High Performance Computing Systems (CMSC714)

Lecture 8: Perf. Analysis & Visualization

Abhinav Bhatele, Department of Computer Science

UNIVERSITY OF MARYLAND

Summary of last lecture

- Task-based programming models and Charm++
- Key principles:
  - Over-decomposition, virtualization
  - Message-driven execution
- Automatic load balancing, checkpointing, fault tolerance

Tracing tools

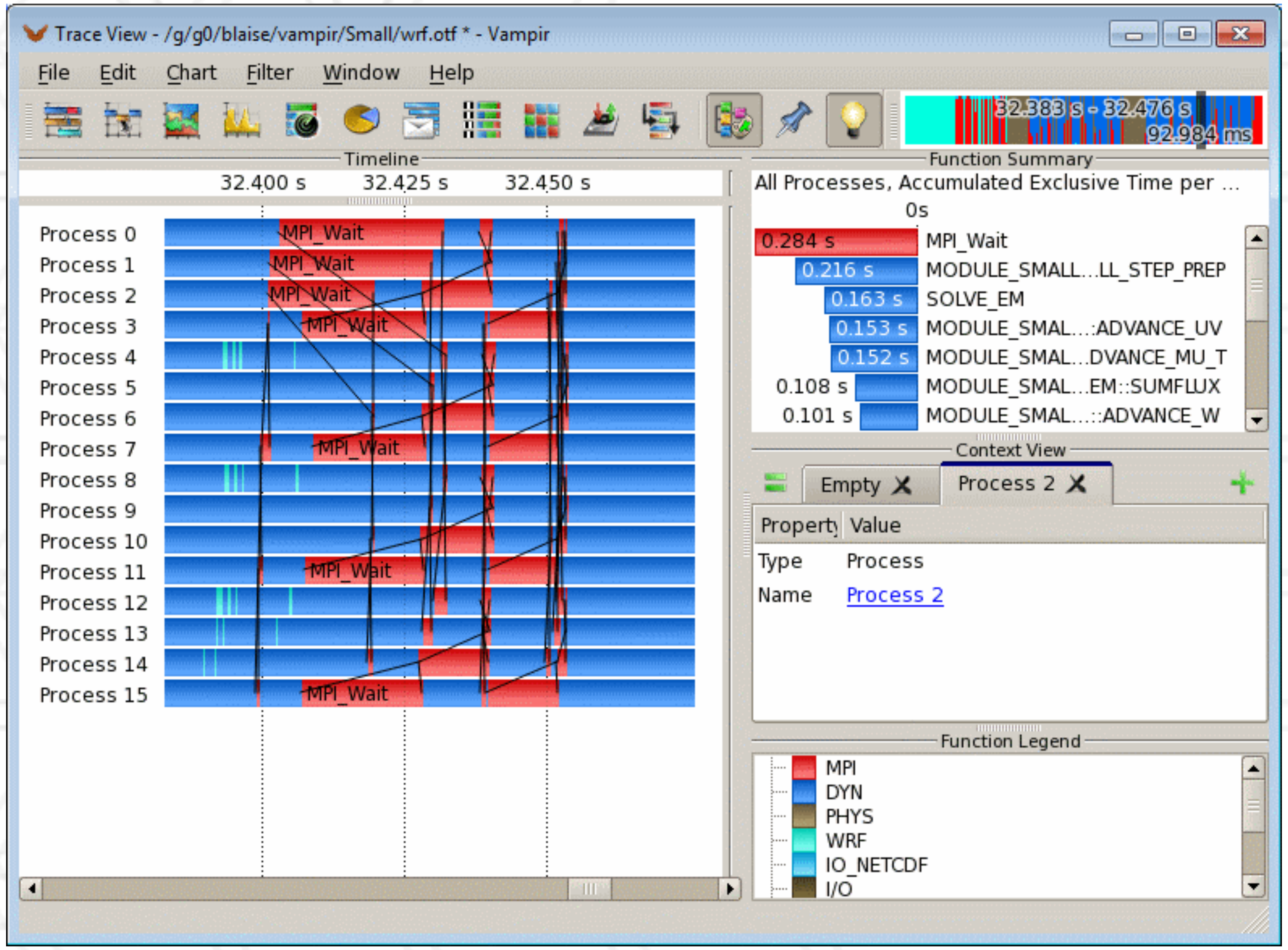
- Record all the events in the program with timestamps
- Events: function calls, MPI events, etc.

Vampir visualization: https://hpc.llnl.gov/software/development-environment-software/vampir-vampir-server

Tracing tools

- Record all the events in the program with timestamps

- Events: function calls, MPI events, etc.



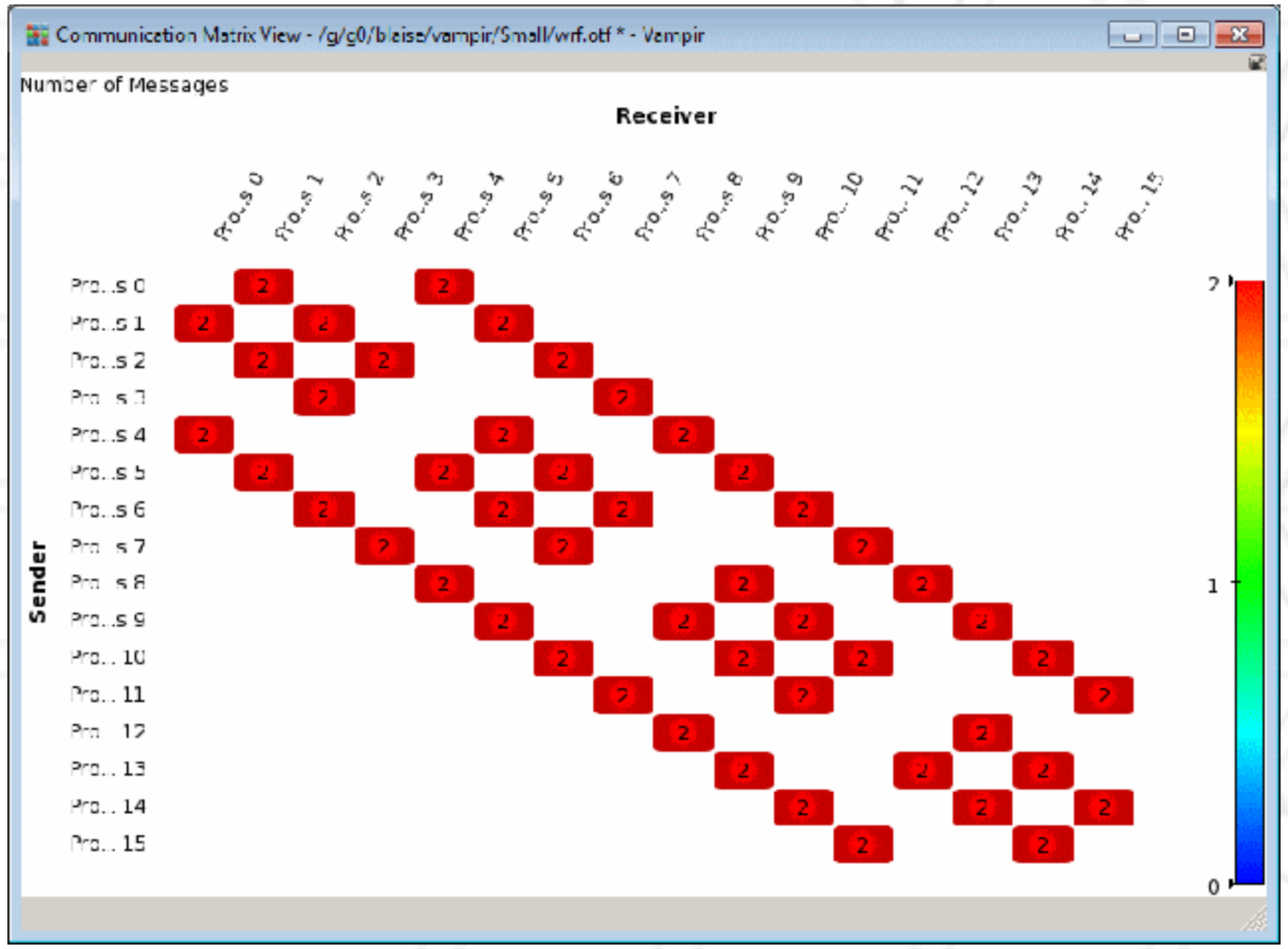
0.284 s	MPI_Wait
0.216 s	MODULE_SMALL...LL_STEP_PREP
0.163 s	SOLVE_EM
0.153 s	MODULE_SMAL...:ADVANCE_UV
0.152 s	MODULE_SMAL...DVANCE_MU_T
0.108 s	MODULE_SMAL...EM::SUMFLUX
0.101 s	MODULE_SMAL...::ADVANCE_W

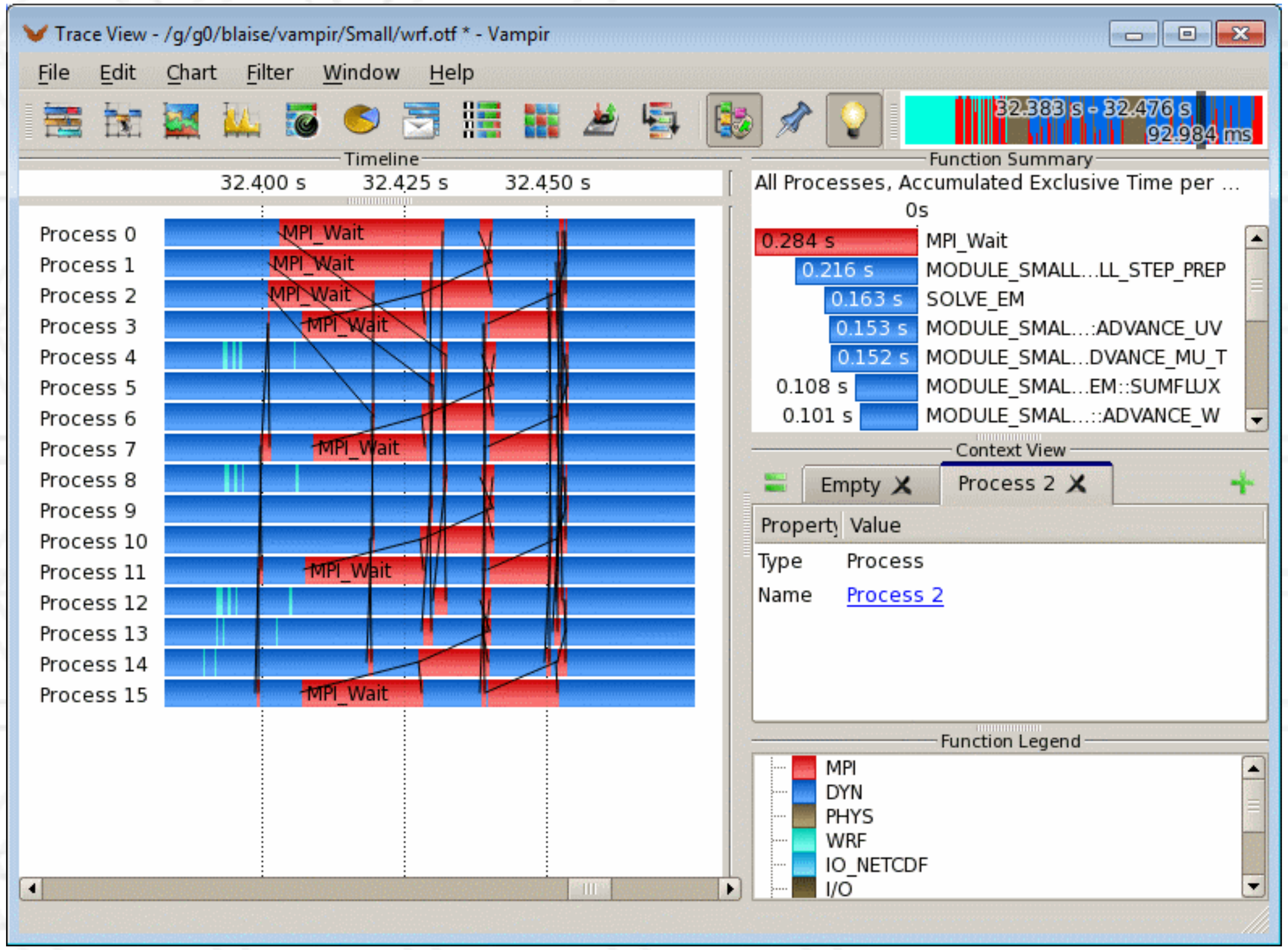
Property	Value
Type	Process
Name	Process 2

Vampir visualization: https://hpc.llnl.gov/software/development-environment-software/vampir-vampir-server

Tracing tools

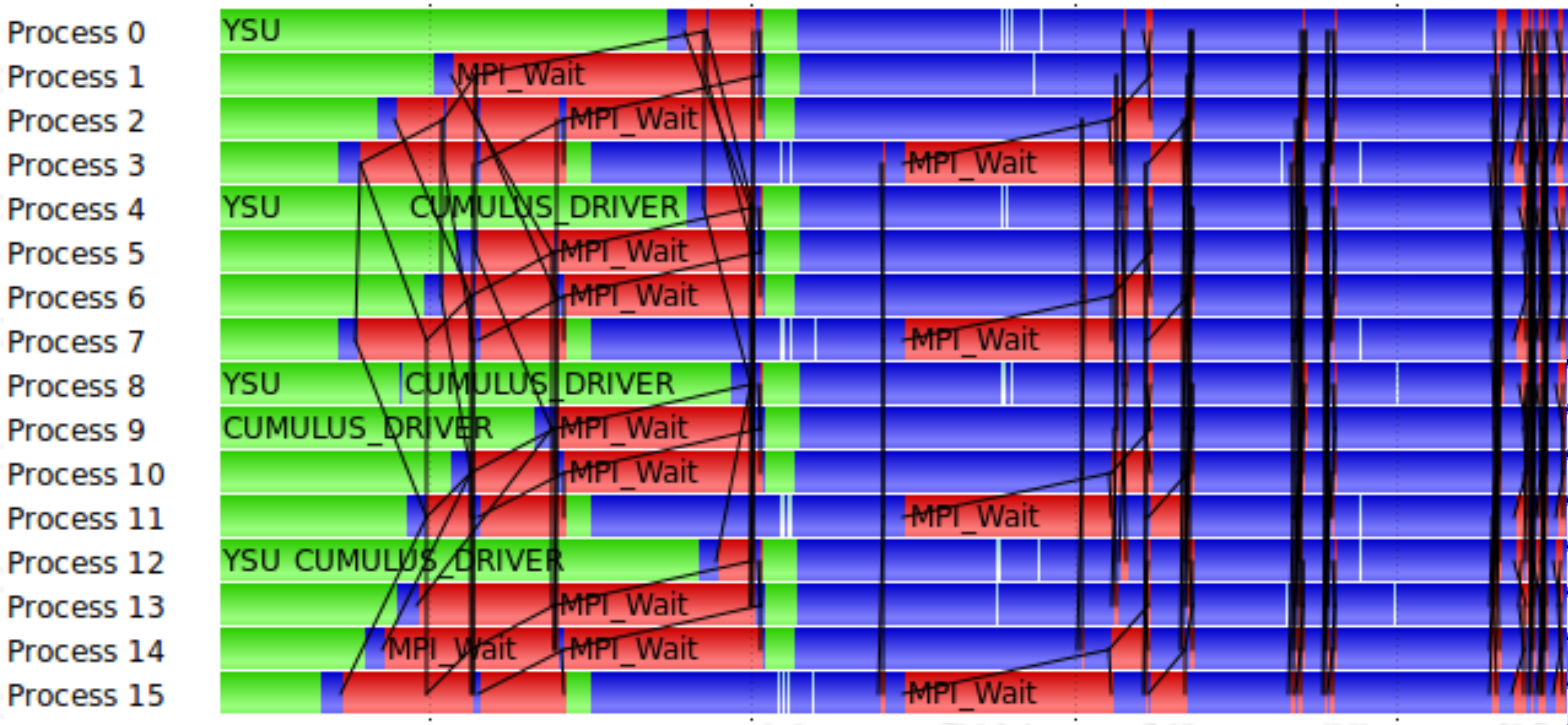
- Record all the events in the program with timestamps
- Events: function calls, MPI events, etc.



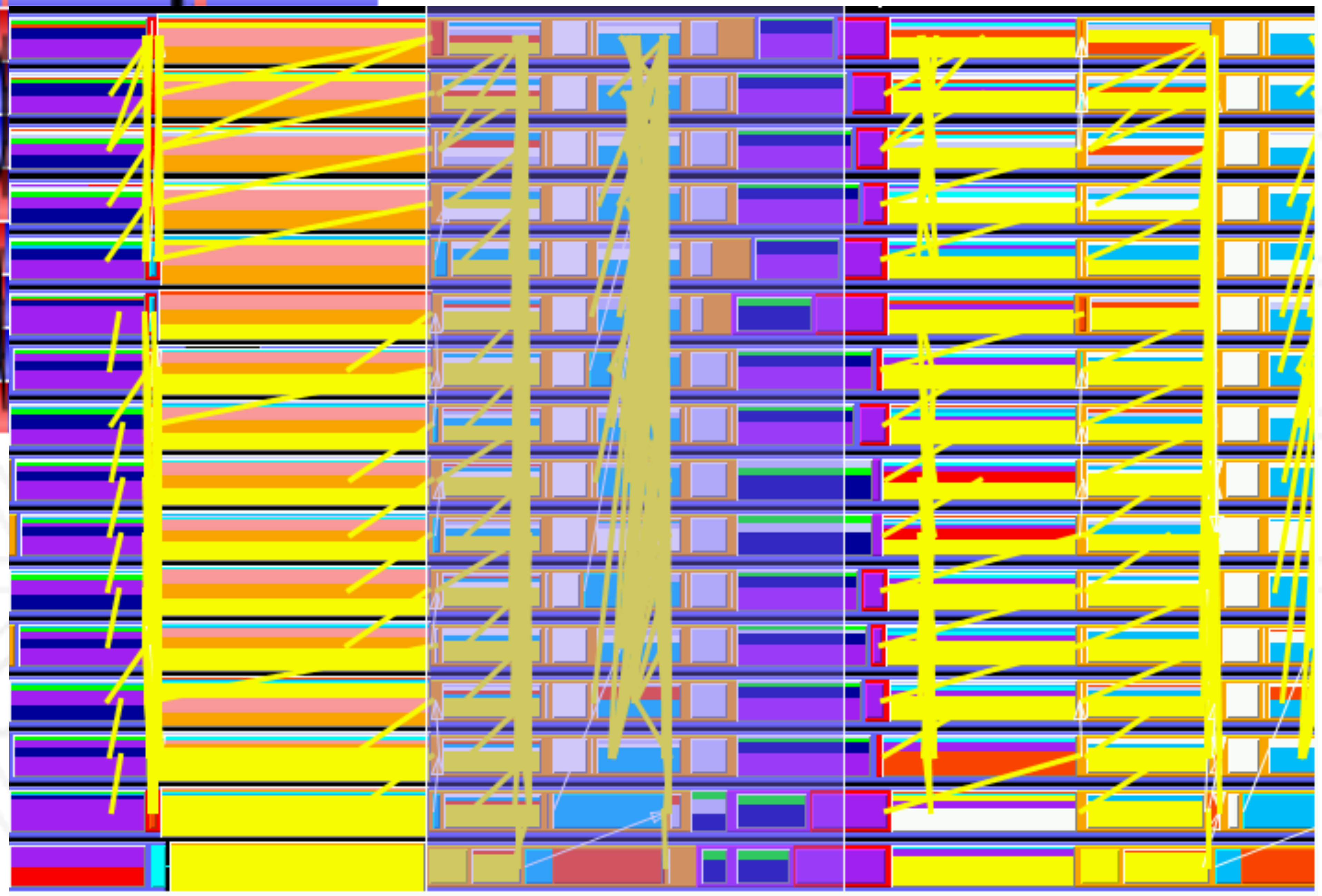


Vampir visualization: https://hpc.llnl.gov/software/development-environment-software/vampir-vampir-server

MPI trace visualization



Vampir

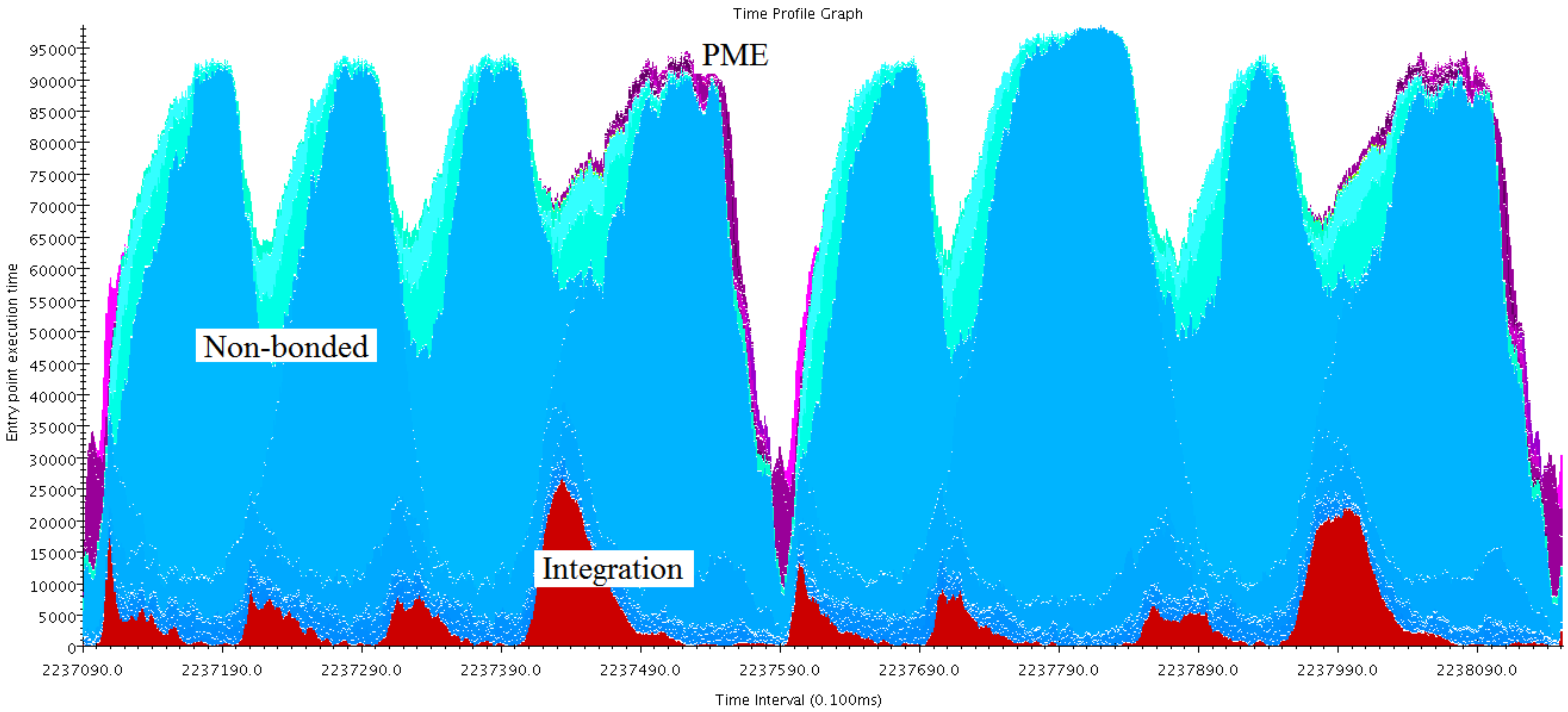


Jumpshot

Projections Performance Analysis Tool

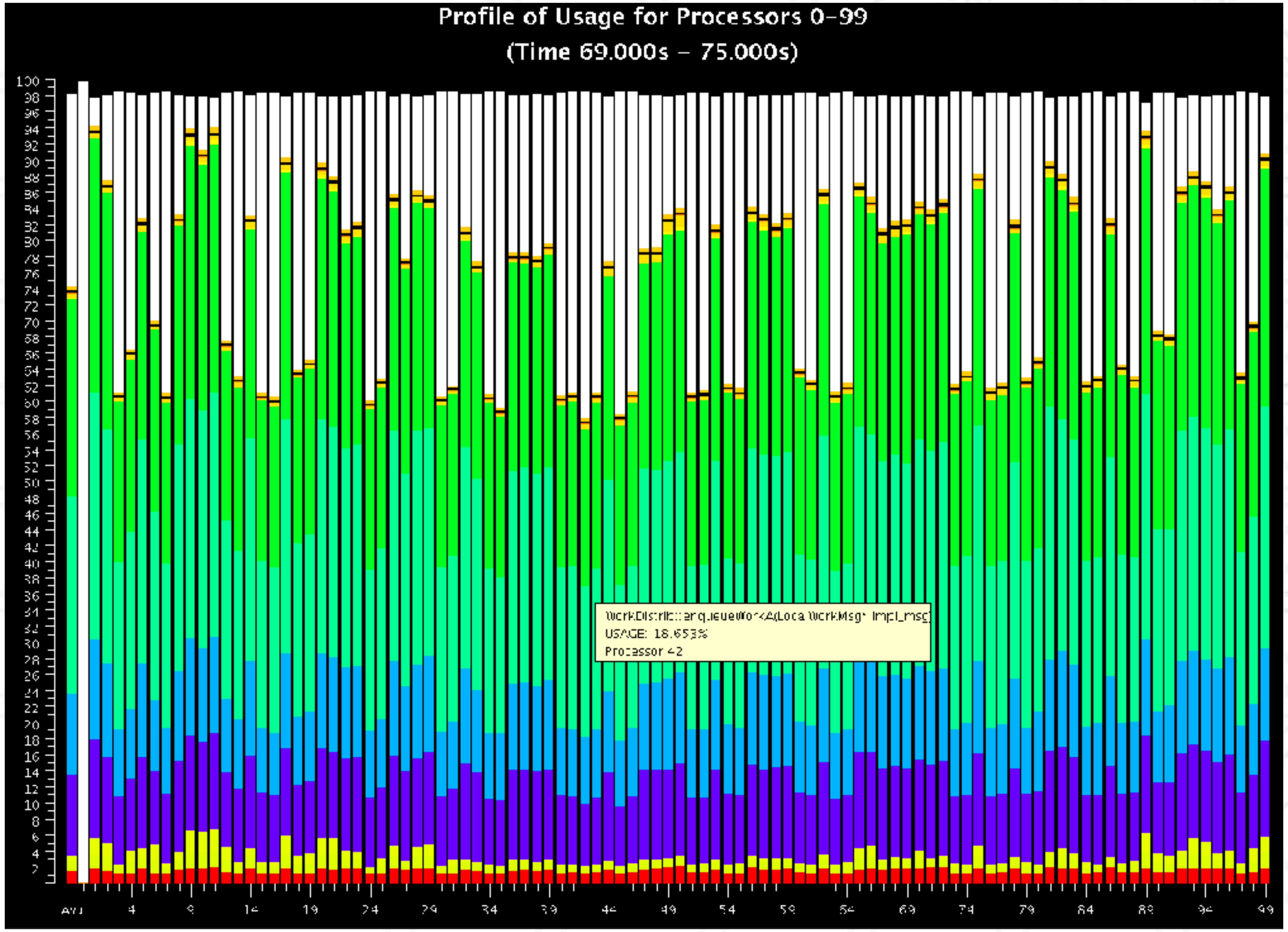
- For Charm++/Adaptive MPI programs
- Instrumentation library
  - Records data at the granularity of chares (Charm++ objects)
- Java-based GUI

Time Profile



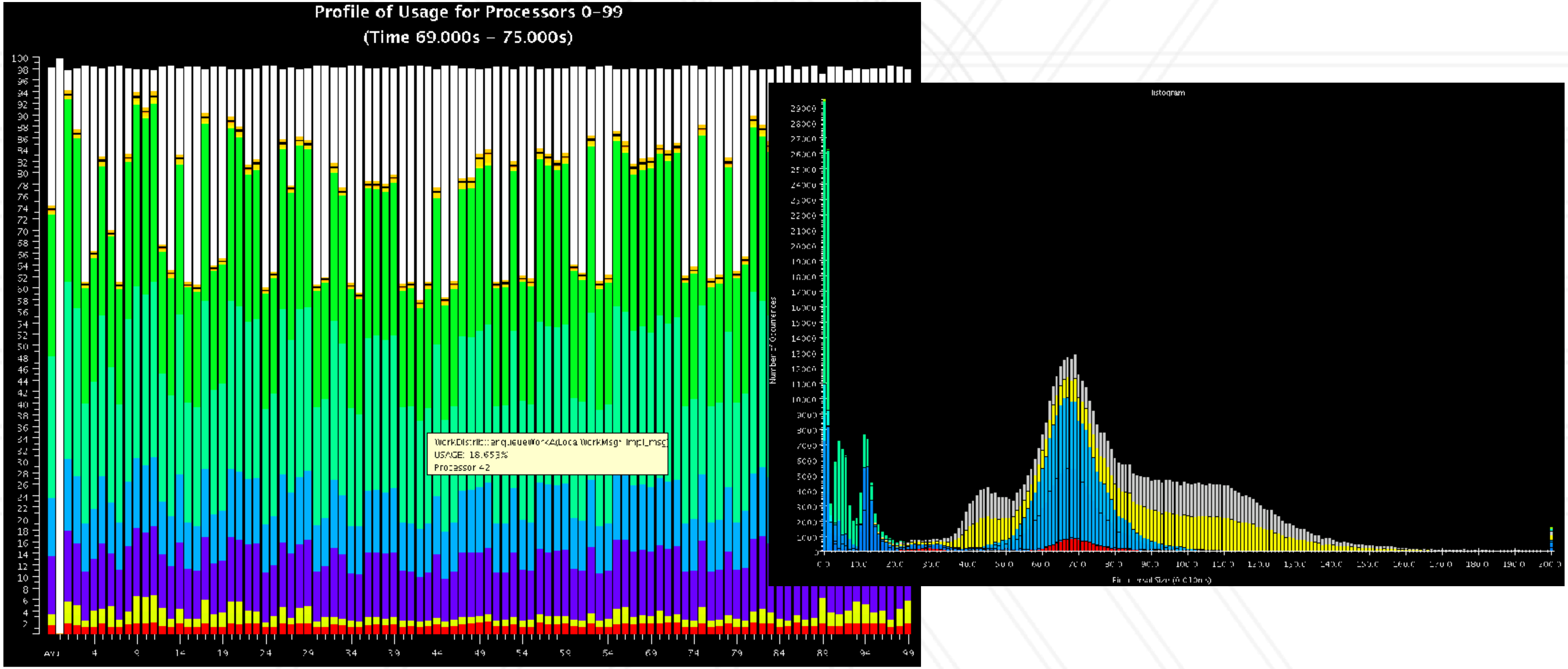
https://charm.readthedocs.io/en/latest/projections/manual.html

Usage Profile & Histogram View



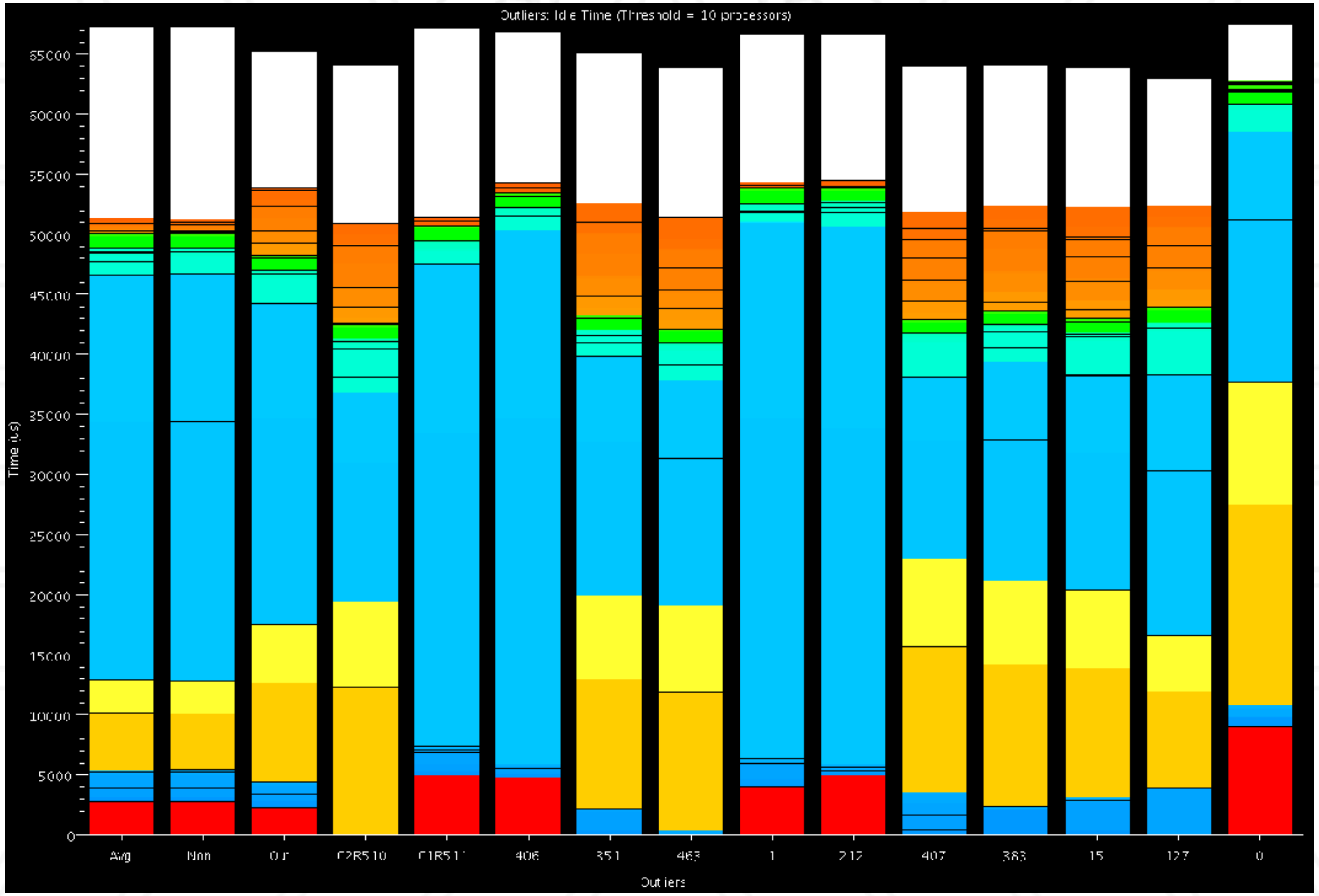
WorkDistrib::enqueueWorkA(LocalWorkMsg* impl_msg)
USAGE: 18.653%
Processor 42

Usage Profile & Histogram View

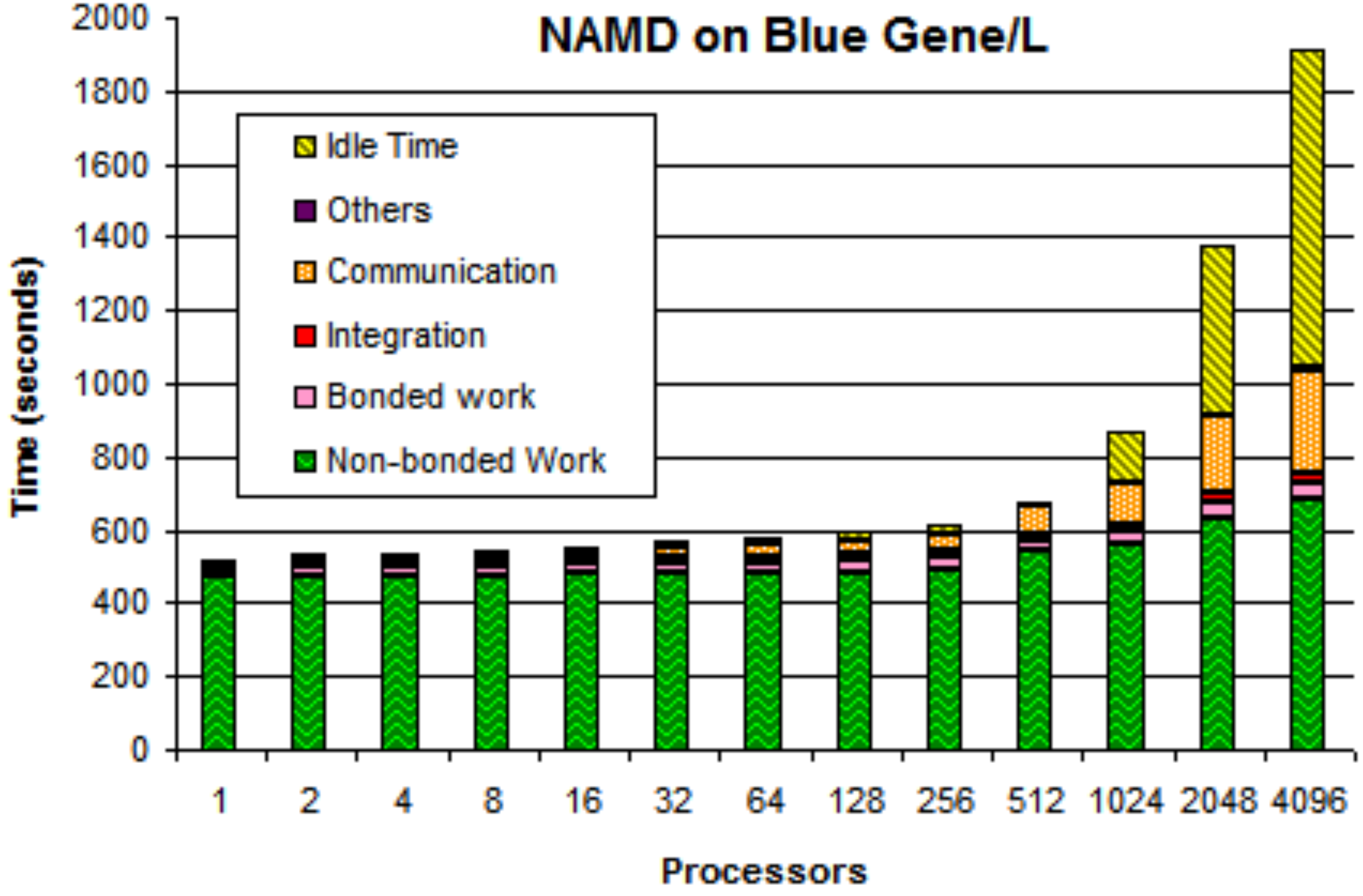


WorkDistrib::enqueueWorkA(LocalWorkMsg* impl_msg)
USAGE: 18.653%
Processor 42

Outlier Analysis

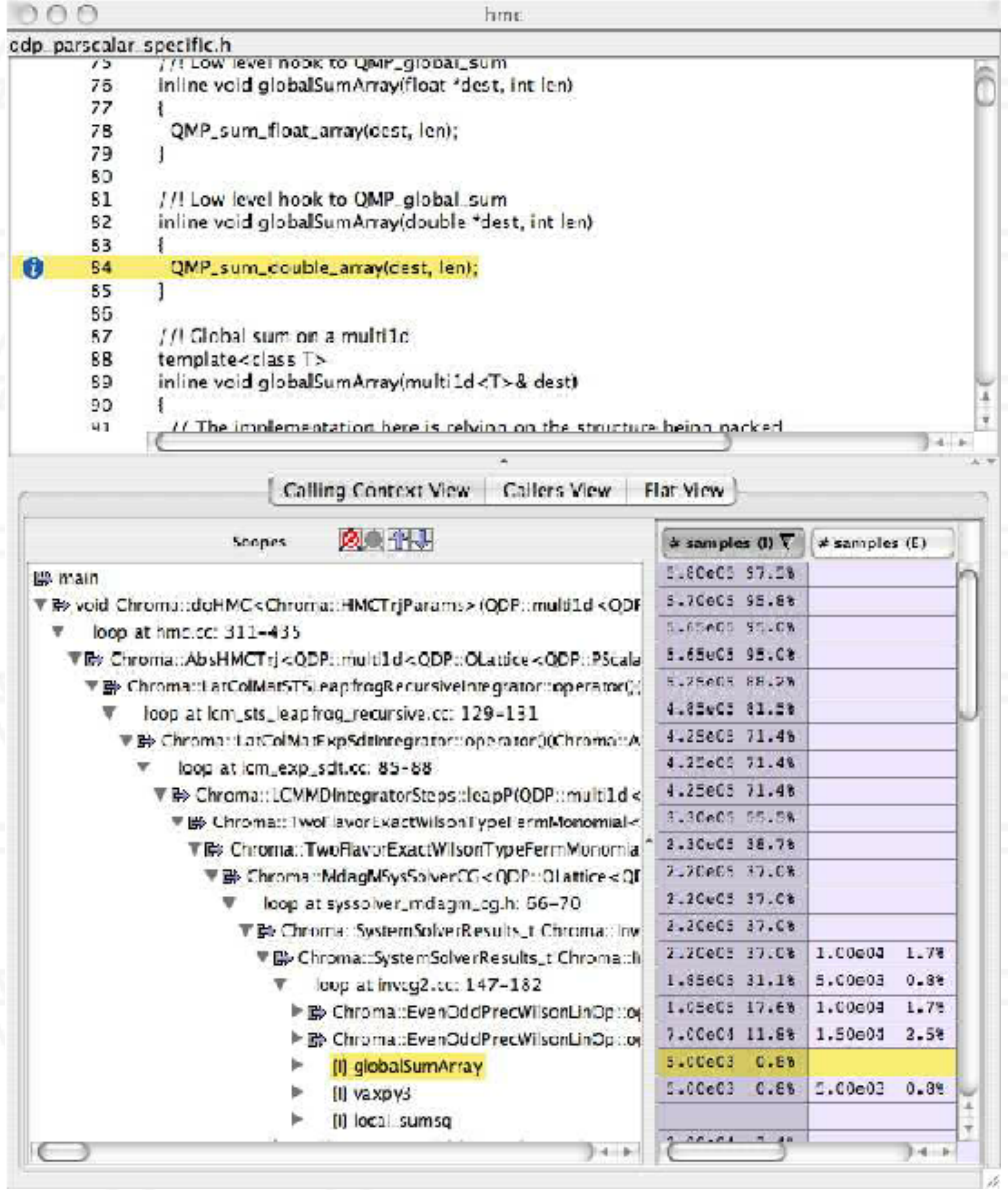


Scripting for multi-run comparisons



Limitations of current *analysis* tools

- Support their own unique format(s)
- Limited support for saving or automating analysis
- Most tools only support viewing one dataset at a time
- Lack capabilities to sub-select and focus on specific parts



```
//! Low level hook to QMP_global_sum
inline void globalSumArray(float *dest, int len)
{
  QMP_sum_float_array(dest, len);
}

//! Low level hook to QMP_global_sum
inline void globalSumArray(double *dest, int len)
{
  QMP_sum_double_array(dest, len);
}

//! Global sum on a multi1d
template<class T>
inline void globalSumArray(multi1d<T>& dest)
{
  // The implementation here is relying on the structure being packed
```

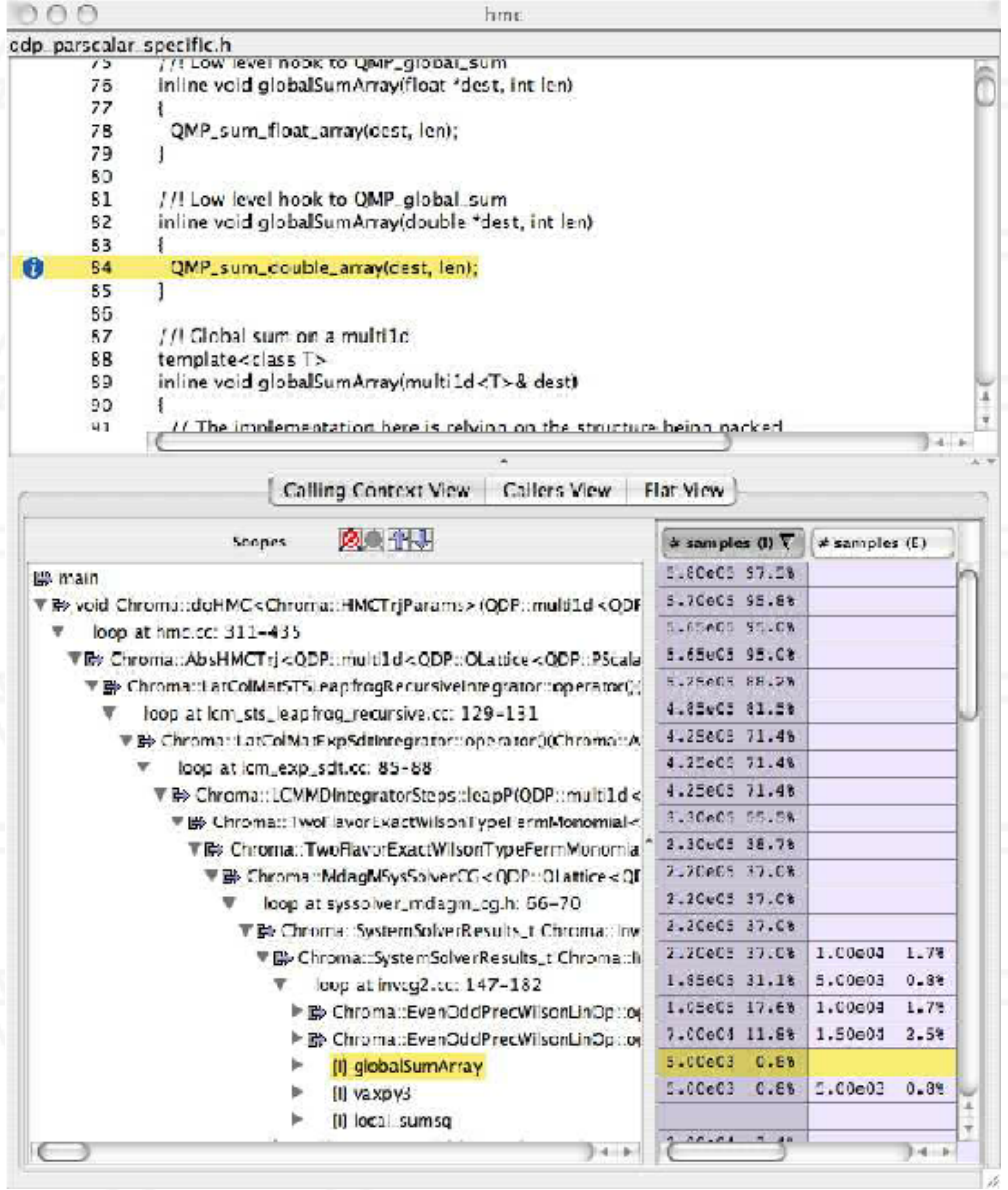
Scopes	# samples (I)	# samples (E)
main	5.80e05 97.2%	
void Chroma::doHMC<Chroma::HMCTrjParams>(QDP::multi1d<QDF	5.70e05 95.6%	
loop at hmc.cc: 311-435	5.65e05 95.0%	
Chroma::AbsHMCTrj<QDP::multi1d<QDP::OLattice<QDP::PScala	5.65e05 95.0%	
Chroma::LatColMatSTSLeapfrogRecursiveIntegrator::operator()(	5.25e05 88.2%	
loop at lcm_sts_leapfrog_recursive.cc: 129-131	4.85e05 81.5%	
Chroma::LatColMatExpSdtIntegrator::operator()(Chroma::A	4.25e05 71.4%	
loop at lcm_exp_sdt.cc: 85-88	4.25e05 71.4%	
Chroma::LCMMDIntegratorSteps::leapP(QDP::multi1d<	4.25e05 71.4%	
Chroma::TwoFlavorExactWilsonTypeFermMonomial<	3.30e05 55.5%	
Chroma::TwoFlavorExactWilsonTypeFermMonomia	2.30e05 38.7%	
Chroma::MdagMSysSolverCG<QDP::OLattice<QI	2.20e05 37.0%	
loop at syssolver_mdagm_cg.h: 56-70	2.20e05 37.0%	
Chroma::SystemSolverResults_t Chroma::Inv	2.20e05 37.0%	
Chroma::SystemSolverResults_t Chroma::I	2.20e05 37.0%	1.00e04 1.7%
loop at invcg2.cc: 147-182	1.85e05 31.1%	5.00e03 0.8%
Chroma::EvenOddPrecWilsonLinOp::o	1.05e05 17.6%	1.00e04 1.7%
Chroma::EvenOddPrecWilsonLinOp::o	7.00e04 11.8%	1.50e04 2.5%
[I] globalSumArray	5.00e03 0.8%	
[I] vaxpy3	5.00e03 0.8%	5.00e03 0.8%
[I] local_sumsq		

hpcviewer's GUI

Limitations of current *analysis* tools

- Support their own unique format(s)
- Limited support for saving or automating analysis
- Most tools only support viewing one dataset at a time
- Lack capabilities to sub-select and focus on specific parts

Do not enable programmatic analysis of the data by the end user



```
//! Low level hook to QMP_global_sum
inline void globalSumArray(float *dest, int len)
{
  QMP_sum_float_array(dest, len);
}

//! Low level hook to QMP_global_sum
inline void globalSumArray(double *dest, int len)
{
  QMP_sum_double_array(dest, len);
}

//! Global sum on a multi1d
template<class T>
inline void globalSumArray(multi1d<T>& dest)
{
  // The implementation here is relying on the structure being packed
```

Scopes	# samples (I)	# samples (E)
main	5.80e05 97.2%	
void Chroma::doHMC<Chroma::HMCTrjParams>(QDP::multi1d<QDF	5.70e05 95.6%	
loop at hmc.cc: 311-435	5.65e05 95.0%	
Chroma::AbsHMCTrj<QDP::multi1d<QDP::OLattice<QDP::PScala	5.65e05 95.0%	
Chroma::LatColMatSTSLeapfrogRecursiveIntegrator::operator()(	5.25e05 88.2%	
loop at lcm_sts_leapfrog_recursive.cc: 129-131	4.85e05 81.5%	
Chroma::LatColMatExpSdtIntegrator::operator()(Chroma::A	4.25e05 71.4%	
loop at lcm_exp_sdt.cc: 85-88	4.25e05 71.4%	
Chroma::LCMMDIntegratorSteps::leapP(QDP::multi1d<	4.25e05 71.4%	
Chroma::TwoFlavorExactWilsonTypeFermMonomial<	3.30e05 55.5%	
Chroma::TwoFlavorExactWilsonTypeFermMonomia	2.30e05 38.7%	
Chroma::MdagMSysSolverCG<QDP::OLattice<QI	2.20e05 37.0%	
loop at syssolver_mdagm_cg.h: 56-70	2.20e05 37.0%	
Chroma::SystemSolverResults_t Chroma::Inv	2.20e05 37.0%	
Chroma::SystemSolverResults_t Chroma::I	2.20e05 37.0%	1.00e04 1.7%
loop at invcg2.cc: 147-182	1.85e05 31.1%	5.00e03 0.8%
Chroma::EvenOddPrecWilsonLinOp::o	1.05e05 17.6%	1.00e04 1.7%
Chroma::EvenOddPrecWilsonLinOp::o	7.00e04 11.8%	1.50e04 2.5%
[I] globalSumArray	5.00e03 0.8%	
[I] vaxpy3	5.00e03 0.8%	5.00e03 0.8%
[I] local_sumsq		

hpcviewer's GUI

Hatchet

- A Python-based library to enable programmatic analysis
- Creates an in-memory representation of the graph
- Leverage pandas which supports multi-dimensional tabular datasets
  - Use graph as structured index to index pandas dataframes
- A set of operators to sub-select and/or aggregate profile data
- A set of operators to compare multiple datasets

Pandas and dataframes

Pandas and dataframes

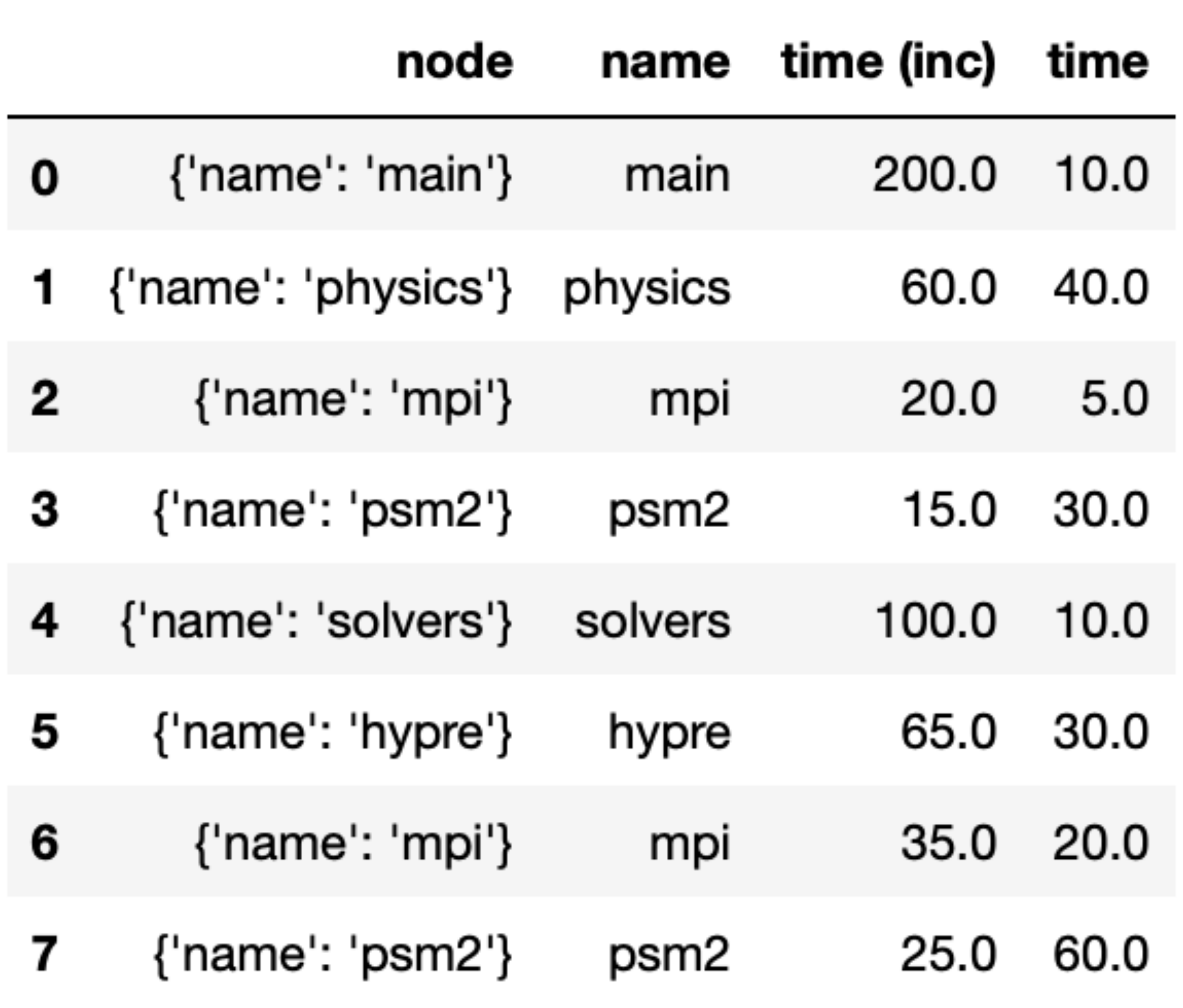
- Pandas is an open-source Python library for data analysis

Pandas and dataframes

- Pandas is an open-source Python library for data analysis
- Dataframe: two-dimensional tabular data structure
  - Supports many operations borrowed from SQL databases

Columns

Rows



	node	name	time (inc)	time
0	{'name': 'main'}	main	200.0	10.0
1	{'name': 'physics'}	physics	60.0	40.0
2	{'name': 'mpi'}	mpi	20.0	5.0
3	{'name': 'psm2'}	psm2	15.0	30.0
4	{'name': 'solvers'}	solvers	100.0	10.0
5	{'name': 'hypre'}	hypre	65.0	30.0
6	{'name': 'mpi'}	mpi	35.0	20.0
7	{'name': 'psm2'}	psm2	25.0	60.0

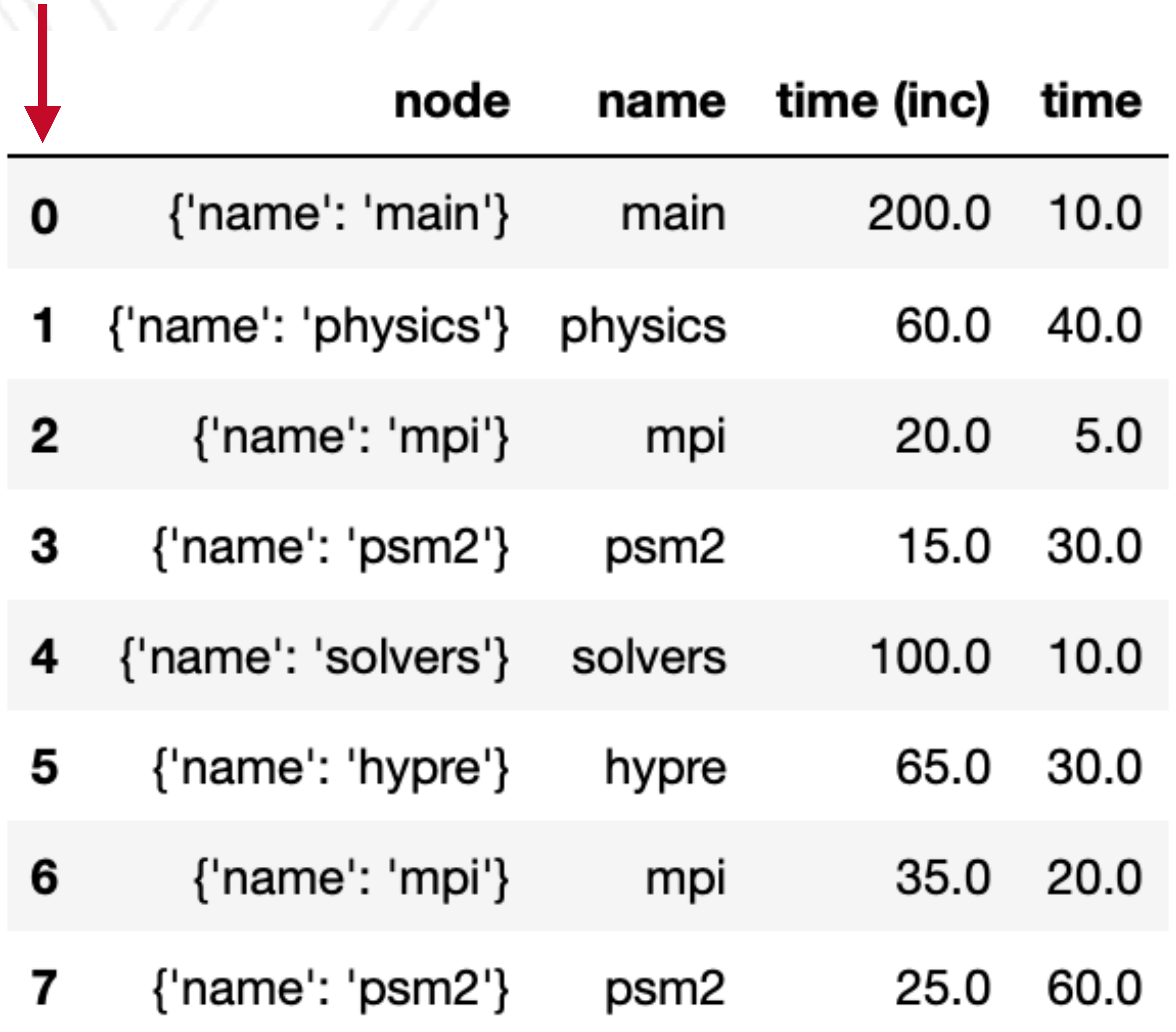
Pandas and dataframes

- Pandas is an open-source Python library for data analysis
- Dataframe: two-dimensional tabular data structure
    - Supports many operations borrowed from SQL databases

Index

Columns

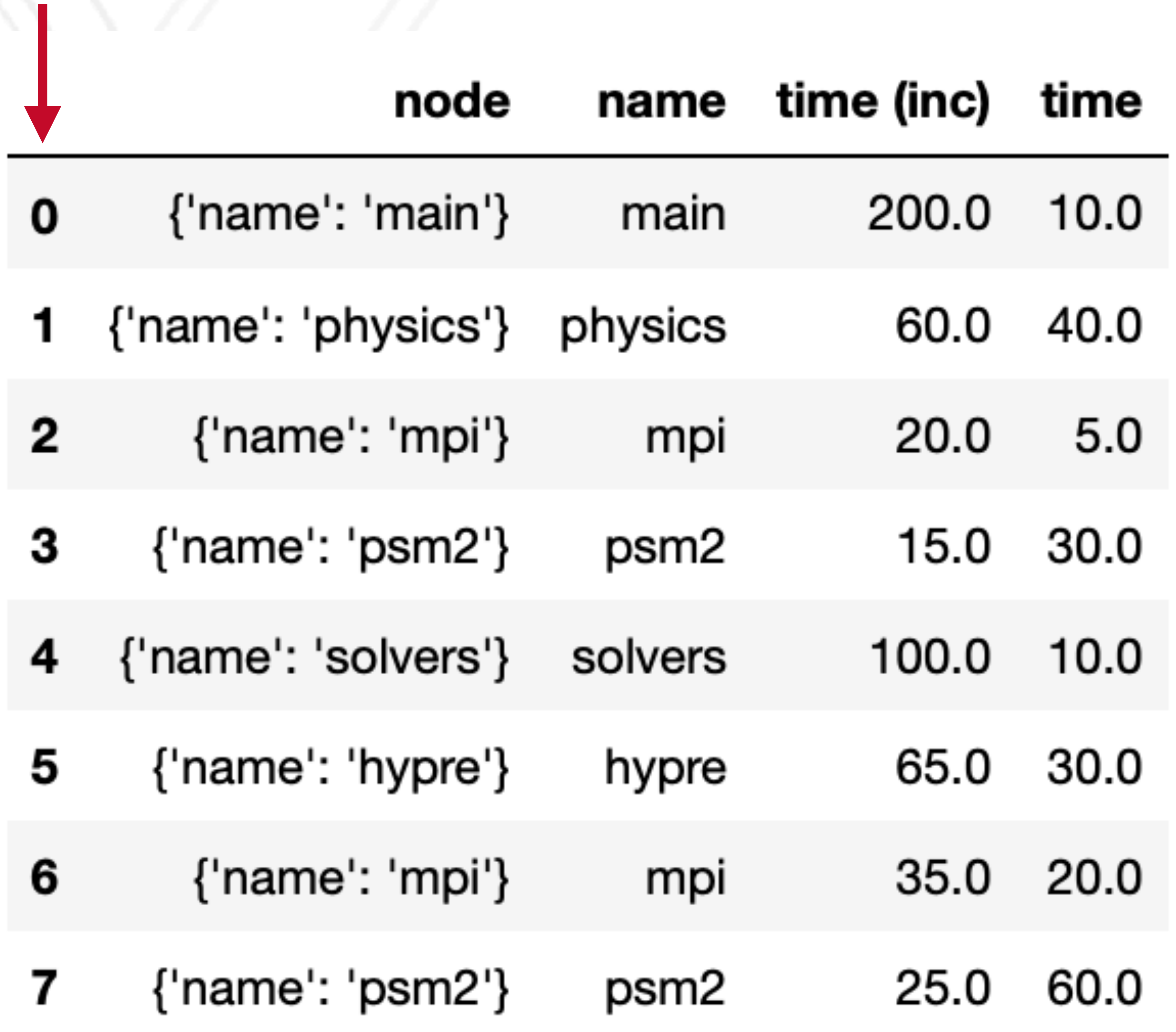
Rows



	node	name	time (inc)	time
0	{'name': 'main'}	main	200.0	10.0
1	{'name': 'physics'}	physics	60.0	40.0
2	{'name': 'mpi'}	mpi	20.0	5.0
3	{'name': 'psm2'}	psm2	15.0	30.0
4	{'name': 'solvers'}	solvers	100.0	10.0
5	{'name': 'hypre'}	hypre	65.0	30.0
6	{'name': 'mpi'}	mpi	35.0	20.0
7	{'name': 'psm2'}	psm2	25.0	60.0

Pandas and dataframes

- Pandas is an open-source Python library for data analysis
- Dataframe: two-dimensional tabular data structure
  - Supports many operations borrowed from SQL databases
- MultiIndex enables working with high-dimensional data in a 2D data structure



Index

Columns

Rows

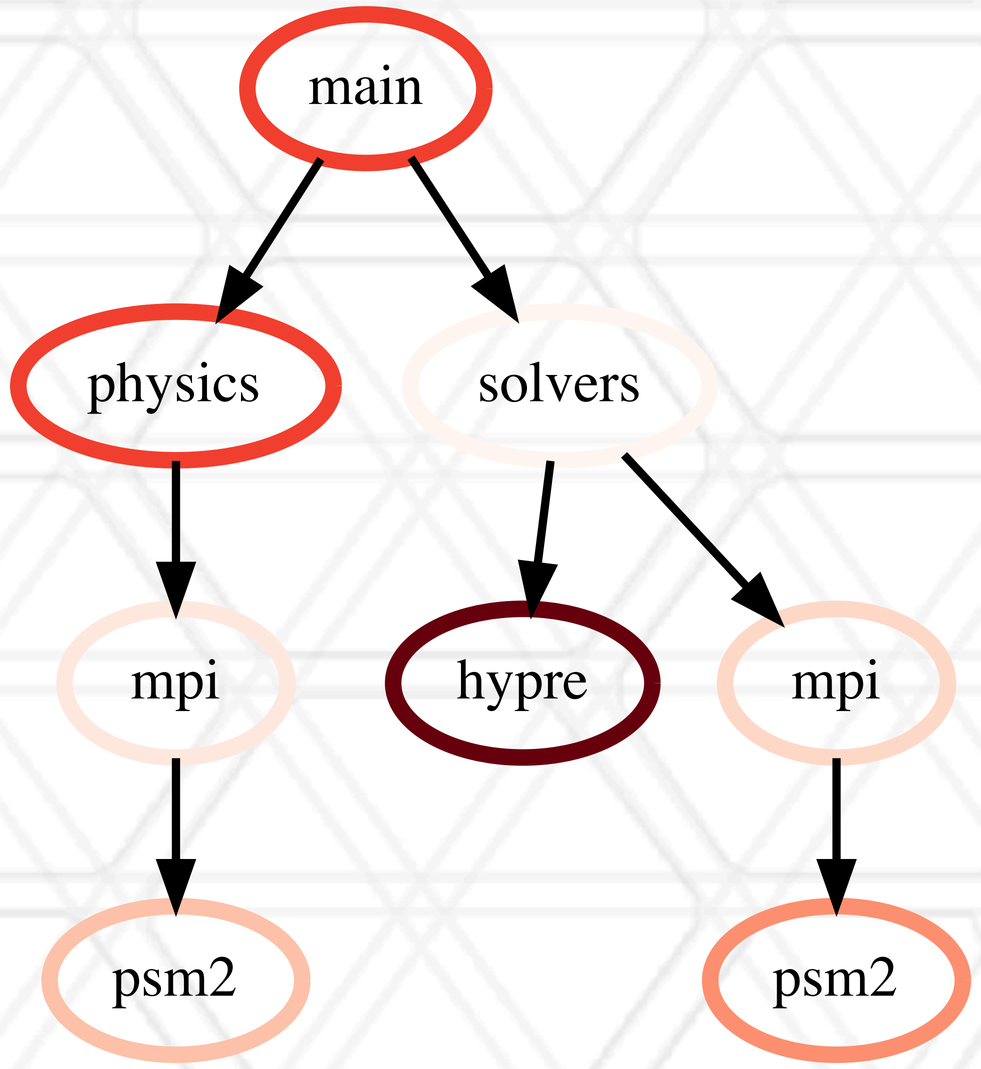
	node	name	time (inc)	time
0	{'name': 'main'}	main	200.0	10.0
1	{'name': 'physics'}	physics	60.0	40.0
2	{'name': 'mpi'}	mpi	20.0	5.0
3	{'name': 'psm2'}	psm2	15.0	30.0
4	{'name': 'solvers'}	solvers	100.0	10.0
5	{'name': 'hypre'}	hypre	65.0	30.0
6	{'name': 'mpi'}	mpi	35.0	20.0
7	{'name': 'psm2'}	psm2	25.0	60.0

Central data structure: a *GraphFrame*

- Consists of a structured index graph object and a pandas dataframe
- Graph stores caller-callee relationships
- Dataframe stores all numerical and categorical data

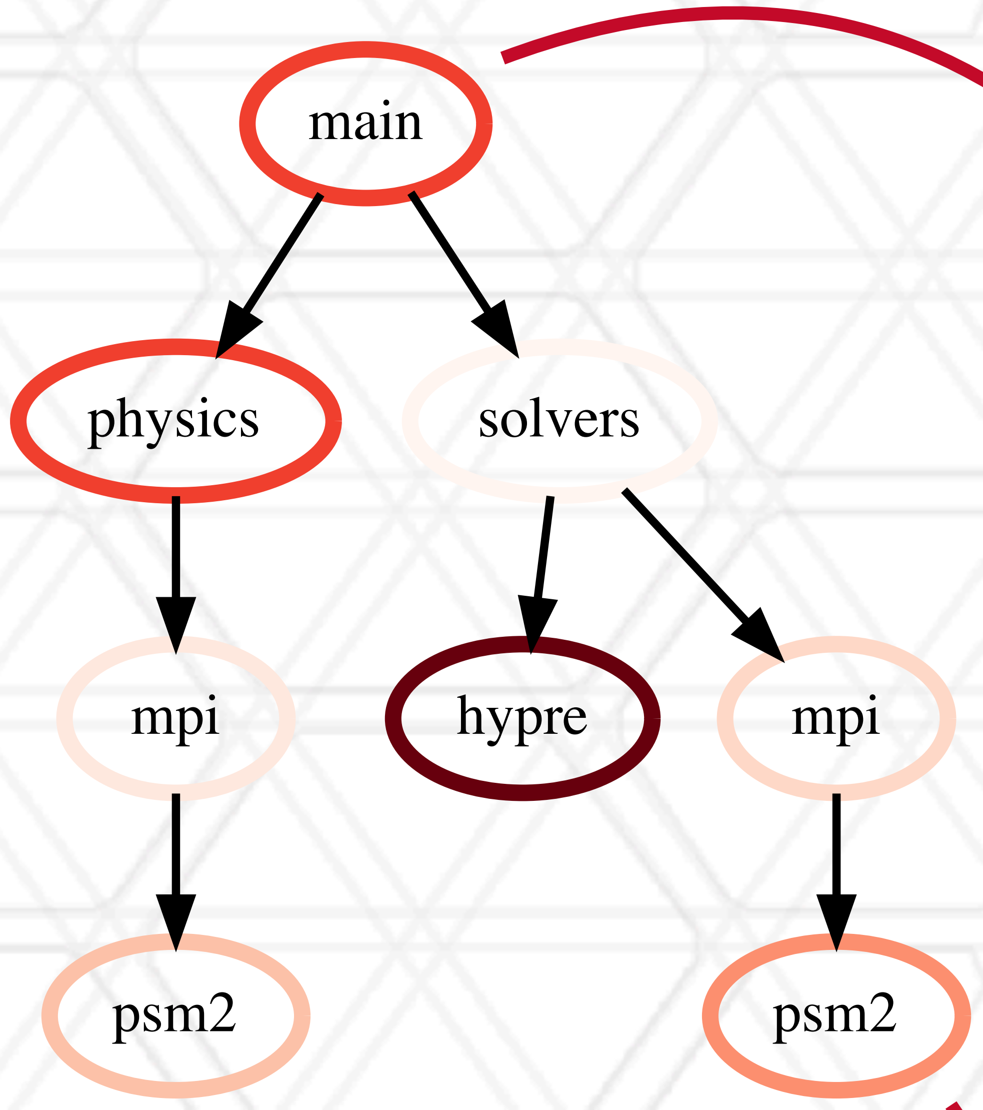
Central data structure: a *GraphFrame*

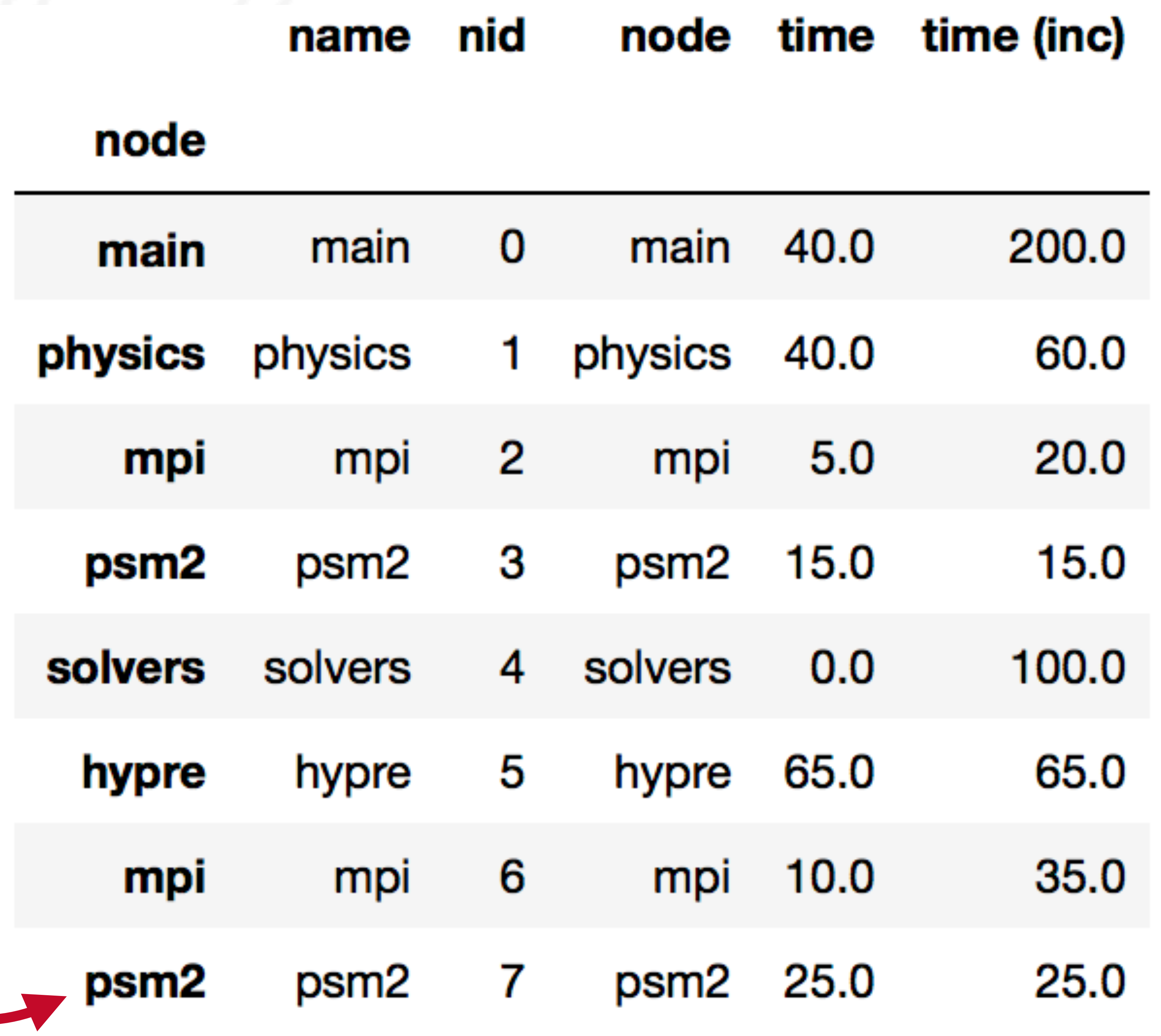
- Consists of a structured index graph object and a pandas dataframe
- Graph stores caller-callee relationships
- Dataframe stores all numerical and categorical data



Central data structure: a *GraphFrame*

- Consists of a structured index graph object and a pandas dataframe
- Graph stores caller-callee relationships
- Dataframe stores all numerical and categorical data





	name	nid	node	time	time (inc)
node					
main	main	0	main	40.0	200.0
physics	physics	1	physics	40.0	60.0
mpi	mpi	2	mpi	5.0	20.0
psm2	psm2	3	psm2	15.0	15.0
solvers	solvers	4	solvers	0.0	100.0
hypre	hypre	5	hypre	65.0	65.0
mpi	mpi	6	mpi	10.0	35.0
psm2	psm2	7	psm2	25.0	25.0

Questions?

UNIVERSITY OF MARYLAND

Abhinav Bhatele

5218 Brendan Iribe Center (IRB) / College Park, MD 20742

phone: 301.405.4507 / e-mail: bhatele@cs.umd.edu