CAS-ESM2的编译和运行

汇报人：郝卉群

2023年　6月　7日

01 CAS-ESM2 代码获取

02 配置CAS-ESM2及环境

03 创建case、编译、运行

04 输出数据说明

01 CAS-ESM2 代码获取

02 配置CAS-ESM2及环境

03 创建case、编译、运行

04 输出数据说明

01、CAS-ESM2 代码获取

✓接入CAS-ESM版本库

- 在本地（能够访问外网或者能够访问159.226.234.62，例如地球模拟器10.64.202.2节点）
  输入ssh-keygen -t rsa生成密钥
  将生成的.pub文件发送给管理员配置权限（haohq@sccas.cn）

```
[haohq@server02 .ssh]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/public/home/haohq/.ssh/id_rsa): /public/home/haohq/.ssh/haohuiqun_id
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /public/home/haohq/.ssh/haohuiqun_id.
Your public key has been saved in /public/home/haohq/.ssh/haohuiqun_id.pub.
The key fingerprint is:
SHA256:MtnfxrZ4gqGcGuI+H2Qmb4ZfC5WU9LDkn8rqXhhP1DA haohq@server02
The key's randomart image is:
+---[RSA 2048]----+
|      E          |
|     + O         |
|      * o        |
|     o = .       |
|  . = B S        |
|   B B +.. +     |
|  o O.*o o. =    |
| ..= B+.. .o .   |
| .o=B..   .oo    |
+----[SHA256]-----+
[haohq@server02 .ssh]$ ls
authorized_keys  config  git-admin  haohuiqun_id  haohuiqun_id.pub  id_rsa  id_rsa.pub  known_hosts
[haohq@server02 .ssh]$
```

01、CAS-ESM2 代码获取

✓接入CAS-ESM版本库

- 在本地创建~/.ssh/config文件，编辑如下内容

  host git-server-huan
  user git-server
  hostname 159.226.234.62
  port 22
  identityfile ~/.ssh/***（***为生成的私钥文件名，默认为id_rsa）

```
host git-server-huan
user git-server
hostname 159.226.234.62
port 22
identityfile ~/.ssh/haohuiqun_id
```

- chmod 600 config
  chmod 755 ~/.ssh/
  chmod 600 ~/.ssh/id_rsa ~/.ssh/id_rsa.pub
  chmod 644 ~/.ssh/known_hosts

- 切换至工作目录

  git clone git-server-huan:cas-esm-huan.git
  开始下载代码

```
[haohq@server02 haohq]$ git clone git-server-huan:cas-esm-huan.git
Cloning into 'cas-esm-huan'...
remote: Enumerating objects: 10448, done.
remote: Counting objects: 100% (10448/10448), done.
remote: Compressing objects: 100% (7074/7074), done.
remote: Total 10448 (delta 3224), reused 10448 (delta 3224)
Receiving objects: 100% (10448/10448), 240.01 MiB | 23.15 MiB/s, done.
Resolving deltas: 100% (3224/3224), done.
Checking out files: 100% (16286/16286), done.
[haohq@server02 haohq]$ ls
cas-esm-huan  DGVM  hhq  inputdata
```

01 CAS-ESM2 代码获取

02 配置CAS-ESM2及环境

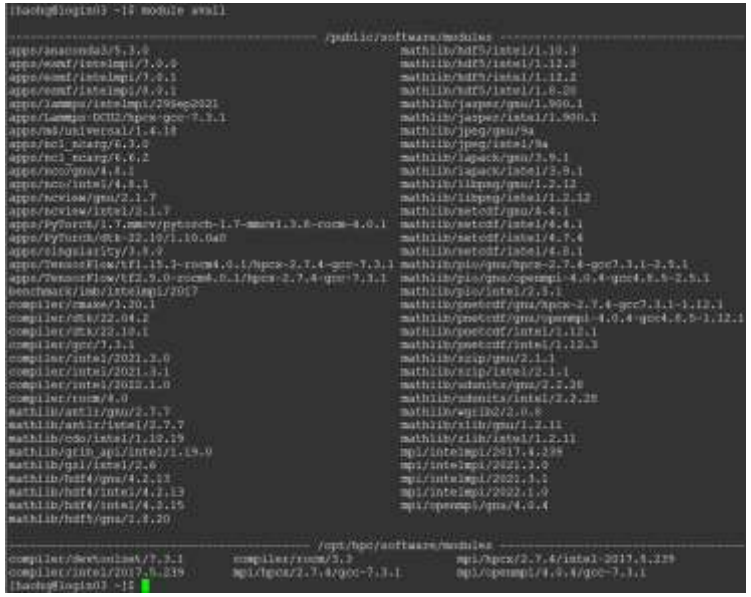
03 创建case、编译、运行

04 输出数据说明

02、配置CAS-ESM2及环境

✓账号环境配置

- 查看本机可用环境
  module avail



- Intel编译器：2017.5.239版本
- mpi编译器：intel-2017.5.239版本
- ROCM编译器：3.3版本
- Netcdf库：intel/4.4.1版本
- hdf5库：intel/1.8.20版本
- szip库：intel/2.1.1版本

- ncl：6.3.0版本
- nco：intel/4.8.1版本

```
module purge
module load compiler/intel/2017.5.239
module load mpi/intelmpi/2017.4.239
#module load mpi/hpcx/2.7.4/intel-2017.5.239
module load compiler/rocm/3.3
module load mathlib/hdf5/intel/1.8.20
module load mathlib/szip/intel/2.1.1
module load mathlib/netcdf/intel/4.4.1
module load apps/ncl_ncarg/6.3.0
module load apps/nco/intel/4.8.1
```

- 常用环境可写入~/.bashrc中
  source ~/.bashrc 生效
- 查看已加载环境
  module list

```
[haohq@login03 ~]$ module list
Currently Loaded Modulefiles:
  1) compiler/intel/2017.5.239     4) mathlib/hdf5/intel/1.8.20     7) apps/ncl_ncarg/6.3.0
  2) mpi/intelmpi/2017.4.239       5) mathlib/szip/intel/2.1.1      8) apps/nco/intel/4.8.1
  3) compiler/rocm/3.3             6) mathlib/netcdf/intel/4.4.1
[haohq@login03 ~]$
```

02、配置CAS-ESM2及环境

✓配置CAS-ESM

✓ 切换至路径Machines/

cd cas-esm-huan/scripts/casesm_utils/Machines

✓ 编辑文件config_machines.xml

```
<machine MACH="huan_default"
         DESC="Huan_Miyun, 64 pes/node, PBS batch system"
         EXEROOT="/data/user_default/cas-esm/run/$CASE"
         OBJROOT="$EXEROOT"
         LIBROOT="$EXEROOT/lib"
         INCROOT="$EXEROOT/lib/include"
         DIN_LOC_ROOT_CSMDATA="/data/zhangh/data/inputdata"
         DOUT_S_ROOT="/data/user_default/cas-esm/run/archive/$CASE"
         DOUT_L_HTAR="FALSE"
         DOUT_L_MSROOT="csm/$CASE"
         OS="Linux"
         GMAKE_J="16"
         MAX_TASKS_PER_NODE="64"
         MPISERIAL_SUPPORT="TRUE"
         PES_PER_NODE="64" />
```

- 找到MACH=“huan_default”字段。
- 编辑EXEROOT（编译后的可执行程序的目录）和DOUT_S_ROOT（输出数据临时存放目录），修改为当前对应路径。

```
[haohq@login03 Machines]$ pwd
/data/haohq/cas-esm-huan/scripts/casesm_utils/Machines
```

如图中当前路径为/data/haohq/cas-esm-huan/scripts/casesm_utils/Machines

- 则将上图中蓝色框内修改为/data/haohq/cas-esm-huan

```
<machine MACH="huan_default"
         DESC="Huan_Miyun, 64 pes/node, PBS batch system"
         EXEROOT="/data/haohq/cas-esm-huan/run/$CASE"
         OBJROOT="$EXEROOT"
         LIBROOT="$EXEROOT/lib"
         INCROOT="$EXEROOT/lib/include"
         DIN_LOC_ROOT_CSMDATA="/data/zhangh/data/inputdata"
         DOUT_S_ROOT="/data/haohq/cas-esm-huan/run/archive/$CASE"
         DOUT_L_HTAR="FALSE"
         DOUT_L_MSROOT="csm/$CASE"
         OS="Linux"
         GMAKE_J="16"
         MAX_TASKS_PER_NODE="64"
         MPISERIAL_SUPPORT="TRUE"
         PES_PER_NODE="64" />
```

01 CAS-ESM2 代码获取

02 配置CAS-ESM2及环境

03 创建case、编译、运行

04 输出数据说明

03、创建case、编译、运行

✓ 创建case命令格式为

./create_newcase -case [case名] -compset [耦合模式组合] -res [分辨率] -mach [机器名]

其中[case名]为用户自定义，[耦合模式组合]和[分辨率]可选项目参见表1，[机器名]为第2步骤中设置的huan_default。

表 1 CAS-ESM当前支持的Compset

名称 (-compset)	分辨率	-res
PI_C6	1.4°×1.4°	fd14_licom
PI_C6_C	1.4°×1.4°	fd14_licom
HIST_C6_C	1.4°×1.4°	fd14_licom
HIST_C6_B	1.4°×1.4°	fd14_licom
AMIP_C6	1.4°×1.4°	fd14_fd14
	0.5°×0.5°	fd05_fd05
	1.0°×1.0°	fd1_fd1
	0.25°×0.25°	fd02_fd02
4XCO2	1.4°×1.4°	fd14_licom
1PCTCO2	1.4°×1.4°	fd14_licom
SSP126	1.4°×1.4°	fd14_licom
SSP245	1.4°×1.4°	fd14_licom
SSP370	1.4°×1.4°	fd14_licom
SSP585	1.4°×1.4°	fd14_licom
ESM_PI_C6_C	1.4°×1.4°	fd14_licom
ESM_HIST_C6_B	1.4°×1.4°	fd14_licom

注：PI表示piControl试验，C6是CMIP6的简写，HIST是历史试验，4XCO2为4倍CO2突增试验，1PCTCO2为CO2每年1%递增试验，SSP为未来情景预估试验，ESM_PI为全耦合piControl试验，ESM_HIST为全耦合历史试验。

03、创建case、编译、运行

- ✓ 切换至路径scripts/

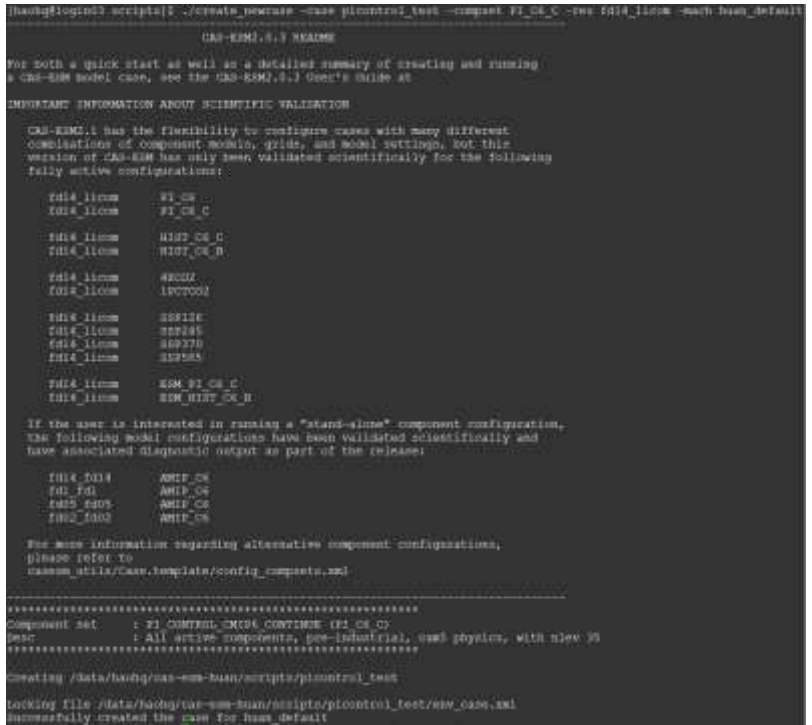
  cd cas-esm-huan/scripts

- ✓ 创建Case（以picontrol试验为例）

  在命令行输入

  ./create_newcase -case picontrol_test_02 -compset PI_C6_C -res fd14_licom -mach huan_default

  生成名为picontrol_test的目录，作为新case的目录。



```
[haohq@login03 scripts]$ ls
casesm_utils     create_newcase_storm  link_dirtree     sample_compset_file.xml
create_clone     create_test           picontrol_test   sample_pes_file.xml
create_newcase   create_test_suite     README
[haohq@login03 scripts]$
```

03、创建case、编译、运行

✓编译

- ✓ 切换至路径picontrol_test

  cd picontrol_test

- ✓ 执行预编译

  ./configure –case

```
[haohq@login03 picontrol_test]$ ./configure -case
Generating resolved namelist, prestage, and build scripts
configure done.
Successfully generated resolved namelist, prestage, and build scripts
Locking file env_conf.xml
branch licom
Generating clean_build script
Generating submit script
Generating build script
Generating run script
Locking file env_mach_pes.xml
Successfully configured the case for huan_default
If an old build exists for this case, you might want to
   run the *.clean_build script before building
[haohq@login03 picontrol_test]$
```

```
[haohq@login03 picontrol_test]$ ls
Buildconf                env_conf.xml                picontrol_test.huan_default.clean_build
CaseStatus               env_derived                 picontrol_test.huan_default.run
check_case               env_mach_pes.xml            picontrol_test.huan_default.submit
check_input_data         env_mach_specific           README
configure                env_run.xml                 README.case
create_production_test   LockedFiles                 SourceMods
env_build.xml            Macros.huan_default         Tools
env_case.xml             picontrol_test.huan_default.build  xmlchange
```

Earth Lab

03、创建case、编译、运行

✓编译

✓ 执行编译

./picontrol_test.huan_default.build

```
[haohq@login03 picontrol_test]$ ./picontrol_test.huan_default.build
------------------------------------------------------------------------
 CAS-ESM BUILDNML SCRIPT STARTING
 - To prestage restarts, untar a restart.tar file into /data/haohq/cas-esm-huan/run/picontrol_test/run
 CAS-ESM BUILDNML SCRIPT HAS FINISHED SUCCESSFULLY
------------------------------------------------------------------------
 CAS-ESM PRESTAGE SCRIPT STARTING
 - CAS-ESM input data directory, DIN_LOC_ROOT_CSMDATA, is /data/zhangh/data/inputdata
 - Case input data directory, DIN_LOC_ROOT, is /data/zhangh/data/inputdata
 - Checking the existence of input datasets in DIN_LOC_ROOT
 CAS-ESM PRESTAGE SCRIPT HAS FINISHED SUCCESSFULLY
------------------------------------------------------------------------
 CAS-ESM BUILDEXE SCRIPT STARTING
 - Build Libraries: mct pio csm_share
Tue Jun 6 23:07:35 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/mct/mct.bldlog.230606-230718
Tue Jun 6 23:08:25 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/pio/pio.bldlog.230606-230718
Tue Jun 6 23:09:21 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/csm_share/csm_share.bldlog.230606-230718
Tue Jun 6 23:09:31 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/cpl.bldlog.230606-230718
Tue Jun 6 23:09:31 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/atm.bldlog.230606-230718
Tue Jun 6 23:10:33 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/wrf.bldlog.230606-230718
Tue Jun 6 23:10:33 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/gea.bldlog.230606-230718
Tue Jun 6 23:10:33 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/lnd.bldlog.230606-230718
Tue Jun 6 23:10:51 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/ice.bldlog.230606-230718
Tue Jun 6 23:11:10 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/ocn.bldlog.230606-230718
Tue Jun 6 23:11:20 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/glc.bldlog.230606-230718
Tue Jun 6 23:11:20 CST 2023 /data/haohq/cas-esm-huan/run/picontrol_test/run/casesm.bldlog.230606-230718
 - Locking file env_build.xml
 - Locking file Macros.huan_default
 CAS-ESM BUILDEXE SCRIPT HAS FINISHED SUCCESSFULLY
[haohq@login03 picontrol_test]$
```

03、创建case、编译、运行

✓运行

- ✓ 在完成编译步骤之后，会生成对应上例中picontrol_test用例的运行路径

  cas-esm-huan/run/picontrol_test/run/

  切换至运行路径

  cd cas-esm-huan/run/picontrol_test/run

```
[haohq@login03 picontrol_test]$ cd /data/haohq/cas-esm-huan/run/picontrol_test/run
[haohq@login03 run]$ ls
ahv_back.txt                                dncoef.h1                       MODEL.FRC
atm.bldlog.230606-230718.gz                 domain_licom.nc                 nyf.giss.T62.stream.txt.change
atm_in                                      drv_flds_in                     nyf.gxgxs.T62.stream.txt.change
atm_in.change                               drv_in                          nyf.ncep.T62.stream.txt.change
atm_modelio.nml                             drv_in.change                   ocn.bldlog.230606-230718.gz
B1850C5X_C35.cam2.h0.0001-12.nc             Eq1x1_130824pm2.grid            ocn_modelio.nml
B1850C5X_C35.cice.r.0002-01-01-00000.nc     Eq1x1_130824pm2.kmt             ocn.parm
B1850C5X_C35-colm-restart-0002-01-01-00000  fort.22.0002-01-01              rdirc.05
B1850C5X_C35.cpl.r.0002-01-01-00000.nc      gea.bldlog.230606-230718.gz     rpointer.atm
B1850C5X_C35.iap.r.0002-01-01-00000.nc      gea_modelio.nml                 rpointer.drv
B1850C5X_C35.iap.rs.0002-01-01-00000.nc     glc.bldlog.230606-230718.gz     rpointer.ice
BASIN.nc                                    glc_modelio.nml                 rpointer.lnd
casesm.bldlog.230606-230718.gz              ice.bldlog.230606-230718.gz     rpointer.ocn
casesm.exe                                  ice_in                          runoff.1x1.stream.txt.change
CoLM-ini-IAP-CMIP-128x256                   ice_in.change                   seq_maps.rc
CoLM-lai-IAP-CMIP-128x256                   ice_modelio.nml                 seq_maps.rc.change
CoLM-sbc-IAP-CMIP-128x256                   ig2004.nc                       ssmi_ifrac.clim.x0.5.txt.change
CoLM-srf-IAP-CMIP-128x256                   INDEX.DATA                      timing
CoLM-surf-dust-128x256                      licom_in                        TSinitial
cpl.bldlog.230606-230718.gz                 lnd.bldlog.230606-230718.gz     wrf.bldlog.230606-230718.gz
cpl_modelio.nml                             lnd_in                          wrf_modelio.nml
cpscript.sh                                 lnd_in.change
dice_ice_in.change                          lnd_modelio.nml
[haohq@login03 run]$
```

03、创建case、编译、运行

- ✓ namelists文件：atm_in、 drv_in 、 ice_in 、 licom_in 、 lnd_in

  vi drv_in

- ✓ 详细的设置参考quickstart文件

```
/
&seq_timemgr_inparm
  calendar        = 'NO_LEAP'
  atm_cpl_dt      = 1800
  lnd_cpl_dt      = 1800
  ocn_cpl_dt      = 10800
  ice_cpl_dt      = 1800
  glc_cpl_dt      = 86400
  start_ymd       =  00010101
  start_tod       = 0
  stop_option     ='nmonths'
  stop_n          = 1
  stop_ymd        = -999
  restart_option ='nmonths'
  restart_n       = 1
  restart_ymd     = -999
  end_restart     = .false.
  history_option ='never'
  history_n       = -999
  history_ymd     = -999
  histavg_option ='never'
  histavg_n       = -999
  histavg_ymd     = -999
  tprof_option    ='never'
  tprof_n         = -999
  tprof_ymd       = -999
/
```

模式结束时间
及方式设置

重启动文件
输出频率设置

03、创建case、编译、运行

✓ 编辑作业脚本 run.slurm

- #SBATCH -J　作业名
- #SBATCH -p　队列名 常用normal
- #SBATCH -N　申请节点数
- #SBATCH -n　申请进程数
- #SBATCH --ntasks-per-node=　每个节点分配进程数
- #SBATCH --cpus-per-task=　每个进程分配核数
- #SBATCH --mem=　指定内存
- #SBATCH -o casesm.o%j　标准输出
- #SBATCH -e casesm.e%j　标准错误
- #SBATCH –exclusive　独占节点（可选）

✓ 提交作业

sbatch run.slurm

```
#!/bin/bash
#SBATCH -J prcontrol
#SBATCH -p cpu_parallel
#SBATCH -N 16
#SBATCH -n 512
#SBATCH --ntasks-per-node=32
#SBATCH --cpus-per-task=2
#SBATCH --mem=200G
#SBATCH -o casesm.o%j
#SBATCH -e casesm.e%j
#SBATCH --exclusive
#SBATCH -t 1-00:00

module purge
module load compiler/intel/2017.5.239
module load mpi/intelmpi/2017.4.239
#module load mpi/hpcx/2.7.4/intel-2017.5.239
module load mathlib/hdf5/intel/1.8.20
module load mathlib/szip/intel/2.1.1
module load mathlib/netcdf/intel/4.4.1

scontrol show hostname > nd
NP=$SLURM_NPROCS
mpirun -np $NP -machinefile nd ./casesm.exe
```

Earth Lab

01 CAS-ESM2 代码获取

02 配置CAS-ESM2及环境

03 创建case、编译、运行

04 输出数据说明

04、输出数据说明

✓ 输出数据文件

序号	分量	类型	名称
1	大气	历史文件	[case名].iap.h0.[年]-[月].nc
2	大气	重启动文件	[case名] .iap.r.[年]-[月]-[日].nc [case名] .iap.rs.[年]-[月]-[日].nc
3	海洋	历史文件	MMEAN[年]-[月].nc
4	海洋	重启动文件	fort.22.[年]-[月]-[日]
5	陆面	历史文件	[case名]-colm-[年]-[月].nc
6	陆面	重启动文件	[case名]-colm-restart-[年]-[月]-[日]-00000
7	海冰	历史文件	[case名].cice.h.[年]-[月].nc
8	海冰	重启动文件	[case名].cice.r.[年]-[月]-[日]-00000.nc
9	耦合器	重启动文件	[case名].cpl.r.[年]-[月]-[日]-00000.nc

```
B1850C5X_C35.cam2.h0.0001-12.nc
B1850C5X_C35.cice.h.0002-01.nc
B1850C5X_C35.cice.r.0002-01-01-00000.nc
B1850C5X_C35.cice.r.0002-02-01-00000.nc
B1850C5X_C35-colm-0002-01.nc
B1850C5X_C35-colm-restart-0002-01-01-00000
B1850C5X_C35-colm-restart-0002-02-01-00000
B1850C5X_C35-colm-restart-0002-02-01-00000-sbc
B1850C5X_C35.cpl.r.0002-01-01-00000.nc
B1850C5X_C35.cpl.r.0002-02-01-00000.nc
B1850C5X_C35.iap.h0.0002-01.nc
B1850C5X_C35.iap.r.0002-01-01-00000.nc
B1850C5X_C35.iap.r.0002-02-01-00000.nc
B1850C5X_C35.iap.rs.0002-01-01-00000.nc
B1850C5X_C35.iap.rs.0002-02-01-00000.nc
```

谢　谢