C-Coupler: A Coupler for Earth System Model

Xiaoge Wang, Li Liu, Guangwen Yang
Department of Computer Science and Technology
Center for Earth System Science
Tsinghua University, Beijing, China
Outline

• **Background**
• Design Consideration
• Architecture
• Details
• Future work
Two Projects

- “Software platform for Supporting Earth System Model Research and Development” (210-2012)
  - Massive Data Management
  - Visualization for ESM
  - Integrated Software Development Environment
  - Tools for Parallel Programming
  - Application Showcases

  - Parallel Numerical Algorithms and Performance Optimization
  - Framework for Parallel Application Development
  - Parallel Coupling Technology for ESM
  - Development of a High-performance Physical Climate System Model


4
Coupling Technology Today in China

• Coupling software in use
  – NCAR couplers
    • Cpl5/Cpl6: IAP (LASG), Beijing Normal University, National Climate Center
  – OASIS
    • OASIS3/OASIS4: regional coupling, atm-ocn coupling, typhoon model coupling

• Need for a Coupler
  – Which coupler should be used?
  – Develop a new coupler to meet the need?

• C-Coupler project started in 2010
Demand for Coupler

• User Friendliness and Flexibility
  – Easier parameter control
  – Easier customization of coupling method

• Higher Performance
  – Better Scalability
  – System-level Load balance
  – Efficient global communication

• More Functions
  – Interactive ensemble
  – 3D coupling
Outline

• Background
• **Design Consideration**
• Architecture
• Details
• Future work
General Idea in a Picture
Software Technologies

• Component oriented programming
  – Component based integration
  – Component reuse
    • Model reuse
    • Coupling algorithm reuse

• Software configuration management
  – Model configuration
  – Coupling algorithm configuration
  – Coupler configuration
  – ESM case configuration
Features of C-Coupler

• User friendly and flexible
  – Both GUI and scripts for configuration
  – More configurable points

• High performance
  – Efficient communication
    • Parallel communication
    • Asynchronous communication
    • Direct communication between models
  – System-level load balance
    • Monitoring the performance of models and coupler
    • Schedule the coupling computation among models and the coupler
  – PIO
    • A PIO-specific module based on PNetCDF
Features of C-Coupler (cont.)

• Supporting Functions
  – Interactive ensemble
  – 3D coupling
  – Regional coupling
Outline

• Background
• Design Consideration
• Architecture
• Details
• Future work
Architecture(1)
Architecture (2)
Outline

• Background
• Design Consideration
• Architecture
• Some Details
• Future work
Model Configuration

1. Set model name
2. Select / set model type
3. Select / import grid info
4. Set model region
5. Select / set fields for I/O
6. Select / set fields for coupling
7. Set the algorithm sequence of initial stage
8. Configure model namelist
9. Import model codes
10. Insert coupling interfaces into model code
Coupling Algorithm Configuration

1. Set algorithm name
2. Select / set algorithm type
3. Set the constraints for using this algorithm
4. Input algorithm description
5. Set the info of input and output variables
   (1) The component type
   (2) Field name
   (3) The component type of computing grid
6. Register the interface of the algorithm
7. Import the code of the algorithm
Coupler Configuration

1. Set coupler name
2. Select / set the info of local variables of coupler
3. Register the namelist of coupler
4. Set the coupling flow
   (1) initialization stage
   (2) kernel stage
   (3) finalization stage
ESM Case Configuration

**Front End**

1. Select models
2. Select coupler configuration

**Back End**

1. Select the ensemble algorithms
2. Select the grid for coupling

- Select remap algorithms, flux algorithms, etc.

**Front End**

- Validate the component models match the coupler version or not;
- Detect requirement for ensemble

**Back End**

- Detect remap requirement according coupling data flow and grids of models and coupler

**Front End**

- Detect the diagnosis algorithm can be used, according to the selected algorithm

**Back End**

- Generate the configure files used for ESM

1. Select diagnosis algorithms, trace algorithms
2. Configure data model
3. Set the frequency of coupling, restart, I/O, diagnosis, etc.
4. Set the namelist for models and coupler
Coupling Driver

1. Import configuration files
2. Generate dynamic parallel decomposition of coupler
3. Generate parallel communication
4. Generate the coupling algorithm object sequence for all stages
5. Run the coupling algorithm sequence for the initialization stage

For each in the coupling algorithm object sequence for kernel stage
   If (time to run)
      Run current coupling algorithm object

whether finish coupling kernel?

1. Run the coupling algorithm sequence for the finalization stage
2. Destroy all objects
Outline

• Background
• Design consideration
• Architecture
• Some Details
• Future work
Future Work

• Complete the first version of C-coupler
  • FGOALS

• Integrity and robustness

• Scalability and efficiency

• Portability and adaptability
Thanks!
Q & A