STARTING THE
DDT DEBUGGER
ON MIO, AUN, & MC2

(Mouse over to the left to see thumbnails of all of the slides)
Allinea DDT is a powerful, easy-to-use graphical debugger capable of debugging a wide variety of scenarios found in today’s development environments. With Allinea DDT, it is possible to debug:

- Single process and multithreaded software
- OpenMP
- Parallel (MPI) software
- Heterogeneous software such as that written to use GPUs
- Hybrid codes mixing paradigms such as MPI + OpenMP, or MPI + CUDA
- Multi-process software of any form, including client-server applications
PURPOSE:

- Show you how to start the ddt debugger, specifically on Mio
- The procedure is the same on AuN and Mc2
- There is a link to the full documentation at the end of these slides
ASSUMPTIONS

- You “home” machine supports X-Windows applications
  - Most desktop Linux machines support X out of the box
  - On Mac - use Quartz
  - On Windows - use Cygwin/X
- We have an example “buggy” application
- We will show how to launch ddt and run until we get to the bug
- We will run from a Mac
- There is a Mac and Windows client that does not require X but that requires an additional install, however, in the long run it is a better option
CONNECTING TO MIO

- ddt is an X-Windows application

- You must use the -Y option on your ssh line to allow X-Windows applications.

- If you are login in thru another machine you must use the -Y option there also

ssh -Y mio.mines.edu
Reset your environment by doing a `module purge` and loading the modules you will need for the example.

- The modules we use are:
  - PrgEnv/Debug/ddt
  - StdEnv

- The “StdEnv” module gives you a default compiler environment. PrgEnv/Debug/ddt gives you the debugger.

- Load the modules:
  ```
  module load PrgEnv/Debug/ddt
  module load StdEnv
  ```

- Create a new directory, “buggy”
  ```
  mkdir buggy
  ```

- Go there
  ```
  cd buggy
  ```

- The example files are in a tar file. Copy the example files to the directory by using the command
  ```
  tar -xf /opt/utility/quickstart/debug.tar
  ```
BUILD AND STARTING DDT

- Build the example applications by entering `make`
- **Note:** You must use the `-g` option when you build a program to be able to debug it
- You can ignore the warning
- The "make" also copies the script file `batch.qtf` to your local directory
- Launch `ddt` on the command line

```
[tkaiser@mio001 buggy]$ make
mpif90 -fopenmp -g hybrid.f90 -o hybrid
Ifort: command line remark #10010: option '-pthread' is deprecated and will be
removed in a future release. See '-help deprecated'
mpicc -fopenmp -g hoasctimef.c -o hoasctimef
cc /opt/ddt/forge/6.1/templates/batch.qtf.
[tkaiser@mio001 buggy]$ ddt
```

**Note:** ddt creates batch scripts on the fly. The file `batch.qtf` is the template it uses to create the script. The templates are slightly different on each machine.
ACCOUNTS STRING

On AuN and Mc2 you need to an account number in your batch.qtf file.

The line in batch.qtf file that has the account number is of the form

```bash
#SBATCH --account=123456789
```

You can get your valid account numbers by running the command

```
/opt/utility/accounts
```

The makefile provided with this example will automatically set the account number for you in batch.qtf.
THE DDT MAIN WINDOW…

- This is the main ddt window
- In the ddt main window
  - Select File then Options
  - An Options window will come up
  - Select “System” and set MPI/UPC Implementation to “SLURM (generic)”
    Note: For Mc2 select Auto-Detect (BlueGene/Q (SLURM))
THE DDT MAIN WINDOW...

- This is the main ddt window
- In the ddt main window
  - Select File then Options
  - An Options window will come up
  - Select Job Submission
Click on the “folder” icon and navigate to your “buggy” folder

Select `batch.qtf`

Hit OK

You will be back at the starting window

Note: ddt creates batch scripts on the fly. The file `batch.qtf` is the template it uses to create the script. The templates are slightly different on each machine.
In the starting window

Click on Run

This will bring up a window in which you can select the program you will run and the number of MPI tasks
APPLICATION AND MPI SETTINGS

- Again select the folder icon for Application and navigate to your “buggy” folder and select `phostname`.
- Also set the Working Directory to “buggy”.
- Click next to MPI and then Processes per Node.
- Enter 4 in each of the two boxes as shown. This will allow a run on 1 node with 4 MPI tasks.
- Make sure “Submit to Queue” is checked.
- Click Submit.

Note: On AuN and Mc2 you need to add an account number as the `srun` argument. The format is `-A #######`.

You can get your valid account numbers by running the command `/opt/utility/accounts`.
JOB SUBMISSIONS SETTINGS

- You job will go into the queue as normal to wait to start. Then...
- You will get a window that looks something like this with your source code in the middle.
- Click on the Go arrow to start your program
JOB SUBMISSIONS SETTINGS

- After a few seconds you will see the Program Stopped window indicating that there was a problem.
- Note the source window shows the line number where the problem occurred.

Here is the problem:
USING THE REMOTE CLIENTS

- There are standalone “front ends” for ddt for Mac-OS X and Windows
- The interface is nearly the same
- Advantages:
  - Don’t require X-windows install on your desktop
  - Much faster!
- Disadvantages:
  - Have to install the client
  - Connection to AuN and Mc2 requires a minor “strange” setting
GETTING THE CLIENTS


- Select your version from under “Remote client for OS/X, Windows and Linux”

- Install as normal

- Note: The version number of the client must match the version of ddt installed on our systems. If it does not you can get older versions or let the HPC group know to do an update.
SETTING UP THE CLIENT

- When you start the client you will get a window like this.
- You need to set up “Remote Launch” to point to our machines.
- Select “Configure” then “Add” to create configurations for the machines on which you want to run.
**SETTINGS FOR: MIO**

- The Host Name: *mio*
- Assuming you are using version 6.1 the Remote Install is at /opt/ddt/forge/6.1
- To make your life easier you should have password less ssh set up to get to Mio
The settings for AUN are:

- **Host Name:** `bluem aun`
- **Remote Installation Directory:** `/opt/ddt/forge/6.1`
- **Remote Script:** Optional
- **Always look for source files locally:** No

(Note you need both. Click on the How do I... link for more information)

Assuming you are using version 6.1 the Remote Install is at `/opt/ddt/forge/6.1`

To make your life easier you should have password less ssh set up to get to bluem.
• The Host Name: **bluem mc2**
• (Note you need both. Click on the How do I… link for more information)
• Assuming you are using version 6.1 the Remote Install is at `/opt/ddt/forge/6.1`
• To make your life easier you should have password less ssh set up to get to bluem
TO LAUNCH...

- Select one of your configurations under Remote Launch
- Select “Run”
TO LAUNCH...

- Selecting “Run” will bring up this window
- Hitting the “Options” button will allow you to set “System” and “Job Submission” values as discussed above.
- When all is set, hit Submit
MORE INFORMATION

- A movie of the ddt start-up process
  - http://geco.mines.edu/prototype/ddt/movie.html

- PDF of these slides

- User guide