

Архитектура европейской климатической модели: опыт установки на суперкомпьютерах в России

А.Н. Сальников^{1,2}, Н.П. Тучкова¹, И. Кирхнер³

ФИЦ ИУ РАН¹,
ВМК МГУ имени М.В. Ломоносова²,
Институт Метеорологии Свободного Университета Берлина³

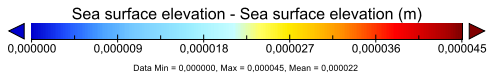
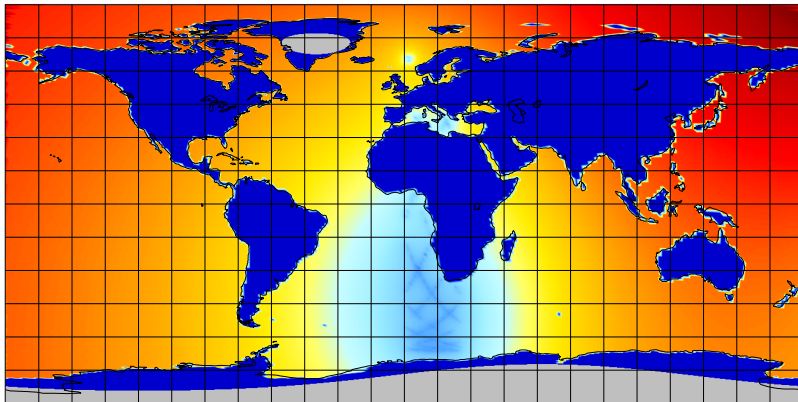
Задача моделирования климата

- Построение климатической модели позволяет учитывать особенности строительства крупных инфраструктурных объектов: города, порты, автомагистрали и т.п.
- Задача построения модели чрезвычайно сложна, поскольку требует учёта многих факторов.
- Модель строится не только с учетом физики процессов, но и истории метеорологических наблюдений.

Особенности работы с вычислительной областью

Все климатические модели работают с определенной сеткой значений параметров раскиданных по земному шару, однако необходимость изменения шага сетки весьма важна и зависит от исследуемого региона и сценария моделирования. Кроме того, метеостанции на суше и буи в океане, передающие данные наблюдений, покрывают не все участки земного шара и это приводит к необходимости экстраполяции данных на неохваченные наблюдениями районы и к решению задачи усвоения данных.

Sea surface elevation

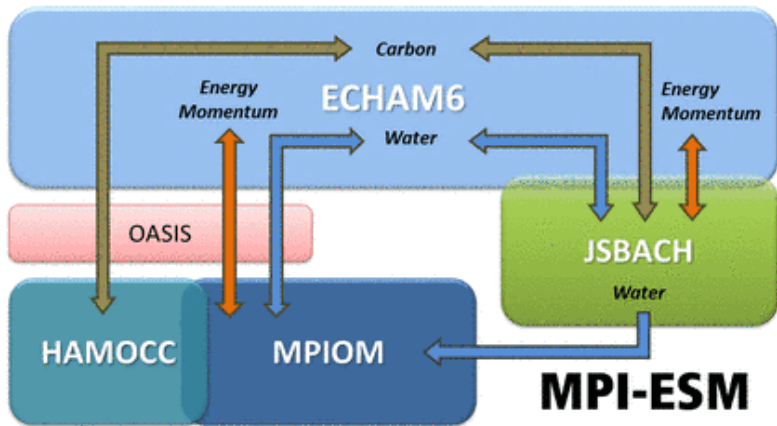


Европейская климатическая модель (MPI Earth System Model)

МРІЕСМ(ЕСНАМ6/МРІОМ/JSBACH/НАМОСС) — современная крупномасштабная совместная модель «океан-земля-атмосфера» европейского метеорологического сообщества.

- ЕСНАМ6 – модель динамики атмосферы
- МРІОМ – модель динамики океана, включающей динамику льда
- НАМОСС – блок газообмена, включающего динамику карбонных соединений в океане
- JSBACH – поведение атмосферы в приземном слое, учитывает биохимию и биофизику
- OASIS3-MCT – специальный интерфейс обмена «океан-земля-атмосфера»

Связь между компонентами



Особенности сетки в океанской части

- Сетка океанской части модели имеет следующую структуру: **биполярная (GR)** два полюса (например: Гренландия, Антарктида), **трехполюсная (TP)** три полюса (например: Северная Америка, Сибирь Антарктида)
- В программном коде используются:
 - CR (GR30L40) 3 град. – грубое разрешение для отладочных целей
 - LR (GR15L40) 1.5 град. – по-умолчанию для численного моделирования, (рекомендации Международной рабочей группы по моделированию климата на основе совместных моделей (WGCM,CMIP5)) (каждые ≈ 150 км.)
 - MR (TP04L40) 0.4 град. – высокое разрешение для учета вихревых потоков (по рекомендации WGCM,CMIP5 для сравнения с другими совместными моделями)
 - STORM (TP6ML40/80) 0.1 град. – очень высокое разрешение для учета вихревых потоков (каждые ≈ 10 км.)

Особенности сетки в атмосферной части

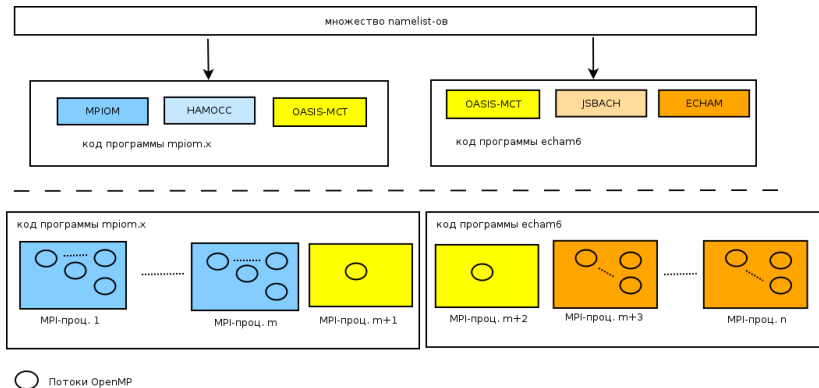
Сетка для атмосферной части также реализуется в разных конфигурациях для 47 и 95 уровней по высоте:

- CR (T031L47) – для отладочных целей;
- LR (T063L47) – разрешение по умолчанию для океанской маски GR15 (указана выше);
- MR (T063L95) – высокое разрешение для средних атмосфер для океанской маски TP04;
- HR (T127L95) – высокое разрешение;
- XR (T255L95) – экспериментальное очень высокое разрешение;

Структура программного кода климатической модели

- Код изначально ориентирован на выполнение в среде IBM кластера, под управлением **AIX** и **LoadLeveler**.
- Используется множество технологий и библиотек.
 - Компилируемый код написан на **C**, **C++**, **FORTRAN77**, **FORTRAN90**.
 - Для вспомогательных целей используется: **Python2.7**, **Perl**, **Korn Shell**.
 - Для распараллеливания используется: **MPI2** и **OpenMP**.
 - Из сторонних библиотек используется: **ESSL/MKL**, **NetCDF**.

Структура программы для одного шага по времени



В текущий момент инфраструктура развёрнута на суперкомпьютерах доступных авторам:

- “**Ломоносов-1**” — МГУ имени М.В. Ломоносова
- “**МВС-100**” — ИПМ РАН имени М.В. Келдыша (Требуется интеграция с runmvs)
- “**mistral**” — DKRZ Гамбург

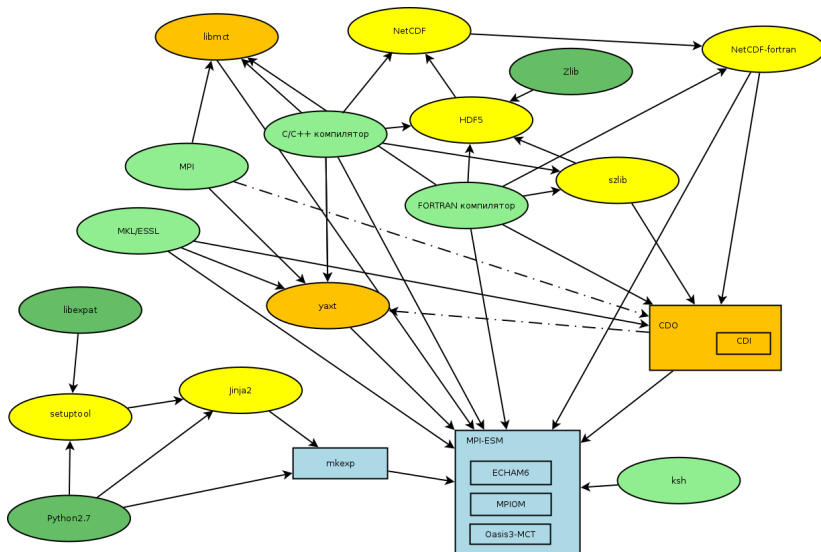
Развёрнутая инфраструктура для работы с моделью (2)

- 1 Создан каталог `__scratch/opt` в который затем устанавливается все программное обеспечение и библиотеки, как в корневой каталог файловой системы UNIX.
- 2 Модифицировались переменные окружения в файлах `.profile` так, чтобы динамически подгружаемые библиотеки и программы искали из каталогов `__scratch/opt/bin` и `__scratch/opt/lib`. Также здесь производится настройка на опеределенные версии компиляторов Intel, реализаций MPI и версии библиотеки MKL.

Развёрнутая инфраструктура для работы с моделью (3)

- 3 Резвернут git репозиторий, в который помещается исходный код модели скачанный с сайта разработчиков. В дальнейшем в репозиторий вносятся изменения необходимые для того, чтобы скомпилировать и запустить модель в чужеродной среде.
- 4 В домашнем каталоге компилировались необходимые компоненты от которых зависит код модели так, что установка их производится в `__scratch/opt`.

Список зависимостей



- 5 Вносятся изменения в заголовочный скрипт *configure*, а так же в некоторые Makefile и файлы отвечающие за интерфейс языка C и языка Фортран-90.
- 6 Запускается настройка кода с параметрами указанными на следующем слайде и после компиляция. Далее весь каталог с моделью копируется в *_scratch*, в дальнейшем вся работа ведётся именно там.

Параметры для настройки и компиляции

```
#!/bin/bash
```

```
export CC=mpiicc  
export CXX=mpiicpc  
export FC=mpiifort  
export F77=mpiifort  
export MPIFC=mpiifort
```

```
export FORT_REAL_LEN=8
```

```
export FCFLAGS="-cpp -mkl -I$HOME/_scratch/opt/include"  
export CFLAGS="-std=c99 -no-ipo -mkl -I$HOME/_scratch/opt/include"  
export FCLIBS="-mkl -no-ipo -L$HOME/_scratch/opt/lib -lnetcdf -lnetcdf"
```

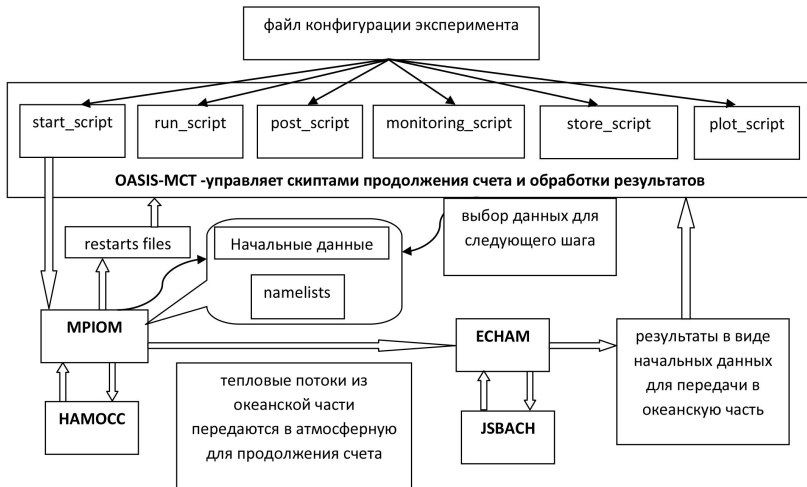
```
export FCFLAGS_ECHAM="-mkl -no-ipo -cpp -O2 -I$HOME/_scratch/opt/include"  
export FCFLAGS_MPIOM="-cpp -mkl -no-ipo -O2 -I$HOME/_scratch/opt/include"
```

```
export HDF5ROOT=$HOME/_scratch/opt  
export NETCDFROOT=$HOME/_scratch/opt  
export NETCDFLIB="-lnetcdf -lnetcdf"  
export CDIRoot=$HOME/_scratch/opt
```

```
export OASIS3MCTROOT="$HOME/_scratch/model_v2_1"  
export prefix="$HOME/_scratch/model_v2_1"
```

```
export PATH="/common/intel/impi/4.0.0.028/bin64:$PATH"  
./configure --prefix="$HOME/_scratch/model_v2_1"\  
--host=x86_64-unknown-linux-gnu
```


- 7 производится адаптация скриптов, под конкретную систему ведения очередей (модифицируются файлы “функции” *util/running/functions*).
- 8 Запускается python код *util/mkexp* для создания структуры каталогов эксперимента с файлом, описывающем параметры эксперимента. Он создает структуру каталогов для обеспечения жизненного цикла эксперимента.
- 9 Для запуска в очередь ставится скрипт *имя_эксперимента.run_start*.



- Запуски экспериментов показали, что численная схема устойчива у усвоению данных по методу Калмана, что позволяет применять методику для климатических прогнозов на 100-200 лет.
- Результаты проведённой работы позволяют выявить реальные проблемы разворачивания сложных программных систем на “отечественных” кластерных системах.

Спасибо за внимание!