

Erlangen Regional
Computing Center



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

Arm DDT

HPC Café 14.07.2020

Georg Hager

HPC Division, RRZE Erlangen



- DDT is part of **Arm Forge**, which also contains MAP, a performance profiler
- DDT can debug
 - Serial programs
 - **OpenMP**-parallel programs
 - **MPI**-parallel SPMD and MPMD programs
- Usual feature set
 - Single stepping, into, out of, over functions, inspect variables, watchpoints, tracepoints, etc.
 - C/C++/Fortran support
 - Some nice integration of parallel features

Arm DDT interface

Thread/process control

File and function navigation

Stack trace/watchpoints/etc.

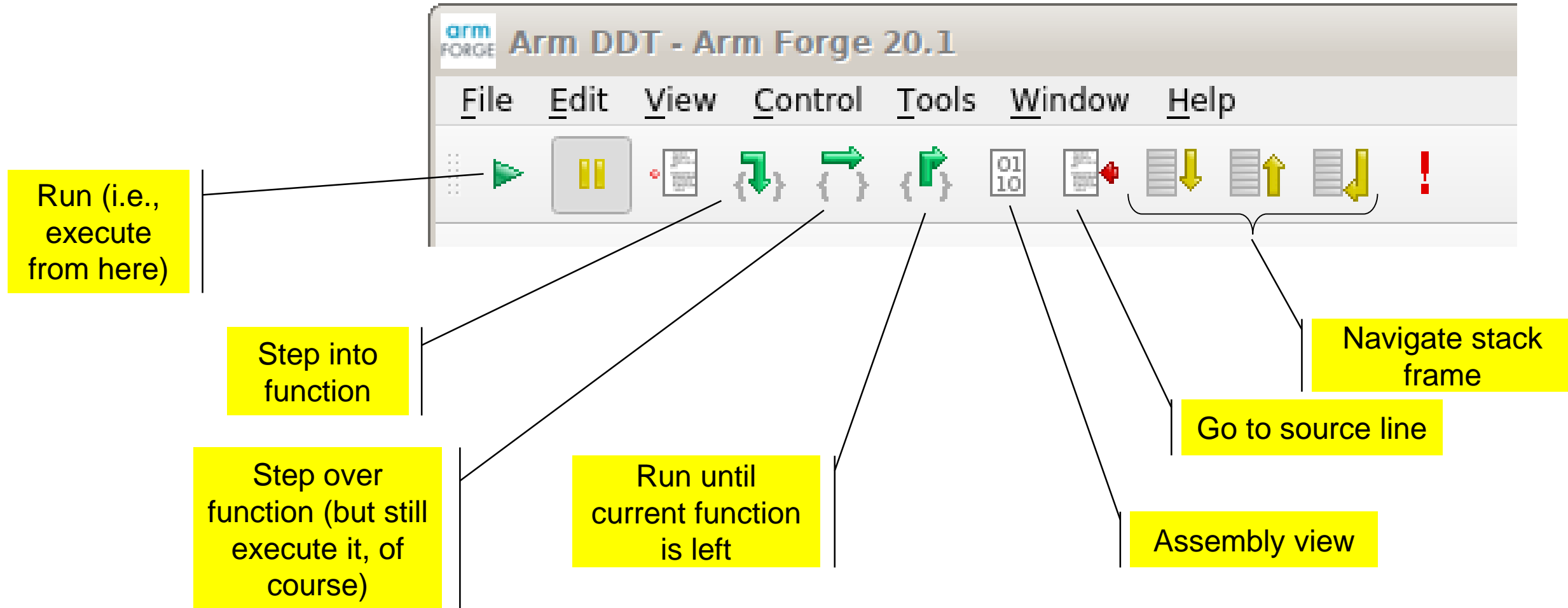
Variable and array inspection

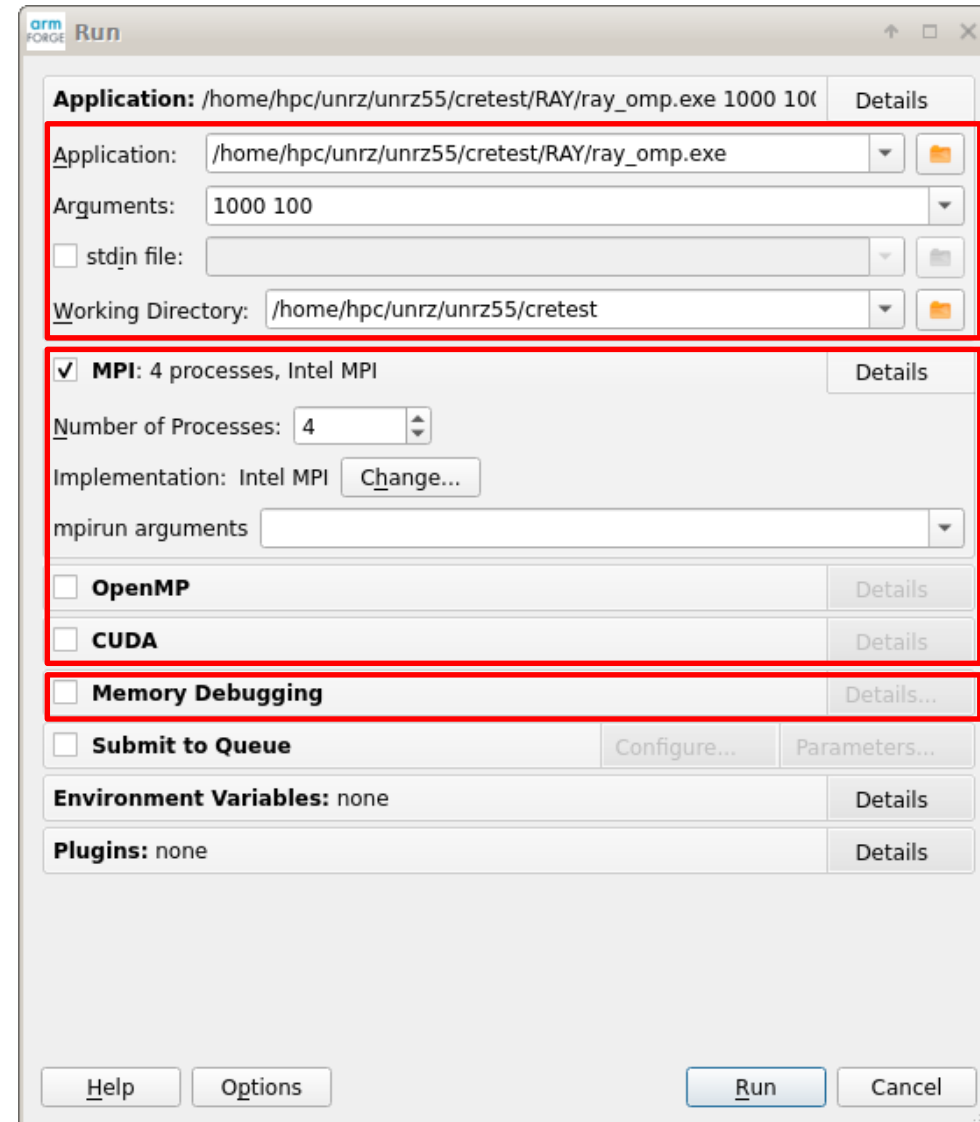
Code window

Evaluation of expressions

The screenshot displays the Arm DDT interface with the following components:

- Threads:** A control bar at the top with buttons for running, pausing, and stepping through threads.
- Project Files:** A tree view on the left showing the project structure, including source files like `ray_omp.c`.
- Code Window:** The central editor showing C code with OpenMP pragmas. The current line is highlighted.
- Locals:** A window on the right showing the current scope's variables and their values, such as `xc` (5), `i` (0), and `yc` (2).
- Stacks:** A window at the bottom showing the current call stack, including functions like `main`, `shade`, and `calc_tile`.
- Evaluate:** A window at the bottom right for evaluating expressions.





... or just

```
$ ddt -start -n 4 ./a.out arg1 arg2 arg3...
```

...or:

```
$ OMP_NUM_THREADS=4 ddt -start ./a.out arg1 arg2 arg3...
```

- **OpenMP debugging** is “difficult”
 - Weird behavior of private vs. shared variables
 - No stepping into and out of parallel regions
 - Inactive threads sit in OpenMP runtime functions
 - Some compiler optimizations may be applied despite -O0
- Always distinguish between “step all” and “step current thread/process”
- Use **tracepoints** and **watchpoints** for fun and profit