Obtaining Extremely Detailed Information at Scale

Jesus Labarta
Barcelona Supercomputing Center

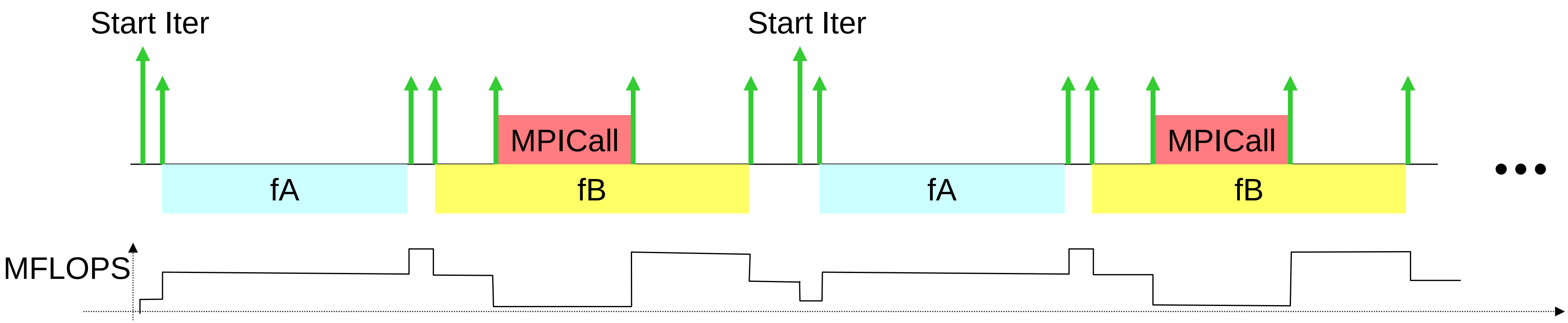
Topics

- Instrumentation + Sampling
  - Objective
    - Can we get very fine grain Information …
    - … of time distribution of metrics …
    - … with little overhead?
  - Work by Harald Servat

- Use of MRNET
  - Miminize the captured data/information ratio
    - Statistics, profile, traces
  - Work by German Llort

Instrumentation

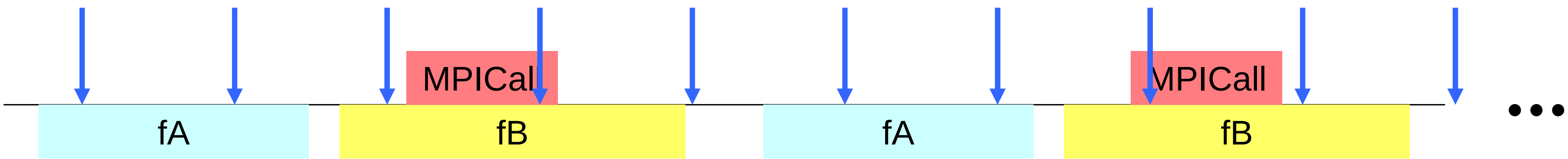
- Events correlated to specific program activity
  - Start/exit iterations, functions, loops,…



- Different intervals:
  - May be very large, may be very short
  - Variable precision
- Captured data:
  - Hardware counters, call arguments, call path,….

Sampling

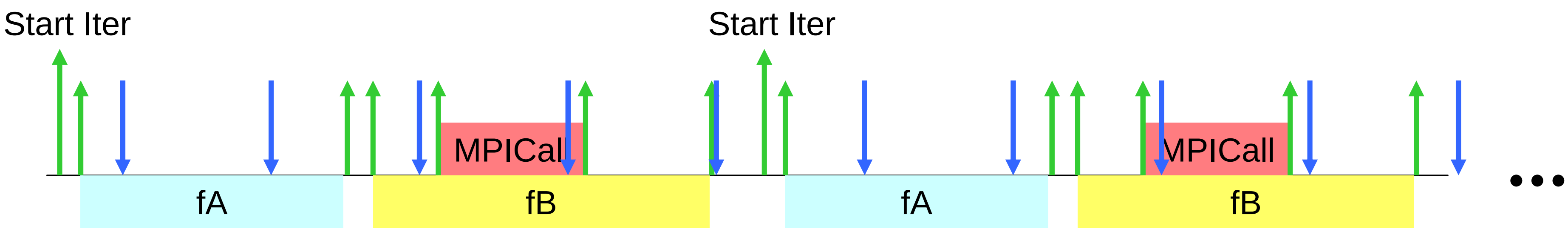
- Events uncorrelated to program activity (at least not specific)
  - Time (or counter) overflow



- Controlled granularity:
  - Sufficiently large to minimize overhead
  - Guaranteed acquisition interval/precision
- Statistical projection
  - %Counts = %time (or metric)
  - Assuming no correlation, sufficiently large #samples

Instrumentation + sampling

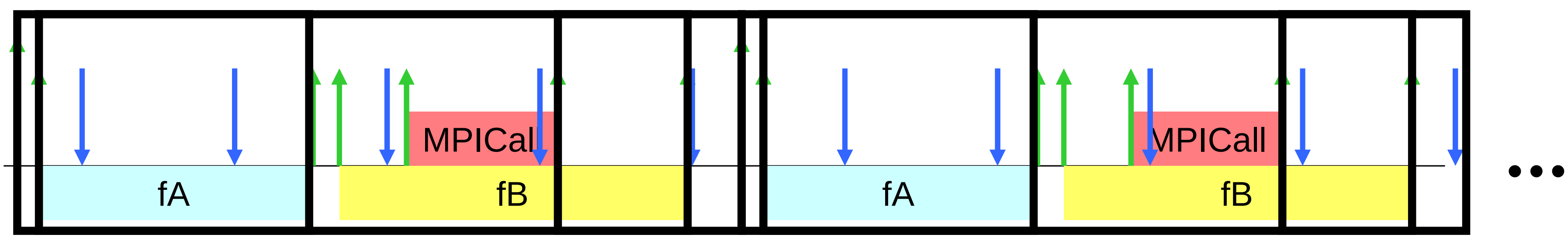
- Both



- Guaranteed interval
- Captured data:
  - Hardware counters (since previous probe)
  - call path
  - Call arguments in some probes
- How to use it?

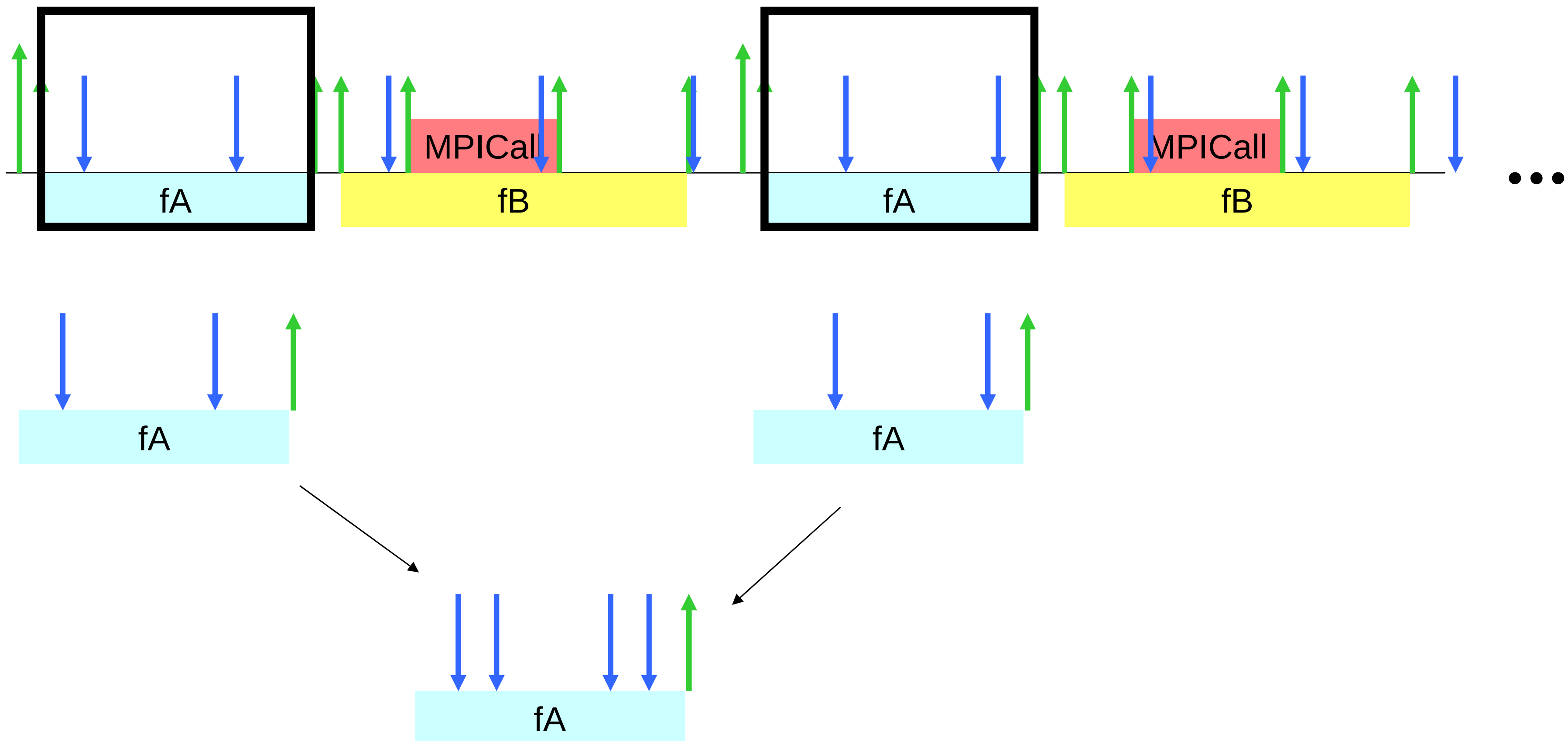
New roles

- Instrumentation → Reference
  - Identify different instances of a region for which to obtain detailed time evolution of metrics
    - Stationary behaviour assumed
  - Target region:
    - Iteration
    - Routine
    - Routine excluding MPI calls
    - …



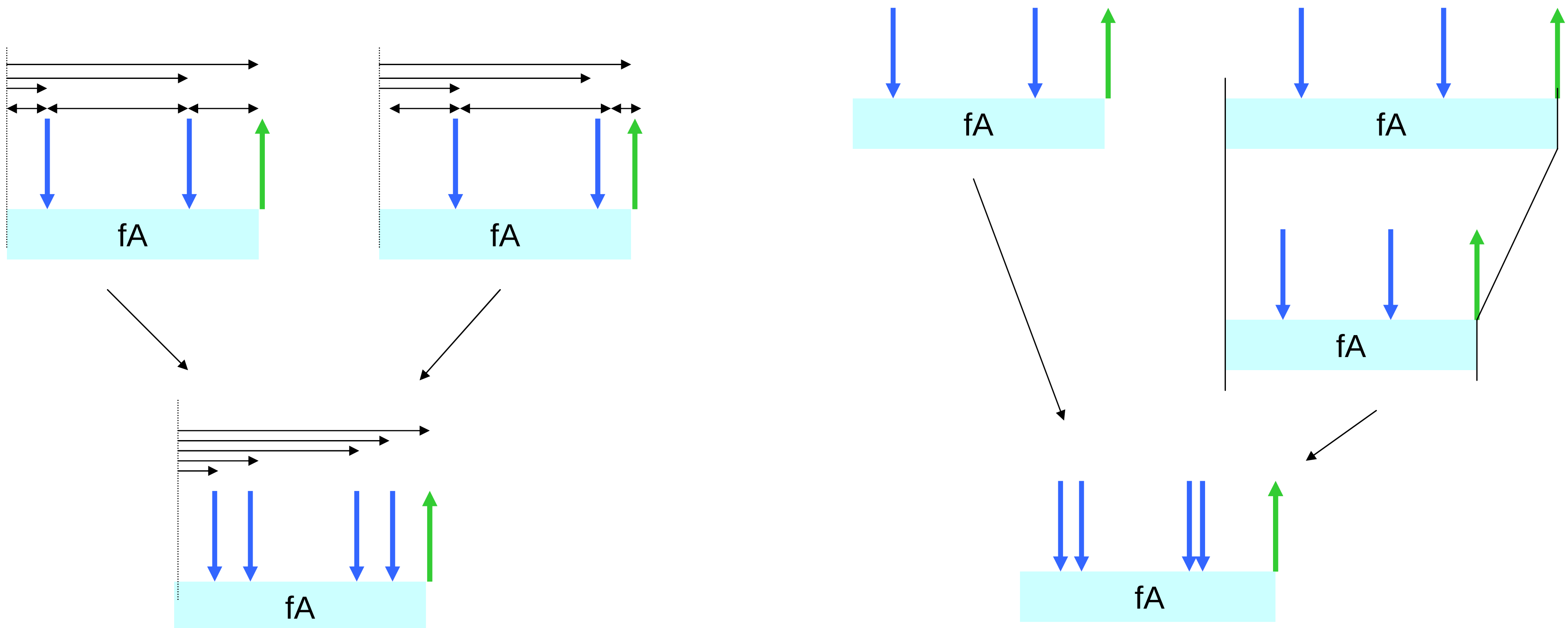
New roles

- Sampling role → relative data
  - Guarantee granularity
  - Provide data to increase granularity



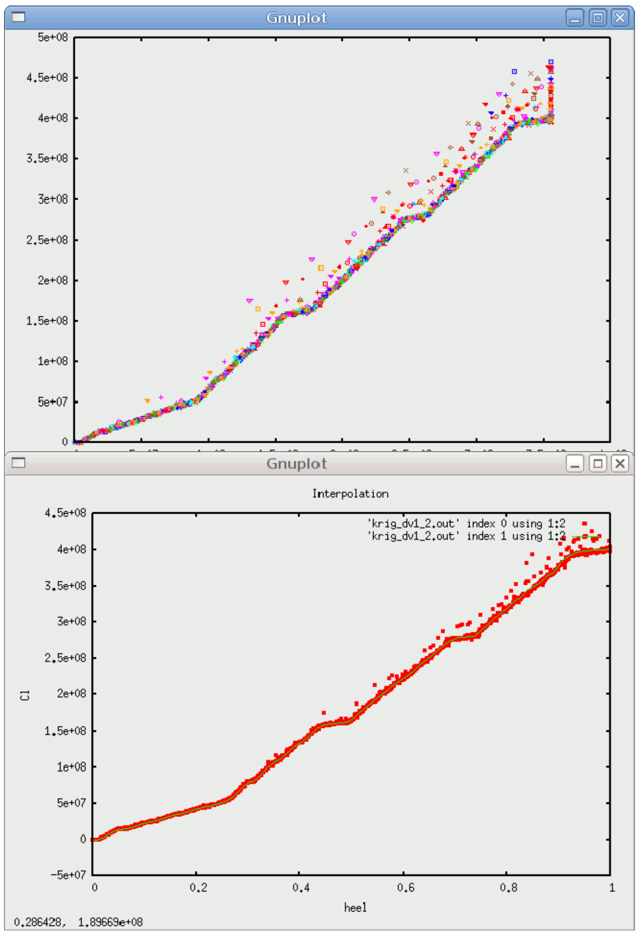
Folding counters: Projecting

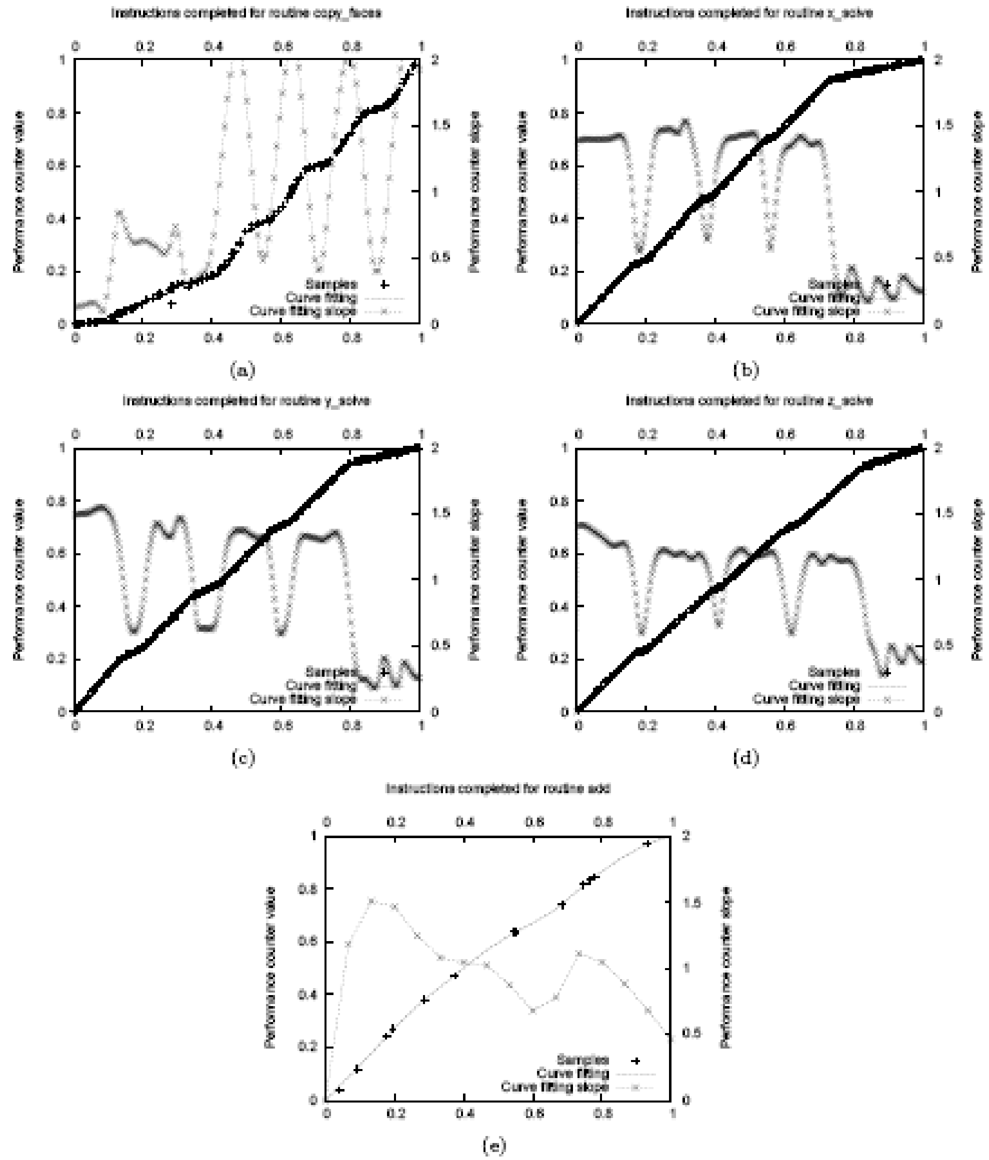
- Cumulative count since reference
- Internal variance
- Variance in duration
  - Eliminate outliers
  - Scale



Folding counters: Fitting

- Eliminate outliers
- Kriging interpolation

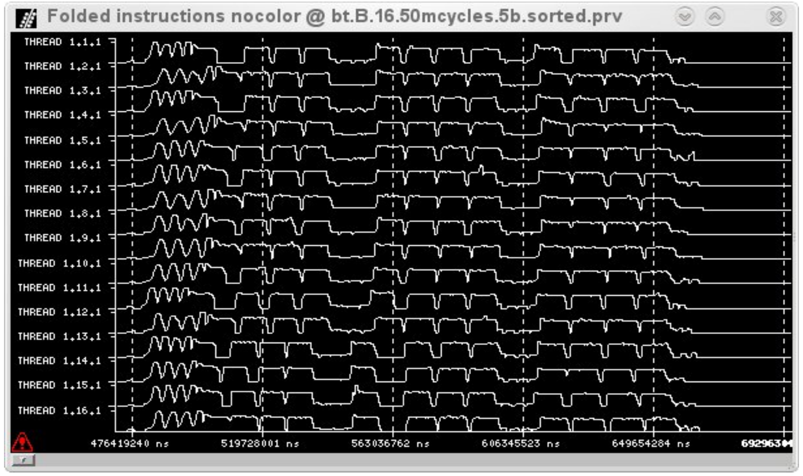


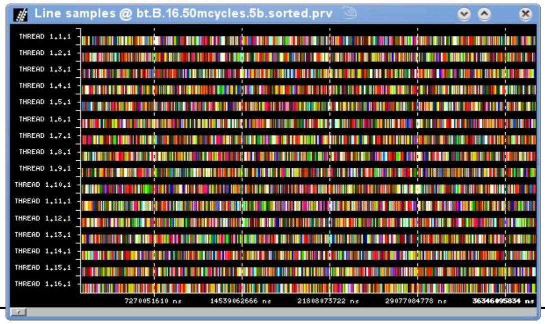


Normalized MIPS @ Routines in NAS BT

Timeline display

- Present
  - Analytical expression
  - Plots, statistics
    - Time, IPC,…
  - Timelines
- Performance counters:
  - Sample again fitted function and inject synthetic events into trace
- Call stack
  - Truncated by specifying routines of interest



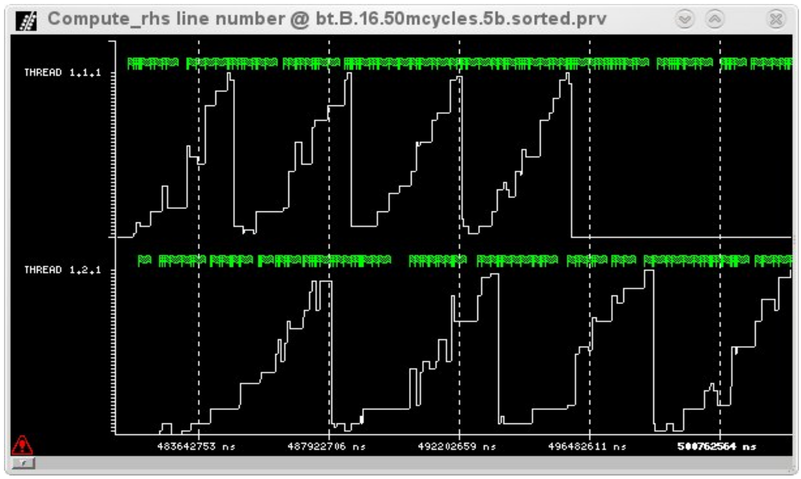


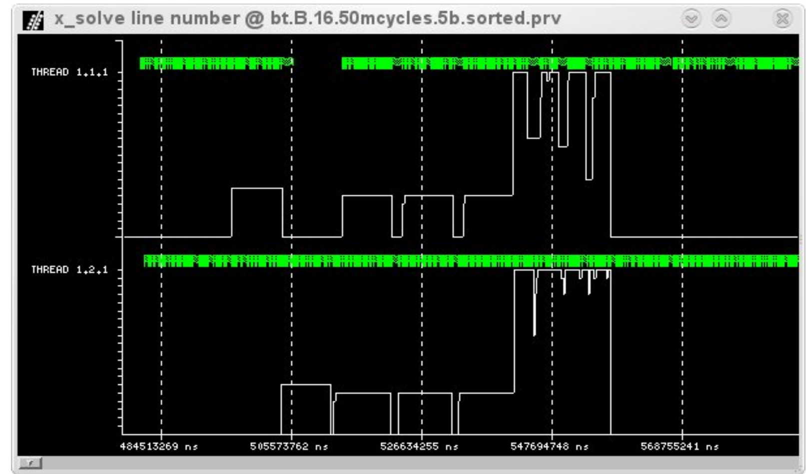
Lots of detail form a single run with low overhead

Compute_rhs

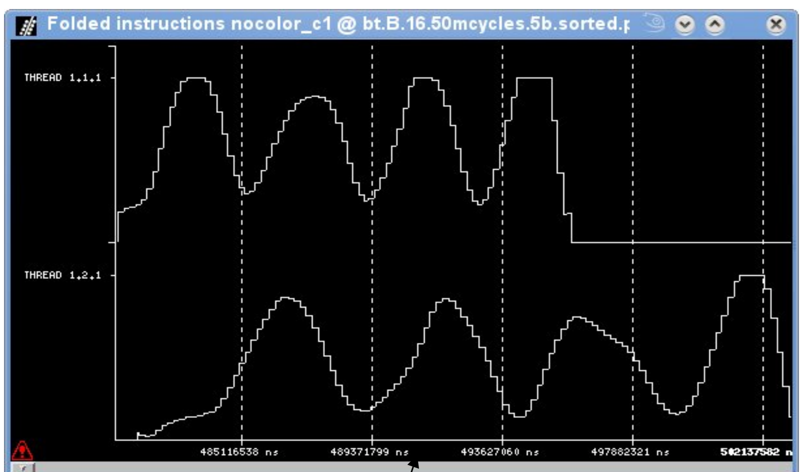
x_solve

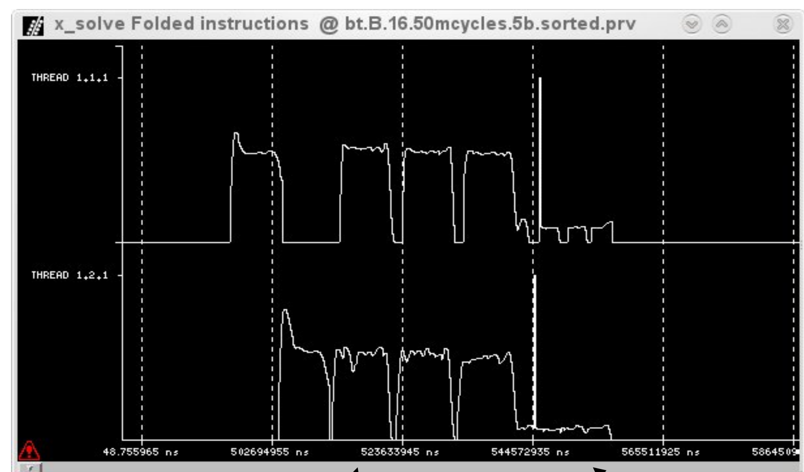
Source line





Instantanous MIPS





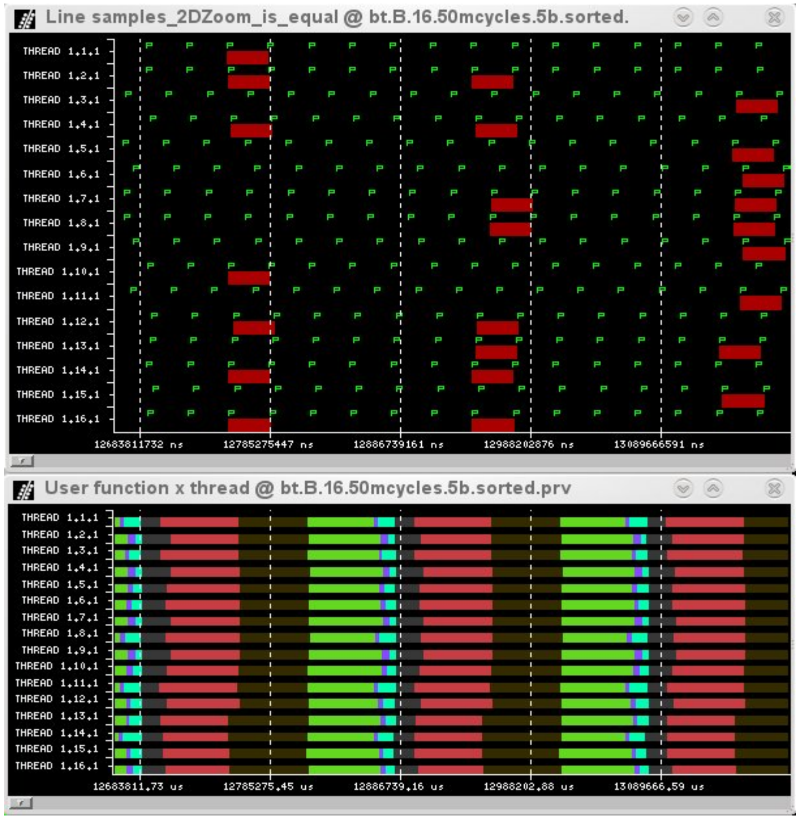
4 iterations outer loop

4 x_solve_cell

4 backsubstitute

Beware of correlations

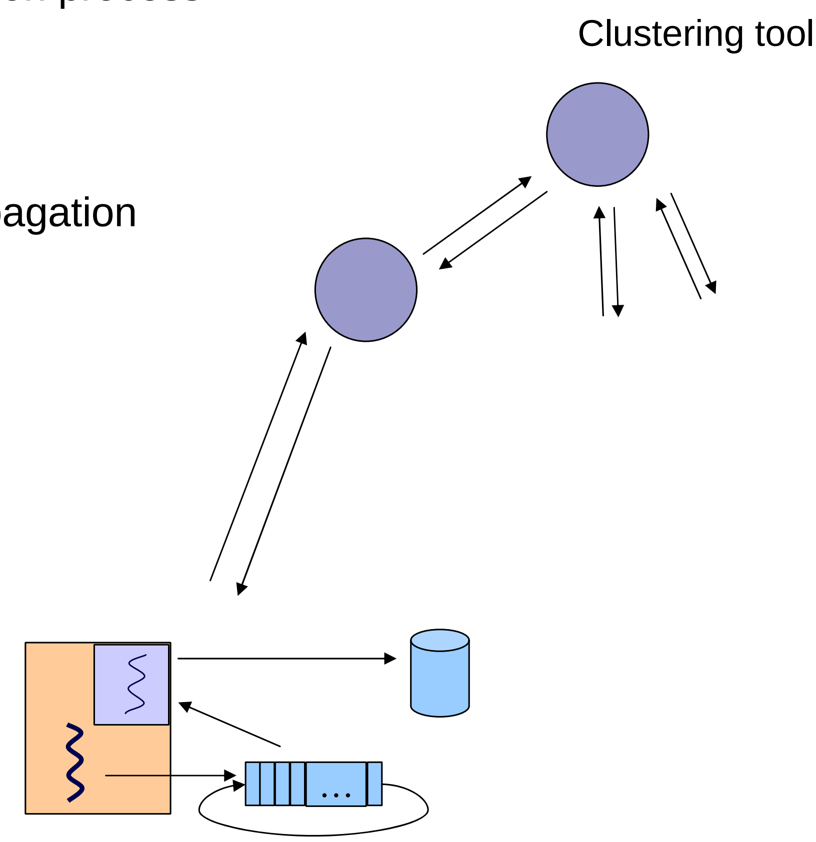
- 50 Mcycles seemed a reasonable interval
- Ends up being slightly correlated to application periodicity ☹
  - Can bias a pure statistical profile
  - Folding has potential to reduce this effect



Maximizing information/data ratio

Distributed trace control

- MRNet based mechanism
  - Local instrumentation on a circular buffer
  - Periodic MRNet front-end initiation of collection process
    - “Intelligent” determination of interval
  - Local algorithm
  - Reduction on tree, selection at root and propagation
    - Cluster a subset of all data
    - Classification of all data
    - Generation of global statistics
  - Locally emit trace events

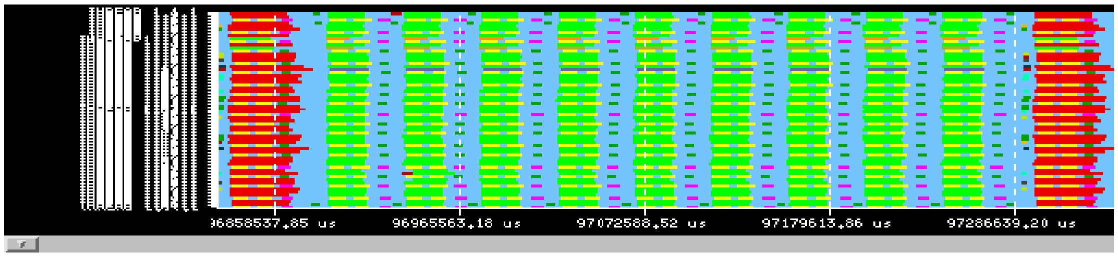


Clustering vs. classification

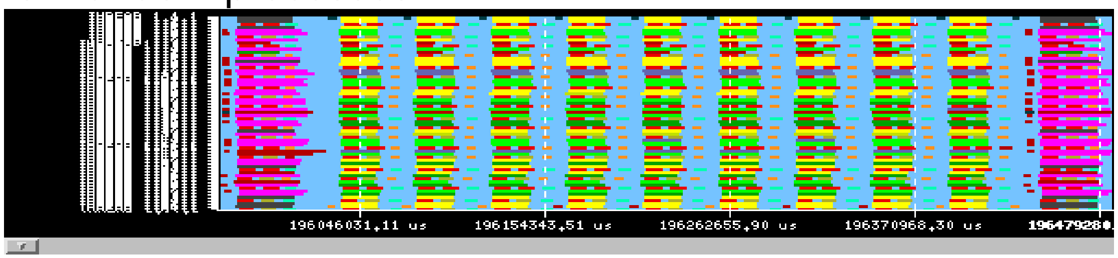
- Clustering time drastically grows with number of points
- Selection of a subset of data to clusterize
  - Space: Select a few processes. Full time sequence
  - Random sampling: wide covering
- Remaining data: “nearest” neighbor classification

GROMACS 64 processes

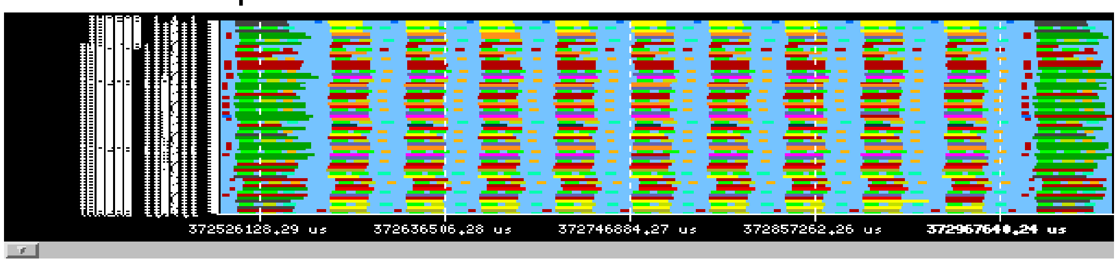
64 processes



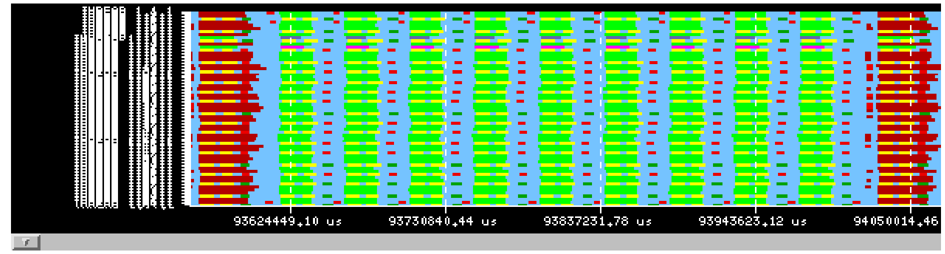
32 random processes



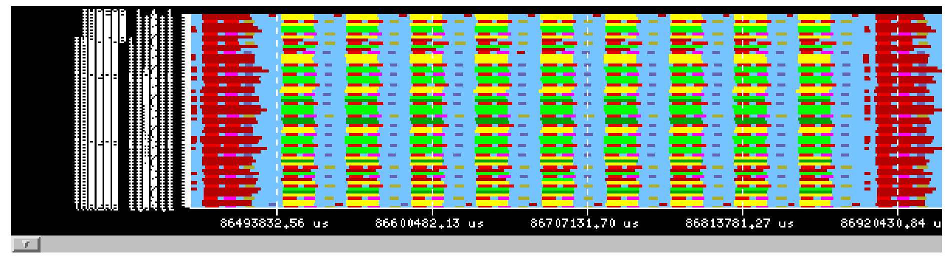
16 random processes



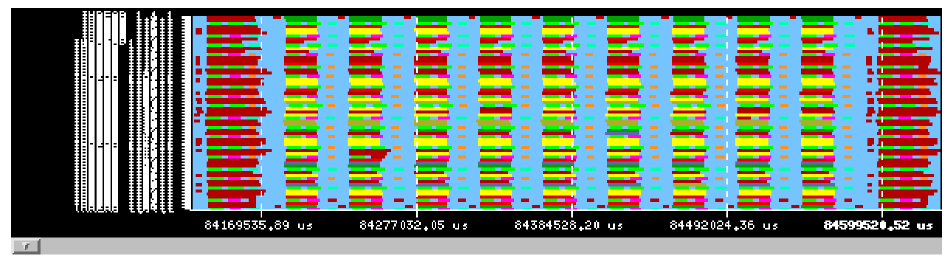
25% random records



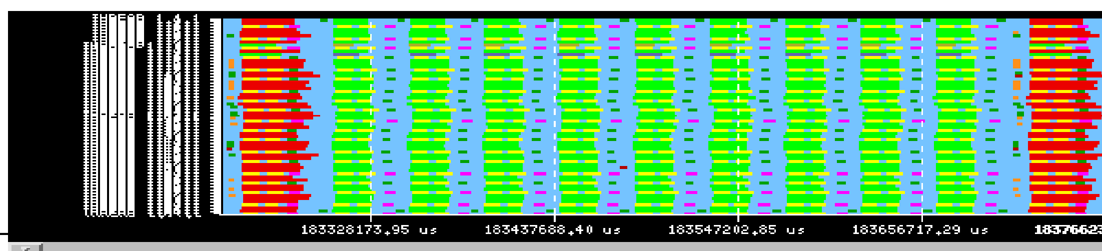
15% random records



10% random records



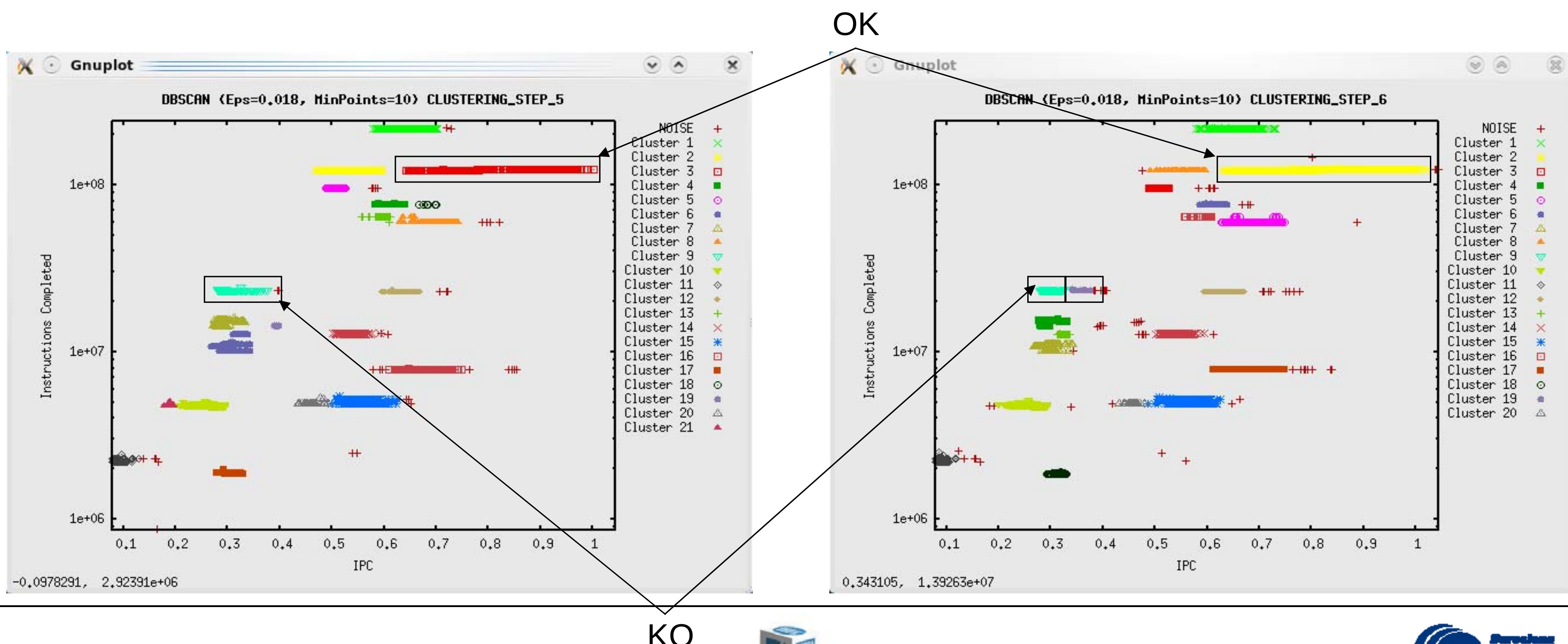
8 random processes + 15% random records



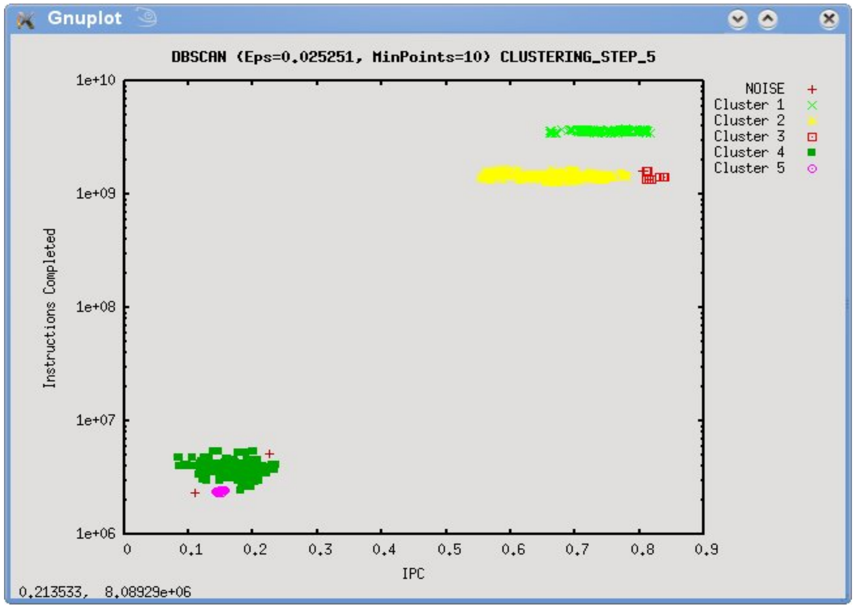
Good quality
Fast analysis

Tracking evolution

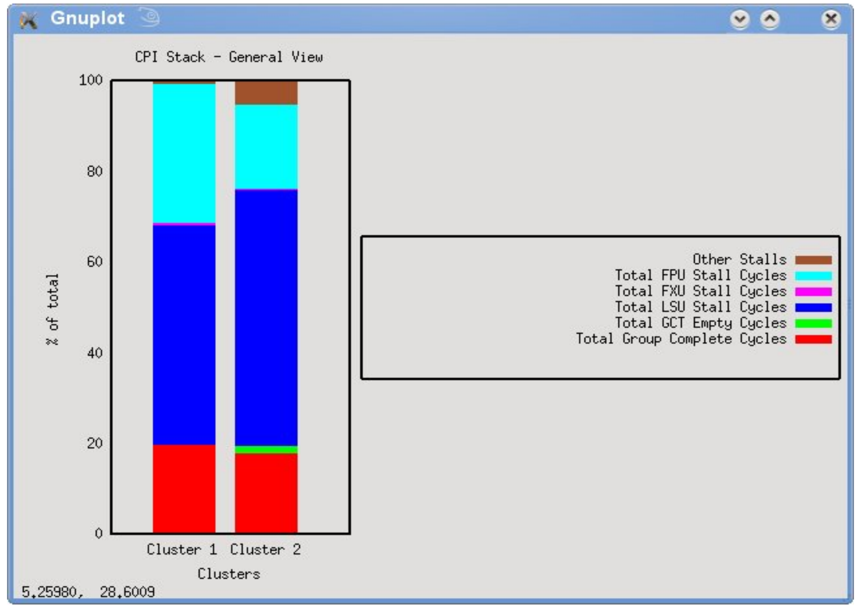
- Compare 2 clusterings cluster per cluster
  - Inscribe clusters into a rectangle with a 5% margin of variance. Match those that overlap.
  - Matched clusters must represent at least the 85% of the total computation time.
- Stability = 3 equivalent clusterings “in-a-row”.
- Requisites are gradually lowered if can not be matched



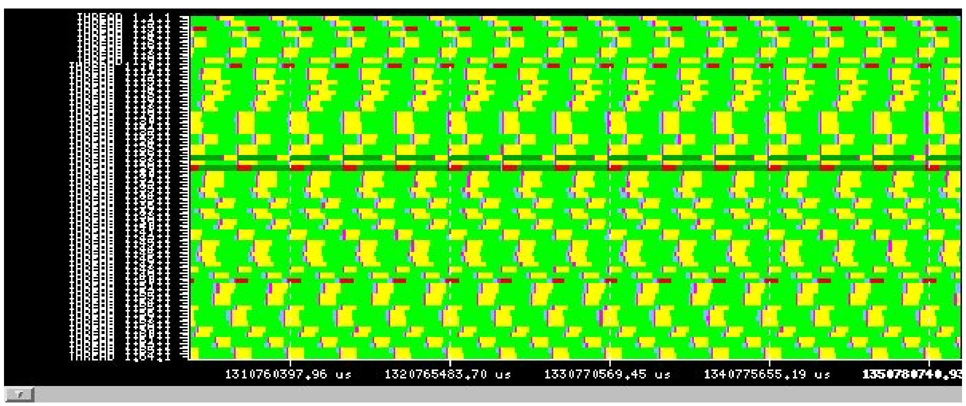
SPECFEM3D (64 tasks)

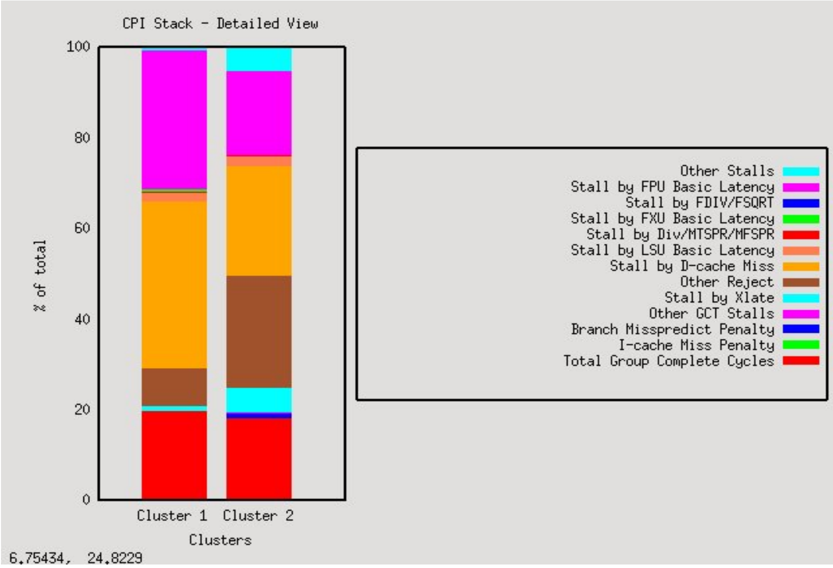


Clusters distribution



CPI STACK model





SPECFEM3D (64 tasks)

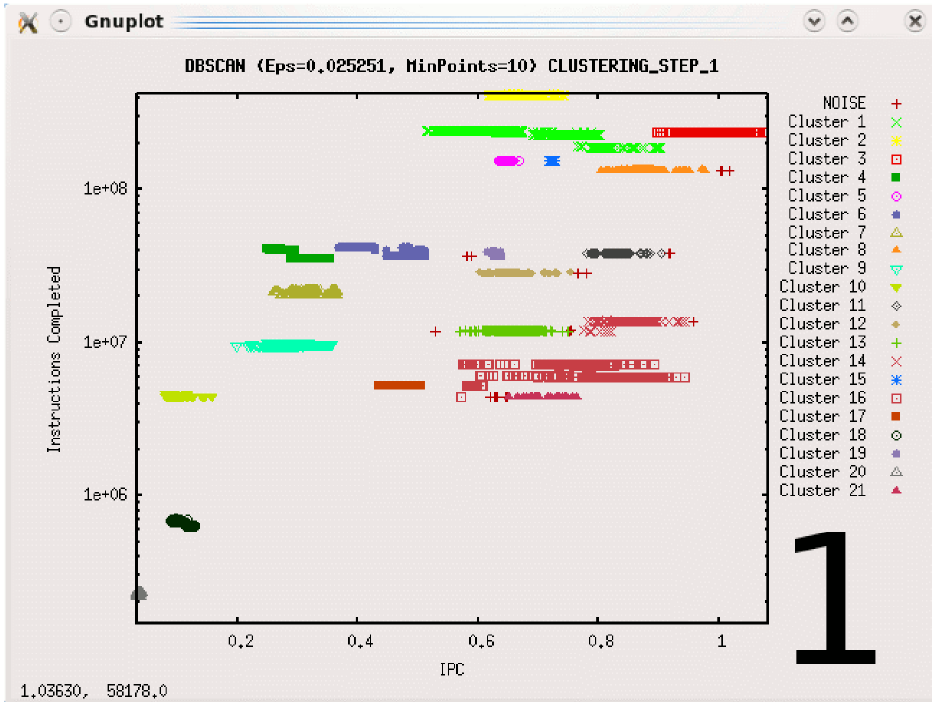
- Other statistics

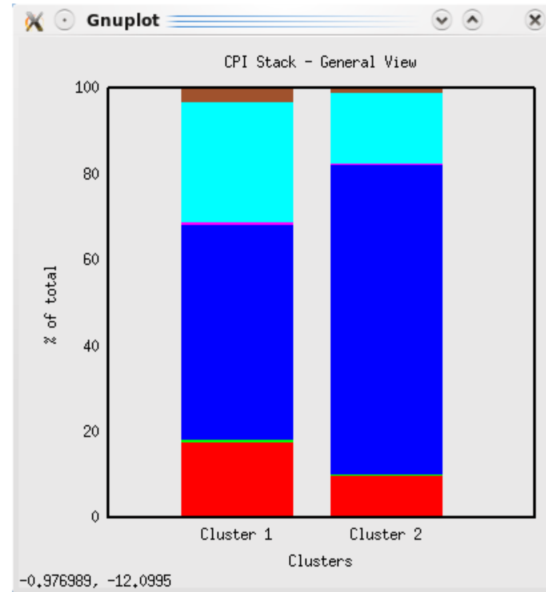
Category	Metric Description	Cluster 1	Cluster 2
Architecture	% Instr. Completed	32,25%	32,22%
Architecture	L1 misses per Kinstr.	41,37	34,82
Architecture	L2 misses per Kinstr.	1,194	1,060
Architecture	Bytes from maim memory per floating point instruction finished	1,649	1,361
Architecture	Number of Loads per Load miss	24,81	66,68
Architecture	Number of Stores per Store miss	6,74	5,27
Architecture	Number of Loads&Stores per L1 miss	18,81	33,28
Architecture	L1 cache hit rate	94,68%	97,00%
Architecture	Number of Loads per (D)TLB miss	11.403,59	5.873,04
Architecture	Number of Loads&Stores per (D)TLB miss	12.945,98	6.425,90
Architecture	% TLB misses per cycle	0,005%	0,012%
Architecture	Total Loads from local L2 (M) (total_ld_l_L2)	94,320	17,828
Architecture	Local L2 load traffic (MB)	12.073,007	2.281,926

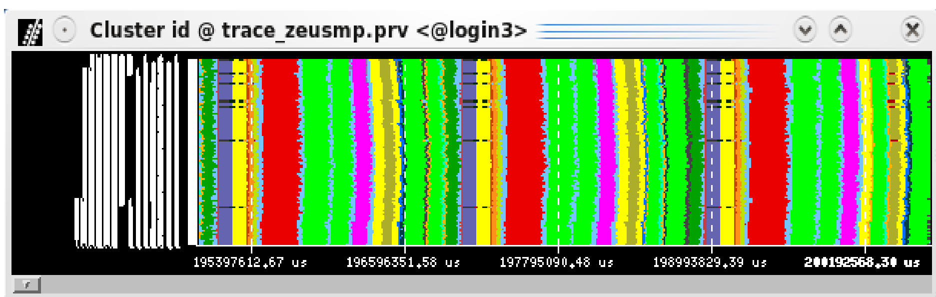
Category	Metric Description	Cluster 1	Cluster 2
Performance	% Duration	0.70236	0.26024
Performance	Avg. Burst Duration (µs)	2.142.664,27	947.044,53
Performance	Total preemted time (µs)	31394,19	13759,21
Performance	% preempted time	1,465%	1,453%
Performance	IPC	0,75	0,66
Performance	CPI	1,33	1,51
Performance	MIPs	1702,70	1501,75
Performance	Mem.BW (MB/s)	260,22	203,84
Performance	Memory instructions per second	1324,80	1740,67
Performance	HW floating point instructions per cycle	0,290	0,168
Performance	Flop rate (MFLOPs)	1.421,36	555,95
Performance	HW floating point instructions rate	656,95	381,15
Performance	Computation intensity	2,004	1,128
Performance	Local L2 load bandwidth per processor (MB/s)	5.634,58	2.409,52
Performance	% Loads from local L2 per cycle	2,039%	0,872%

Category	Metric Description	Cluster 1	Cluster 2
Instruction Mix	FMA ops per floating point instruction	0,763	0,633
Instruction Mix	Instructions per Load/Store	1,285	0,863
Instruction Mix	HW floating point instructions (flips)	1.407.628.943	360.968.332
Instruction Mix	Total floating point operations (flops)	3.045.492.593	526.507.674
Instruction Mix	Total FP Load&Store operations (fp_tot_ls)	1.519.651.784	466.816.330
Instruction Mix	FMA %	81,04%	82,15%
Instruction Mix	Memory Mix	25,10%	37,35%
Instruction Mix	Load Mix	22,11%	34,14%
Instruction Mix	Store Mix	2,99%	3,21%
Instruction Mix	FPU Mix	12,44%	8,18%
Instruction Mix	FXU Mix	3,93%	12,64%

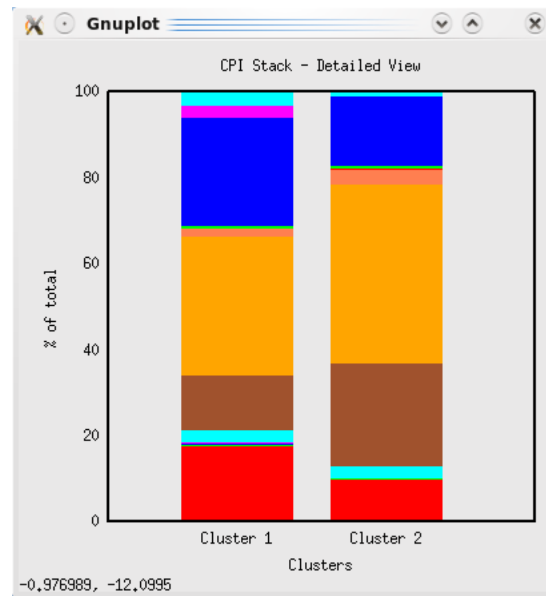
SPECMPI'07 ZEUSMP2 (128 tasks)



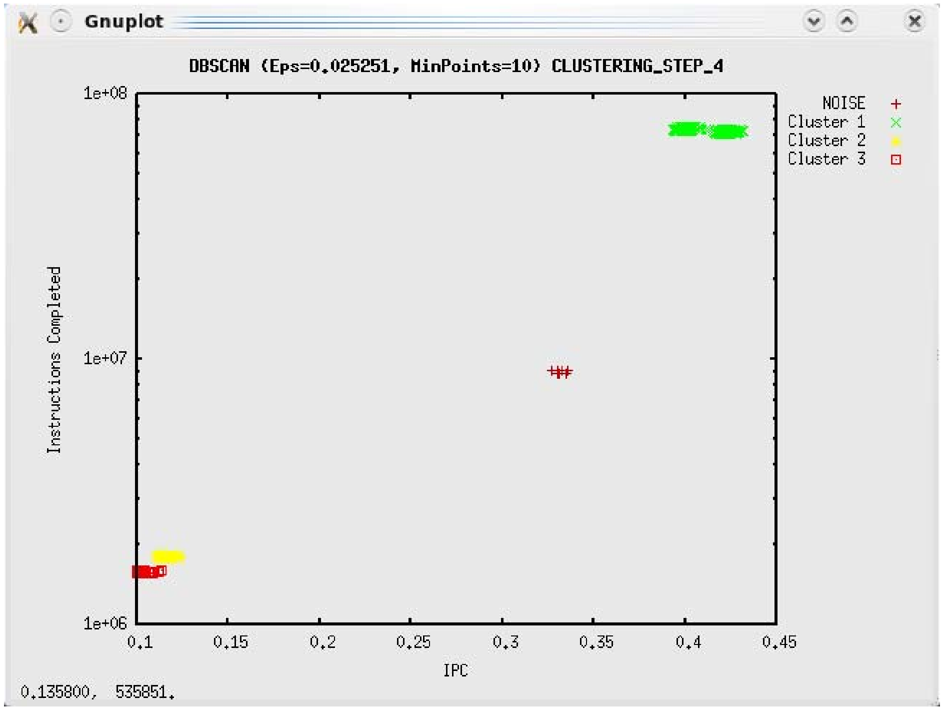


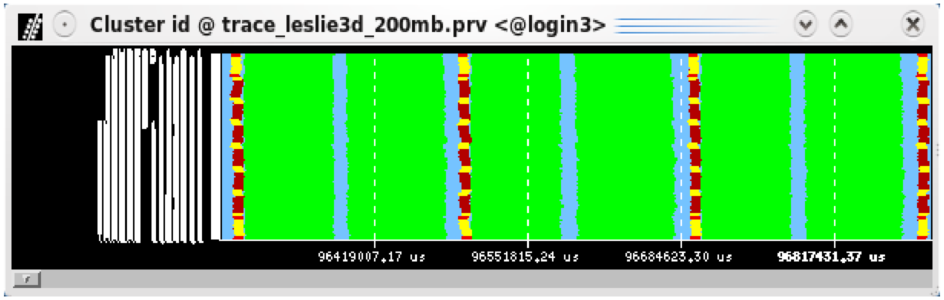


128 tasks, 200 Mb

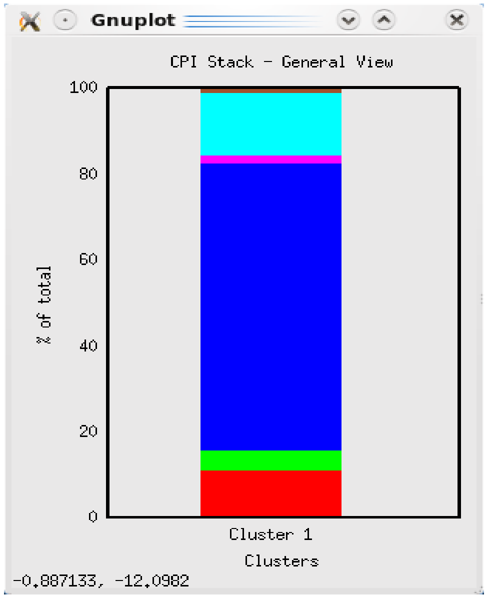


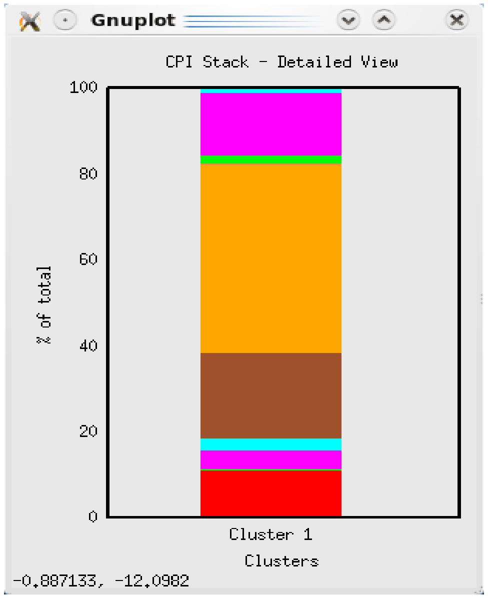
SPECMPI’07 – LESLIE3D (256 tasks)



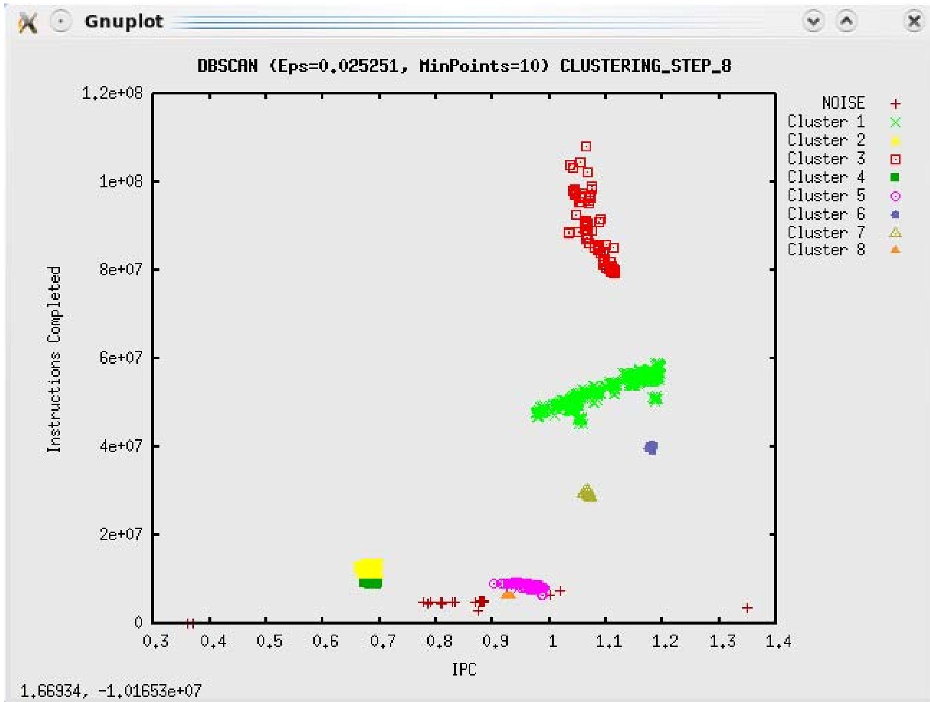


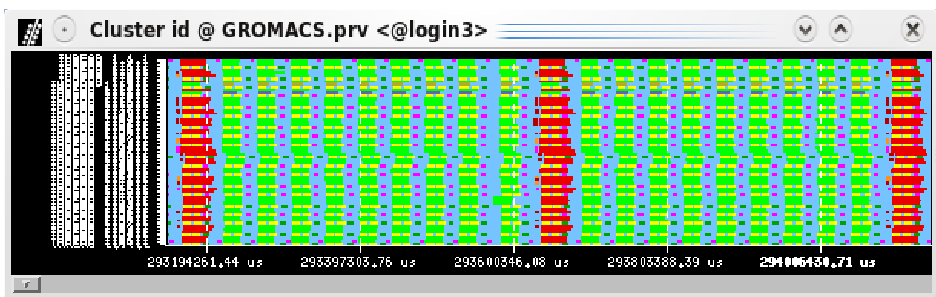
256 tasks, 300 Mb



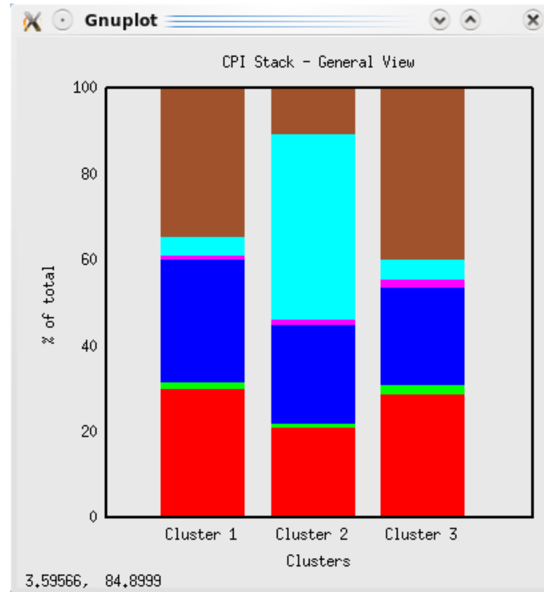


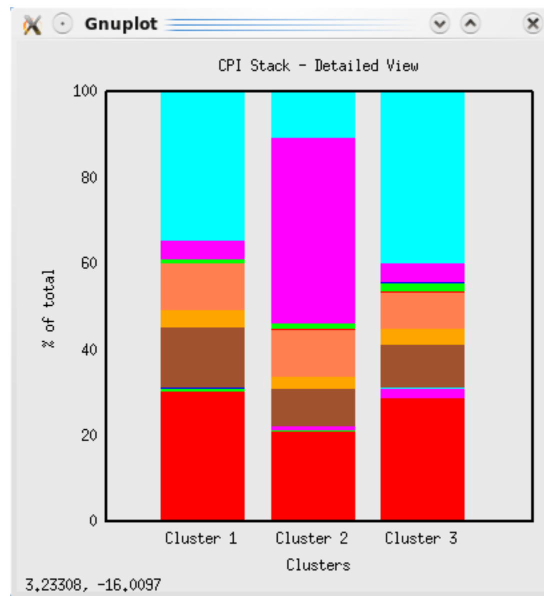
GROMACS (64 tasks)





64 tasks, 150 Mb





Conclusion

- We can get out much mode information than we currently do
- Can be done on-line
- Detailed analysis and visualization to find out how