Performance Tools for MPI

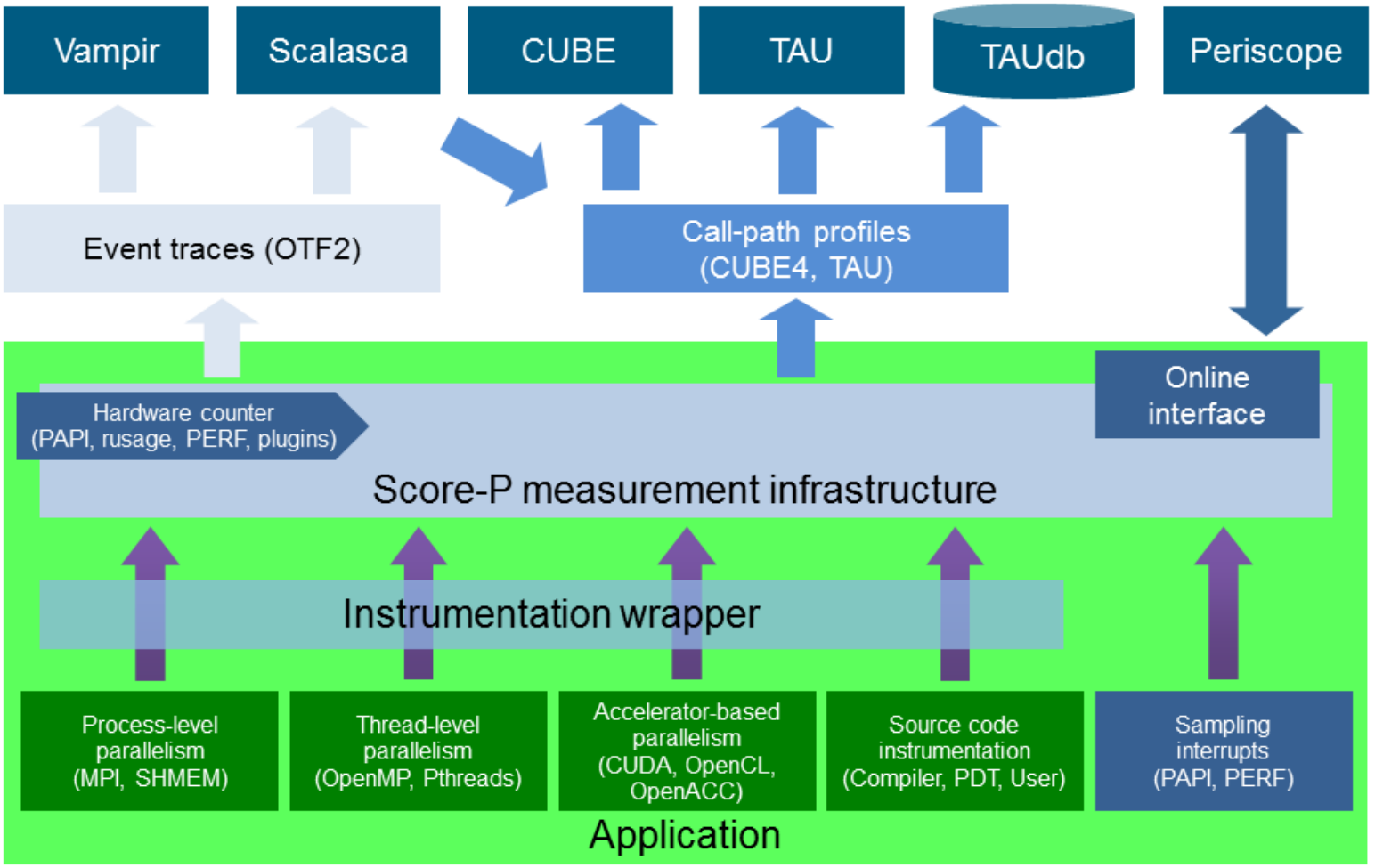
PPCES 2017

Hristo Iliev
IT Center / JARA-HPC

Performance Tools

- **MPI programs do not always behave as expected**
  - → Communication delays
  - → Synchronisation overhead
  - → Inefficient communication patterns
- **Tools exist for profiling and tracing the program execution**
  - → Profiling
    - →How often certain functions are called?
    - →How much time is spent in certain parts?
    - →How much data is exchanged in certain parts?
  - → Tracing
    - →A timeline of events happening during the execution

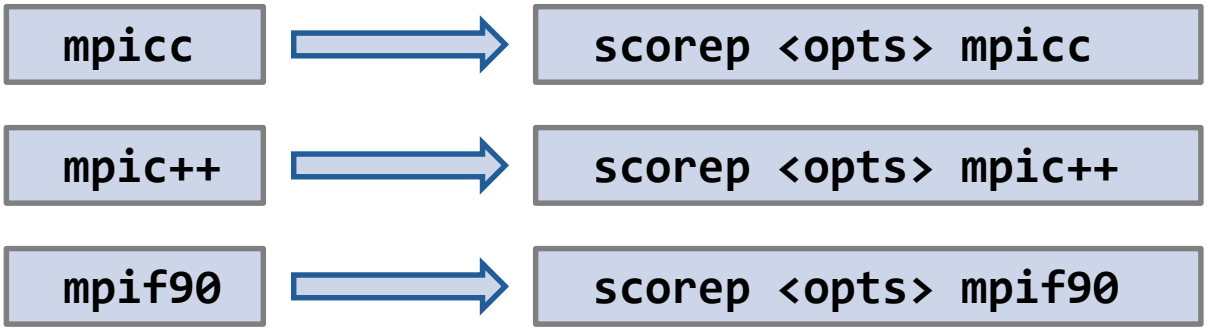
Performance Infrastructure Score-P



Source: Score-P website (ZIH)

Code Instrumentation with Score-P

- **Code first has to be instrumented accordingly:**
  - → Recompile with instrumentation



  - → When run, the *instrumented binary* produces event data in several formats

- **Instrumentation type**

→ **`--compiler`**	compiler assisted instrumentation (default) very detailed; huge trace files
→ **`--user --nocompiler`**	manual tracing using the Score-P API
→ **`--mpp=mpi`**	traces MPI events (auto)
→ **`--thread=openmp`**	traces OpenMP parallel constructs (auto)

Code Instrumentation with Score-P

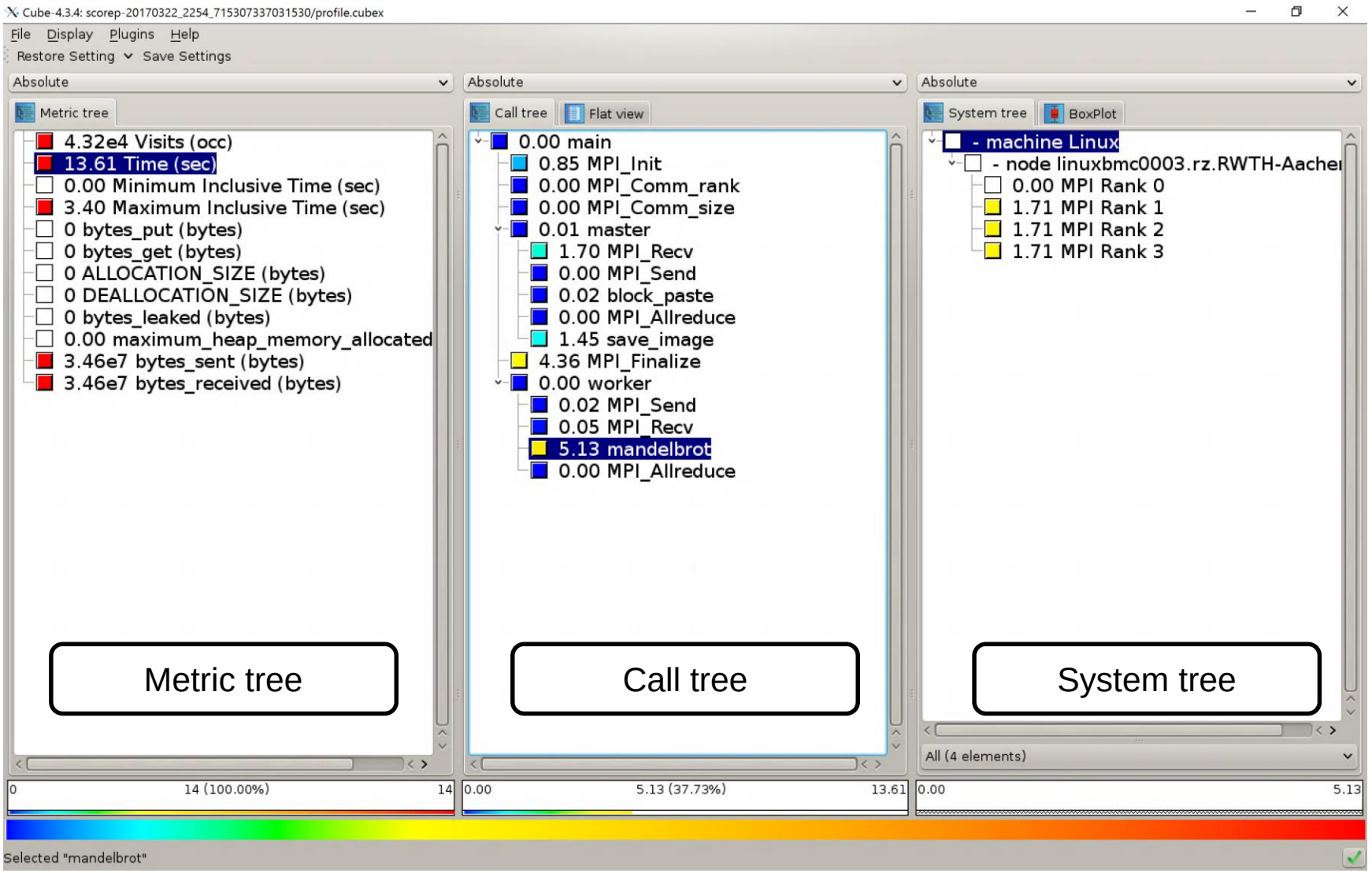
- **Score-P is controlled by environment variables**
  - → **`SCOREP_ENABLE_TRACING`** – enables writing of trace files when set to **1** / **true**
  - → **`SCOREP_ENABLE_PROFILNG`** – enables writing of profiling information
  - → **`SCOREP_TOTAL_MEMORY`** – size of the event buffers, e.g. **200M**
  - → **`SCOREP_EXPERIMENT_DIRECTORY`** – location of the output files
  - → **`SCOREP_FILTERING_FILE`** – location of a file containing filtering rules

- **Event buffers are flushed when full, which takes time**
  - → Significant skew in program's performance profile possible
  - → Set buffer size accordingly to prevent flushes or limit what is being recorded

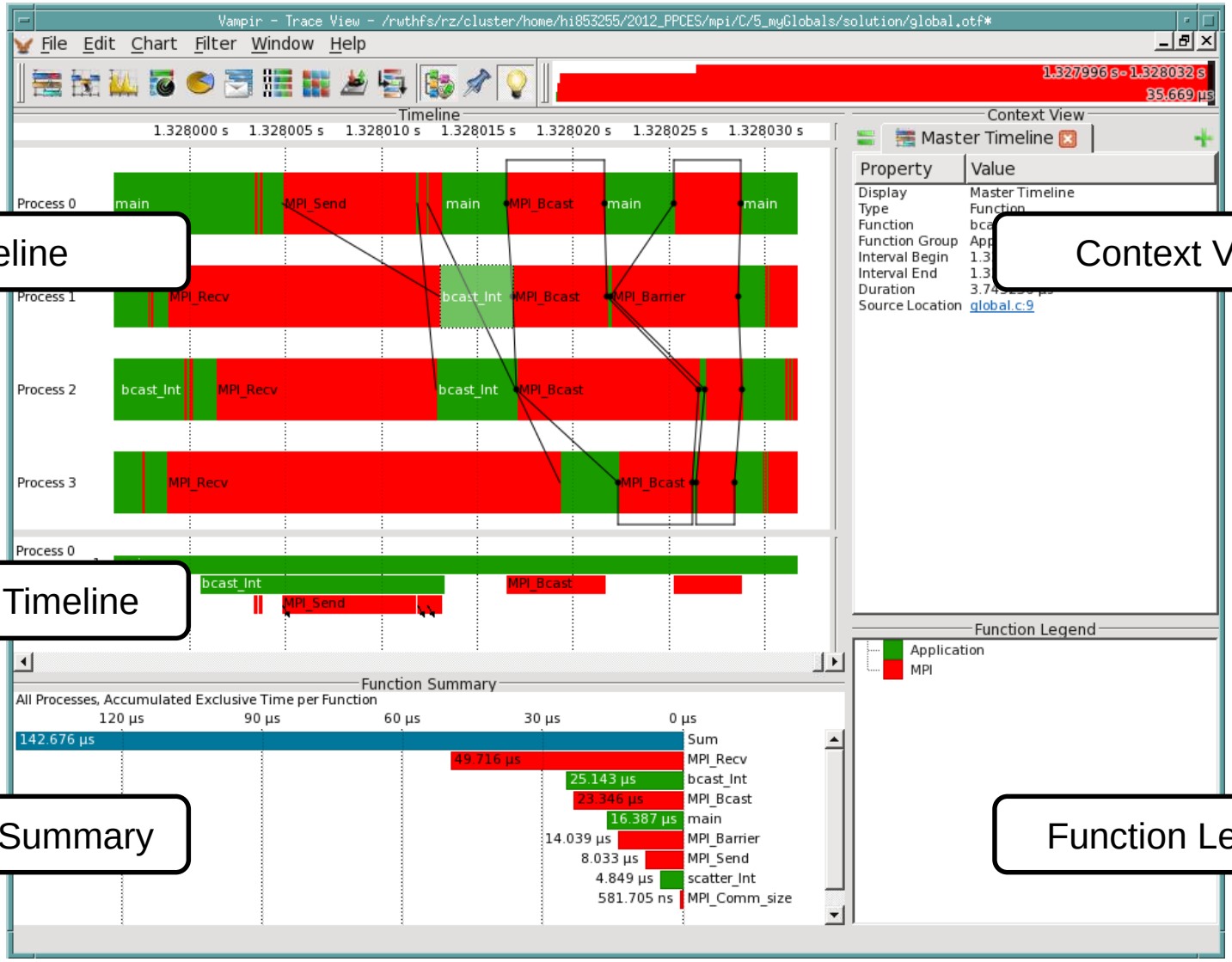
Sample Workflow with Score-P

```
> module load UNITE scorep cube vampir
> scorep $MPICC -o a.out program.c
--- an instrumented executable produced ---
> mpiexec -n 4 -x SCOREP_ENABLE_TRACING=0 -x SCOREP_ENABLE_PROFILING=1 \
          a.out
--- program output ---
--- profiling data written in scorep-<timestamp>/profile.cubex ---
> cube scorep-<timestamp>/profile.cubex
--- Cube GUI opens ---
> scorep-score scorep-<timestamp>/profile.cubex
Estimated aggregate size of event trace:                   26kB
Estimated requirements for largest trace buffer (max_buf): 11kB
Estimated memory requirements (SCOREP_TOTAL_MEMORY):       4097kB
                                                           ^^^^^^
> mpiexec -n 4 -x SCOREP_ENABLE_TRACING=1 -x SCOREP_ENABLE_PROFILING=0 \
          -x SCOREP_TOTAL_MEMORY=5M a.out
--- traces written in scorep-<timestamp>/traces.otf2 ---
> vampir scorep-<timestamp>/traces.otf2
--- Vampir GUI opens ---
```

Cube


Cube-4.3.4: scorep-20170322_2254_715307337031530/profile.cubex
File Display Plugins Help
Restore Setting Save Settings
Absolute
Metric tree
4.32e4 Visits (occ)
13.61 Time (sec)
0.00 Minimum Inclusive Time (sec)
3.40 Maximum Inclusive Time (sec)
0 bytes_put (bytes)
0 bytes_get (bytes)
0 ALLOCATION_SIZE (bytes)
0 DEALLOCATION_SIZE (bytes)
0 bytes_leaked (bytes)
0.00 maximum_heap_memory_allocated
3.46e7 bytes_sent (bytes)
3.46e7 bytes_received (bytes)
Absolute
Call tree
Flat view
0.00 main
0.85 MPI_Init
0.00 MPI_Comm_rank
0.00 MPI_Comm_size
0.01 master
1.70 MPI_Recv
0.00 MPI_Send
0.02 block_paste
0.00 MPI_Allreduce
1.45 save_image
4.36 MPI_Finalize
0.00 worker
0.02 MPI_Send
0.05 MPI_Recv
5.13 mandelbrot
0.00 MPI_Allreduce
Absolute
System tree
BoxPlot
- machine Linux
- node linuxbmc0003.rz.RWTH-Aache
0.00 MPI Rank 0
1.71 MPI Rank 1
1.71 MPI Rank 2
1.71 MPI Rank 3
All (4 elements)
Metric tree
Call tree
System tree
0
14 (100.00%)
14
0.00
5.13 (37.73%)
13.61
0.00
5.13
Selected "mandelbrot"

Vampir



Timeline

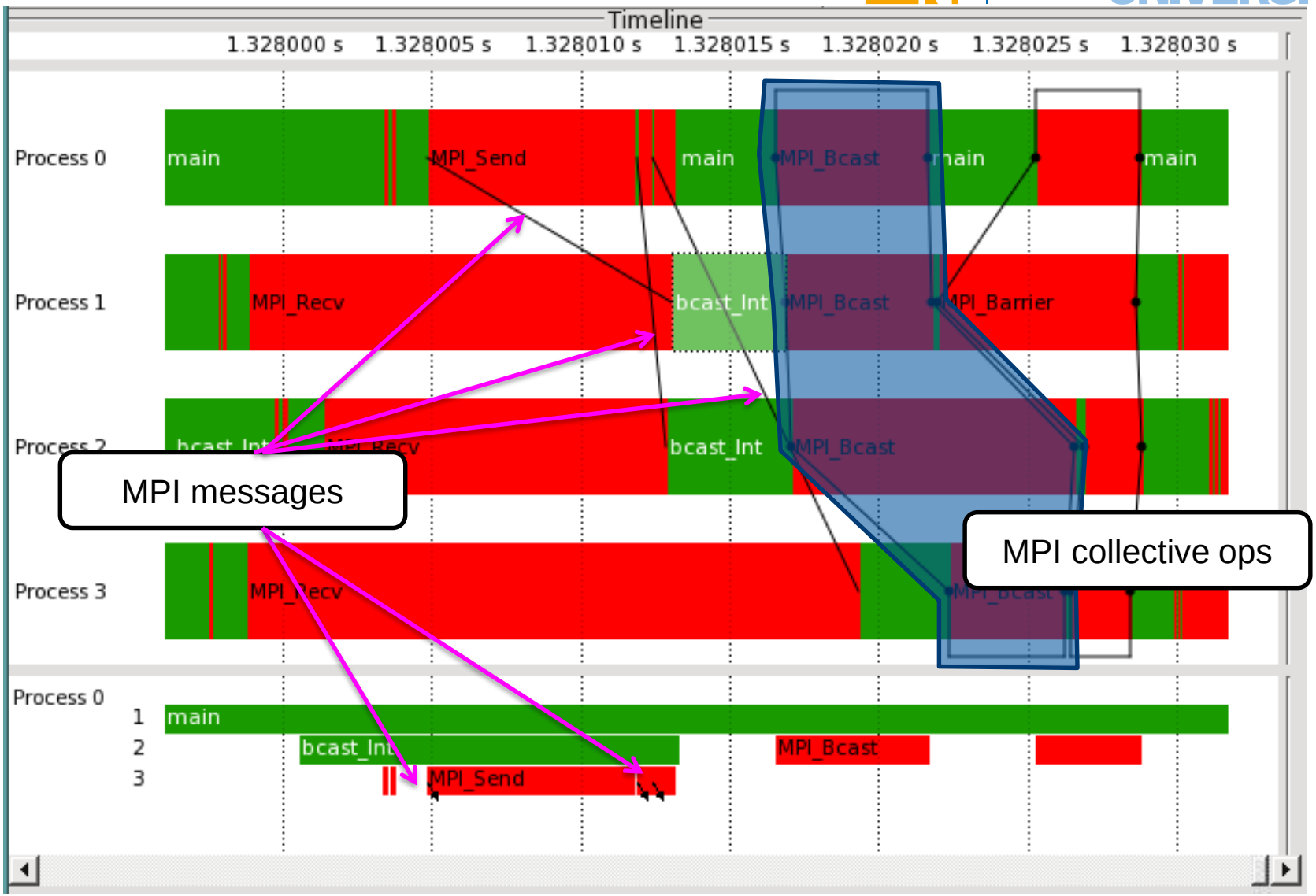
Process Timeline

Function Summary

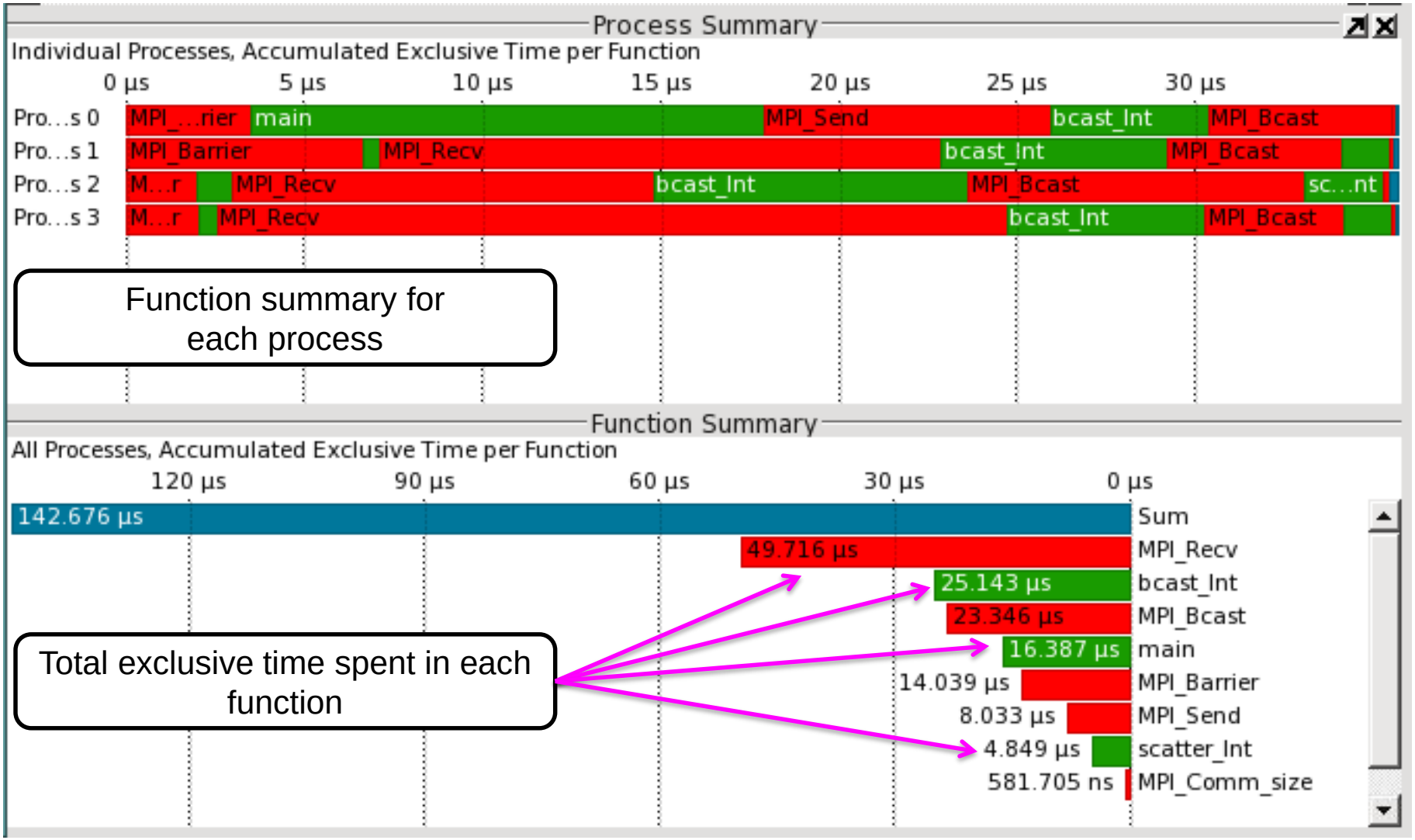
Context View

Function Legend

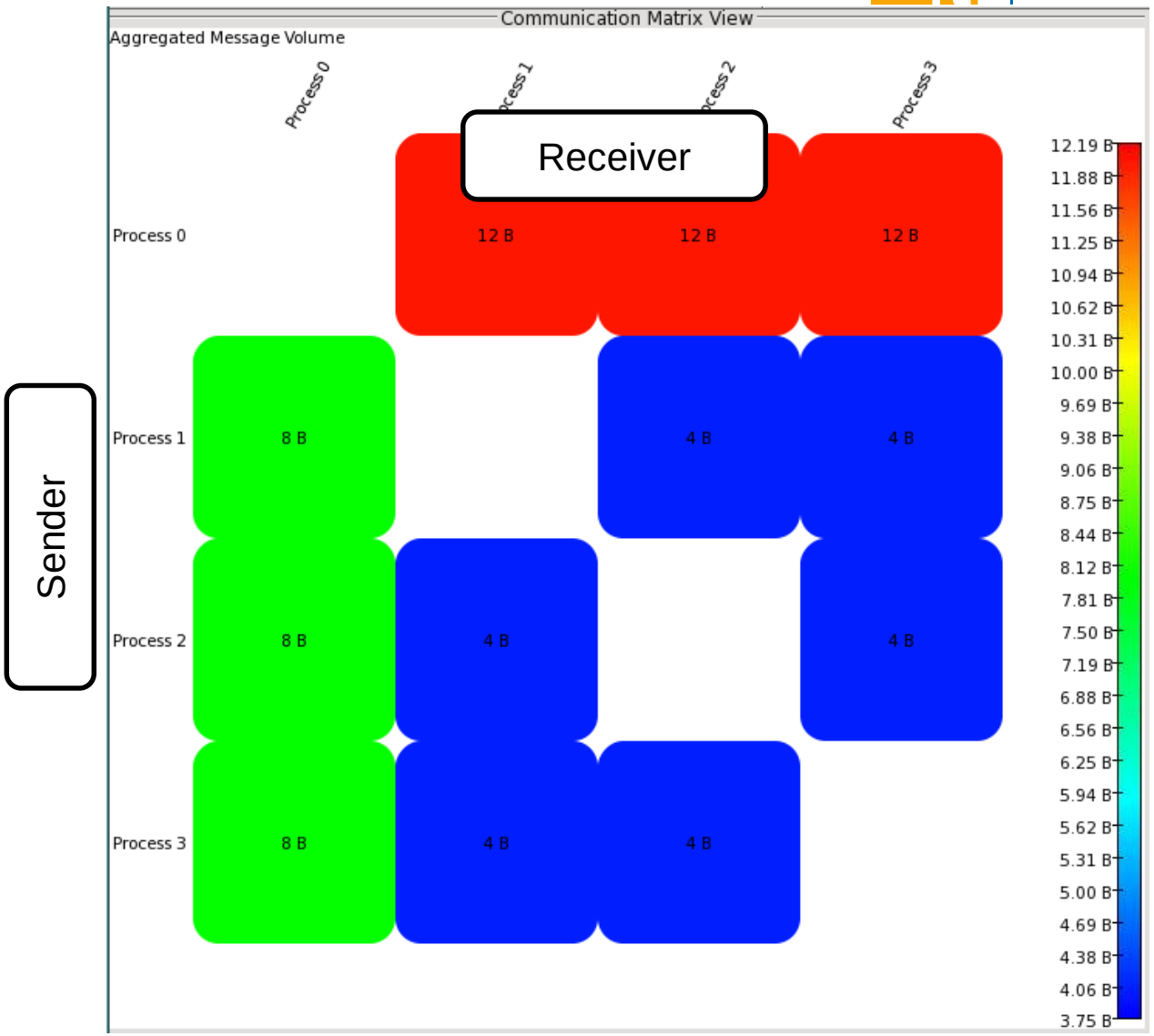
Vampir: Timelines



Vampir: Summaries



Vampir: Communication Matrix



Live Demonstration

Thank you for your attention!