Parallel Performance Visualization on the Cray XT4

Luiz DeRose
Programming Environment Director
Cray Inc.
ldr@cray.com

NERSC
September 18-20, 2007

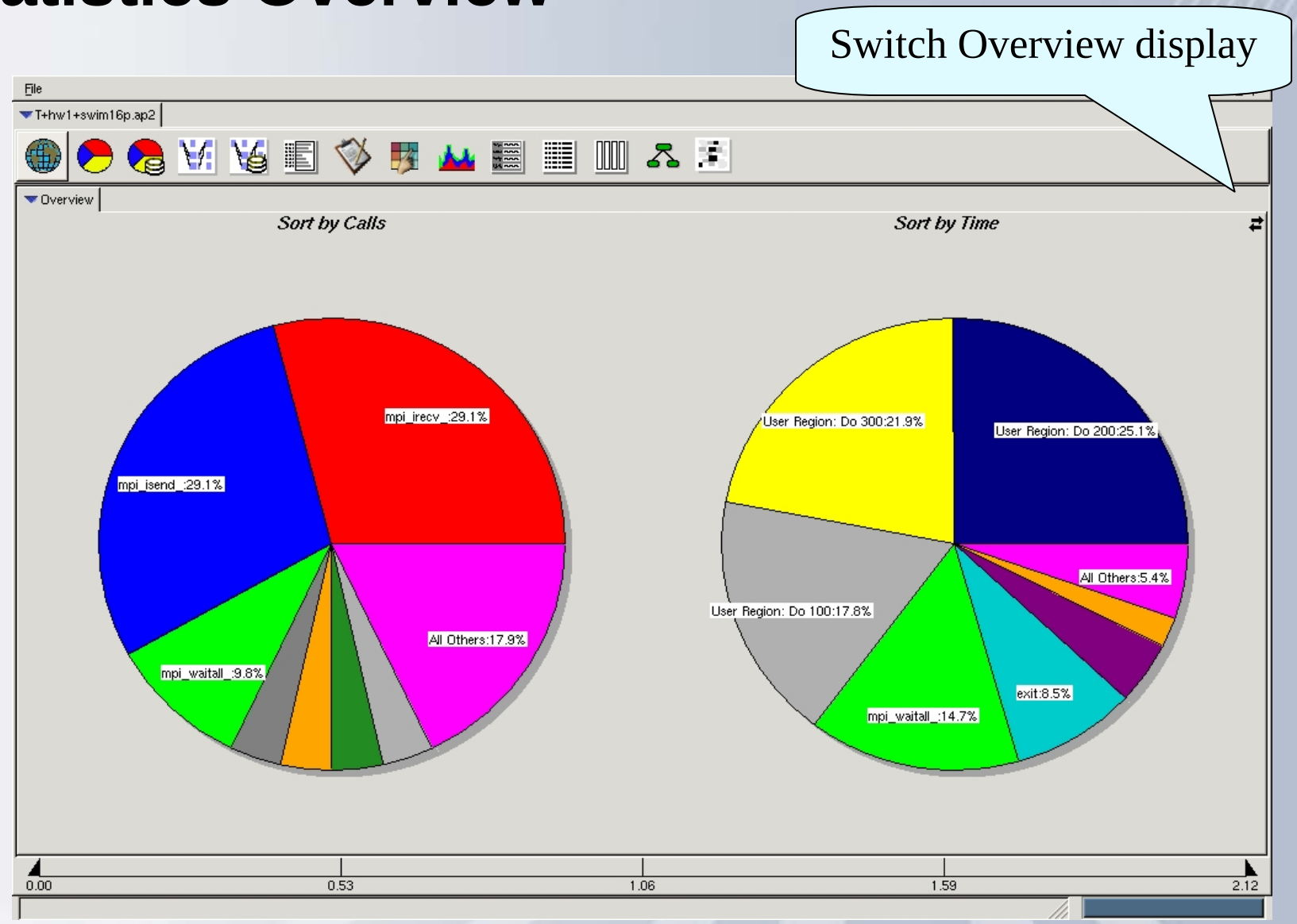
Luiz DeRose (ldr@cray.com) © Cray Inc.

Cray Apprentice^2

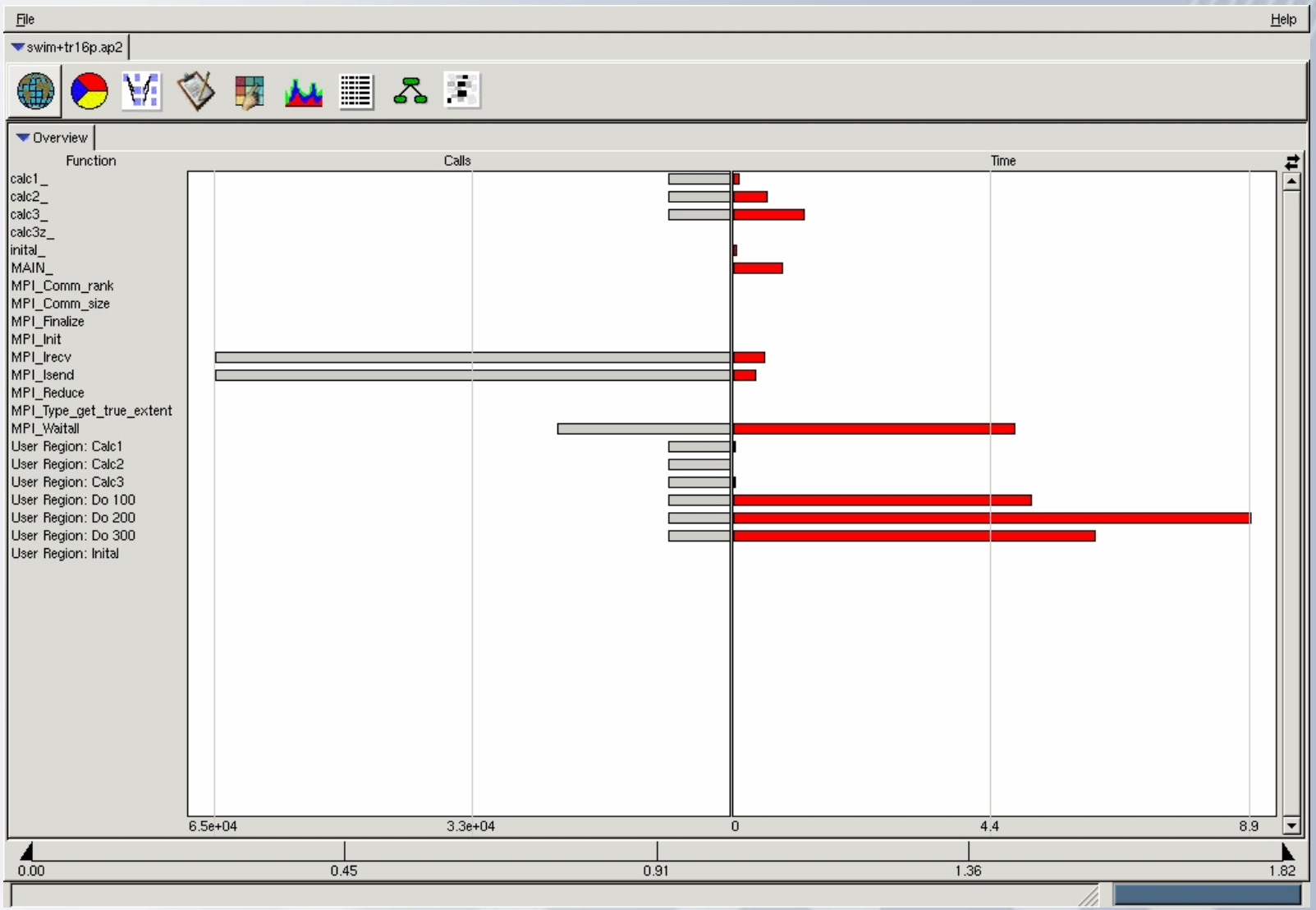
- Call graph profile
- Communication statistics
- Time-line view
  - Communication
  - I/O
- Activity view
- Pair-wise communication statistics
- Text reports
- Source code mapping

- Cray Apprentice^2
- is target to help and correct:
  - Load imbalance
  - Excessive communication
  - Network contention
  - Excessive serialization
  - I/O Problems

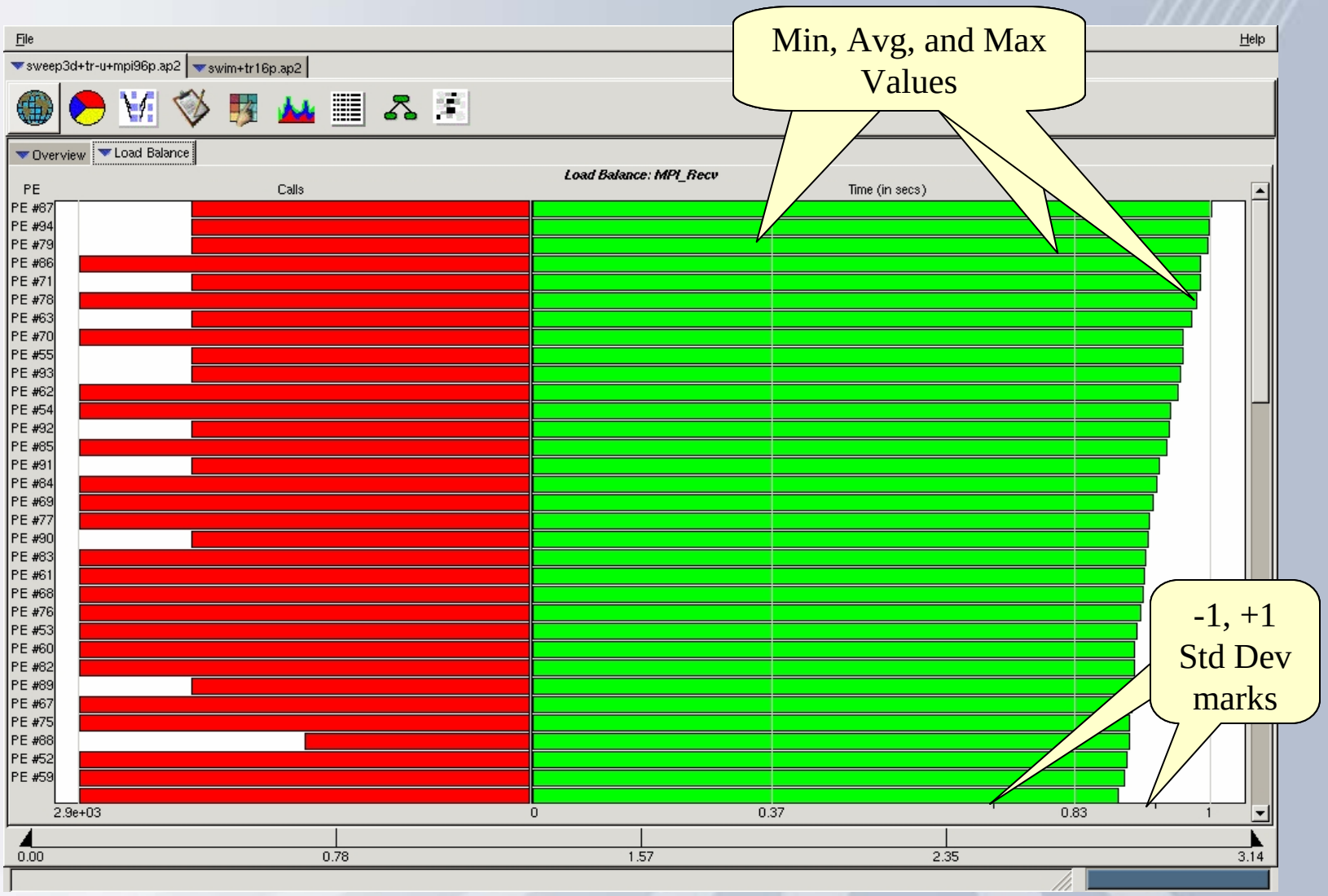
Statistics Overview



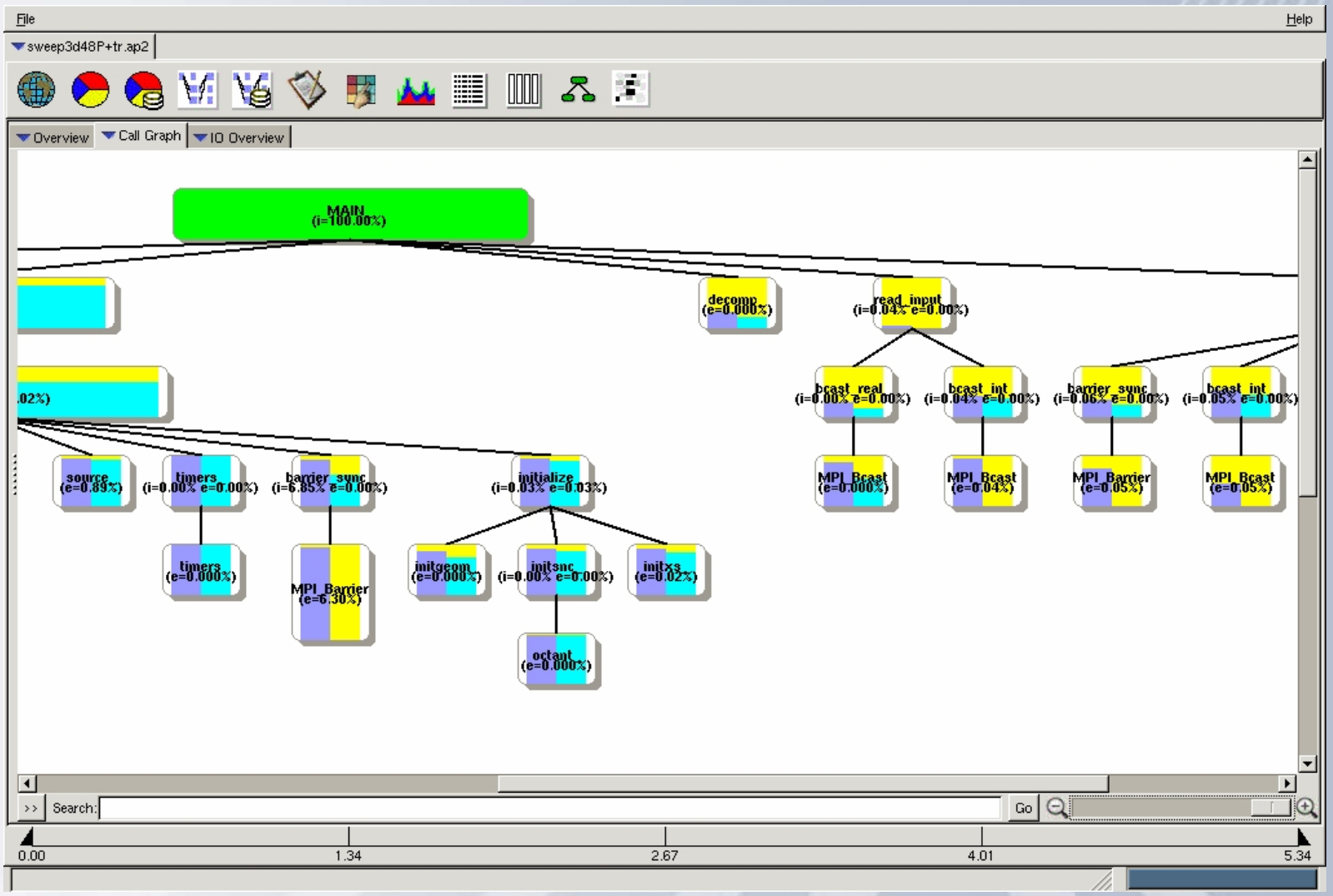
Function Profile



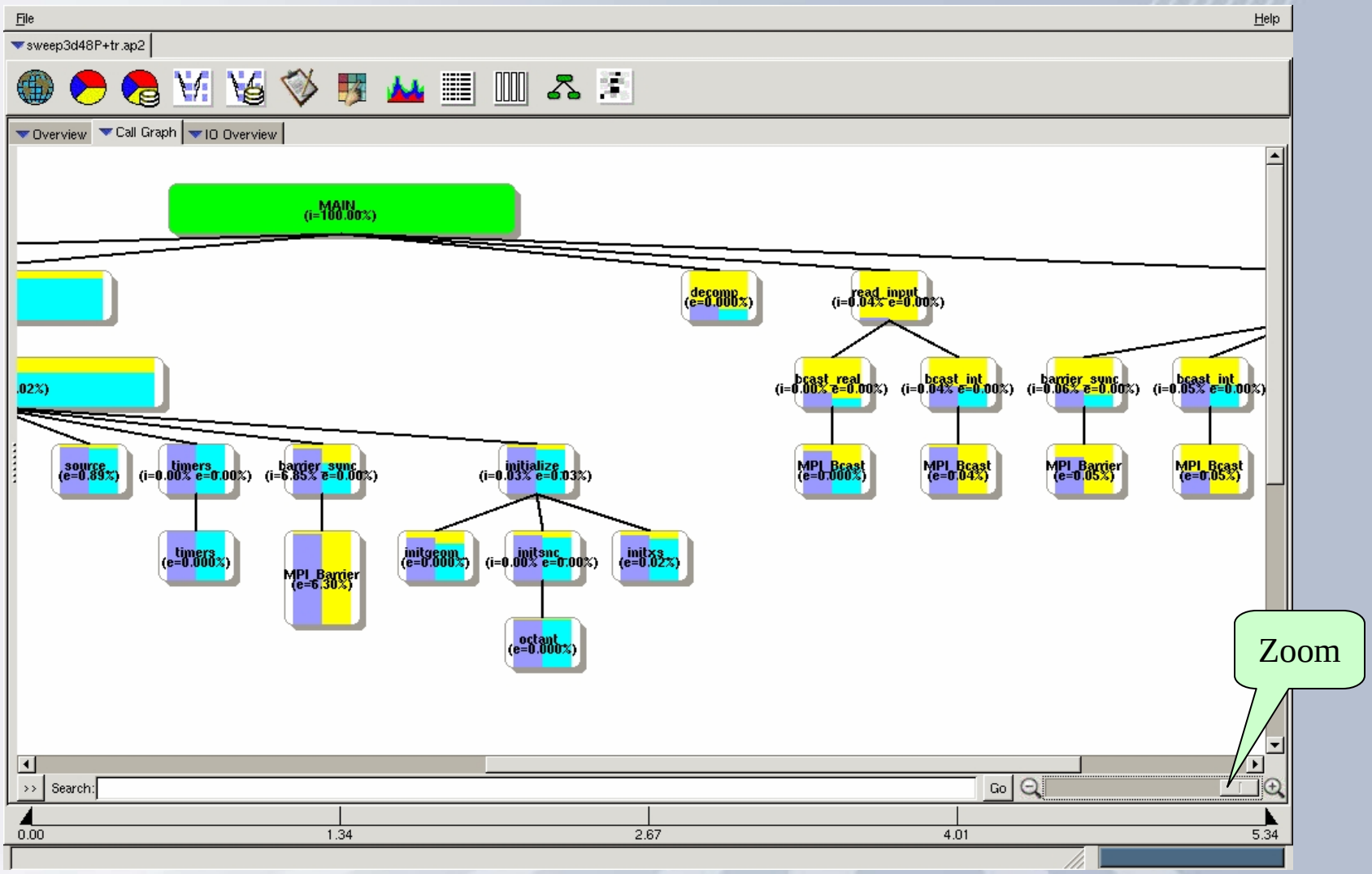
Load Balance View (Aggregated)



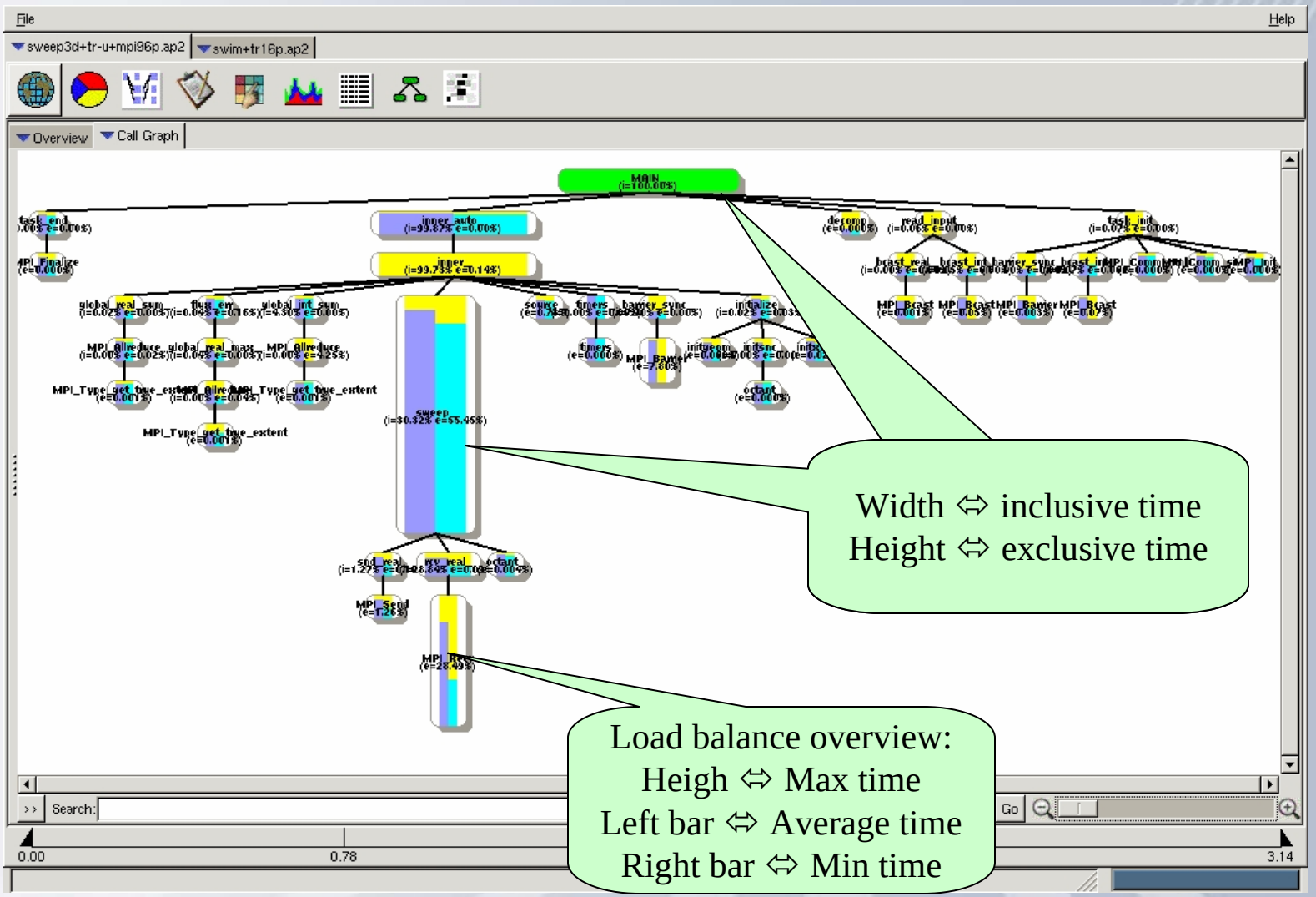
Call Graph View



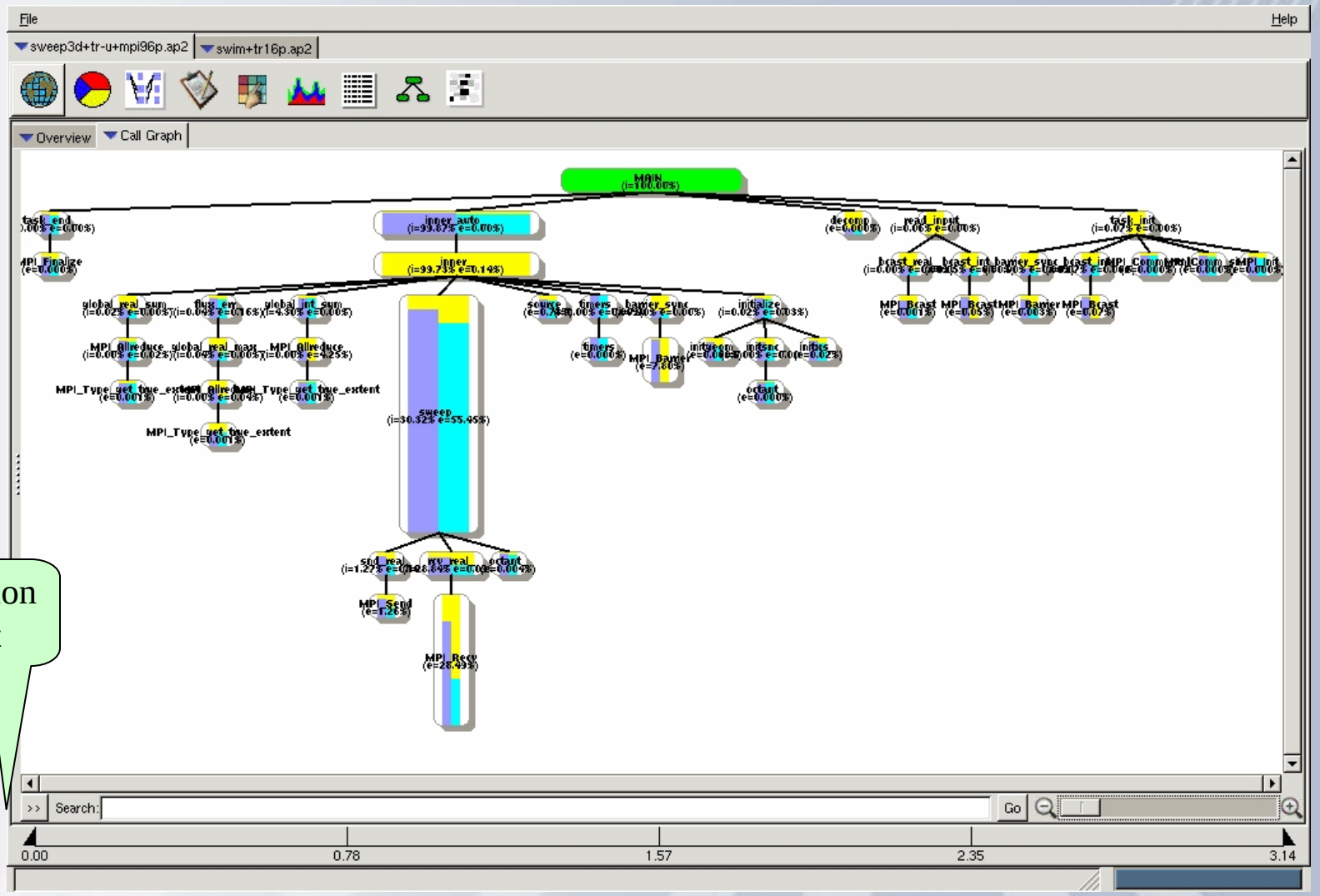
Call Graph View



Call Graph View - Zoom

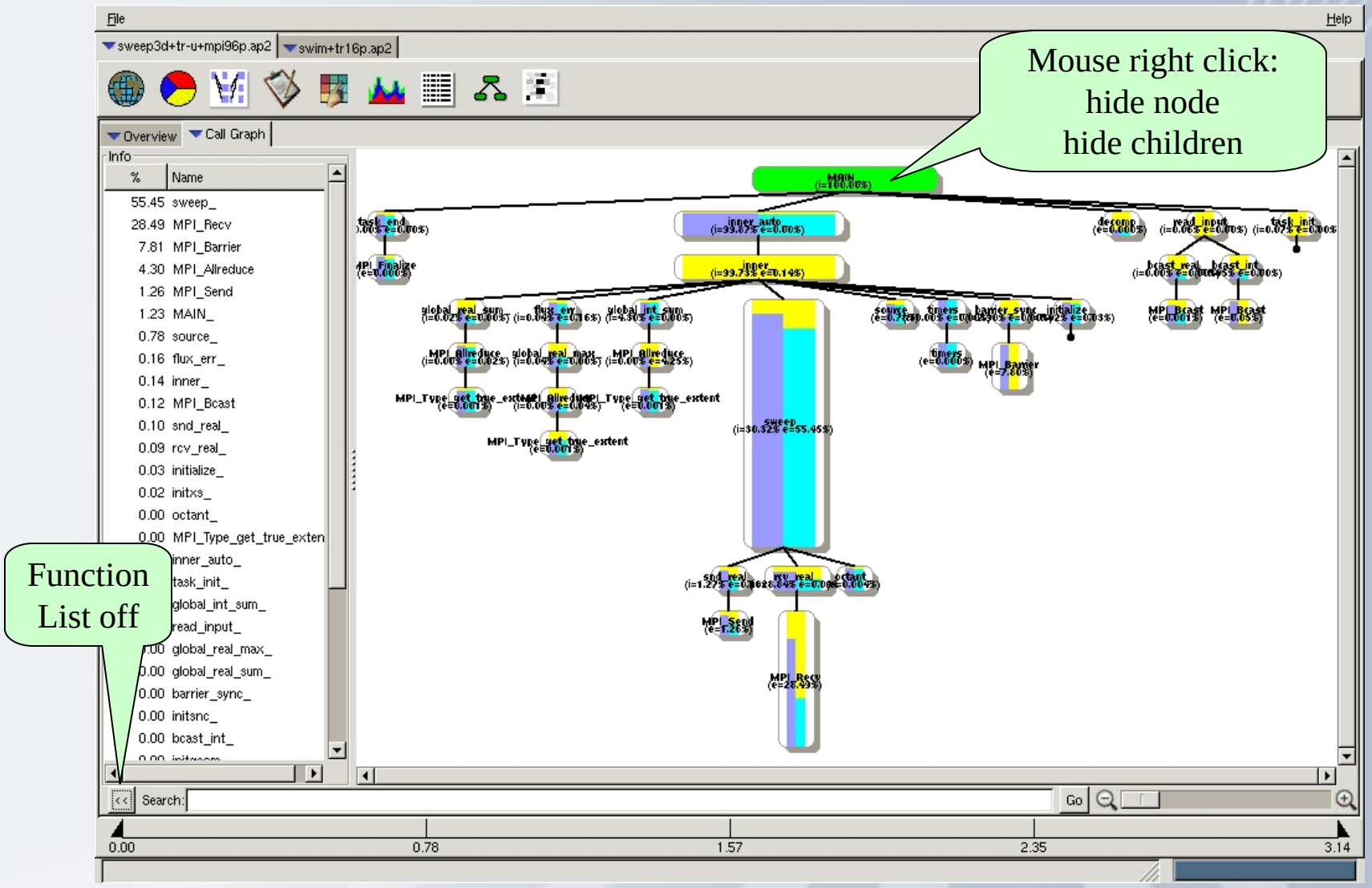


Call Graph View - Zoom

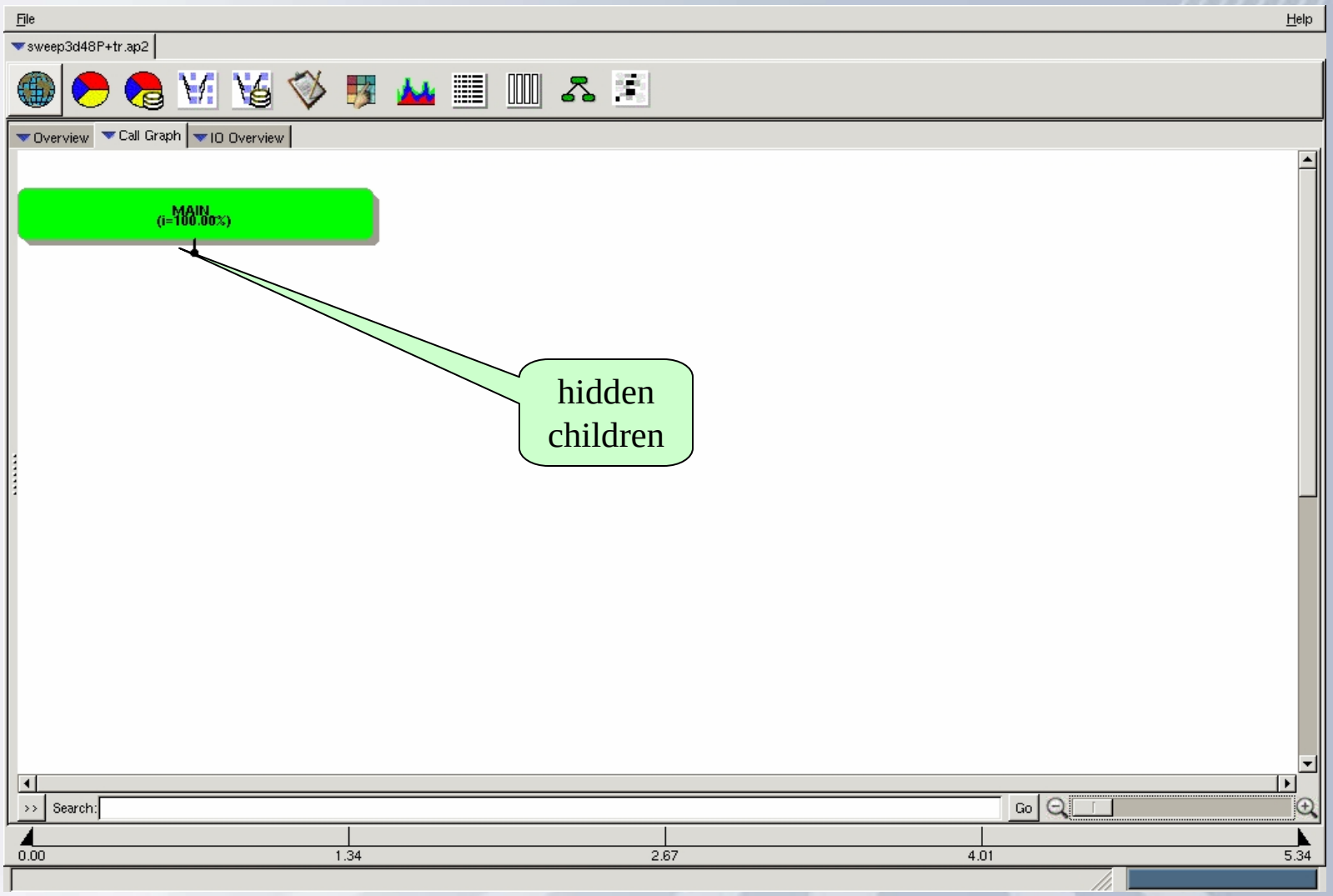


Function List

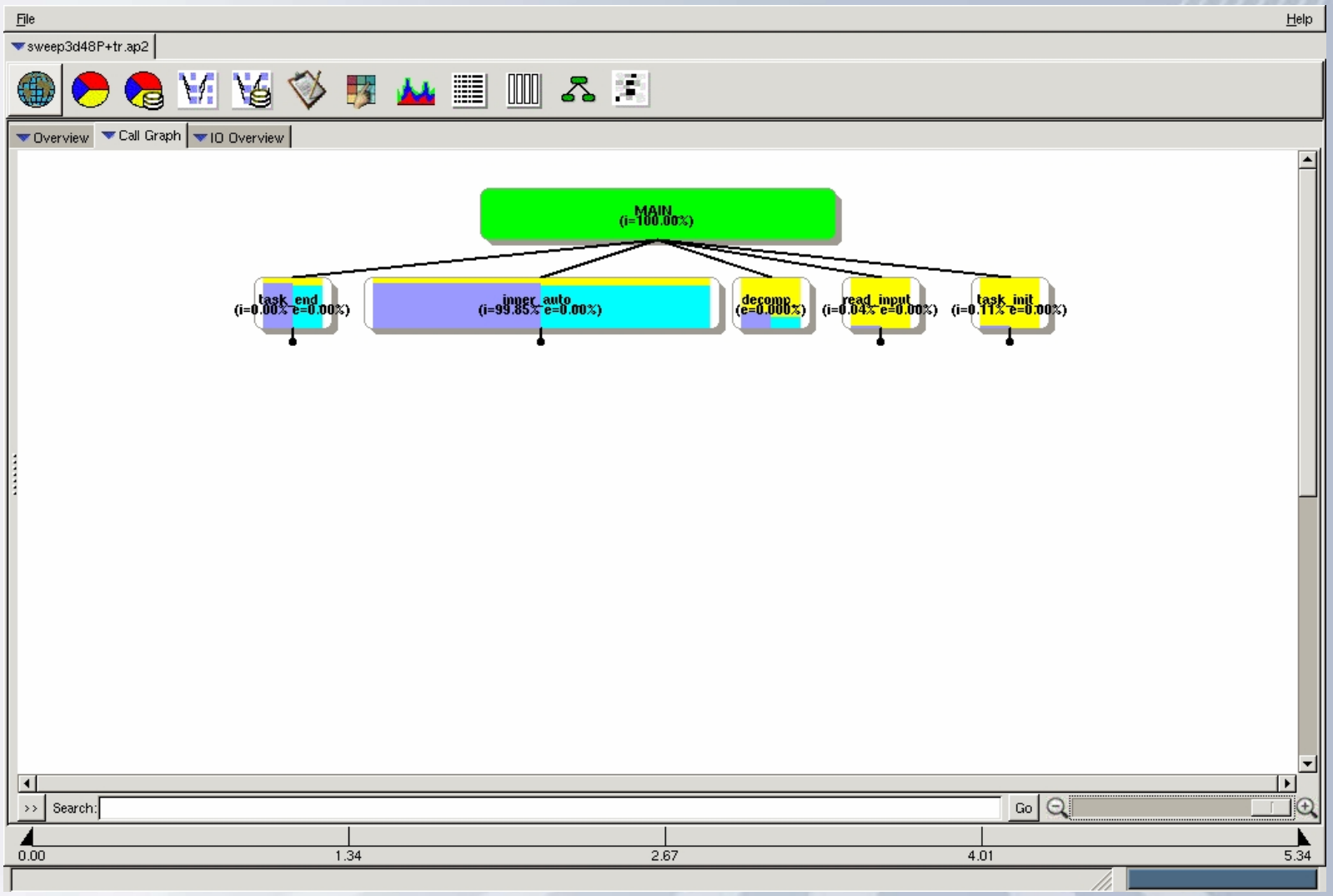
Call Graph View – Function List



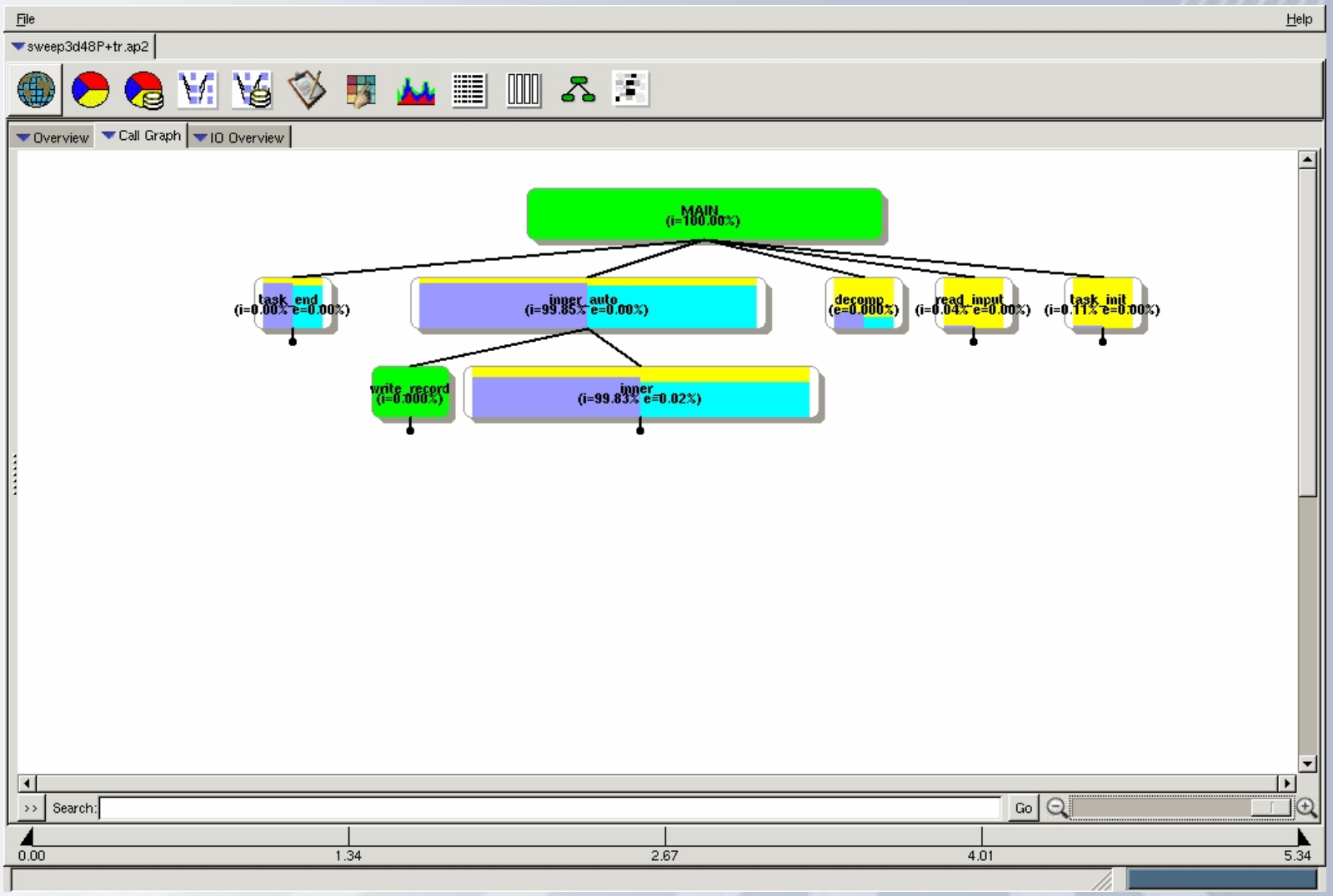
Call Graph Hide Children



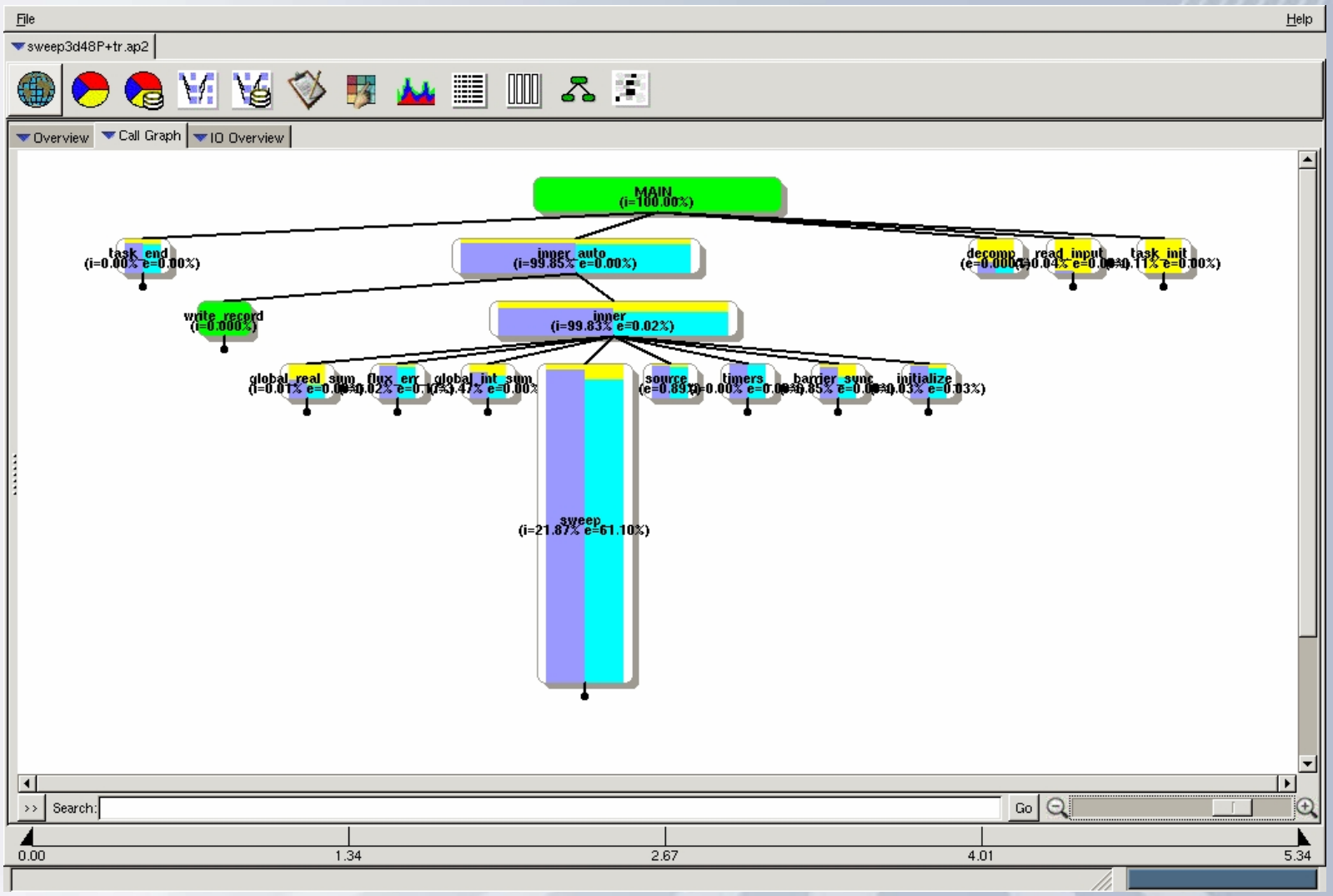
Call Graph Unhide One Level



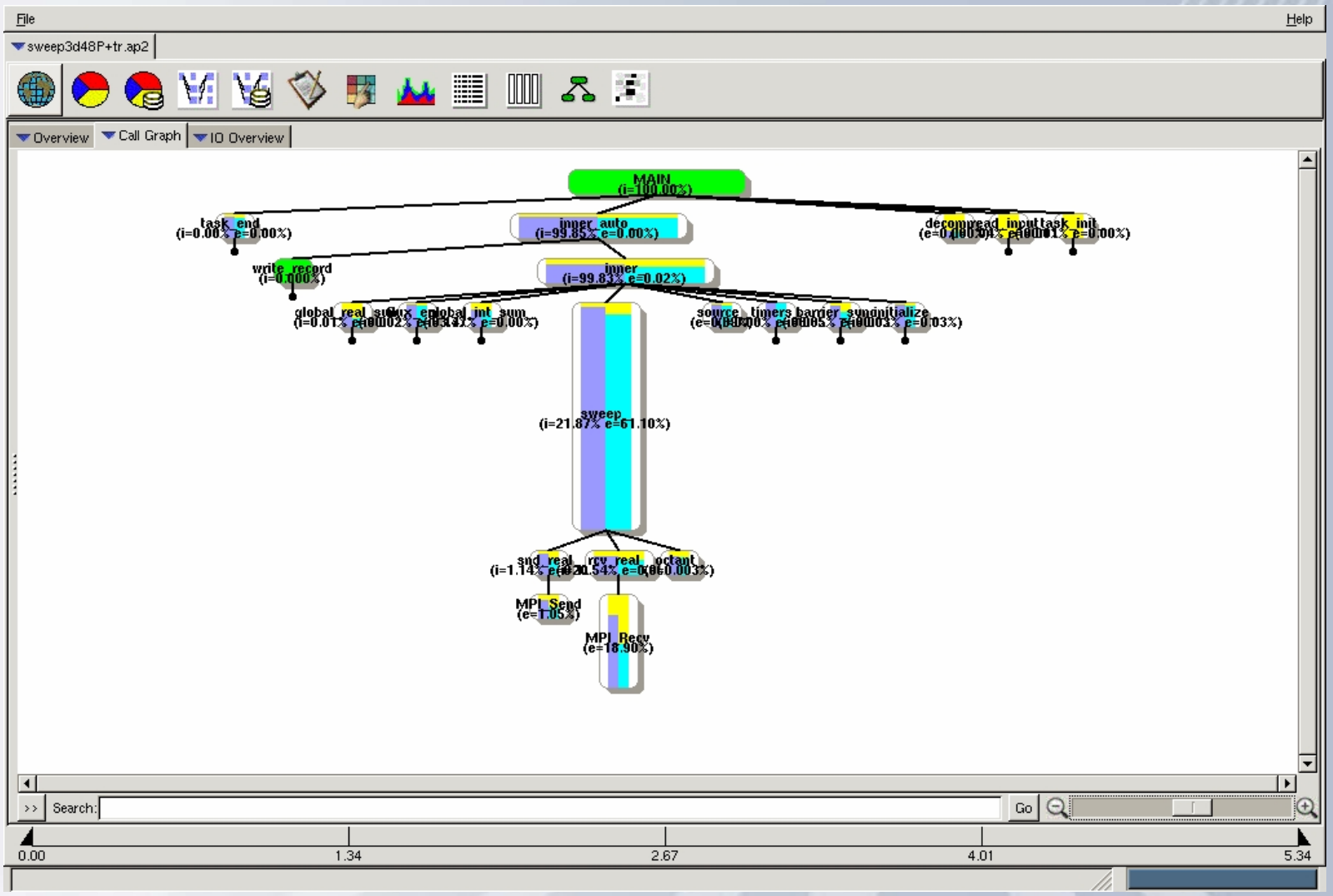
Call Graph Unhide One Level (2)



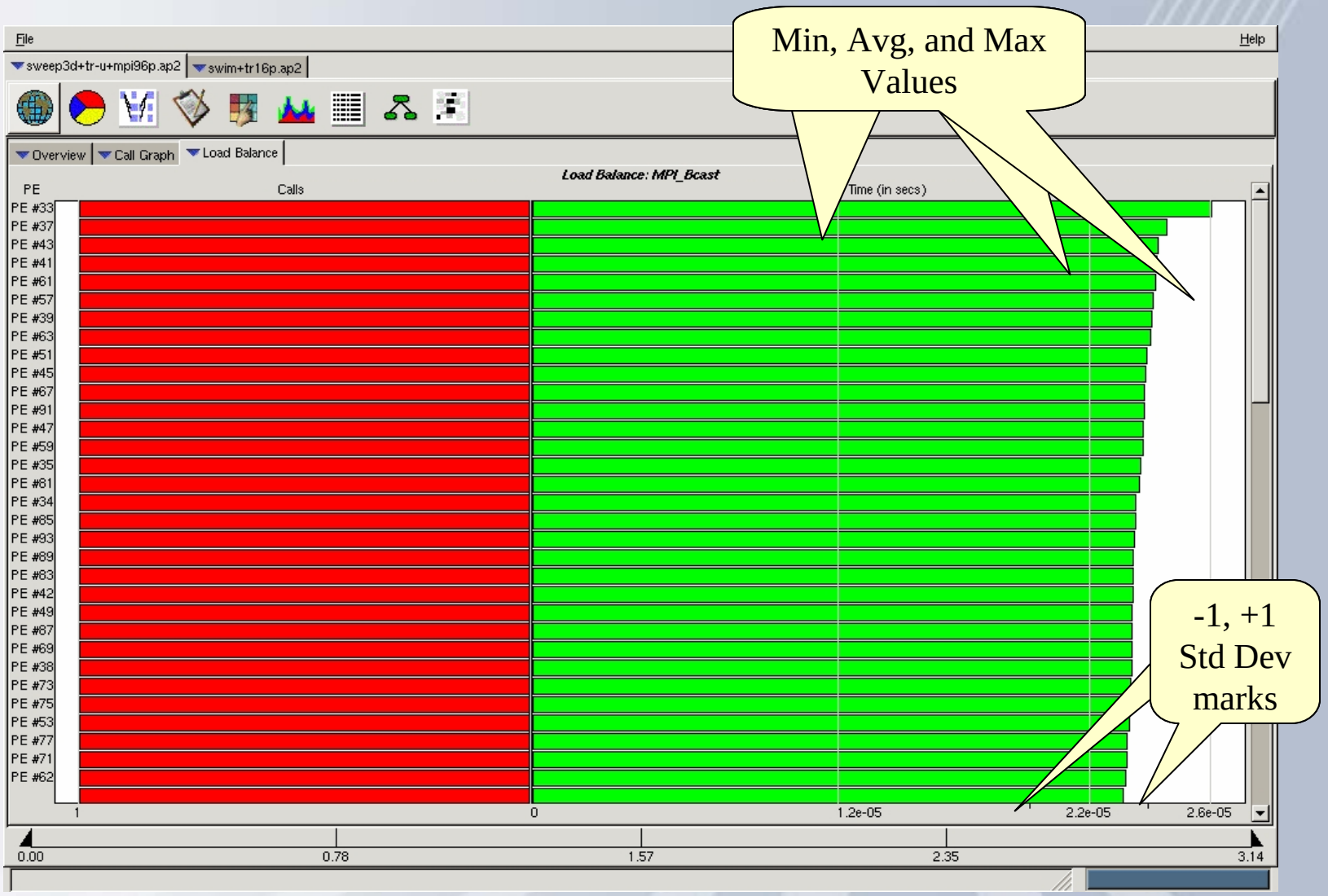
Call Graph Unhide One Level (3)



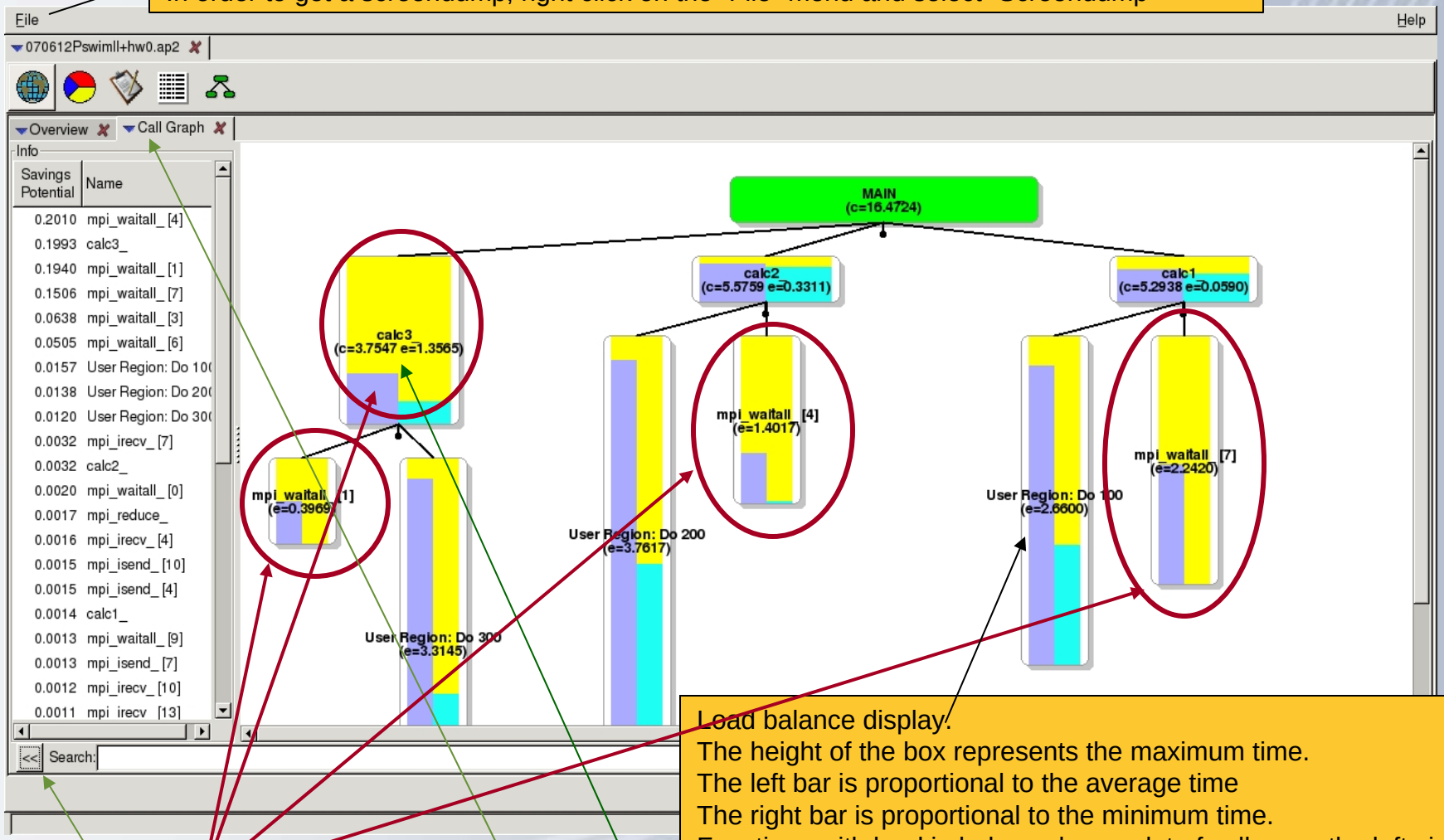
Call Graph Unhide All Children



Load Balance View (from Call Graph)



In order to get a screendump, right click on the “File” menu and select “Screendump”

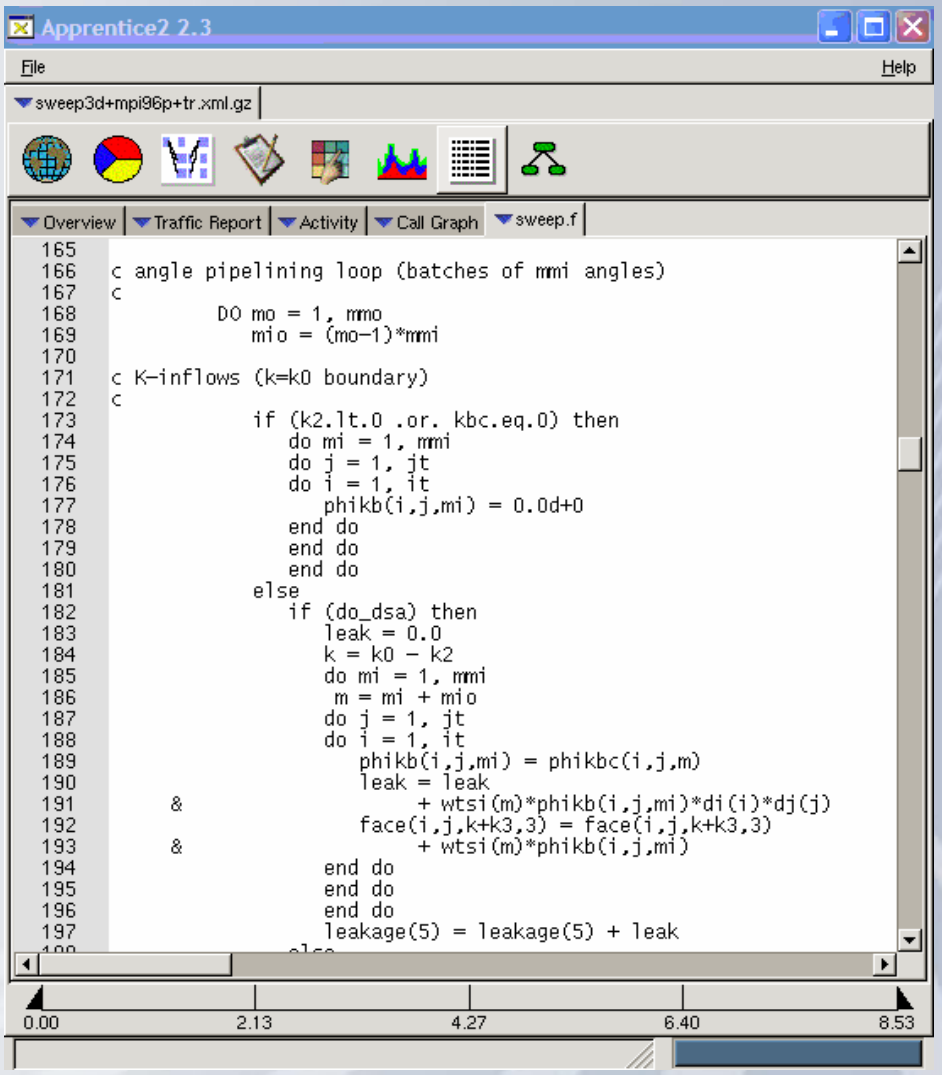


Load balance display.
The height of the box represents the maximum time.
The left bar is proportional to the average time
The right bar is proportional to the minimum time.
Functions with load imbalance have a lot of yellow on the left side

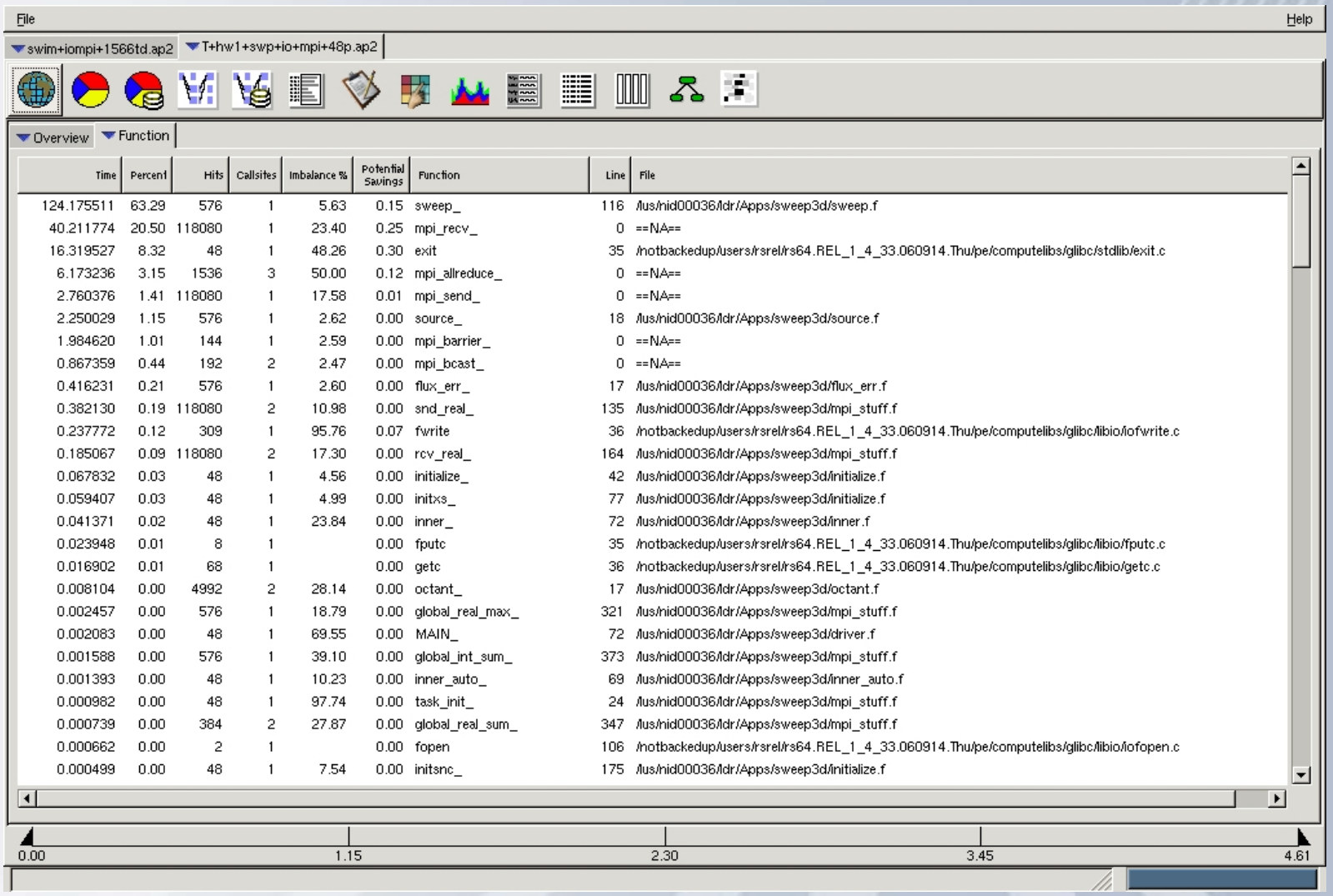
Main functions showing load imbalance. Normally load balance in MPI functions is caused by some load imbalance in computation. In this case, on “Calc3”, which has an artificial imbalance.

List of functions and regions sorted by load imbalance (potential savings). In order to get this list in the call graph, click on the “>>” button and right click the call graph tab and select “potential savings”.
This call graph is filtered to show only the functions that take a reasonable amount of time.

Source Mapping from Call Graph

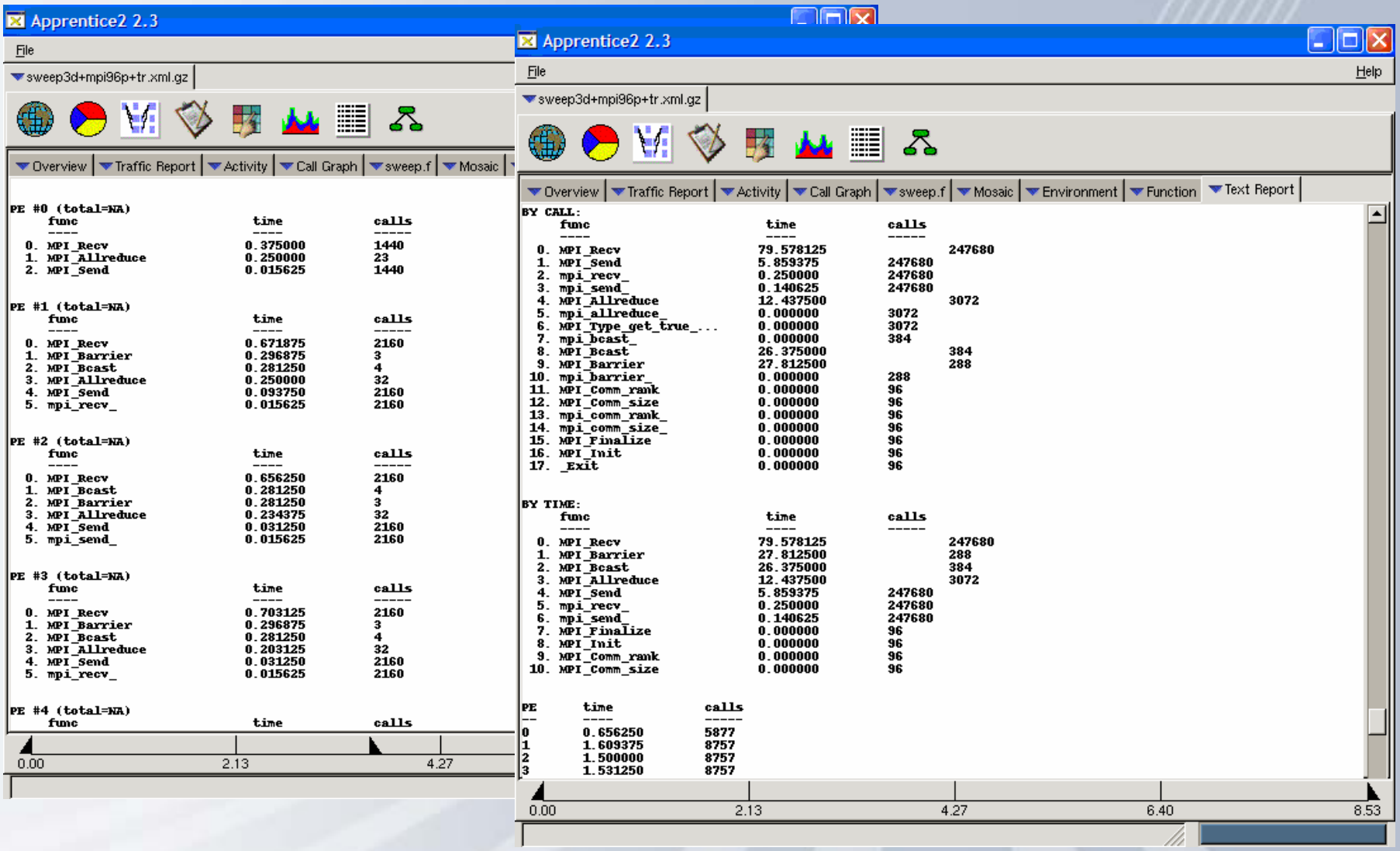


Function Profile

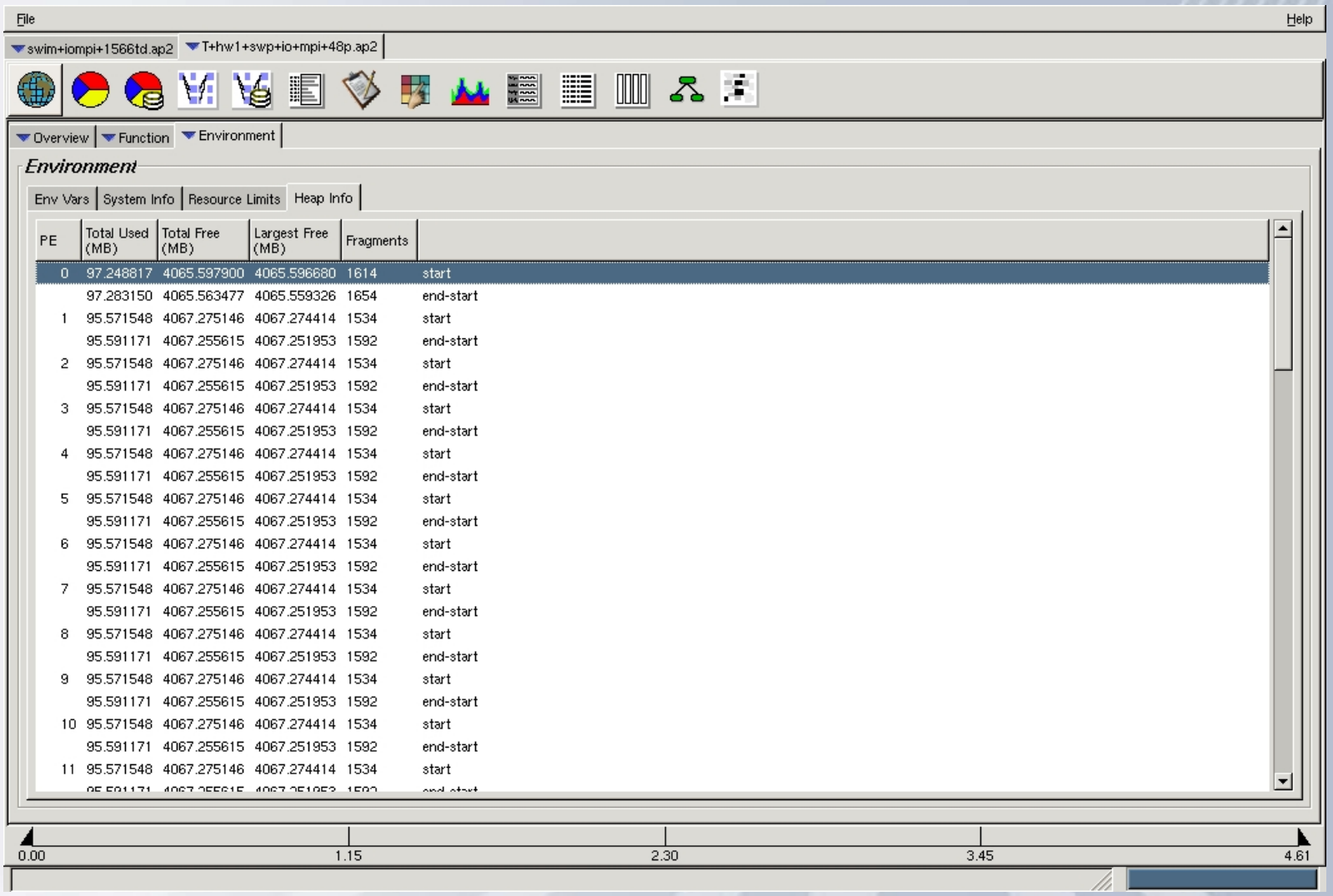


Time	Percent	Hits	Callsites	Imbalance %	Potential Savings	Function	Line	File
124.175511	63.29	576	1	5.63	0.15	sweep_	116	/lus/nid00036/ldr/Apps/sweep3d/sweep.f
40.211774	20.50	118080	1	23.40	0.25	mpi_recv_	0	==NA==
16.319527	8.32	48	1	48.26	0.30	exit	35	/notbackedup/users/rsrel/rs64.REL_1_4_33.060914.Thu/pe/computelibs/glibc/stdlib/exit.c
6.173236	3.15	1536	3	50.00	0.12	mpi_allreduce_	0	==NA==
2.760376	1.41	118080	1	17.58	0.01	mpi_send_	0	==NA==
2.250029	1.15	576	1	2.62	0.00	source_	18	/lus/nid00036/ldr/Apps/sweep3d/source.f
1.984620	1.01	144	1	2.59	0.00	mpi_barrier_	0	==NA==
0.867359	0.44	192	2	2.47	0.00	mpi_bcast_	0	==NA==
0.416231	0.21	576	1	2.60	0.00	flux_err_	17	/lus/nid00036/ldr/Apps/sweep3d/flux_err.f
0.382130	0.19	118080	2	10.98	0.00	snd_real_	135	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.237772	0.12	309	1	95.76	0.07	fwrite	36	/notbackedup/users/rsrel/rs64.REL_1_4_33.060914.Thu/pe/computelibs/glibc/libio/iofwrite.c
0.185067	0.09	118080	2	17.30	0.00	rcv_real_	164	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.067832	0.03	48	1	4.56	0.00	initialize_	42	/lus/nid00036/ldr/Apps/sweep3d/initialize.f
0.059407	0.03	48	1	4.99	0.00	initxs_	77	/lus/nid00036/ldr/Apps/sweep3d/initialize.f
0.041371	0.02	48	1	23.84	0.00	inner_	72	/lus/nid00036/ldr/Apps/sweep3d/inner.f
0.023948	0.01	8	1		0.00	fputc	35	/notbackedup/users/rsrel/rs64.REL_1_4_33.060914.Thu/pe/computelibs/glibc/libio/fputc.c
0.016902	0.01	68	1		0.00	getc	36	/notbackedup/users/rsrel/rs64.REL_1_4_33.060914.Thu/pe/computelibs/glibc/libio/getc.c
0.008104	0.00	4992	2	28.14	0.00	octant_	17	/lus/nid00036/ldr/Apps/sweep3d/octant.f
0.002457	0.00	576	1	18.79	0.00	global_real_max_	321	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.002083	0.00	48	1	69.55	0.00	MAIN_	72	/lus/nid00036/ldr/Apps/sweep3d/driver.f
0.001588	0.00	576	1	39.10	0.00	global_int_sum_	373	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.001393	0.00	48	1	10.23	0.00	inner_auto_	69	/lus/nid00036/ldr/Apps/sweep3d/inner_auto.f
0.000982	0.00	48	1	97.74	0.00	task_init_	24	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.000739	0.00	384	2	27.87	0.00	global_real_sum_	347	/lus/nid00036/ldr/Apps/sweep3d/mpi_stuff.f
0.000662	0.00	2	1		0.00	fopen	106	/notbackedup/users/rsrel/rs64.REL_1_4_33.060914.Thu/pe/computelibs/glibc/libio/iofopen.c
0.000499	0.00	48	1	7.54	0.00	initsnc_	175	/lus/nid00036/ldr/Apps/sweep3d/initialize.f

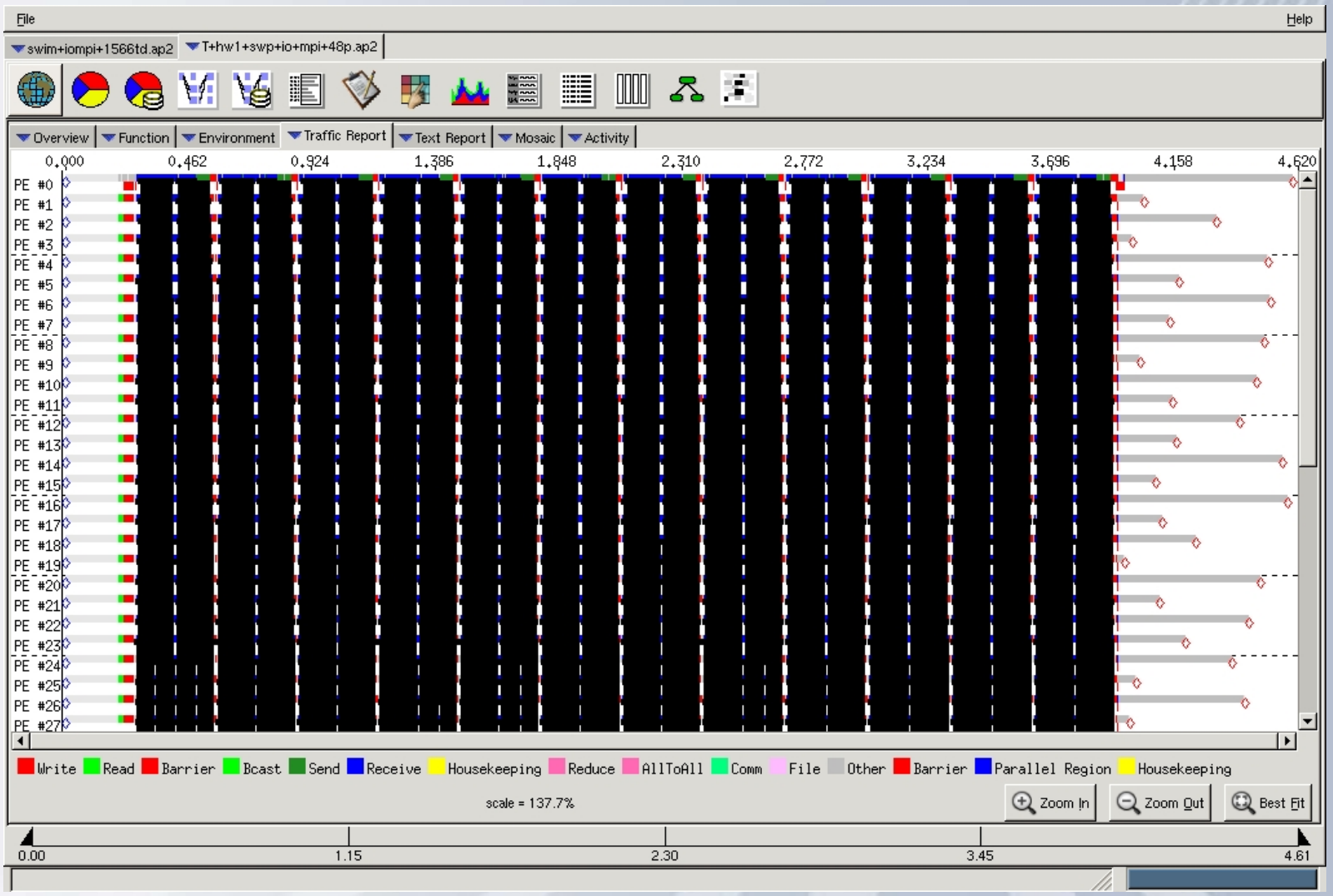
Distribution by PE, by Call, & by Time



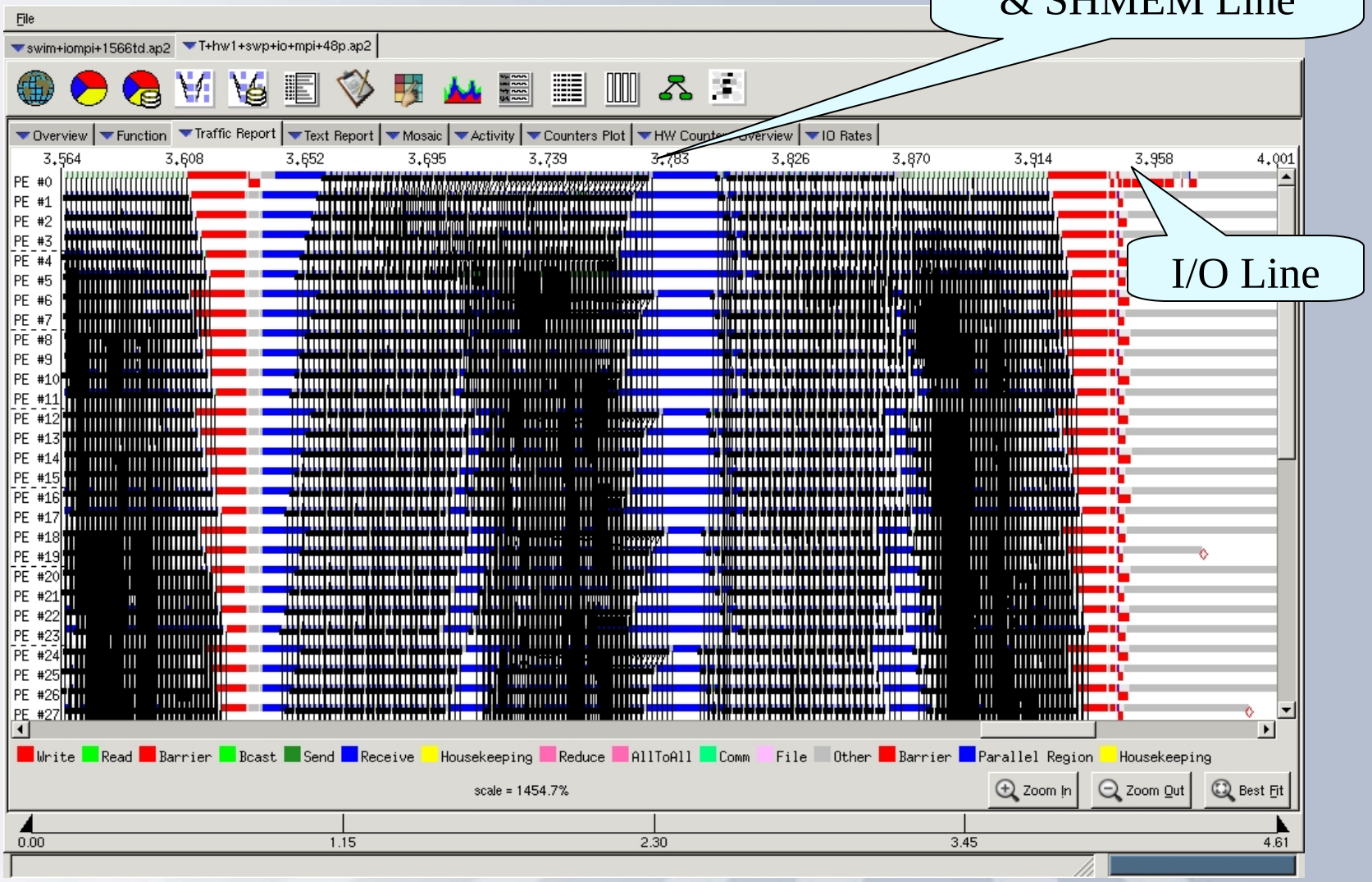
Environment & Execution Details



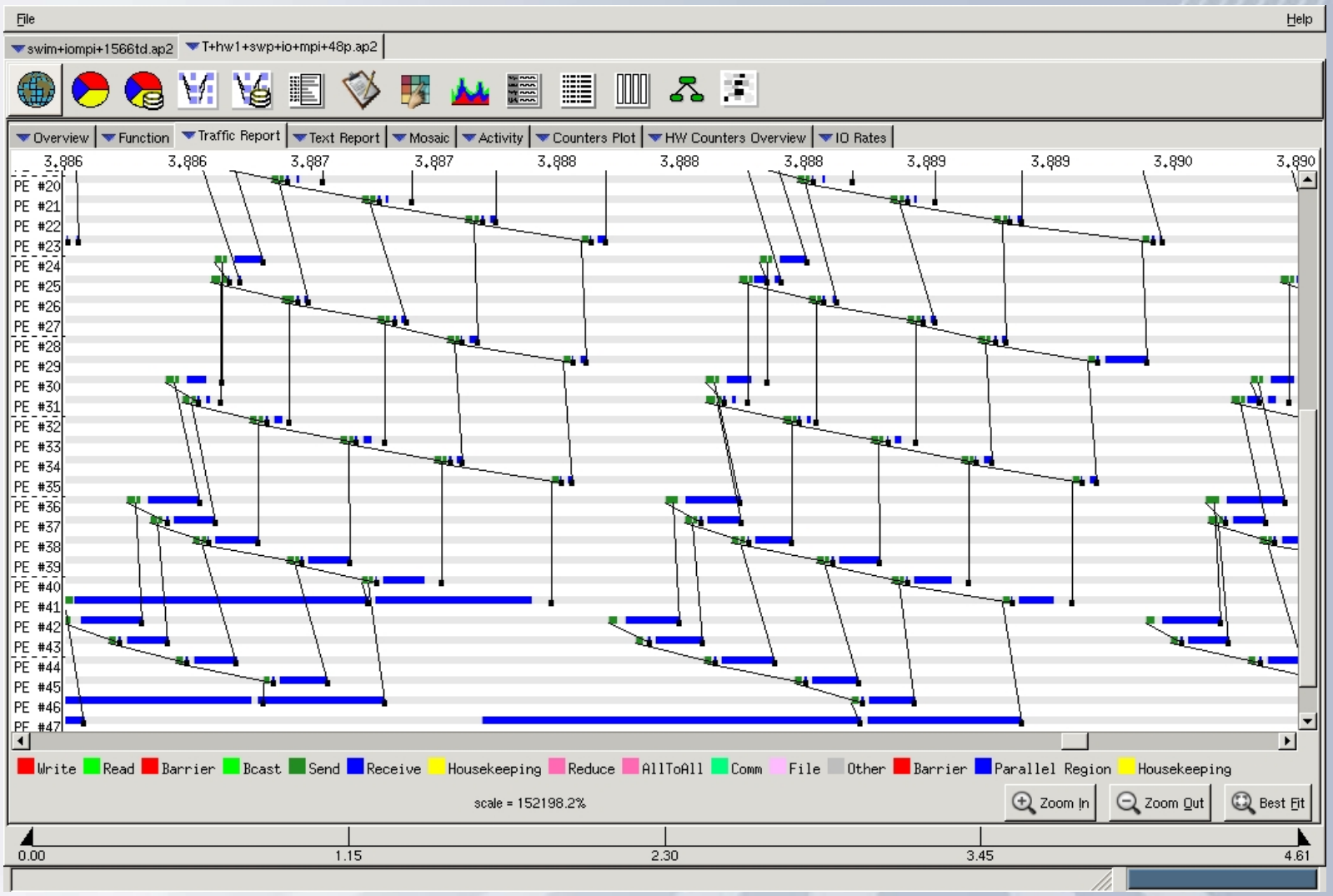
Time Line View (Sweep3D)



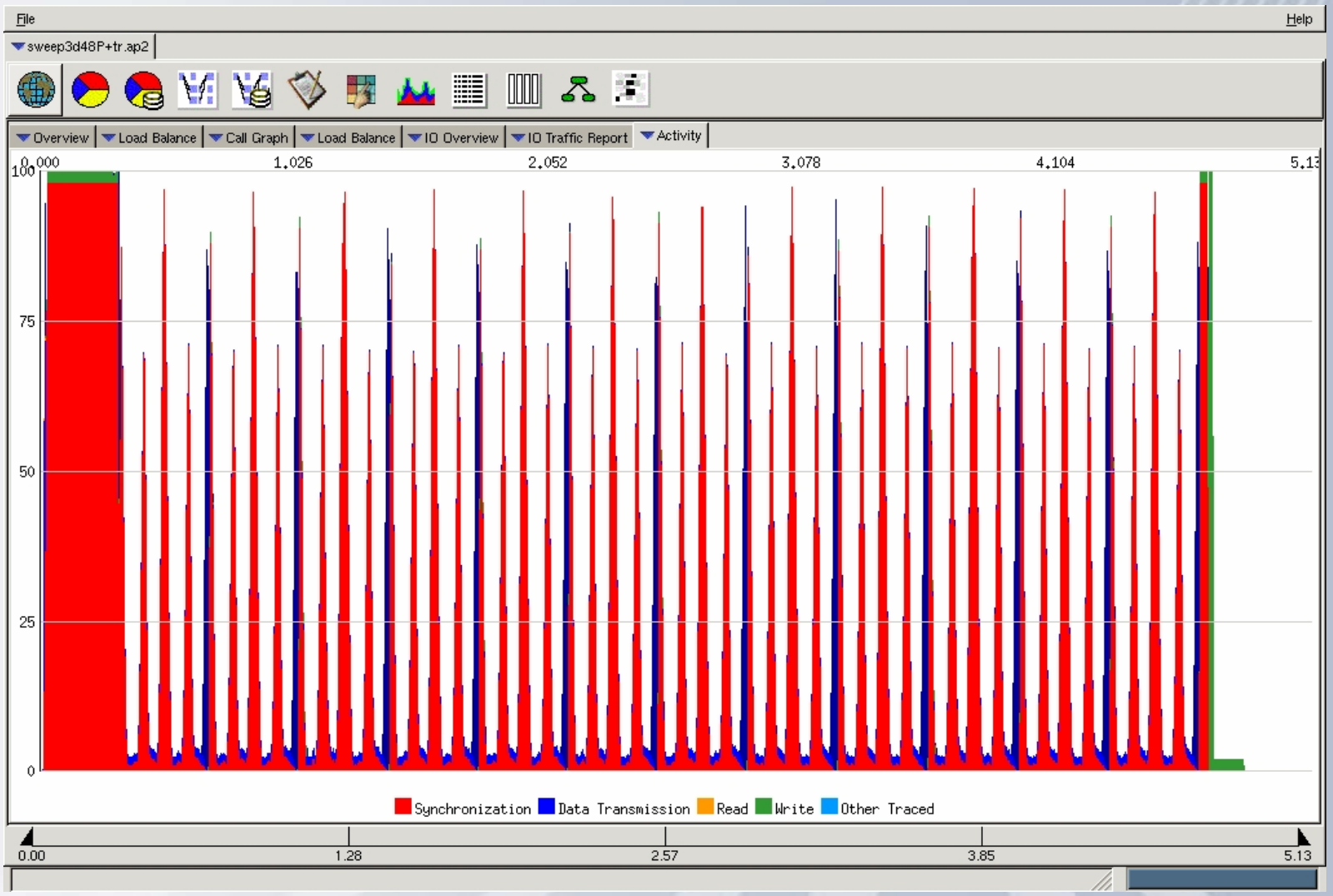
Time Line View (Zoom)



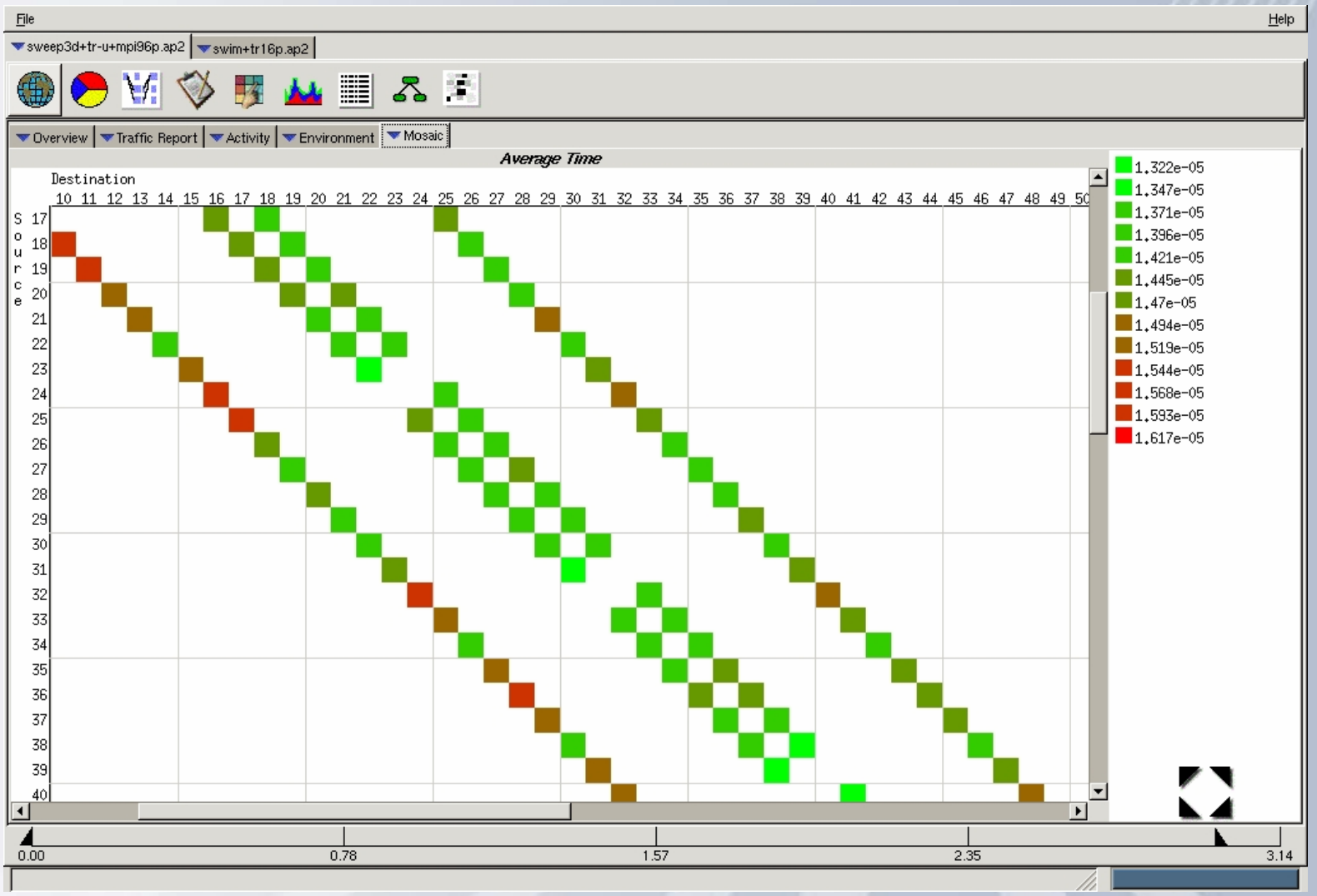
Time Line View (Fine Grain Zoom)



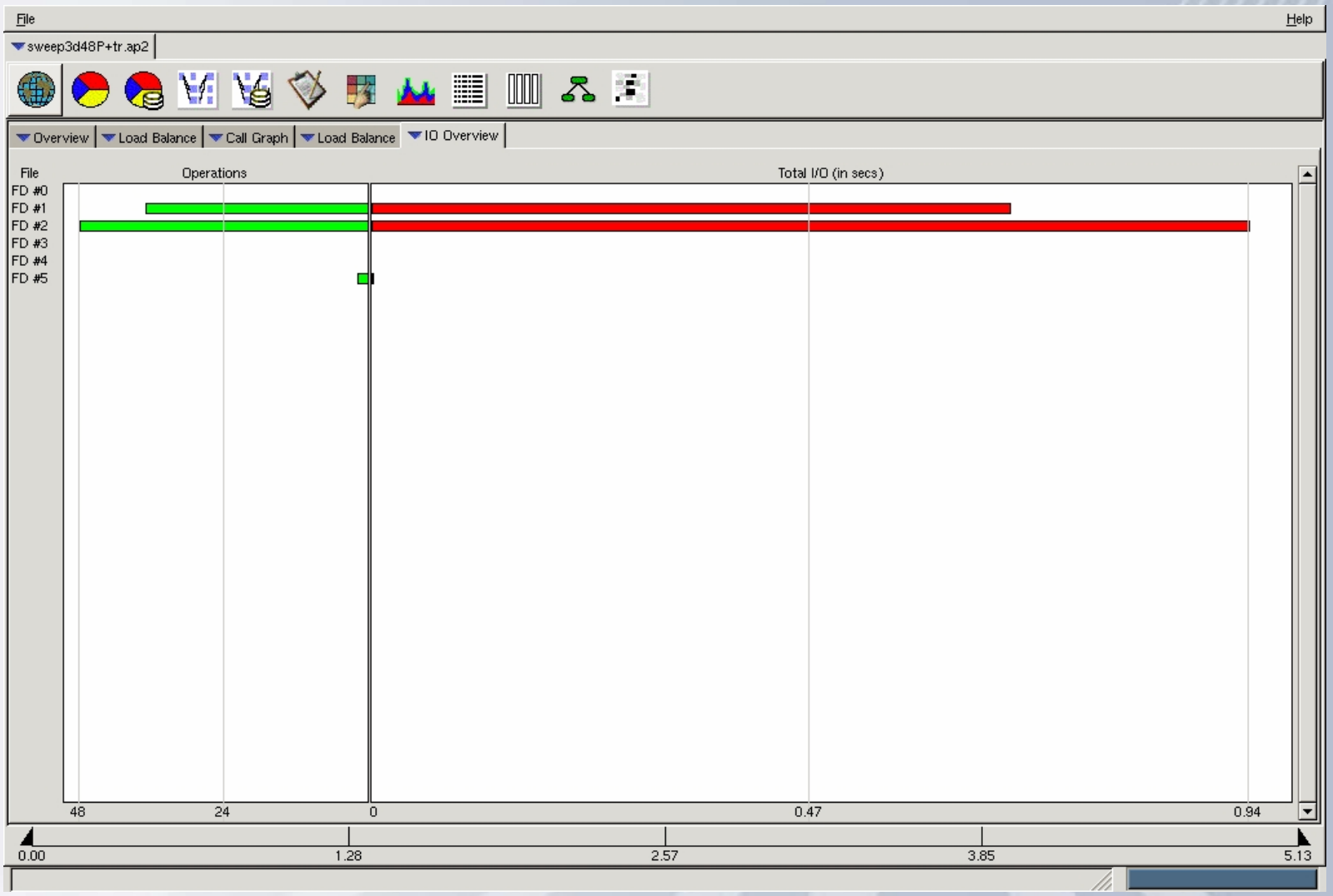
Activity View



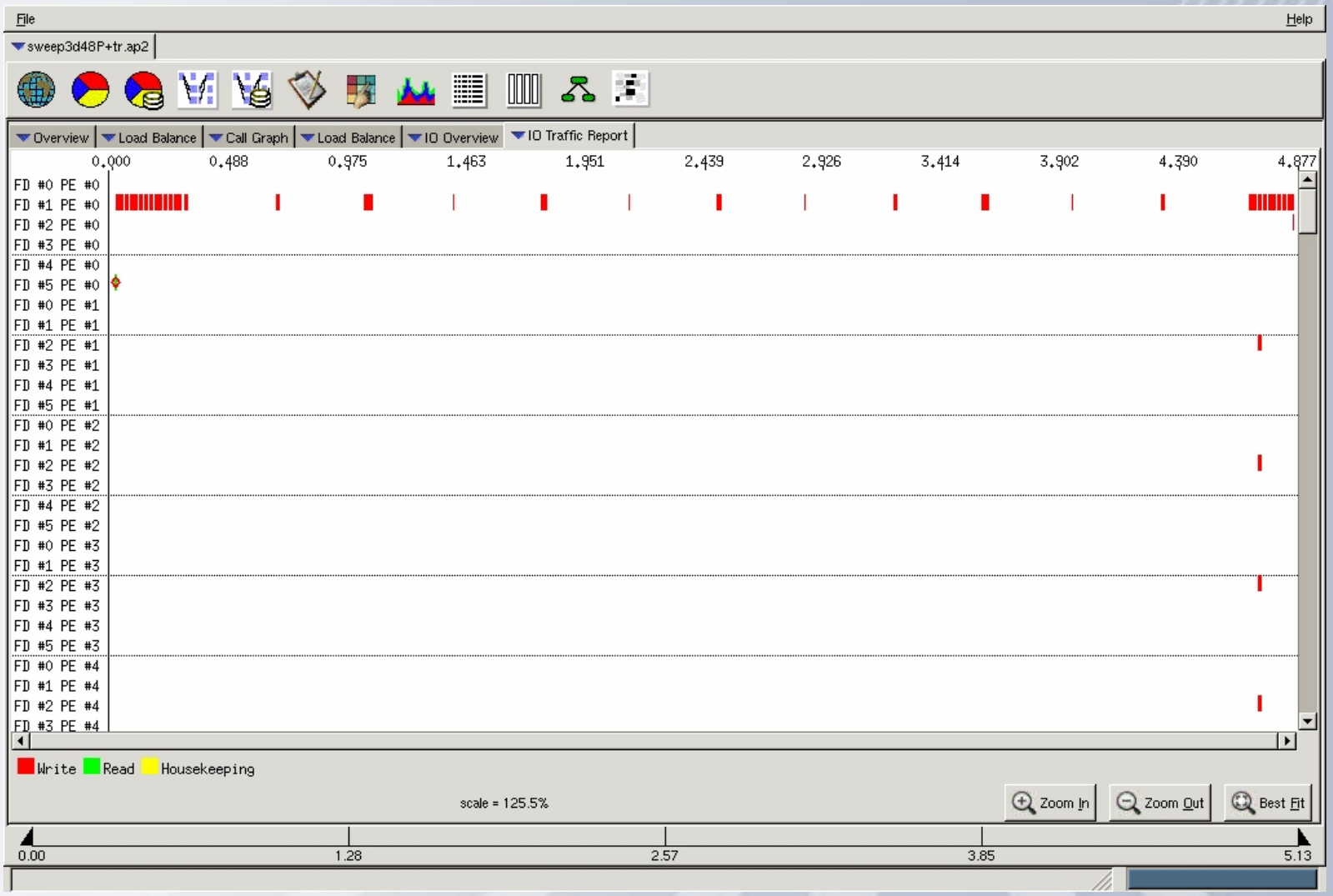
Pair-wise Communication



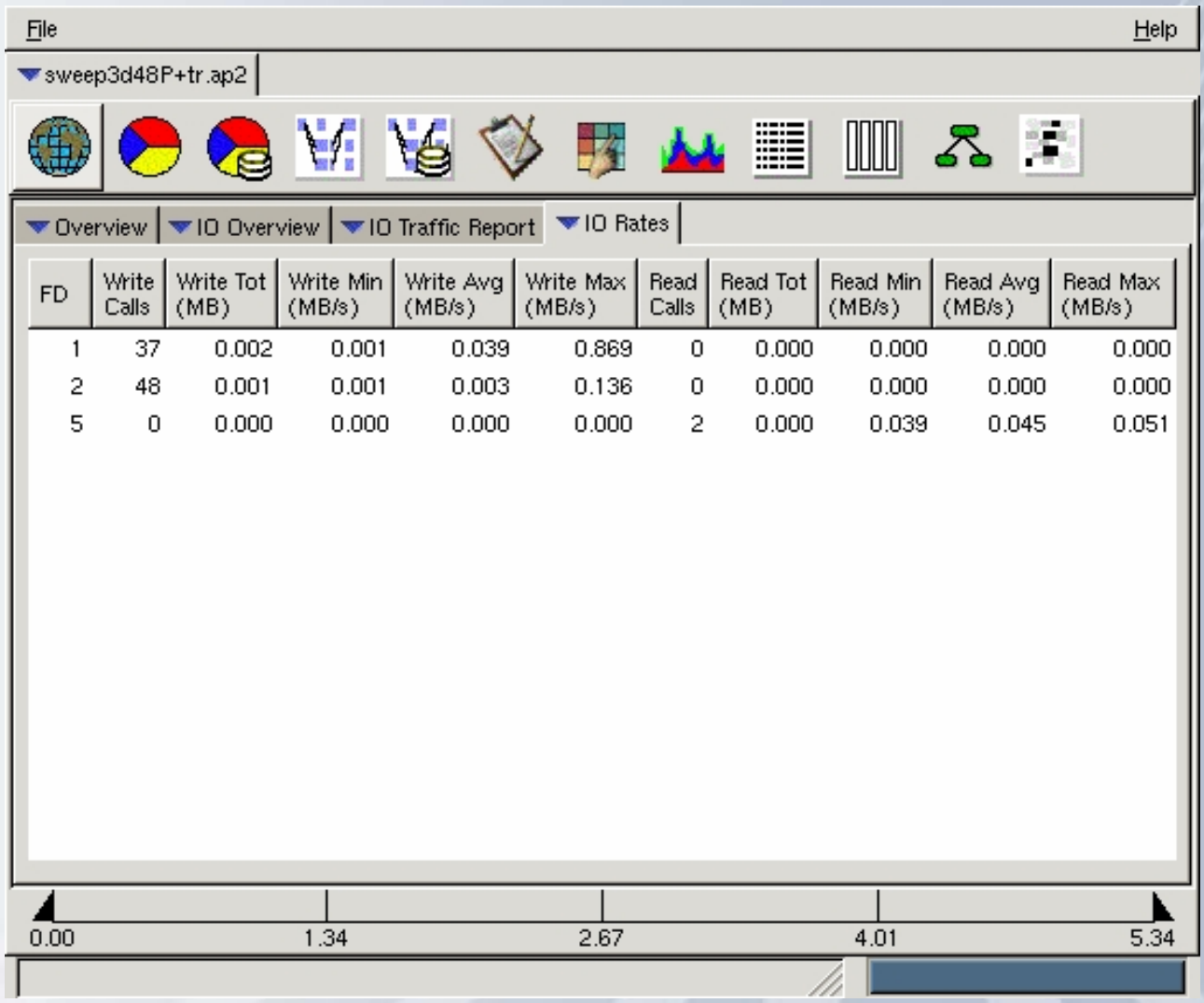
I/O Overview


File
Help
sweep3d48P+tr.ap2
Overview
Load Balance
Call Graph
Load Balance
IO Overview
File
Operations
Total I/O (in secs)
FD #0
FD #1
FD #2
FD #3
FD #4
FD #5
48
24
0
0.47
0.94
0.00
1.28
2.57
3.85
5.13

I/O Traffic Report

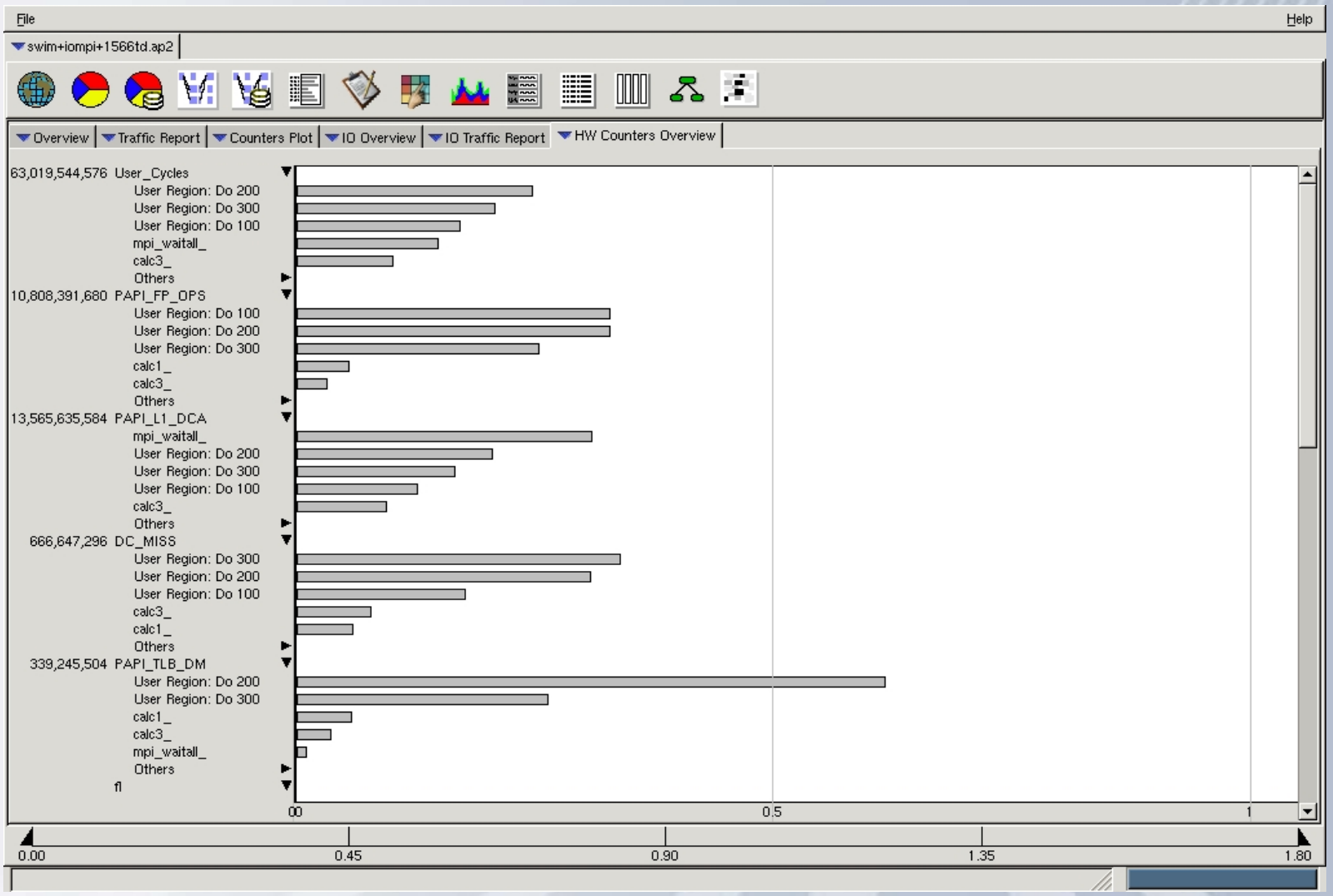


I/O Rates

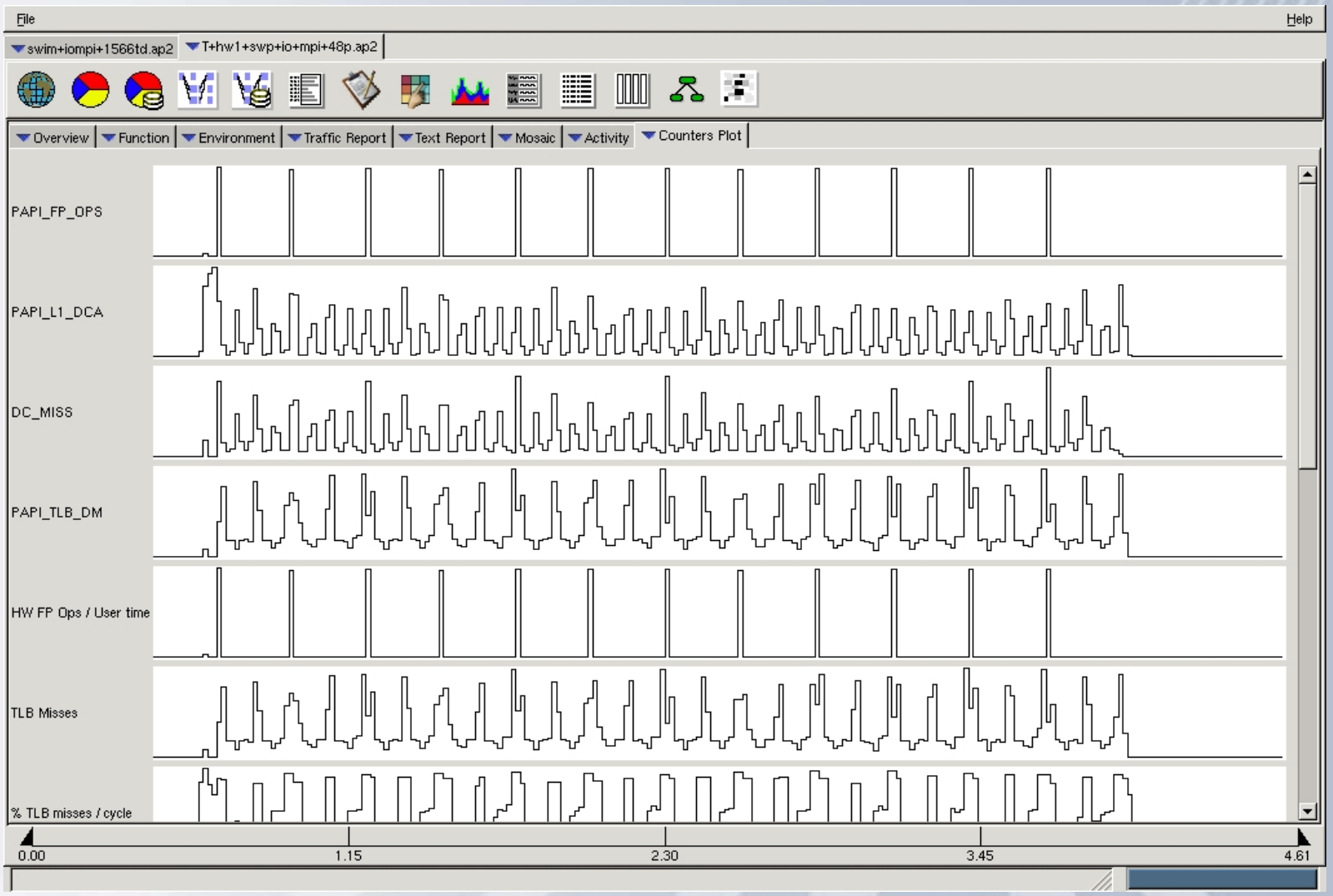


FD	Write Calls	Write Tot (MB)	Write Min (MB/s)	Write Avg (MB/s)	Write Max (MB/s)	Read Calls	Read Tot (MB)	Read Min (MB/s)	Read Avg (MB/s)	Read Max (MB/s)
1	37	0.002	0.001	0.039	0.869	0	0.000	0.000	0.000	0.000
2	48	0.001	0.001	0.003	0.136	0	0.000	0.000	0.000	0.000
5	0	0.000	0.000	0.000	0.000	2	0.000	0.039	0.045	0.051

Hardware Counters Overview



Hardware Counters Time Line



Controlling Trace File Size

- Several environment variables are available to limit trace files to a reasonable size:
  - PAT_RT_CALLSTACK
    - Limit the depth to trace the call stack
  - PAT_RT_HWPC
    - Avoid collecting hardware counters (unset)
  - PAT_RT_RECORD_PE
    - Collect trace for a subset of the PEs
  - PAT_RT_TRACE_FUNCTION_ARGS
    - Limit the number of function arguments to be traced
  - PAT_RT_TRACE_FUNCTION_LIMITS
    - Avoid tracing indicated functions
  - PAT_RT_TRACE_FUNCTION_MAX
    - Limit the maximum number of traces generated for all functions for a single process
- Use CrayPat API to toggle trace state (on / off) or to select functions to trace
  - int PAT_tracing_state (int state)
  - int PAT_trace_function (const void *addr, int state)
- Use the limit built-in command for ksh(1) or csh(1) to control how much disk space the trace file can consume

Additional API Functions

- int PAT_profiling_state (int state)
- int PAT_record (int state)
- int PAT_sampling_state (int state)
- int PAT_tracing_state (int state)
- int PAT_trace_function (const void *addr, int state)
- State can have one of the following:
  - PAT_STATE_ON
  - PAT_STATE_OFF
  - PAT_STATE_QUERY
- int PAT_flush_buffer (void)

Parallel Performance Visualization on the Cray XT4

Questions / Comments

Thank You!

NERSC
September 18-20, 2007

CRAY
THE SUPERCOMPUTER COMPANY