SCore 入門

## SCore


- 新情報処理開発機構（RWCP）で開発されたクラスタ計算機用並列プログラム実行環境
- RWCP
  - 1992年から10年間の研究プロジェクト
  - 2001年10月に最後の研究成果発表会を開催
- 現在はPCクラスタコンソーシアム（PCCC）が開発、普及活動を行っています
- ギャングスケジューリングを用いたマルチユーザ環境
- 対話型実行環境
- 高速通信ライブラリ：PMv2
- MPI,OpenMP,MPC++などの並列プログラミング環境
- チェックポイント・リスタート機能
- これらの機能をトータルに開発した実行環境です。
- 「LINUXで並列処理しよう　第2版」共立出版を参考にして下さい。

SCore 入門

## SCoreの歴史


- 形に成り出したのは1994年頃から
- 元々はRWC-1用のOS（実行環境）として研究開発
- RWC-1の開発が進むにつれて、コモディティなマシン上で並列プログラム実行環境の重要性が高まってきました。
  - NoW、Shrimp、Beowulf
- SUNワークステーション上にSCoreを開発しました。
  - SUN SS20 x 36 ノード　Myrinet LANAi3/SBUS
  - ノード間通信性能 30MBytes/sec 程度

SCore 入門

## SCoreの歴史


- PCを用いてNetBSD上に移植
  - 66MHz DXII DEC PC
- PICMG PC 32ノードのPCクラスタを製作
  - エンクロージャ等は東清物産（当時）様に発注
  - 東清システムインテグレーションズ様 (http://www.tosei-si.com/) は“COSMO”クラスタとして商品化
  - SC96(ピッツバーグ)に展示して注目を集めました
    - SC97 San Jose
    - SC98 Orland
    - SC99 Portland
    - SC00 Dallas
    - SC01 Denver (展示は中止)
    - SC02 Baltimore
    - SC03 Phoenix
    - SC04 Pittsburgh

RWC PC Cluster I
NetBSD
Diskless Boot

SCore 入門

## SCoreの歴史

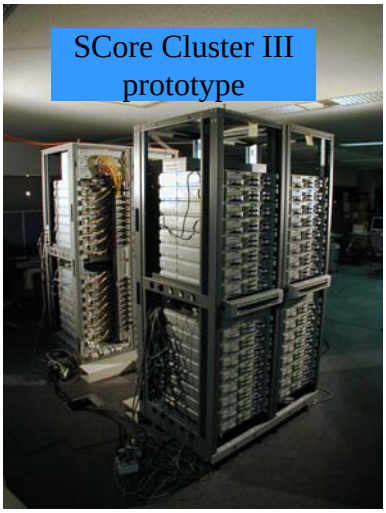

- SC97(サンノゼ)でLINUX対応
- RWC PC Cluster IIクラスタ
  - PICMG PenPro 200MHz 128ノード
  - Myrinet LANAI 5
  - 19インチラック4本
  - SCで展示しました。
- 90年代中頃
  - PC互換機なんて…
  - x86なんて…
  - LINUXなんて…
- 10年後の現在は？

RWC PC Cluster II
Linux

SCore 入門

## SCoreの歴史


- SCore Cluster III
  - Pen3 933MHz Dual
  - NEC Express 512 ノード
  - Myrinet 2000 シリアルケーブル




SCore 入門

## SCoreの歴史


SCore 入門

## SCore6


- 2006年11月13日、SCore6をリリースしました。
- 2002年3月SCore5.0リリース以来、4年半ぶりのメジャーバージョンアップ
- SCore6の主な変更点についてご紹介します。

SCore 入門

## SCore6
## CentOSの採用


- ベースOSをCentOS4.4に変更しました。
  - SCore5.8では FedoraCore3でした。
  - バイナリ配布イメージには CentOS4.4（http://www.centos.org/）が含まれています。
- Redhat EL クローン採用の結果…
  - より沢山のx86サーバでSCoreを動作させることができます。
  - 信頼性の高いLINUX環境でSCoreを使って頂けます。
  - Security Patchなどへの対応も容易
  - 企業ユーザの導入も容易になったかと思います。
  - SIerにとっても対応しやすい。
- ソースコードの配布も継続しています。
  - 他のLinux ディストリビューションでも動作可能です。

SCore 入門

## SCore6 カーネルパッチが不要になりました。


- デバイスドライバ類がモジュール化されました。
- カーネルパッチ、カーネルの配布が不要になりました。
  - CentOSはRH ELのクローンです。
  - 結果的に、RH EL 4でも…
- カーネルを入れ替える必要がありません。
- インストールが格段に容易になりました。
  - いくつかのスクリプトをchkconfig/service on する
- 普段ご使用のLINUX環境をそのまま使って頂けます。
- SCore導入の敷居が低くなった。

SCore 入門

## SCore6 インストレーションツールの復活


- インストレーションツールとしてEITが含まれています。
  - Anacondaベースのインストレーションツールです。
  - GUI環境から容易にCentOSベースのSCoreクラスタ環境を構築できます。
  - 計算ノードの追加インストールにも対応しています。
  - 計算ノードはブートCDから起動します。
    - PXEブート環境の構築は不要です。
    - PXEファームのないx86マシンにも対応。
  - **（ホワイトボックスなど）ベンダー製Deploymentツールのないx86サーバでも、SCoreクラスタ環境を構築できます。**
  - これまで通り、ソースコードインストール、バイナリインストールも可能です。

SCore 入門

## SCore6アップデート チェックポイント・リスタート


- CRTを加えました。
  - CRT：Checkpoint Restart for Threads
- スレッドセーフ
- ダイナミックリンクに対応
  - スタティックリンク不要です。
  - バイナリサイズの削減。
- チェックポイント可能なプログラムの拡大

SCore 入門

## SCore6アップデート YAMPIの導入


- MPI環境として YAMPIが導入されました。
  - http://www.il.is.s.u-tokyo.ac.jp/yampii/
  - 石川会長を中心に研究開発
- MPIはクラスタ環境の要
- 大規模化、安定化、GRID対応などクラスタ計算機に対する課題に迅速に対応するには、MPICH以外の自前のMPIの実装が必要

SCore 入門

## SCore6アップデート PM/Ethernet HXBの導入


- PM/Ethernet HXBが組み込まれました。
- 筑波大学様、富士通様からのコントリビューション。
- 筑波大学 PACS-CSプロジェクトで開発されました。
- イーサネットによる3次元ハイパークロスバーネットワークを構築できます。

SCore 入門

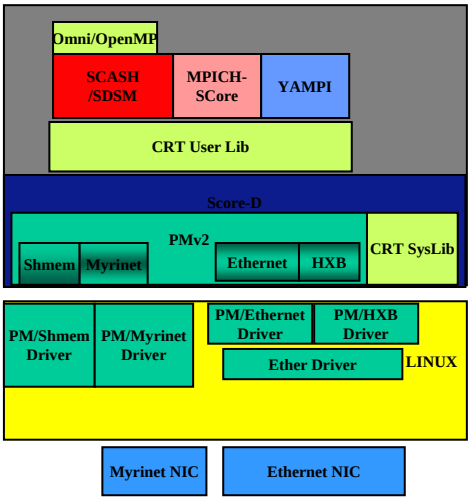
## SCore6アップデート SCore-Dのデーモンモードでの実行


- SCore-Dの起動にscoutは不要
- マルチユーザモードにおいて、動的に計算ホストを変更することができるようになりました。

SCore 入門

## SCore ソフトウェア構成


- プログラミング環境
  - MPICH-SCore
  - YAMPI
  - Omni/OpenMP on SCASH
- SCore-D
  - マルチユーザ環境
    - ギャングスケジューリング
    - 時空間分割スケジューリング
  - リソース管理
  - 実時間ロードモニタ
  - チェックポイントリスタート
  - デッドロック検出
- PMv2通信ライブラリ
  - Myrinet
  - Ethernet
  - Shmem



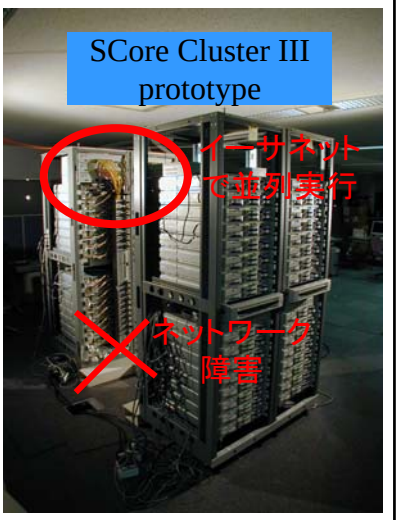

SCore 入門

## PMv2：複数通信メディアへの対応


- 元々はMyrinet専用の通信APIとして開発
  - 最初は複数の通信メディアへの対応は予定していませんでした。
- Ethernet
  - 大量生産品が高性能化するとコストパフォーマンスで有利
  - Trunking
    - 複数のNICをまとめて使うことが出来ます
    - GbE 2枚でほぼ2倍のバンド幅で通信できます
    - HXBでは1GB/S以上が確認されています。

SCore 入門

## PMv2：複数通信メディアへの対応




·SCoreでは複数の通信メディアを1つのバイナリから使うことができます。つまり1つのa.out でMyrinetもEthernetも使うことができます
  ·% scrun –network=myrinet
  ·%scrun –network=gigaethernet
  ·%scrun –network=ethernet
·通信メディア別にa.outをlinkする必要はありません
  ·但しMPIはMPICHベースのためコンパイラごとにランタイムが存在します。。。。
·PMは通信メディアごとに実装されています。
  ·Type = myrinet,myrinet2k,myrinetxp,ethernet,etc

•デフォルトのネットワークが故障してもa.outを再コンパイルせずにEthernet で並列処理を行う事が可能です。

SCore 入門

## MPICH-SCore

ここからプログラミング環境

- MPICHは米国立アルゴンヌ研究所が開発したフリーソフトウェア
- MPI（Message Passing Interface)の実装の一つ
- MPICH-SCoreはMPICHをSCoreに移植
  – 通信部分をPMv2で実装
- クラスタ上のほとんどのプログラムはMPIで記述されています。
- これからクラスタを導入される方は、、、
  – 既存のソフトウェア資産の移植等を考慮する必要があります。

SCore 入門

## MPICH-SCore


- コンパイラ
  - mpicc、mpic++、mpif77、mpif90
  - Intel,PGI,Fujitsu,Pathscaleコンパイラを使用することも可能です
    - %mpicc –compiler intel ……
  - デフォルトコンパイラはGNUコンパイラ
  - デフォルトコンパイラは設定ファイルを変更することで変更できます
  - mpif90については、GNUのF90コンパイラがないので、別途上記コンパイラを購入する必要があります
    - gcc4.0以降はf90 に対応している

SCore 入門

## MPICH-SCore


通信プロトコル切り替え

- Short protocol
  - MPIヘッダーとメッセージデータを1つのパケットで送信する
- Eager Protocol
  - MPIヘッダー＋複数のメッセージデータパケットで送信
  - 受信側が受信待ち状態になっていなくともメッセージを送信します
  - 受信待ち状態にない場合、受信側はメッセージを受信バッファに一時的にコピーします
- Randevous Protocol
  - MPIヘッダー＋複数のメッセージデータパケットで送信
  - 受信側が受信待ち状態になってからメッセージを送信します
  - 受信側はアプリケーション領域にコピー可能
  - 受信側でメッセージのコピーを削減
  - 受信待ち状態を送信側に通知する必要があります
- Short/Eagerの切り替えは 1KBytes　固定
  - 変更するには、ソース書き換え、リコンパイル
- Eager/Randevousの切り替えは16KBytes
  - % scrun –nodes=8,mpi_eager=40960

SCore 入門

## MPICH-SCore


- PMv2のRMA通信機能をMPICH-SCoreから使うことが出来ます
  - % mpirun –np 8 –score mpi_zerocopy=on ./mpi_app
  - % scrun –nodes=8,mpi_zerocopy=on ./mpi_app
- ゼロコピー通信はRandevous プロトコルにのみ使われます
  - 1KBytes以下のメッセージ（Shortプロトコル）には適用されません
  - 上記の例では16KBytes以上（Eager/Randevous切り替え)にのみ適用されます
- プロトコル切り替え・ゼロコピー通信はPM/Myrinetで効果を発揮する
- PM/Ethernetはmaxnsend,backoff等PM/Ethernetのパラメータチューニングが効果的
- PM/Ethernetはトランキングも効果的

SCore 入門

## MPICH-SCore


- 通信ログ
  - % mpicc –mpilog –o hello hello.c
- 視覚化ツール
  - Upshot
    - Alog 形式
    - % setenv MPE_LOG_FORMAT=ALOG
    - % Scrun ….
    - % logviewer hello.alog
  - Jumpshot3
    - Slog 形式
    - % setenv MPE_LOG_FORMAT=SLOG
    - % scrun ….
    - %logviewer hello.slog
- 通信の視覚化
  - % mpicc –mpianim –o hello hello.c –LX11 –lm –L/usr/X11R6/lib

SCore 入門

## デバッガ


- **アプリケーション実行中の例外発生時にデバッガを起動することが出来ます**
- **% scrun –nodes=8,debug ./hello**

SCore 入門

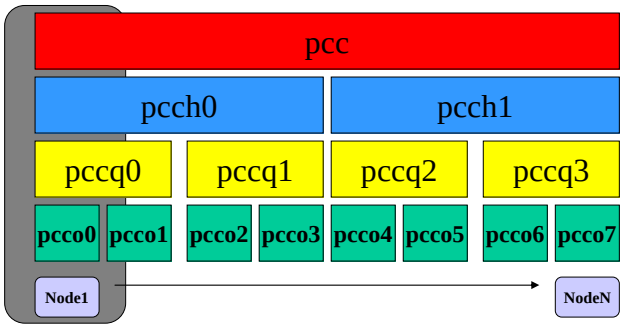
## 並列プロセスの起動


- SCoreでは scrun コマンドで並列プロセスを起動します。
- MPICH/SCore では mpirun コマンドも提供されています。
  - MPICH/SCore のmpirun コマンドでは、引数“-score”で score オプションを指定します
  - % mpirun –np 4 –score monitor=load　a.out
- scrun/mpirun コマンドを実行すると、全てのノード計算機上に、実行ファイルがコピーされます。(SCORE OPTIONでコピーを抑制することも可能)
- 他のMPI環境（mpich等）の mpirun コマンドのように、全てのノードで実行ファイルを共有する必要はありません。
- ファイルサーバがなくても、クラスタを構築することができます。
  - 大規模クラスタを構築する場合に有利
- クラスタ計算機の規模拡大、ディスクの大容量化への対応は今後の課題の一つ

SCore 入門

## ノードのグループ管理


- SCoreでは 複数のノードをまとめてグループとして管理できます。
- ノードは複数のグループに所属できます。
  - 下図では、Node1はpcco1,pccq0,pcch0,pccの4つのグループに属しています。
- % scrun –group=pcch0,nodes=4x2,network=ether  a.out
- % scorehosts –g pcch0




SCore 入門

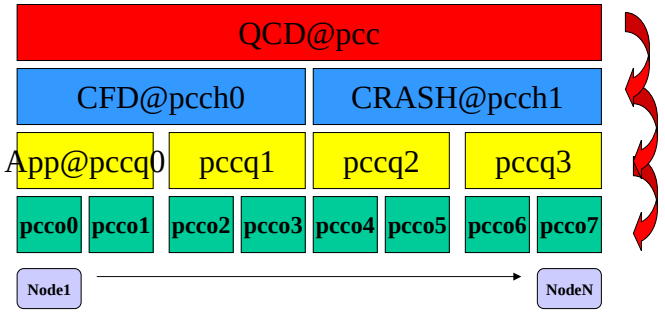
## 実行環境


- シングルユーザ環境
  - クラスタ環境の全体または一部を排他的に使用する
- マルチユーザ環境
  - SCoreが目指した実行環境
  - 複数ユーザの複数プログラムを同時処理
  - ギャングスケジューリングによる時分割処理
  - 空間分割
  - あまり使われていないという話も聞きますが、クラスタ計算機が身近になるにつれて普及してほしい

SCore 入門

## マルチユーザ環境


- タイムシェアリング環境のようにクラスタ計算機を使用できる
- 複数ユーザの複数ジョブを同時処理
- 時分割
- 空間分割




SCore 入門

## マルチユーザ環境


- 同時に複数のプログラムを実行可能

```
% scout –g pcc
% scored –server  nodeX
SYSLOG:  ++++++++++++++++++
SYSLOG: -----SCore-D (5.8.3) bootup ----
```

```
% mpirun –np 32 –score scored=nodeX yellow
```

```
% scrun –nodes=8x2,scored=nodeX    green
```

```
% export SCORE_OPTIONS=scored=nodeX
% scrun –nodes=16x2,checkpoint=8h    red
```

```
% sc_console  nodeX –c   shutdown
```

SCore 入門

## SCoreのノード障害対応（1）


- /opt/score/etc/scorehosts.defectsに記述されたノードには、ジョブが割り当てられなくなります。
- 上記のファイルに障害が発生したノードを記述することによって縮退運転を行うことが出来ます

```
% scorehosts –g  pccg0
 node0 node1 node 2 node3
① /opt/score/etc/scorehosts.defects に   node1 を記述
        % sceptic –g pcc
        …… 障害ノードの検出
② /etc/init.d/scoreboard reload
% scorehosts –g pccg0
 node0 node2 node3
```

SCore 入門

## SCoreのノード障害対応（2）


- /opt/score/etc/scorehosts.db の各ノードに対して、スペアノードを記述することができます。
- /etc/score/etc/scorehosts.defects に記述されたノードに対しては、スペアノードにジョブが割り当てられます。
- 予めホットスペアノードを準備することによって、ノードに障害が発生しても、被害を最小限に食い止めることが出来ます。

```
% scorehosts –g  pccg0
 node0 node1 node 2 node4
① /opt/score/etc/scorehosts.defects に   node1 を記述
② /etc/init.d/scoreboard reload
% scorehosts –g pccg0
 node0 sparenode node2 node3
```

SCoreはスケジューラの助けがなくともノード障害に対応できます。

SCore 入門

## SCore-D


- チェックポイント・リスタート
  - 実行中のプログラムをファイルに退避
  - 退避したプログラムを再実行
- % scrun –checkpoint=10m ./hello
  - -checkpoint オプションでチェックポイント間隔を指定
  - チェックポイント間隔毎にプログラムが退避されます
  - 時間指定は sS,mM,hH,dDを指定可能
- %scored –restart
  - -restartオプション付きでscoredを起動すると退避されたプログラムが再起動されます

**SCoreのチェックポイントリスタートは、全てのプログラムに適用できます！**
**MPI環境やノード内といった制限はありません。**

SCore 入門

## バッチ環境

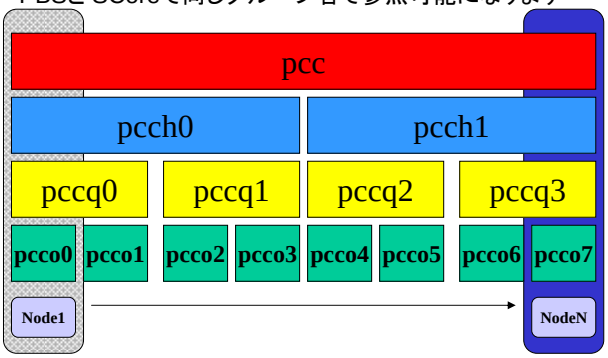

- シングルユーザモードでバッチ環境を構築できます。
- PBSPro、TORQUE/Maui、LSF、SGEなどで運用実績があります。
- クラスタ計算機用バッチスケジューラはMPI環境に対応しています。

```
Sample_job.sh
#!/bin/sh
#PBS –j oe
#PBS –l nodes=4:score
#PBS –q queue
mpirun  -machinefile pcc.hosts   hello_world
```

SCore 入門

# PBS/Torque の導入


- PBSのノードプロパティ
  - SCoreのノードグループに対応
  - PBSと SCoreで同じグループ名で参照可能になります




```
# qmgr : set node node1 properties="pcc,pcch0,pccq0,pcco0"
# qmgr : set node nodeN properties="pcc,pcch1,pccq3,pcco7"
```