Profiling with TAU - Basics

Dr Chennakesava Kadapa

Swansea Academy of Advanced Computing (SA2C), Swansea University, UK.

Email: c.kadapa@swansea.ac.uk

General workflow

- Preparation
- Measurement
- Analysis
- Examination
- Optimisation

Performance analysis steps

- Program instrumentation
- Summary measurement collection
- Summary analysis report examination

TAU + MPI

- **Some of the features (unwinding) are not supported by Intel MPI**.
- PAPI counters do not work with Intel MPI.
- The MPI library used with TAU is OpenMPI (3.1.1).

Modules on Sunbird

- To use TAU, we need to load the appropriate modules first.
- TAU is available for use with Intel and GNU compilers on *Sunbird*.
- The default compiler is GNU 8.1.0
  - `module purge`
  - `module load tau/2.28.1`
- To use TAU with Intel/2018/2 compiler, we need to load the compiler first.
  - `module purge`
  - `module load compiler/gnu/8/1.0`
  - `module load tau/2.28.1`

Instrumentation methods

TAU provides a list of instrumentation methods. Among them, the following three are popular.

- Dynamic instrumentation
- Source instrumentation
- Selective instrumentation

Dynamic instrumentation

Dynamic instrumentation

- The most straightforward way of instrumentation.
- Does not require modifications to the binary. (Build the executable using `mpicxx`.)
- It makes use of statistical sampling to estimate the percentage of execution times taken by each function.
- Add `tau_exec` before the name of the binary.
  - `mpirun -n 10 ./a.out`
  - `mpirun -n 10 tau_exec ./a.out`
  - `mpirun -n 10 tau_exec -T mpi,papi -ompt ./a.out`
- Unfortunately, this method doesn't support profiling user-defined routines but only those from the MPI library.

Dynamic instrumentation - Example

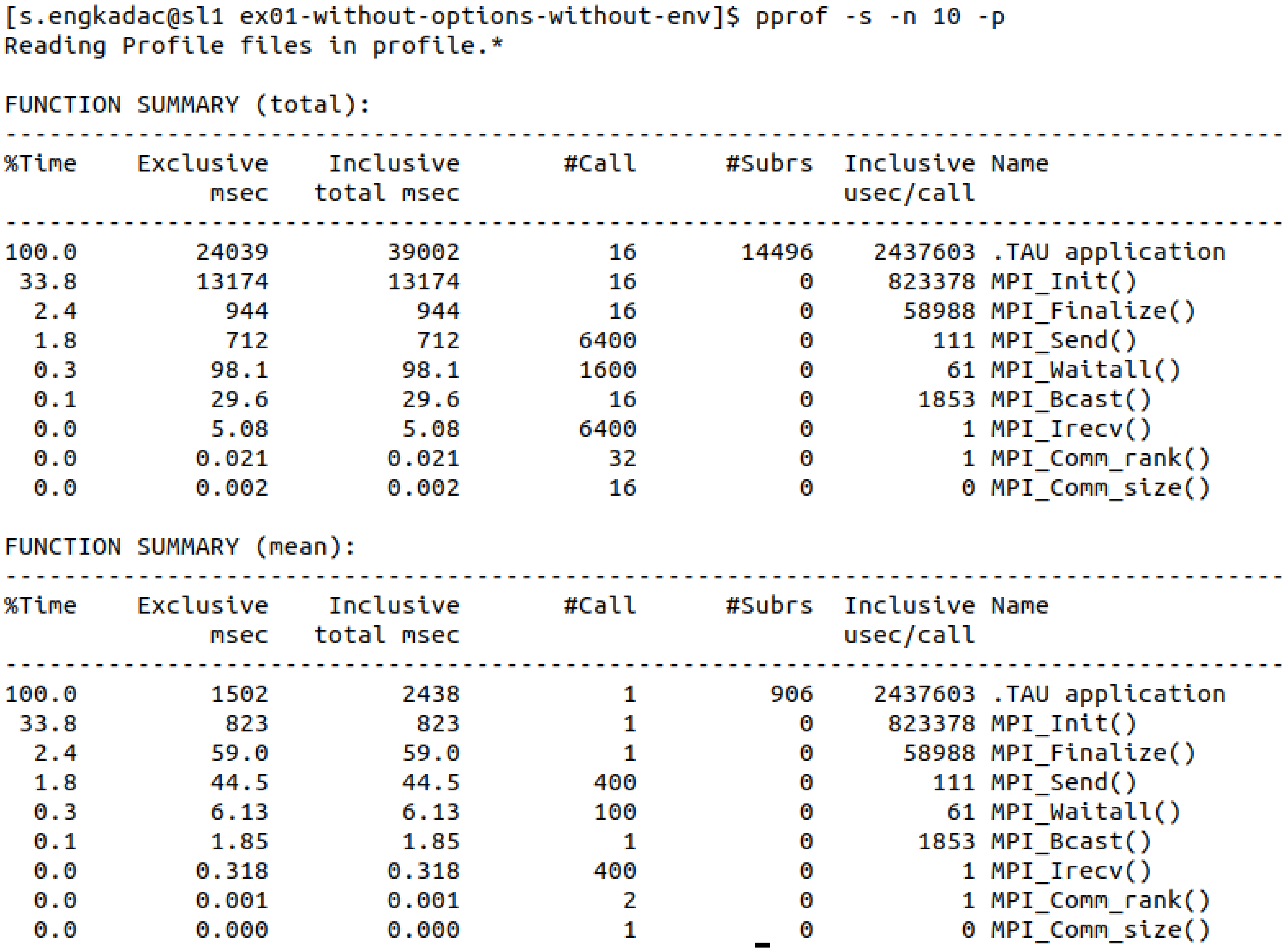
- Use the wave2d.cpp code from https://github.com/chennachaos/TAUhandson
- **Compile**: `mpicxx wave2d.cpp -o wave2d`
- **Run**: `mpirun -n 12 tau_exec ./wave2d 500 500 3 4 5`
- Once the run is successfully completed, we will get 12 files in `profile.a.b.c` format. `a` is the rank of processor, `b` is the context number and `c` is the thread number.

Visualisation of profiling data

- **pprof**: Text based visualisation.
    - Check with `pprof --help` for the list of options
- **paraprof**: GUI based visualisation.

Dynamic instrumentation - Example - Output with pprof

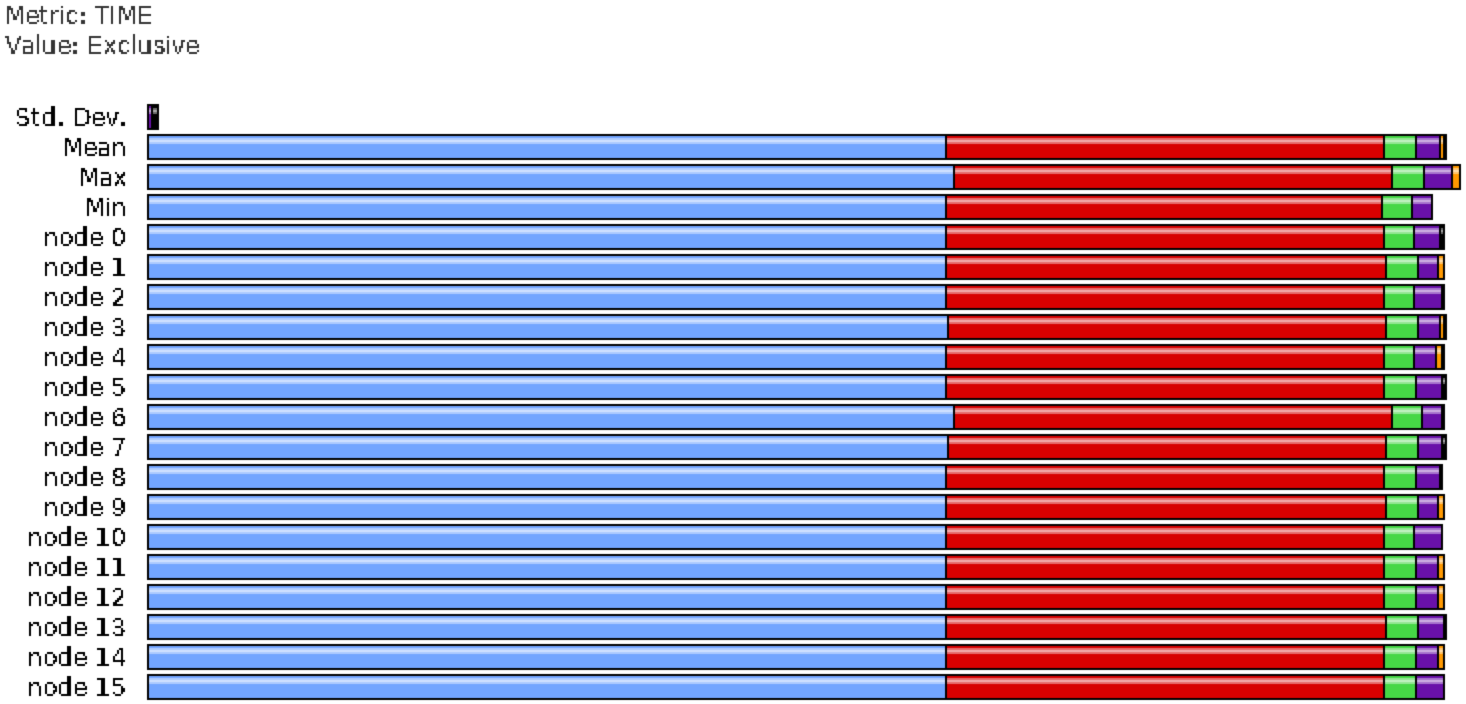
- Visualisation of profiling data using `pprof`.



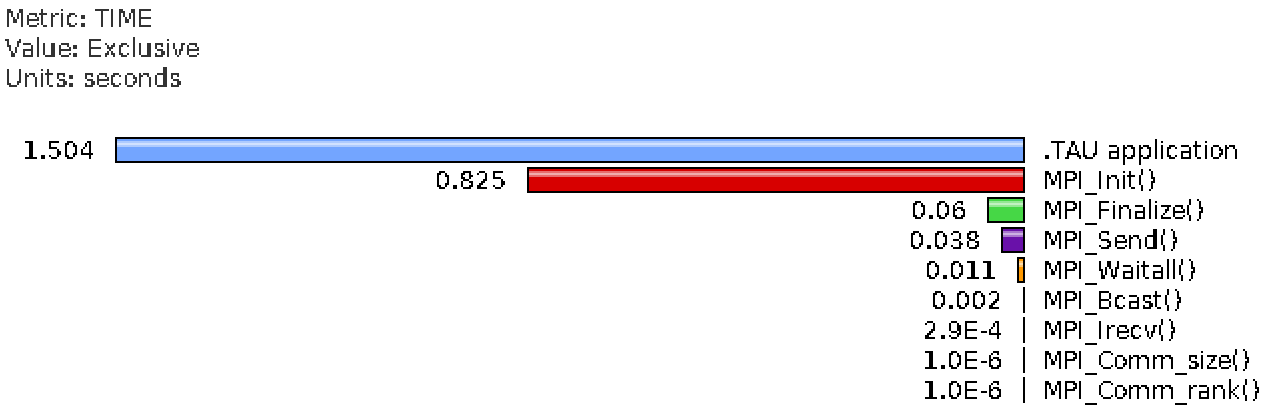
```
[s.engkadac@sl1 ex01-without-options-without-env]$ pprof -s -n 10 -p
Reading Profile files in profile.*

FUNCTION SUMMARY (total):
---------------------------------------------------------------------------------------
%Time    Exclusive    Inclusive       #Call      #Subrs  Inclusive Name
              msec   total msec                          usec/call
---------------------------------------------------------------------------------------
100.0        24039        39002          16       14496    2437603 .TAU application
 33.8        13174        13174          16           0     823378 MPI_Init()
  2.4          944          944          16           0      58988 MPI_Finalize()
  1.8          712          712        6400           0        111 MPI_Send()
  0.3         98.1         98.1        1600           0         61 MPI_Waitall()
  0.1         29.6         29.6          16           0       1853 MPI_Bcast()
  0.0         5.08         5.08        6400           0          1 MPI_Irecv()
  0.0        0.021        0.021          32           0          1 MPI_Comm_rank()
  0.0        0.002        0.002          16           0          0 MPI_Comm_size()

FUNCTION SUMMARY (mean):
---------------------------------------------------------------------------------------
%Time    Exclusive    Inclusive       #Call      #Subrs  Inclusive Name
              msec   total msec                          usec/call
---------------------------------------------------------------------------------------
100.0         1502         2438           1         906    2437603 .TAU application
 33.8          823          823           1           0     823378 MPI_Init()
  2.4         59.0         59.0           1           0      58988 MPI_Finalize()
  1.8         44.5         44.5         400           0        111 MPI_Send()
  0.3         6.13         6.13         100           0         61 MPI_Waitall()
  0.1         1.85         1.85           1           0       1853 MPI_Bcast()
  0.0        0.318        0.318         400           0          1 MPI_Irecv()
  0.0        0.001        0.001           2           0          1 MPI_Comm_rank()
  0.0        0.000        0.000           1           0          0 MPI_Comm_size()
```

Dynamic instrumentation - Example - Output with paraprof

- To visualise the profiling data, we prefer `paraprof`.
- Call `paraprof` from the command line. If you are connecting remotely, then you need to connect with X server activated; `ssh -X sunbird`.

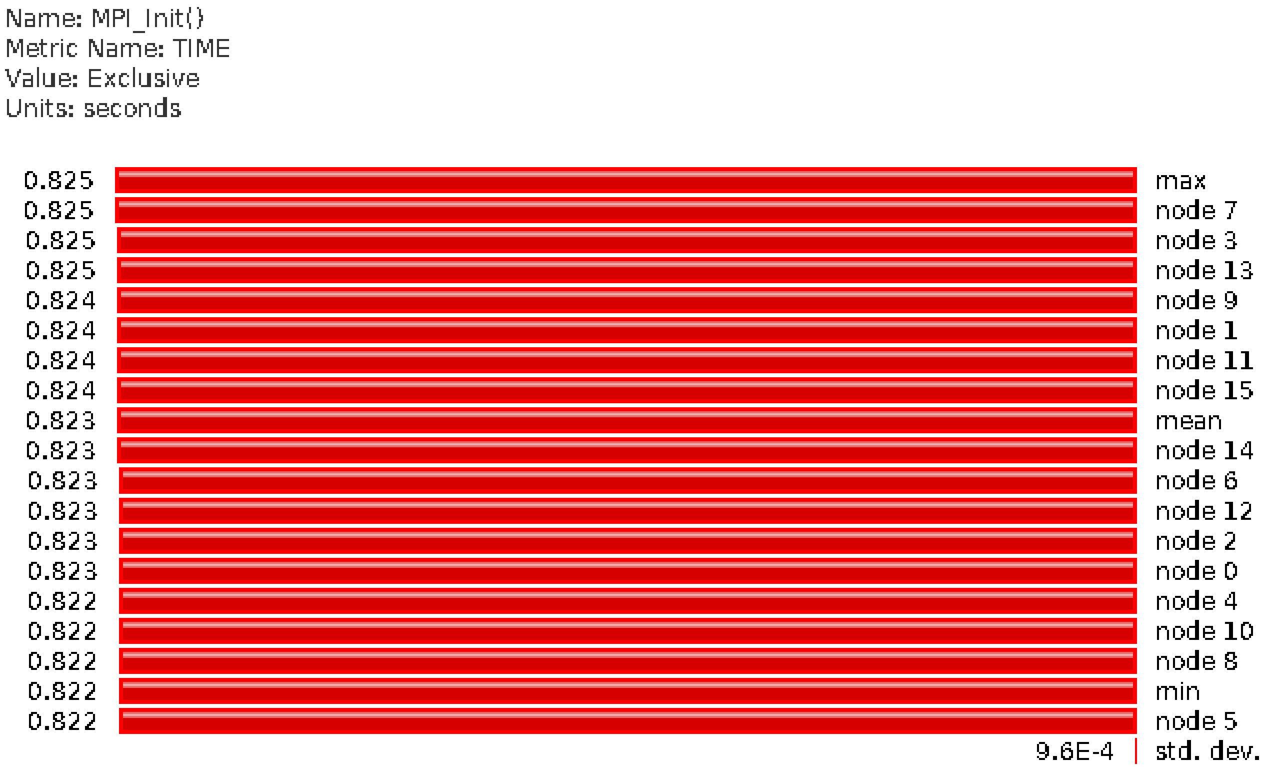


Dynamic instrumentation - Example - Output with paraprof



```
Metric: TIME
Value: Exclusive
Units: seconds

1.504     .TAU application
0.825     MPI_Init()
0.06      MPI_Finalize()
0.038     MPI_Send()
0.011     MPI_Waitall()
0.002     MPI_Bcast()
2.9E-4    MPI_Irecv()
1.0E-6    MPI_Comm_size()
1.0E-6    MPI_Comm_rank()
```

Dynamic instrumentation - Example - Output



Name: MPI_Init()
Metric Name: TIME
Value: Exclusive
Units: seconds

Compiler-based instrumentation

- Compiler instrumentation method sits in between *dynamic* and *source* instrumentation.
- It requires compilation of the source code with additional arguments.
- It cannot provide information about finer constructs such as loops.
- To use this method, we should add `-tau-options=-optCompInst` to `tau_cxx.sh`.
- There is not much we can understand about the performance of code using this method.
- Recommended practice is to follow the *source* instrumentation which is fine-grain.

TAU's environment variables

Variable	Description
TAU_PROFILE	Set to 1 to have TAU profile your code
TAU_TRACE	Set to 1 to have TAU trace your code
TAU_SAMPLING	Default value is zero (i.e., Sampling is OFF by default). When set to 1, TAU will collect additional profile or trace information via periodic sampling at runtime

TAU's environment variables (cont'd)

Variable	Description
TAU_CALLPATH	TAU will generate call-path data when this is set to 1
TAU_CALLPATH_DEPTH	Sets the depth of the callpath profiling
TAU_VERBOSE	When set TAU will print out information about its configuration when a running a instrumented application

More details at https://www.cs.uoregon.edu/research/tau/docs/newguide/bk05apa.html

Source instrumentation

- Replace `mpicc`, `mpicxx` and `mpif90` with `taucc`, `taucxx` and `tauf90`.
- Add `-dynamic` flag to the linking phase.
- Use files from `ex4`

Commonly encountered errors

- profile.x.x.x files are not generated.
- Run time error

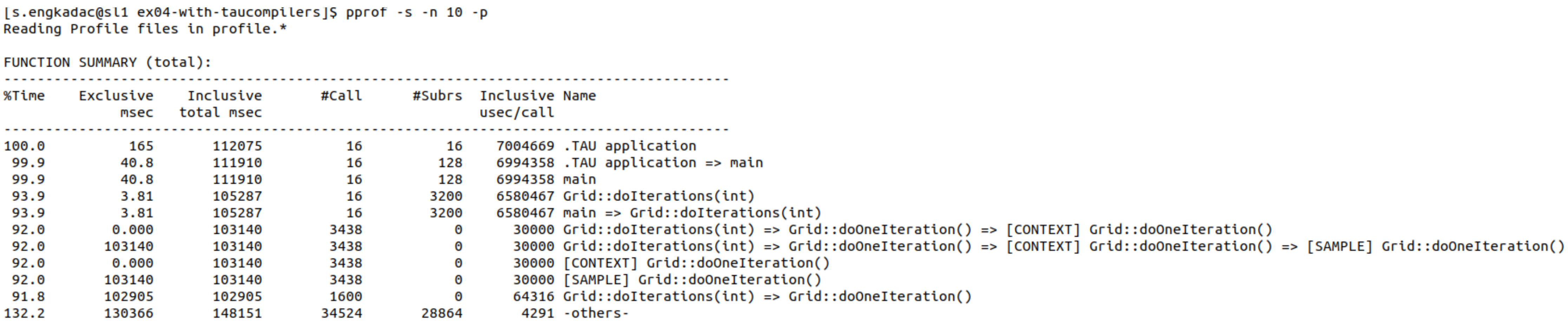
  ```
  wave2d: ../../scorep-5.0/./build-mpi/../src/measurement/paradigm/mpi/scorep_ipc_mpi.c:230: SCOREP_IpcGroup_GetRank: Assertion 'SCOREP_Status_IsMppInitialized()' failed.
  ```
- Turns out that the error is due to the incorrect value of `TAU_MAKEFILE`.

  ```
  TAU_MAKEFILE=/apps/tools/tau/2.28.1/el7/AVX512/gnu-8.1/openmpi-3.1/x86_64/lib/Makefile.tau-papi-mpi-pdt-scorep
  ```
- Need to be extra careful when setting the `TAU_MAKEFILE` variable

- Works correctly after using `TAU_MAKEFILE=/apps/tools/tau/2.28.1/el7/AVX512/gnu-8.1/openmpi-3.1/x86_64/lib/Makefile.tau-papi-mpi-pdt`
- `TAU_PROFILE`, `TAU_CALLPATH` and `TAU_SAMPLING` are set to 1.
- `-g` flag is not used.

Source instrumentation - Example - Output with pprof

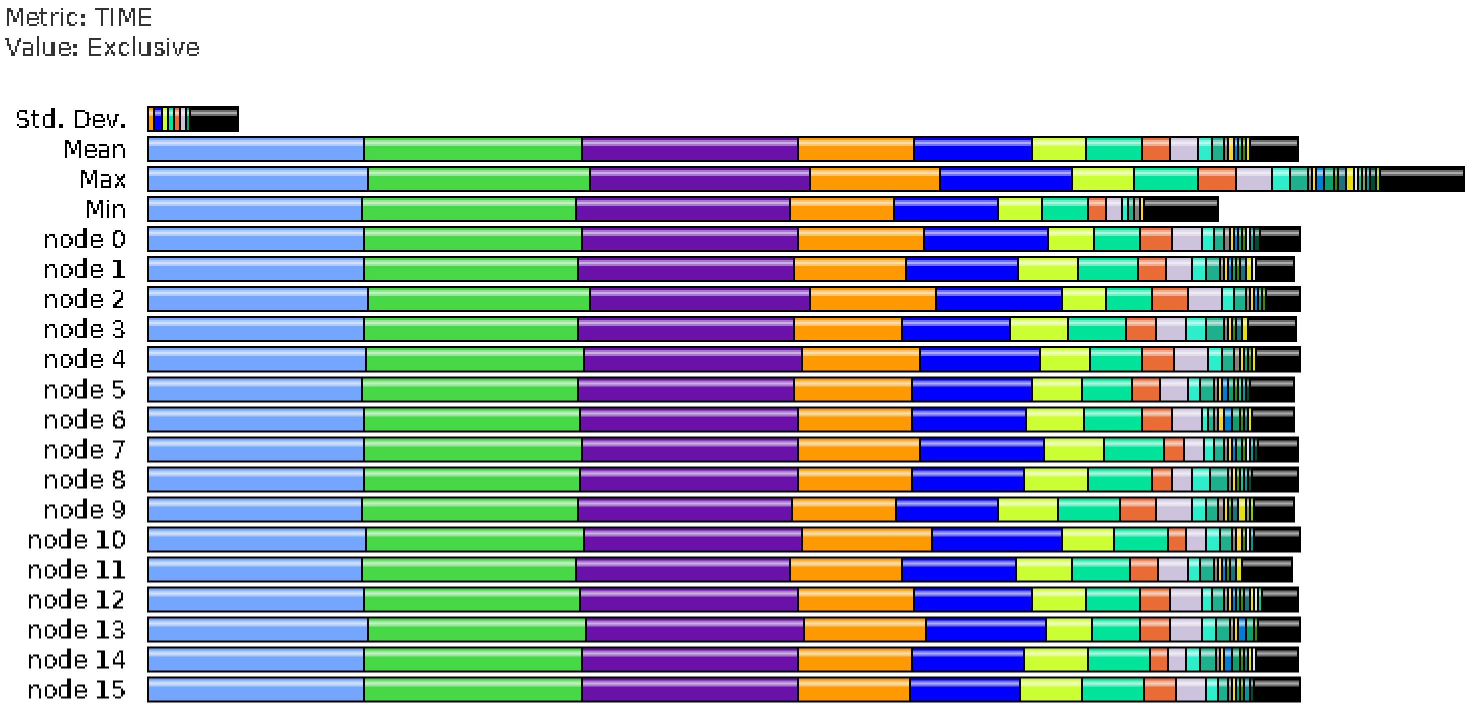
- Visualisation of profiling data using `pprof`.
- Notice the change in the output. We now see the user-defined functions in the profiling data.



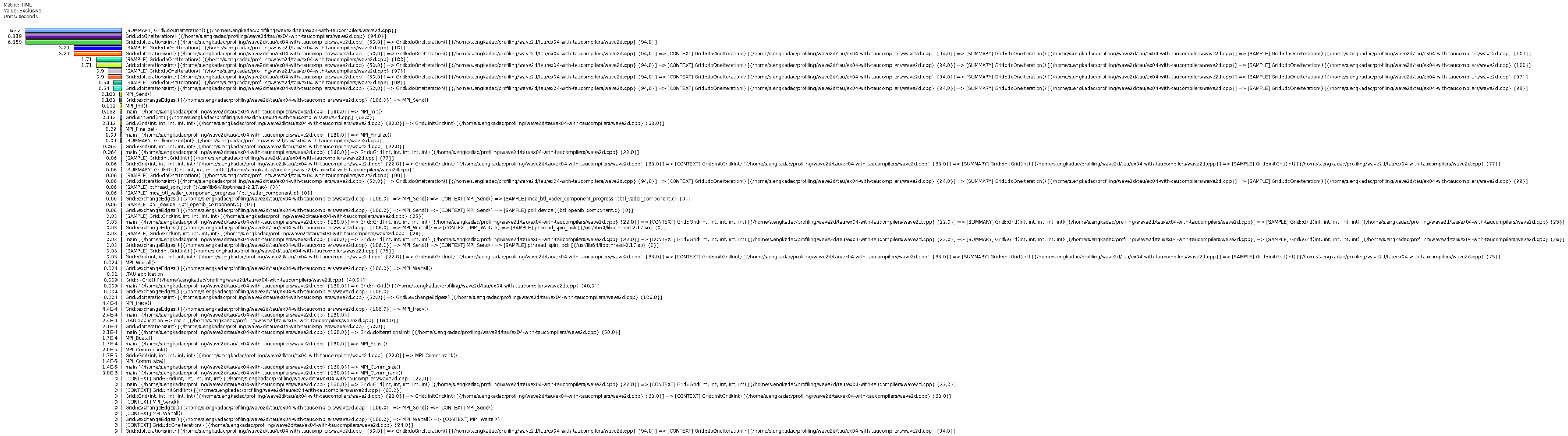
```
[s.engkadac@sl1 ex04-with-taucompilers]$ pprof -s -n 10 -p
Reading Profile files in profile.*

FUNCTION SUMMARY (total):
---------------------------------------------------------------------------------------
%Time    Exclusive    Inclusive       #Call      #Subrs  Inclusive Name
              msec   total msec                          usec/call
---------------------------------------------------------------------------------------
100.0          165       112075          16          16    7004669 .TAU application
 99.9         40.8       111910          16         128    6994358 .TAU application => main
 99.9         40.8       111910          16         128    6994358 main
 93.9         3.81       105287          16        3200    6580467 Grid::doIterations(int)
 93.9         3.81       105287          16        3200    6580467 main => Grid::doIterations(int)
 92.0        0.000       103140        3438           0      30000 Grid::doIterations(int) => Grid::doOneIteration() => [CONTEXT] Grid::doOneIteration()
 92.0       103140       103140        3438           0      30000 Grid::doIterations(int) => Grid::doOneIteration() => [CONTEXT] Grid::doOneIteration() => [SAMPLE] Grid::doOneIteration()
 92.0        0.000       103140        3438           0      30000 [CONTEXT] Grid::doOneIteration()
 92.0       103140       103140        3438           0      30000 [SAMPLE] Grid::doOneIteration()
 91.8       102905       102905        1600           0      64316 Grid::doIterations(int) => Grid::doOneIteration()
132.2       130366       148151       34524       28864       4291 -others-
```

Source instrumentation - Example - Output with paraprof

- To visualise the profiling data, we prefer `paraprof`.
- Call `paraprof` from the command line. If you are connecting remotely, then you need to connect with X server activated; `ssh -X sunbird`.

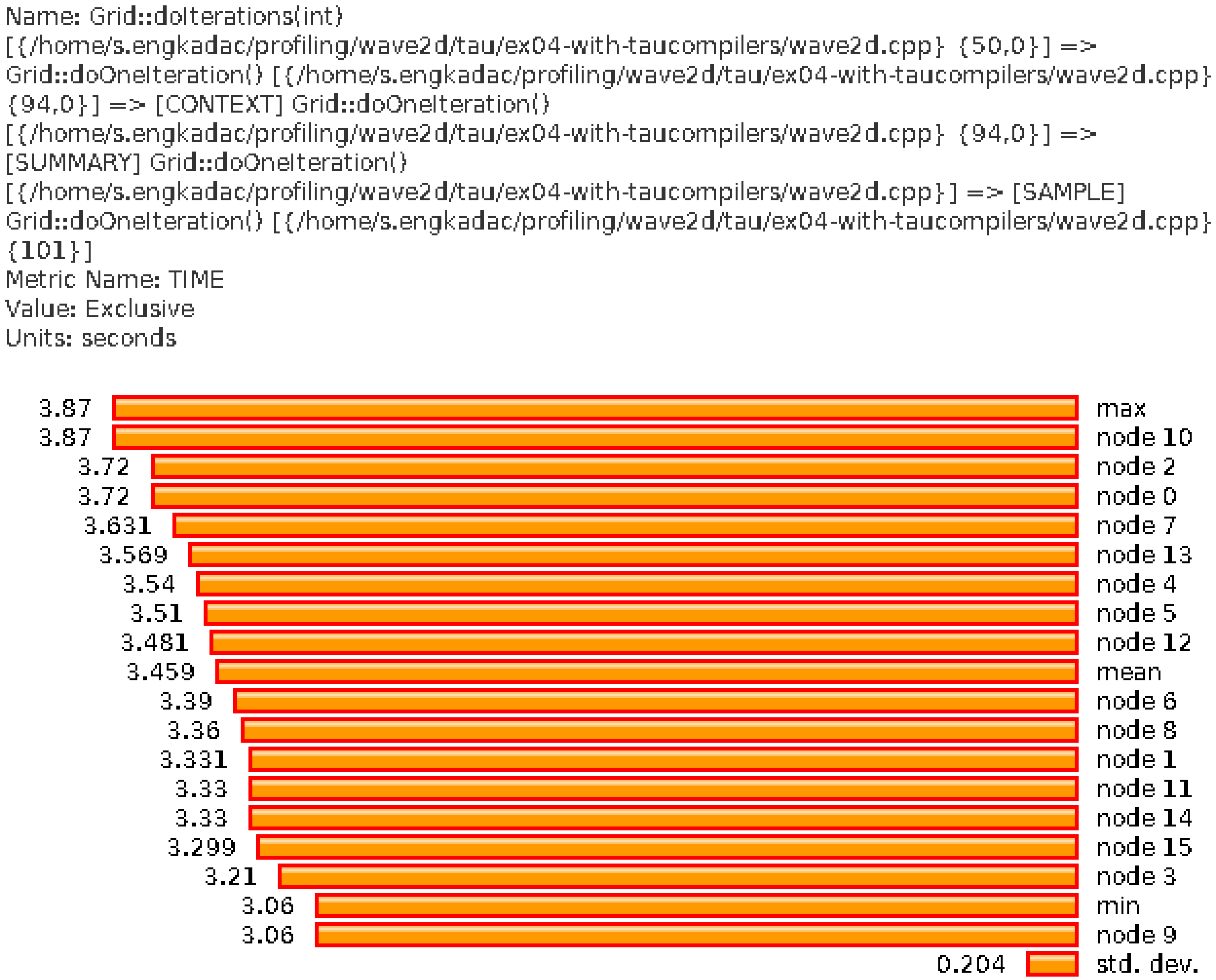


Source instrumentation - Example - Output with paraprof



```
Metric: TIME
Value: Exclusive
Units: seconds

  6.42  [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}]
 6.389  Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}]
 6.389  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}]
  3.21  [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {101}]
  3.21  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {101}]
  1.71  [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {100}]
  1.71  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {100}]
   0.9  [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {97}]
   0.9  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {97}]
  0.54  [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {98}]
  0.54  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {98}]
 0.163  MPI_Send()
 0.163  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Send()
 0.132  MPI_Init()
 0.132  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => MPI_Init()
 0.112  Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}]
 0.112  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}]
  0.09  MPI_Finalize()
  0.09  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => MPI_Finalize()
  0.09  [SUMMARY] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}]
 0.064  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}]
 0.064  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}]
  0.06  [SAMPLE] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {77}]
  0.06  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}] => [CONTEXT] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}] => [SUMMARY] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {77}]
  0.06  [SUMMARY] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}]
  0.06  [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {99}]
  0.06  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [SUMMARY] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {99}]
  0.06  [SAMPLE] pthread_spin_lock [{/usr/lib64/libpthread-2.17.so} {0}]
  0.06  [SAMPLE] mca_btl_vader_component_progress [{btl_vader_component.c} {0}]
  0.06  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Send() => [CONTEXT] MPI_Send() => [SAMPLE] mca_btl_vader_component_progress [{btl_vader_component.c} {0}]
  0.06  [SAMPLE] poll_device [{btl_openib_component.c} {0}]
  0.06  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Send() => [CONTEXT] MPI_Send() => [SAMPLE] poll_device [{btl_openib_component.c} {0}]
  0.03  [SAMPLE] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {25}]
  0.03  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => [CONTEXT] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => [SUMMARY] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {25}]
  0.03  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Waitall() => [CONTEXT] MPI_Waitall() => [SAMPLE] pthread_spin_lock [{/usr/lib64/libpthread-2.17.so} {0}]
  0.03  [SAMPLE] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {26}]
  0.03  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => [CONTEXT] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => [SUMMARY] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {26}]
  0.03  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Send() => [CONTEXT] MPI_Send() => [SAMPLE] pthread_spin_lock [{/usr/lib64/libpthread-2.17.so} {0}]
  0.03  [SAMPLE] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {75}]
  0.03  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}] => [CONTEXT] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}] => [SUMMARY] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {75}]
 0.024  MPI_Waitall()
 0.024  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Waitall()
  0.01  .TAU application
 0.009  Grid::~Grid() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {40,0}]
 0.009  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::~Grid() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {40,0}]
 0.004  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}]
 0.004  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}]
4.4E-4  MPI_Irecv()
4.4E-4  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Irecv()
2.4E-4  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}]
2.4E-4  .TAU application => main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}]
2.1E-4  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}]
2.1E-4  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}]
1.7E-4  MPI_Bcast()
1.7E-4  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => MPI_Bcast()
2.0E-5  MPI_Comm_rank()
1.7E-5  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => MPI_Comm_rank()
1.4E-5  MPI_Comm_size()
1.4E-5  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => MPI_Comm_size()
3.0E-6  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => MPI_Comm_rank()
     0  [CONTEXT] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}]
     0  main [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {160,0}] => Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => [CONTEXT] Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}]
     0  [CONTEXT] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}]
     0  Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {22,0}] => Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}] => [CONTEXT] Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {63,0}]
     0  [CONTEXT] MPI_Send()
     0  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Send() => [CONTEXT] MPI_Send()
     0  [CONTEXT] MPI_Waitall()
     0  Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {106,0}] => MPI_Waitall() => [CONTEXT] MPI_Waitall()
     0  [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}]
     0  Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] => Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] => [CONTEXT] Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}]
```

Source instrumentation - Example - Output



```
Name: Grid::doIterations(int)
[{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {50,0}] =>
Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}
{94,0}] => [CONTEXT] Grid::doOneIteration()
[{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp} {94,0}] =>
[SUMMARY] Grid::doOneIteration()
[{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}] => [SAMPLE]
Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex04-with-taucompilers/wave2d.cpp}
{101}]
Metric Name: TIME
Value: Exclusive
Units: seconds
```

Tracing

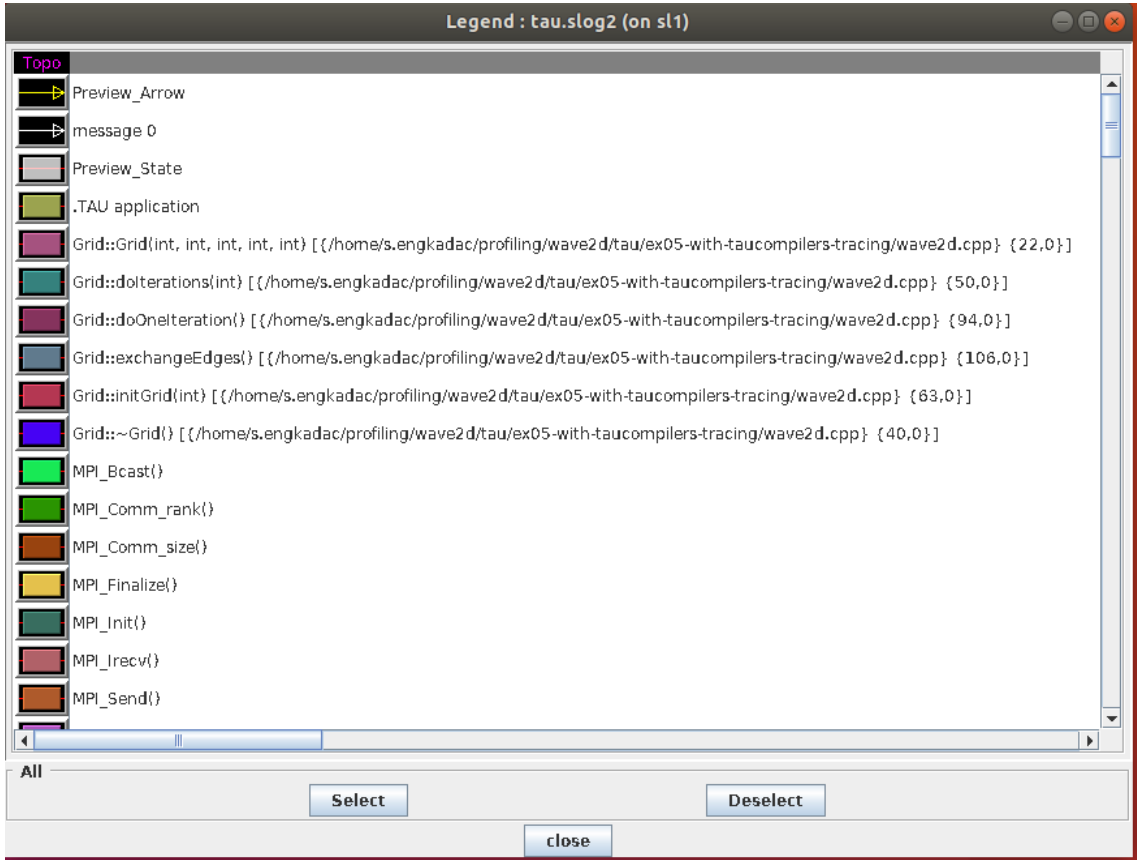
Tracing with TAU

- TAU can automatically instrument the code to do tracing without user intervention (unlike MPE, which requires manual insertion of tracing code).
- This tracing functionality depends on PDT, its C/C++ parser in particular.
- To enable this, we need to the TAU Makefile with the *PDT* option.
- **Note**: tracing consumes an enormous amount of disk space. So, use it with caution.

Tracing with TAU (ex5)

- Tracing is disabled by default. To enable it
  - `export TAU_TRACE=1`
- Enable tracing disables profiling. That is, no profile.* files will be generated.
- The execution of the program will generate .edf and .trc files. These files must be merged first before visualisation.
  - `tau_treemerge.pl`
  - `tau2slog2 tau.trc tau.edf -o tau.slog2`
- Tracing data is dumped into `tau.slog2` which can visualised with *Jumpshot*. (TAU doesn't have a visual tracer).

Visualising tracing data with Jumpshot

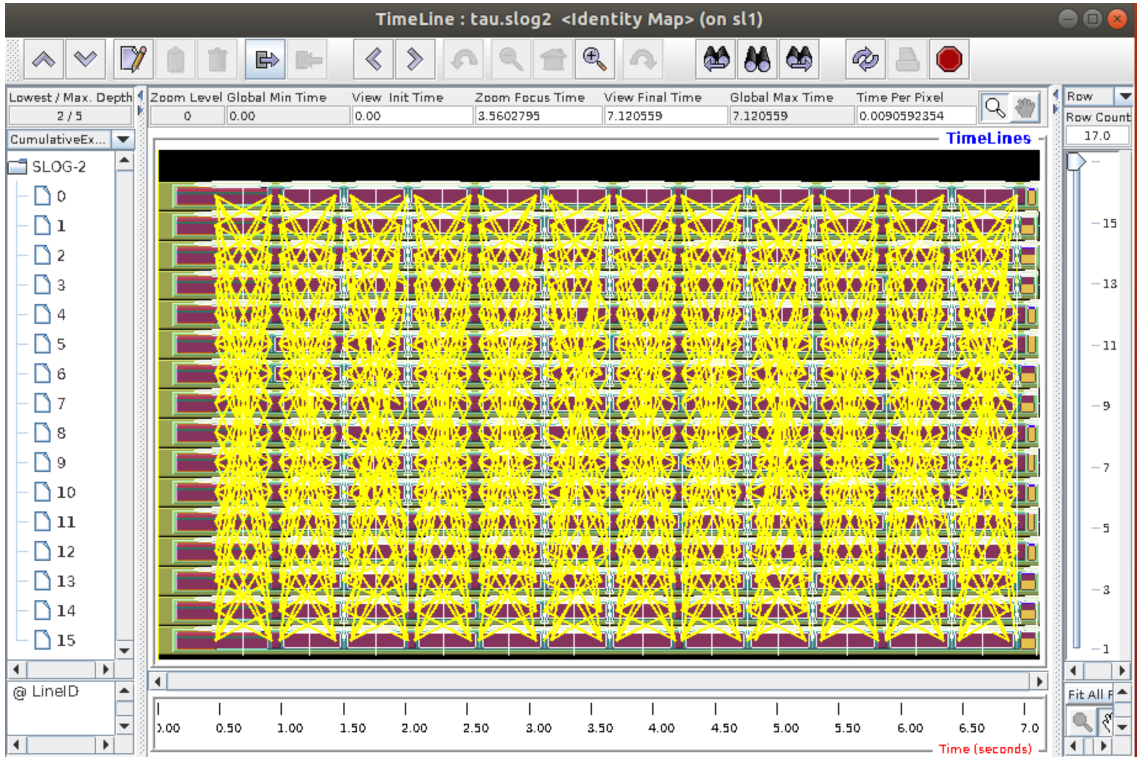


Legend : tau.slog2 (on sl1)

Topo

- → Preview_Arrow
- → message 0
- ⬜ Preview_State
- 🟨 .TAU application
- 🟪 Grid::Grid(int, int, int, int, int) [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {22,0}]
- 🟦 Grid::doIterations(int) [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {50,0}]
- 🟪 Grid::doOneIteration() [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {94,0}]
- 🟦 Grid::exchangeEdges() [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {106,0}]
- 🟥 Grid::initGrid(int) [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {63,0}]
- 🟦 Grid::~Grid() [{/home/s.engkadac/profiling/wave2d/tau/ex05-with-taucompilers-tracing/wave2d.cpp} {40,0}]
- 🟩 MPI_Bcast()
- 🟩 MPI_Comm_rank()
- 🟫 MPI_Comm_size()
- 🟨 MPI_Finalize()
- 🟩 MPI_Init()
- 🟥 MPI_Irecv()
- 🟧 MPI_Send()

Visualising tracing data with Jumpshot



TimeLine : tau.slog2 <Identity Map> (on sl1)

Lowest / Max. Depth	Zoom Level	Global Min Time	View Init Time	Zoom Focus Time	View Final Time	Global Max Time	Time Per Pixel
2 / 5	0	0.00	0.00	3.5602795	7.120559	7.120559	0.0090592354

Row Count: 17.0

- We can zoom into the graphs by drawing a window around the portion of interest.