The Cupid Training Environment

Tutorial

This tutorial will take you through the process of installing the Cupid Plugin for the Eclipse Integrated Development Environment (IDE) and using some of its basic functionality. Cupid is a training environment for graduate students and early career geoscience modelers who have some basic programming experience, but are new to the geoscience model development process. Cupid simplifies the process of configuring a computational environment capable of compiling and executing high-performance geoscience models such as NASA's ModelE and models using advanced coupling infrastructure software such as the Earth System Modeling Framework (ESMF).

IDEs package a lot of development tools into a single application to help manage and simplify the software development workflow. Although IDEs aim to increase developer productivity, they can still introduce a steep learning curve for new geoscience modelers. Some challenges with using IDEs include:

- Understanding the basic steps involved in moving from source code to a running model
- Making sense of the many development tools and features available in the IDE
- Setting up a high-performance computational environment capable of configuring, compiling and executing model code
- Configuring the IDE to connect to remote computational environments

Using Cupid, you will be able to select a training scenario and, within a few minutes, configure, compile, execute, and view the output of both skeleton models and realistic models.

More information about Cupid can be found at: https://earthsystemcog.org/projects/cupid/

Install Eclipse and the Cupid Plugin

1. **Download and install Eclipse for Parallel Application Developers, version 4.3.1 SR1 (Kepler).**

   The main download page is: http://www.eclipse.org/downloads/.

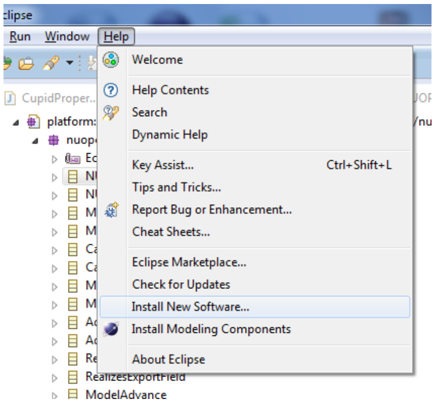
   There is a list of available Eclipse packages. Be sure to choose "Eclipse for Parallel Application Developers" as it will come pre-bundled with the necessary plugins for working with remote systems.

2. **Unpack the downloaded file into a local directory and run Eclipse by double clicking on the Eclipse executable.**

   The first time you start Eclipse, you will be prompted to select a location for your workspace. Choose an empty folder.

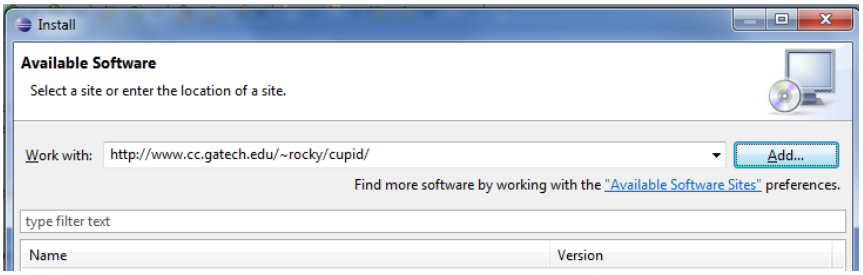
3. Install the Cupid Plugin from the Cupid Update Site.

a. Click Help→Install New Software



b. Put the Cupid Update Site URL into "Work with..." You will be prompted to give the update site a name of your choosing.

The URL is: **`http://www.cc.gatech.edu/~rocky/cupid/`**



Install

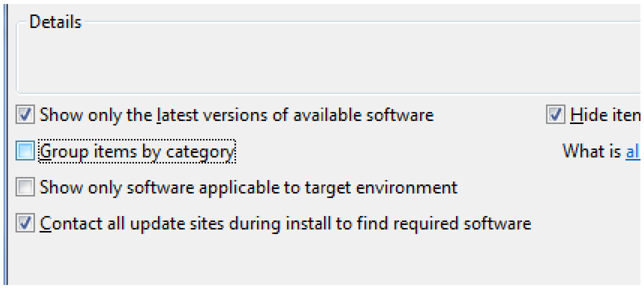
Available Software

Select a site or enter the location of a site.

Work with: http://www.cc.gatech.edu/~rocky/cupid/

Find more software by working with the "Available Software Sites" preferences.

c. Uncheck the "Group items by category" option.



Details

☑ Show only the latest versions of available software

☐ Group items by category

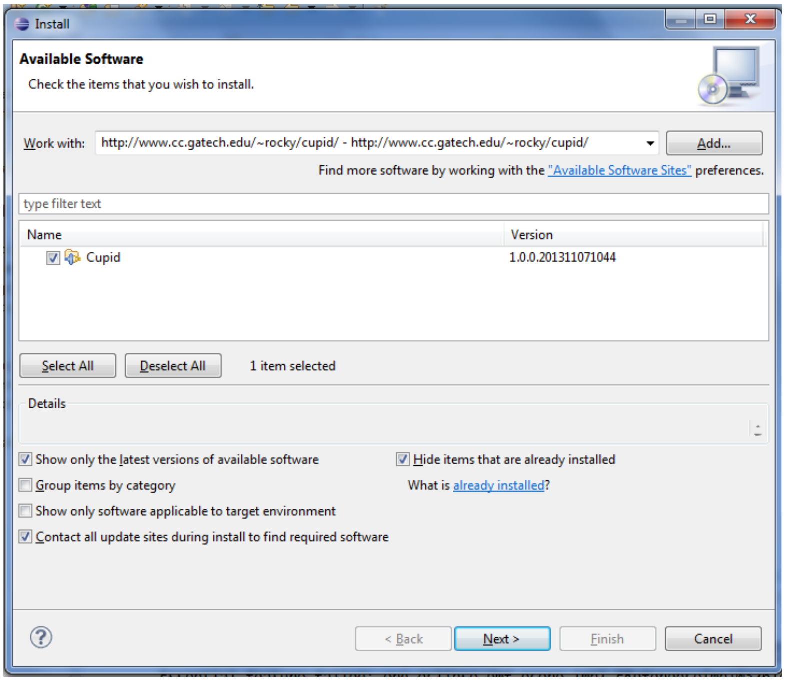
☐ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

☑ Hide iten

What is al

d. Select "Cupid" from the list and click Next.



Install

Available Software

Check the items that you wish to install.

Work with: http://www.cc.gatech.edu/~rocky/cupid/ - http://www.cc.gatech.edu/~rocky/cupid/

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
☑ Cupid	1.0.0.201311071044

1 item selected

Details

☑ Show only the latest versions of available software

☑ Hide items that are already installed

□ Group items by category

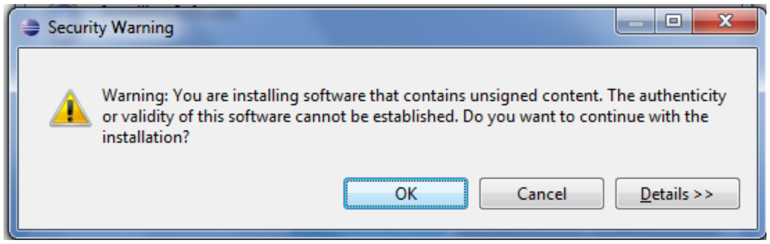
What is already installed?

□ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

e. You will need to click Next a couple more times and accept the license agreements. Then click Finish. The Cupid plugin and its dependencies will be downloaded and installed.

During the process, you may receive a message that the software contains unsigned content. Click OK.



Security Warning

Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

f. After installation, you will be prompted to restart Eclipse. Click Yes.

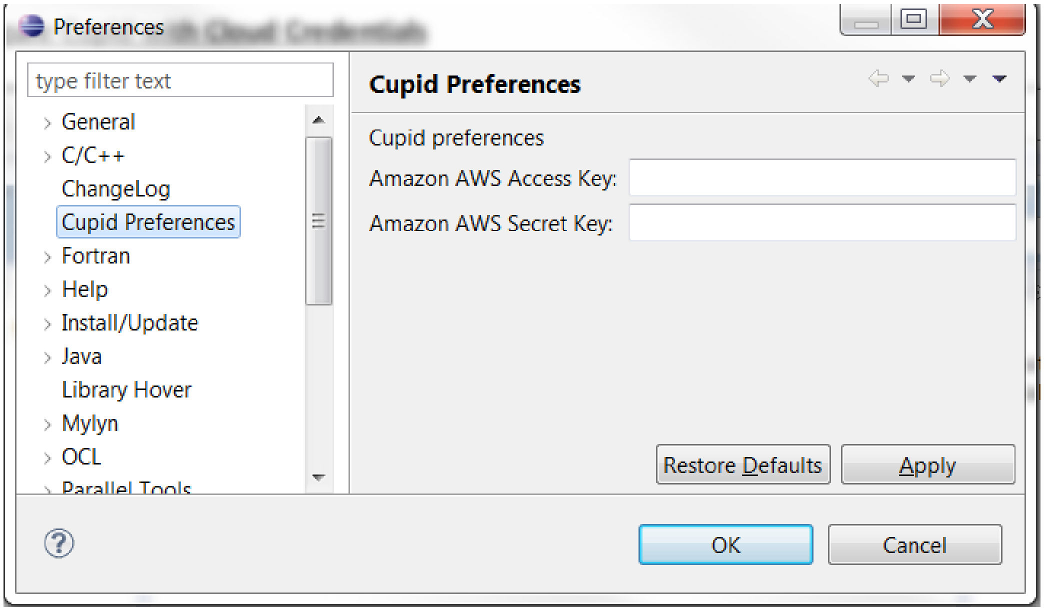
Configure Cupid with Cloud Credentials

Cupid creates cloud-based virtual machine instances to serve as the computational environment for the training scenarios. Currently, the only supported cloud provider is Amazon EC2. The first time you run Eclipse, you will need to set the Amazon EC2 credentials. You may use your own credentials or request access to the NESII cloud.

(Note: Cupid automatically launches Amazon machine instances when you create a new project. Alarms are set up to automatically kill instances after 50 minutes of idle CPU utilization. *However, it is your responsibility to ensure that any unused instances are terminated to avoid unnecessary cloud computing charges to your account*.)

1. In the Eclipse menu, select Window→Preferences. Then select Cupid Preferences in the list on the left.

2. Enter your Amazon Web Services (AWS) access key and secret key.



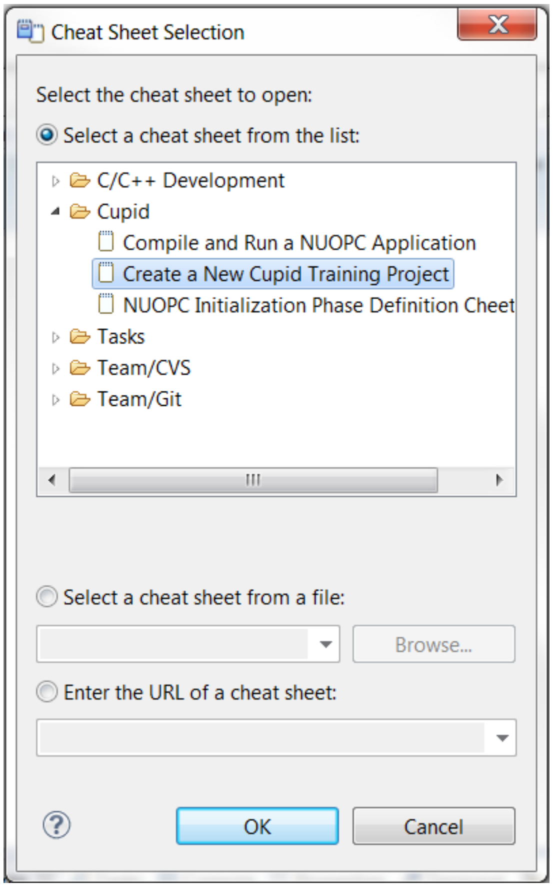
3. Click OK.

Using Cheat Sheets

An Eclipse Cheat Sheet is a step-by-step guide for performing some task in the IDE. Several Cheat Sheets have been provided that will help you get started using Cupid.

1. Select Help→Cheat Sheets from the Eclipse menu.

2. In the list at the top, choose the Cheat Sheet called "Create a New Cupid Training Project" in the Cupid folder.



Cheat Sheet Selection

Select the cheat sheet to open:

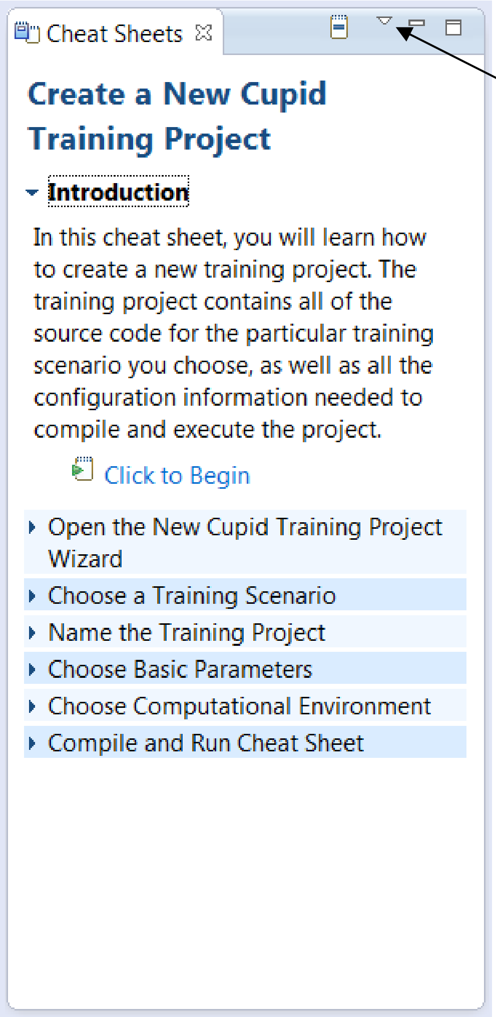
● Select a cheat sheet from the list:

- C/C++ Development
- Cupid
  - Compile and Run a NUOPC Application
  - Create a New Cupid Training Project
  - NUOPC Initialization Phase Definition Cheet
- Tasks
- Team/CVS
- Team/Git

○ Select a cheat sheet from a file:

○ Enter the URL of a cheat sheet:

3. Click OK. You should see a new view in Eclipse showing the Cheat Sheet. At any time, to change Cheat Sheets, click on the small downward-facing triangle in the Cheat Sheet toolbar.



Create a New Cupid Training Project

Introduction

In this cheat sheet, you will learn how to create a new training project. The training project contains all of the source code for the particular training scenario you choose, as well as all the configuration information needed to compile and execute the project.

Click to Begin
Open the New Cupid Training Project Wizard
Choose a Training Scenario
Name the Training Project
Choose Basic Parameters
Choose Computational Environment
Compile and Run Cheat Sheet

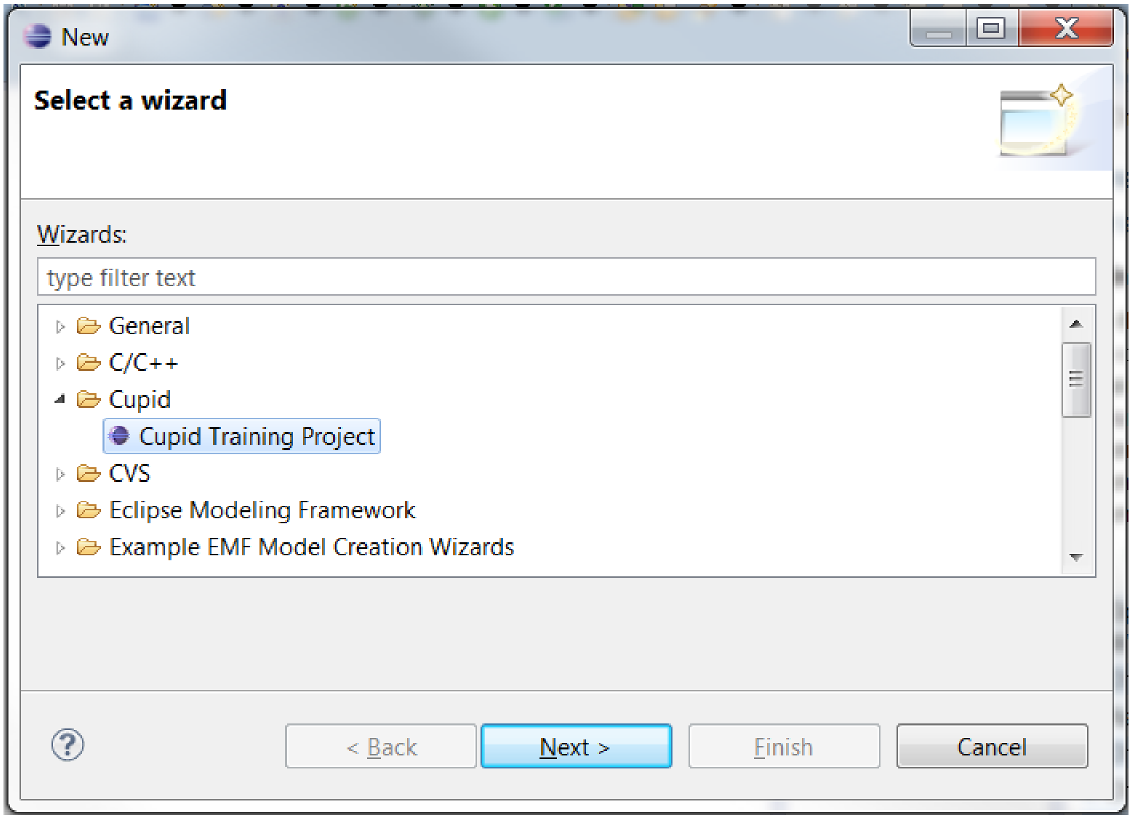
Click here to change Cheat Sheets

4. Follow the steps in the Cheat Sheet to create a new Cupid Training Project. The steps for creating a new project are also included in the next section of this tutorial.

Create a New Cupid Training Project

This section of the tutorial describes how to create a new Cupid Training Project. It is the same steps as outlined in the Cheat Sheet mentioned in the previous section.

1. Click File→New→Other… and choose Cupid Training Project in the Cupid folder. Click Next.



New

Select a wizard

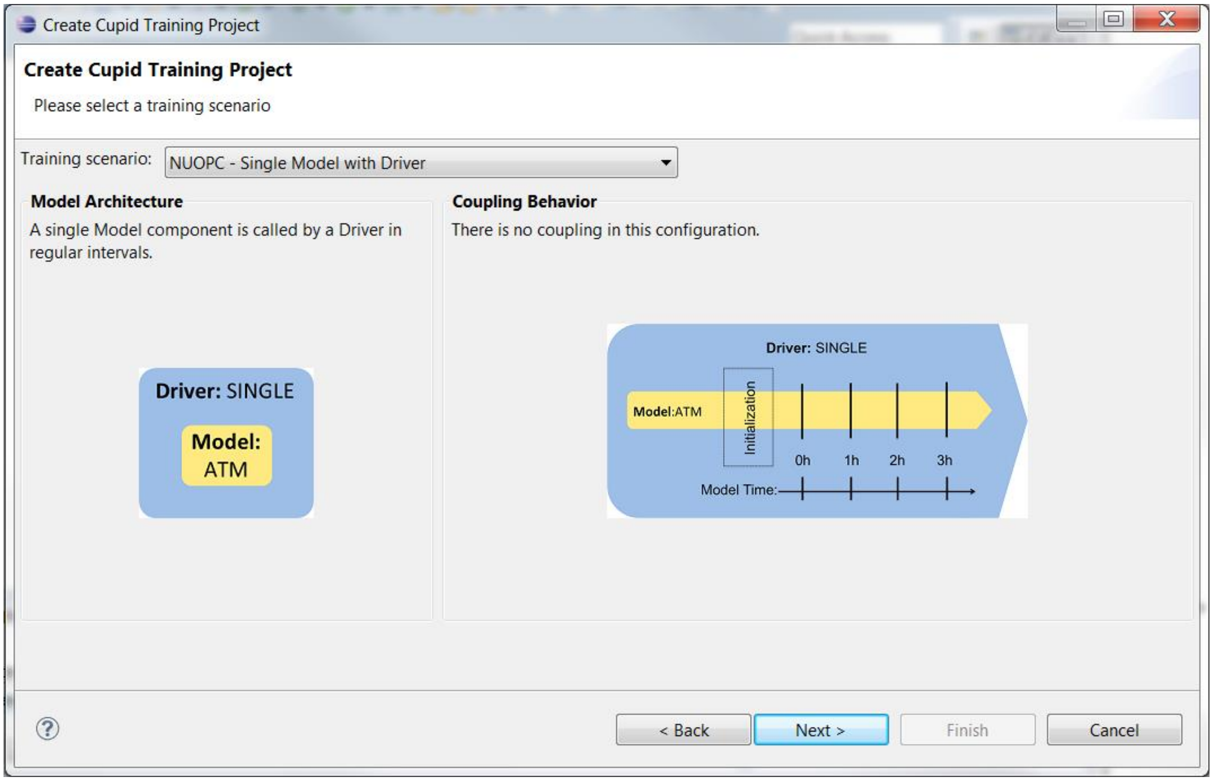
Wizards:

type filter text

2. On the first page of the wizard, choose a training scenario. Several options are available, but currently on the first option is supported:

   - NUOPC – Single Model with Driver
   - NUOPC – Coupled Atmosphere-Ocean Driver ***(coming soon)***
   - NUOPC – Coupled Atmosphere-Ocean with Mediator and Driver ***(coming soon)***
   - ModelE – Basic Configuration (EM20 rundeck) ***(coming soon)***

   Changing the training scenario in the list will update the screen to show the model architecture and coupling behavior of the selected scenario. Choose the scenario “NUOPC – Single Model with Driver” and click Next.



Create Cupid Training Project

Create Cupid Training Project

Please select a training scenario

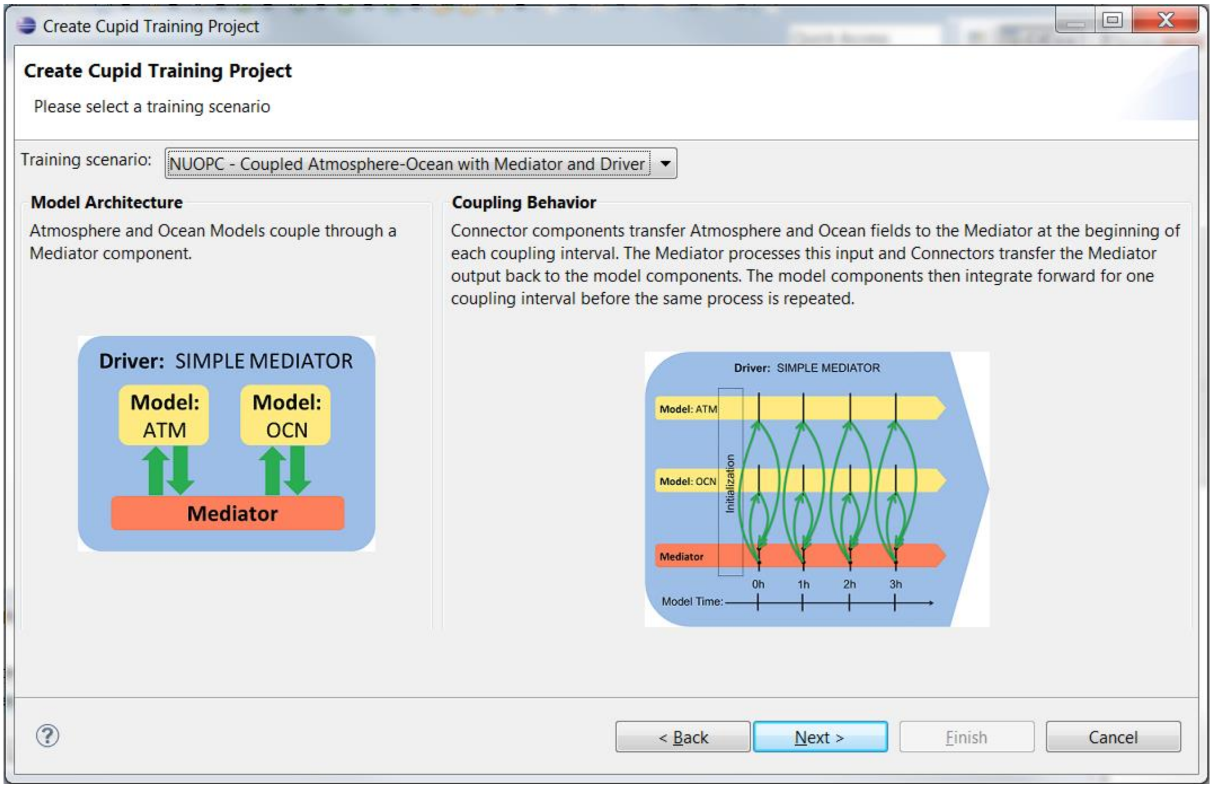
Training scenario: NUOPC - Single Model with Driver

Model Architecture

A single Model component is called by a Driver in regular intervals.

Coupling Behavior

There is no coupling in this configuration.



Create Cupid Training Project

Create Cupid Training Project

Please select a training scenario

Training scenario: NUOPC - Coupled Atmosphere-Ocean with Mediator and Driver

Model Architecture

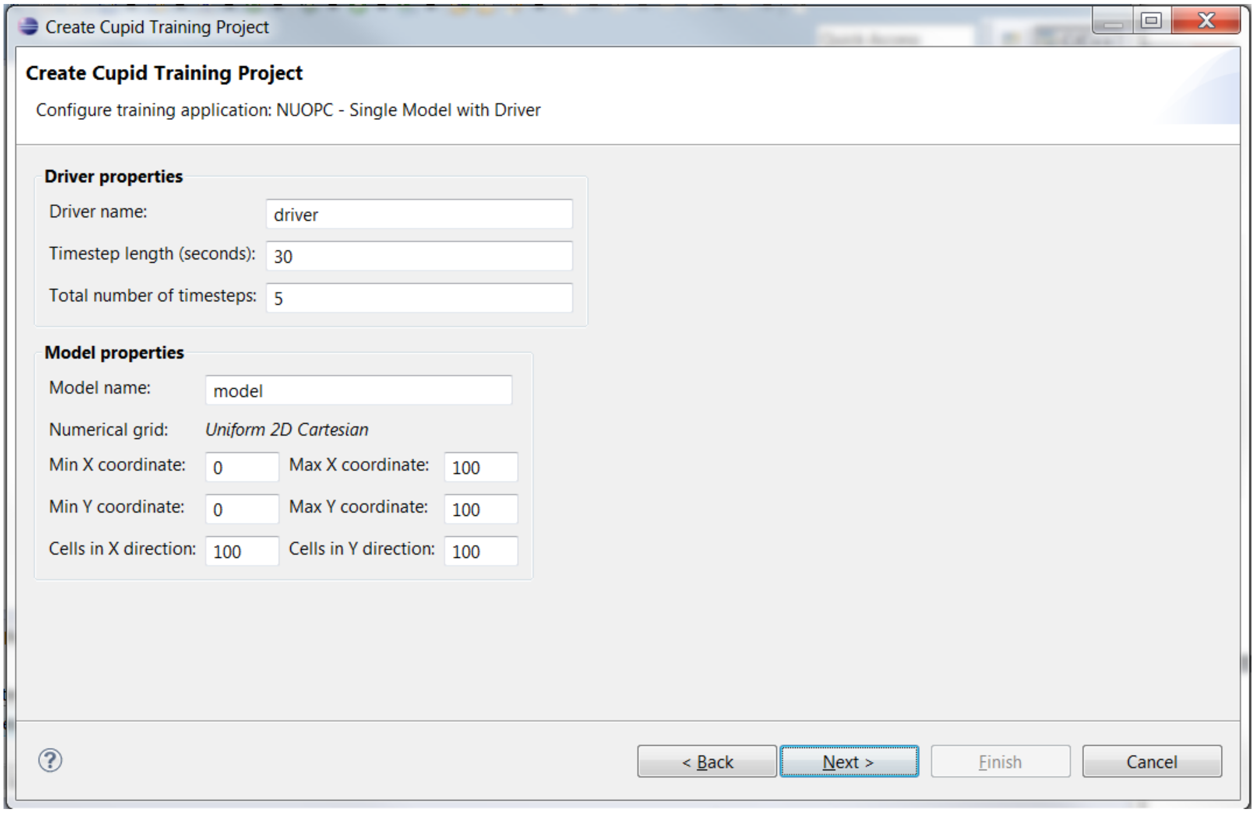
Atmosphere and Ocean Models couple through a Mediator component.

Coupling Behavior

Connector components transfer Atmosphere and Ocean fields to the Mediator at the beginning of each coupling interval. The Mediator processes this input and Connectors transfer the Mediator output back to the model components. The model components then integrate forward for one coupling interval before the same process is repeated.

3. On this page, choose a name for your project or accept the default name. Note that project names must be unique, so you must choose a name that does not already exist in your workspace. Click Next.

4. On this page you can optionally set a few parameters supported by the training scenario. It is okay to leave all the default values. Click Next.



Create Cupid Training Project

Configure training application: NUOPC - Single Model with Driver

Driver properties

Driver name: driver

Timestep length (seconds): 30

Total number of timesteps: 5

Model properties

Model name: model

Numerical grid: *Uniform 2D Cartesian*

Min X coordinate: 0 Max X coordinate: 100

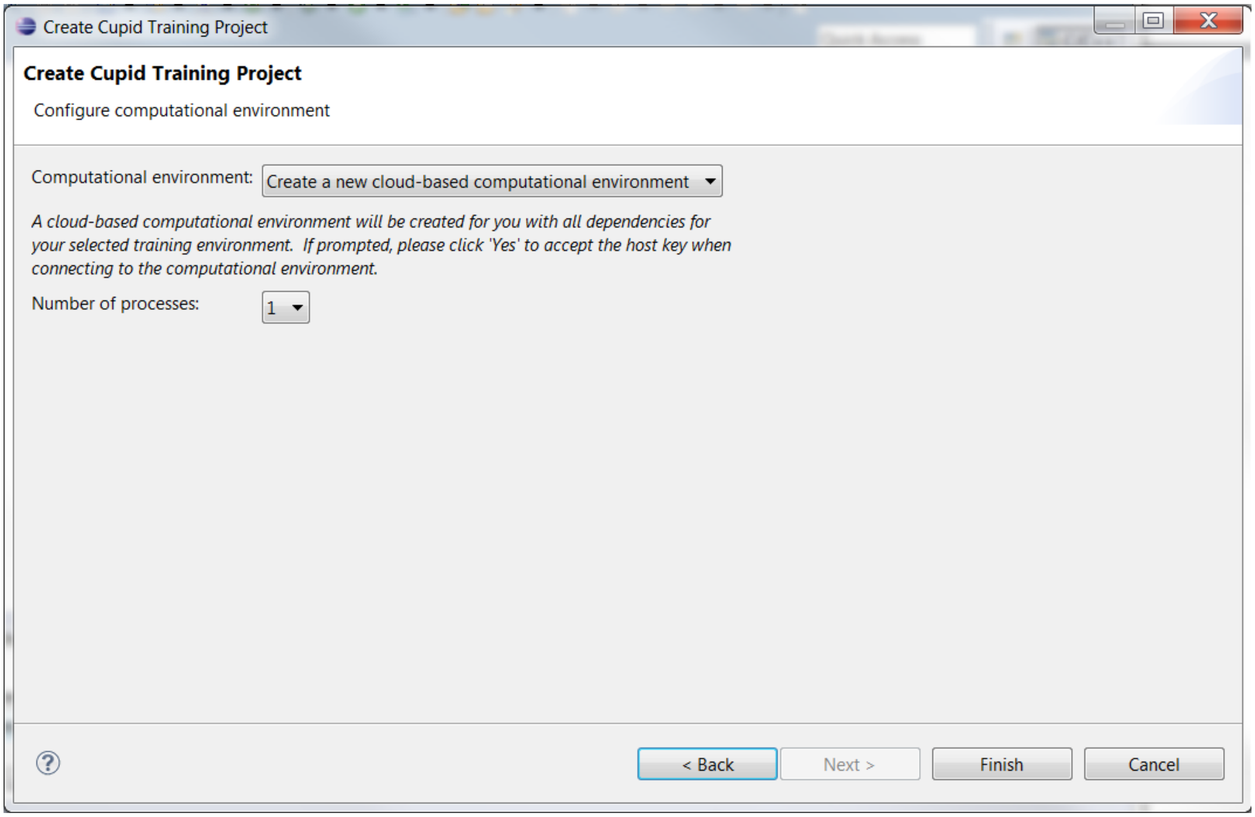
Min Y coordinate: 0 Max Y coordinate: 100

Cells in X direction: 100 Cells in Y direction: 100

5. On this page you will choose a computational environment in which to compile and run the training scenario. The two options are:

- Create a cloud-based computational environment
- Use my local machine

The first option will create a preconfigured computational environment for you with all dependent software (model source code, Fortran compiler, MPI, NetCDF, ESMF, etc.). This is the recommended option unless you know you have a supported environment already set up locally. Choose the first option and leave the number of processes at 1. Click Finish.


Create Cupid Training Project

Create Cupid Training Project

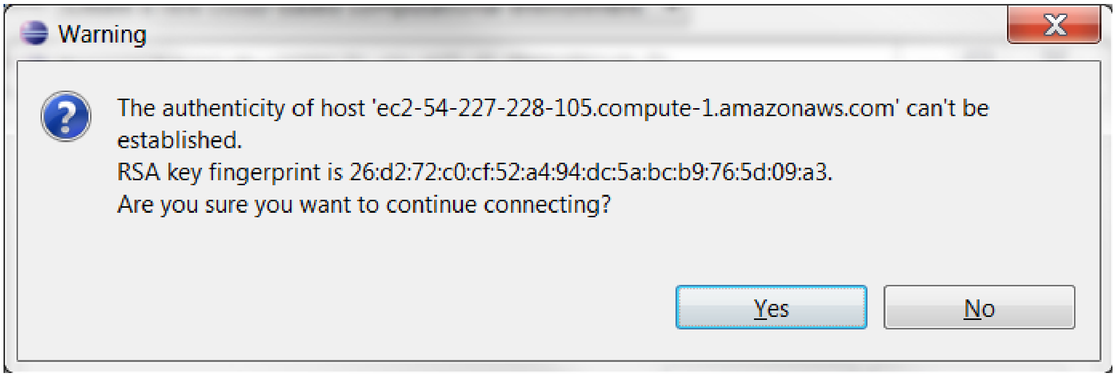
Configure computational environment

Computational environment: Create a new cloud-based computational environment

A cloud-based computational environment will be created for you with all dependencies for your selected training environment. If prompted, please click 'Yes' to accept the host key when connecting to the computational environment.

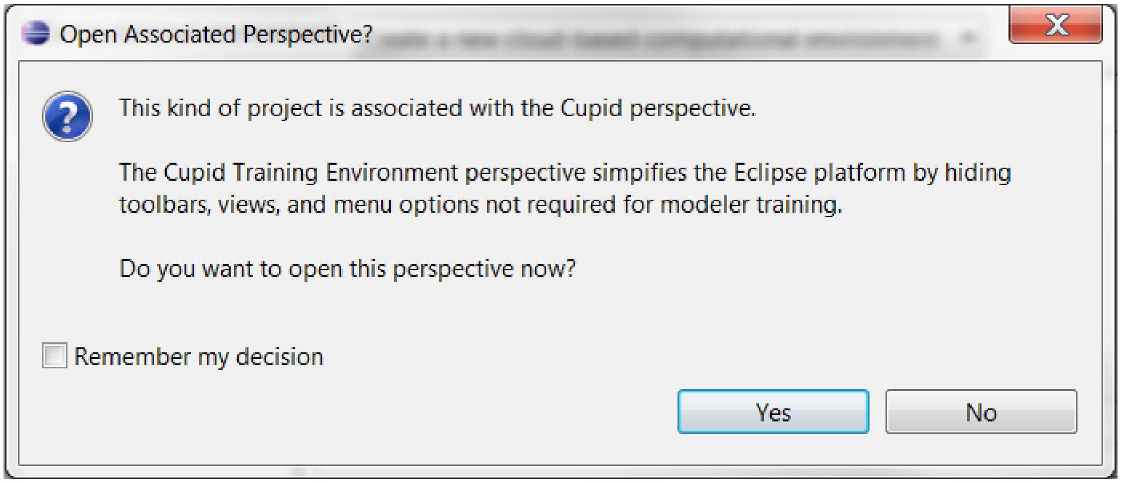
Number of processes: 1

6. It will take up to several minutes for the new computational environment to start up. During this process you will be asked to accept the host SSH key. Click Yes.


Warning

The authenticity of host 'ec2-54-227-228-105.compute-1.amazonaws.com' can't be established.
RSA key fingerprint is 26:d2:72:c0:cf:52:a4:94:dc:5a:bc:b9:76:5d:09:a3.
Are you sure you want to continue connecting?

You may also be asked if you would like to switch to the Cupid Perspective. An Eclipse perspective is a particular screen layout customized for specific tasks. The Cupid Perspective hides a number of Eclipse tools and commands that are not required for the training. Click Yes to switch to the Cupid Perspective.



Open Associated Perspective?

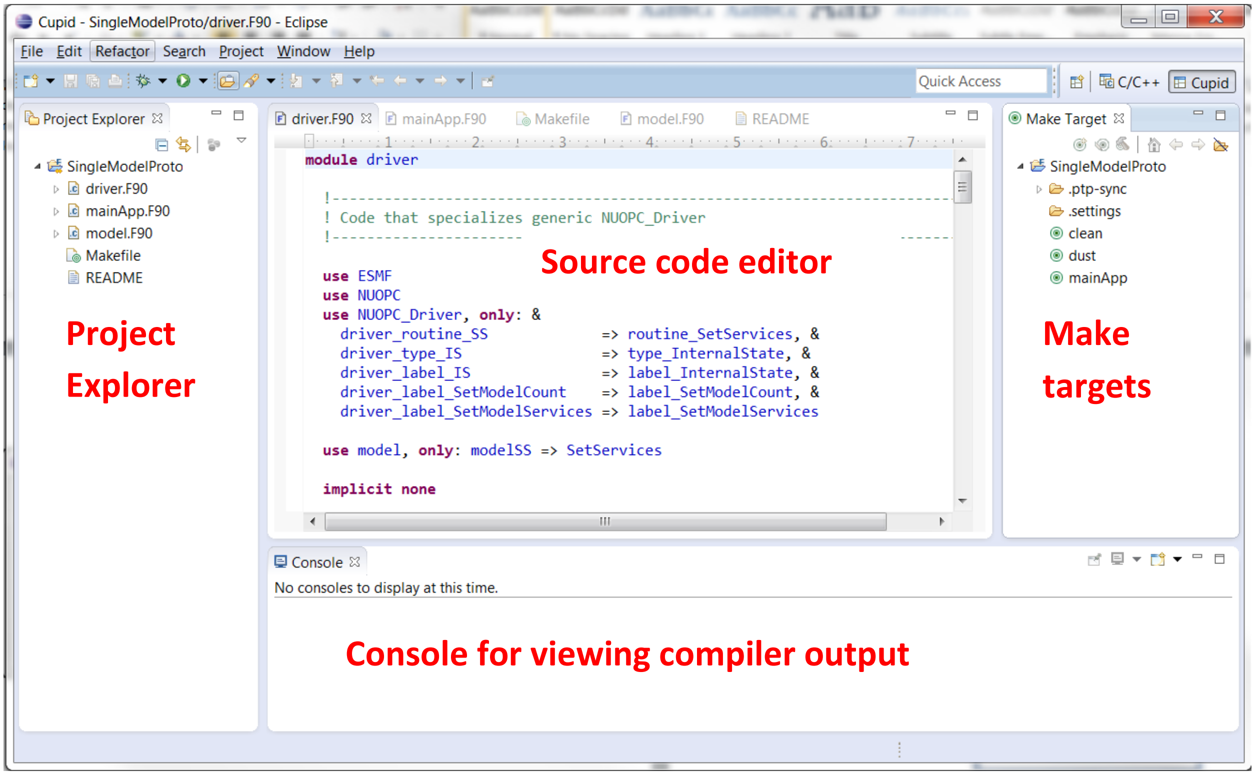
This kind of project is associated with the Cupid perspective.

The Cupid Training Environment perspective simpifies the Eclipse platform by hiding toolbars, views, and menu options not required for modeler training.

Do you want to open this perspective now?

□ Remember my decision

7. You should now see your project in the Project Explorer. The Cupid Perspective also exposes several other views including a Fortran source code editor, a Make Target view on the right for compiling your code and a Console at the bottom for viewing output from the compiler.



Cupid - SingleModelProto/driver.F90 - Eclipse

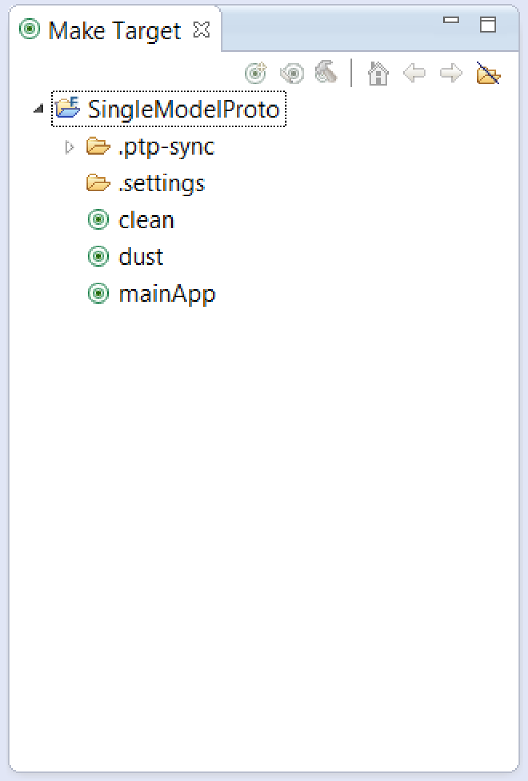
```
module driver

  !-----------------------------------------------------------------------------
  ! Code that specializes generic NUOPC_Driver
  !-----------------------------------------------------------------------------

  use ESMF
  use NUOPC
  use NUOPC_Driver, only: &
    driver_routine_SS            => routine_SetServices, &
    driver_type_IS               => type_InternalState, &
    driver_label_IS              => label_InternalState, &
    driver_label_SetModelCount   => label_SetModelCount, &
    driver_label_SetModelServices => label_SetModelServices

  use model, only: modelSS => SetServices

  implicit none
```

No consoles to display at this time.

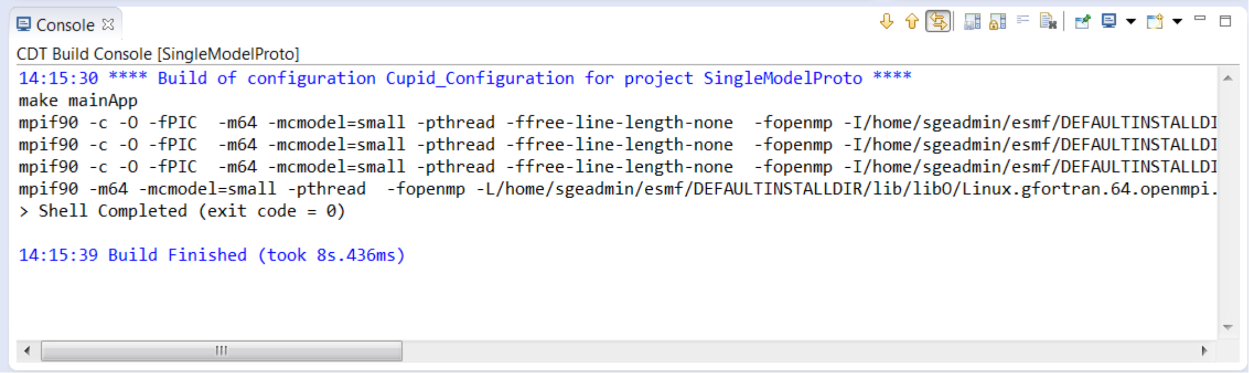
Compile and Run the NUOPC Single Model with Driver training scenario

In this section of the tutorial, you will learn how to compile and execute the source code provided in the NUOPC Single Model with Driver training scenario.   A Cheat Sheet is available for this task. (Click Help→Cheat Sheets and choose “Compile and Run a NUOPC Application” in the Cupid folder.)

1. Ensure that the Make Target view is showing (see below). If not, choose Window→Show View→Other… and select “Make Target” under the Make folder.



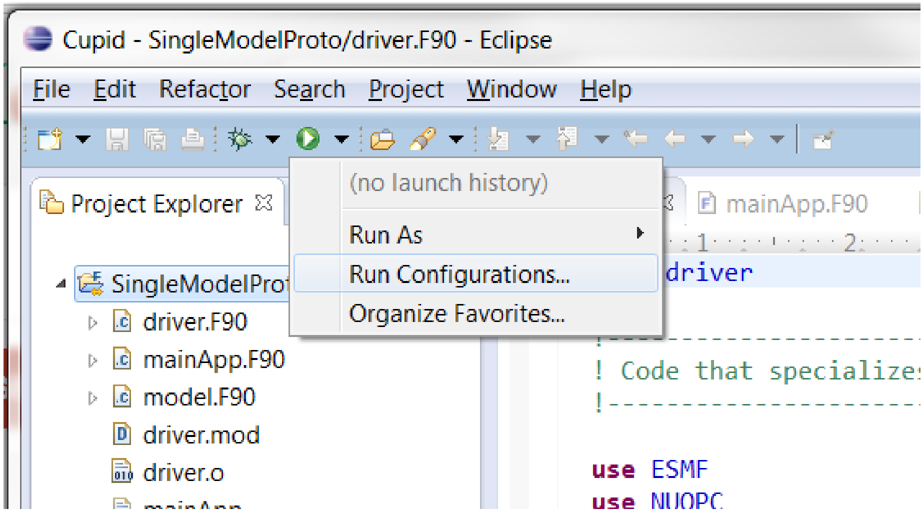
2. If you used the New Cupid Training Project wizard to set up the training scenario, then the correct make targets have already been set up. To compile the NUOPC application, double click the “mainApp” target. You should be able to see the compiler output in the Console.



CDT Build Console [SingleModelProto]

```
14:15:30 **** Build of configuration Cupid_Configuration for project SingleModelProto ****
make mainApp
mpif90 -c -O -fPIC  -m64 -mcmodel=small -pthread -ffree-line-length-none  -fopenmp -I/home/sgeadmin/esmf/DEFAULTINSTALLDI
mpif90 -c -O -fPIC  -m64 -mcmodel=small -pthread -ffree-line-length-none  -fopenmp -I/home/sgeadmin/esmf/DEFAULTINSTALLDI
mpif90 -c -O -fPIC  -m64 -mcmodel=small -pthread -ffree-line-length-none  -fopenmp -I/home/sgeadmin/esmf/DEFAULTINSTALLDI
mpif90 -m64 -mcmodel=small -pthread  -fopenmp -L/home/sgeadmin/esmf/DEFAULTINSTALLDIR/lib/libO/Linux.gfortran.64.openmpi.
> Shell Completed (exit code = 0)

14:15:39 Build Finished (took 8s.436ms)
```

3. To run the compiled code, you must set up a run configuration. *(Note: This step will be automated in a future release of Cupid.)* Click the down arrow next to the Run As... button in the toolbar (green circle with a while arrow) and choose Run Configurations… from the popup menu.

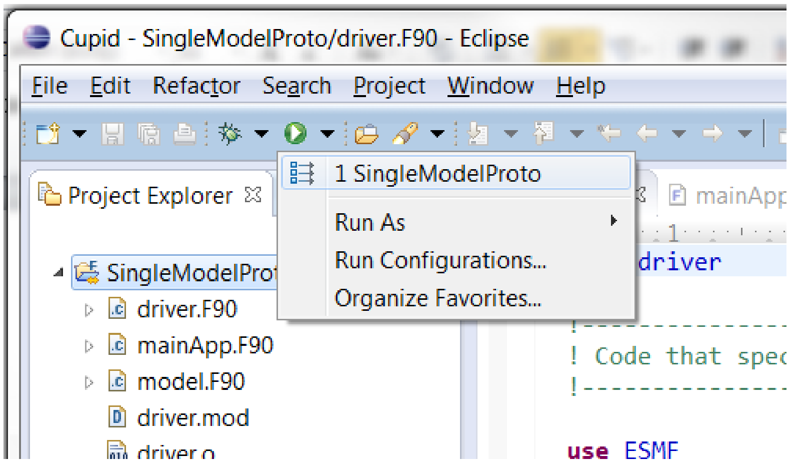


```
driver
!---------------------
! Code that specialize
!---------------------

use ESMF
use NUOPC
```

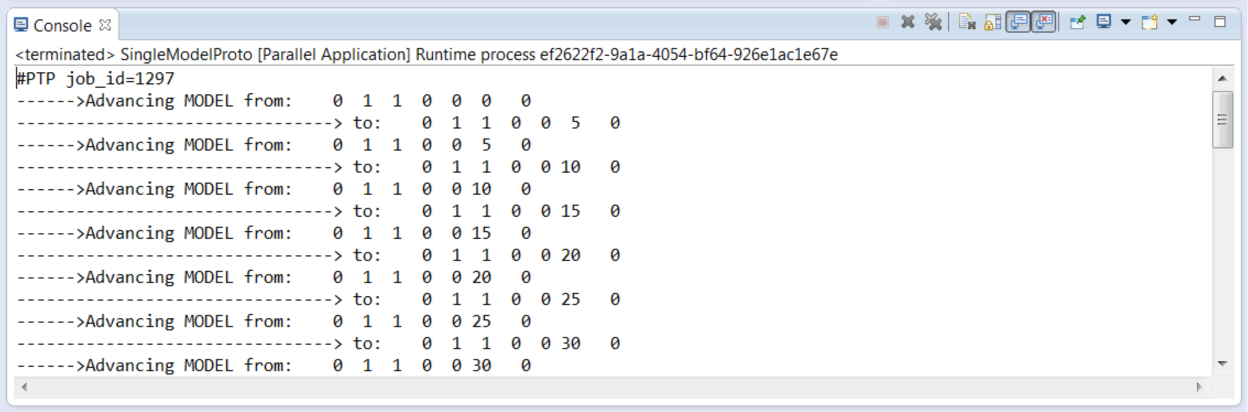
4. In the Run Configurations dialog, right click on Parallel Application and choose “New.” Configure the run configuration as follows:

- Resources tab
    - Target System Configuration: **Open MPI-Generic-Interactive**
    - Connection type: **Remote**
    - Connection: **Cupid Environment (Amazon EC2 <ip address>)**
- Application tab
    - Project: should default to your project name
    - Application program: Click Browse and select “mainApp”. The final path should be **/home/sgeadmin/SingleModelProto/mainApp**
- Click Apply then Run

5. Once the Run Configuration has been created, you do not need to set it up again unless you need to change some configuration settings. After running it the first time, the run configuration should be available in the Run Configurations dropdown list on the toolbar.



6. Output from the run will be shown in the Console view.



```
<terminated> SingleModelProto [Parallel Application] Runtime process ef2622f2-9a1a-4054-bf64-926e1ac1e67e
#PTP job_id=1297
------>Advancing MODEL from:    0  1  1  0  0  0    0
--------------------------------> to:     0  1  1  0  0  5    0
------>Advancing MODEL from:    0  1  1  0  0  5    0
--------------------------------> to:     0  1  1  0  0 10    0
------>Advancing MODEL from:    0  1  1  0  0 10    0
--------------------------------> to:     0  1  1  0  0 15    0
------>Advancing MODEL from:    0  1  1  0  0 15    0
--------------------------------> to:     0  1  1  0  0 20    0
------>Advancing MODEL from:    0  1  1  0  0 20    0
--------------------------------> to:     0  1  1  0  0 25    0
------>Advancing MODEL from:    0  1  1  0  0 25    0
--------------------------------> to:     0  1  1  0  0 30    0
------>Advancing MODEL from:    0  1  1  0  0 30    0
```