

# Running DDT, MAP and PR jobs on Cheyenne

The Arm Forge tools, **DDT** and **MAP**, are installed on Cheyenne for debugging, profiling, and optimizing code.

Follow the procedures below to configure the DDT and MAP client interface on your local machine and then start your debugging and profiling jobs. We recommend using the client software to get the best performance, but the tools also run from the Cheyenne command line interface.

**Performance Reports** is another Arm tool for Cheyenne users. It summarizes the performance of HPC application runs. Get details and a sample batch script for generating reports below.

## Note

The default configuration of MPT on Cheyenne can lead to launch timeouts when using these Arm Forge tools, including Performance Reports. CISL recommends setting the environment variables **MPI\_SHEPHERD=true** and **MPI\_INIT\_LATE=false** in any job that uses them with the MPT MPI library.

Your job's memory footprint increases when you use these tools, especially when you use MAP. If your job normally uses most of the memory on a node, you might need to distribute your cores across more nodes to avoid running out of memory.

Arm Forge was formerly known as Allinea Forge, and "allinea" remains in use in some cases.

## Page contents

- [Preparing your code](#)
- [Client interface setup](#)
- [Running DDT and MAP](#)
- [Performance Reports](#)

---

## Preparing your code

Include the **-g** option when you compile your code for debugging with DDT or profiling with MAP. See our [Compiling code documentation](#) for the compilation commands to use.

Do not move or remove the source code, binary, or executable files from the directory or directories in which you compiled them.

Review the following Cheyenne-specific documentation and the [Arm Forge User Guide](#).

---

## Client interface setup

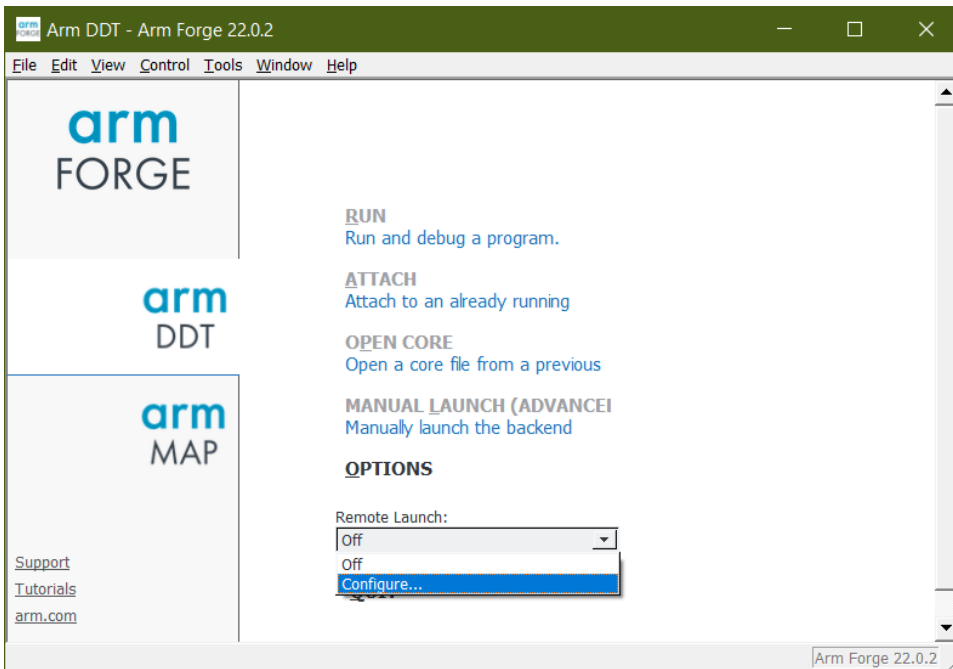
The client software version that you use locally and the server version that you use on Cheyenne must be the same. We recommend using the latest version that is available on Cheyenne. Run **module av arm-forge** to identify the latest version.

### Procedure

Download the client software from the [Arm site](#).

Install and start the client on your local machine.

From the "Remote Launch" menu (see image), select **Configure**.



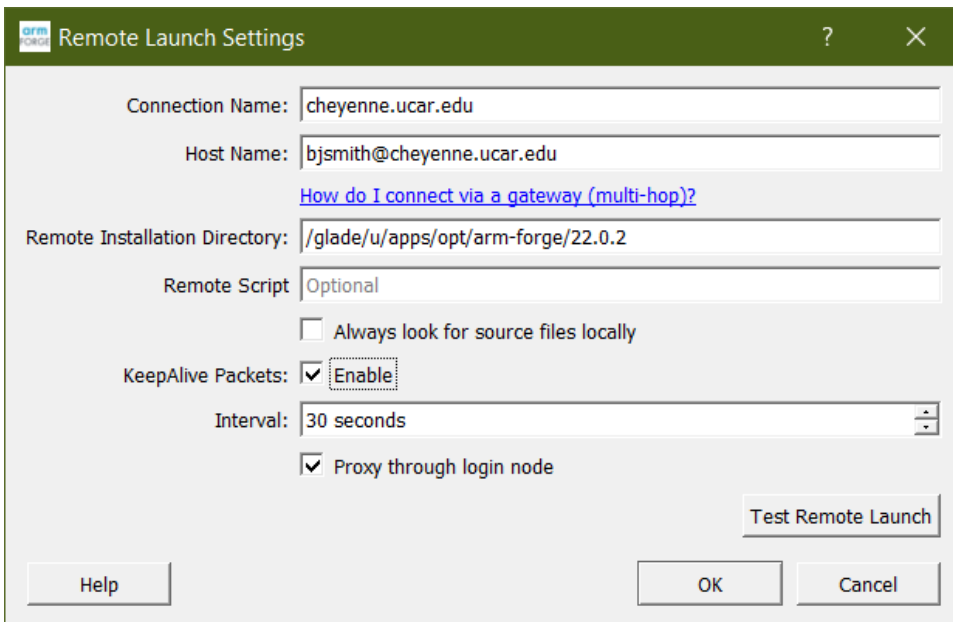
Configure as shown in the following image. The configuration will apply to both DDT and MAP, so you only need to do it once.

Enter your **username** followed by @ and the connection name (**cheyenne.ucar.edu**) in the "Host Name" field.

Then, fill in the "Remote Installation Directory" field. You can copy the text from here and **change the version number to match the version you are using**:

```
/glade/u/apps/opt/arm-forge/22.0.2
```

Leave the "Remote Script" field blank.



Click **OK**.

## Running DDT and MAP

Prepare a [job script](#). Specify the [regular queue](#) and customize the script with your own project code, job name, and so on.

On the last line of your script, replace `mpiexec_mpt` with `ddt --connect` (or `map --connect`).

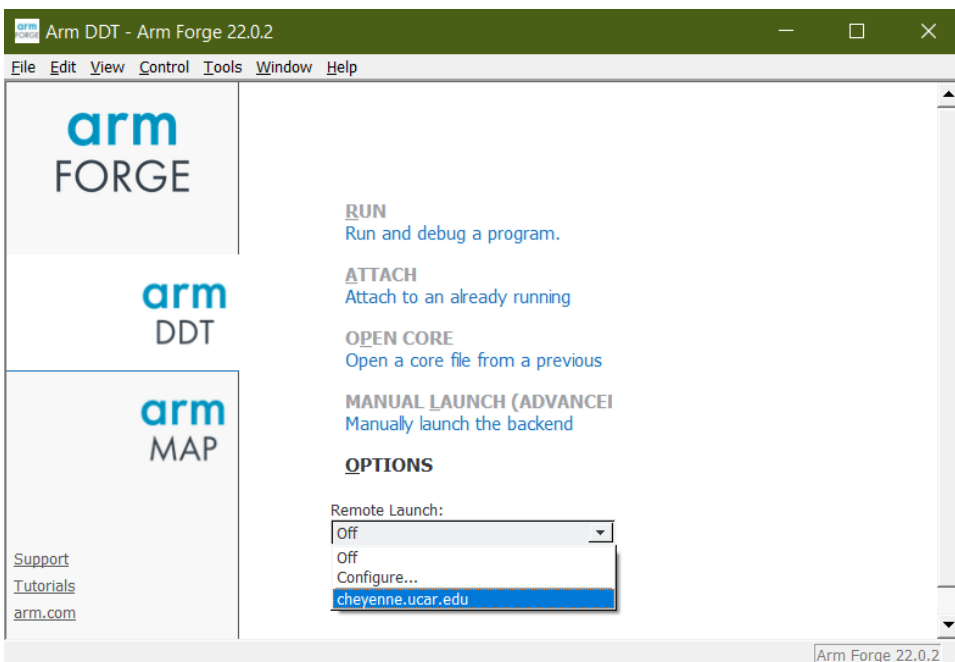
```
ddt --connect ./myjob_f
```

Submit your job when indicated below.

## Procedure

Start the client interface on your local machine.

From the “Remote Launch” menu, select your personal host name.



When the following dialog box appears, authenticate as usual. (It may be necessary to click *Show Terminal* to see the authentication window.)



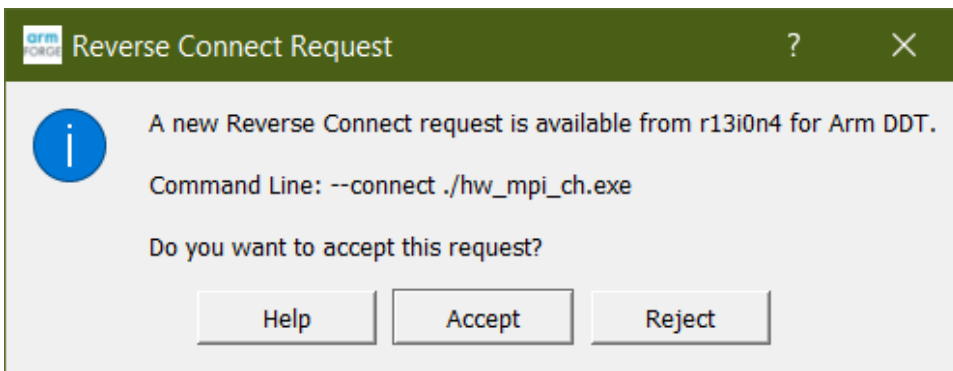
After logging in, return to your normal terminal window and load the modules that you need. (We recommend including the **module load** commands in your job script.)

```
module load arm-forge/22.0.2
```

Submit your job script on your command line.

```
qsub myscript.bash
```

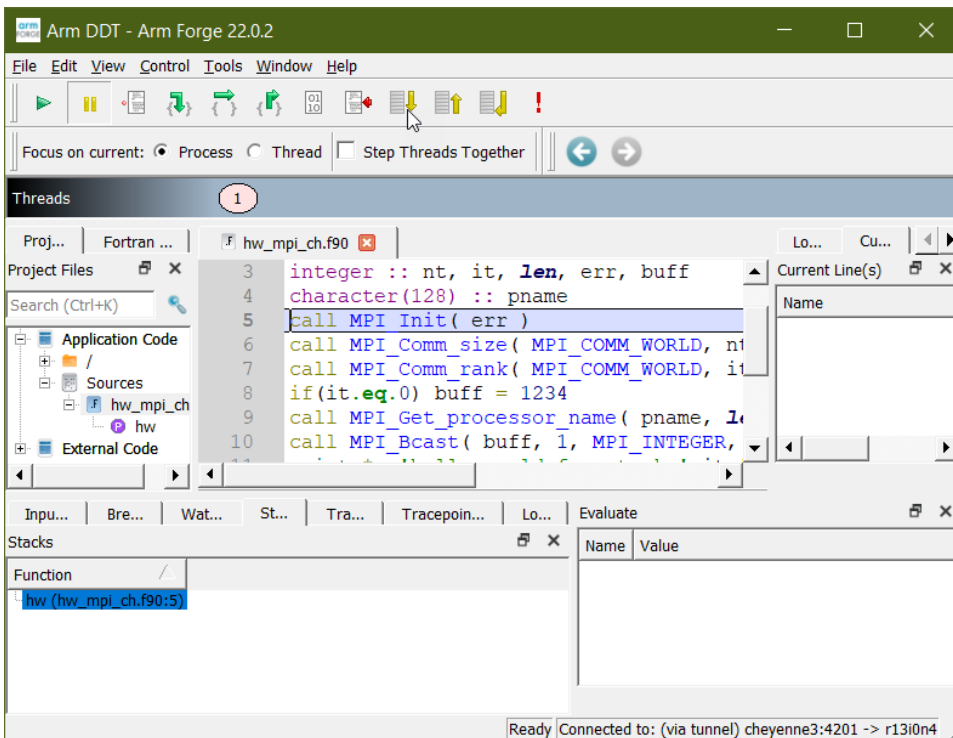
When your job starts, the Forge GUI will show that a “Reverse Connect Request” has been made. Accept the request to continue.



A “Run” window will open, displaying settings imported from your job script. Review the settings. If your program uses MPT, specify the MPI implementation as **HPE MPT 2.18+**. (See [Run window](#).)

Click **Run**.

The DDT (or MAP) window will open.



**Quit** when you're finished so the license is available to other users.

## Performance Reports

To generate a report on the performance of your program, submit a batch job to run it. You do not need to compile it with the **-g** debug option first.

Modify your batch script to load the **arm-forge** module that you want to use and include **perf-report** as shown in the sample scripts below.

When your job runs, the output will include both text and HTML report files.

See the [Arm product documentation](#) for additional information.

### Sample bash script

```
#!/bin/bash
#PBS -N prjob
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q regular
#PBS -j oe
#PBS -k eod
#PBS -l select=2:ncpus=36:mpiprocs=36
#PBS -m abe
#PBS -M email_address

module load arm-forge/22.0.2
export MPI_SHEPHERD=true
export MPI_INIT_LATE=false

export TMPDIR=/glade/scratch/$USER/temp
mkdir -p $TMPDIR

### Run the executable
perf-report --mpi -n 72 ./executable_name.exe
```

### Sample tcsh script

```
#!/bin/tcsh
#PBS -N prjob
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q regular
#PBS -j oe
#PBS -k eod
#PBS -l select=2:ncpus=36:mpiprocs=36
#PBS -m abe
#PBS -M email_address

module load arm-forge/22.0.2
setenv MPI_SHEPHERD true
setenv MPI_INIT_LATE false

setenv TMPDIR /glade/scratch/$USER/temp
mkdir -p $TMPDIR

### Run the executable
perf-report --mpi -n 72 ./executable_name.exe
```