

NGX 計算機

ユーザマニュアル

NGX-LM^{DT}- LX20S/RX20S/WX20S

NGX-LM^{RM}-LX20S/RX20S/WX20S



株式会社 TS テクノロジー

目 次

1 NGX 計算機をお使いになる前に	5
1-1 NGX 計算機について	5
1-2 ご注意していただきたいこと	5
1-3 免責事項(保証内容については保証書をご参照ください。)	5
2 起動から終了まで.....	6
2-1 パワーオン.....	6
2-2 ログイン	6
2-3 ログアウト.....	6
2-4 シャットダウン	6
3 ユーザ環境の設定	8
3-1 ユーザパスワードの変更	8
3-2 ユーザシェルの変更	8
3-3 ユーザの新規作成	9
3-4 ユーザの変更(su, sudo)	9
4 ファイル・ディレクトリの操作	10
4-1 ディレクトリ内容の確認(ファイル一覧表示)	10
4-2 ディレクトリの作成	10
4-3 ディレクトリの移動	11
4-4 ファイル・ディレクトリの名前変更、移動	11
4-5 ファイル・ディレクトリのコピー	12
4-6 ファイル・ディレクトリの削除	12
4-7 ファイルリンクの作成	12
4-8 ファイルサイズの確認	13
4-9 ファイルの圧縮と伸張	13
4-10 ファイルの検索	14
4-11 ファイル権限の変更	14
4-12 所有ユーザ/グループの変更	14
4-13 文字列の検索.....	14
4-14 ファイル内容の比較.....	14
5 プロセス管理.....	15
5-1 現在実行中のプロセスリストの表示(top).....	15
5-2 プロセスの確認	16
5-3 プロセスの停止(kill/killall)	16
5-4 プロセス時間の計測	16

6 システム情報の表示	18
6-1 日付・時刻の確認	18
6-2 CPU 情報の表示	20
6-3 メモリ情報の表示	21
6-4 ディスク容量の確認	22
6-5 ネットワーク情報の表示	22
6-6 ethtool コマンド	23
7 システム管理	24
7-1 ホスト名の設定	24
7-2 IP アドレス	24
7-3 ネームサーバ	25
7-4 時刻設定・NTP	25
8 クラスタの管理	26
8-1 各ノードへのアクセス	26
8-2 ボンディットネットワーク	26
9 アプリケーションの操作	27
9-1 Gaussian	27
9-2 GaussView	28

1 NGX 計算機をお使いになる前に

1-1 NGX 計算機について

この度は当社 NGX 計算機をご購入いただき、誠にありがとうございます。

本製品を安全に正しく使うために重要な事項が、本マニュアルに記載されています。

必ずお読みになり、正しくお使いください。

1-2 ご注意していただきたいこと

本製品に使われているプロセッサ(CPU)の処理能力は次のような条件によって違いが現れます。

- ・周辺機器を接続して本製品を使用する場合
 - ・本製品を低速度のネットワークに接続して使用する場合
 - ・気圧が低い高所にて本製品を使用する場合(目安として、標高1000メートル以上をお考えください。)
 - ・気温5～30℃(高所の場合は25℃)の範囲を超えるような外気温の状態では本製品を使用する場合
- ある状況下においては、本製品は自動的にシャットダウンする場合があります。これは、当社が推奨する設定、使用環境の範囲を超えた状態で本製品が使用された場合、お客様のデータ喪失、破損、本製品自体に対する損害の危険を低減するための保護機能です。なお、このような状況下でのデータ喪失、破損などの危険に備えて、必ず定期的にデータを外部記録装置に保存してください。

1-3 免責事項（保証内容については保証書をご参照ください。）

- ・火災、地震、第三者による行為、その他の事故、お客様の故意または過失、誤用、その他異常な条件下でのご使用による損害に関して、当社は一切責任を負いません。
- ・本製品の使用または使用不能から生ずる二次的な損害(事業利益の損失、事業の中断、記録内容の変化・消失など)に関して、当社は一切責任を負いません。
- ・この取扱説明書の記載内容に従わない使い方によって生じた損害に関して、当社は一切責任を負いません。
- ・接続機器との組み合わせによる誤動作などから生じた損害に関しては当社は一切責任を負いません。
- ・本製品に記録されているデータの破損または消失が発生した場合、原因、損害の内容、程度に関わらず当社は一切責任を負いません。また、データの復旧作業については、当社ではお受けしていません。重要なデータについては、別のメディアにバックアップを作成してください。
- ・本製品は日本国内仕様です。本製品の日本国外での使用に関し、当社は一切責任を負いません。
- ・本製品が原因で起こる債務不履行または不法行為に基づく損害賠償の責任は、弊社に故意又は過失があった場合を除いて、本製品の購入代金と同額を上限とさせていただきます

2 起動から終了まで

2-1 パワーオン

NGX 計算機の筐体前面にある、マークのついたスイッチを押し、電源を ON にします。電源 ON 後、NGX 計算機は自動的に起動します。

2-2 ログイン

起動できたら、まず最初にログイン操作を行います。ログイン操作をするためには、ユーザ ID とパスワードが必要となります。

NGX 計算機の工場出荷時設定では、以下のユーザ ID・パスワードが設定されています。一般ユーザは、計算機上で作業を行うためのユーザ、管理者(root)は計算機の管理を行うための権限を有するユーザです。なお、パスワードは適宜変更して下さい。

(一般ユーザ)	
ユーザ ID :	user
パスワード :	user
(管理者)	
ユーザ ID :	root
パスワード :	nopasswd

ユーザ ID をキーボードから入力し、[Enter] キーを打ちます。

正しくログイン操作ができないときは、ログインが失敗したという趣旨のメッセージが表示されます。もう一度、ユーザ ID の入力からやり直して下さい。

2-3 ログアウト

計算機の使用を止めるときにはログインとは逆に「ログアウト」します。ログアウトはあくまで、ユーザが計算機を使うことをやめること表しており、計算機を停止させるわけではありません。

```
% exit
```

と打ちます。

2-4 シャットダウン

計算機を終了するときには、電源スイッチを押して電源を OFF しないでください。いきなり電源を OFF にすると、重要なデータが破壊される場合があります。以下の手続きに従って終了して下さい。

コマンドラインで電源を OFF にします。

電源を落すには root になる必要があります。まず su でルートユーザに切り替えてから、終了のためのコマンド shutdown を実行します。

```
% su -l  
password: (rootのパスワードを入力)  
# shutdown -h now
```

この後、OS が必要な後始末をした後に自動的に電源が切断されます。

なお su とは Switch User の意味のコマンドで、一時的に違うユーザになりかわるためのコマンドです。通常は一時的に管理者の権限が必要なときにユーザ名の指定なしに su を実行します。(su とは switch user の略です) これで一時的に root ユーザになります。一時的なユーザから元のユーザに戻る時は exit コマンドを実行します。

再起動については

```
% reboot
```

または、

```
% shutdown -r now
```

を用います。

3 ユーザ環境の設定

3-1 ユーザパスワードの変更

パスワードは、いつでも `passwd` コマンドを用いて変更することができます。

計算機上で、

```
% passwd
```

とすると

```
Changing password for XXXXXX  
Old password:
```

と聞かれます。(XXXXXX のところは、ユーザ ID が入ります)

なお、パスワードの変更に関わるこれ以降の入力文字は、機密保持のため画面に表示されません。ここで、これまで使っていたパスワードを入力し [Enter] キーを打ちます。次に

```
New password:
```

と聞かれますから、新しいパスワードを入力して [Enter] キーを打ちます。さらに確認のため

```
Re-enter new password:
```

と聞かれますので、もう一度同じものを入力し、[Enter] キーを打ちます。変更が成功すれば、

```
Password changed
```

という表示がされます。しかし、もし 2 度の入力が異なっていたり、パスワード文字列が短すぎたり、パスワードとして必要な条件が満たされていない場合には、その旨メッセージが表示されパスワードは変更されません。原因をよく確かめてから再度実行します。

パスワードに、容易に想像できるようなもの(自分の名前、誕生日、アイドルの名前等)を設定するのはセキュリティ上好ましくありません。他人に推測されにくい文字や数値の組み合わせを設定すると共に、定期的に変更するようにして下さい。

3-2 ユーザシェルの変更

シェルは、ユーザに計算機を利用するためにインタフェースを提供するプログラムです。NGX 計算機の工場出荷時設定では、

```
% /bin/tcsh
```

が設定されています。

利用するシェル(ログインシェル)を変更する際には、`chsh` コマンドを使用します。

```
% chsh
```

利用できるシェルには、`/bin/tcsh` の他、`/bin/bash`、`/bin/dash` などがあります。ご利用環境に合わせて設定して下さい。また、シェルを設定する場合は、常にフルパスを指定して下さい。

3-3 ユーザの新規作成

ユーザを新規に作成(追加)する場合は、以下のようにします。**[newuser]**の部分に新規ユーザ名を指定して下さい。

```
% useradd [newuser]
```

ユーザ作成後、パスワードを設定します。

```
% passwd [newuser]
```

3-4 ユーザの変更(su, sudo)

su を用いて、他のユーザへの変更を行います。

```
% su
```

変更したいユーザを指定しない場合は、管理者(root)になります。変更するには、変更しようとするユーザのパスワードが必要となります。ただし、root ユーザから他のユーザに変更する際は、パスワードが不要となります。

ユーザを変更せずにコマンドを実行した場合は sudo コマンドを用います。

```
% sudo [実行したいコマンド名]
```

root ユーザとして、コマンドを実行します。root ユーザのパスワードが必要となります。

4 ファイル・ディレクトリの操作

4-1 ディレクトリ内容の確認（ファイル一覧表示）

ディレクトリ内容の確認(ディレクトリ内のファイルの一覧表示)を行うには **ls** コマンドを用います。

```
% ls
Makefile  hello  hello.c  hello.f  hello2  hello2.c
```

代表的なオプションによる表示は以下のようになります。

-l オプションは、詳細表示を行います。

-a オプションは、隠しファイルの表示を行います。

ファイル名が.(ドット)で始まるファイルは隠しファイルとなります。.(ドット)はカレントディレクトリ(現在のディレクトリ)を示しています。..(ドットドット)は親ディレクトリ(一つ上のディレクトリ)を示しています。

```
% ls -al
total 43736
drwxrwxr-x   2 user users    4096 May 11 17:51 .
drwx----- 46 user users    4096 May  9 16:32 ..
-rw-rw-r--   1 user users    6788 May 11 17:51 .hidden
-rw-rw-r--   1 user users     101 Apr  6  2010 Makefile
-rw-rw-r--   1 user users 44645376 May 11 15:19 file.data
-rwxrwxr-x   1 user users    6788 Apr  6  2010 hello
-rw-rw-r--   1 user users     64 Apr  6  2010 hello.c
-rw-rw-r--   1 user users     43 Apr  6  2010 hello.f
-rwxrwxr-x   1 user users    6789 Apr  6  2010 hello2
-rw-rw-r--   1 user users     65 Apr  6  2010 hello2.c
```

-h オプションは、ファイルサイズに補助単位を付加し、見やすい形式で表示を行います。

```
total 43M
-rw-rw-r--   1 user users   101 Apr  6  2010 Makefile
-rw-rw-r--   1 user users  43M May 11 15:19 file.data
-rwxrwxr-x   1 user users   6.7K Apr  6  2010 hello
-rw-rw-r--   1 user users    64 Apr  6  2010 hello.c
-rw-rw-r--   1 user users    43 Apr  6  2010 hello.f
-rwxrwxr-x   1 user users   6.7K Apr  6  2010 hello2
-rw-rw-r--   1 user users    65 Apr  6  2010 hello2.c
```

4-2 ディレクトリの作成

ディレクトリを作成する場合は、**mkdir** コマンドを使用します。

```
% mkdir temp
% ls -dl temp
drwxrwxr-x   2 user users 4096 May 11 18:05 temp
```

-p オプションは、階層構造を持つ複数のディレクトリをまとめて作成します。

```
% mkdir -p class1/class2
% ls -l class1
total 8
drwxrwxr-x   2 user users 4096 May 11 18:16 class2
```

4-3 ディレクトリの移動

現在のディレクトリを表示するには **pwd** コマンドを使用します。

ディレクトリを移動するには **cd** コマンドを使用します。

```
% pwd
/home/user/test
% cd temp
% pwd
/home/user/test/temp
```

一つ前のディレクトリに戻るには、引数に**-**(マイナス)を指定します。

```
% cd -
% pwd
/home/user/test
```

とします。

親ディレクトリに移動するには、引数に**..**(ドットドット)を指定します。

```
% cd ..
% pwd
/home/user
```

引数に何も指定しない場合、もしくは、**~**(チルダ)を指定した場合は、ユーザのホームディレクトリ (**/home/user**)に移動します。

```
% cd
% pwd
/home/user
% cd ~
% pwd
/home/user
```

4-4 ファイル・ディレクトリの名前変更、移動

ファイルの名前を変更、またはファイルを別のディレクトリに移動するには **mv** コマンドを使用します。

ファイル名を変更するには、変更前のファイル名と変更後のファイル名を指定します。以下の例では **abc** が変更前のファイル名、**def** が変更後のファイル名となります。

```
% mv abc def
```

ファイルを別のディレクトリに移動するには、移動対象のファイルと移動先のディレクトリもしくはファイル名を指定します。

```
% mv def temp/
% ls -al temp
total 20
drwxrwxr-x  2 user users 4096 May 11 18:49 .
drwxrwxr-x  4 user users 4096 May 11 18:49 ..
-rw-rw-r--  1 user users   0 May 11 18:14 def
```

4-5 ファイル・ディレクトリのコピー

ファイルをコピーするには **cp** コマンドを使用します。

```
% cp file.data file2.data
% ls -lh*.data
-rw-rw-r--  1 user users 43M May 11 15:19 file.data
-rw-rw-r--  1 user users 43M May 11 18:52 file2.data
```

ディレクトリをコピーする場合は、**-r** オプションを使用します。**-r** オプションは再帰的コピーの指定で、ディレクトリ内に含まれるファイル、サブディレクトリ、サブディレクトリ内に含まれるファイル及びディレクトリの全てがコピーされます。

```
% cp -r temp temp2
% ls -al temp*
temp:
total 20
drwxrwxr-x  2 chemist chemist 4096 May 11 18:49 .
drwxrwxr-x  5 chemist chemist 4096 May 11 18:57 ..
-rw-rw-r--  1 chemist chemist    0 May 11 18:14 def

temp2:
total 20
drwxrwxr-x  2 chemist chemist 4096 May 11 18:57 .
drwxrwxr-x  5 chemist chemist 4096 May 11 18:57 ..
-rw-rw-r--  1 chemist chemist    0 May 11 18:57 def
```

4-6 ファイル・ディレクトリの削除

ファイルを削除するには **rm** コマンドを使用します。

```
% rm def
```

ディレクトリ及びディレクトリに含まれる全てのファイルを削除するには、**-rf** オプションを使用します。

```
% rm -rf temp2
```

4-7 ファイルリンクの作成

ファイルリンクには、「シンボリックリンク」と「ハードリンク」の2種類があります。「シンボリックリンク」は、現在あるファイルもしくはディレクトリに別名（リンク）を与え、ファイルを別名で参照できるようにする仕組みです。シンボリックリンクを削除しても元のファイルは削除されませんが、元のファイルを削除するとシンボリックは無効となり利用できなくなります。

「ハードリンク」はシンボリックリンク同様ファイルを別名で参照できるようにする仕組みですが、元のファイルを削除した場合でもハードリンクを削除しない限り利用できます。

シンボリックリンクを作成するには **ln** コマンドを使用し、**-s** オプションを使用します。1 つ目の引数にリンク先のファイル名、2 つ目の引数に別名を指定します。

```
% cd temp/
% ls
def
% ln -s ../file.data link.data
```

```
% ls -l
total 8
-rw-rw-r-- 1 chemist chemist 0 May 11 18:14 def
lrwxrwxrwx 1 chemist chemist 12 May 11 19:09 link。data -> ../file.data
```

ハードリンクを作成するには、単に **ln** コマンドのみを使用します。

```
% ln ../file2.data link2.data
% ls -l
total 43660
-rw-rw-r-- 1 chemist chemist 0 May 11 18:14 def
lrwxrwxrwx 1 chemist chemist 12 May 11 19:09 link。data -> ../file.data
-rw-rw-r-- 2 chemist chemist 44645376 May 11 18:52 link2.data
```

4-8 ファイルサイズの確認

ファイルサイズを確認するには、**ls** コマンドを用いる方法と **du** コマンドを用いる方法があります。

ls コマンドを用いる場合には **-l** オプションを指定し詳細表示を行うことでファイルサイズを確認できます。また、**-h** オプションを使用することで、補助単位を付加した見やすい形式で表示することができます。

```
% ls -l
total 87376
-rw-rw-r-- 1 chemist chemist 101 Apr 6 2010 Makefile
drwxrwxr-x 3 chemist chemist 4096 May 11 18:16 class1
-rw-rw-r-- 1 chemist chemist 44645376 May 11 15:19 file.data
-rw-rw-r-- 1 chemist chemist 44645376 May 11 18:52 file2.data
-rwxrwxr-x 1 chemist chemist 6788 Apr 6 2010 hello
-rw-rw-r-- 1 chemist chemist 64 Apr 6 2010 hello.c
-rw-rw-r-- 1 chemist chemist 43 Apr 6 2010 hello.f
-rwxrwxr-x 1 chemist chemist 6789 Apr 6 2010 hello2
-rw-rw-r-- 1 chemist chemist 65 Apr 6 2010 hello2.c
drwxrwxr-x 2 chemist chemist 4096 May 11 19:13 temp

% ls -lh
total 86M
-rw-rw-r-- 1 chemist chemist 101 Apr 6 2010 Makefile
drwxrwxr-x 3 chemist chemist 4.0K May 11 18:16 class1
-rw-rw-r-- 1 chemist chemist 43M May 11 15:19 file.data
-rw-rw-r-- 1 chemist chemist 43M May 11 18:52 file2.data
-rwxrwxr-x 1 chemist chemist 6.7K Apr 6 2010 hello
-rw-rw-r-- 1 chemist chemist 64 Apr 6 2010 hello.c
-rw-rw-r-- 1 chemist chemist 43 Apr 6 2010 hello.f
-rwxrwxr-x 1 chemist chemist 6.7K Apr 6 2010 hello2
-rw-rw-r-- 1 chemist chemist 65 Apr 6 2010 hello2.c
drwxrwxr-x 2 chemist chemist 4.0K May 11 19:13 temp
```

du コマンドでは

4-9 ファイルの圧縮と伸張

Linux では複数の圧縮形式がサポートされています。ここでは代表的な圧縮形式である **gzip**、**bzip2**、**xz** の 3 つを説明します。

Linux での圧縮は伝統的に、複数のファイルを 1 つのファイル形式に変換する **tar** コマンドとともに使われます。**tar** コマンドでは上の 3 つの圧縮形式をサポートし、それぞれ **-z** オプション(**gzip**)、**-j** オプション

(bzip2)、-y オプション(xz)を指定することで圧縮が可能です。圧縮形式のオプションを指定しない場合 tar コマンドは、ファイルの結合だけを行います。

4-10 ファイルの検索

検索したいファイルなどの名前が分かっている場合は、find コマンドに「-name」判別式を利用します。例えば、sample.txt というファイルをファイル・システム内から検索したい場合は、

```
% find / -name sample.txt
```

と入力します。検索中、見つかったファイルやディレクトリが随時一覧表示されていきます。

4-11 ファイル権限の変更

ファイルやディレクトリのパーミッションを変更します。

```
% chmod [-cfvR] モード ファイル
```

4-12 所有ユーザ/グループの変更

ユーザは、何らかのグループに所属しています。第一に登録されているグループをプライマリーグループといます。グループを変更することにより、諸所の権限の管理ができます。

```
(hogeユーザをgroup1に所属させる場合)
% usermod -G group1 hoge

(hogeユーザをgroup1とgroup2に所属させる場合)
% usermod -G group1, group2 hoge
```

4-13 文字列の検索

テキスト・ファイルから特定の文字列を含む(論理)行を抜き出すには、「grep」コマンドを用いる。grep の後に検索したい文字列を指定し、さらにその後に検索対象とするファイルを指定する。

```
% grep "検索したい文字列" ファイル名
```

4-14 ファイル内容の比較

2つのファイル file1 file2 を行毎に比較する

```
(ファイルhoge1とhoge2を比較)
% diff hoge1 hoge2
```

5 プロセス管理

5-1 現在実行中のプロセスリストの表示(top)

```
% top [-] [d delay] [p pid] [q] [c] [C] [S] [s] [i] [n iter] [b]
```

CPU をたくさん使っているプロセスを表示する。

top コマンドは、CPU をたくさん使っている順に、システムのプロセスを一定間隔で表示します。またプロセスを操作するための対話的なインタフェースがあり、結果を、CPU の使用度、メモリの使用量、実行時間などでソートしたりすることができます。

-d delay

画面を更新する間隔(秒)を「delay」に指定します。実行後は、対話コマンド「s」で変更することができます。デフォルトの更新間隔は 5 秒です。

-p pid

「pid」に指定したプロセス ID のプロセスのみを監視します。このフラグは 20 回まで指定することができます。

-q

遅延時間なしで更新を続けます。スーパーユーザ特権を持って実行すると、最高の優先度で実行します。

-c

コマンド名だけでなく、コマンドラインを表示します。

-C

各々の CPU の状態だけでなく全体の CPU の統計も表示します。SMP(CPU を複数搭載した)システムにのみ有効です。

-S

累積モードを有効にします。ゾンビプロセスの分も含めて、利用した CPU 時間とともに表示します。

-s

セキュアモードで実行して、対話的コマンドの実行を禁止します。

-i

アイドルプロセスやゾンビプロセスを無視します。

-n iter

更新回数を「iter」に指定します。指定した回数だけ表示を更新して終了します。

-b

バッチモードを指定します。出力を他のプログラムやファイルに送る時に使用します。この時 top コマンドはコマンドライン入力を受け付けません。オプション「n」で指定した回数だけ表示を繰り返すか、[Ctrl]+[C]キーを押すことなどによって中断されるまで実行を続けます。

5-2 プロセスの確認

```
% ps (option)
```

プロセスの状態を一覧表示する。オプションを省略すると、コマンド実行ユーザが起動しているプロセスのみを全て表示する。プロセス状態の表示。

-a

ユーザのプロセスの状態を表示する。

-e

全てのプロセスの状態を表示する。

-f

プロセスの親子関係をツリー表示する。

-l

プロセス詳細情報を表示する。

-u

プロセス実行ユーザ名と開始時刻も表示する。

-x

デーモンなど制御端末を持たないプロセスも表示する。

5-3 プロセスの停止(kill/killall)

```
% kill [signal] [job]
```

プロセスやジョブを強制終了させる。

```
% kill -l
```

シグナル名の一覧を表示する。

```
% killall
```

指定したコマンドを実行している全てのプロセスにシグナルを送る。シグナルの指定がなければ SIGTERM を送る。

5-4 プロセス時間の計測

time でコマンドの実行時間を測定する。

```
% time command
```

実際の使用例

```
% time cc -c myprogram.c
0.600u 0.704s 0:03.77 13.1%      1+109k 88+15io 66pf+0w
```

0.600u ユーザモードの CPU 時間。(user time)

0.704s システムモードの CPU 時間。(system time)

0:03.77 処理にかかった実時間。(real time)

13.1% コマンド実行中の CPU 使用率。
1+109k テキストセグメントで平均 1K バイト、データセグメントで平均 109K バイトのメモリが使用された。
88+15io ディスクからの読み出しが 88 回、ディスクへの書き込みが 15 回行われた。
66pf+0w 実行中、ページフォルトは 66 回起こったが、スワップアウトは 1 度も起こらなかった。

6 システム情報の表示

6-1 日付・時刻の確認

```
% date
```

日付や時刻を表示、設定する。

```
% date [-u] [-d datestr] [+format] [-s setstr] [MMDDHHmm[[CC]YY][.ss]]
```

-u

協定世界時を使用する。

-d datestr

datestr に指定された日付と時刻を表示する。

+format

+(プラス)で始まる引数は、日付と時刻の表示方法を指定するフォーマット文字列です。フォーマット文字列には、以下の表に示す変換文字列と、任意のテキストを含むことができます。

-s setst

日付と時刻を設定する。setstr には一般的に分かる形であれば形式を問わない。

例: 2000/01/08 10:11:21

MMDDHHmm[[CC]YY][.ss]

日付を指定し、別のフォーマットで表示する。MM は月、DD は日、HH は時、mm は分、CC は西暦の上 2 けた、YY は西暦の下 2 けた、ss は秒を表します。

※ 注意

日付 / 時刻の設定はスーパーユーザだけが行えます。

JST

グリニッジ標準時からの時差をもつための情報で、地域によって決まるタイムゾーンというものがあります。日本のタイムゾーンは JST という定義になります。

%H 時 (00～23)

%I 時 (01～12)

%k 時 (0～23)

%l 時 (1～12)

%M 分 (00～59)

%p AM あるいは PM のロケール(国や地域に合わせた文字列)

%r 12 時間形式の時刻 (HH:mm:ss [AP]M)

%s 1970-01-01 00:00:00 UTC からの秒数

%S 秒 (00～61)

%T 24 時間形式の時刻 (HH:mm:ss)

%a	ロケールによる省略形の曜日の名前 (Sun～Sat)
%A	ロケールによる完全に表記した曜日の名前 (Sunday～Saturday)
%b	ロケールによる省略形の月の名前 (Jan～Dec)
%B	ロケールによる完全に表記した月の名前 (January～December)
%c	ロケールによる日付と時刻 (Sat Nov 04 12:02:33 EST 1989)
%d	日(月内通算日数) (01～31)
%D	日付 (MM/DD/YY)
%j	年内通算日数 (001～366)
%m	月 (01～12)
%w	週のうちの曜日(0～6)で0が日曜日に対応
%x	ロケールによる日付の表現 (MM/DD/YY)
%y	西暦の下2けた (00～99)
%Y	年 (1970～)

注意

「% S」つまり秒では、うるう秒を表示できるように、00 から 61 までの値で返されます。

ロケール

ロケールとは、地域情報のことであり、言語環境、日付や時刻の表現方法、通貨の単位などを決めるために用いられます。

フォーマット文字列と変換文字列

フォーマット文字列とは、date コマンドにより日付／時刻を表示する場合に、どのような形式で表示するかを決めるための表現方法で、変換文字列は、日付／時刻を設定する場合に用いるフォーマット(形式)を表記するための文字列のことです。

```
% DATETIME
```

スーパーユーザは date コマンドを使って、日付／時刻の設定を行うことができます。

ディスク容量の確認

ディレクトリのディスク使用量を表示する。

```
% du [-a] [-k] [-m] [-s] [DIR...]
```

du コマンドは、DIR...に指定したディレクトリのディスク使用量を表示します。表示される値は、すべてのサブディレクトリおよびファイルのディスク使用量を含みます。

-a

ディレクトリのディスク使用量だけでなく、その中に含まれるすべてのファイルのディスク使用量を、それぞれ表示します。

-k

ディスク使用量をキロバイト(KB)単位に表示します。

-m

ディスク使用量をメガバイト(MB)単位に表示します。

-s

DIR。。。に指定したディレクトリの中に含まれる、サブディレクトリやファイルのディスク使用量を計算しません。

引数[DIR。。。]

ディスク使用量を表示するディレクトリのパスをスペースで区切って指定します。ワイルドカードによる指定も可能です。省略時には、カレントディレクトリのディスク使用量を表示します。

ディレクトリのディスク使用量をキロバイト単位で表示する。

```
% du -k dir01
1384 dir01
```

ディレクトリ「dir01」のディスク使用量は 1384KB

ディレクトリ内のファイルのディスク使用量も表示する。

```
% du -ak dir01
4 dir01/fl001
4 dir01/fl002
4 dir01/fl003
4 dir01/fl004
4 dir01/fl005
4 dir01/fl006
4 dir01/fl007
4 dir01/fl008
4 dir01/fl009
4 dir01/fl010
4 dir01/x001
4 dir01/x002
4 dir01/fa001
4 dir01/fa002
4 dir01/fx001
4 dir01/fx002
4 dir01/fo001
1312 dir01/fd
dir01
$
```

6-2 CPU 情報の表示

cpuinfo を出力

```
% cat /proc/cpuinfo
```

出力結果例

```
% cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 15
model         : 4
model name    : Intel(R) Pentium(R) D CPU 3.20GHz
stepping      : 4
cpu MHz       : 2800.000
cache size    : 1024 KB
physical id   : 0
siblings      : 2
```

```

core id      : 0
cpu cores   : 2
apicid      : 0
initial apicid : 0
fpu         : yes
fpu_exception : yes
cpuid level : 5
wp          : yes
flags       : fpu vme de pse tsc msr pae mce cx8 apic mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc
pebs bts pni dtes64 monitor ds_cpl est cid cx16 xtpr
bogomips    : 6399.51
clflush size : 64
cache_alignment : 128
address sizes : 36 bits physical, 48 bits virtual
power management:

processor    : 1
vendor_id    : GenuineIntel
cpu family   : 15
model        : 4
model name   : Intel(R) Pentium(R) D CPU 3.20GHz
stepping     : 4
cpu MHz      : 2800.000
cache size   : 1024 KB
physical id   : 0
siblings     : 2
core id      : 1
cpu cores    : 2
apicid       : 1
initial apicid : 1
fpu          : yes
fpu_exception : yes
cpuid level   : 5
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc
pebs bts pni dtes64 monitor ds_cpl est cid cx16 xtpr
bogomips     : 6399.51
clflush size  : 64
cache_alignment : 128
address sizes  : 36 bits physical, 48 bits virtual
power management:

```

6-3 メモリ情報の表示

```
% cat /proc/meminfo
```

実行すると

```

% cat /proc/meminfo
MemTotal:      2018616 kB
MemFree:       148192 kB
Buffers:       605124 kB
Cached:        878036 kB
SwapCached:      0 kB
Active:        752120 kB
Inactive:      762956 kB
Active(anon):   10688 kB
Inactive(anon): 26916 kB
Active(file):   741432 kB

```

```

Inactive(file):    736040 kB
Unevictable:        0 kB
Mlocked:           0 kB
SwapTotal:         2096472 kB
SwapFree:          2096472 kB
Dirty:             4 kB
Writeback:         0 kB
AnonPages:         31924 kB
Mapped:            7248 kB
Shmem:             5680 kB
Slab:              307636 kB
SReclaimable:      217720 kB
SUnreclaim:        89916 kB
KernelStack:       1312 kB
PageTables:        6024 kB
NFS_Unstable:       0 kB
Bounce:            0 kB
WritebackTmp:      0 kB
CommitLimit:       3105780 kB
Committed_AS:      256376 kB
VmallocTotal:      34359738367 kB
VmallocUsed:        549852 kB
VmallocChunk:      34359170564 kB
HardwareCorrupted:  0 kB
AnonHugePages:     2048 kB
HugePages_Total:   0
HugePages_Free:    0
HugePages_Rsvd:    0
HugePages_Surp:    0
Hugepagesize:      2048 kB
DirectMap4k:       9088 kB
DirectMap2M:       2052096 kB

```

メモリの場合は「% cat /proc/meminfo」と実行すると確認できます。

6-4 ディスク容量の確認

ファイル・システムについて、使用領域と空き領域のサイズを表示します。標準での表示はファイル・システム名、全容量、使用量、残量、使用割合、マウント・ポイント。

```

% df
Filesystem          1k-blocks      Used Available Use% Mounted on
/dev/hda2            2732745    1300968   1290456   50% /
/dev/hdb1            998013      23348    923108    2% /usr/local

```

6-5 ネットワーク情報の表示

ifconfig コマンドは、おもに UNIX 系 OS や Linux で使用されるネットワーク環境の状態確認／設定のためのコマンドです。ホストに設置された NIC(ネットワーク・インターフェイス)に対し、IP アドレスやサブネットマスク、ブロードキャストアドレス、そのほかの基本的な設定を行うとともに、現在の設定の確認もできます。

```

% ifconfig
eth0      Link encap:Ethernet  HWaddr 00:80:00:48:AA:88

```

```

BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
Interrupt:5 Base address:0x120

lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        UP LOOPBACK RUNNING MTU:3924 Metric:1
        RX packets:16 errors:0 dropped:0 overruns:0 frame:0
        TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0

```

eth0 は、対応するネットワーク・ドライバが正常にロードされていれば、**OS** が自動的に認識してくれるものです。**eth** の後に続く「0」「1」の番号は認識した順に割り振られます。

lo は「ローカルループバック」と呼ばれる特別な仮想インターフェイスです。ホスト自身を示しており、**OS** のプロトコルスタックが必ず提供することになっています。従って、まったく **NIC** を設置していない場合でも、この **lo** だけは存在します。また、対応する **IP** アドレスは必ず **127.0.0.1** が割り当てられます。

6-6 ethtool コマンド

ネットワークに接続できなかったり、接続はできるが通信に時間がかかることがあります。そのようなときは、**ethtool** コマンドでネットワークインターフェイスの状況を調べてみましょう。正常に動作していれば、以下のように表示されます。

```

% ethtool eth0
Settings for eth0:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Supports auto-negotiation: Yes
    Advertised link modes:  10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Advertised auto-negotiation: Yes
    Speed: 1000Mb/s
    Duplex: Full
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    Auto-negotiation: on
    Supports Wake-on: umbg
    Wake-on: g
    Current message level: 0x00000007 (7)
    Link detected: yes

```

7 システム管理

7-1 ホスト名の設定

再起動しても変更を有効にするためには、2箇所のファイルの変更が必要になります。

```
% vi /etc/hosts
127.0.0.1 localhost.localdomain localhost
xxx.xxx.xxx.xxx ホスト名

% vi etc/sysconfig/network
NETWORKING=yes
HOSTNAME=ホスト名
GATEWAY=xxx.xxx.xxx.xxx
```

7-2 IP アドレス

計算機のネットワーク設定ファイルは `/etc/sysconfig/network-scripts` 以下の `ifcfg-`インターフェース名で設定します。以下の設定は、`eth0` に対しての設定です。

```
% vi /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=static
BROADCAST=xxx.xxx.xxx.xxx
IPADDR=設定したいIPアドレスを記述します。
NETMASK=xxx.xxx.xxx.xxx
NETWORK= xxx.xxx.xxx.xxx
ONBOOT=yes
TYPE=Ethernet
```

このファイルでは、以下の様な設定が可能です。

DEVICE	ネットワークインタフェース(デバイス名)を指定
BOOTPROTO	ブート時などに IP アドレス情報を取得するためのプロトコルを指定。IP を指定する場合は' none '、DHCP を使用する場合は' dhcp 'を指定
BROADCAST	ブロードキャストを IP アドレスで指定
IPADDR	DEVICE で指定したネットワークインタフェースに設定する IP アドレス
NETMASK	サブネットマスクを指定
NETWORK	直接接続するネットワークを示す、ネットワーク IP アドレスを指定。
ONBOOT	ブート時にこのインタフェースを有効にするか指定。有効にする場合は' yes '、無向にする場合は' no '
GATEWAY	このインタフェースの GATEWAY を指定。設定しない場合は、この行は不用
TYPE	このインタフェースのタイプ(種類)を指定。イーサネットは' Ethernet '、モデムは' Modem '
USERCTL	一般ユーザーがこのインタフェースを制御できるか指定。root のみの場合は' no '、一般ユーザも含める場合は' yes '

7-3 ネームサーバ

ネーム・サーバーの IP アドレスは、`/etc/resolv.conf` ファイルに記述します。複数のネーム・サーバーを登録できます。

```
% vi /etc/resolv.conf
nameserver xxx.xxx.xxx.xxx    ← ネーム・サーバーのIPアドレス
nameserver xxx.xxx.xxx.xxx    ← 複数個の設定ができる
```

7-4 時刻設定・NTP

日付と時刻の設定にはルート権限が必要です。設定には「**date**」コマンドを「**--set**」オプションと共に用います。

```
% hwclock --systohc
```

また、インターネット上のタイム・サーバーから時刻情報を得て、正確な時刻をセットすることもできる。タイム・サーバーは Public NTP Time Servers の Web サイト <http://www.eecis.udel.edu/~mills/ntp/servers.html> に一覧があります。

時間を設定した後に上述の **hwclock** により書き換えを行います。

servers.html に一覧があります。時間を設定した後に上述の **hwclock** により書き換えを行います。

```
% ntpdate タイム・サーバーのホスト名またはIPアドレス
% hwclock --systohc
```

8 クラスタの管理

計算機がクラスタを構成している場合、下記の操作が可能です。

8-1 各ノードへのアクセス

各計算ノードへのアクセスは、**rsh** コマンドにてパスワードを入力することなくログインすることができます。

```
% rsh 計算機のホスト名
```

8-2 ボンディットネットワーク

ボンディットネットワークを使うと、複数の LAN ケーブルの差込口 (NIC) を一つのデバイスとして管理することができるようになり、冗長性を持たせたり負荷分散をさせることができるようになります。たとえば NIC2 つを 1 つのデバイスにした場合、どちらかの LAN ケーブルが抜けたり、スイッチやサーバのポートが壊れたりしてももう片方が繋がっているので、完全に通信できない状態になる事を回避できたりすることができるようになります。ボンディットに関する設定は、

```
% vi /etc/sysconfig/network-scripts/ifcfg-bond0
```

にて行います。ボンディット設定ではない通常の **ifcfg-eth0** のように、IP アドレス等を設定することができます。本計算機では、**eth0** と **eth1** をボンディットしており、デフォルト設定は

```
% cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

```
% cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

となっています。

9 アプリケーションの操作

計算機に各アプリケーションがインストールされている場合、下記手順により各アプリケーションの操作が可能です。

9-1 Gaussian

Gaussian を実行するには、引数に入力ファイルを指定して、以下の実行コマンドを実行してください。入力ファイルには、拡張子 **.com** 又は **.gjf** が付加されている必要があります。**Gaussian** の実行が完了すると、同階層のディレクトリに、**.log** ファイルや、入力ファイル内で指定している場合、**.chk** ファイル等が生成されます。

※ **Gaussian03** の場合

```
% g03 入力ファイル名 (.com又は.gjf)
```

※ **Gaussian09** の場合

```
% g09 入力ファイル名 (.com又は.gjf)
```

※ **Gaussian16** の場合

```
% g16 入力ファイル名 (.com又は.gjf)
```

上記のコマンド実行後、コントロールキーと **z** を同時に押す(**Ctrl-z**)ことにより、**Gaussian** の中断ができます。**Gaussian** の中断中に、

```
% bg
```

とすることで、**Gaussian** がバックグラウンドで実行され、ターミナルやプロンプトの利用が可能になります。

このバックグラウンドでの実行は、**Gaussian** のコマンド実行時に、

```
% g16 入力ファイル名 (.com又は.gjf) &
```

などとして、入力ファイル名の後にスペースを空けて**&**(アンパサンド)を入力することでも可能です。

Gaussian が実行中であるかは、

```
% top
```

や、

```
% w
```

で確認できます。

Gaussian の環境設定は、`~/.cshrc`(`csh`, `tcsh` の場合), `~/.bashrc`(`bash` の場合)のファイルに設定されています。設定内容は以下で確認することができます。

```
% cat ~/.cshrc (又は.bashrc)
```

9-2 GaussView

GaussView を実行するには、XWindow 上のターミナルソフト(`xterm` など)より、以下のコマンドを実行してください。

```
% gv
```

GaussView の実行には、XWindow が立ち上がっている必要があります。XWindow が実行されていない場合は、

```
% startx
```

として実行することができます。

GaussView の環境設定は、`~/.cshrc`(`csh`, `tcsh` の場合), `~/.bashrc`(`bash` の場合)のファイルに設定されています。設定内容は以下で確認することができます。

```
% cat ~/.cshrc (又は.bashrc)
```