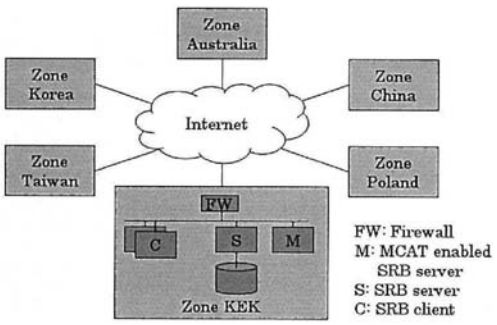

図 3 Belle SRB 連携

この試験的導入では、各サイトに SRB サーバを少なくとも 1 台と MCAT をインストールし、ストレージリソースを登録した。全てのサイトで使用した SRB のバージョンは 3.1.2p で、それぞれが独立した Zone である。各サイトの SRB が動作していることを確認した後、MCAT の連携に必要な情報を相互に交換し連携を行った。連携に必要な情報とは、MCAT-enabled SRB サーバのホスト名、SRB の認証ポート番号、Zone 名、sysadmin 権限を持つ SRB ユーザ名、SRB ドメイン名などである。このうち Zone 名、SRB ユーザ名、SRB ドメイン名は SRB 上で使用される論理名である。連携の際、KEK のネットワークにはファイアウォールが導入されているため、リモートの SRB サーバから接続ができるようにポートを開ける必要があった。

SRB では Zone 名のディレクトリをトップとする階層構造のファイルシステムが作成される。MCAT の連携後に SRB に接続すると、連携している全ての Zone 名がルートディレクトリの下に見えるようになり、どの Zone から接続しても同じツリー構造を見ることができる。また、MCAT の同期を取ることで他の MCAT に登録されている SRB ユーザの情報も連携でき、それにより SRB ユーザは他の Zone へのアクセスが可能になる。図 4 は連携後の SRB に Windows GUI ツールを使用して接続したものである。連携している全ての Zone を見ることができる。

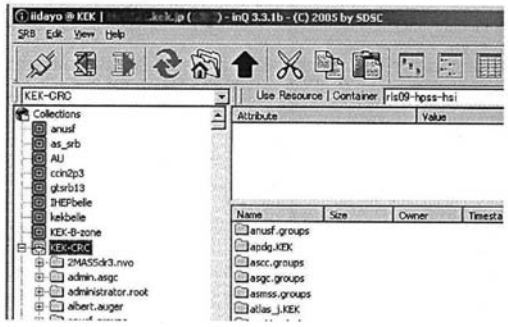

図 4 SRB ファイルシステム

次にデータ共有のテストを行った。KEK の SRB に複数のデータファイルを用意し、他の Zone のユーザがそのデータをそれぞれの Zone のストレージリソースにコピーした。

同様に、他の 4 つの Zone にコピーされたデータにもアクセスし、コピーを行った。

SRB でデータファイルをコピーする際には SRB 上のアクセス権限の設定を行う必要がある。デフォルトではファイルのリストを得ることはできるが、そのファイルを開いたり、コピーしたりすることはできない。そのため、テストに使用するファイルにはテストを行う SRB ユーザへの read 権限を付与しなければならない。SRB でのアクセス権限はユーザ単位、グループ単位、全ユーザで設定することができ、グループは Zone に依存することなくどの SRB ユーザも登録することができる。そのため、テストを行う SRB ユーザを全て一つのグループに登録し、そのグループへ read 権限を与えて行った。

これらのテストにより、MCAT の連携によって他のサイトとのデータ共有が容易になることが実証された。

7. SRB の性能測定

次に、SRB の性能を測定するために、KEK 内の同一 Zone 内でのファイル転送と、高遅延環境にある 2 つの Zone 間でのファイル転送を行った。

KEK Zone 内でのファイル転送に使用したツールは SRB で用意されているコマンドラインで、ローカルファイルを SRB 空間にインポートする Sput コマンドと、SRB 空間のファイルをローカルファイルシステムにエクスポートする Sget コマンドである。これらのコマンドに-m オプションをつけ、最大 16 本のスレッドを使用する並列転送モードで測定を行った。SRB のリソースとしては Fiber-Channel で接続されたディスクストレージを使用した。測定はファイルサイズ 1GB のデータを使用し、インポート、エクスポートをそれぞれ 100 回ずつ行った。その結果が図 5 である。縦軸が転送速度(MByte/sec)、横軸が回数である。SRB 空間への平均インポート速度は 64MByte/sec、SRB 空間からの平均エクスポート速度は 83MByte/sec であった。

SRB サーバと SRB クライアントはそれぞれ GbE で接続されているが、hdparm コマンドを使用して今回 SRB のストレージとして使用したディスクストレージの読み込み速度を計ったところ約 100MByte/sec であったことからそれ以上の性能は期待できない。それを考えると、SRB を使用してのファイル書き込み、読み込みの速度は少し劣るもののそれほど差はないと考えられる。

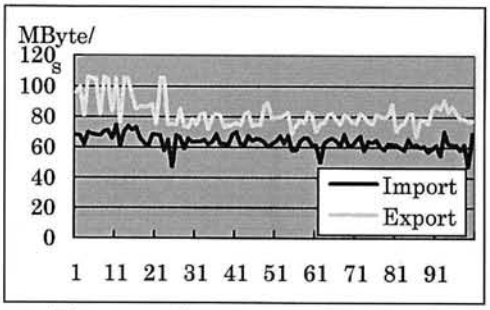

図 5 KEK 内での SRB 空間への Import/Export

高遅延環境にあるリモート Zone 間でのファイル転送の測定として、今回は KEK とフランスの CCIN2P3 の間で行った。KEK と CCIN2P3 の間の RTT(Round Trip Time)は 262ms である。使用したツールは SRB で用意されているコマンドラインで、SRB 空間内でファイルをコピーする Scp コマンドである。このコマンドはデフォルトで並列転送を使用する。測定は SRB 空間にあるファイルサイズ 1GB のデータを、KEK のリソースから CCIN2P3 のリソースへのコピーと逆へのコピーをそれぞれ 100 回ずつ行い、測定した。

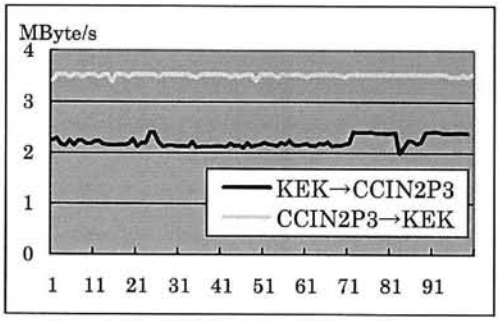

図 6 高遅延環境での SRB コピー

その結果が図 6 である。縦軸が転送速度(MByte/sec)、横軸が回数である。KEK から CCIN2P3 への平均転送速度は 2.3MByte/sec、

CCIN2P3 から KEK への平均転送速度は 3.5MByte/sec であった。

日本とフランスの間のネットワークは少なくとも 2.4Gbps あり、それぞれの SRB サーバは GbE の NIC(Network Interface Card) を持っているが、今回のテストの結果は非常に性能が悪い。そこで、iperf を使って KEK と CCIN2P3 の間のその時点で利用可能なバンド幅を測定してみると、KEK から CCCIN2P3 へは 17Mbps 程度、CCIN2P3 から KEK へは 30Mbps 程度しかないことがわかった。したがって、性能があまり出ないのは SRB のオーバーヘッドなどによるものではないことがわかる。このことから、高遅延環境において性能を上げるためにはネットワークチューニングを行い、バンド幅を広げる必要がある。

8. まとめ

本研究では、SRB を使った分散ファイルシステムの構築を行った。SRB はユーザにグローバルな論理名前空間とファイル階層を提供するミドルウェアである。

KEK 計算科学センターでは大容量ストレージシステムとして HPSS を導入しているが、SRB を使用することでストレージシステムの種類を気にすることなく、ローカルファイルシステムと同じように HPSS を使用することが可能である。ただし、KEK では HPSS への認証方法が特殊であるため、ソースコードの編集が必要であった。また、SRB の開発元が動作確認済みの HPSS は v5.x までであるが、KEK では v6.2 を使用しているため、HPSS で変更された箇所についてもソースコードの編集が必要となった。HPSS への並列転送モードに関するソースコードは現在も修正中である。

また、SRB は MCAT を連携することにより、ローカル Zone の管理を維持したまま他の Zone とデータやストレージの共有を行うことが可能になる。KEK では世界 6 カ国の研究機関と連携を行い、その有効性を確認した。

SRB ではメタデータを MCAT に登録、問合せする必要があることから、そのオーバーヘッドが予想される。しかし、今回行った性能測定では並列転送などを使用することで十分な性能が期待できることがわかった。ただし、高遅延環境においてはネットワークのチューニングを行うことが必要であると思われる。

現在 KEK では SRB システムを分散ファイルシステムとデータグリッドシステムを兼ねたものとして Belle 計算機システムへ導入を開始した。

9. 謝辞

本研究を行うにあたり、ご協力いただいた以下の方々に感謝します。

Adil Hasan 氏、石川公基氏、松井学氏、Glenn Moloney 氏、Jean-Yves Nief 氏、Michael Wan 氏、山本智美氏

参考文献

[1] SRB (The SDSC Storage Resource Broker)
http://www.sdsc.edu/srb/index.php/Main_Page

[2] HPSS (High Performance Storage System)
http://www.hpss-collaboration.org/hpss/index.jsp

[3] DCE (Distributed Computing Environment)
http://www.opengroup.org/dce/

[4] Belle Collaboration http://belle.kek.jp/