ディスクフルクラスタの構築

知的システムデザイン研究室

1　はじめに

シミュレーションや解析の分野では高性能な計算資源が必要であるため、PC クラスタを用いることが多くなってきている。PC クラスタとは複数の汎用 PC を接続し、仮想的に一つの計算機とする技術である。PC クラスタには全てのノードがディスクを持つディスクフルクラスタと、ディスクを持たないディスクレスクラスタがあり、その構築方法は異なる。本マニュアルでは、ディスクフルクラスタの構築について述べる。なお、PC クラスタに用いる OS は Debian GNU/Linux 4.0(etch) とする。

2　PC クラスタとは

PC クラスタは、以下の図のようにマスタノード (ホスト名：m1) と計算ノード (ホスト名：c1,c2…) によって構成される。

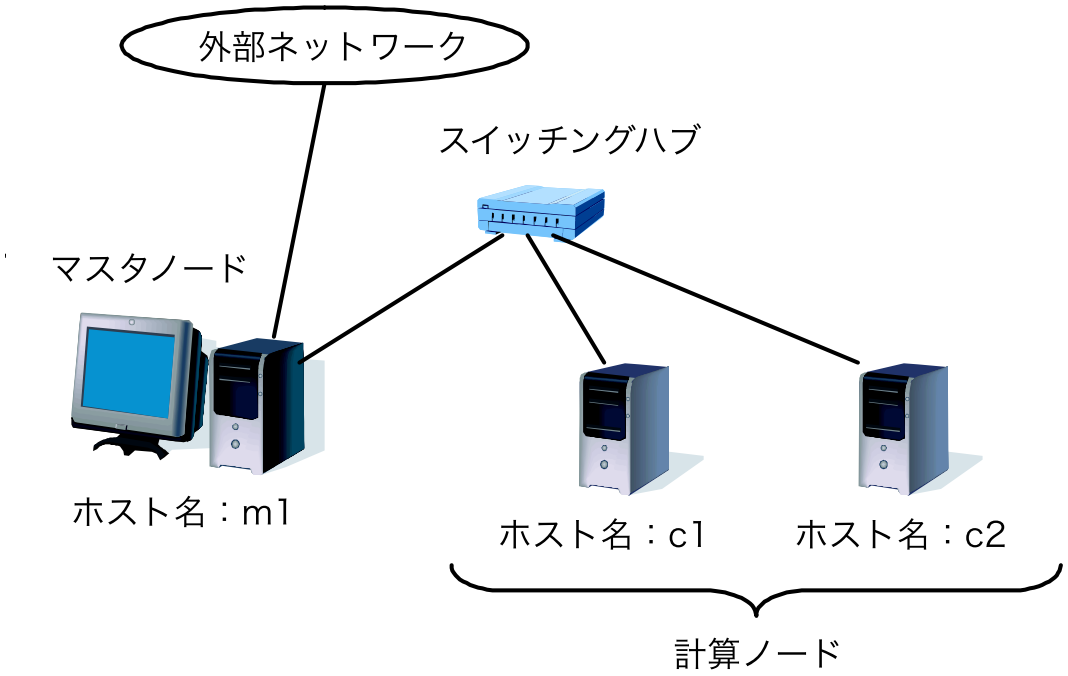

Fig. 1　ディスクフルクラスタの構成

マスタノードと計算ノードは、閉じたネットワークで繋がっており、マスタノードのみが外部のネットワークと繋がっている。PC クラスタを利用したいユーザは、外部ネットワークからマスタノードにログインし、計算ノードに対してジョブを投入する。

3　構築手順

1. OS のインストール
   クラスタに用いる全てのマシンに Debian をインストールする。インストール方法については、別資料を参考にしてください。

2. ネットワークの設定
   マスタノードと計算ノードの IP アドレスの設定は今回は次節で説明するように行った。ネットワークインターフェースに関するアドレス等の設定は以下のファイルで行う。

```
# vi /etc/network/interfaces
```

3.1 マスタノードでのネットワークの設定

マスタノードでは、外部との通信・計算ノードとの通信という 2 つのネットワークインターフェースの設定を行う。eth0 を外部用、eth1 を内部用に設定する。eth0 は DHCP で IP アドレスを取得するようにした。
※ NIC（Network Interface Card）が一つしかない PC を利用する際には、次節以降のフローで説明するパッケージを事前にインストールし、th0 に対してクラスタ内部のネットワーク設定を行う。

1. eth0 に DHCP により自動的にアドレスを割り当てるには、以下を追記する。

```
auto eth0
iface eth0 inet dhcp
```

2. eth1 に固定 IP アドレスを割り当てるには、以下を追記する。

```
auto eth0
iface eth1 inet static
address 192.168.1.1
network 192.168.1.0
netmask 255.255.255.0
broadcast 192.168.1.255
```

3. networking の再起動
   ネットワークの設定を反映するために networking を再起動する。

```
# /etc/init.d/networking restart
```

3.2 計算ノードでのネットワークの設定

計算ノードでは、内部での通信の設定のみを行う。eth0 は以下のように IP アドレスを設定する。
※ NIC（Network Interface Card）が一つしかない PC を利用する際には、次節以降のフローで説明するパッケージを事前にインストールし、th0 に対してクラスタ内部のネットワーク設定を行う。

1. eth0 に固定 IP アドレスを割り当てるには、以下を追記する（c1 の場合）。

```
auto eth0
iface eth0 inet static
address 192.168.1.2
network 192.168.1.0
netmask 255.255.255.0
broadcast 192.168.1.255
```

2. networking の再起動
   ネットワークの設定を反映するために networking を再起動する。

```
# /etc/init.d/networking restart
```

3.3 マスタノードでの rsh のインストールと設定

rsh(remote shell) とは、リモートシステム上で指定したコマンドを実行するためのコマンドである。

1. rsh クライアントをインストールする。セキュリティの問題上、マスタノードとなるノードには rsh サーバはインストールせず、rsh クライアントのみをインストールする。

```
# aptitude install rsh-client
```

2. /etc/hosts の編集
DNS を用いないクラスタでは、IP アドレス、ドメイン名およびホスト名の照合は/etc/hosts のファイルで行われる。下記のように IP アドレスとホスト名を/etc/hosts に記述する（追記する）。ドメイン名の部分、つまり master.domain.name や slave1.domain.name などは書かなくてもよい。

```
192.168.1.1    m1
192.168.1.2    c1
192.168.1.3    c2
```

3.4 計算ノードでの rsh のインストールと設定

1. 計算ノードには rsh サーバと rsh クライアントをインストールする。

```
# aptitude install rsh-client rsh-server
```

2. /etc/hosts の編集
マスタノードと同様に IP アドレスとホスト名を/etc/hosts に記述する（追記する）。

```
192.168.1.1    m1
192.168.1.2    c1
192.168.1.3    c2
```

3. /etc/hosts.equiv の編集
PC クラスタを構成しているノード間の通信は計算効率を考えて暗号化無しで行いたい。/etc/hosts.equiv にノードの IP アドレスまたはホスト名を記述（追記）すると、そのノードとは暗号化無しで通信を行うことができる。

```
192.168.1.1
192.168.1.2
192.168.1.3
```

4. 設定を有効にするために、inetd を再起動する。

```
# /etc/init.d/openbsd-inetd restart
```

3.5 マスタノードでの MPICH のインストールと設定

PC クラスタで並列計算を行う場合、並列計算ライブラリを用いて実行ファイルを作成する。今回は MPICH という MPI(Message Passing Interface) 実装の 1 つである MPICH を用いる。

1. MPICH のインストール

```
# aptitude install mpich-bin libmpich1.0-dev
```

2. MPICH の設定
MPICH は計算ノードのみ設定を行う。/etc/mpich/machines.LINUX に並列計算に用いるノードの IP アドレスまたはホスト名を記述する（追記する）。ここに記述されていないノードは使用されない。今回はマスタノードは計算させないので、記述しない。

```
192.168.1.2
192.168.1.3
```

3.6 マスタノードでの NFS のインストールと設定

並列計算を行う際は、実行ファイルやそれに用いられるソフトウェアを全ノードが持っていなければならない。NFS(Network File System) を用いると、任意のディレクトリを全ノードで共有することができる。ここではマスタノードの/home をすべての計算ノードと共有する設定を行う。

1. マスタノードに NFS サーバをインストールする。

```
# aptitude install nfs-kernel-server
```

2. /etc/exports の編集

   /etc/exports に共有させたいディレクトリと、共有を許可するノードの設定を行う。

```
/home c1(rw,sync) c2(rw,sync)
```

3. NFS サーバを再起動する。

```
# /etc/init.d/nfs-kernel-server restart
```

3.7 計算ノードでの NFS のインストールと設定

1. 計算ノードに NFS クライアントをインストールする。

```
# aptitude install nfs-common
```

2. /etc/fstab の編集
   起動時に NFS サーバにマウントするように設定する。/etc/fstab に以下のように追記する。

```
m1:/home /home nfs defaults,rw 0 0
```

3. NFS サーバのディスクにアクセスするために以下のコマンドを実行する。

```
# mount -a
```

3.8 マスタノードでの NIS のインストールと設定

NIS(Network Information Service) とは、ネットワーク上で全ての計算機で必要な情報を共有するサービスのことである。ユーザ名が同じでも使うマシンが違うと ID は一致せず、別のユーザと認識されてしまう。そこで、NIS を使い、異なるノード間でも同じユーザ ID、グループ ID が使用できるようにする。マスタノードが NIS サーバに、計算ノードが NIS クライアントにする。

1. マスタノードに NIS をインストールする。インストール中に、NIS ドメイン名の入力が求められるので、任意で入力する（ここでは nis-dom とする)。インストール後に/etc/defaultdomain を編集してもよい。セキュリティ上 DNS のドメイン名とは違うものにする。

```
# aptitude install nis
```

   次に、マスタノードが NIS サーバとなるように設定を行う。

2. /etc/init.d/nis の編集

```
NISSERVER=master
```

3. /etc/default/nis の編集

```
NISSERVER=master
```

4. NIS サーバが提供する情報（ユーザ、グループ）はマップと呼ばれる。下記のコマンドでマップの再構築を行う。

```
# /usr/lib/yp/ypinit -m
```

NIS サーバ名の追加を聞かれるが、通常は追加を行わないので、Crtl+D を押す。

5. NIS の再起動
設定を反映させるために NIS を再起動する。

```
# /etc/init.d/nis restart
```

3.9 計算ノードでの NIS のインストールと設定

1. 計算ノードに NIS をインストールする。

```
# aptitude install nis
```

インストール中に NIS ドメインを聞かれるので、マスタノードと同じ NIS ドメインを入力する。（ここでは nis-dom とした。）

2. /etc/init.d/nis の編集

```
NISSERVER=false
```

3. /etc/default/nis の編集

```
NISSERVER=false
```

4. /etc/yp.conf の編集（追記する）
NIS サーバのホスト名を ypserver という文字列に続けて記述する.

```
ypserver m1
```

5. /etc/passwd と/etc/group の編集
最後の行に以下を追加する。（これ以降の行にユーザ、グループを追加しない、という意味）

```
+::::::
```

6. /etc/nsswitch.conf の編集

```
passwd: nis files
group nis files
shadow: nis files
hosts: nis files dns
```

7. NIS の再起動

   設定を反映させるために NIS を再起動する。

```
# /etc/init.d/nis restart
```

3.10 NIS の動作確認

1. マスタノードでユーザを追加する。

```
# adduser test
```

2. 下記のコマンドでマップの再構築を行う。

```
# /usr/lib/yp/ypinit -m
```

   NIS サーバの追加を聞かれるが、通常は追加を行わないので、[Crtl]+[D] キーを押す。

3. root から test ユーザになる。

```
# su - test
```

4. 計算ノードにログインできることを確認する。

```
$ rsh slave1
```

4 PC クラスタとして動作するかのテスト（円周率の計算）

マスタノードでテストを行う。

1. まず、自分のホームディレクトリに計算させるマシンのリストファイルを作成する（ファイル名は任意)。

```
$ cd
$ vi machinelist
```

2. ファイルは以下のように計算させるマシンのホスト名を記述する。

```
c1
c2
```

3. gcc とライブラリをインストールする。（注：マスタノードと計算ノードに以下のパッケージのインストールを行う必要がある。）

```
# apt-get install gcc libc6-dev
```

4. サンプルプログラムを自分のホームディレクトリにコピーする。

```
$ cp /usr/share/doc/libmpich1.0-dev/examples/cpi.c ~/
```

5. コンパイルする。

```
$ mpicc cpi.c
```

6. 実行ファイルを生成した後に並列計算を実行する。-np の後に計算に使用するプロセス数を指定する。今回の場合、プロセッサ数は 3 とした。また、-machinefile の後には計算に使用するノードのリストファイルを指定する。

```
$ mpirun -np 3 -machinefile machinelist a.out
```

計算に用いたマシン名と円周率の計算結果が表示されていることを確認する。

5 計算に用いたマシン名、または計算結果が表示されない場合の問題の切り分け方

1. rsh の確認
   以下のコマンドで、計算ノードと接続できているか確認する。

   ```
   $ rsh c1
   ```

   これが失敗した場合、rsh の設定に問題があると考えられる。

2. マウント状況の確認
   計算ノードで以下のコマンドを入力し、マスタノードと/home ディレクトリの共有ができているか確認する。

   ```
   $ df
   ```

   マスタノードと共有できていない場合は NFS の設定に問題があると考えられる。