OAK RIDGE National Laboratory

Performance Analysis with Scalasca part II

George S. Markomanolis

8 August 2019

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

MiniWeather MPI+OpenACC

- Compilations
- MPI+OpenACC: make PREP="scorep --mpp=mpi --openacc –pdt" openacc
- Cube4 does not show OpenACC information

LSMS compilation - Issues

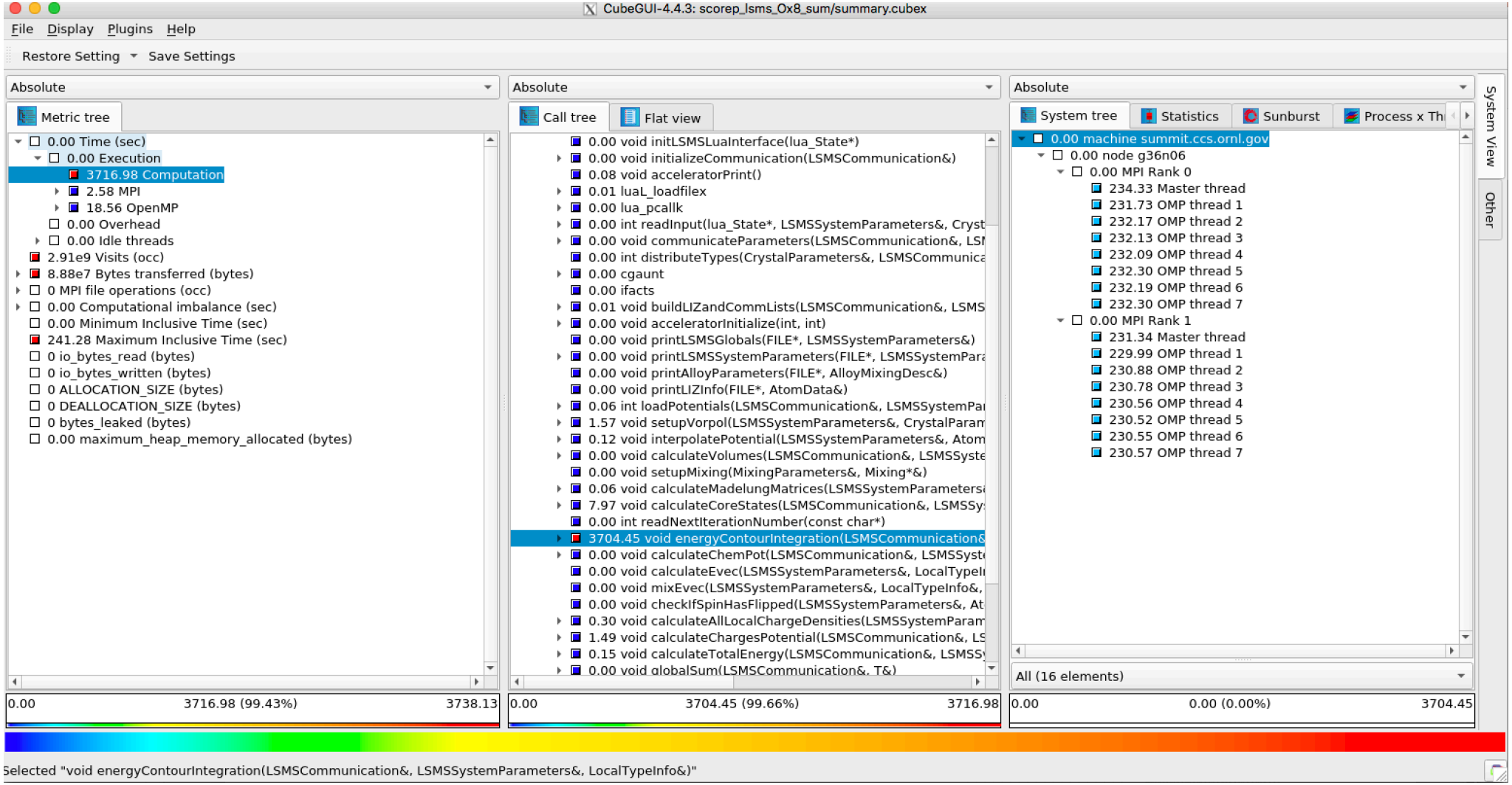
- scorep --cuda nvcc… /gpfs/alpine/gen110/scratch/gmarkoma/scorep/LSMS3/Source/LSMS_3/src/Accelerator/buildKKRMatrix_kernels.cu(242): **error: identifier "omp_get_thread_num" is undefined**
- Adding the option --keep-files, does not delete the temporary files, so we can observe what is wrong

LSMS compilation – Issues (cont.)

Original .cu file	Temporary opari.cu file during instrumentation
#ifdef _OPENMP #include <omp.h> #else inline int omp_get_max_threads() {return 1;} inline int omp_get_num_threads() {return 1;} inline int omp_get_thread_num() {return 0;} #endif	#ifdef _OPENMP #else inline int omp_get_max_threads() {return 1;} inline int omp_get_num_threads() {return 1;} inline int omp_get_thread_num() {return 0;} #endif

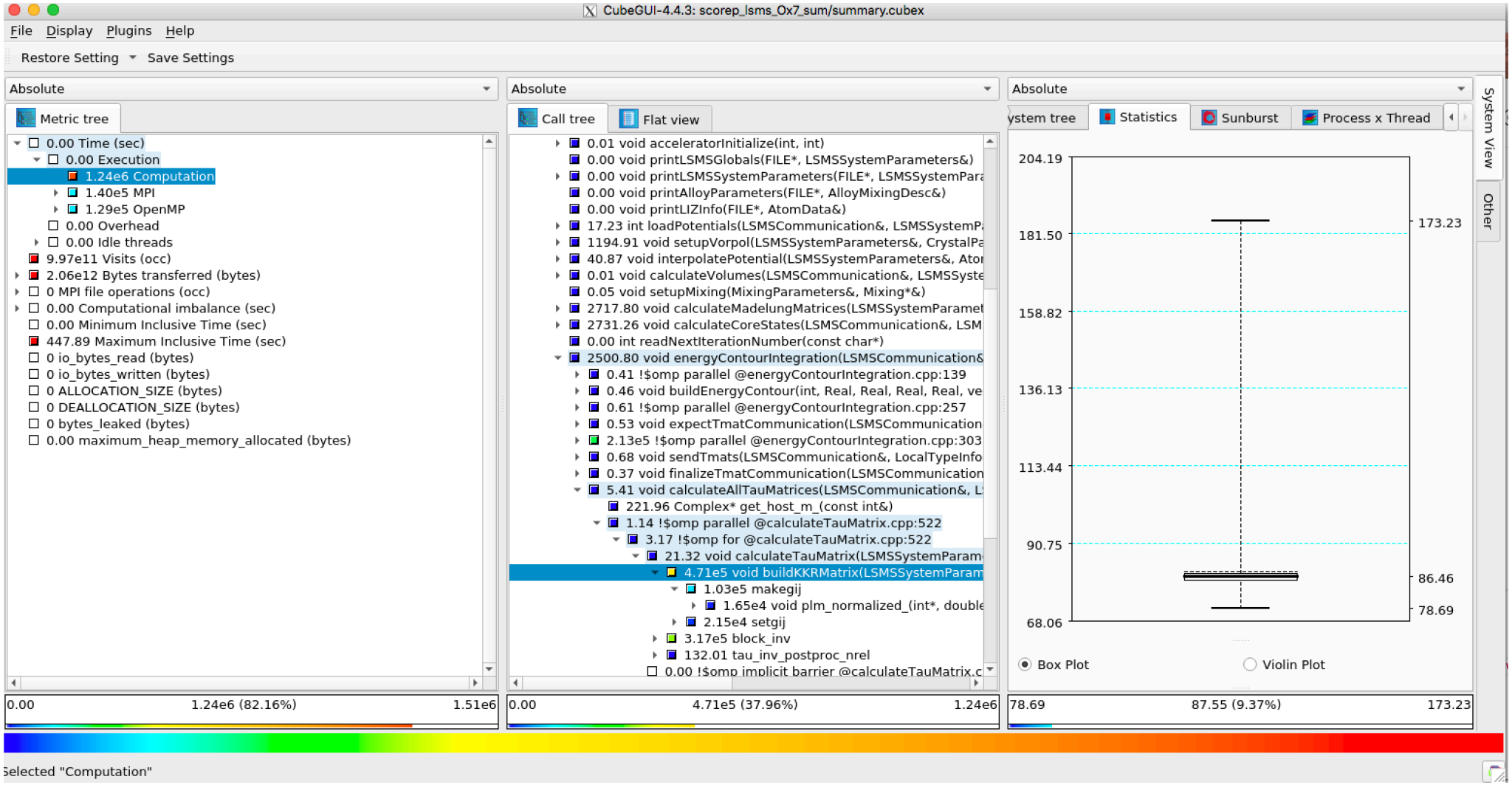
Adding manually the #include <omp.h> to the temporary file, it solves the issue

LSMS – Profiling one compute node

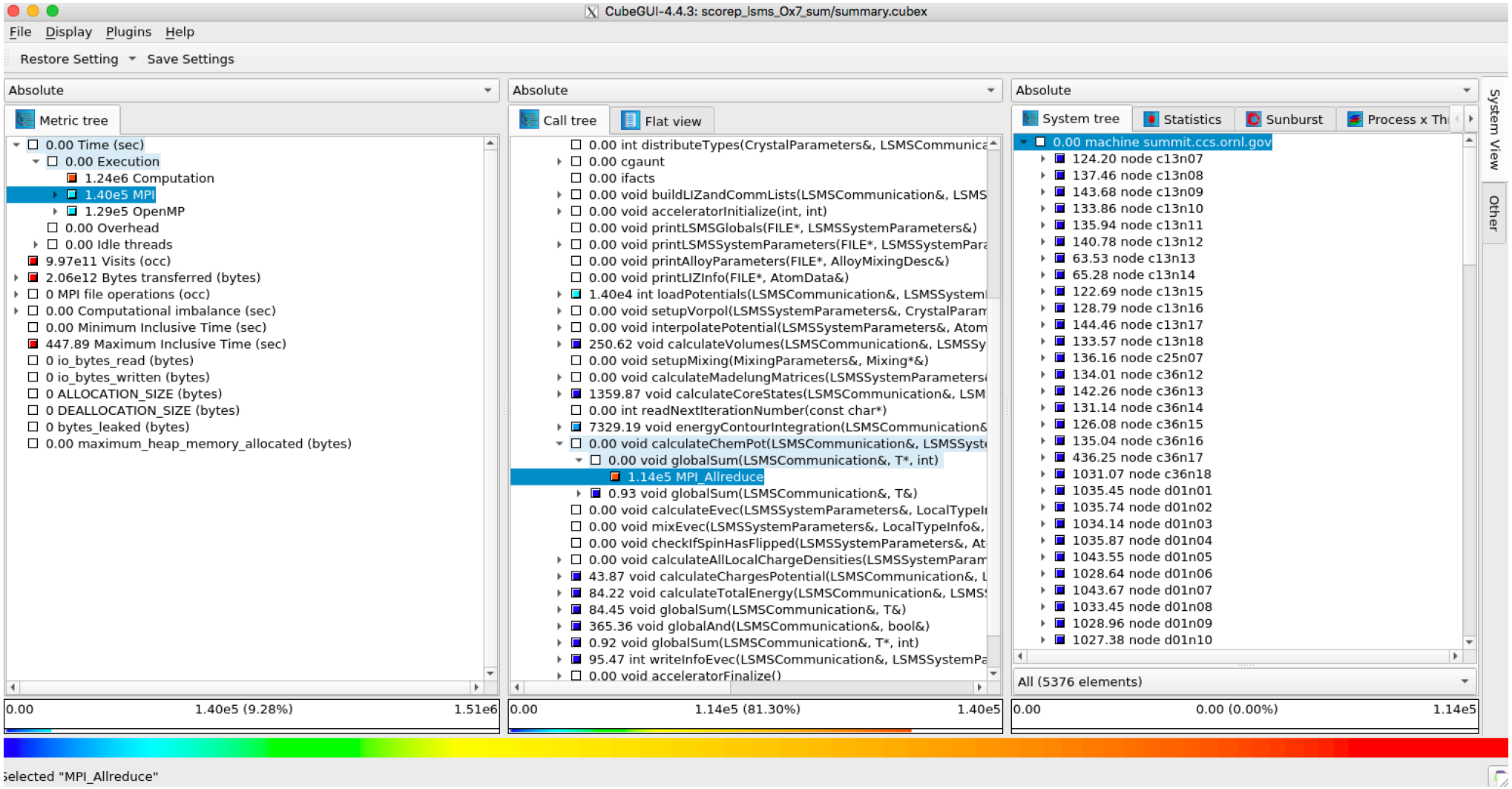


OAK RIDGE National Laboratory

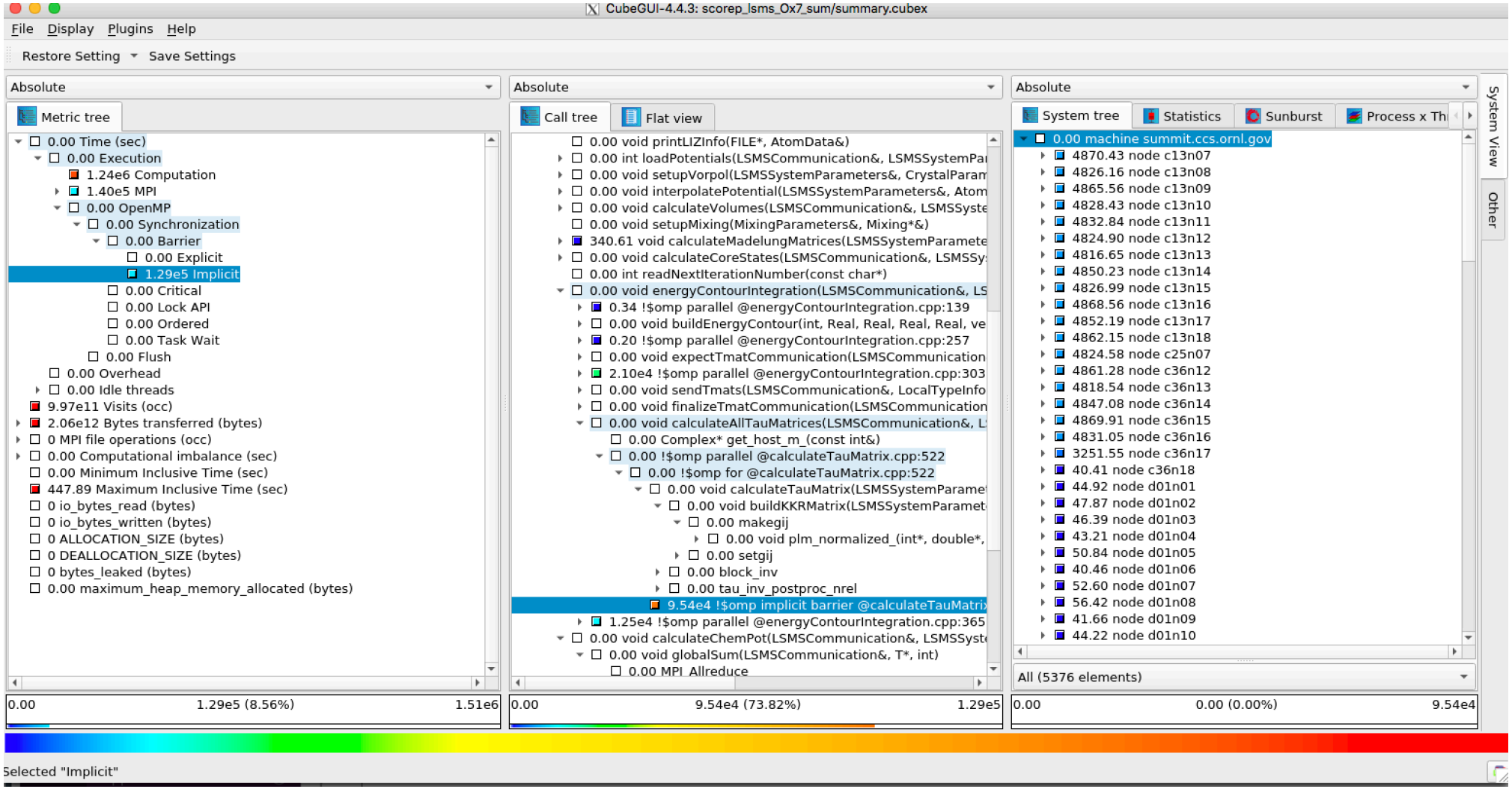
LSMS on 128 compute nodes – Compute variation



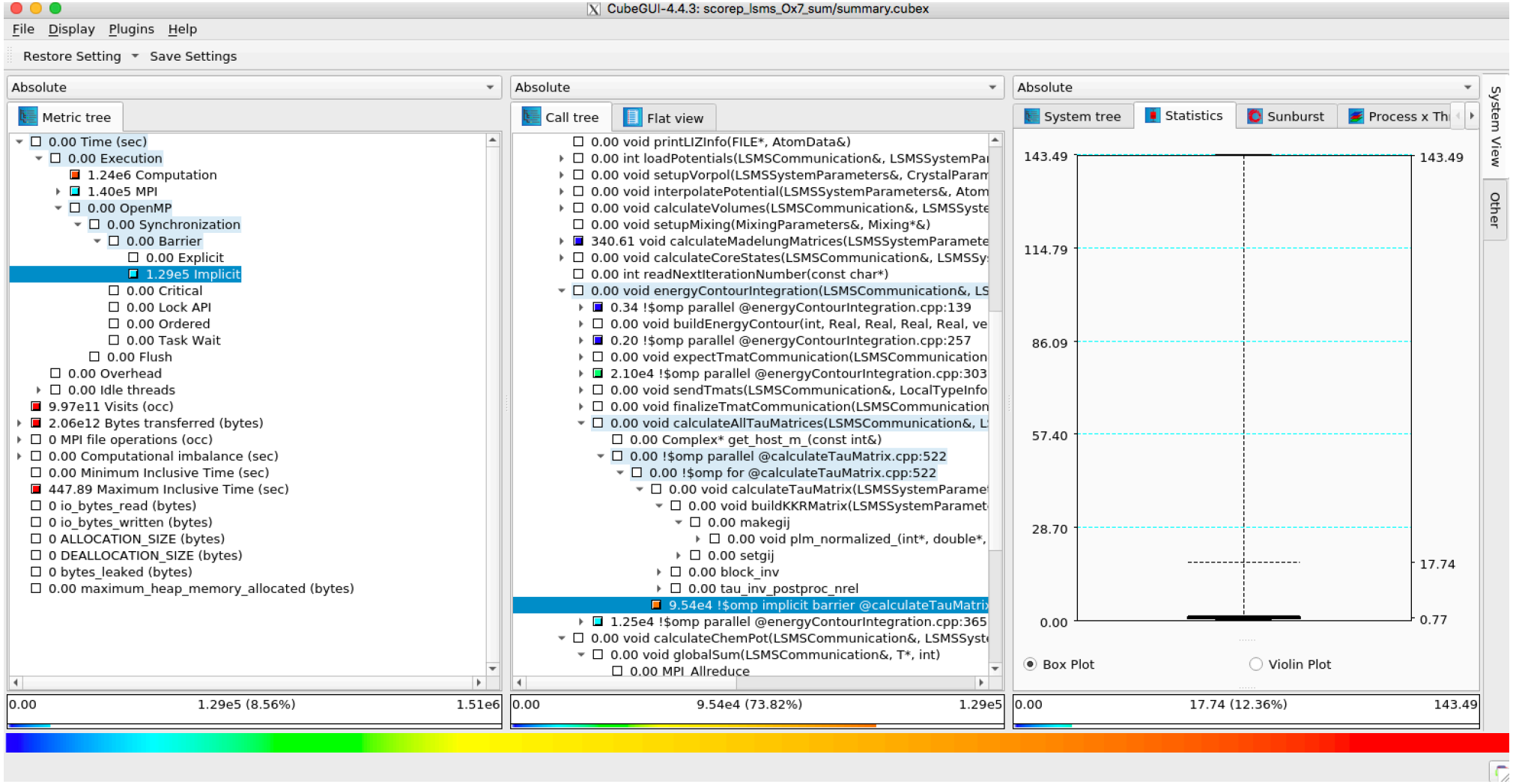
LSMS on 128 compute nodes – MPI_Allreduce variation



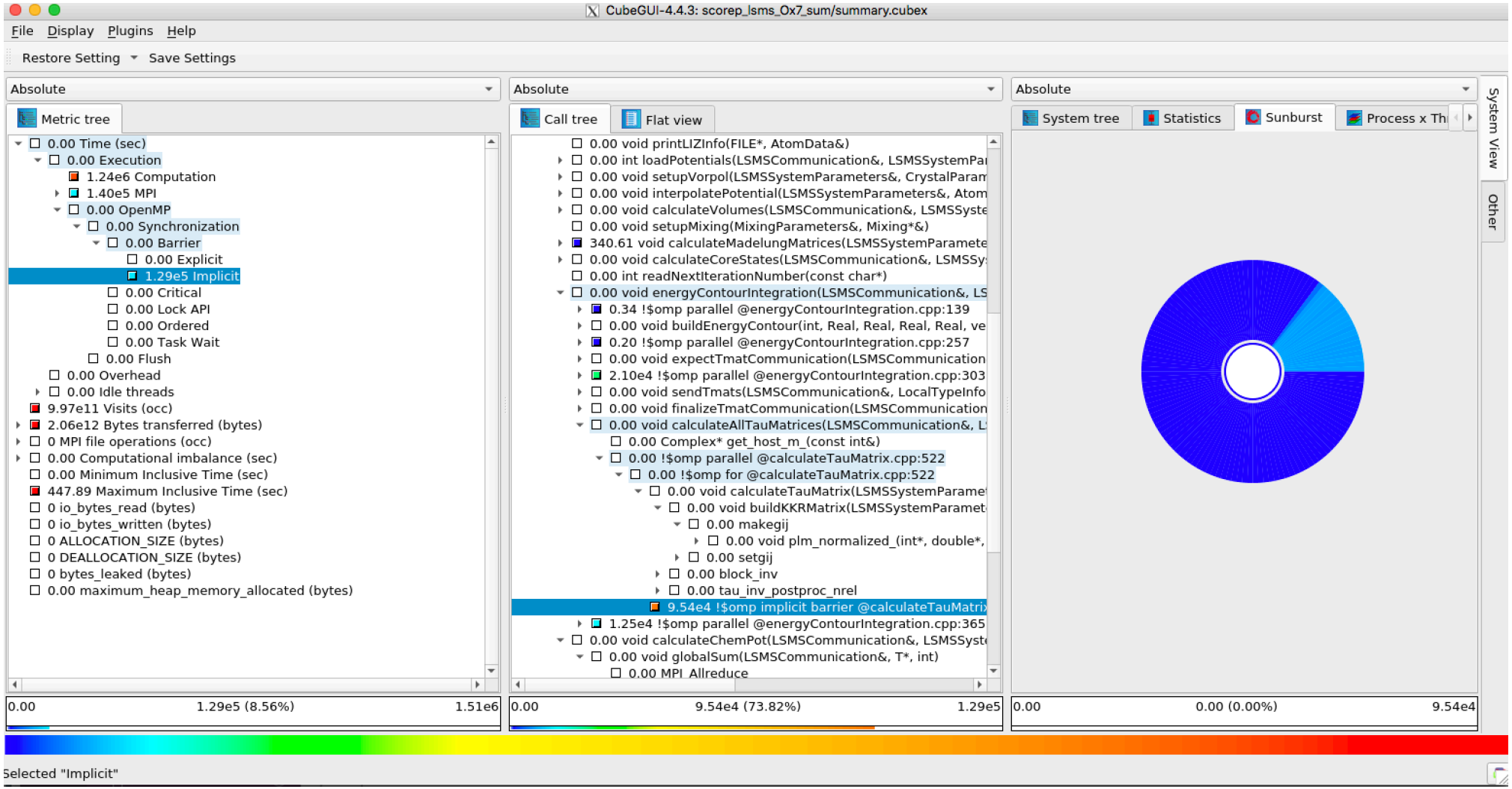
LSMS on 128 compute nodes – OpenMP Barrier



LSMS on 128 compute nodes – OpenMP Barrier



LSMS on 128 compute nodes – OpenMP Barrier - Sunburst



LSMS – Buffer for tracing

- scalasca -examine -s /gpfs/alpine/…/scorep_lsms_Ox8_sum/
  INFO: Score report written to /gpfs/…/scorep_lsms_Ox8_sum/scorep.score
  vim /gpfs/…/scorep_lsms_Ox8_sum/scorep.score

Estimated aggregate size of event trace: **24TB**

Estimated requirements for largest trace buffer (max_buf): 36GB

Estimated memory requirements (SCOREP_TOTAL_MEMORY): **36GB**

(warning: The memory requirements cannot be satisfied by Score-P to avoid intermediate flushes when tracing. Set SCOREP_TOTAL_MEMORY=4G to get the maximum supported memory or reduce requirements using USR regions filters.)

flt	type	max_buf[B]	visits	time[s]	time[%]	time/visit[us]	region
	ALL	37,758,243,091	2,906,934,475	3738.13	100.0	1.29	ALL
	USR	**37,757,438,730**	2,906,899,463	3714.45	99.4	1.28	USR
	OMP	757,272	32,688	18.59	0.5	568.66	OMP
	MPI	29,168	932	2.58	0.1	2772.73	MPI
	COM	17,880	1,390	2.50	0.1	1798.55	COM
	SCOREP	41	2	0.00	0.0	242.20	SCOREP
	USR	22,486,328,384	1,729,717,541	260.77	7.0	0.15	dfv_m
	USR	9,921,071,888	763,159,376	126.99	3.4	0.17	fit
	USR	1,064,960,000	81,920,000	343.03	9.2	4.19	void bulirsch_stoer_integrator
	USR	846,102,582	65,083,854	19.65	0.5	0.30	void plm_normalized_(int*, double*, double*)

Selective instrumentation

- File exclude.filt, we select the USR calls with high volume of max_buf and the ones with many calls:

```
SCOREP_REGION_NAMES_BEGIN EXCLUDE
dfv_m
fit
*bulirsch_stoer_integrator_*
*plm_normalized_*
...
SCOREP_REGION_NAMES_END
```

- Apply on the profiling data:

```
scalasca -examine -s -f exclude.filt scorep_lsms_Ox7_sum/
```

Selective instrumentation (cont.)

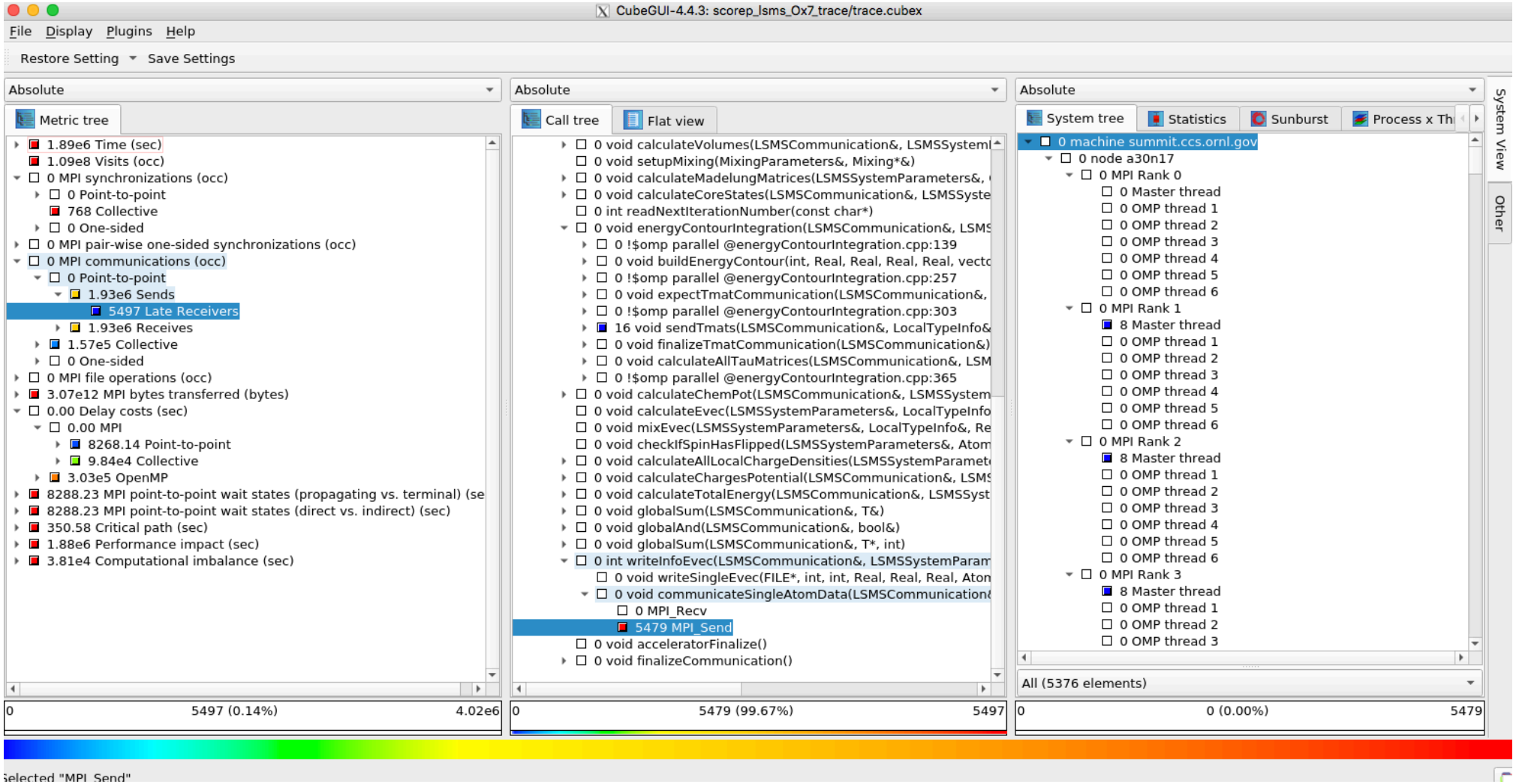
- vim scorep_lsms_Ox7_sum/scorep.score_exclude.filt

Estimated aggregate size of event trace: **3100MB**
Estimated requirements for largest trace buffer (max_buf): 6MB
Estimated memory requirements (SCOREP_TOTAL_MEMORY): **20MB**
(hint: When tracing set SCOREP_TOTAL_MEMORY=20MB to avoid intermediate flushes or reduce requirements using USR regions filters.)

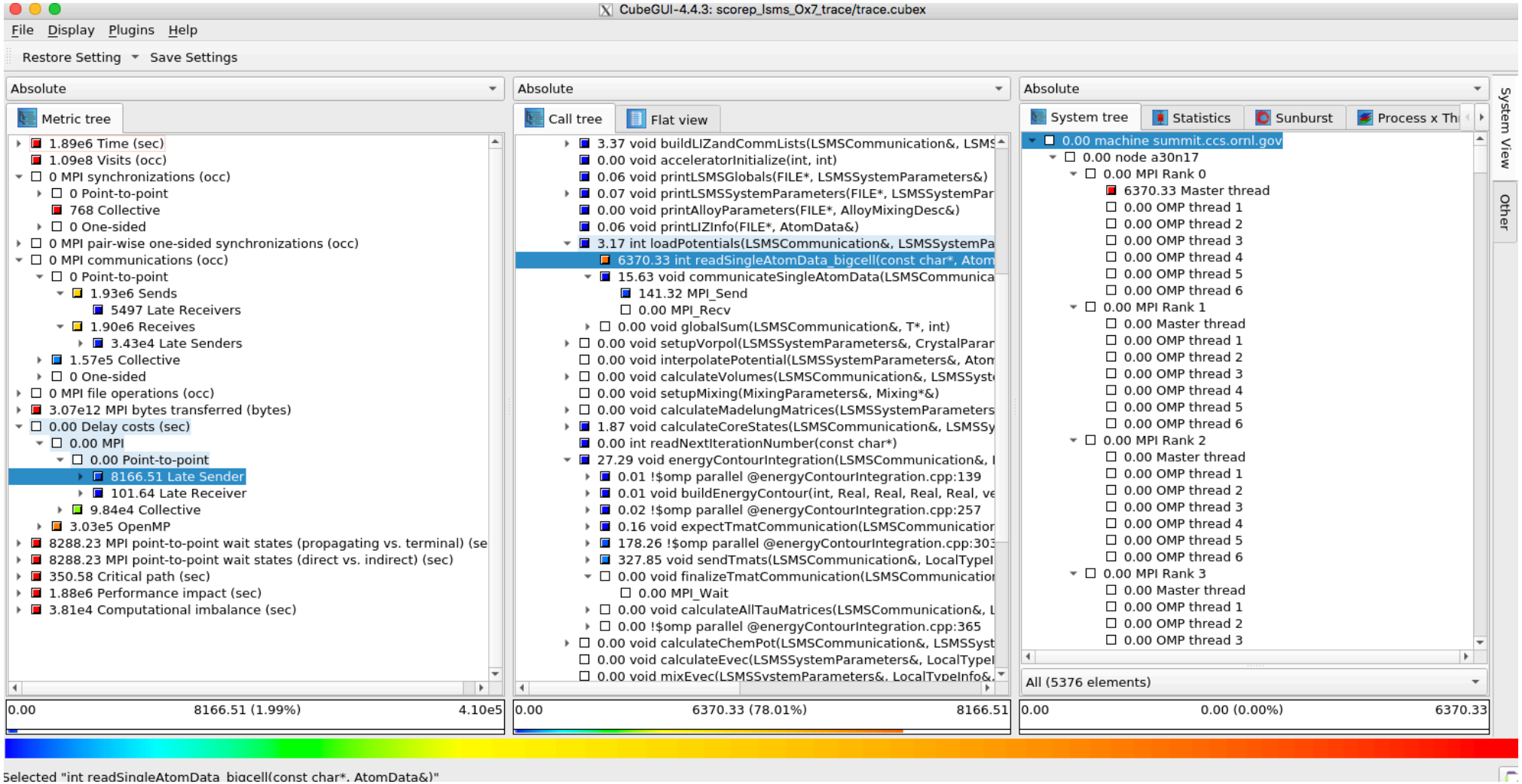
flt	type	max_buf[B]	visits	time[s]	time[%]	time/visit[us]	region
-	ALL	37,768,557,978	997,234,022,247	1508955.54	100.0	1.51	ALL
-	USR	37,767,383,966	997,216,546,439	1237070.04	82.0	1.24	USR
-	MPI	1,160,452	5,948,480	139997.90	9.3	23535.07	MPI
-	OMP	662,613	10,983,168	129188.73	8.6	11762.43	OMP
-	COM	302,424	543,392	2698.66	0.2	4966.33	COM
-	SCOREP	41	768	0.21	0.0	270.75	SCOREP
*	ALL	5,851,181	108,920,629	1061962.47	70.4	9749.87	ALL-FLT
+	FLT	37,763,922,579	997,125,101,618	446993.07	29.6	0.45	FLT
*	USR	3,725,651	91,444,821	790076.96	52.4	8639.93	USR-FLT
-	MPI	1,160,452	5,948,480	139997.90	9.3	23535.07	MPI-FLT
-	OMP	662,613	10,983,168	129188.73	8.6	11762.43	OMP-FLT
*	COM	302,424	543,392	2698.66	0.2	4966.33	COM-FLT
-	SCOREP	41	768	0.21	0.0	270.75	SCOREP-FLT

Execute: scalasca -analyze -q -t -f /full_path/exclude.filt jsrun …

LSMS on 128 compute nodes – Late Receivers (occ)

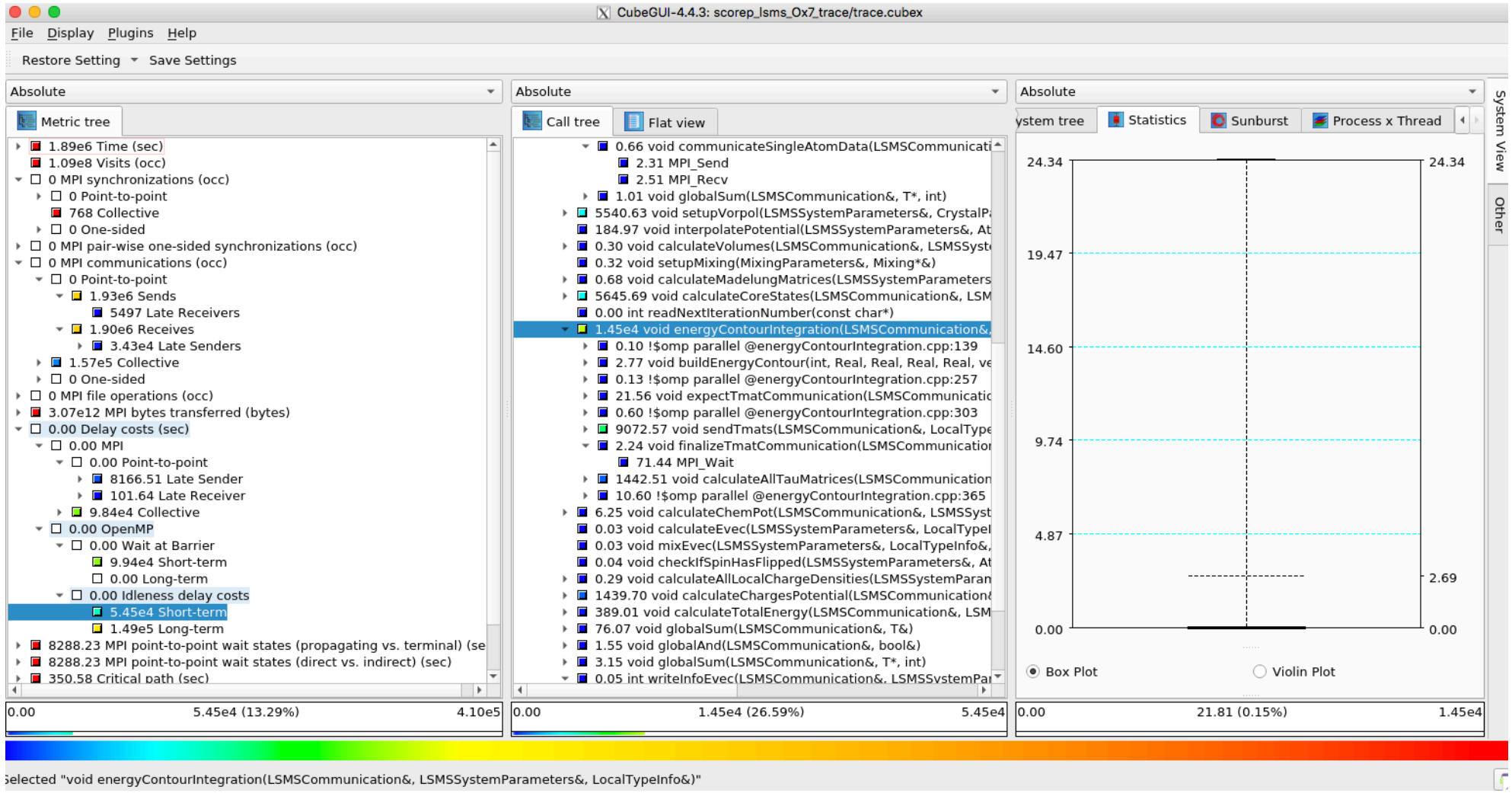


LSMS on 128 compute nodes – Late Sender (sec.)



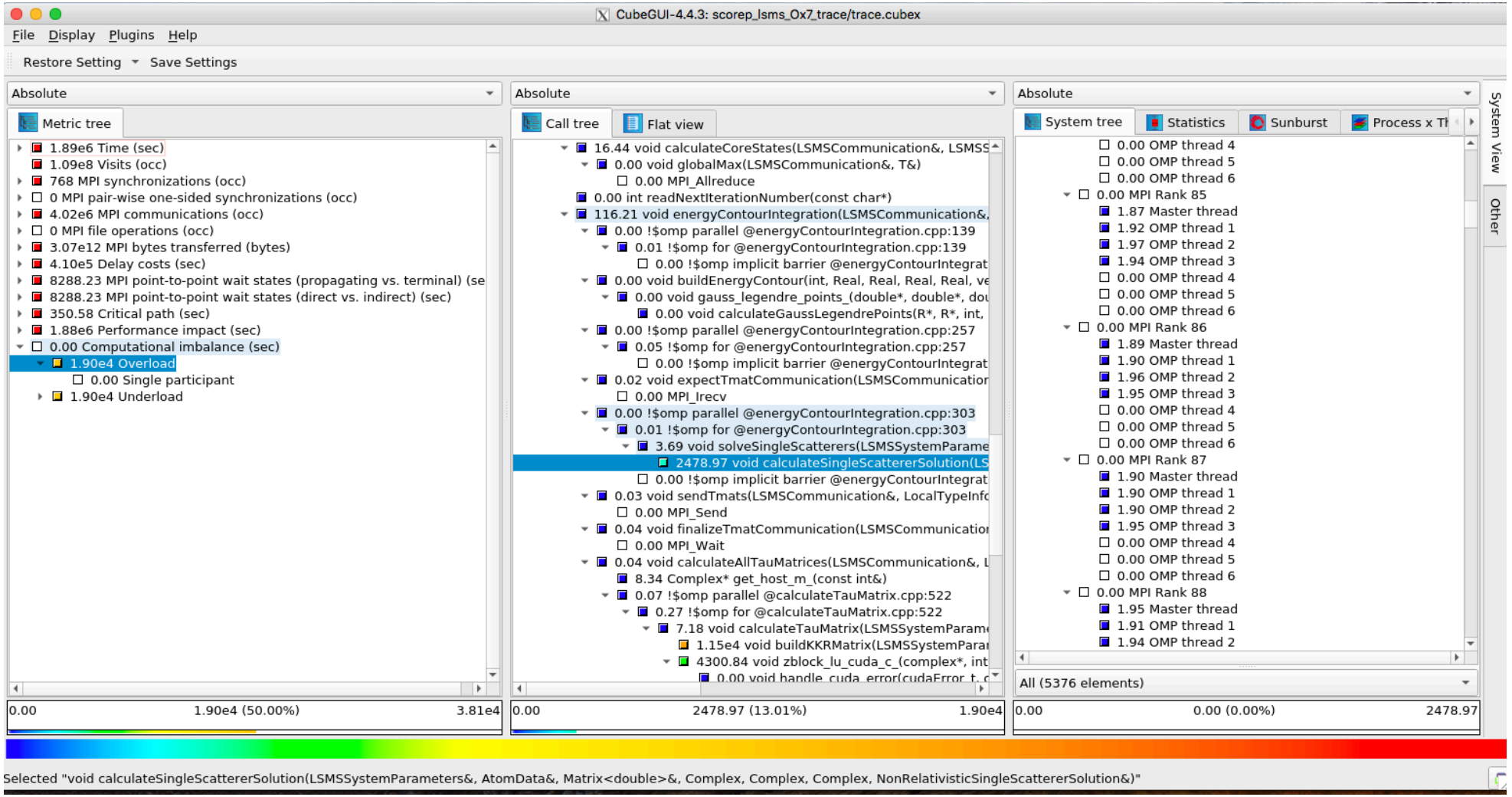
OAK RIDGE National Laboratory

LSMS on 128 compute nodes – Short-term Idleness delay



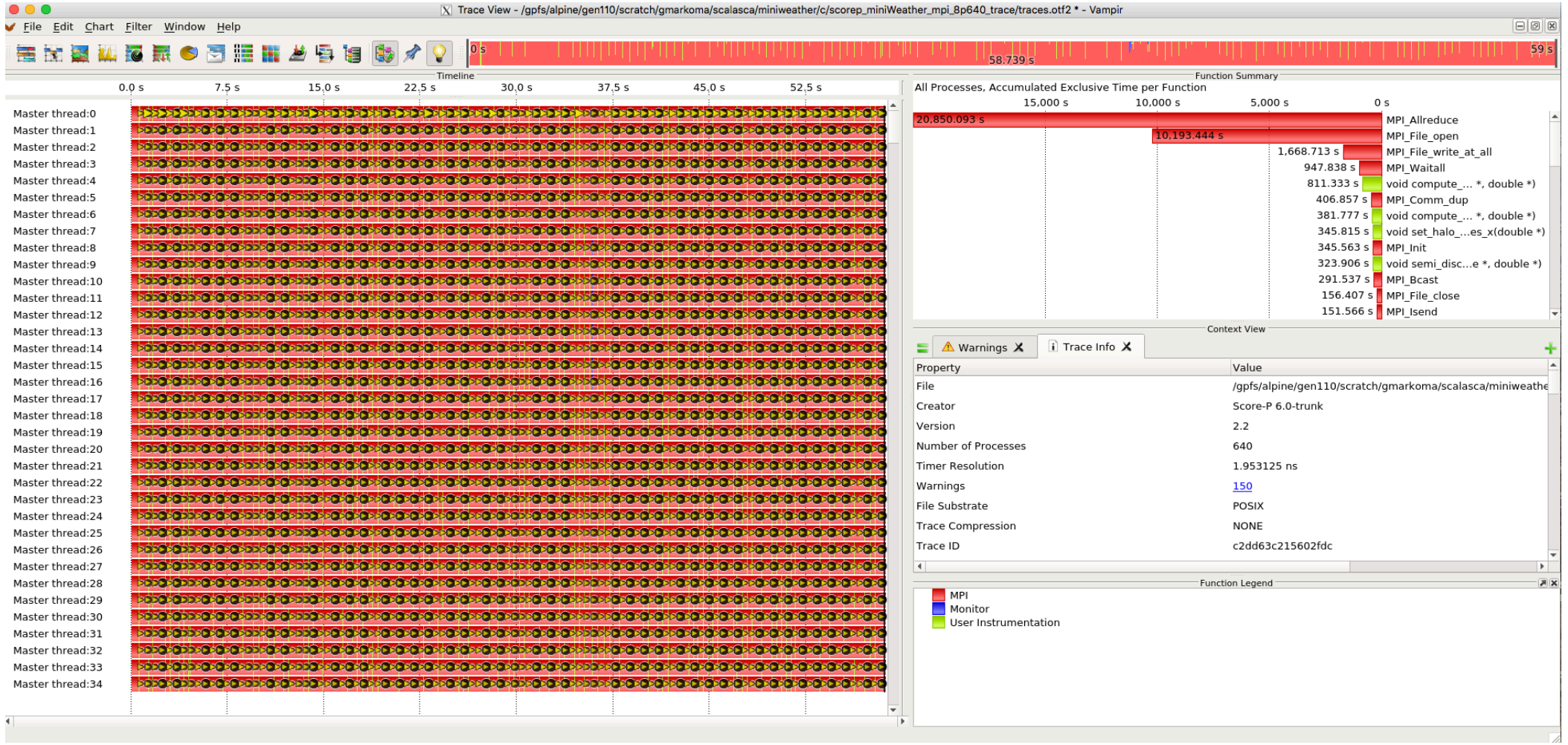
OAK RIDGE National Laboratory

LSMS on 128 compute nodes – Short-term Idleness delay



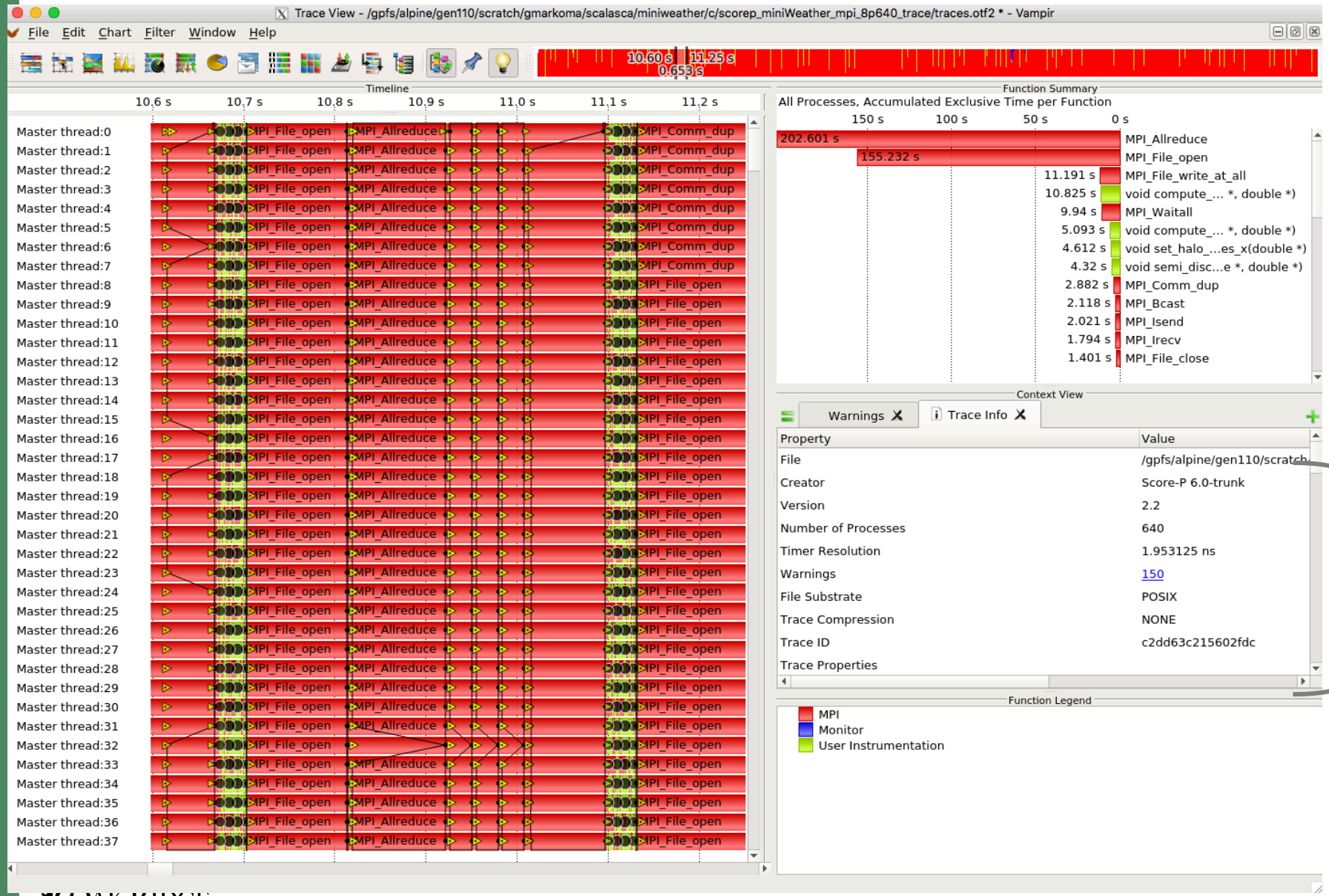
OAK RIDGE National Laboratory

Vampir with OTF2 trace

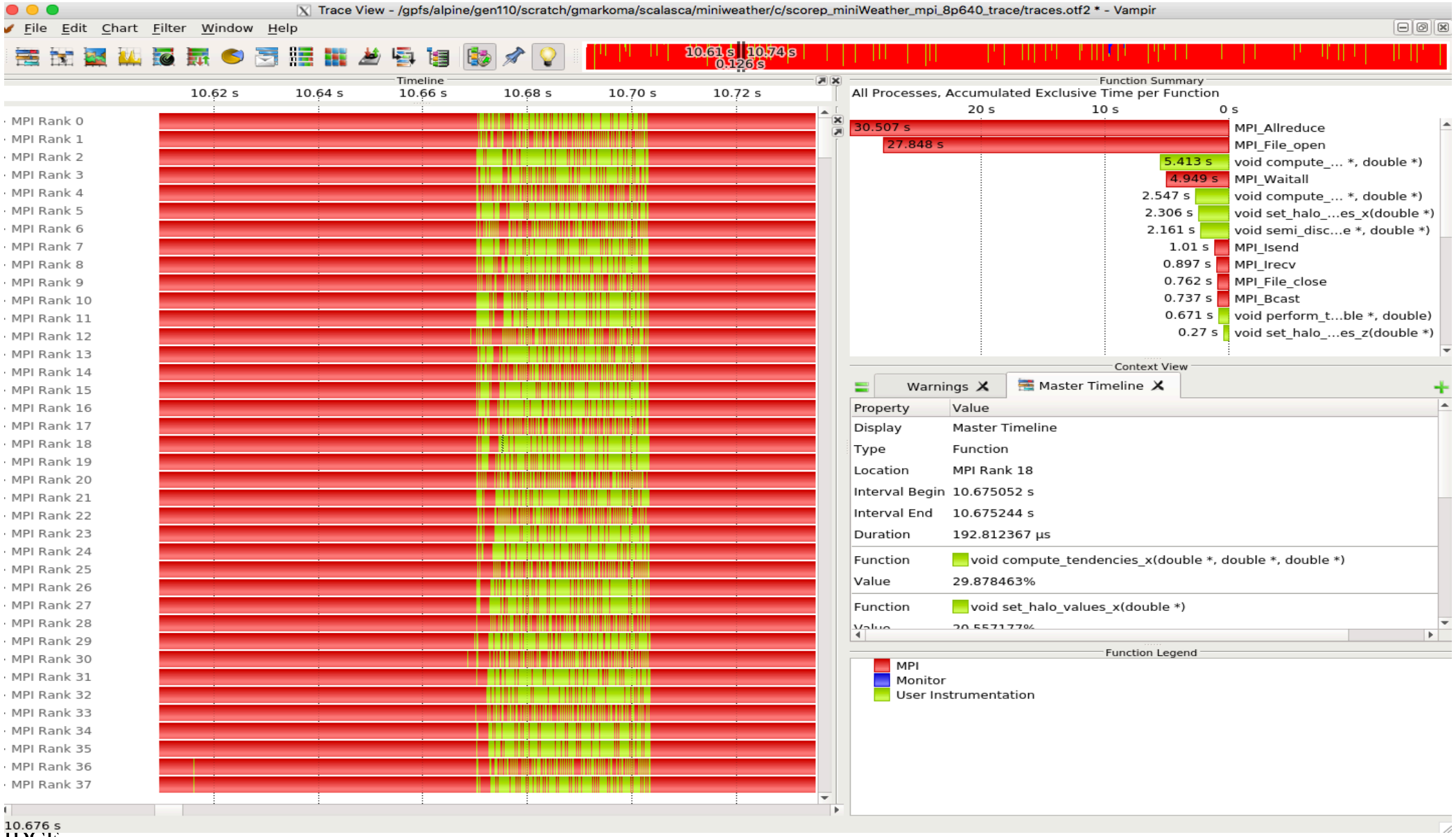


OTF2 trace is created inside the scorep folder after the scalasca –analyze, example of 640 MPI processes

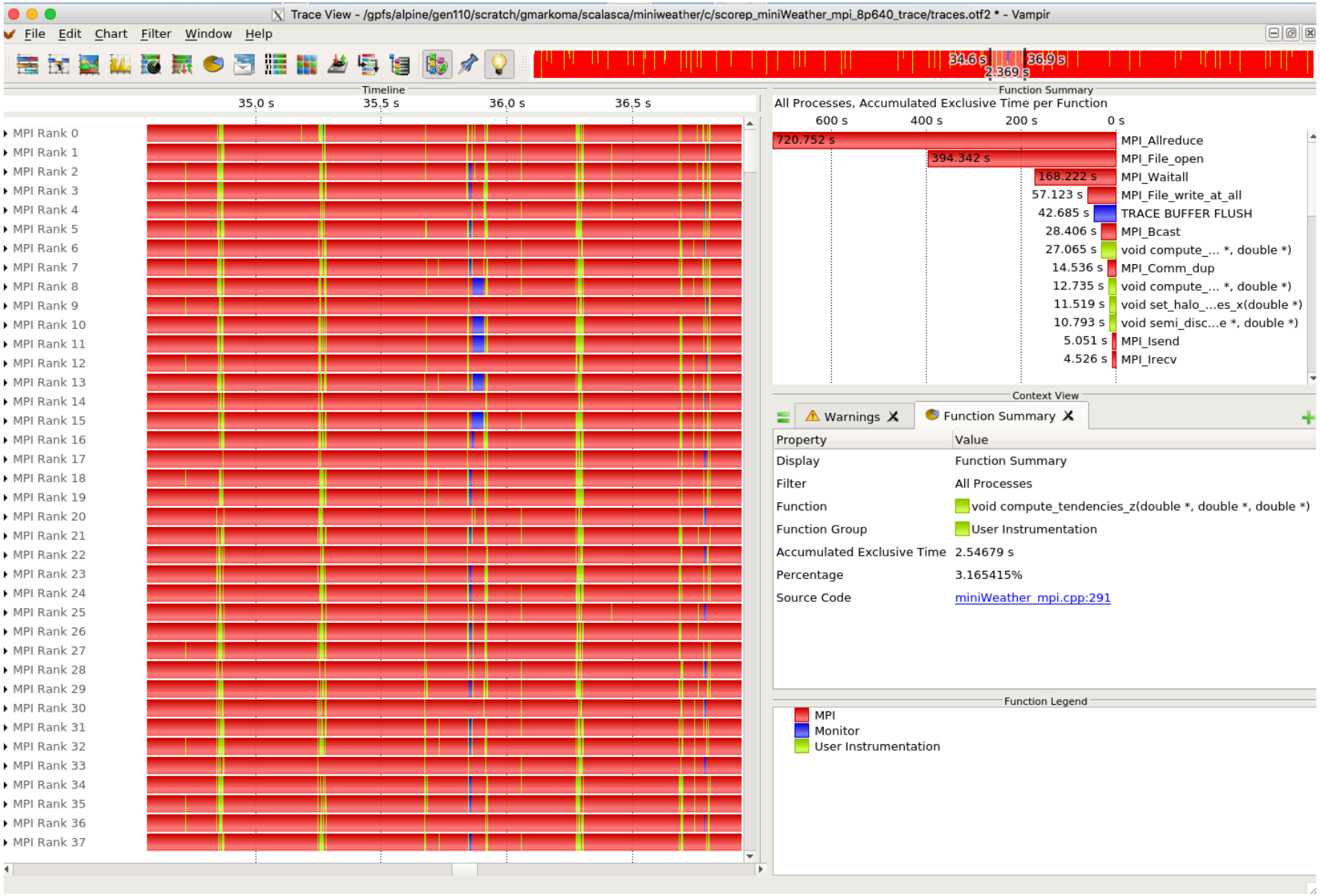
Vampir with OTF2 trace



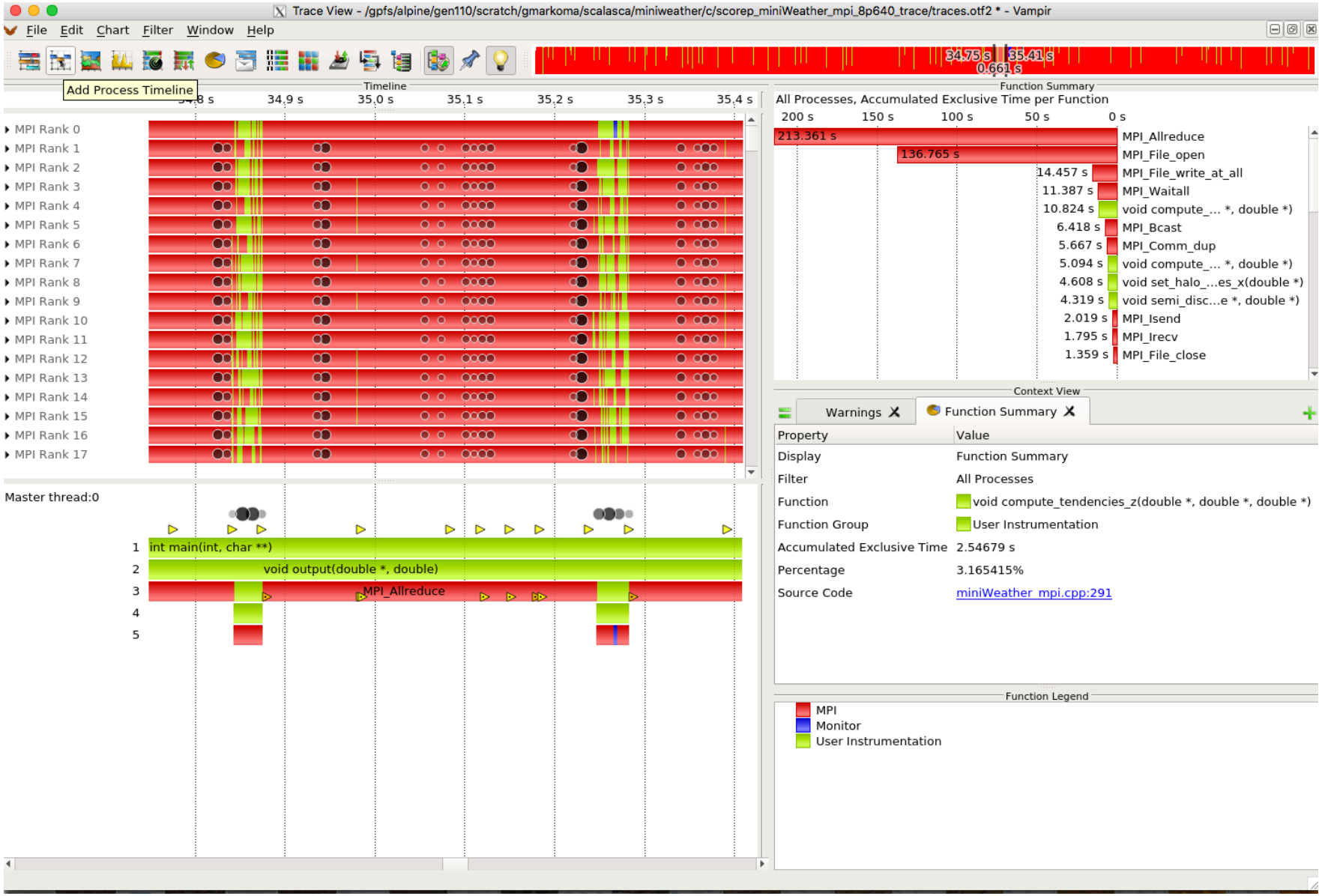
Vampir with OTF2 trace



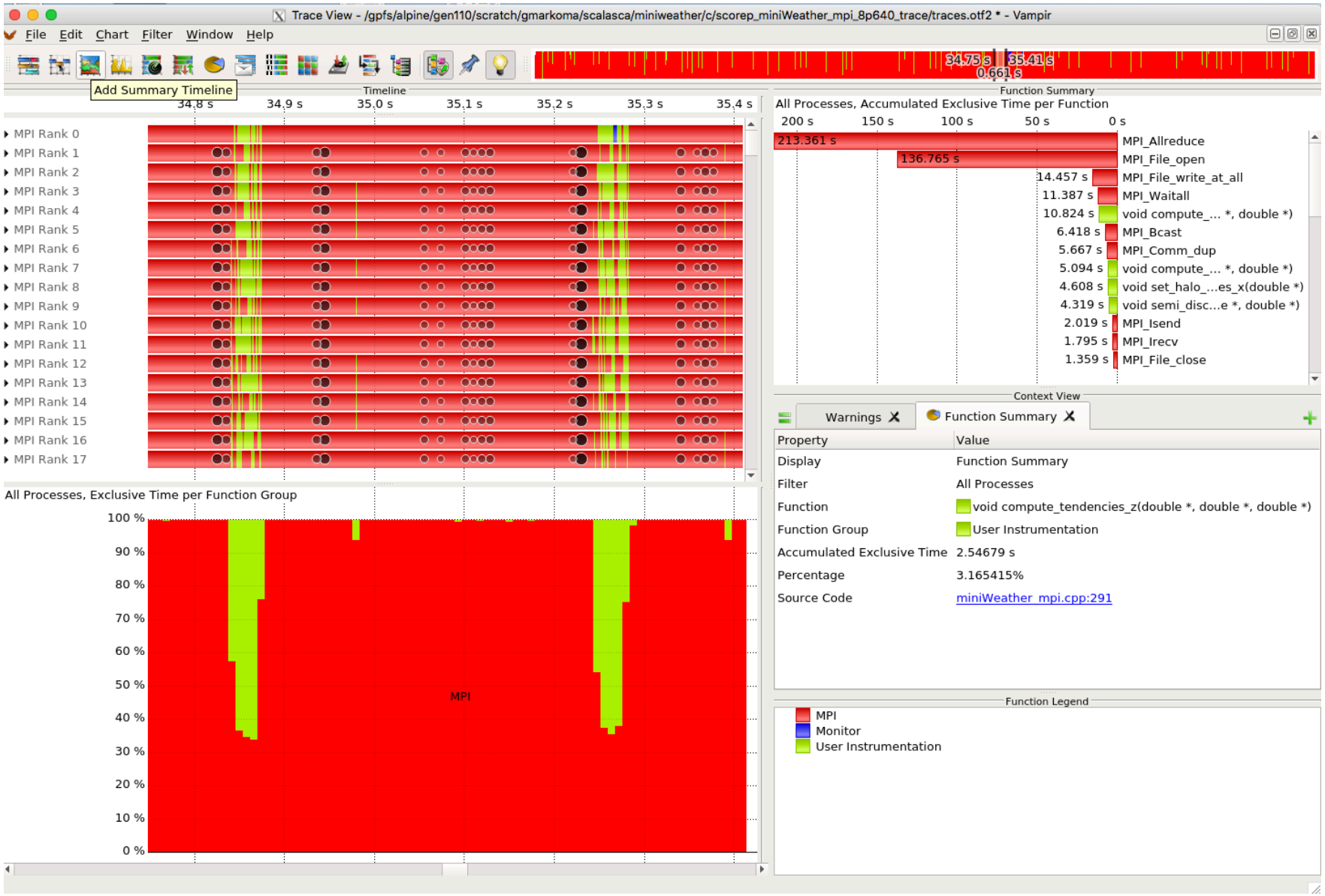
Vampir with OTF2 trace



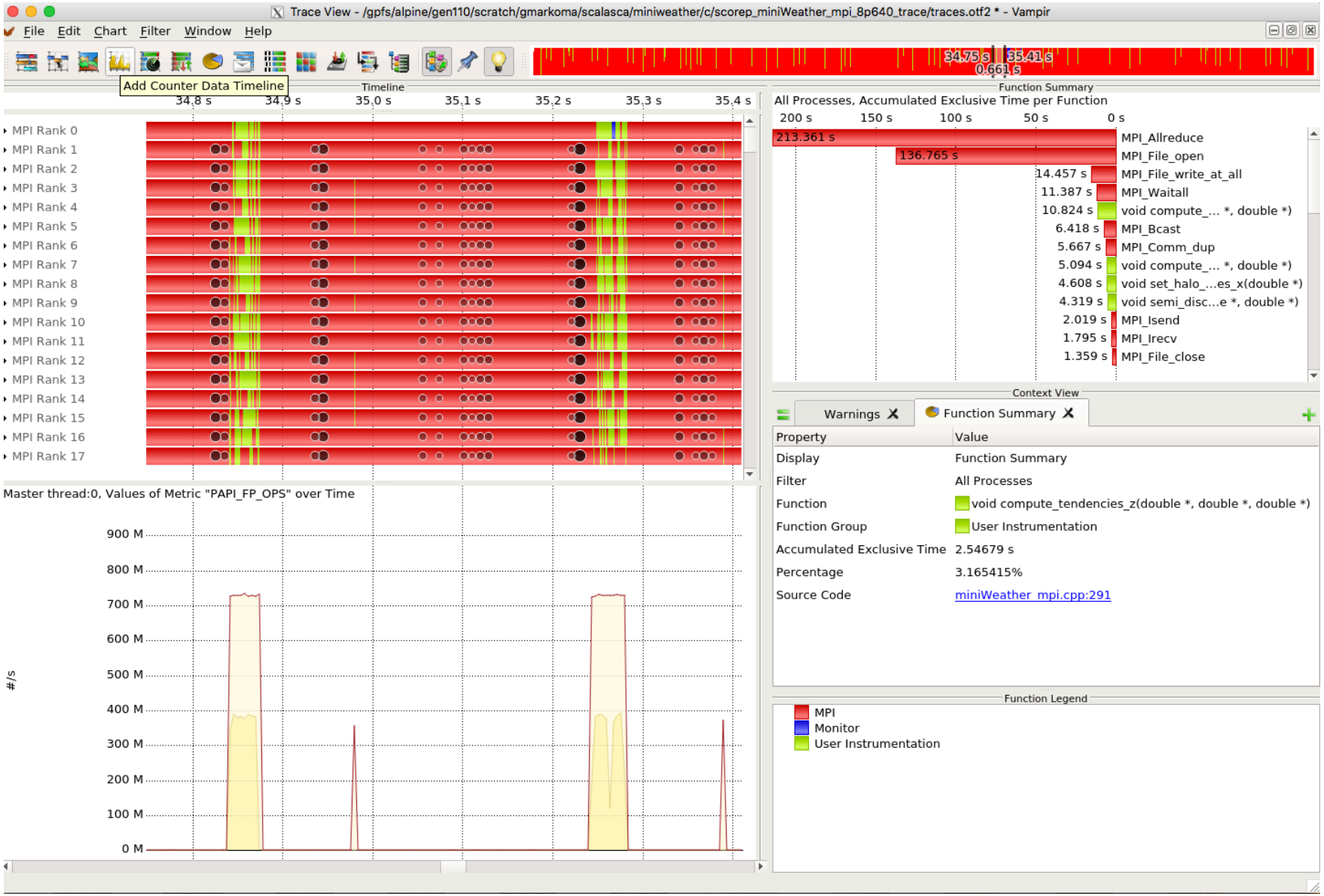
Vampir with OTF2 trace



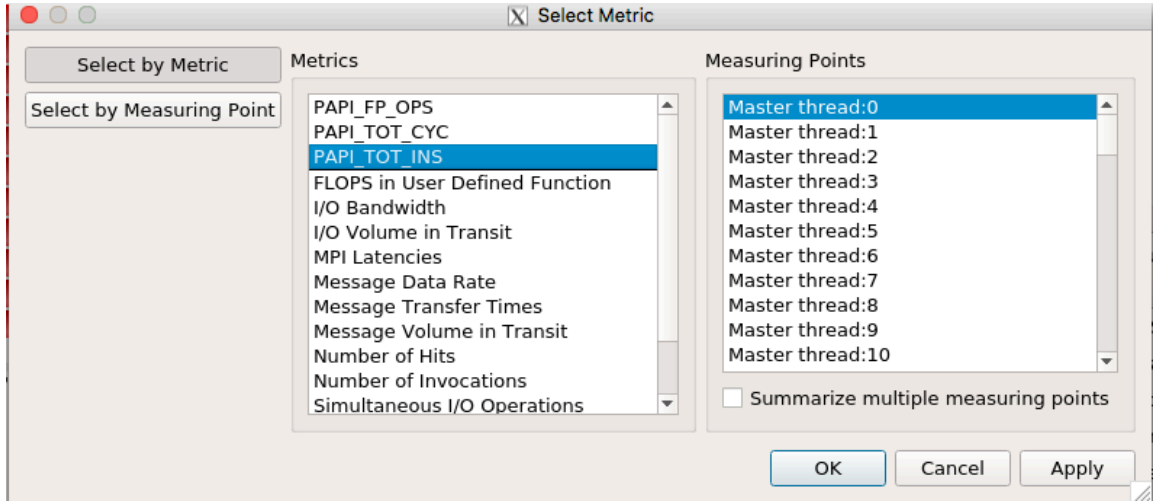
Vampir with OTF2 trace

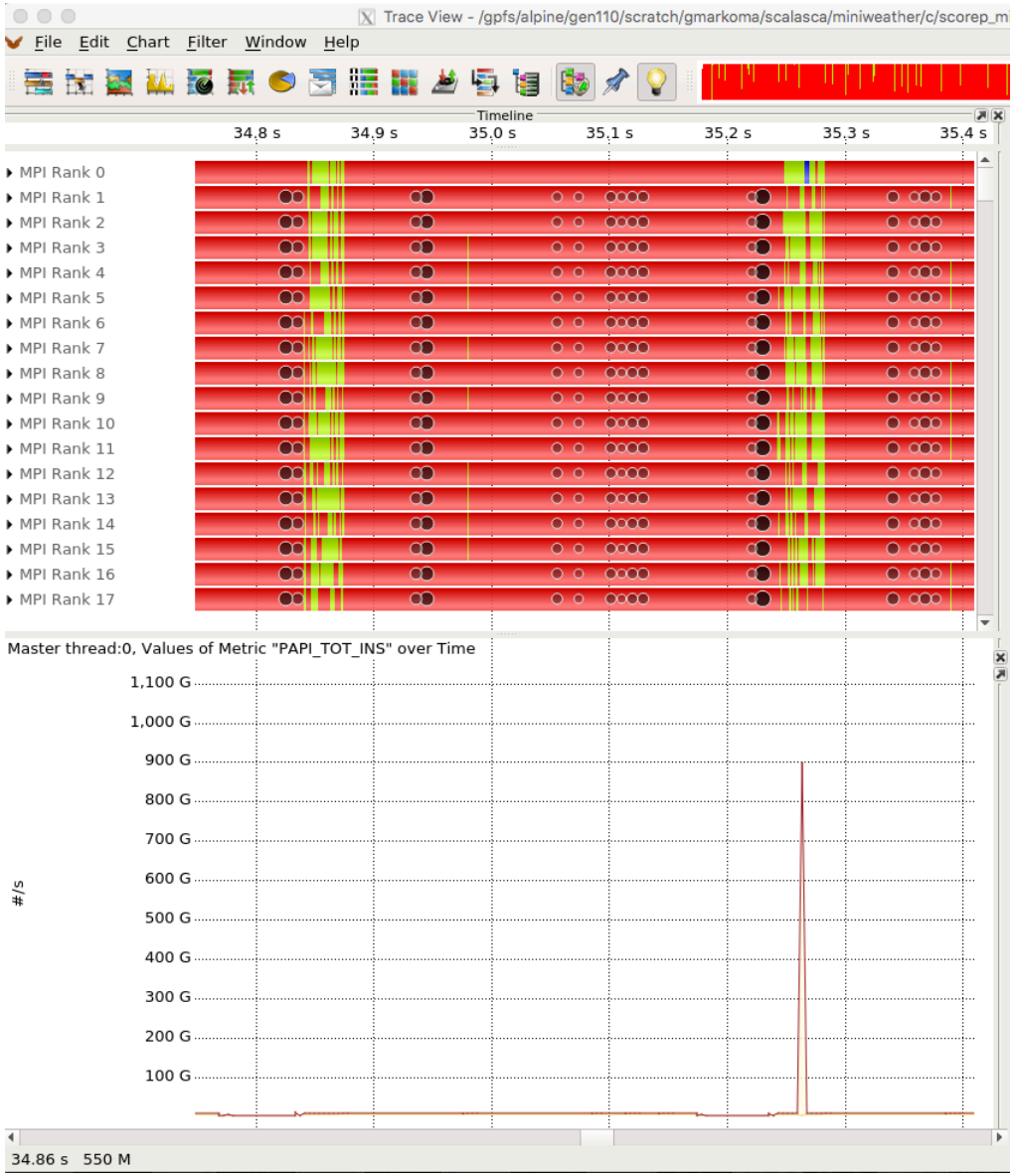


Vampir with OTF2 trace

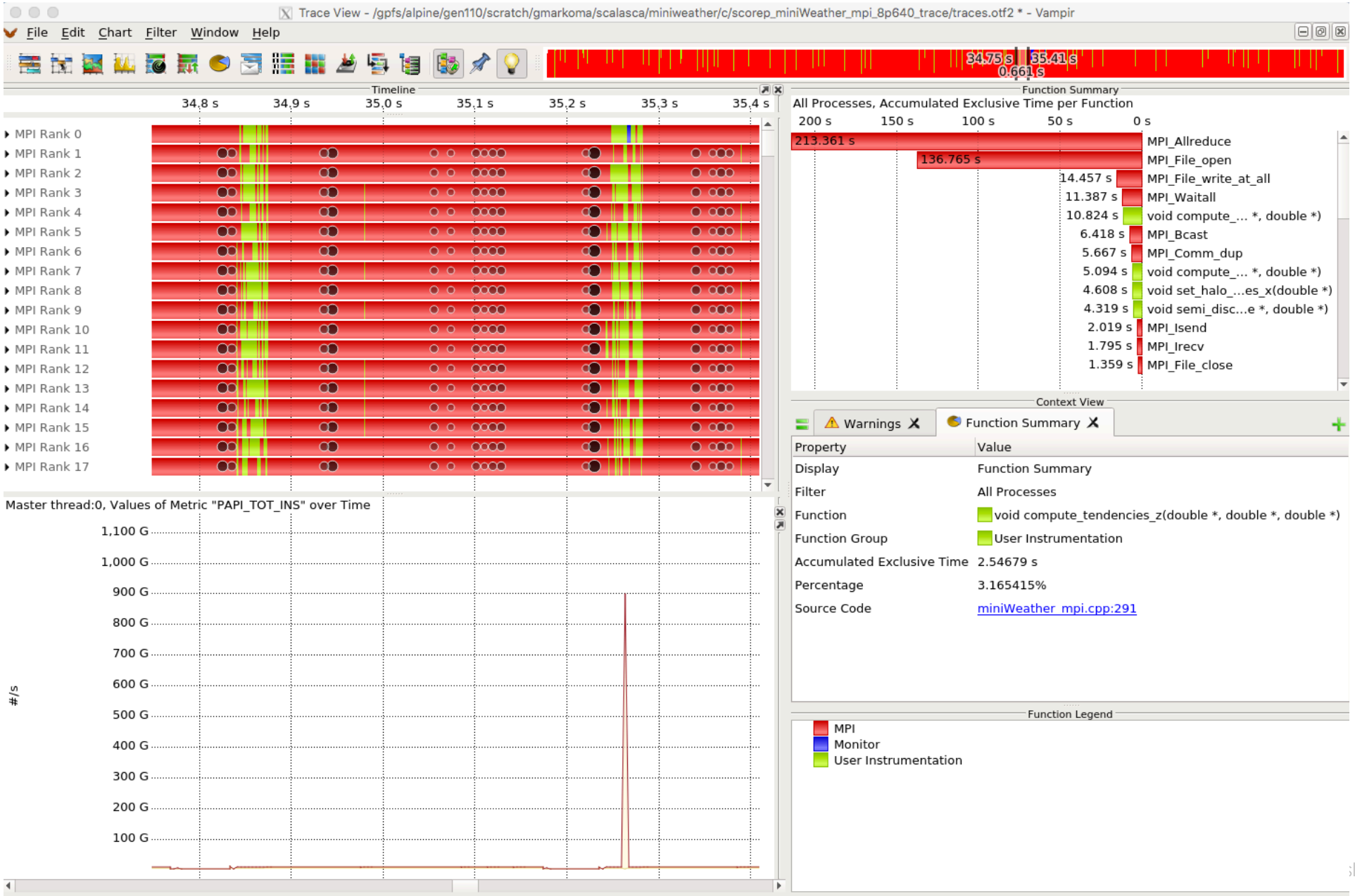


Vampir with OTF2 trace

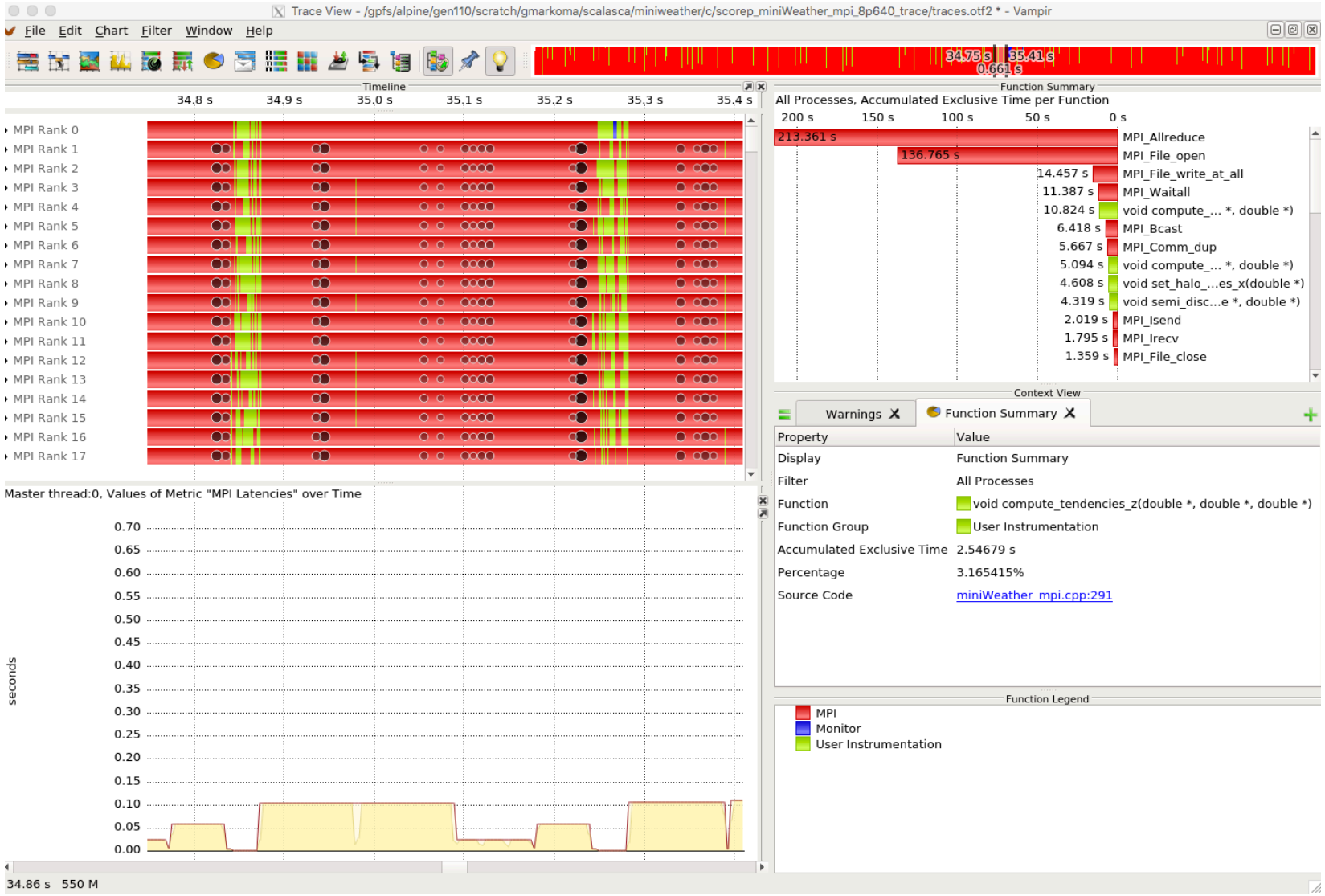




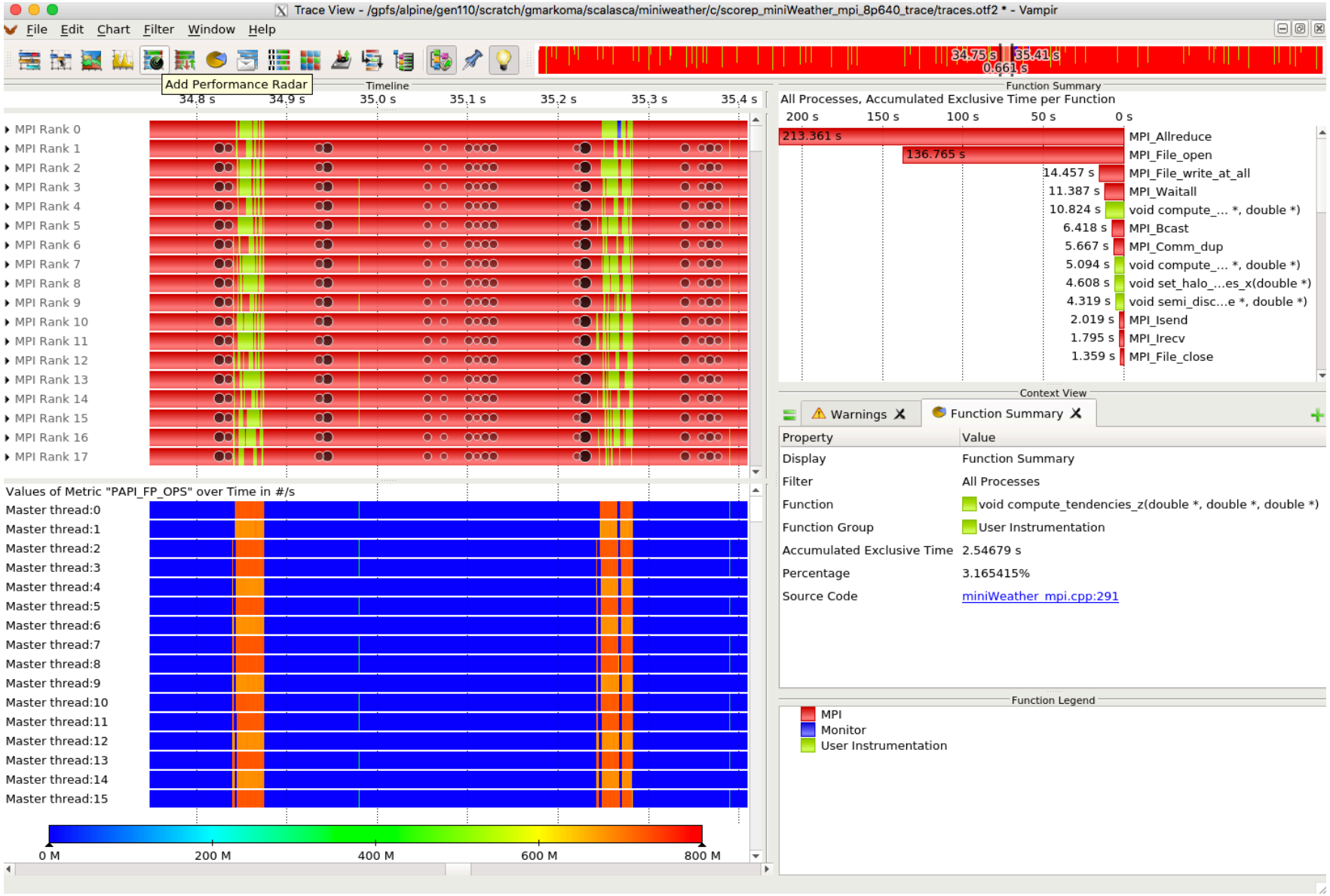
Vampir with OTF2 trace



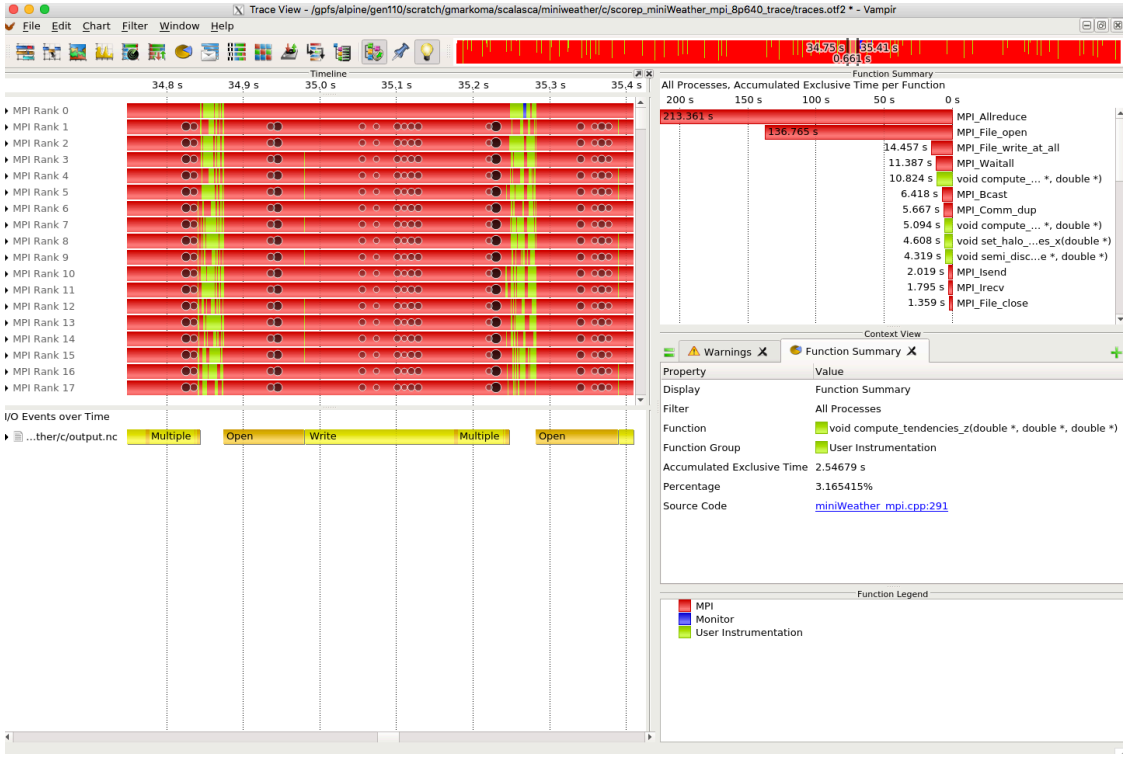
Vampir with OTF2 trace



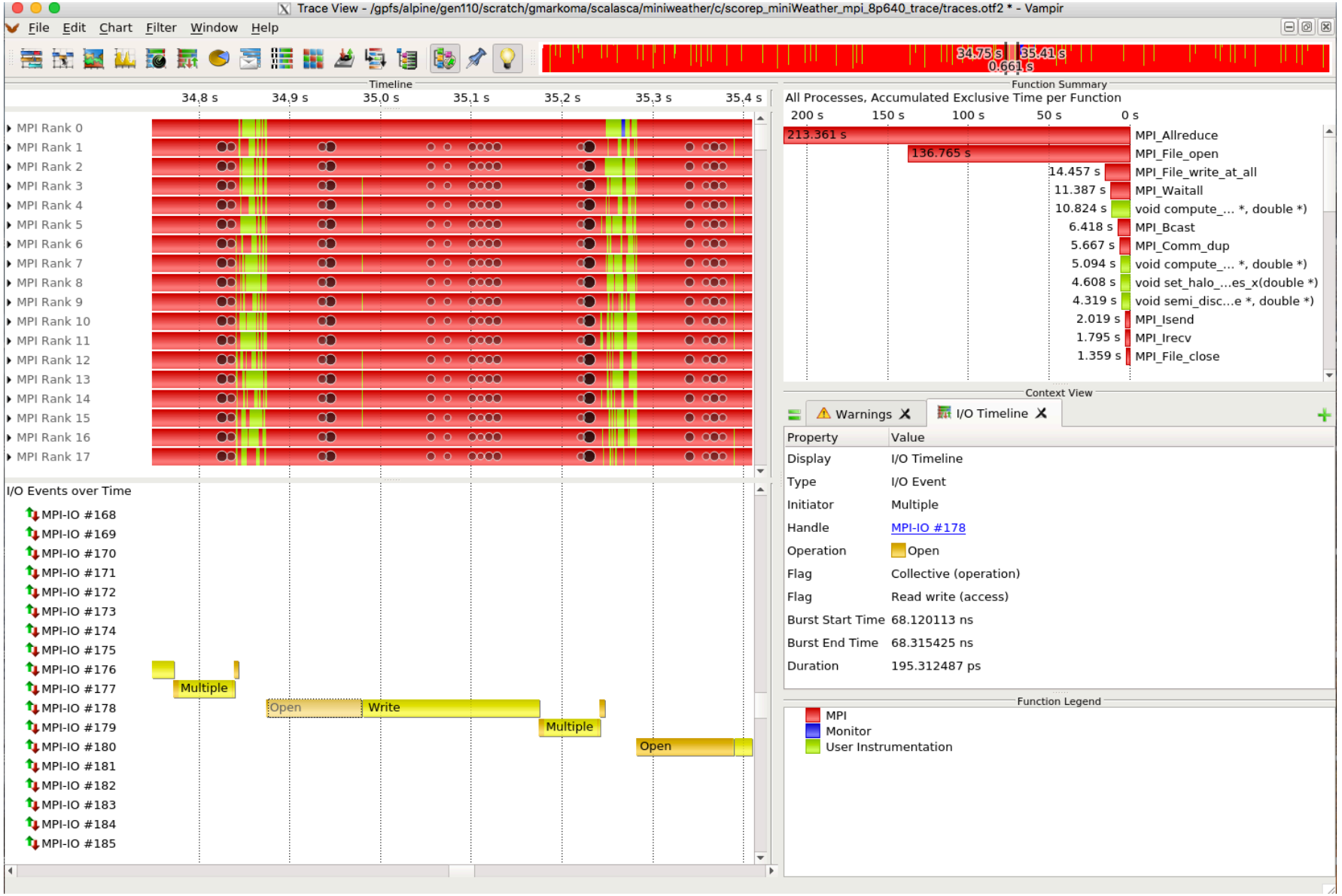
Vampir with OTF2 trace



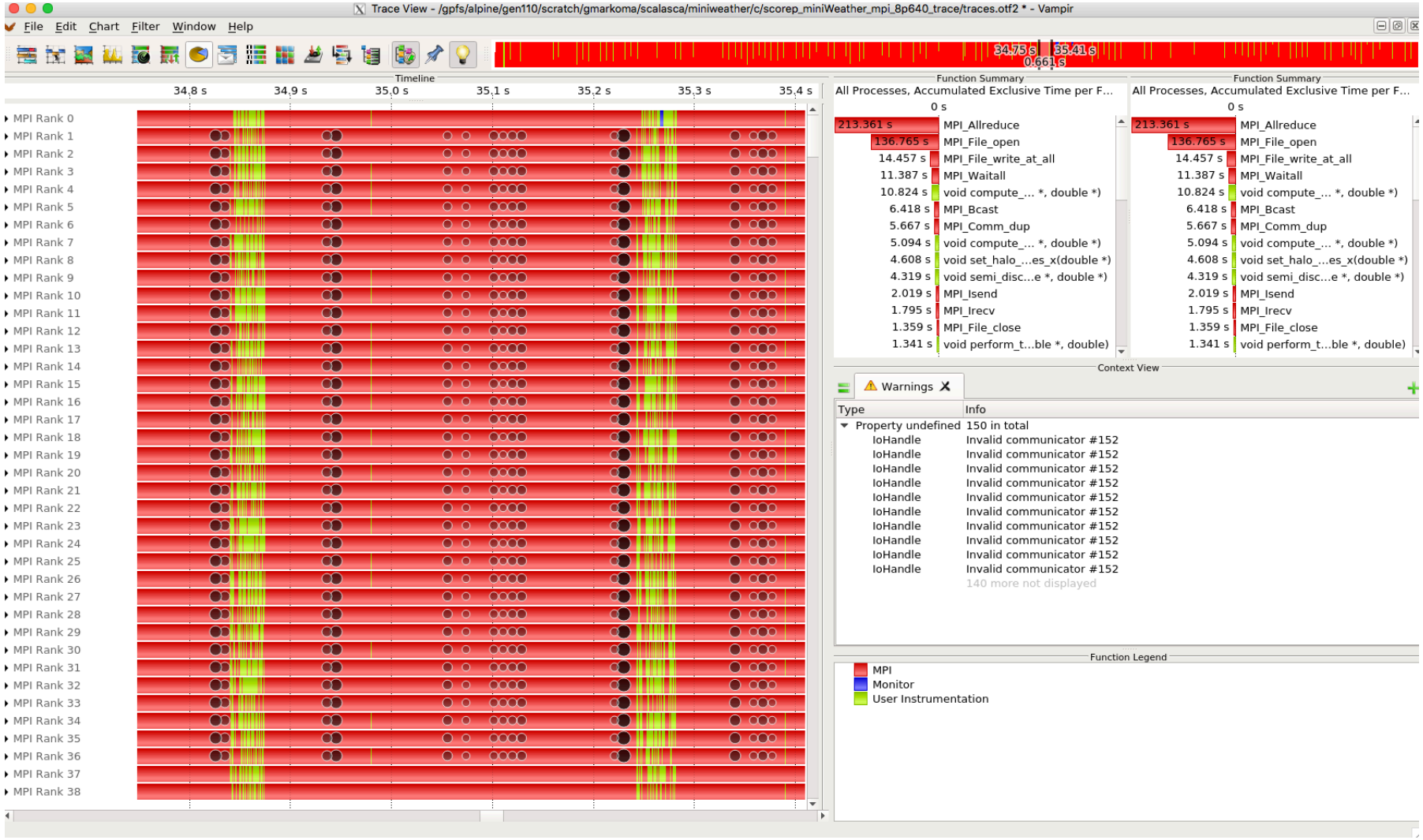
Vampir with OTF2 trace



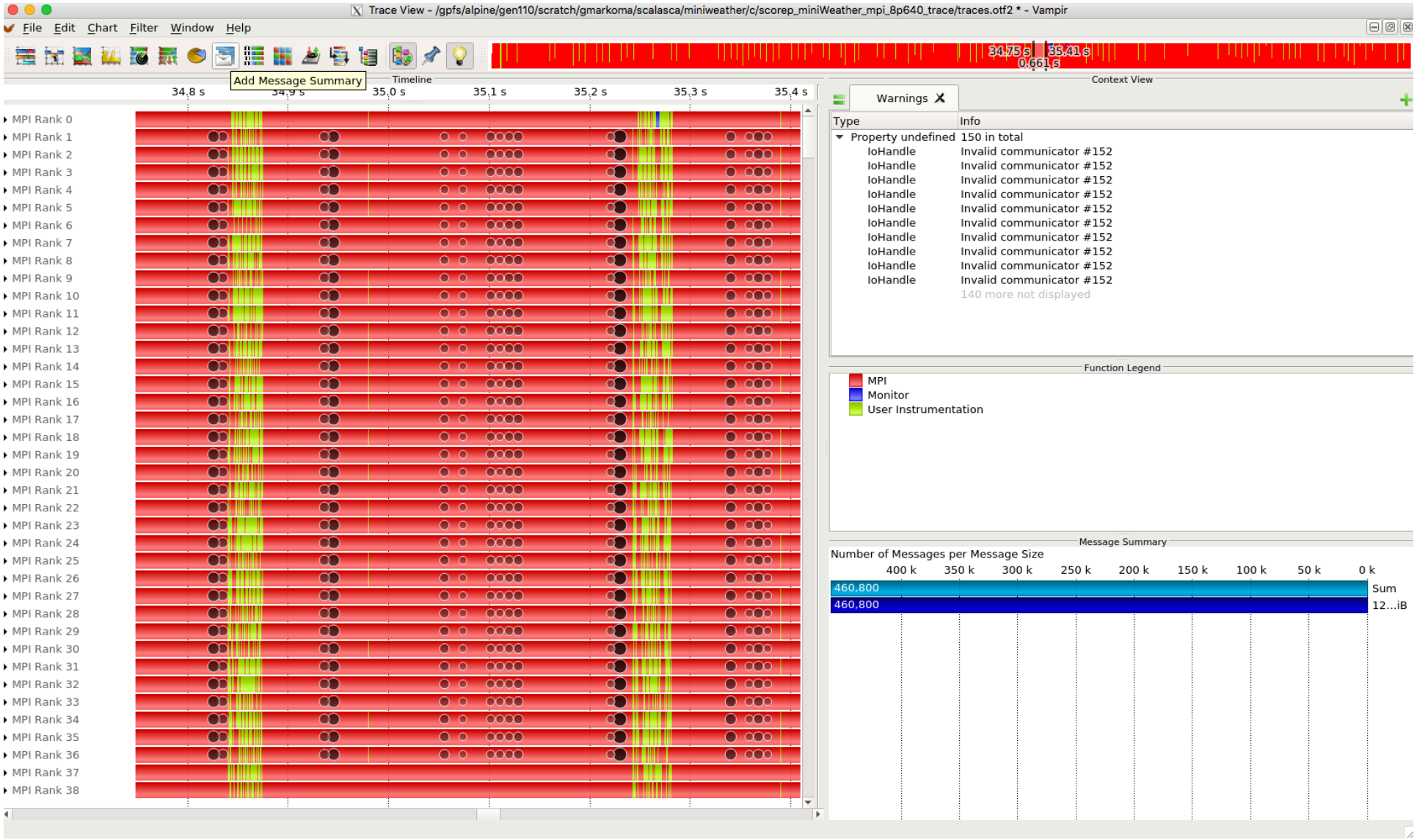
Vampir with OTF2 trace



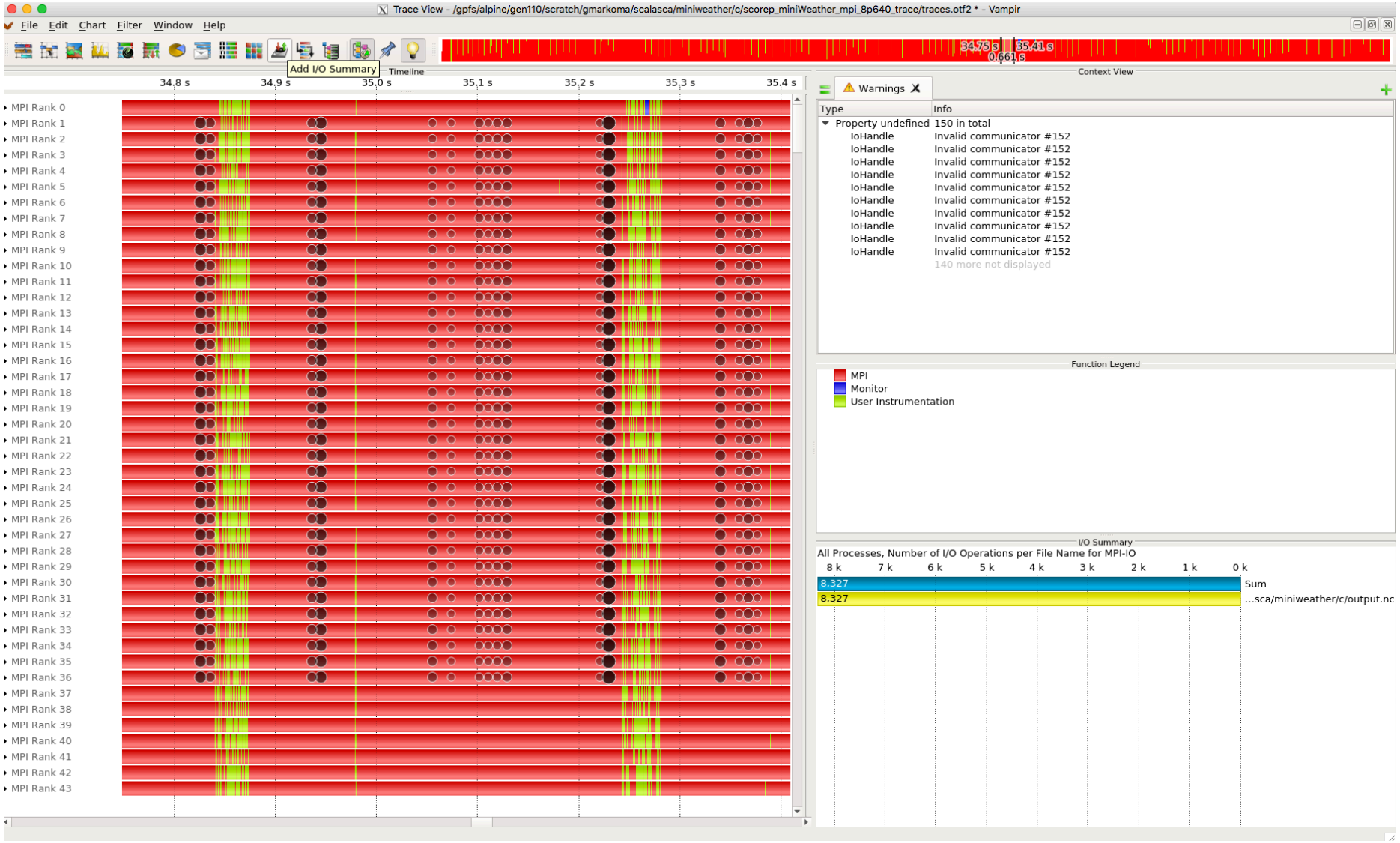
Vampir with OTF2 trace



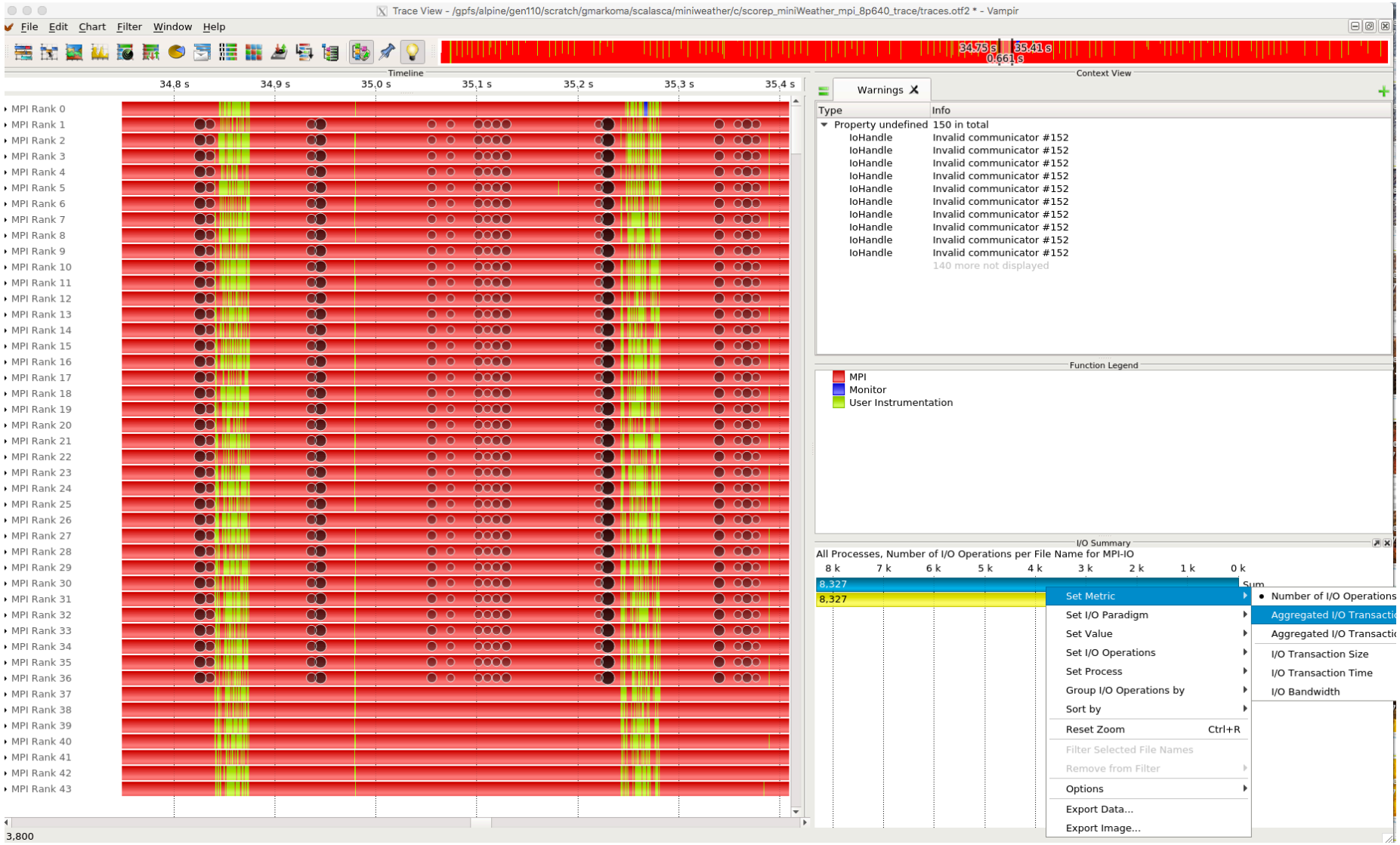
Vampir with OTF2 trace



Vampir with OTF2 trace



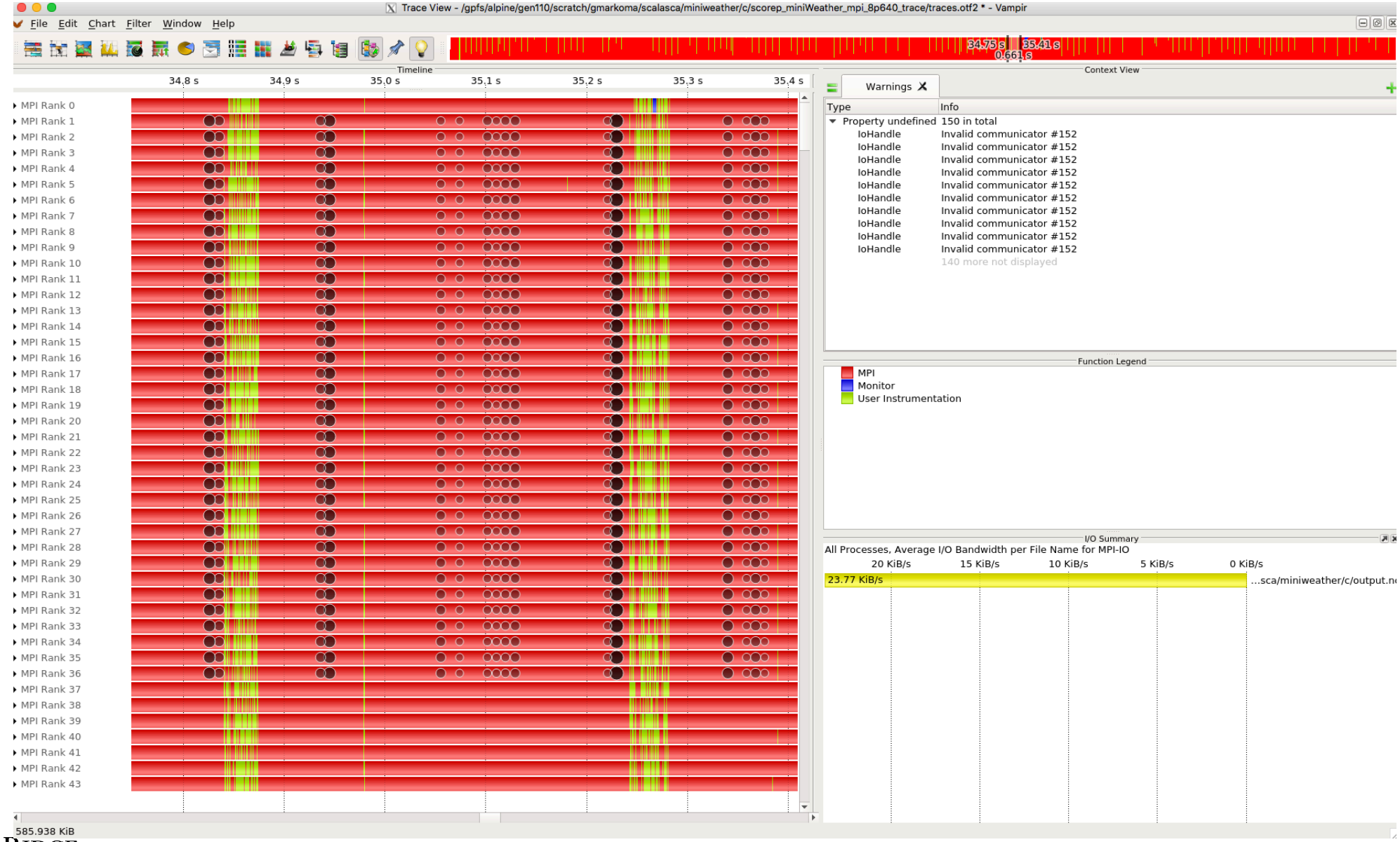
Vampir with OTF2 trace



OAK RIDGE National Laboratory

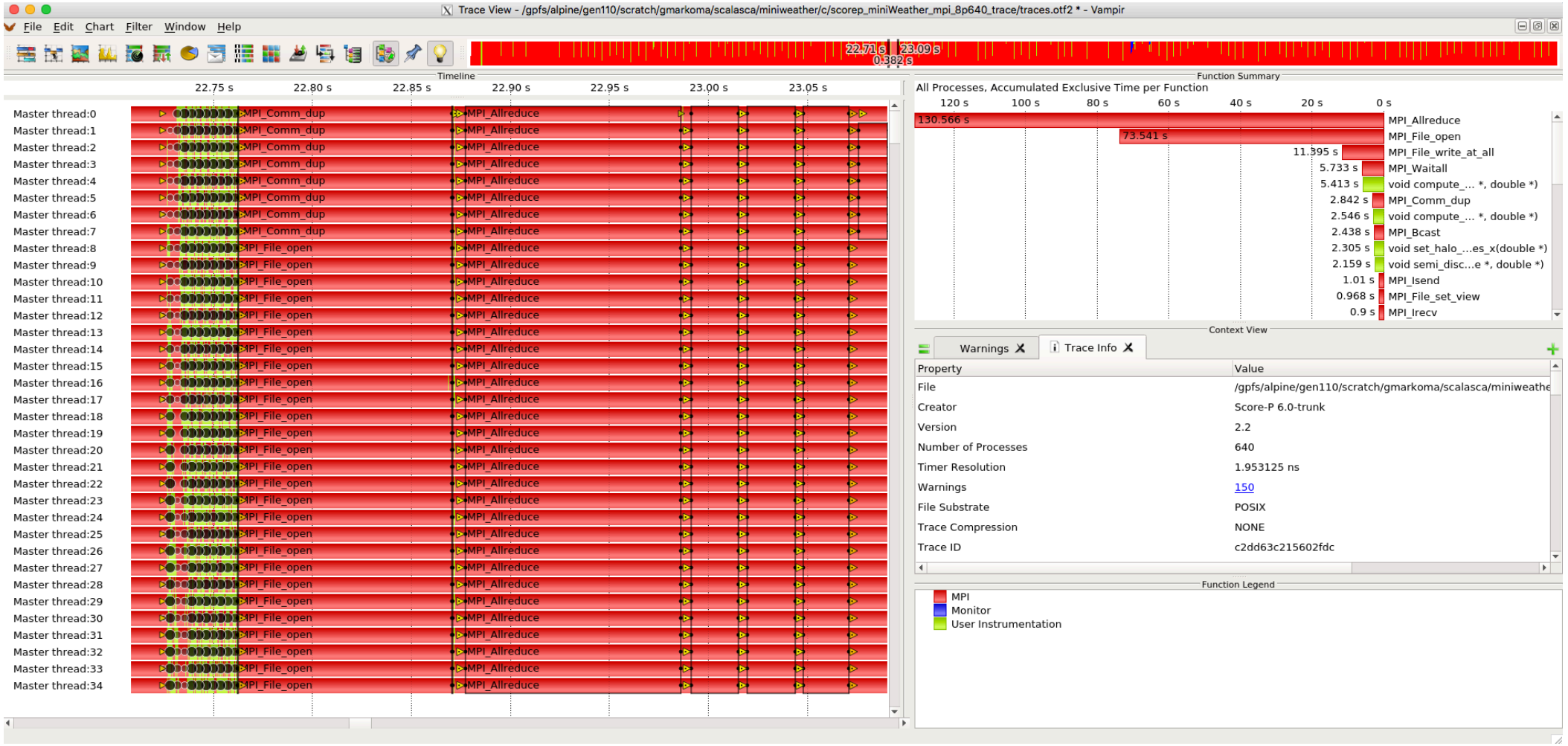
Open slide master to edit

Vampir with OTF2 trace



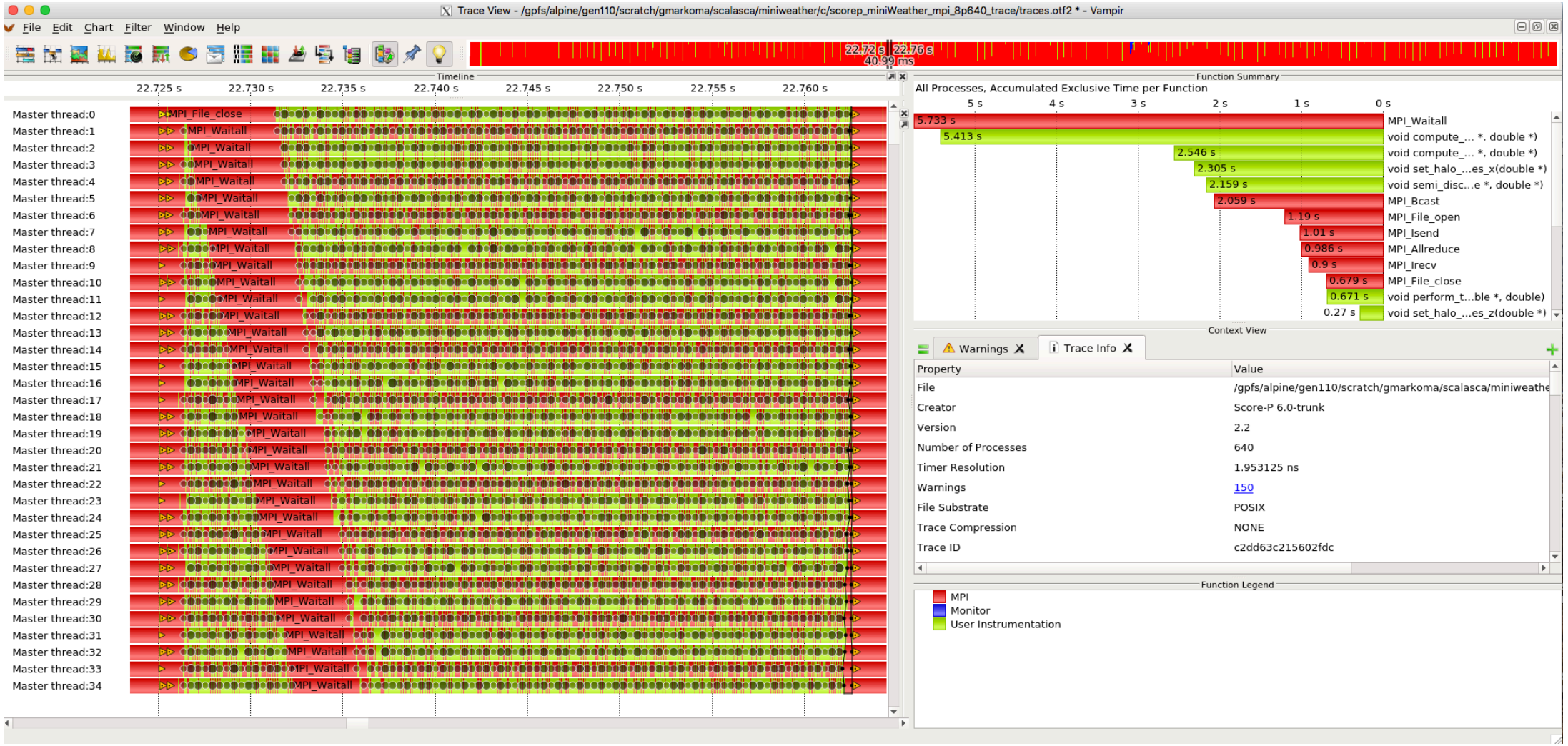
OAK RIDGE National Laboratory

Vampir with OTF2 trace II



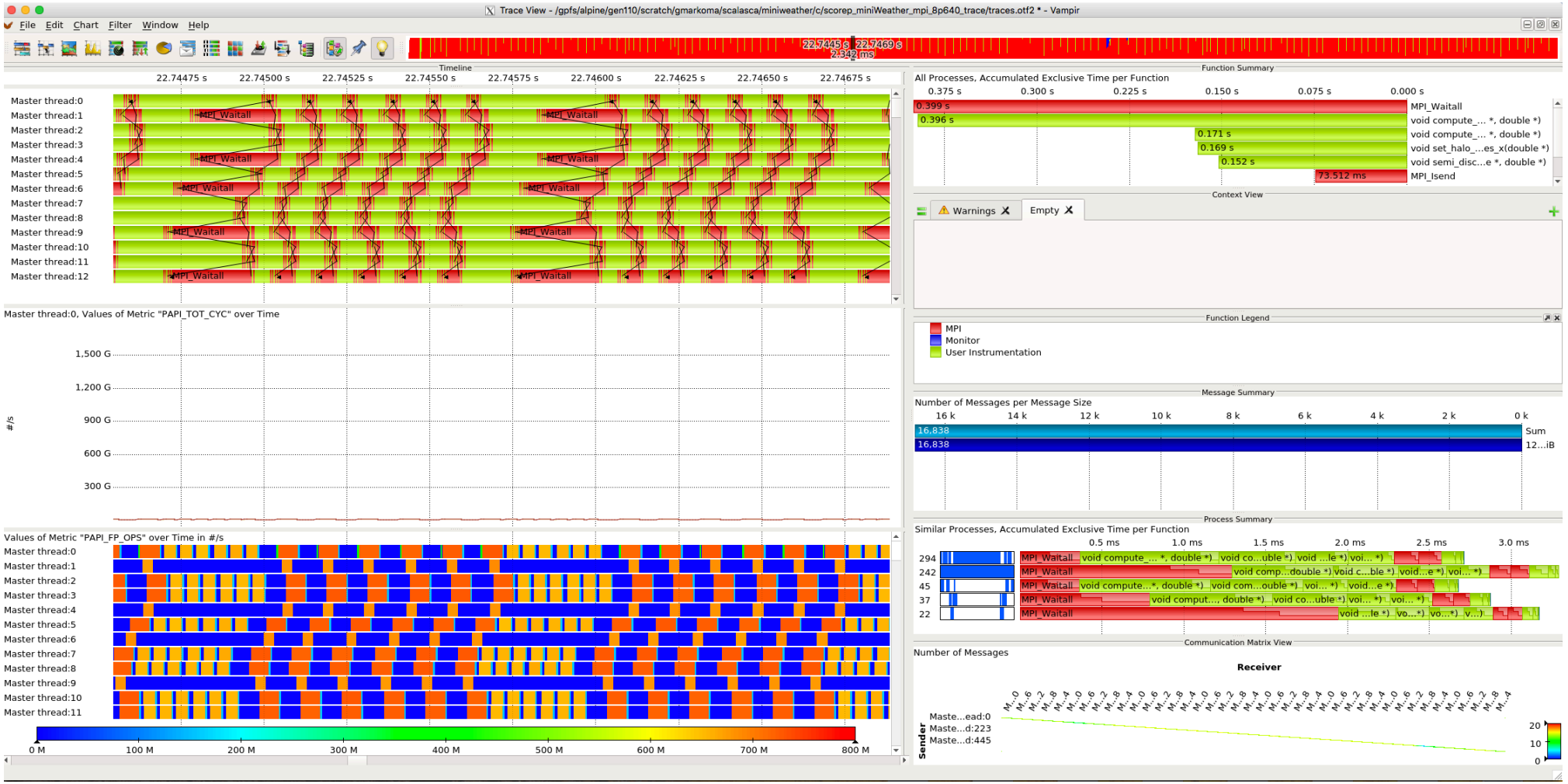
Zoom in with the mouse

Vampir with OTF2 trace III



Zoom in with the mouse

Vampir with OTF2 trace IV



Add functionalities from the menu

Conclusions

- Scalasca can identify bottlenecks based on some patterns
- Useful for MPI and OpenMP
- It can not provide useful information about GPUs but the information exists inside the traces
- It can create OTF2 trace files and use Vampir to visualize