共通データ解析システムの導入と運用

八代 茂夫[*1]、佐々木 節[*1]、川端 節彌[*1]、渡瀬芳行[*1]
東 孝司[*2]、大塚 薫[*2]、伊藤 義彦[*2]
宮下 勉[*3]

[*1] 高エネルギー加速器研究機構(KEK)
[*2] 日本アイ・ビー・エム株式会社
[*3] ケイエスケイアルパ株式会社

概要

素粒子実験で収集した実験データの解析を主目的とした分散処理型のデータ解析システムを構築した。120TB のデータ管理のために HPSS を導入した。Client-API インターフェイスを利用することにより、容易にアプリケーションプログラムからの高速なアクセスが実現できた。また、統合管理ソフトウェアの活用、システムの状態監視の自動化、アカウンティング情報収集の自動化により運用工数の削減が可能になった。

1. はじめに

共通データ解析システム(KEKCC)は、高エネルギー加速器研究機構(KEK)で行なっている 12GeV 陽子シンクロトロン加速器を用いて行われる実験で収集したデータの解析を主目的としたシステムである。長基線ニュートリノ振動実験（K2K）をはじめとし、さまざまな素粒子実験、原子核実験のデータ解析に利用されている。一つの実験あたり最大数十テラバイトにおよぶ大量の実験データを高速に解析することが可能なように KEKCC は設計されている。実験は 24 時間連続して数ヶ月続くこともあるので、たとえ計算機システムの一部に障害が発生しても、全体の機能が停止しないように設計されていいる。

システム構成は、ホームディレクトリサーバ、データサーバ、計算サーバ、ログインサーバ、および DCE サーバから成り、ハードウエアは IBM RS/6000-SP 79 ノードと総容量 120TB の磁気テープライブラリ装置を主構成要素としている。分散処理環境の構築には DCE/DFS を、データサーバの HSM (Hierarchical Storage Management、階層型ストレージ管理)ソフトウエアには HPSS(High Performance Storage System)[1][2][3]を、データ解析プログラムのジョブスケジューリングには Platform Computing 社の LSF(Load Sharing Facility) [4]を採用した。

本論文では、データサーバを中心とした、KEKCC のシステム構築および運用について報告する。

Construction of a Distributed Data Analysis System and Its Performance

S. Yashiro[*1], T. Sasaki[*1], S. Kawabata[*1], Y. Watase[*1], K. Azuma[*2], K, Ohtsuka[*2], Y. Ito[*2], and T. Miyashita[*3]
[*1] High Energy Accelerator Research Organization(KEK), [*2] IBM Japan Ltd.,
[*3] KSK Alpa Co.

2. 分散処理システム環境

分散処理環境においては、性能と利便性あるいは管理容易性がトレードオフとなる。システム構築にあたっては、スケーラビリティをそこなうことなく、これらが容易に実現できるように特段の配慮をしなければならない。KEKCC においては各サブシステムを、分散システムとして統合することによって、この目的を果たした。

分散処理システムの利便性、高セキュリティ、性能、耐障害性、管理容易性を達成するために、DCE を採用した。DCE はセキュリティサービス、ディレクトリサービス、タイムサービスといった分散システムを構築するための基本的な諸機能を提供している。また、ファイル I/O の透過性を確保するために、DCE/DFS(Distributed File Service)を採用した。

DCE と HPSS は別個のソフトウエアであるが、HPSS 自身も DCE の認証を用いているので、システム全体の DCE 環境に HPSS を統合し、ユーザ認証の一元化を行った。

3. HPSS

HPSS は、米国の ASCI 計画の要求を満たすために計画に参加している、米国エネルギー省管轄下の研究所、ローレンスリバモア国立研究所、ロスアラモス国立研究所、サンディア国立研究所などと IBM によって共同開発され、維持されている。HPSS の特徴は、オープンでスケーラブルな階層型ストレージ管理機能を持ち、多種にわたるストレージ・デバイスを単一システムのイメージとして管理できる環境を有することである。HPSS 自身も分散システムとして設計され、実装されており、機能をネットワーク上の複数のサーバホストに目的ごとに分散させている。HPSS においては、コアサーバがメタデータの管理を行っており、これによってファイルの位置透過性が実現されている。HPSS 環境においては、実際のデータはムーバと呼ばれるサーバプロセスが稼働するノードに接続されたストレージ・デバイスに格納される。ディスク装置に対してはディスクムーバがあり、テープ装置に対しては、テープムーバがあり、それぞれのサーバプロセスがファイル転送を行う。KEKCC においては、ディスクムーバとテープムーバが異なったホストの上に構成されているので、ファイルのマイグレーションやステージングの際に発生するムーバ間のファイル転送は、ネットワークを介して行われる。分散環境においてはオーバーヘッドが少なく、かつ高スループットのネットワークが必須であるが、KEKCC では、SP スイッチと呼ばれる双方向で 300MB/sec の通信速度を持つネットワークが採用されている。図 1に HPSS の主要な構成要素、およびクライアントからのデータ要求時の処理概要をを示す。

高性能なストレージシステムを構築するためには、導入前の入念なシステム設計と運用計画の策定が不可欠である。従って、計画策定局面においては、HPSS 関連の計画を重点的に実施した。導入を開始するにあたって、既に運用しているサイトに対しての聞取調査、ユーザとの運用についての打ち合わせ、開発者との打ち合わせを重ねて実施した。

HPSS は、米国、欧州の約 20 の研究機関で稼働しており、日本においては現在、KEK と理化学研究所で稼働している。

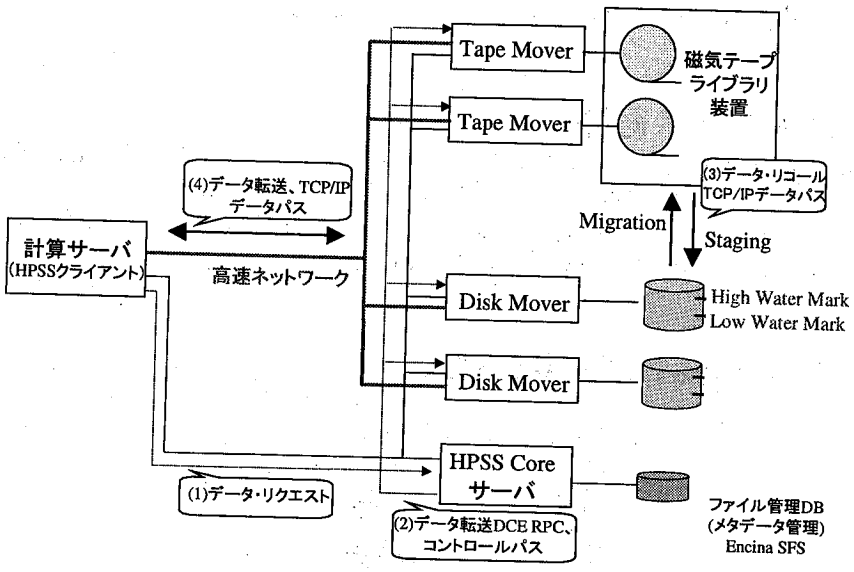

図 1 HPSS の概略図

4. データ管理とアクセス方法

共通データ解析システムにおいては、約 120TB の容量のデータ管理を行わなければならないため、全てのデータをディスクの上に保持しておくことは予算上困難であり、磁気テープと磁気ディスクをシームレスに利用することができるような HSM が必要とされた。過去に HSM を運用した経験から、システム構築にあたっては以下の点に留意した。

a) HSM のシステム設計にあたっては、各構成要素のバランスが非常に重要である。仮にテープ装置が非常に高速であっとしても、ディスク装置の構成が原因となって期待される性能を得られない可能性がある。なぜならば、通常の運用形態では、HSM のディスク・キャッシュは多くのプロセスから共有され、マイグレーションやステージングなどのプロセスが同時に発生する可能性があるからである。

b) 商用で提供されている HSM には、ディスク・キャッシュ・スペースの利用率が閾値(High Water Mark)を越えてから実際のマイグレーションが発生する仕様となっているものもある。このようなケースでは複数のアクセスが同時に発生することによって磁気ディスクへのアクセスが非常に混雑する。従って、データのマイグレーションに関しては、この閾値を越える前に定期的にマイグレーションを行い、閾値を越えた時にはファイルの削除のみが行われるようにする必要がある。

c) 柔軟なキャッシュ・スペースの割当が可能となるように、1 つの論理ディスク・キャッシュ・スペースには、複数の物理デバイスがプールとして割り当てられる必要がある。

d) 高性能なデータ転送を実現するために、ユーザー・インターフェイスとして最適化された API が提供されることが必須である。NFS やその他の分散型のファイルシステムは高速データ転送には不向きである。

以上の点を考慮に入れ、KEKCC の仕様を決定した。

5.HPSS へのアクセス経路

HPSS でサポートされているクライアントアクセスのインターフェイスのうち KEKCC では、Client-API, Parallel FTP, FTP, および DFS を運用している(図 2)。その他 NFS, MPI-IO インターフェイスがサポートされているが、これらは KEKCC で運用していない。ファイルシステムとしてのアクセスは DFS と NFS である。

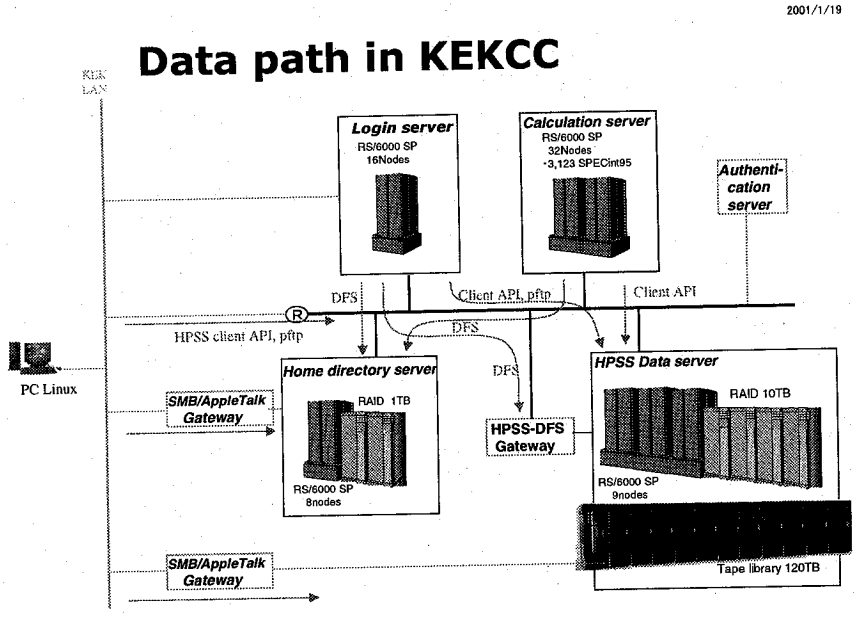

図 2 KEKCC での HPSS へのアクセス経路

DFS はファイルシステムとしてアクセスできるので、既存のプログラムからのアクセスがそのまま可能であり、またファイルの状態やパスを調べたりすることが用意に行なえる。一方データ解析には十分な性能を有していない。

データ解析には client-API を主として使用することを検討した。KEKCC 内部でのアクセスに加えて、ユーザの Linux 機にプログラムをインストールして KEKCC 外部からアクセスすることができる。Client-API は Kerberos 認証をサポートしているので、併せて Kerberos も Linux 機にインストールしてセキュアなユーザ認証が可能である。

Parallel FTP と FTP も、ユーザのワークステーションとの間のデータ転送の主要な経路になる。FTP は多くのマシンにクライアントが導入されているので直ちに使えるが、認証がセキュアでないので、通常は Kerberos 対応となっている Parallel FTP が使用されている。

6. HPSS Client-API によるラッパープログラムの設計

HPSS によって提供される Client-API そのものは、1MB 以上のような比較的大きなレコード長のデータに最適化されたものである。しかし KEKCC の多くのユーザのデータ解析プログラムは、64KB 以下の小さなレコード長でデータを作成している。このまま HPSS をアクセスしたのでは性能的に問題がでることが予想された。問題を解決するために、バッファリング・メカニズムを実装したラッパープログラムを開発した。

ラッパーの設計としては、open、read、write、close、seek、stat といったシステムコールを置き換える形とした。アプリケーション・プログラムが C 言語で書かれている場合は、ソースプログラムにヘッダ・ファイルをインクルードするだけでこのラッパープログラムを利用可能である。

このラッパープログラムを使用した HPSS へのアクセスは、KEKCC 内部はもとより、所内 LAN を経由して利用者自身が保有する PC-Linux 機などの機器からも利用可能となっている。

7. ベンチマーキング

Client-API のラッパープログラムのベンチマークテストを 17 ノードの RS/6000-SP(POWER3)を用いて実施した。10 ノードのクライアントと 6 ノードのディスクムーバおよび 1 ノードの Core サーバによって構成され、ソフトウエア環境は、AIX4.3.3, DCE 2.2, Encina 4.2, そして HPSS R4.1.1 である。ベンチマークテストの計測項目は、入出力の性能およびクライアントにおける CPU 利用率を計測した。1 つのクライアントあたり 1 つの read と 1 つの write で合計 20 プロセスを実行した。それぞれのプロセスのターゲットは 4GB のファイルとし、データ・リクエストのレコード長を 64KB とした。このベンチマーク方法は、実利用環境のプロセスあたりの入出力性能とスループットおよび CPU の消費をシミュレートするためであり、ベンチマークテストとしては極めて厳しいものとなっている。

得られた性能は、以下の通りである。

a) 1 クライアント・プロセスあたりの平均 read 性能 : 32MB/s
b) 1 クライアント・プロセスあたりの平均 write 性能 : 15MB/s
c) クライアントにおける平均 CPU 利用率 : 16%

HPSS のスループットは、6 台のディスクムーバ構成において 464MB/s に達し、1 台のムーバあたりでは 77MB/s を実現した。高い転送性能ながら、クライアント側の CPU 利用率は平均で 16% と低く押さえられており、計算をしながら入出力を行うという実験のデータ解析に適したシステムであることが実証された。更に実機でも運用開始前に同様のベンチマークテストを行い、この結果が追証された。

8. 運用

このような複雑な分散処理システムでは、運用工数がそのままでは相当のものとなる。そこで、これらの工数削減を目的として、RS/6000-SP の統合管理ソフトウェアである PSSP (Parallel System Support Program) の利用や、システムの状態監視やアカウンティング情報収集を自動化する独自ツールの作成によって運用工数削減を図った。

PSSP は、全ての RS/6000-SP ノードをコントロールワークステーションから一元管理するソフトウエアである。管理項目には、ノードの停止、起動、インストール、複数ノードへの同

一コマンド発行等がある。PSSP を使用することで、人的工数に頼らざるを得ない管理作業についても、その作業工数削減を行うことができる。

システム監視は、独自作成ツールを CRON で 1 時間に 1 回実行して、監視項目について異常があった場合システム管理者にメールで通報する仕組みとなっている。現在、監視している項目は、ハードウェアエラー、ファイルシステムの使用率、プロセスの状況、磁気ディスクの動作状況、HPSS ログである。これにより、通常運用時には監視体制にかかる人的工数が大幅に削減され、より高度なシステム改善活動に作業時間を割り当てることが可能となっている。

分散システムを常に最適なシステム構成とするためには、システム使用状況を表すアカウンティング情報が重要である。アカウンティング情報を分析して、資源の最適配置を見直すことができる。そこで、アカウンティング情報を自動的に取得するためのツールを作成した。KEKCC で収集しているアカウンティング情報は、以下の項目である。

a) DFS 領域のファイル量
b) HPSS 領域のファイル量、ファイル数
c) ディスクからテープにマイグレートされたデータ量
d) テープドライブ使用率、メディアマウント回数

9. サマリー

これまでのシステム構築と運用の経験を生かして、新しいシステムを導入した。分散処理型のデータ解析システムを DCE および HPSS を軸として構築した。ラッパープログラムはエンドユーザが HPSS の Client-API を有効利用することを助け、高速なデータ転送の実現が容易にでるようになった。これを用いて、共通データ解析システムにて用いられる多くのツールが HPSS 対応された。ラッパープログラムは Linux においても利用可能なので、エンドユーザのマシンから HPSS をデータサーバとして利用することが可能になった。

参考文献

[1] D. Teaff, R. W. Watson, and R. A. Coyne, "The Architecture of the High Performance Storage System (HPSS)", *Proc. Third Goddard Conference on Mass Storage Systems and Technologies*, March 1995

[2] HPSS User's Guide, August 2001

[3] S. Yashiro, T. Sasaki, S. Kawabata, M. Yamaga, M. Aoki, Y. Ito, K. Azuma, K, Ohtsuka, S. Masuda and J. L Schaefer, Data transfer using buffered I/O API with HPSS, *CHEP2001, Beijing, P. R. China, September 2001, KEK preprint 2001-51*

[4] LSF 管理者ガイド Version 4.0, Platform Computing Corporation, Feb. 2000