Understanding applications using the BSC performance tools

Judit Gimenez (judit@bsc.es), German Llort

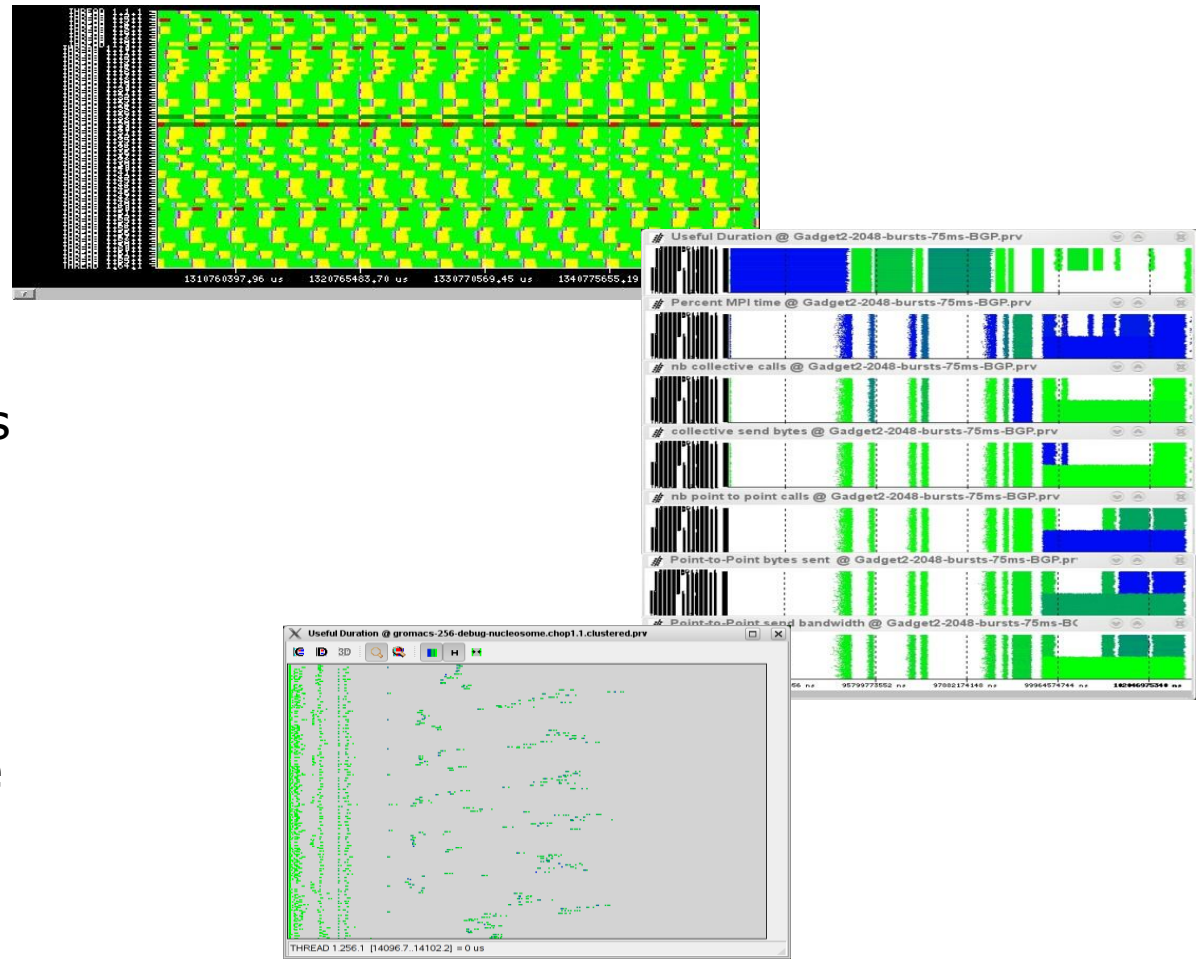
Barcelona Supercomputing Center

Humans are visual creatures

- Films or books? PROCESS
  - Two hours vs. days (months)
- Memorizing a deck of playing cards STORE
  - Each card translated to an image (person, action, location)
- Our brain loves pattern recognition IDENTIFY
  - What do you see on the pictures?

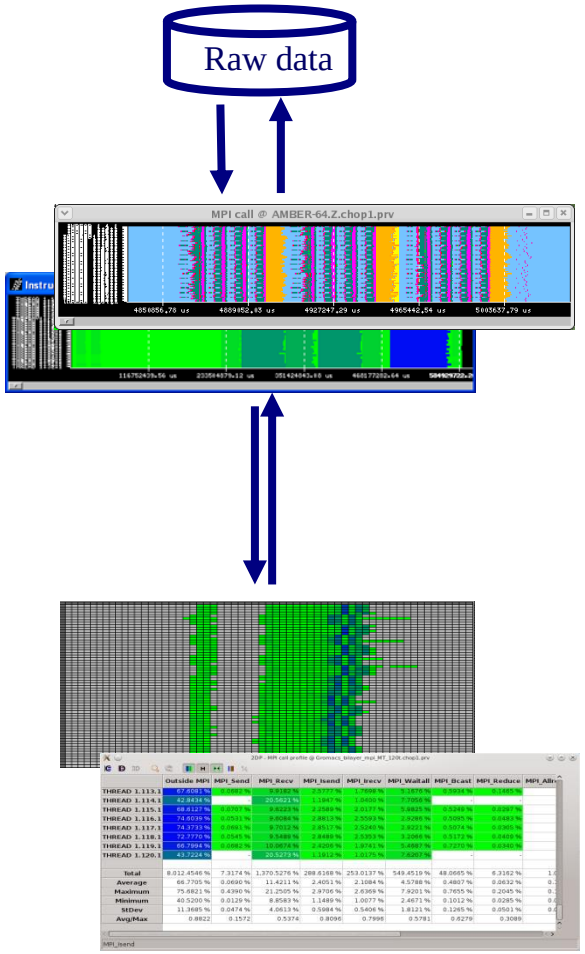
Our Tools

- Since 1991
- Based on traces
- Open Source
  - http://www.bsc.es/paraver
- Core tools:
  - Paraver (paramedir) – offline trace analysis
  - Dimemas – message passing simulator
  - Extrae – instrumentation
- Focus
  - Detail, variability, flexibility
  - Behavioral structure vs. syntactic structure
  - Intelligence: Performance Analytics



Paraver

Paraver: Performance data browser



Timelines

2/3D tables (Statistics)

Trace visualization/analysis

+ trace manipulation

Goal = Flexibility

No semantics

Programmable

Comparative analyses

Multiple traces

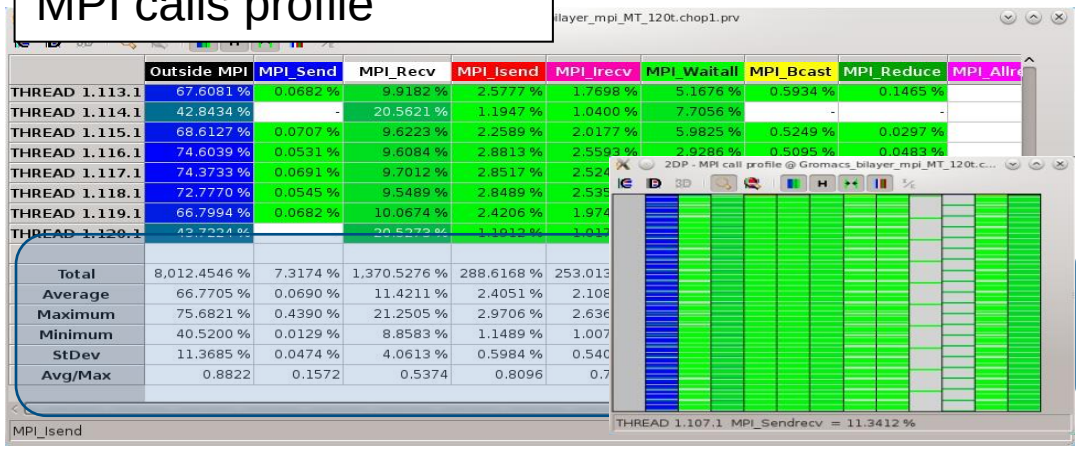
Synchronize scales

Tables: Profiles, histograms, correlations

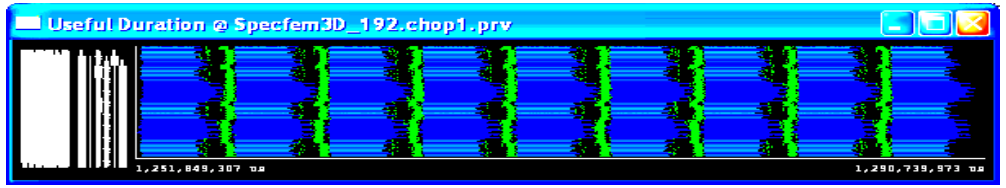
- From timelines to tables

MPI calls

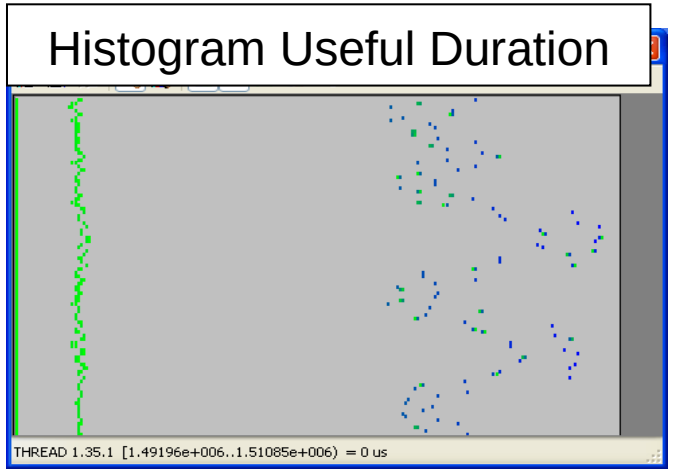
MPI calls profile



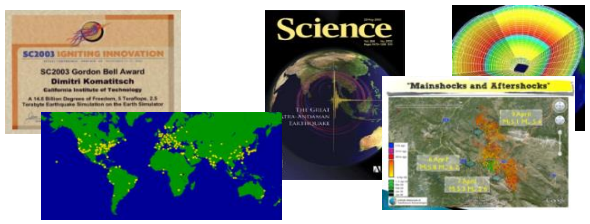
Useful Duration



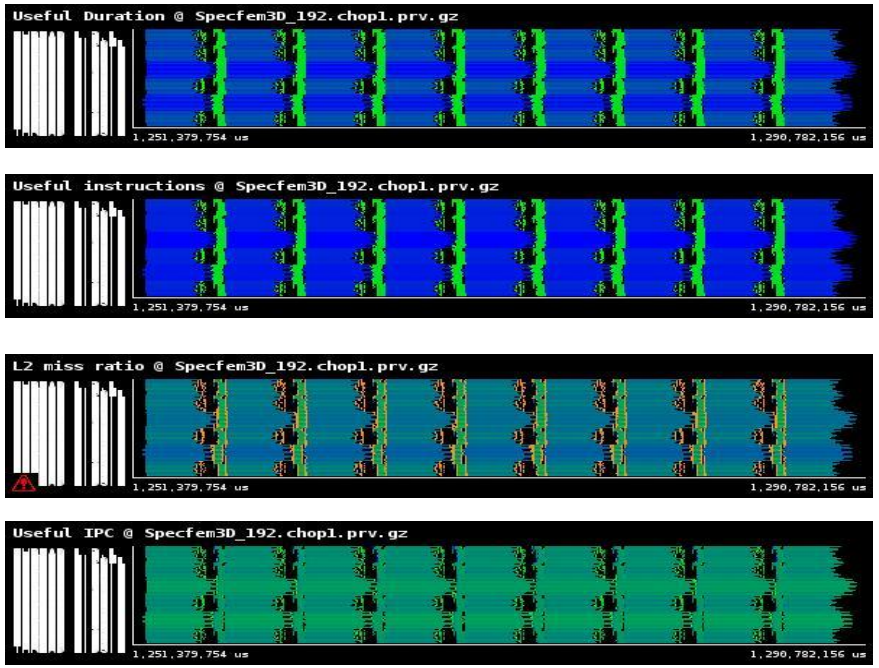
Histogram Useful Duration

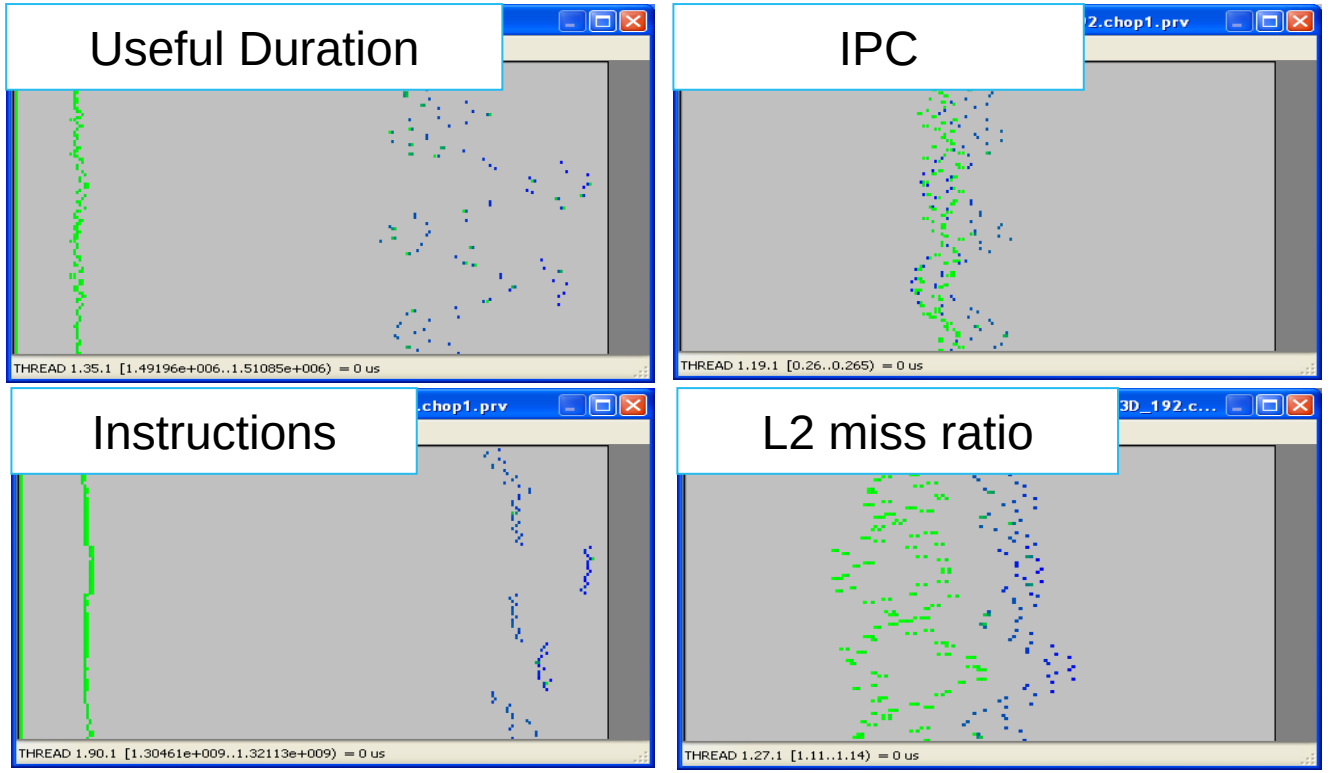


Analyzing variability through histograms and timelines



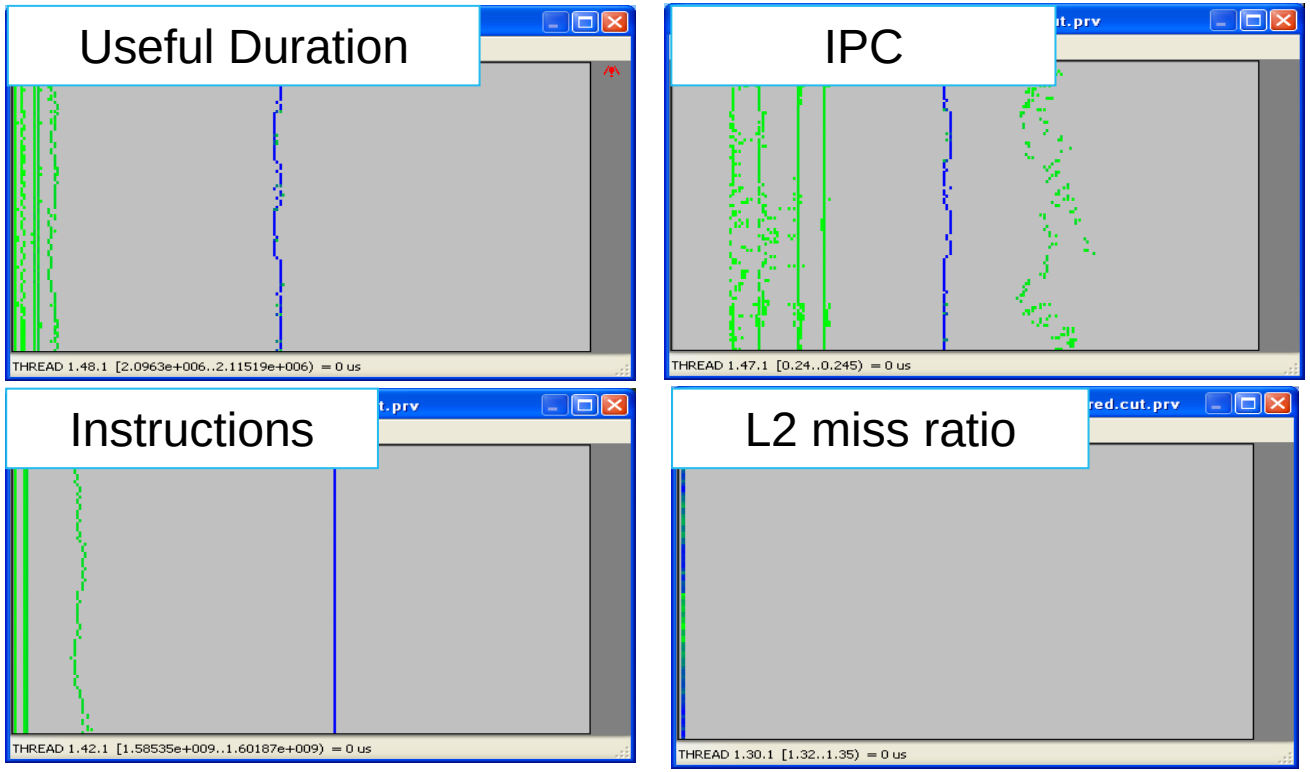
SPECFEM3D





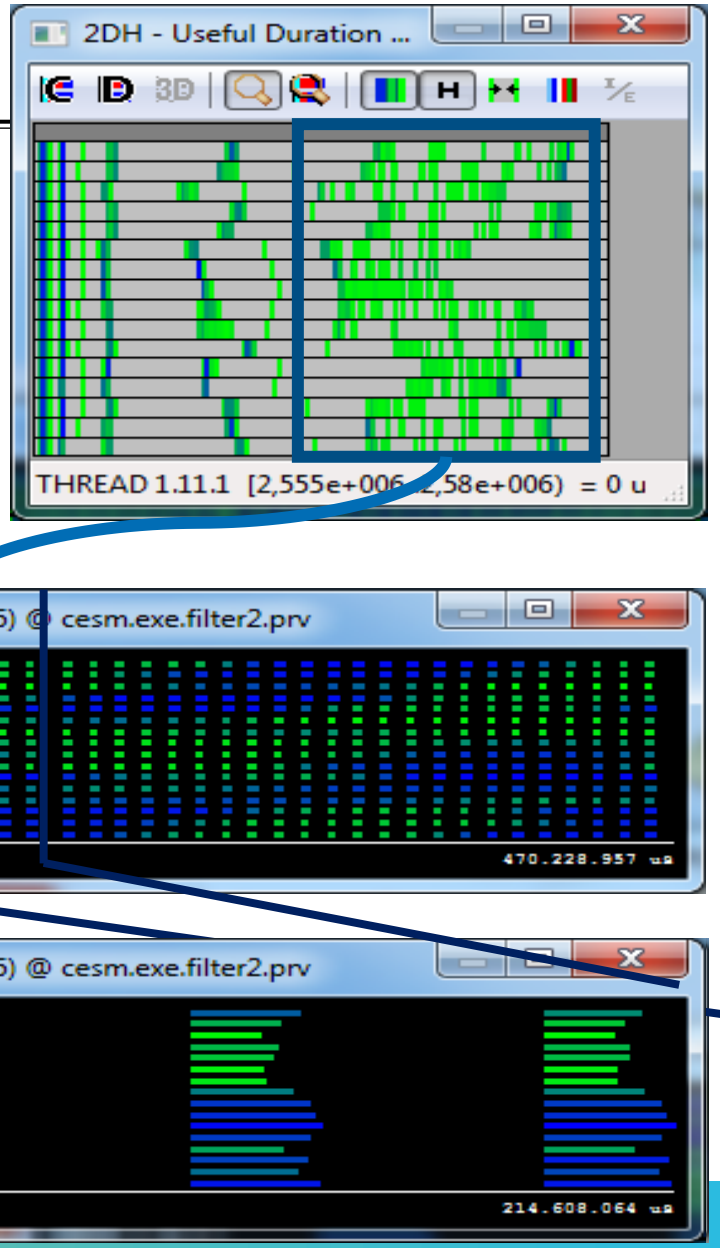
Analyzing variability through histograms and timelines

- By the way: six months later ….



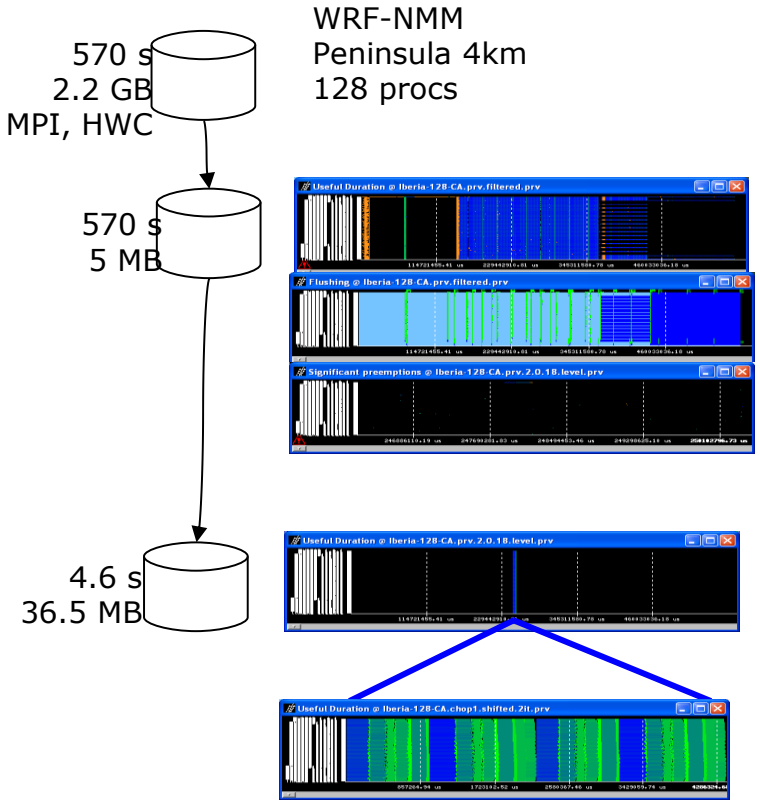
Variability … is everywhere

- CESM: 16 processes, 2 simulated days
- Histogram useful computation duration shows high variability
- How is it distributed?
- Dynamic imbalance
  - In space and time
  - Day and night.
  - Season ? ☺



Trace manipulation

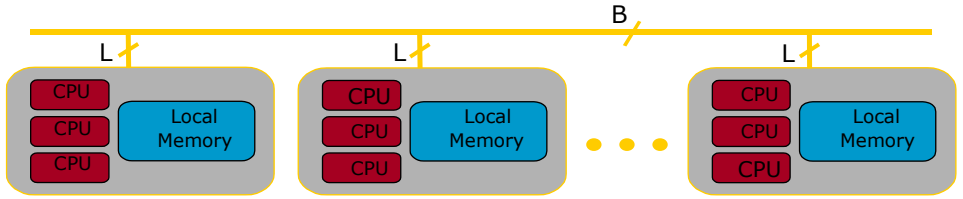
- Data handling/summarization capability
- Filtering
  - Subset of records in original trace
  - By duration, type, value,…
  - Filtered trace IS a paraver trace and can be analysed with the same cfgs (as long as needed data kept)
- Cutting
  - All records in a given time interval
  - Only some processes
- Software counters
  - Summarized values computed from those in the original trace emitted as new even types
  - #MPI calls, total hardware count,…

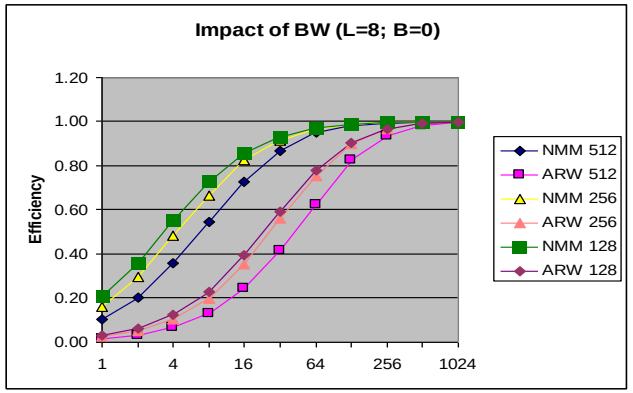


Dimemas

Dimemas: Coarse grain, Trace driven simulation

- Simulation: Highly non linear model
  - MPI protocols, resources contention…
- Parametric sweeps
  - On abstract architectures
  - On application computational regions
- What if analysis
  - Ideal machine (instantaneous network)
  - Estimating impact of ports to MPI+OpenMP/CUDA/…
  - Should I use asynchronous communications?
  - Are all parts of an app. equally sensitive to network?
- MPI sanity check
  - Modeling nominal
- Paraver – Dimemas tandem
  - Analysis and prediction
  - What-if from selected time window

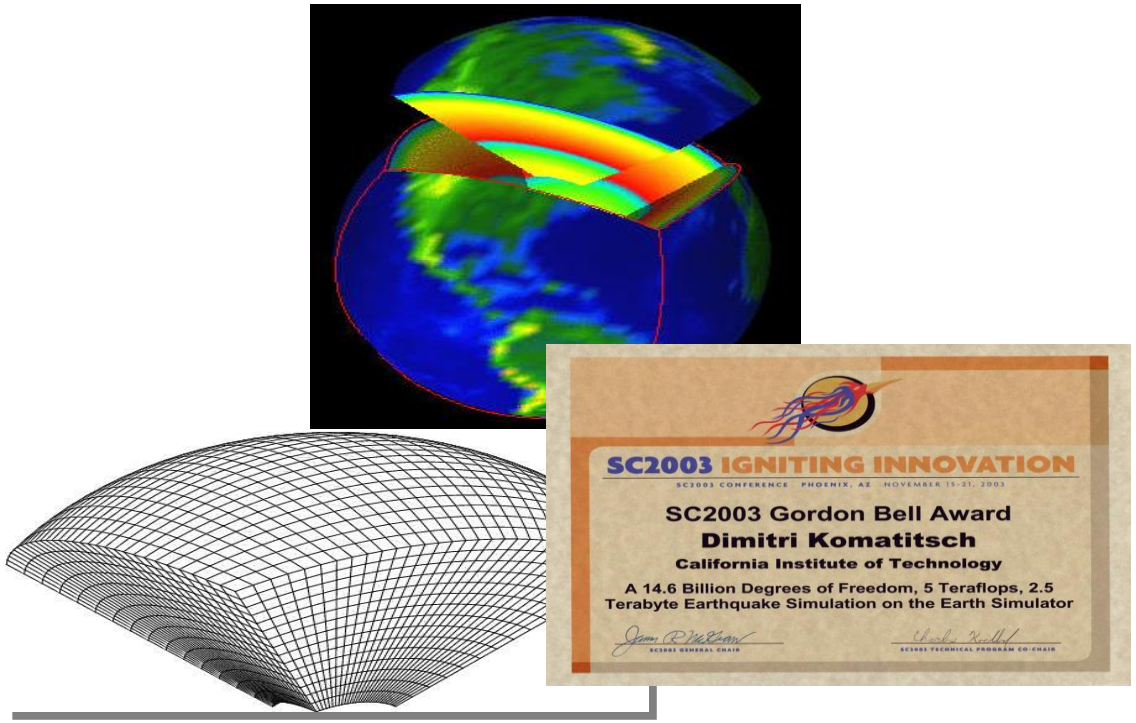




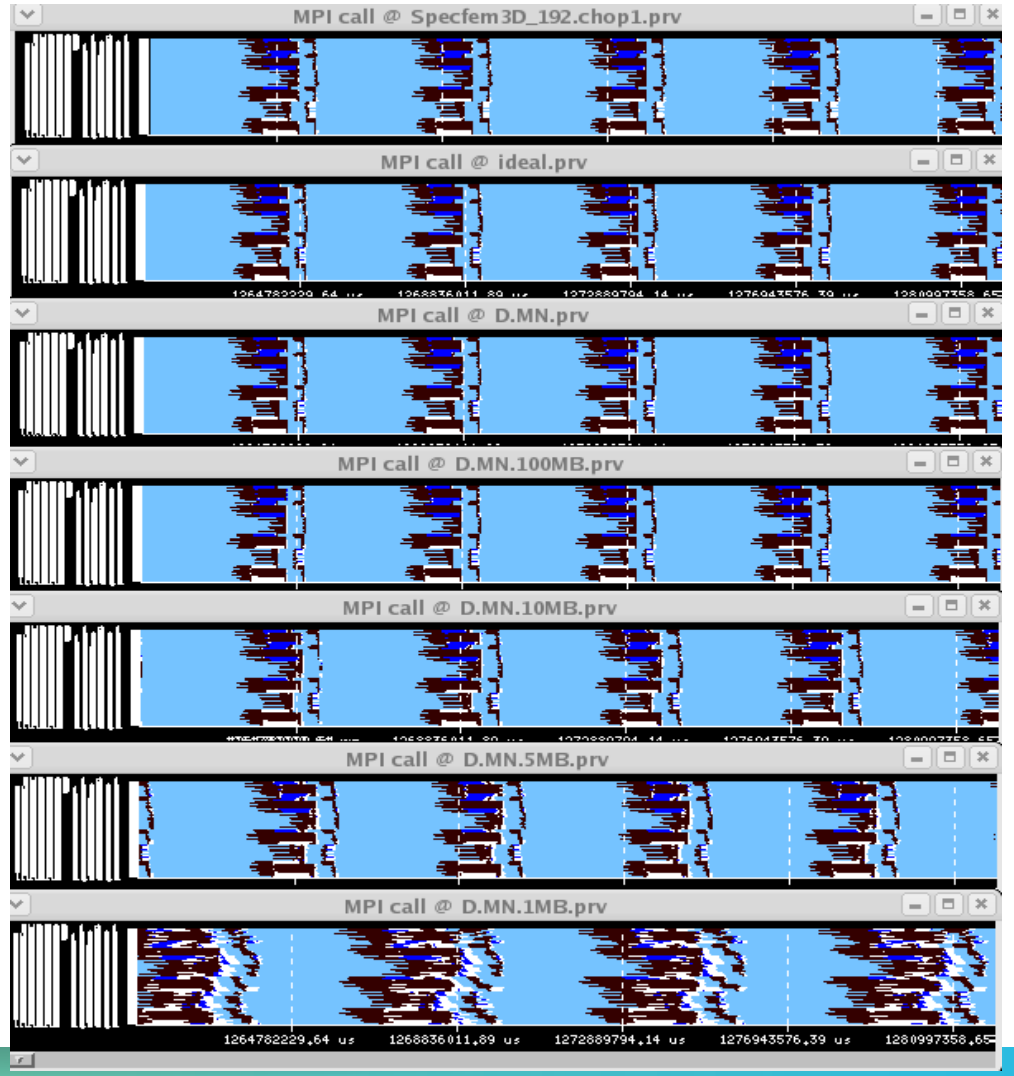
Detailed feedback on simulation (trace)

Would I will benefit from asynchronous communications?

- SPECFEM3D



Courtesy Dimitri Komatitsch



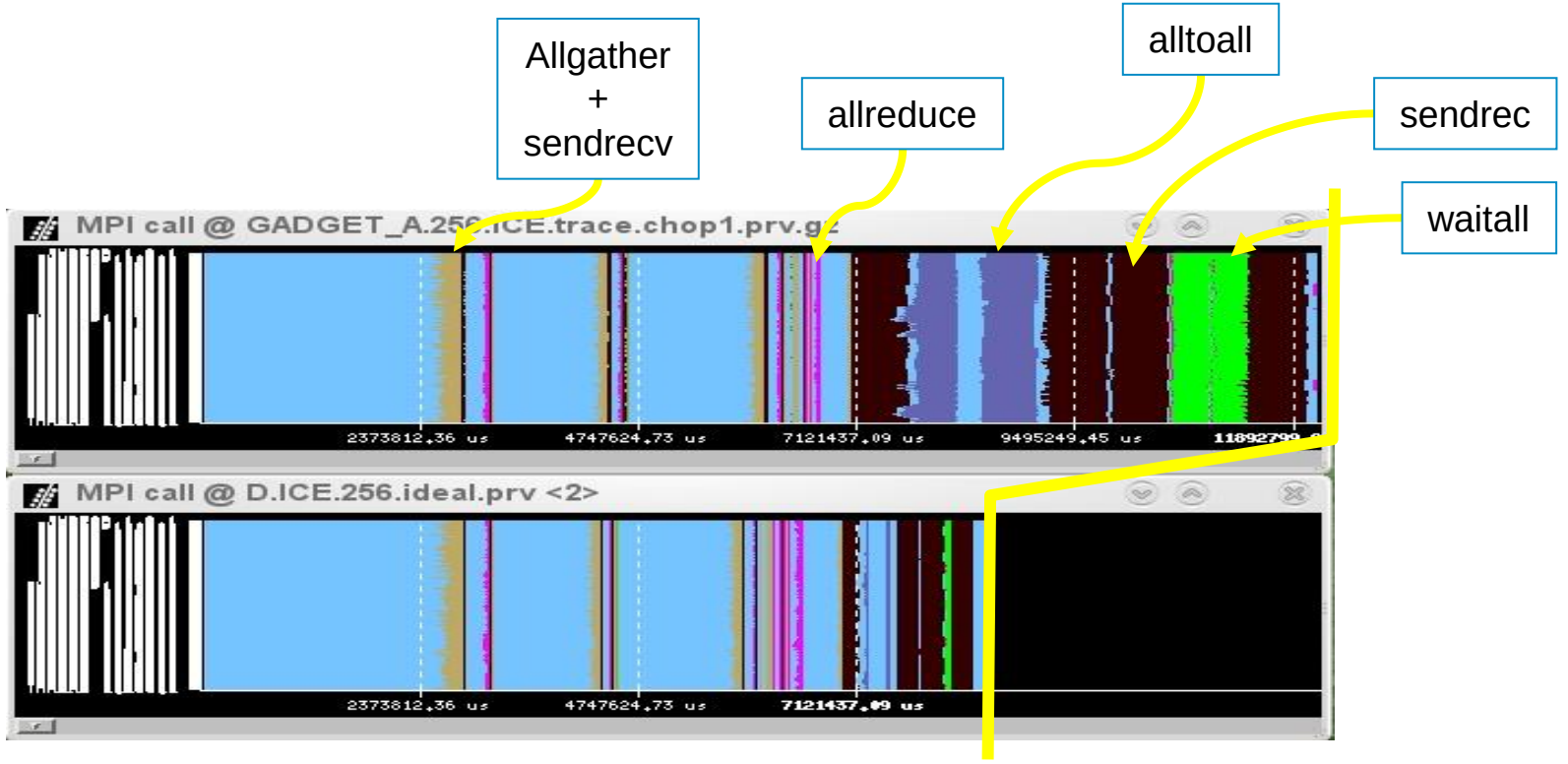
Ideal machine

- The impossible machine: BW = ∞, L = 0
  - Actually describes/characterizes Intrinsic application behavior
    - Load balance problems?
    - Dependence problems?

GADGET @ Nehalem cluster
256 processes

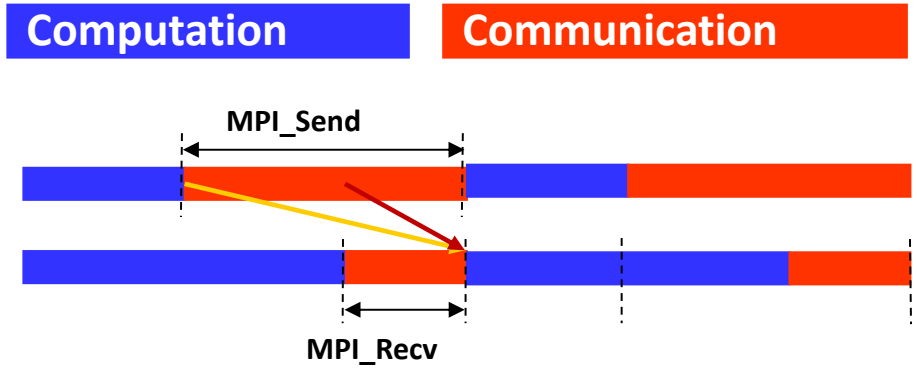
Real run

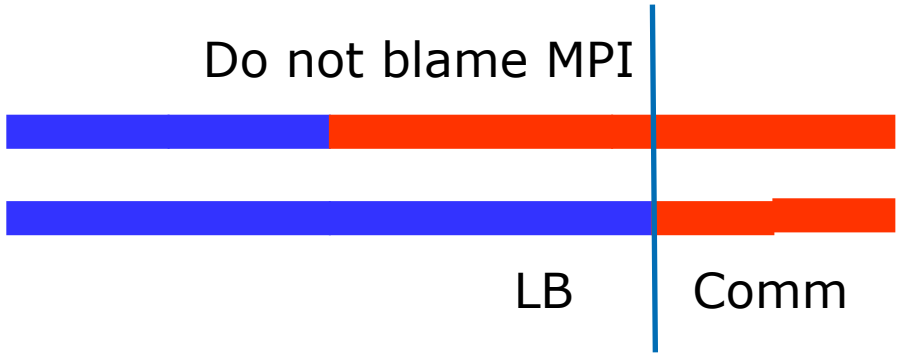
Ideal network



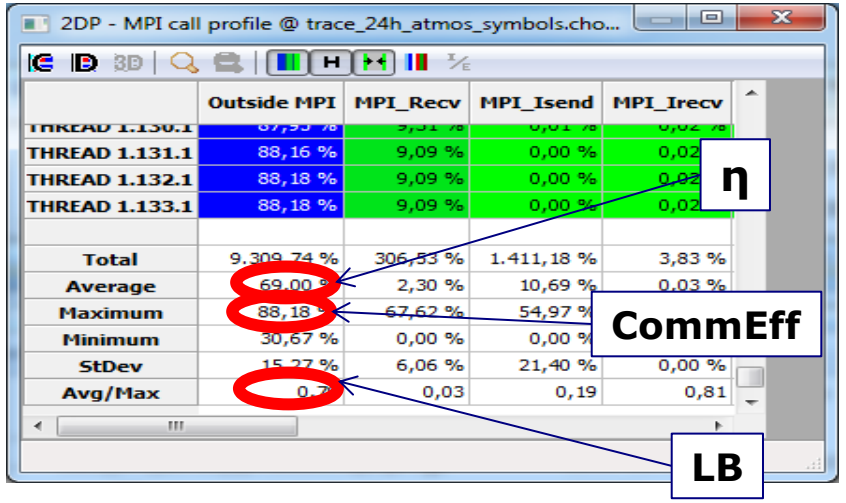
Models

Parallel efficiency model

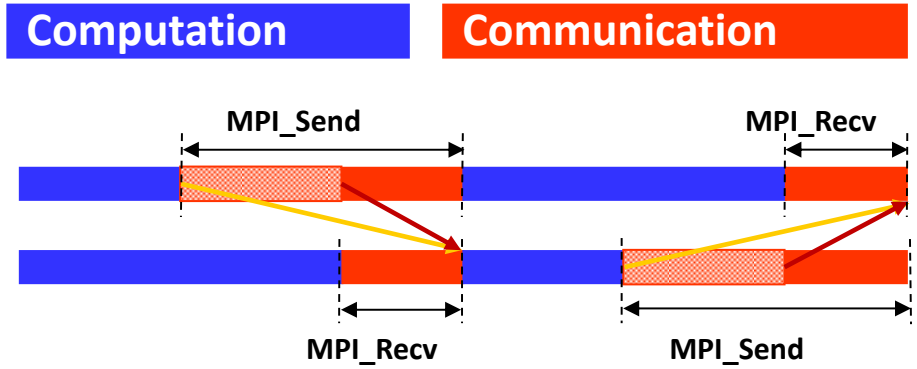


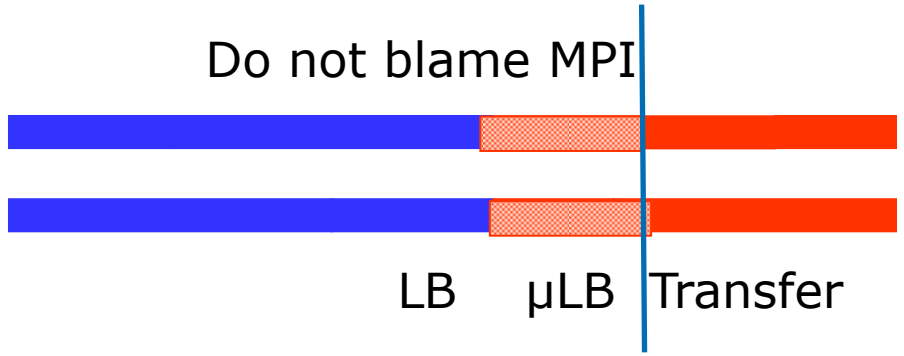


- Parallel efficiency = LB eff * Comm eff



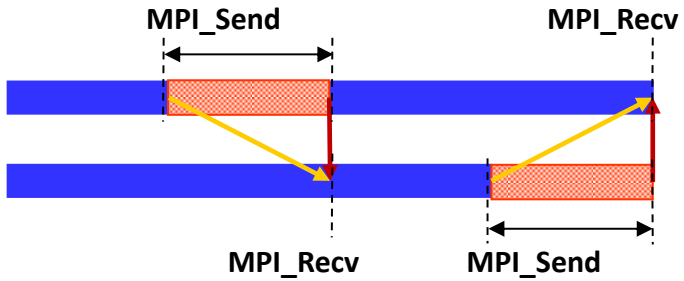
Parallel efficiency refinement: LB * µLB * Transfer





$$LB=1$$

- Serializations / dependences (µLB)
- Dimemas ideal network → Transfer (efficiency) = 1

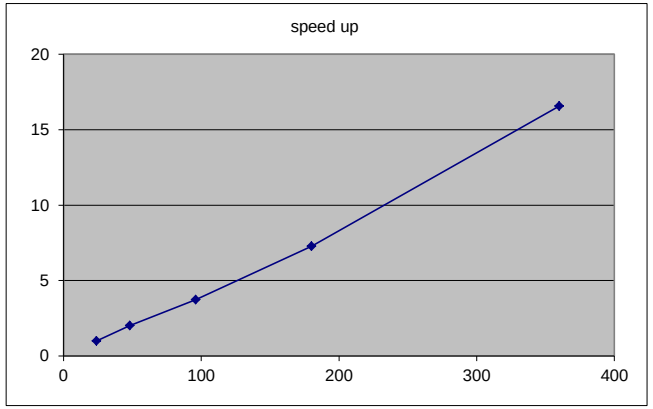


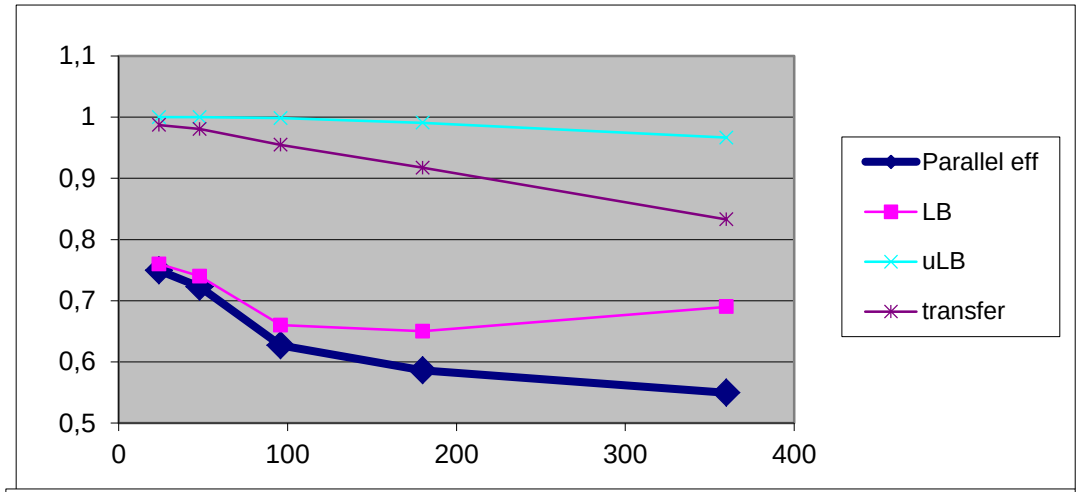
Why scaling?

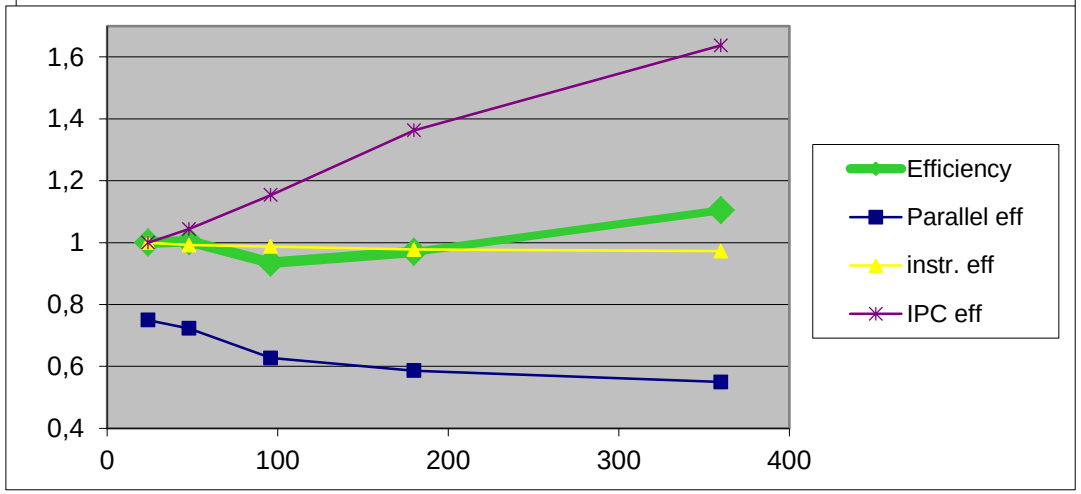
$$\eta_{\parallel} = LB * Ser * Trf$$

CG-POP mpi2s1D - 180x120

Good scalability !!
Should we be happy?

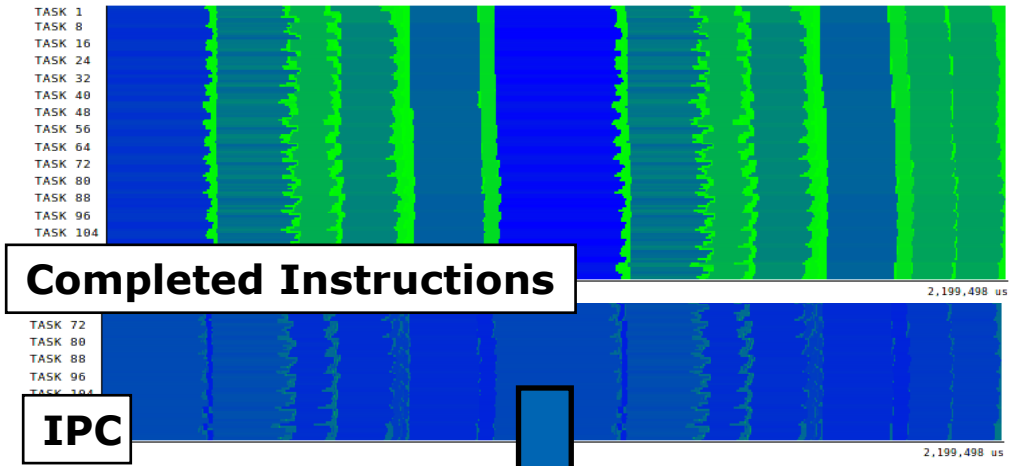


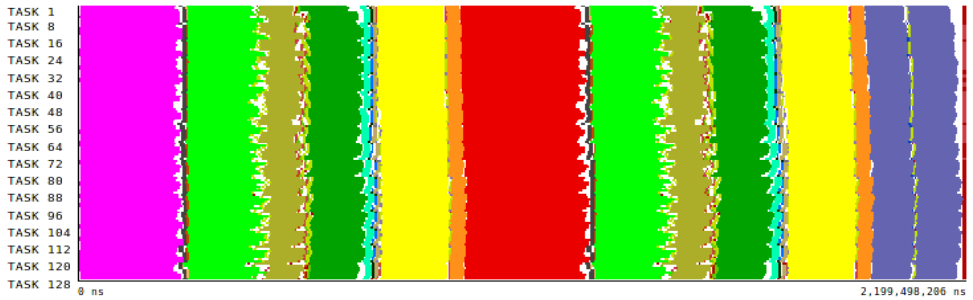


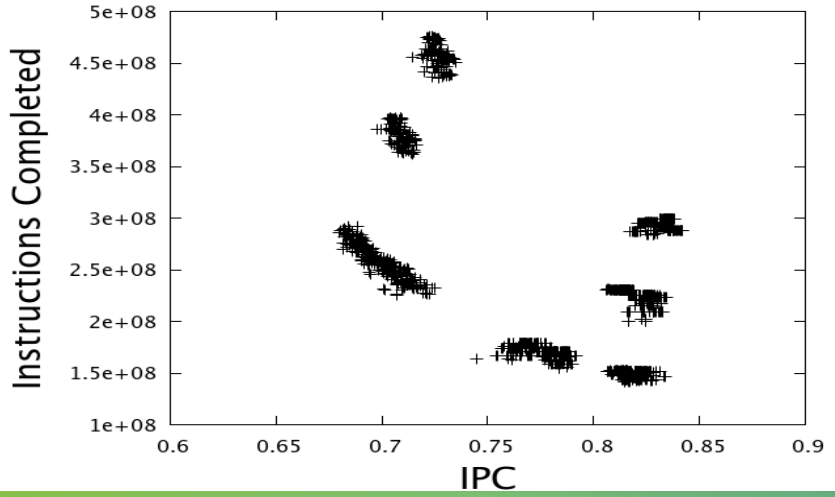


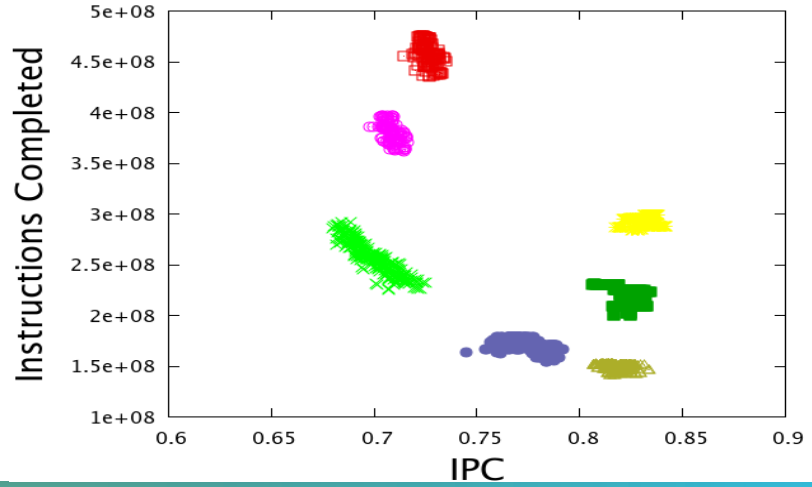
Peformance Analytics

Using Clustering to identify structure



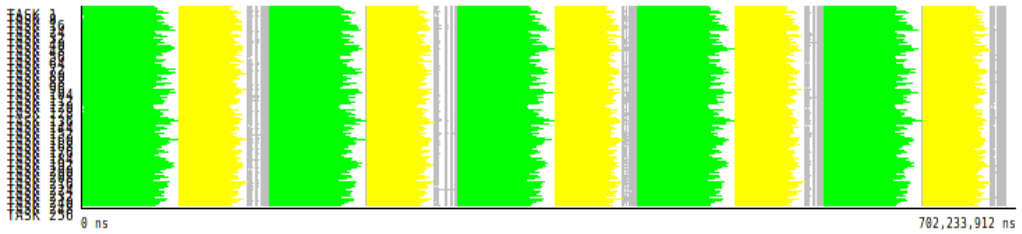






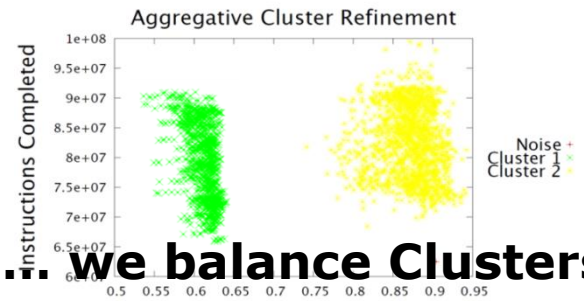
Integrating models and analytics

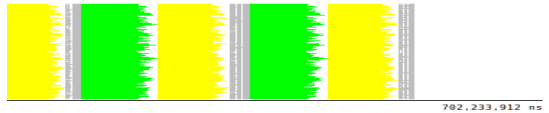
What if ….



PEPC

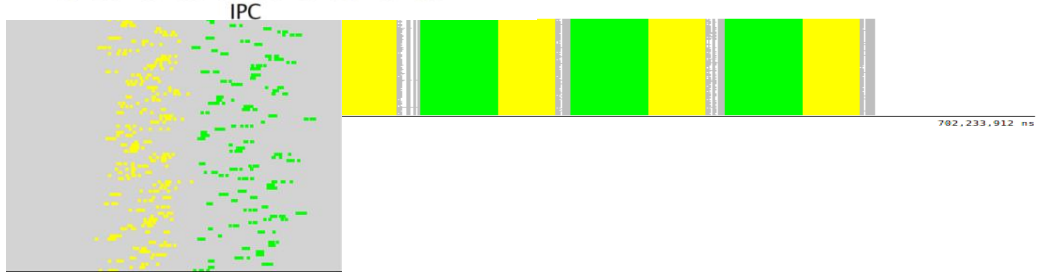
… we increase the IPC of Cluster1?





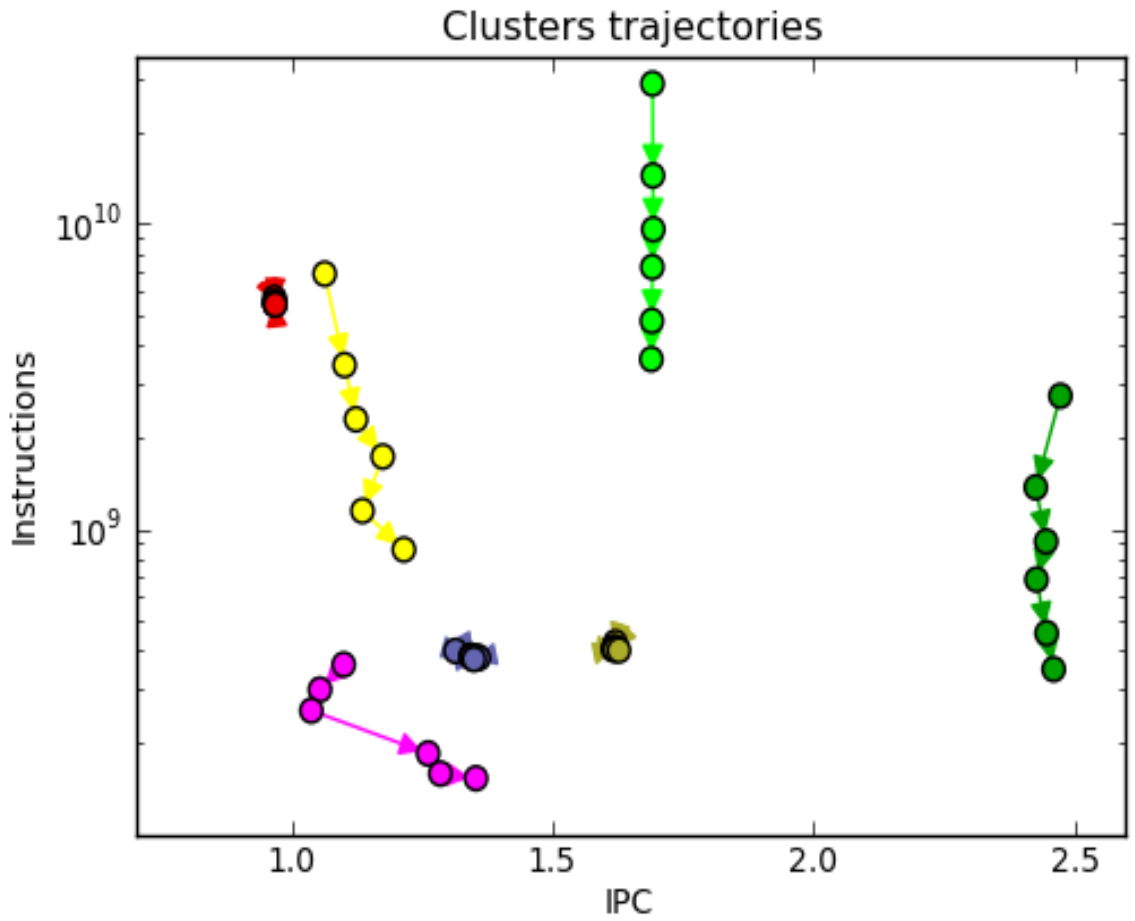
13% gain

… we balance Clusters 1 & 2?



19% gain

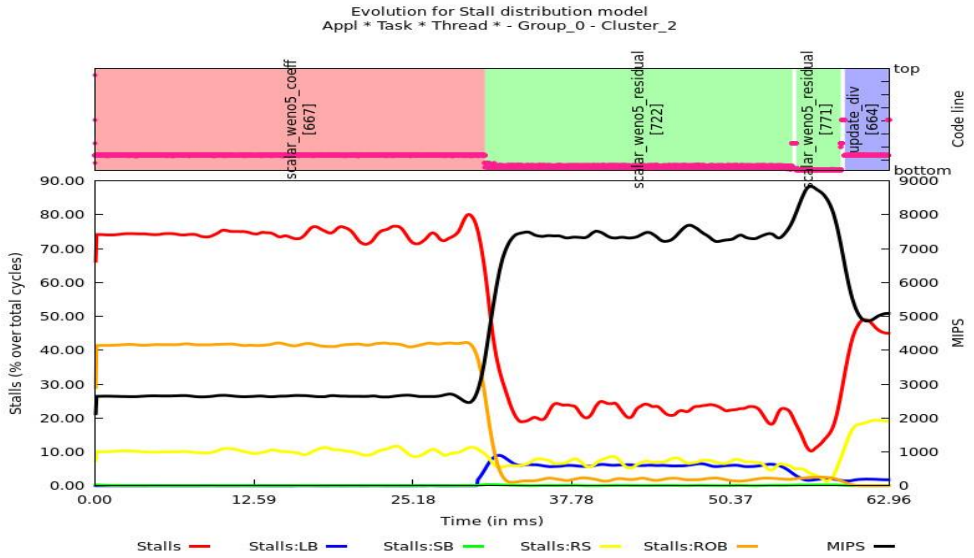
Tracking: scability through clustering



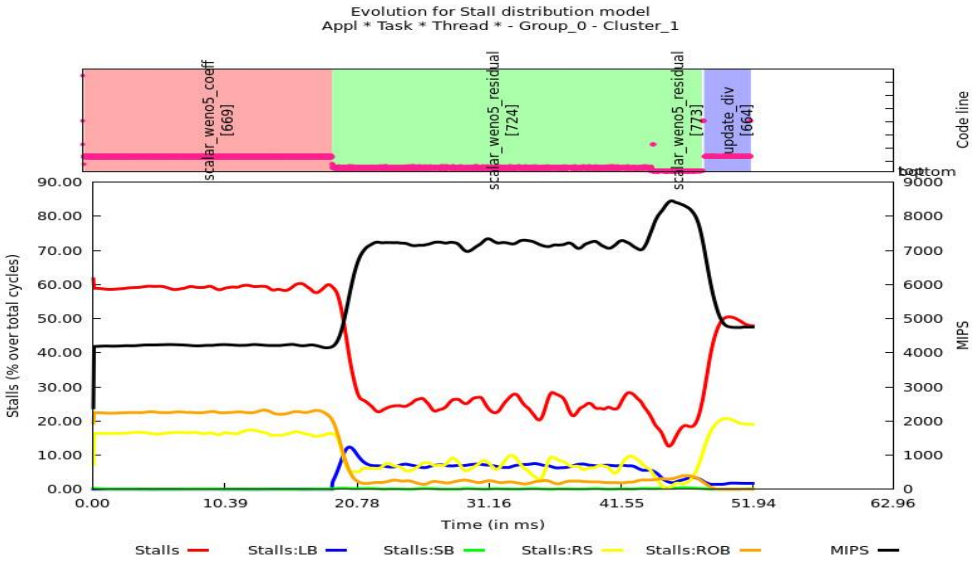
OpenMX
strong scale from 64 to 512 tasks

"Blind" optimization

- From folded samples of a few levels to timeline structure of "relevant" routines



Recommendation without access to source code



Methodology

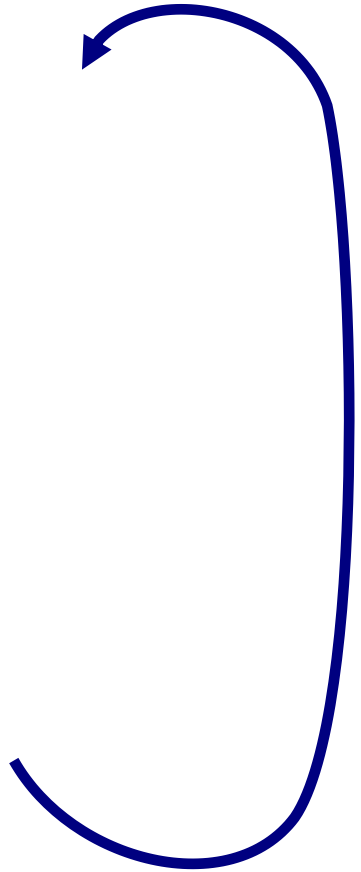
Performance analysis tools objective

Help generate hypotheses

Help validate hypotheses

Qualitatively

Quantitatively



First steps

- Parallel efficiency – percentage of time invested on computation
  - Identify sources for "inefficiency":
    - load balance
    - Communication /synchronization
- Serial efficiency – how far from peak performance?
  - IPC, correlate with other counters
- Scalability – code replication?
  - Total #instructions
- Behavioral structure? Variability?

BSC Tools web site

- www.bsc.es/paraver
- downloads
  - Sources / Binaries
  - Linux / windows / MAC
- documentation
  - Training guides
  - Tutorial slides

- Getting started
  - Start wxparaver
  - Help → tutorials and follow instructions
  - Follow training guides
    - Paraver introduction (MPI): Navigation and basic understanding of Paraver operation

Paraver Demo

Some examples of efficiencies

Code	Parallel efficiency	Communication efficiency	Load Balance efficiency
Gromacs@mt	66.77	75.68	88.22
BigDFT@altamira	59.64	78.97	75.52
CG-POP@mt	80.98	98.92	81.86
ntchem_mini@pi	92.56	94.94	97.49
nicam@pi	87.10	75.97	89.22
cp2k@jureca	75.34	81.07	92.93
icon@mistral	79.86	84.02	95.05
k-Wave@salomon	89.08	92.84	95.96
fleur@claix	76.22	90.66	84.07

Same code, different behaviour

Code	Parallel efficiency	Communication efficiency	Load Balance efficiency
lulesh@mn3	90.55	**99.22**	91.26
lulesh@leftraru	**69.15**	99.12	**69.76**
lulesh@uv2 (mpt)	70.55	96.56	73.06
lulesh@uv2 (impi)	85.65	95.09	90.07
lulesh@mt	83.68	95.48	87.64
lulesh@cori	**90.92**	98.59	**92.20**
lulesh@thunderX	73.96	97.56	75.81
lulesh@jetson	75.48	**88.84**	84.06
lulesh@claix	77.28	92.33	83.70
lulesh@jureca	88.20	98.45	89.57
lulesh@mn4	86.59	98.77	87.67
lulesh@inti	88.16	98.65	89.36

Warning::: Higher parallel efficiency does not mean faster!

Performance Optimization and Productivity

EU H2020 Center of Excellence (CoE)

1 October 2015 – 31 March 2018 (30 months)

Partners

- **Who?**
  - BSC (coordinator), ES
  - HLRS, DE
  - JSC, DE
  - NAG, UK
  - RWTH Aachen, IT Center, DE
  - TERATEC, FR

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

HLRS

JÜLICH FORSCHUNGSZENTRUM

nag®

RWTH AACHEN UNIVERSITY

Ter@tec

A team with

- Excellence in performance tools and tuning
- Excellence in programming models and practices
- Research and development background AND
  proven commitment in application to real academic and industrial use cases

3 levels of services

? Application Performance Audit

- Primary service
- Identify performance issues of customer code (at customer site)
- Small Effort (< 1 month)

! Application Performance Plan

- Follow-up on the service
- Identifies the root causes of the issues found and qualifies and quantifies approaches to address the issues
- Longer effort (1-3 months)

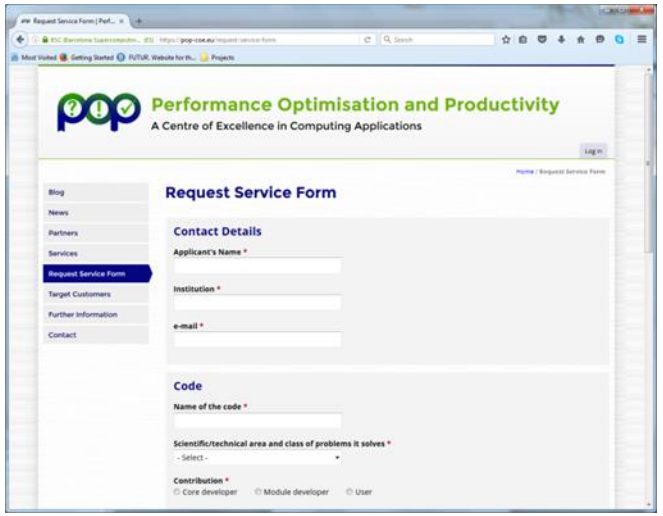
✓ Proof-of-Concept

- Experiments and mock-up tests for customer codes
- Kernel extraction, parallelization, mini-apps experiments to show effect of proposed optimizations
- 6 months effort

Reports

Software demonstrator

Apply @
http://www.pop-coe.eu



Codes analyzed

- DPM
- Quantum Espresso
- DROPS
- Ateles
- SHP-Fluids
- GraGLeS2D
- NEMO
- VAMPIRE
- psOpen
- GYSELA
- AIMS
- OpenNN
- FDS
- Baleen
- Mdynamix
- ParFlow
- GITM
- BPMF
- FIRST
- SHEMAT
- GS2
- ADF
- DFTB
- ICON
- dwarf2-ellipticsolver
- EPW
- Code Saturne
- ONETEP
- Ms2
- SIESTA
- Oasys GSA
- SOWFA
- BAND
- NGA
- Fidimag
- LAMMPS
- ScalFMM
- CHAPSIM K.W.
- ArgoDSM
- CIAO
- FFEA
- k-Wave
- DSHplus
- RICH
- COOLFluiD
- Ondes3D
- ATK
- Molcas
- GBMol_DD
- Kratos
- cf-python

+ few under NDAs

Performance Optimisation and Productivity

A Centre of Excellence in Computing Applications

Contact:

https://www.pop-coe.eu

mailto:pop@bsc.es

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 676553.