|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
| WAVEWATCH III-6.07.1 | | |  |
| 应用**移植指南** | | |
| **文档版本** | **01** | |
| **发布日期** | **2022-07-30** | |
| 华为网格系统---方案4-032.png | | | | |
|  | 华为技术有限公司 | | 附件1-16K |  |

|  |
| --- |
| 版权所有 © 华为技术有限公司2022。保留一切权利。  非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。  商标声明  附件3-版权声明页图和其他华为商标均为华为技术有限公司的商标。  本文档提及的其他所有商标或注册商标，由各自的所有人拥有。  注意  您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。  由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。 |

|  |  |
| --- | --- |
| 华为技术有限公司 | |
| 地址： | 深圳市龙岗区坂田华为总部办公楼邮编：518129 |
| 网址： | <https://www.huawei.com> |
| 客户服务邮箱： | [support@huawei.com](mailto:support@huawei.com) |
| 客户服务电话： | 4008302118 |

目录

[目录 ii](#_Toc108269197)

[1 介绍 1](#_Toc108269198)

[2 环境要求 2](#_Toc108269199)

[3 移植规划数据 4](#_Toc108269200)

[4 配置编译环境 5](#_Toc108269201)

[4.1 搭建鲲鹏基座软件环境 5](#_Toc108269202)

[4.2 安装Zlib 6](#_Toc108269203)

[4.3 安装HDF5 7](#_Toc108269204)

[4.4 安装NETCDF-C 8](#_Toc108269205)

[4.5 安装NETCDF-Fortran 9](#_Toc108269206)

[4.6 安装ParMETIS 10](#_Toc108269207)

[5 获取源码 13](#_Toc108269208)

[6 编译和安装 14](#_Toc108269209)

[7 运行和验证 18](#_Toc108269210)

[8 修订记录 26](#_Toc108269211)

# 介绍

WAVEWATCH III是一个社区波浪建模框架，包括风浪建模和动力学领域的最新科学进展。WAVEWATCH III解决了波数方向谱的随机相位谱动作密度平衡方程，该模型包括浅水（冲浪区）应用的选项，以及网格点的润湿和干燥。

关于WAVEWATCH III的更多信息请访问[WAVEWATCH III官网](https://polar.ncep.noaa.gov/waves/wavewatch/)。

语言：Fortran

一句话描述：第三代海浪模式。

开源协议：自定义开源协议

建议的版本

建议使用版本为“WAVEWATCH III-6.07.1”。

# 环境要求

硬件要求

硬件要求如表2-1所示。

硬件要求

| 项目 | 说明 |
| --- | --- |
| CPU | Kunpeng 920 |

软件要求

软件要求如表2-2所示。

注意

* 不同HPC应用的依赖软件不同，建议按照如下步骤判断其依赖软件：
* 1、查看其上游社区是否提供安装指导文档；
* 2、搜索网络上是否已经有社区提供相关安装文档或博客；
* 3、尝试安装该软件，根据报错情况，决定安装哪些依赖软件；
* 4、咨询华为工程师是否有相关经验。

软件要求

| 项目 | 版本 | 下载地址 |
| --- | --- | --- |
| 毕昇编译器 | 2.1.0 | https://www.hikunpeng.com/zh/developer/devkit/compiler/bisheng |
| Hyper-MPI | 1.1.1 | <https://github.com/kunpengcompute/hmpi/archive/refs/tags/v1.1.1-huawei.tar.gz> |
| HUCX | 1.1.1 | <https://github.com/kunpengcompute/hucx/archive/refs/tags/v1.1.1-huawei.tar.gz> |
| XUCG | 1.1.1 | <https://github.com/kunpengcompute/xucg/archive/refs/tags/v1.1.1-huawei.tar.gz> |
| Zlib | 1.2.11 | <http://zlib.net/zlib-1.2.11.tar.gz> |
| HDF5 | 1.12.1 | https://support.hdfgroup.org/ftp/HDF5/releases/hdf5-1.12/hdf5-1.12.1/src/hdf5-1.12.1.tar.gz |
| NETCDF-C | 4.8.1 | https://downloads.unidata.ucar.edu/netcdf-c/4.8.1/netcdf-c-4.8.1.tar.gz |
| NETCDF-Fortran | 4.5.3 | https://downloads.unidata.ucar.edu/netcdf-fortran/4.5.3/netcdf-fortran-4.5.3.tar.gz |
| ParMETIS | 4.0.3 | http://glaros.dtc.umn.edu/gkhome/fetch/sw/parmetis/parmetis-4.0.3.tar.gz |
| WAVEWATCH III | 6.07.1 | https://github.com/NOAA-EMC/WW3/archive/refs/tags/6.07.1.tar.gz |
| 自带标准测试算例 | 6.07.1 | HPC应用自带算例，路径：6.07.1/WW3-6.07.1/regtests |
| 实际应用测试算例 | 6.07.1 | <ftp://polar.ncep.noaa.gov/tempor/ww3ftp/ww3_from_ftp.v6.07.tar.gz> |

操作系统要求

操作系统要求如表2-3所示。

操作系统要求

| 项目 | 版本 | 下载地址 |
| --- | --- | --- |
| openEuler | openEuler 20.03 SP3 | https://repo.openeuler.org/openEuler-20.03-LTS-SP3/ |
| Kernel | 4.19.90 | https://gitee.com/openeuler/kernel |

# 移植规划数据

本章节给出WAVEWATCH III软件在移植过程中涉及到的相关软件安装规划路径的用途及详细说明。

移植规划数据

| 序号 | 软件安装规划路径 | 用途 | 说明 |
| --- | --- | --- | --- |
| 1 | /usr/local/bisheng | 毕昇编译器的安装规划路径。 | 这里的安装规划路径只是一个举例说明，建议部署在共享路径中。需要根据实际情况调整，**后续章节凡是遇到安装路径的命令，都以现网实际规划的安装路径为准进行替换，不再单独说明。** |
| 2 | /usr/local/hmpi | HyperMPI的安装规划路径。 |
| 3 | /usr/local/ZLIB | Zlib的安装规划路径。 |
| 4 | /usr/local/HDF5 | HDF5的安装规划路径。 |
| 5 | /usr/local/NETCDF | NETCDF-C和NETCDF-Fortran的安装规划路径。 |
| 6 | /usr/local/PARMETIS | ParMETIS的安装规划路径。 |
| 7 | /usr/local/WW3\_build | WAVEWATCH III的构建规划路径。 |
| 8 | /usr/local/WW3\_case | WAVEWATCH III的测试规划路径。 |

# 配置编译环境

前提条件

使用SFTP工具将各安装包上传至服务器对应目录下。

配置流程

配置流程

| 序号 | 配置项 | 说明 |
| --- | --- | --- |
| 1 | 搭建鲲鹏基座软件环境 | 参考4.1 搭建鲲鹏基座软件环境。 |
| 2 | 安装ZLIB | 参考4.2 [安装Zlib](#_安装Zlib)。 |
| 3 | 安装HDF5 | 参考4.3 [安装HDF5](#_安装HDF5)。 |
| 4 | 安装NETCDF-C | 参考4.4 [安装NETCDF-C](#_安装NETCDF-C)。 |
| 5 | 安装NETCDF-Fortran | 参考4.5 [安装NETCDF-Fortran](#_安装NETCDF-Fortran)。 |
| 6 | 安装ParMETIS | 参考4.6 [安装ParMETIS](#_安装ParMETIS)。 |

## 搭建鲲鹏基座软件环境

操作步骤

请参考《鲲鹏基座软件搭建指南.docx》

## 安装Zlib

操作步骤

使用PuTTY工具，以root用户登录服务器。

解压zlib安装包。

**tar -zxf zlib-1.2.11.tar.gz**

注：解压缩时请以实际压缩包名称为准。

进入解压后的目录。

**cd zlib-1.2.11**

进行配置。

**CC=clang ./configure --prefix=/usr/local/ZLIB**

编译，安装。

**make -j$(nproc)**

**make install**

配置环境变量。

**cat>"/usr/local/ZLIB/zlib\_modulefiles"<<EOF**

**#%Module1.0**

**conflict zlib**

**variable modfile [file normalize [info script]]**

**proc getModulefileDir {} {**

**variable modfile**

**set modfile\_path [file dirname \$modfile]**

**return \$modfile\_path**

**}**

**set pwd [getModulefileDir]**

**set ZLIB \$pwd**

**setenv ZLIB \$ZLIB**

**prepend-path LD\_LIBRARY\_PATH \$ZLIB/lib**

**EOF**

在当前shell中加载环境变量

**module use /usr/local/ZLIB/**

**module load /usr/local/ZLIB/zlib\_modulefiles**

若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。

**vi /etc/profile**

新增如下内容：

**module use /usr/local/ZLIB/**

**module load /usr/local/ZLIB/zlib\_modulefiles**

----结束

## 安装HDF5

操作步骤

使用PuTTY工具，以root用户登录服务器。

解压HDF5安装包。

**tar -xvf hdf5-1.12.1.tar.gz**

注：解压缩时请以实际压缩包名称为准。

进入解压后的目录。

**cd hdf5-1.12.1**

进行编译配置。

**CC=mpicc CXX=mpicxx FC=mpifort F77=mpi77 CFLAGS="-fPIC -DPIC" FCFLAGS="-fPIC -DPIC" FFLAGS="-fPIC" \**

**./configure --prefix=/usr/local/HDF5 --with-zlib=/usr/local/ZLIB --enable-shared --enable-fortran --enable-static --enable-parallel**

修改当前文件夹下的“libtool”文件。

**sed -i 's@\\\$wl-soname@-install\_name@g' libtool**

编译，安装。

**make -j$(nproc)**

**make install**

配置环境变量。

**cat>"/usr/local/HDF5/hdf5\_modulefiles"<<EOF**

**#%Module1.0**

**conflict hdf5**

**variable modfile [file normalize [info script]]**

**proc getModulefileDir {} {**

**variable modfile**

**set modfile\_path [file dirname \$modfile]**

**return \$modfile\_path**

**}**

**set pwd [getModulefileDir]**

**set HDF5 \$pwd**

**setenv HDF5 \$HDF5**

**prepend-path LD\_LIBRARY\_PATH \$HDF5/lib**

**EOF**

在当前shell中加载环境变量。

**module use /usr/local/HDF5/**

**module load /usr/local/HDF5/hdf5\_modulefiles**

若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。

**vi /etc/profile**

新增如下内容：

**module use /usr/local/HDF5/**

**module load /usr/local/HDF5/hdf5\_modulefiles**

----结束

## 安装NETCDF-C

操作步骤

使用PuTTY工具，以root用户登录服务器。

解压NETCDF-C安装包。

**tar -xvf netcdf-c-4.8.1.tar.gz**

注：解压缩时请以实际压缩包名称为准。

进入解压后的目录。

**cd netcdf-c-4.8.1**

进行编译配置。

**CC=mpicc CXX=mpicxx CPPFLAGS="-I/usr/local/HDF5/include -I/usr/local/ZLIB/include" \**

**CFLAGS="-I/usr/local/HDF5/include -I/usr/local/ZLIB/include --fPIC" \**

**LDFLAGS="-L/usr/local/HDF5/lib -L/usr/local/ZLIB/lib" \**

**./configure --prefix=/usr/local/NETCDF --enable-shared --enable-netcdf-4 --enable-dap --with-pic --disable-doxygen --enable-static --enable-largefile**

编译，安装。

**make -j$(nproc)**

**make install**

配置环境变量。

**cat>"/usr/local/NETCDF/netcdf\_modulefiles"<<EOF**

**#%Module1.0**

**conflict netcdf**

**variable modfile [file normalize [info script]]**

**proc getModulefileDir {} {**

**variable modfile**

**set modfile\_path [file dirname \$modfile]**

**return \$modfile\_path**

**}**

**set pwd [getModulefileDir]**

**set NETCDF \$pwd**

**setenv NETCDF \$NETCDF**

**prepend-path LD\_LIBRARY\_PATH \$NETCDF/lib**

**EOF**

在当前shell中加载环境变量。

**module use /usr/local/NETCDF/**

**module load /usr/local/NETCDF/netcdf\_modulefiles**

若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。

**vi /etc/profile**

新增如下内容：

**module use /usr/local/NETCDF/**

**module load /usr/local/NETCDF/netcdf\_modulefiles**

**----结束**

## 安装NETCDF-Fortran

操作步骤

使用PuTTY工具，以root用户登录服务器。

解压 NETCDF-Fortran 安装包。

**tar -zxvf netcdf-fortran-4.5.3.tar.gz**

注：解压缩时请以实际压缩包名称为准。

进入解压后的目录。

**cd netcdf-fortran-4.5.3**

进行编译配置。

**CC=mpicc CXX=mpicxx FC=mpifort F77=mpifort CPPFLAGS="-I/usr/local/HDF5/include -I/usr/local/NETCDF/include" \**

**CFLAGS="-I/usr/local/HDF5/include -I/usr/local/NETCDF/include -fPIC -DPIC" \**

**FCFLAGS="-I/usr/local/HDF5/include -I/usr/local/NETCDF/include -fPIC -DPIC" \**

**LDFLAGS="-L/usr/local/HDF5/lib -L/usr/local/NETCDF/lib" \**

**./configure --prefix=/usr/local/NETCDF --enable-shared --enable-dap --enable-pic --disable-doxygen --enable-static --enable-largefile**

修改当前文件夹下的“libtool”文件。

**sed -i \**

**-e 's/\\\$wl-soname/-install\_name/g' \**

**-e 's/\\\$wl--whole-archive/-Wl,-force-load,/g' \**

**-e 's/\\\$wl--no-whole-archive//g' libtool**

编译，安装。

**make -j$(nproc)**

**make install**

注：NETCDF-FORTRAN与NETCDF-C的环境变量一样，因此，此处不再设置。

----结束

## 安装ParMETIS

操作步骤

使用PuTTY工具，以root用户登陆服务器。

解压ParMETIS安装包。

**tar -xvf parmetis-4.0.3.tar.gz**

注：解压缩时请以实际压缩包名称为准。

进入解压后的目录。

**cd parmetis-4.0.3**

修改“metis/include/metis.h”文件。

**sed -i -e 's/\#define IDXTYPEWIDTH 32/\#define IDXTYPEWIDTH 64/g' metis/include/metis.h**

进入metis目录，编译安装metis。

**cd metis**

**make config shared=1 prefix=/usr/local/PARMETIS/metis**

进入上层目录。

**cd ..**

修改当前目录下的CMakeLists.txt文件。

**sed -i -e '29i add\_compile\_options(-fPIC)' CMakeLists.txt**

编译，安装。

**make config shared=1 prefix=/usr/local/PARMETIS/metis**

**make install**

配置环境变量。

**cat>"/usr/local/PARMETIS/parmetis\_modulefiles"<<EOF**

**#%Module1.0**

**conflict parmetis**

**variable modfile [file normalize [info script]]**

**proc getModulefileDir {} {**

**variable modfile**

**set modfile\_path [file dirname \$modfile]**

**return \$modfile\_path**

**}**

**set pwd [getModulefileDir]**

**set PARMETIS \$pwd**

**setenv PARMETIS \$PARMETIS**

**prepend-path LD\_LIBRARY\_PATH \$PARMETIS/metis/lib**

**EOF**

在当前shell中加载环境变量。

**module use /usr/local/PARMETIS/**

**module load /usr/local/PARMETIS/parmetis\_modulefiles**

若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。

**vi /etc/profile**

新增如下内容：

**module use /usr/local/PARMETIS/**

**module load /usr/local/PARMETIS/parmetis\_modulefiles**

---结束

# 获取源码

操作步骤

下载WAVEWATCH III安装包。

下载地址：https://github.com/NOAA-EMC/WW3/archive/refs/tags/6.07.1.tar.gz

----结束

# 编译和安装

操作步骤

使用PuTTY工具，以root用户登录服务器。

安装环境依赖包。

**yum install -y environment-modules patch wget**

**set -e**

进入构建目录。

**cd /usr/local/WW3\_build**

通过下面链接获取源码。

**https://github.com/NOAA-EMC/WW3/archive/refs/tags/6.07.1.tar.gz**

解压源码包。

**tar xf 6.07.1.tar.gz && rm -f 6.07.1.tar.gz**

进入解压后的目录。

**cd WW3-6.07.1**

制作环境变量文件。

**vi env\_ww3.sh**

写入以下内容并保存：

############# bisheng+ hmpi #################

export PATH=/usr/local/bisheng-compiler-2.1.0-aarch64-linux/bin:$PATH

export LD\_LIBRARY\_PATH=/usr/local/bisheng-compiler-2.1.0-aarch64-linux/lib:$LD\_LIBRARY\_PATH

export PATH=/usr/local/hmpi/bin:/usr/local/hucx/bin:$PATH

export LD\_LIBRARY\_PATH=/usr/local/hmpi/lib:/usr/local/hucx/lib:$LD\_LIBRARY\_PATH

############# netcdf #################

export NETCDF\_DIR=/usr/local/NETCDF

export NETCDF\_LIBDIR=$NETCDF\_DIR/lib

export NETCDF\_INCDIR=$NETCDF\_DIR/include

export PATH=$NETCDF\_DIR/bin:$PATH

export LD\_LIBRARY\_PATH=$NETCDF\_LIBDIR:$LD\_LIBRARY\_PATH

export WWATCH3\_NETCDF=NC4

export NETCDF\_CONFIG=$NETCDF\_DIR/bin/nc-config

############# metis #################

export METIS\_PATH=/usr/local/PARMETIS/metis

export PATH=$METIS\_PATH/bin:$PATH

export LD\_LIBRARY\_PATH=$METIS\_PATH/lib:$LD\_LIBRARY\_PATH

加载环境变量。

**source env\_ww3.sh**

注：运行WAVEWATCH III的算例时依然调用此路径下的依赖库文件。

进入模型编译环境配置路径。

**cd ./model/bin**

制作“w3\_setup”文件的补丁。

**vi w3\_setup.patch**

写入以下内容并保存：

202,203c202,203

< comp\_fc=gfortran

< comp\_cc=gcc

---

> comp\_fc=flang

> comp\_cc=clang

233c233

< read OK

---

> OK='y'

256c256

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr='printer'

263c263

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr='flang'

270c270

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr='clang'

280c280

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr=$temp\_dir

301c301

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr='yes'

313c313

< instr="$NULL" ; read instr

---

> instr="$NULL" ; instr='yes'

333c333

< read OK

---

> OK='y'

为“w3\_setup”文件打上补丁。

**patch -p0 w3\_setup w3\_setup.patch**

配置编译环境。

**cd /usr/local/WW3\_build/WW3-6.07.1**

**./model/bin/w3\_setup model**

进入model/bin路径，配置使用毕昇编译器的编译环境。

**cd ./model/bin**

制作“cmplr.env”，“comp.Gnu”和“link.Gnu”文件的补丁。

**vi cmplr.env.patch**

写入以下内容并保存：

169c169

< comp\_seq='gfortran'

---

> comp\_seq='flang'

**vi comp.Gnu.patch**

写入以下内容并保存：

93c93

< comp=gfortran

---

> comp=flang

**vi link.Gnu.patch**

写入以下内容并保存：

95c95

< comp=gfortran

---

> comp=flang

为“cmplr.env”，“comp.Gnu”和“link.Gnu”文件打上补丁。

**patch -p0 cmplr.env cmplr.env.patch**

**patch -p0 comp.Gnu comp.Gnu.patch**

**patch -p0 link.Gnu link.Gnu.patch**

设置临时环境变量。

**export ww3\_dir=/usr/local/WW3\_build/WW3-6.07.1/model**

清理编译环境。

**./w3\_clean -c**

设置WAVEWATCH III模型编译环境。

**./w3\_setup $ww3\_dir -c Gnu -s UKMO**

注：需要用户定义的参数，参数-c后面指定编译器，-s后面指定开关文件。

编译WAVEWATCH III模型。

**./w3\_new**

**./w3\_make ww3\_grid**

注：其实这里仅做编译测试，实际上WAVEWATCH III模型的每个算例运行，都使用各自的脚本程序根据具体的参数设置（包括编译器、并行方式、模块开关等），重新编译和运行程序。

----结束

# 运行和验证

操作步骤

WAVEWATCH III模型的算例测试分2部分：（1）自带标准测试算例；（2）实际应用测试算例（需要额外下载）。

1、自带标准测试算例运行

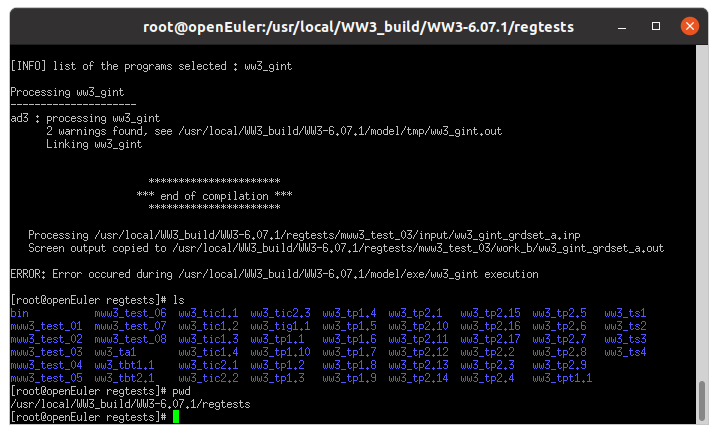
使用PuTTY工具，以root用户登录服务器。

进入自带标准算例目录。

**cd /usr/local/WW3\_build/WW3-6.07.1/regtests**

**ls**

显示WAVEWATCH III模型自带的标准算例，可用于检验WAVEWATCH III模型的功能模块，如下图，每个算例的文件夹中包含：input文件夹（运行算例的配置文件）和info（描述算例的实际设置和运行脚本命令等信息）。



设置测试标准算例的编译环境。

以运行mww3\_test\_01为例，用户进入mww3\_test\_01文件夹，查看info文件信息，选择使用的开关(switch)文件，该文件决定启用WAVEWATCH III模型的哪些功能模块。这里我们选择使用switch\_PR1，将该文件拷贝至/usr/local/WW3\_build/WW3-6.07.1/model/bin，然后设置编译环境，运行如下命令：

**cd mww3\_test\_01**

**cp switch\_PR1 /usr/local/WW3\_build/WW3-6.07.1/model/bin**

**cd /usr/local/WW3\_build/WW3-6.07.1/model/bin**

**./w3\_setup $ww3\_dir -c Gnu -s PR1**

这样完成编译运行mww3\_test\_01的PR1场景的设置。

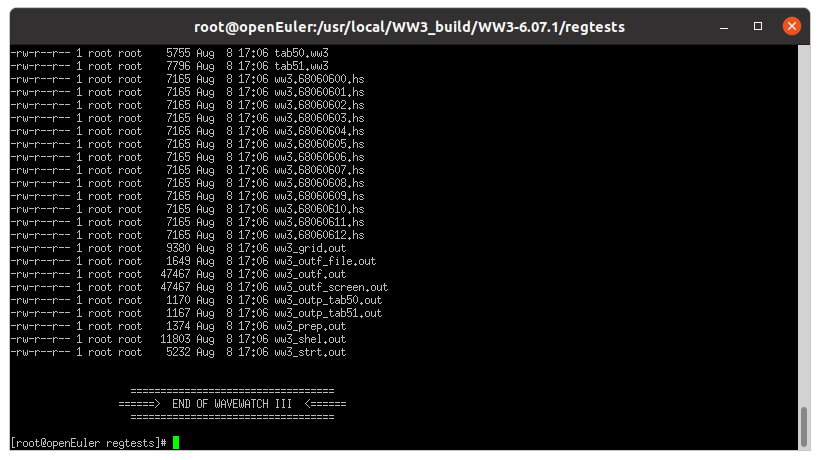
运行测试算例的测试脚本。

**cd /usr/local/WW3\_build/WW3-6.07.1/regtests**

**./bin/run\_test -s PR1 ../model mww3\_test\_01**

注：./bin/run\_test是运行所有测试算例的脚本程序，后面的参数确定了测试模型的不同功能模块，请参考各算例的info文件描述。

运行完毕后，显示下图，表示运行成功。



运行其他标准测试算例。

其他标准测试算例的设置和运行步骤与上述的mww3\_test\_01算例一样。经过测试通过的算例和脚本运行参数可参考如下命令：

./bin/run\_test -g curv -m grdset\_a -n 16 -p mpirun -s PR3\_UQ\_MPI\_SCRIP -w work\_ap\_PR3\_MPI\_SCRIP ../model mww3\_test\_02

./bin/run\_test -m grdset\_a -s PR3\_UQ -w work\_a ../model mww3\_test\_04

./bin/run\_test -m grdset\_a -n 16 -p mpirun -s ST6\_PR3\_UQ\_MPI -w work\_a ../model mww3\_test\_05

./bin/run\_test -m grdset ../model mww3\_test\_07

./bin/run\_test -p mpirun -n 16 -m grdset\_a ../model mww3\_test\_08

./bin/run\_test -n 16 $ww3\_dir ww3\_tbt1.1

./bin/run\_test -n 16 $ww3\_dir ww3\_tbt2.1

./bin/run\_test -g 2500m -w w2500m\_IC3 -i input\_IC3 ../model ww3\_tic1.1

./bin/run\_test -c gnu -i input\_IC3\_A0.5k ../model ww3\_tic1.2

./bin/run\_test -p mpirun -n 16 -w work\_IC4 -i input\_IC4 ../model ww3\_tic2.1

./bin/run\_test -s PR3\_UQ -w work\_PR3 ../model ww3\_tp1.1

./bin/run\_test -s PR1 -w work\_PR1 ../model ww3\_tp1.2

./bin/run\_test -s PR1 -w work\_PR1 ../model ww3\_tp1.3

./bin/run\_test -s PR1 -w work\_PR1 ../model ww3\_tp1.4

./bin/run\_test -s PR1 -w work\_PR1 ../model ww3\_tp1.5

./bin/run\_test -s PR1 ../model ww3\_tp1.6

./bin/run\_test ../model ww3\_tp1.7

./bin/run\_test ../model ww3\_tp1.8

./bin/run\_test ../model ww3\_tp1.9

./bin/run\_test -n 16 -o netcdf ../model ww3\_tp1.10

./bin/run\_test -n 16 -p mpirun -f -s PR1 ../model ww3\_tp2.1

./bin/run\_test -n 16 -p mpirun -f -s PR1\_MPI ../model ww3\_tp2.2

./bin/run\_test -n 16 -p mpirun -f -s PR1 ../model ww3\_tp2.3

./bin/run\_test -n 16 -p mpirun -f -s PR1 ../model ww3\_tp2.4

./bin/run\_test -n 16 -p mpirun -f -s PR1 ../model ww3\_tp2.5

./bin/run\_test -s ST0 ../model ww3\_tp2.6

./bin/run\_test -s ST0 ../model ww3\_tp2.7

./bin/run\_test -o netcdf ../model ww3\_tp2.8

./bin/run\_test -g a -s PR1 ../model ww3\_tp2.9

./bin/run\_test -s MPI -n 16 -p mpirun -f ../model ww3\_tp2.10

./bin/run\_test -n 16 -p mpirun -f ../model ww3\_tp2.11

./bin/run\_test -n 16 -p mpirun -i input -s MPI ../model ww3\_tp2.12

./bin/run\_test -n 16 -p mpirun -s PR3\_MPI ../model ww3\_tp2.13

./bin/run\_test -C OASIS -c Altix -s OASACM -f -o netcdf -p mpirun -n 16 ../model ww3\_tp2.14

./bin/run\_test -n 16 -p mpirun -s MPI -f -w work\_MPI ../model ww3\_tp2.15

./bin/run\_test -n 16 -p mpirun -f -s MPI ../model ww3\_tp2.16

./bin/run\_test -c gnu -S -s MPI -s NO\_PDLIB -w work\_a -g a -f -p mpirun -n 16 ../model ww3\_tp2.17

./bin/run\_test -s ST1 -w work\_ST1 ../model ww3\_ts1

./bin/run\_test -s ST1\_PR1 -w work\_ST1\_PR1 ../model ww3\_ts2

./bin/run\_test -s ST1\_PR1 -w work\_ST1\_PR1 ../model ww3\_ts3

./bin/run\_test -c Gnu -i input\_ug ../model ww3\_ts4

2、实际应用测试算例运行

实际应用测试算例一共有3个，需要额外下载。3个算例分别是：

（1）mww3\_case\_01：使用嵌套网格模拟北大西洋挪威近海区域的风浪；

（2）mww3\_case\_02：阿拉斯加近海地区的风浪模拟；

（3）mww3\_case\_03：NCEP业务化运行模式。

使用PuTTY工具，以root用户登录服务器。

进入测试算例路径。

**cd /usr/local/WW3\_case**

通过以下链接下载测试算例。

**ftp://polar.ncep.noaa.gov/tempor/ww3ftp/ww3\_from\_ftp.v6.07.tar.gz**

解压缩tar包。

**tar -xvf ww3\_from\_ftp.v6.07.tar.gz && rm -f ww3\_from\_ftp.v6.07.tar.gz**

**cd cases**

制作算例mww3\_case\_01文件的Patch补丁。

**vi mww3\_case\_01.patch**

写入以下内容并保存（注意黑体部分，设置了编译器、进程数和命令执行方式）：

--- mww3\_case\_01 2017-10-07 01:27:50.000000000 +0800

+++ mww3\_case\_01\_new 2022-05-30 09:23:16.323879039 +0800

@@ -58,18 +58,18 @@

# For batch run in mpi environment, set this to the location of your

# cases WW3 directory

-cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

+#cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

# Load modules if system requires

# Options for NCEP Zeus R&D machine

-module load intel mpt grads

+#module load intel mpt grads

# 0. Preparations -----------------------------------------------------------

set -e

# User must specify compiler string (as per in ww3\_dir/bin options) here:

- compstr="Intel"

+ **compstr="Gnu"**

# 0.a Define model input

@@ -135,12 +135,12 @@

# 0.f Parallel environment

MPI='yes' # run ww3\_multi in MPI mode

- proc=24

+ **proc=16**

# 0.g Set-up variables

case\_dir=`pwd`

- ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

+# ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

temp\_dir="$case\_dir/tmp\_case\_01"

mkdir -p ${temp\_dir}

@@ -417,7 +417,7 @@

if [ "$MPI" = 'yes' ]

then

- mpirun -np $proc $path\_e/ww3\_multi # > $path\_o/ww3\_multi.out

+ **mpirun --allow-run-as-root --mca btl ^openib -np $proc $path\_e/ww3\_multi**

# poe hpmcount $path\_e/ww3\_multi # > $path\_o/ww3\_multi.out

# poe $path\_e/ww3\_multi # > $path\_o/ww3\_multi.out

else

为mww3\_case\_01文件打上补丁。

**patch -p0 mww3\_case\_01 mww3\_case\_01.patch**

运行算例mww3\_case\_01。

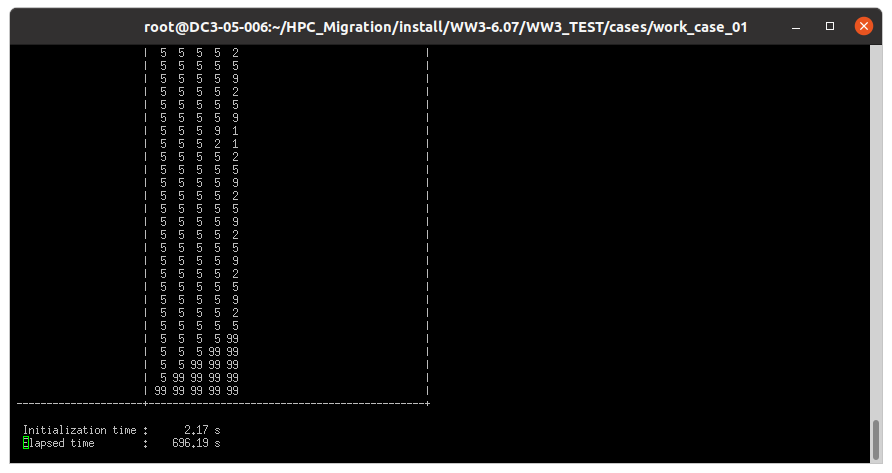
**./mww3\_case\_01**

当WAVEWATCH III程序运行结束后，如下图所示，表示WAVEWATCH III程序运行正常结束。



需要查看“work\_case\_01/log.mww3”日志中的“Elapsed time”数值，单位是“s”，数值越少性能越优。

运行结果样例如下图所示：



制作算例mww3\_case\_02文件的Patch补丁。

**vi mww3\_case\_02.patch**

写入以下内容并保存（注意黑体部分，设置了编译器、进程数和命令执行方式）：

60c60

< cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

---

> #cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

64c64

< module load intel mpt grads

---

> #module load intel mpt grads

71c71

< compstr="Intel"

---

> **compstr="Gnu"**

76c76

< ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

---

> # ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

134c134

< proc=24

---

> **proc=16**

412c412

< mpirun -np $proc $path\_e/ww3\_multi # > $path\_o/ww3\_multi.out

---

>  **mpirun --allow-run-as-root --mca btl ^openib -np $proc $path\_e/ww3\_multi**

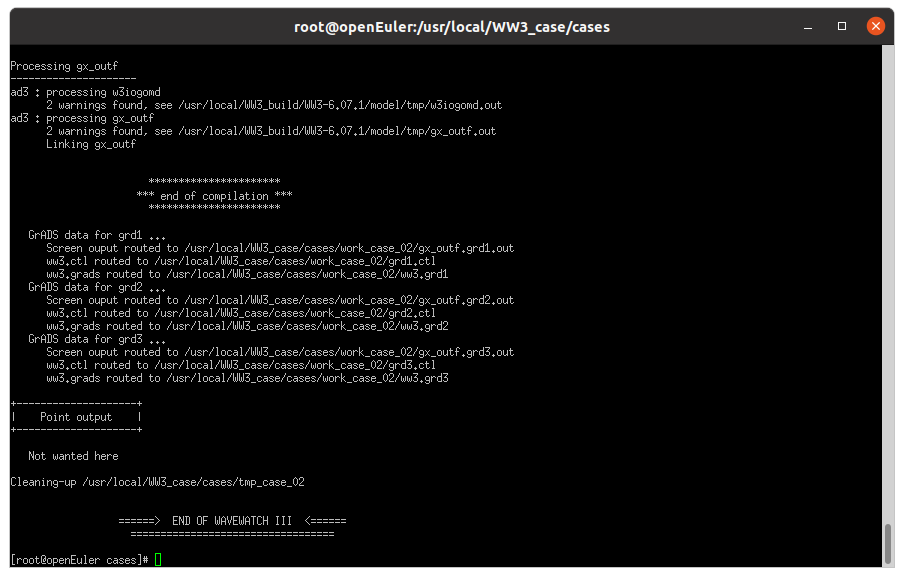
为mww3\_case\_02文件打上补丁。

**patch -p0 mww3\_case\_02 mww3\_case\_02.patch**

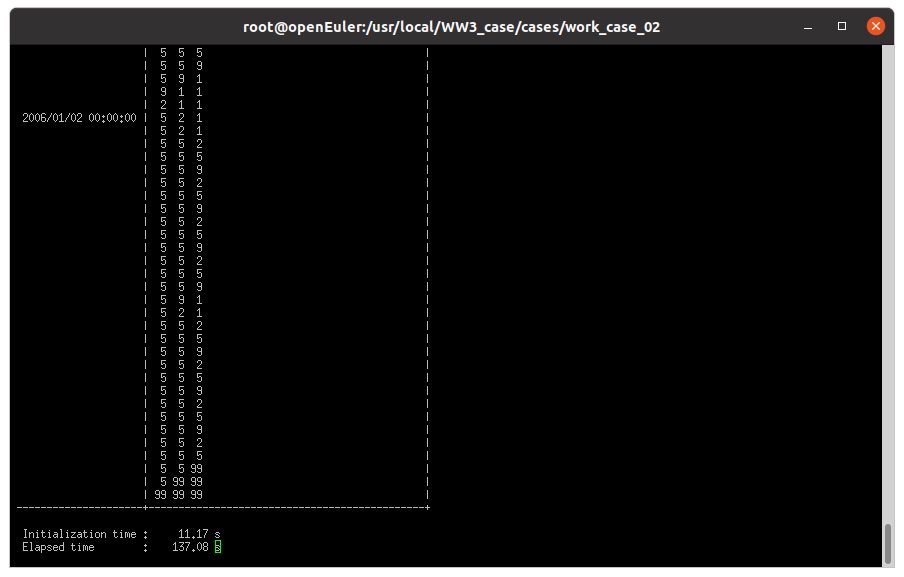
运行算例mww3\_case\_02。

**./mww3\_case\_02**

当WAVEWATCH III程序运行结束后，如下图所示，表示WAVEWATCH III程序运行正常结束。



需要查看“work\_case\_02/log.mww3”日志中的“Elapsed time”数值，单位是“s”，数值越少性能越优。



制作算例mww3\_case\_03文件的Patch补丁。

**vi mww3\_case\_03.patch**

写入以下内容并保存（注意黑体部分，设置了编译器、进程数和命令执行方式）：

65c65

< cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

---

> #cd /scratch1/portfolios/NCEPDEV/ocean/noscrub/Henrique.Alves/WW3\_CASES/cases

69c69

< module load intel mpt grads

---

> #module load intel mpt grads

145c145

< proc=20

---

> **proc=16**

148c148

< compstr="Intel"

---

>  **compstr="Gnu"**

155c155

< ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

---

> # ww3\_dir=`echo $case\_dir | sed 's/\/cases\/\*//g'`

446c446

< mpirun -np $proc $path\_e/ww3\_multi # > $path\_o/ww3\_multi.out

---

> **mpirun --allow-run-as-root --mca btl ^openib -np $proc $path\_e/ww3\_multi**

为mww3\_case\_03文件打上补丁。

**patch -p0 mww3\_case\_03 mww3\_case\_03.patch**

运行算例mww3\_case\_03。

**./mww3\_case\_03**

----结束

# 修订记录

| 发布日期 | 修订记录 |
| --- | --- |
| 2022-07-30 | 第一次正式发布。 |