

# SWANを使用した 波浪推算の実行手順

### 長岡技術科学大学 建設工学専攻 水工学研究室 松永 由希子



※ Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.





# ソフトのインストールの前に





# Linuxにログインして操作





X Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.



### Linuxにログイン①





# Linuxにログイン②

	SSHZE 🔲 🗌 🗌
	ログイン中: estuary5
Weistwary622 - Tera Tera VT シールド 編集章 副業章 副学校 ひとわールロールロールロールロールの切り	認証が必要です。
	ユーザ名( <u>N</u> ): suiko
	1 (プロ)
I I I I I I I I I I I I I I I I I I I	
バスフレーズ(2) *******	
ロエージェント和述する②	
③ プレインテキストを使うQ)	<ul> <li>ブレインテキストを使う(L)</li> </ul>
O BSA/DSAIRE(M)	
○rbsels (SSH1)地域) ローフルのユーザキの♪ オフト級の)	○ RSA/DSA鍵を使う 秘密鍵(K):
	Orbosts(SSH1)を使う ローカルのフーザタ(II)
0K [#8595(0)]	
	ホスト鍵(E):
M	○チャレンジレスポンス認証を使う(キーボードインタラクティブ)(©)
	○ Pageantを使う
LINUXにロクインするときには 🔨	
	$\wedge$
「の但ム	
/ 切り切り	
ユーサ名→●●●	のとも押止げったノいウマ
	しれを押せはロジョン元」
バスワート $\rightarrow$ * * *	



### Linuxの操作①





### Linuxの仕組み



<sup>※</sup> Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.

簡単に...



### Linuxの操作② よく使うやつとか

ディレクトリ内にあるファイルの確認:ls
 カレントディレクトリの変更:cd

- ディレクトリの作成:mkdir
- ●ファイルのコピー:cp
- つアイルの移動:mv
- ●ファイルの削除:rm



### 長岡技術科学大学 Linuxの操作③ ISコマンド(ディレクトリ内にあるファイルやディレクトリの確認)



<sup>※</sup> Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.

### 長岡技術科学大学 Linuxの操作④ cdコマンド(カレントディレクトリの変更)





ホストの設定

suiko

/vukiko

ホストの設定名(T)

ローカルの初期フォルダ(L)

ホストの初期フォルダ(R)

<u>estuary5.suiko</u> ユーザ名(U)

基本

### ファイル転送ソフト FFFTP

🕫 estuary5.suiko (\*) - FFFTP

ドラッグ&ドロップで, Windows,Linux上のファイルやフォルダを やり取りできちゃうソフト

これでLinux上のディレクトリやファイルの コピーや削除も出来ます.

ホスト名(アドレス)(N)

キャンセル

バスワード/バスフレーズ(P)

estuary5.nagaokaut.ac.jp

anonymous

現在のフォルダ

ヘルプ

拡張 文字コード ダイアルアップ 高度

\*\*\*\*\*\*

最後にアクセスしたフォルダを次回の初期フォルダとする(F)

OK





# SWANとGrADSの ダウンロードとインストール

### Windows上でダウンロードして Linux上でインストールします

SWAN→www.swan.tudelft.nl GrADS→http://www.iges.org/grads/

/home/suiko/yukiko/pptのディレクトリの中に インストールします



### Windows上でダウンロードしたファイルを FFFTPを使って転送

🏚 estuary5.suiko (*	•) - FFFTP						
接続(E) コマンド( <u>C</u> ) ブ	ブックマーク( <u>B</u> ) 表示(	<u>い</u> ツール(T) オ	オプション(①) ヘルプ(日)				
<b>₩ % X + +</b>	👌 X 🛛 🗳	A B 📭	Euc Jis 無 か	J III 🏢 🛱	8		Long /
🔁 🗃 🖸 C:¥Program	n Files¥ffftp	*	🔁 🗃 🛛 /yukiko			<u> </u>	タレップに保たした。
名前	日付 2009/09/17 1 1: 2007/10/13 1 2: 2008/09/23 1 3: 2008/09/23 1 2008/09/23 1	サイズ 種… 36,386 exe 61,880 chm 89,120 exe 4,633 txt 45,880 txt (1)	名前 igrads-2.0.a7.1 make-wind make-wind2 swan4041 AB swan4072-1	日付 2009/09/24 … 2010/03/16 … 2010/03/16 … 2010/03/16 … 2009/12/30 …	サイズ 種 <dir> 1 <dir> <dir> <dir> <dir> <dir> <dir></dir></dir></dir></dir></dir></dir></dir>	rwx rwx rwxr-xr rwxr-xr rwxr-x rwxr-x	クトックに保存したイルをドラッグ
L JRE32.DLL 2	2002/03/22 1	27,648 DLL	grads=2.0.a7.1=b1 suiko.f swan4041AB.tgz swy.n4072.tg2	2009/09/24 2010/05/25 2009/12/30 2009/11/20	11,400,9 gz 2,247 f 546,634 tgz 1,400,414 tgz	rw-rr rw-rr rw-rr	swan4041AB
ファイル一覧の取	得は正常終了し	ノました。(75 ドロッ	プするとこ	しな風にL	<mark>inux上に</mark>	転送でる	きます
ホスト ましpaca	Bytes)	口一力ル空 66916.4	40M Bytes   車云送	待ちファイル0個		G	rADS±回样です

ちなみに, このSWANのソフトのファイル(swan4041AB.tgz)は, 拡張子.tgzの圧縮ファイルです.

tarコマンドでまとめられた拡張子.tarファイル(無圧縮)をGZIP形式(.gz)によって 圧縮したファイルらしいです.





# 風データの準備 GRIBファイルからの抜き出しと バイナリデータ化 Fortranプログラムでテキストデータ化



### 使用する風データ

風データをSWANで使用するために, GrADSを使用します. 今回使用する風データは,

気象業務支援センターが提供している

メソ客観解析データ(MSM:Mesoscale Spectral Model)

テキストエディタや エクセルでは 開けません!!!!!





2008年4月1日00時のデータが1つのGRIBファイルに 入っています.

▶ その中に, 上記のようなデータたちが計96個入っています.

その中から必要な風データを抜き出す!!!!!!



### そもそもGrADSとは

The Grid Analysis and Display System の略 格子状に配列された4次元データ(x,y,z,t)を2次元描画するツール



今回は、1ヶ月分の風データをu、v方向それぞれにバイナリデータ化します



### GrADSの起動と終了

/home/suiko/yukiko/ppt/grads-2.0a7.1/binのディレクトリです

[suiko@estuary5 bin]\$ ./grads -b

Grid Analysis and Display System (GrADS) Version 2.0.a7.1 Copyright (c) 1988-2008 by Brian Doty and the Institute for Global Environment and Society (IGES) GrADS comes with ABSOLUTELY NO WARRANTY See file COPYRIGHT for more information

Config; v2.0.a7,1 little-endian readline printim grib2 netcdf hdf4-sds hdf5 geotiff Issue 'q config' command for more information. Landscape mode? ('n' for portrait): GX Package Initialization: Size = 11 8.5 Running in Batch mode ga->

GrADS-2.0a7.1ディレクトリの中のbinディレクトリに入る ※bin中にgrbファイルも入れる









バイナリデータをテキストデータに変換

Fortranプログラムの 作成 テキストデータ化



[suiko@estuary5 bin]\$ chmod a+x grib2ctl.pl [suiko@estuary5 bin]\$ ./grb2ctl.pl MANAL\_2008040100.grb > 0804.ctl -bash: ./grb2ctl.pl: そのようなファイルやディレクトリはありません [suiko@estuary5 bin]\$ ./grib2ctl.pl MANAL\_2008040100.grb > 0804.ctl

grib2ctl.plというperlスクリプトをネットから探してbin中に転送 bin中にあるgrib2ctl.plというperlスクリプトを使って, ctlファイルを作成

perlスクリプトとして実行可能にする

[suiko@estuary5 bin]\$ chmod a+x grib2ctl.pl [suiko@estuary5 bin]\$ ./grib2ctl.pl \_ts3hr MANAL\_2008040100.grb > 0804.ctl

./ grib2ctl.pl実行で, MANAL\_2008040100.grbのctlファイル0804.ctlが作成される







### ctlファイルの書き換え

#### Windows上に0804.ctlを転送して、テキストエディタで編集します.



今回は2008.04分をまとめるので,8個/1day×30日=240個



# idxファイルの作成

#### 0804.ctlをLinux上に戻したら

[suiko@estuary5 bin]\$ ./gribmap -i 0804.ctl griblmap: opening GKIB file: MANAL 2008040100.grb	[suiko@estuary5 bin]\$ ./gribmap -i 0804.ctl
griblmap: opening GRIB file: MANAL_2008040103.grb griblmap: opening GRIB file: MANAL_2008040106.grb griblmap: opening GRIB file: MANAL_2008040109.grb griblmap: opening GRIB file: MANAL_2008040112.grb griblmap: opening GRIB file: MANAL_2008040115.grb	bin中に元から入っているgribmapを実行 MANAL_2008040100.grb.idxファイルが生成
griblmap: opening GRIB file: MANAL_2008040118.grb griblmap: opening GRIB file: MANAL_2008040121.grb griblmap: opening GRIB file: MANAL_2008040200.grb griblmap: opening GRIB file: MANAL_2008040203.grb griblmap: opening GRIB file: MANAL_2008040206.grb griblmap: opening GRIB file: MANAL_2008040209.grb griblmap: opening GRIB file: MANAL_2008040212.grb griblmap: opening GRIB file: MANAL_2008040215.grb griblmap: opening GRIB file: MANAL_2008040215.grb griblmap: opening GRIB file: MANAL_2008040218.grb griblmap: opening GRIB file: MANAL_2008040218.grb griblmap: opening GRIB file: MANAL_2008040221.grb griblmap: opening GRIB file: MANAL_2008040221.grb griblmap: opening GRIB file: MANAL_2008040221.grb	こんな風に 4月分全てを対応付けられたら成功
griblmap: opening GRIB file: MANAL_2008043012.grb griblmap: opening GRIB file: MANAL_2008043015.grb griblmap: opening GRIB file: MANAL_2008043018.grb griblmap: opening GRIB file: MANAL_2008043021.grb griblmap: reached end of files griblmap: writing the map [suiko@estuary5 bin]\$	これでGrADSIこ 読み込む準備が完了 ののののののではないではないです。 しているのでは、 しているので、 こので、 ひたいるので、 しているので、 しているので、 しているので、 しているので、 このので、 このので、 このので、 このので、 このので、 しているので、 このので、 つので、 つので、 つので、 ついので、 つので、 ついので、 ついので、 ついので、 ついので、 ついので、 ついので、 ついので、 ついので、 ついのので、 ついので、 ついので、 ついのので、 ついのので、 ついので、 ついので、 ついのので、 ついのので、 ついので、 ついので、 ついのので、 ついので、 ついので、 ついの ついので、 ついので、 ついので、 ついので ついついので う ついので ついの ついのので ついののでの ついのの ついの ついのの ついの



### GrADSスクリプトの作成①

12ヶ月分を同じようにしたら、バイナリデータ化するためのGrADSのスクリプトをつくる (0804~0903まで)





X Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.



### GrADSスクリプトの作成③

i=1 while (i<=12)

if (i=1 | i=2 | i=3 | i=4 | i=5 | i=6) a=3+i 'open 080'a'.ctl' endif

if (i=7 | i=8 | i=9) a=3+i 'open 08'a'.ctl' endif

if (i=10 | i=11 | i=12) a=i-9 'open 090'a'.ctl' endif

'set lon 136.9 139.8' 'set lat 36.9 38.9' 'set lev 0' if (i=1 | i=3 | i=6 | i=8) time=0 endif if (i=2 | i=4 | i=5 | i=7 | i=9 | i=10 | i=12) time=8 endif if (i=11) time=-16 endif

if (i=1 | i=2 | i=3 | i=4 | i=5 | i=6) yymm=080a endif

if (i=7 | i=8 | i=9) yymm=08a endif

if (i=10 | i=11 | i=12) yymm=090a endif 'set gxout fwrite' 'set fwrite 'yymm'u.bin' t=1 while(t<=240+time) 'set t 't'' 'd ugrdsfc' t=t+1 endwhile 'disable fwrite'

'set gxout fwrite' 'set fwrite 'yymm'v.bin' t=1 while(t<=240+time) 'set t 't" 'd vgrdsfc' t=t+1 endwhile 'disable fwrite'

'close 1' i=i+1 endwhile

これを.gs形式で保存

今回はbinary.gsとします
 ※ Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserved.



### GrADSスクリプトの実行





### Fortranでテキストデータ化①





### Fortranでテキストデータ化② DATファイルにするプログラム

#### program main

parameter(itm=248,ilon=32,ilat=24) character dataname1\*16,dataname2\*16,filebase1\*5,filebase2\*5 character form2\*20 integer n dimension u (ilon,ilat),v (ilon,ilat) dimension u2(ilon,ilat),v2(ilon,ilat)

common u.v.n.it

#### \*\*\*\*\* open file

7月データ (2004/07/25 0:00 - 2004/08/31 18:00) open(10,file='0812u.bin',form='unformatted',access='direct' 1 ,recl=ilon\*ilat) open(50,file='0812v.bin',form='unformatted',access='direct' 1 ,recl=ilon\*ilat)

#### \*\*\*\*\* データの読み込み n=1

do it=1 , itm

read(10,rec=it ) u read(50,rec=it ) v

#### \*\*\*\*\*

#### write(\*,\*)'time',it

if( n.le. 9) then form2='(a5,a1,i1 )' else if(n.ge. 10.and.n.le. 99) then form2='(a5,a1,i2 )' else if(n.ge. 100.and.n.le. 999) then form2='(a5,a1,i3 )' else if(n.ge.1000.and.n.le.9999) then form2='(a5,a1,i4 )' endif

#### filebase1='win-u'

filebase2='win-v' write(dataname1,form2) filebase1,'\_',n write(dataname2,form2) filebase2,'\_',n open( 90,file=dataname1,status='unknown') open(100,file=dataname2,status='unknown')

do j=ilat,1,-1 write( 90,'(361f10.5)') (-u(i,j),i=1,ilon) write(100,'(361f10.5)') (-v(i,j),i=1,ilon) enddo

close( 90) close(100) n=n+1

enddo

#### 10 continue

- close(10)
- \* close(20)
- \* close(30)
  \* close(40)
  close(50)
- \* close(60)
- \* close(70)
- \* close(80) stop end



### Fortranでテキストデータ化③ TXTファイルにするプログラム

wind1.txt(08.4)から wind12.txt(09.3)までの TXTファイルが完成

program main	***** データの書き込
parameter(nt=248 , nx=32, ny=24)	do j=ny,1 , -1
character dataname1*16,dataname2*16,filebase1*5,filebase2*5	write( 95,'(361
character form2*20	enddo
integer T	WRITE(96,'(2F
dimension u(0:nx,0:ny),v(0:nx,0:ny)	ENDDO
common u,v,n,it	

\*\*\*\*\* swanで使用するグラフ open(95,file='wind9.txt',status='unknown') open(96,file='windpx.dat',status='unknown') open(97,file='windpy.dat',status='unknown')

#### \*\*\*\*\* x成分データ

DO T=1,nt write(\*,\*)'time',T

```
if( T.le. 9) then
form2='(a5,a1,i1 )'
else if(T.ge. 10.and.T.le. 99) then
form2='(a5,a1,i2 )'
else if(T.ge. 100.and.T.le. 999) then
form2='(a5,a1,i3 )'
else if(T.ge.1000.and.T.le.9999) then
form2='(a5,a1,i4 )'
endif
```

```
filebase1='win-u'
write(dataname1,form2) filebase1,'_',T
open( 90,file=dataname1,status='unknown')
do j=ny,1,-1
read( 90,'(361f10.5)') (u(i,j),i=1,nx)
enddo
close( 90)
```

\*\*\* データの書き込み do j=ny,1 , -1 write( 95,'(361f10.5)') (u(i,j),i=1,nx) enddo WRITE(96,'(2F10.5)') REAL(T)/4.0 , U(16,15) ENDDO

#### \*\*\*\*\* Y成分データ

DO T=1,NT if( T.le. 9) then form2='(a5,a1,i1 )' else if(T.ge. 10.and.T.le. 99) then form2='(a5,a1,i2 )' else if(T.ge. 100.and.T.le. 999) then form2='(a5,a1,i3 )' else if(T.ge.1000.and.T.le.9999) then form2='(a5,a1,i4 )' endif

filebase2='win-v' write(dataname2,form2) filebase2,'\_',T open(100,file=dataname2,status='unknown') do j=ny,1,-1 read(100,'(361f10.5)') (v(i,j),i=1,nx) enddo close(100)

#### \*\*\*\*\* データの書き込み do j=ny,1,-1 write( 95,'(361f10.5)') (v(i,j),i=1,nx) enddo WRITE(97,'(2F10.5)') REAL(T)/4.0,V(16,15) ENDDO

#### close( 95) ; close(96) ; close(97) stop end

wind1.txt(08.4)から wind12.txt(09.3)までの TXTファイルが完成





# SWANをインストールして 計算をします





# 自動高速インストールをします

[s	uiko@estuary5	swan4041AB]\$	dir				イン	ィストール前	
IN	STALL.README	macros.inc	swan.edt	swancom5.ftn	swanprel.ftn	switch.pl	入って	いてるファー	イル達 🦯
Ma.	kefile	ocpcre.ftn	swancoml.ftn	swanmain.ftn	swanpre2.ftn	swmodl.ftn			
ÞΨ.	ANKUN.KEADME	ocpids.ftn	swancomZ.ftn	swanoutl.ftn	swanrun	swmodZ.ftn			
ge:	tempi shirafila	ocpmix.itn	swancomj.itn	swanout2.1th	swanrun.pat	swmod3.Ith			
ma Fe	uikoMestuarv5	gwan40414B1\$	swancom4.ith	swanparti.ttn	swanser.ttn				
ls.	uiko@estuary5 uiko@estuary5	swan4041AB1\$	make ser						
maj	ke FOR=ifort 1	FFLAGS="-02 -	tem/ Tx- Yegt		1 . 1				
		INCS="" LIBS	=‴″ swan.exe	[suiko@e	stuarv5 swa	an4041AE	31 <b>\$</b> mak	ce confia	
mal	ke[1]: ディレ	クトリ `/home	/suiko/yukiko/						
if	ort swmodl.f	-c -O2 -tpp7_	-xN -mp1 -W0 -		stuarys swa	an4041AE	зј⊅ так	ke ser	
it	ort command	line remark #	10148: オブショ	ארצא עז ∽ <sup>1</sup> י		(CNO			I
11 ;f	ort swmod2.1	-c -UZ -tpp/ lina namank #	-XN -mpl -WU - 101/0፡ ታትማድረጉ	auto -assume by + `/ '_+n' lt+t	tereci ピュトオわプいヨ	= ++ 2			
$\frac{11}{1}$	ort swmod3 f	-c -O2 -tpp7	-vN - mn1 - W0 -	auto -accume hu	tereol	Ken.			
i f	ort: command	line remark #	10148: オプシュ	auco assume by ォン '-tp' はサz	ポートされていす	きせん.			
SWI	nod3.f(9389):	(col. 7) rem	ark: PARTIAL L	OOP がベクトル(	とされました。.	/ 00			
SWI	nod3.f(9389):	(col. 7) rem	ark: PARTIAL L	00P がべクトル(	日されました。.				
SWI	nod3.f(3448):	(col. 9) rem	ark: ループがっ	ベクトル化されま	した。.				
SWI	nod3.f(2526):	(col. 9) rem	ark: ループがっ	>クトル化されま	した。	•			
SW	<u>nod3.t(%544):</u>	<u>lcol. 9) rem</u>	ark: ルーフから	<u>Ko kuisana</u>					$\frown$
,								( )	
	SWAND	Implement	ation Manua	に古法が主い	いてあるので				
	その通り	「ンストール	しよう.						
	1. マシン	依存のマク	コを含むイン	クルードファイ	ルをまず最初	刃につくりま	<b>d</b> .	λ —	- (
		I I tmoore	ここし、 」 いの 上町 ノギノ	n maka con	fioで作これま			/	
	このノアイ	1/12/11acro	SINCCITICA	L, Make Con	ing CIEDILA	<b>59</b> .			$\land$
	2. SWAN	を構築する	ためにmake	serと打ちます	<i>ţ. (シ</i> リアル⁻	モード)		U	U )
					,	,	$\square$		
								$\nabla$	



### インストール終了

#### [suiko@estuary5 ppt]\$ cd swan4041AB [suiko@estuary5\_swan4041AB]\$\_dir INSTALL README ocpcomm3.mod swancoml.o swanoutl.o swcomm1.mod Makefile swanout2.f ocpcomm4.mod swancom2.f swcomm2.mod SWANRUN . README ocpcre.f swancom2.ftn swanout2.ftn swcomm3.mod swanout2.o getcmpl swancom2.0 swcomm4.mod ocpcre.ftn m bndspec.mod swancom3.f swanparll.f switch.pl ocpcre.o swmod1.f m\_constants.mod ocpids.f swancom3.ftn swanparll.ftn m diffr.mod swmod1.ftn ocpids.ftn swancom3.0 swanparll.o m<sup>-</sup>fileio.mod ocpids.o swmod1.o swancom4.f swanprel.f swmod2.f swancom4.ftn swanprel.ftn m genarr.mod ocpmix.f m obsta.mod ocpmix.ftn swancom4.0 swmod2.ftn swanprel.o swmod2.o m parall.mod ocpmix.o swancom5.f swanpre2.f m\_snl4.mod outp data.mod swancom5.ftn swanpre2.ftn swmod3.f platform.pl swmod3.ftn swancom5.0 swanpre2.0 m wcap.mod swmod3.o m xnldata.mod serv\_xnl4v5.mod swanmain.f swanrun machinefile swan.edt swanmain.ftn swanrun.bat timecomm.mod swanmain.o macros.inc swan.exe swanser.f ocpcomm1.mod swancoml.f swanout1.f swanser.ftn

swanoutl.ftn swanser.o

ocpcomm2.mod swancom1.ftn [suiko@estuary5 swan4041AB]\$

0

0

0

インストール後はこんな風になってます.



### SWANの計算実行に必要なもの

無事インストールできたら、計算を実行するために必要なデータやファイルを準備します

必要なもの

①SWANそのもの インストールしました ②SWANのコマンドファイル(拡張子swn) これから作成します. 10.46108 0.72671 前回説明したGrADSやらで作ったデータです. (3)風デ 10.97641 15193 名前は今回wind1.txtです. 9.91554 9.91554 9.98820 8.55467 5.58536 2403 240 2730 6822 2958 1446 240 240 240 1.26934 3,80494 今回はniigata\_dep.datというデータです. ④ 地形デー 格子状の地形データになっています. 240 240 240 240 240 240 240 2401 2400 2401 2401 2401 2401 2402 2408 2402 この3つをさっきSWANをインストールしたSWAN4041ABのディレクトリの中に 転送します.



### SWANのコマンドファイルの作成①

#### インストールしたSWANの中にswan.edtというコマンドファイルの雛形があります.

m_wcap.mod m_xnldata.mod	serv_xnl4v5.mod	swannain.f	swanrun	swmod3.0
machinețile	swan.edt	ewanmain.ftn	swanrun.bat	timecomm.mod
macros.inc	swan.exe	swanmain.o	swanser.t	
locpcomml mod	swancomi t	SWADOUT   1	gwanger ffn	

#### swan.edtの中身はこんな感じ





8 \$ PURPOSE : the formulation of wave forecasting and ↔ 9 \$ hindcasting in nihonkai by SWAN↔ 10 \$↔





[時間,場所的に一定の水面上昇] [x軸から見た北の角度] [最小水深] [最大エラーメッセージ数] [コマンドファイルのエラー設定] [重力加速度] [水の密度] [出力単位の選択] [よくわからない] [波向,風向の決定] [よくわからない] [出力画面選択] [最大フルード数] [テスト出力画面選択]







### SWANのコマンドファイルの作成④ Physics,Numericsコマンド





Output,Lock-upコマンド

<sup>47</sup> <sup>#************ 0UTPUT REQUESTS ***********************************</sup>
100       POINTS 'SWAN22'       191187. 65687.4         11       POINTS 'SWAN22'       139045. 28700.4         12       TABLE 'SWAN11' HEADER 'Hpoint11.dat' DIR OUTPUT 20080401.000000 3. HR+         13       TABLE 'SWAN11' HEADER 'Point11.dat' DIR OUTPUT 20080401.000000 3. HR+         14       TABLE 'SWAN1' HEADER 'Point11.dat' DIR OUTPUT 20080401.000000 3. HR+         15       STABLE 'SWAN1' HEADER 'Point12.dat' DIR OUTPUT 20080401.000000 3. HR+         15       STABLE 'SWAN1' HEADER 'Point12.dat' DIR OUTPUT 20080401.000000 3. HR+         15       TABLE 'SWAN12' HEADER 'Point12.dat' DIR OUTPUT 20080401.000000 3. HR+         16       TABLE 'SWAN22' HEADER 'Point12.dat' DIR OUTPUT 20080401.000000 3. HR+         17       TABLE 'SWAN22' HEADER 'Point22.dat' DIR OUTPUT 20081201.000000 3. HR+         18       TABLE 'SWAN22' HEADER 'Point22.dat' DIR OUTPUT 20081201.000000 3. HR+         17       TABLE 'SWAN22' HEADER 'Point22.dat' DIR OUTPUT 20081201.000000 3. HR+         18       GROUP 'SWANOUT' NOHEADER 'SWANOUTHI.TXT' HSIGN OUTPUT 20080401.000000 3. HR+         18       BLOCK 'SWANOUT' NEADER 'SWANOUTHI.TXT' WIND OUTPUT 20080401.000000 3. HR+         18       BLOCK 'SWANOUT' HEADER 'SWANOUTHI.TXT' WEN OUTPUT 20080401.000000 3. HR+         18       BLOCK 'SWANOUT' HEADER 'SWANOUTHI.TXT' WEN OUTPUT 20080401.000000 3. HR+         19       BLOCK 'SWANOUT' HEADER 'SWANOUTHI.TXT' WEN OUTPUT 20040101.000000 3. HR+<
POINTS → ある1地点だけのデータを欲しいときに設定します.
TABLE → POINTSで定めた地点の欲しい出力データの設定をします.
GROUP → 計算領域の中でまとまった格子データが欲しいときに設定します.
BLOCK → GROUPで定めた領域の欲しい出力データを設定します.
<u>COMPUTE</u> → いつからいつまで計算するのか設定します.
STOP → 入力コマンド終わりの合図. Copyright @ 2010, Hydraulic Engineering Laboratory. All Rights Reserve



### 計算を実行する

#### その前に、SWANにこのコマンドファイルを使ってくださいと言う必要があります.



計算の実行手順を記したものがあって、それがswanrunというものです。 それに、nihonkai-03.swnというコマンドファイルで計算をお願いします、と言います、



### 2回目以降の計算のために

初めての計算が終わったら、SWANの初期ファイルswaninitが生成されます. 次からいちいち前の実行手順を踏まなくても、./swan.exeで計算が実行されるように書き換えます.





### 2回日以降の計算



