Simulation Of Next Generation Systems

SONGS (11 INFR 13)

Mid-term Project Evaluation

Bordeaux (Cepage): L. Eyraud, O. Beaumont, N. Bonichon, F. Mathieu,
(Runtime): D. Barthou, B. Goglin, A. Guermouche

Grenoble (Mescal): A. Legrand, D. Kondo, J.-F. Méhaut, J.-M. Vincent

Nancy (Algorille): **M. Quinson**, L. Nussbaum

Nantes (Ascola): A. Lèbre

Nice (Mascotte): *O. Dalle*, H. Renard

Villeurbanne (CCIN2P3): F. Suter, P.-E. Brinette, *F. Desprez*

Strasbourg (ICPS): S. Genaud, J. Gossa

ANR

November 5, 2013. Paris.

Scientific Context

Modern Computer Systems

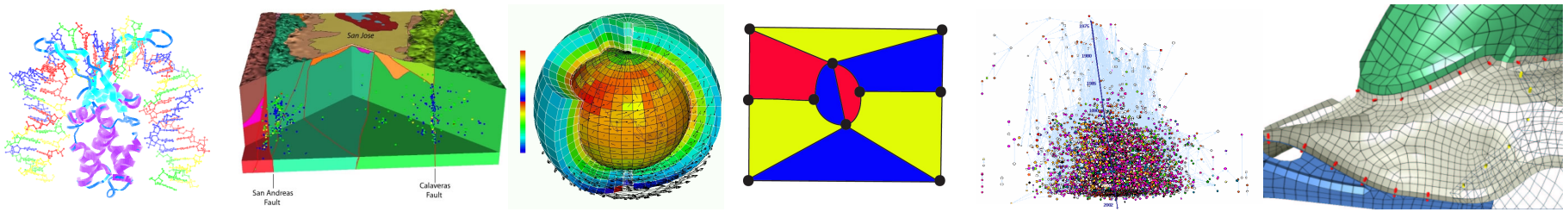
- Grids, P2P, Clouds, HPC, ...
- **Hierarchical**, complex and **heterogeneous**
- **Very large** and **dynamic** systems

Challenge: (**correctness** and) **performance** of these systems

- Reductionism is not satisfactory; experiences are mandatory
- We thus need **Scientific Instruments**, just as in physics or other

Idea: Computational Science *of* Computer Systems

- Computational Science use computers as scientific instruments
- It builds **models to understand** and conducts **simulations to predict**



- Can we reuse this approach to **understand modern computer systems**?

SimGrid and the ANR SONGS project

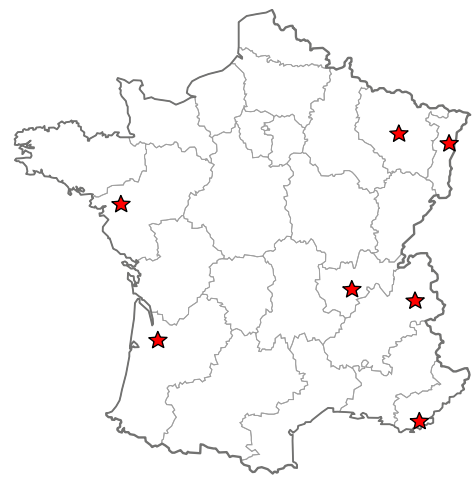
SimGrid: Simulator of distributed applications

- Infancy (1999): Factorizing the code of some students
- Now: Versatile, extensible, verified predictive power, free and open
- Impact (2008-2012): ≈60 publications, ≈100 authors, 3 PhD

SONGS: Simulation Of Next Generation Systems

ANR 11 INFR 13

- **Platform project** (1.8M€, 400 PM founded)
- 7 academic partners, 20+ researchers (420 PM)
- Modeling large-scale computer systems (+= 2 domains)
  - Task 1: [Data]Grid
  - Task 2: Peer-to-Peer and Volunteer Computing
  - Task 3: IaaS Clouds
  - Task 4: High Performance Computing
- Simulation methodology (more of our expertise)
  - Task 5: Simulation Kernel
  - Task 6: Concepts and Models
  - Task 7: Analysis and Visualization
  - Task 8: Experimental Methodology



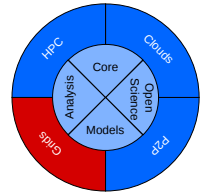
Use-Case Driven Research

- Science pulled by users' needs, not pushed by abilities
- *Scratch your own itches* (more motivating, and leads to better results)
- Longer term goal: Foster the emergence of a vivid research community

Work plan in each domain

- Tx.1: Add models needed by the planned studies
  - Grids: Storage modeling
  - P2P/VC: Scalable network modeling and Churn
  - IaaS Clouds: VMs, hypervisors
  - HPC: HPC networks, memory transfers
- Tx.2: Extend APIs to ease the planned studies
  - Grids: High Performance Storage System API
  - P2P/VC: Higher level API such as catalog handling
  - IaaS Clouds: Provider side, and client side
  - HPC: MPI, OpenMP, Plasma & Magma
- Tx.3: Do planned studies
  - Grids: Distributed Data mgmt for LHC; Hierarchical Storage System
  - P2P/VC: Replica Placement in VOD; Affinities in VC
  - IaaS Clouds: Study from client or provider POV; other metrics (energy)
  - HPC: Exascale; memory & energy models

Task 1: [Data]Grids

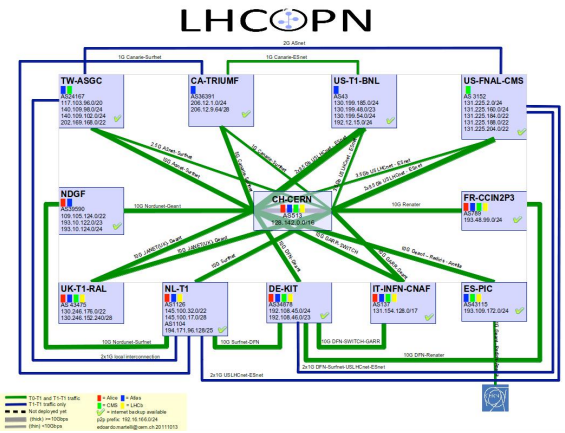


Initial Goals

- SimGrid was intended to study *Grid* Scheduling, but without Storage
- Model: storage; APIs: Integrated Storage Servers
- Study: Data mgmt for LHC, Hierarchical Storage Dimensionning (CC-IN2P3)

Achievements during the first half

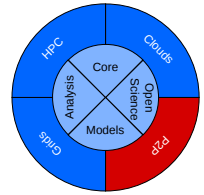
- An API capturing all relevant concepts is integrated (publication under review)
- Preliminary models proposed (by an associated PhD student at CERN)



Roadmap for the second half

- Validation and improvement of the storage models
- Conduct the planned studies
- Further push the collaboration with the CERN users

Task 2: Peer-to-Peer/Volunteer Computing

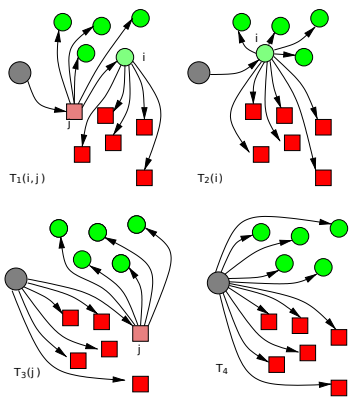


Initial Goals

- Ultra-Scalable Simulation: (achieved) goal of USS-SimGrid ANR project
- Model: Scalable Network Modeling; Churn. APIs: High-level (DHT or similar)
- Study P2P: Replica Placement in VOD; VC: CPU/Network Affinities

Achievements during the first half

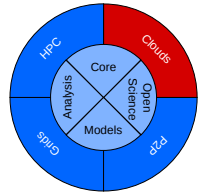
- Theoretical study of data broadcast in NATed environments
- Shown the usefulness of dynamic scheduling according to affinities
- Random platform generation; Ability to specify random churn
- Framework to evaluate network tomography algorithms in practice (PlanetLab)



Roadmap for the second half

- New scheduling strategies in BOINC (with U. Berkeley)
- Splay interface on top of SimGrid (with U. Neuchatel)
- Propose & run a large-scale network tomography

Task 3: IaaS Clouds

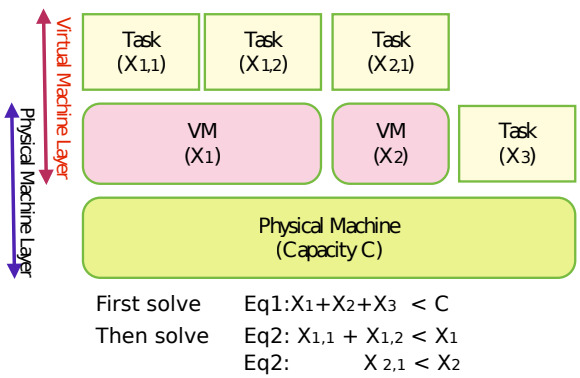


Initial Goals

- Virtualization: Adapted framework for provider- and/or client- side studies
- Model: Tasks over VMs, cloud dynamics; APIs: VMs mgmt, AWS (EC2, S3)
- Provider-side Study: VM orchestration and migration in private clouds
- Client-side Study: Decision helper (the best performance at the best price)

Achievements during the first half

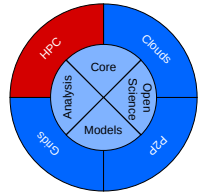
- A model for VM lifecycle management including live migration with precopy
- Provider: VM mgmt optim; Client: strategies optimizing cost/makespan
- All-in-one initial modeling of the AWS infrastructure from the client POV



Roadmap for the second half

- Merge back all efforts into the released SimGrid
- (in)validated models for VM interactions/migrations
- Keep up with EC2 billing modeling (self-invalidating)
- Complete client- and provider-sides studies

Task 4: High Performance Computing

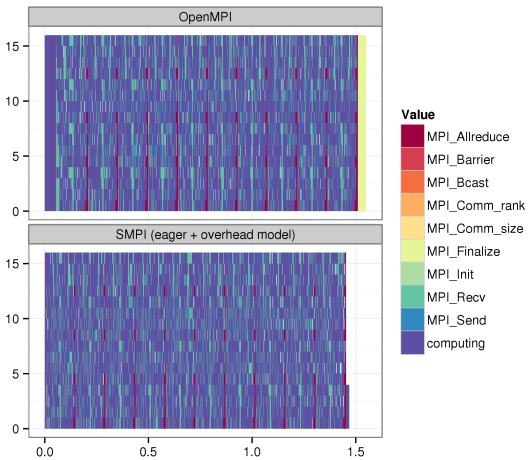


Initial Goals

- Challenging domain: high expectations on prediction accuracy, very large scale
- Model: HPC networks, Memory; APIs: MPI, OpenMP, Plasma & Magma
- Study: BigDFT, SPECFEM, MUMPS; Exascale ARM platform

Achievements during the first half

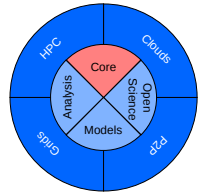
- More MPI coverage (OMPI&MPICH collectives); Online and Offline; Testing
- Hybrid model (fluid+LogOP): Good BigDFT speedup predictions on tibidabo
- Memory modeling is even more challenging than expected $\rightsquigarrow$ slowing OpenMP
- Preliminary modeling of StarPU runtime (in collaboration with tool's authors)



Roadmap for the second half

- Mixed online/offline for rapid testing of algorithms
- (in)Validation of StarPU models, more MPI apps
- Dimensionning tibidabo++ platform before building
- New models: IB networks, Memory, GPU, Xeon Phi

Task 5: Simulation Kernel

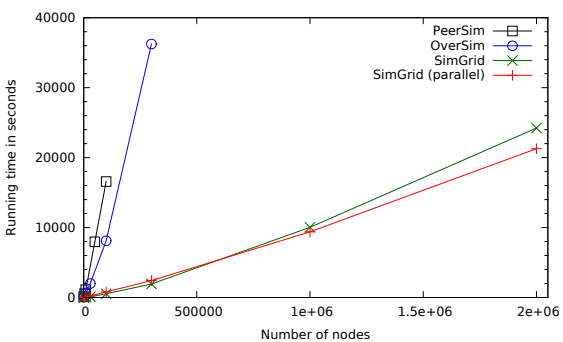


Initial Goals

- Efficient simulation: Big enough, Fast enough (going distributed and parallel)
- Standard simulation: Plug SimGrid with other simulation tools with DEVS

Achievements during the first half

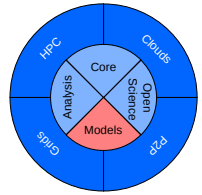
- Exploratory work on distributed simulation, but gross performance loss
- Parallel simulation now works, but disappointing performance
- Kernel fully rewritten in C++ to ease the inclusion of users' models
- Robust and Mature Framework, ready to use



Roadmap for the second half

- Better Parallel and Distributed simulation
- Further ease the users' models inclusion
- Eased interactions with NS3 and others

Task 6: Concepts and Models

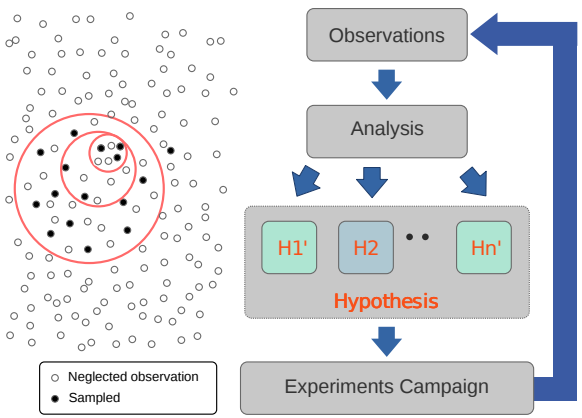


Initial Goals

	Grids	P2P	Clouds	HPC
Power Consumption	•		●	●
Storage Elements	●		•	•
CPU and Memory				●
High Performance Networks			•	●
Volatility & Dynamicity		●	•	•

Achievements during the first half

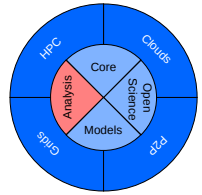
- Expertise hub for all modeling work and (in)validation studies
- Energy API and models (in collaboration with A.-C. Orgerie)
- Various models proposed; Network model thoroughly (in)validated



Roadmap for the second half

- (in)validation of energy models, and others
  This is a never-ending, resource-hungry task
- Very large-scale wide-area network tomography?

Task 7: Analysis and Visualization

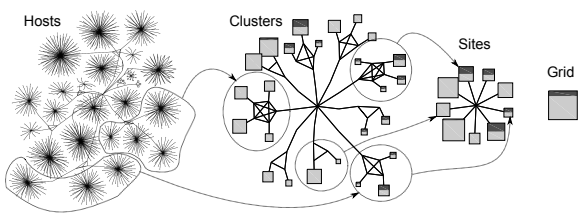


Initial Goals

- Scalable visualization: fast enough, tractable, and without artifacts
- Trace comparison: to better understand how/why a given alternative is better

Achievements during the first half

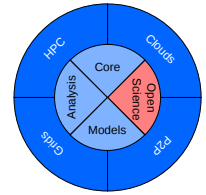
- Spatial and temporal aggregation to control artifacts (zooming effects)
- Entropy-based aggregation evaluation, helps selecting best aggregation
- Trace comparison used in (in)validations, but still very naive unfortunately



Roadmap for the second half

- Many new ideas on data aggregations to play with
- Trace comparison is slowly maturating
- Our main contributor got a permanent position
  $\rightsquigarrow$ temporary slowdown, but new collaboration

Task 8: Experimental Methodology

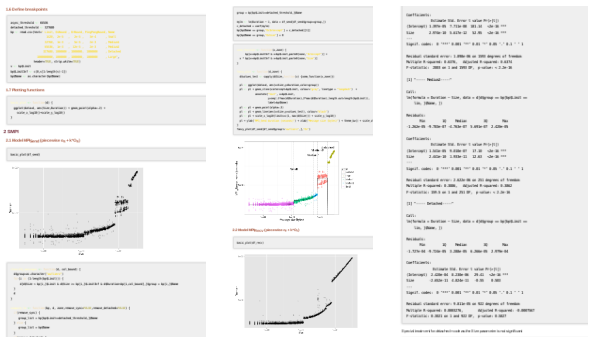


Initial Goals

- Design Of Experiments (DoE): choosing the right experiments to run
- Campaign Management: adapted batch schedulers for 10^6 jobs, DoE-aware
- Open Science: reproducible results, laying a basis for further works

Achievements during the first half

- DoE: Increased proficiency in statistical tools (R, ggplot, org-mode)
  Factorial experience design and ANOVA used in several publications
- Campaign Mgmt: more oriented toward Grid'5000, not yet running simulations
- Open Science: our publications are (more) reproducible



Roadmap for the second half

- Package simulation experiments for reproducibility
  Convenient runner as an incentive to the users
- Tool convergence with emulators and real platforms
- Foster a community on Open Science in our domain
  Still long way to go toward Reproducible Experiments

Work Distribution

Partner	WP1	WP2	WP3	WP4	WP5	WP6	WP7	WP8
Bordeaux	● ●	● ●	● ●	● ●		● ●		
Grenoble	●	● ●	● ●	● ●	● ●	● ●	● ●	● ●
Nancy	● ●	● ●	● ●	● ●	● ●	● ●	● ●	● ●
Nantes			● ●			● ●		
Nice		● ●			● ●	● ●		● ●
Strasbourg			● ●	● ●		● ●	● ●	
Villeurbanne	● ●		● ●	● ●		● ●		●

Planned Work Distribution	Actual Investment at T24
small ● ; ● large	small ● ; ● large

Work Distribution

Partner	WP1	WP2	WP3	WP4	WP5	WP6	WP7	WP8
Bordeaux	● ●	● ●	● ●	● ●		● ●		
Grenoble	●	● ●	● ●	● ●	● ●	● ●	● ●	● ●
Nancy	● ●	● ●	● ●	● ●	● ●	● ●	● ●	● ●
Nantes			● ●			● ●		
Nice		● ●			● ●	● ●		● ●
Strasbourg			● ●	● ●		● ●	● ●	
Villeurbanne	● ●		● ●	● ●		● ●		●

Planned Work Distribution	Actual Investment at T24
small ● ; ● large	small ● ; ● large

- WP1: DQ2 team (CERN) on modeling of data storage servers
- WP2: BOINC authors (U. Berkeley), Splay authors (U. Neuchatel)
- WP3: AIST Japan on VM migrations over WAN; UCMadrid on multi-clouds
- WP4: StarPU authors (Inria Bordeaux), BigDFT authors (CEA INAC)
- WP5: A. Giersch (U. Franche-Comté) on efficient simulation
- WP6: A.-C. Orgerie (CNRS Rennes) on energy modeling
- WP7: L. Schnorr (UFRGS, Brasil) on visualization
- WP8: M. Stillwell (U. Cranfield), S. Hunold (U. Vienna) on Open Science

Scientific Outcomes and Dissemination

Outcomes

- 20 mono-site publications + 4 multi-site publications (many more under review)
- 10 invited talk and keynotes
- 3 major releases of SimGrid (150+ commits/month, 6+ contributors each month)

Dissemination: SimGrid Users' Days

- 3 days conference gathering (potential) users to exchange news and feedback
- Takes place in remote location where there is nothing else to do
- 2010 and 2012 editions were rather classical (presentations)
- 2013 "working workshop": hack your own SimGrid project, under our guidance

Efficient Networking

- We are at SuperComputing every year (on Inria booth); COMPAS in France
- Joint lab with Urbana Champain on HPC; Barcelona Supercomputer Center
- Connections to IETF toward an Informal RFC on P2P simulation
- Ongoing discussions for several European projects, Inria IPL, etc.
- Even recent interactions with science philosophers from Finland!

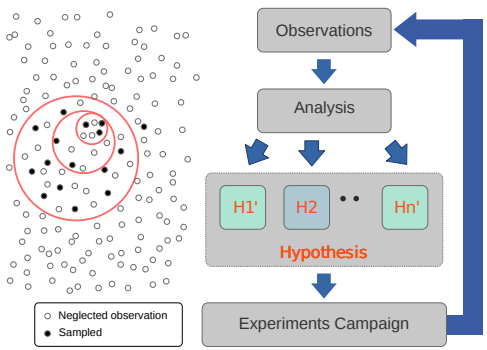
Scientific Zoom: How do we model things?

"Original" epistemological stance

- Things are so complex that reductionism does not work anymore

Computer Systems ≈ Natural Systems

- Empirical measurements, hypothesis, modeling, (in)validation (ad eternam)



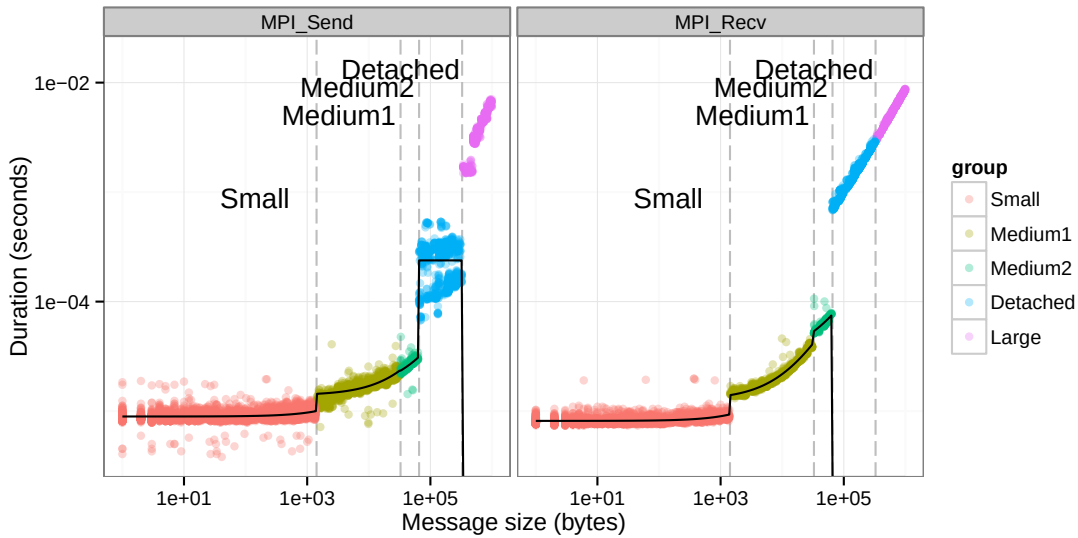
Wanted Models

- Explanatory and interpretable models: We model mainly to understand
- Quantitative: Computation or communication time
- Qualitative: Interactions between streams or between processes (or both)
- Semantic: Search for synchronization bugs

What do we find when doing so?

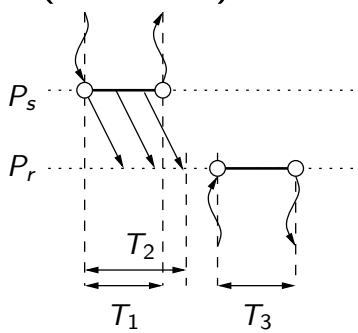
Such models are possible

Measurements MPI_Send / MPI_Recv

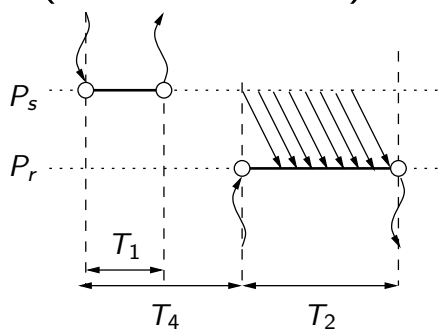


SMPI Model

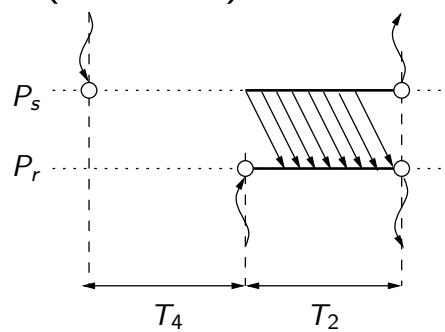
Asynchronous mode ($k \leq S_a$)



Detached mode ($S_a < k \leq S_d$)

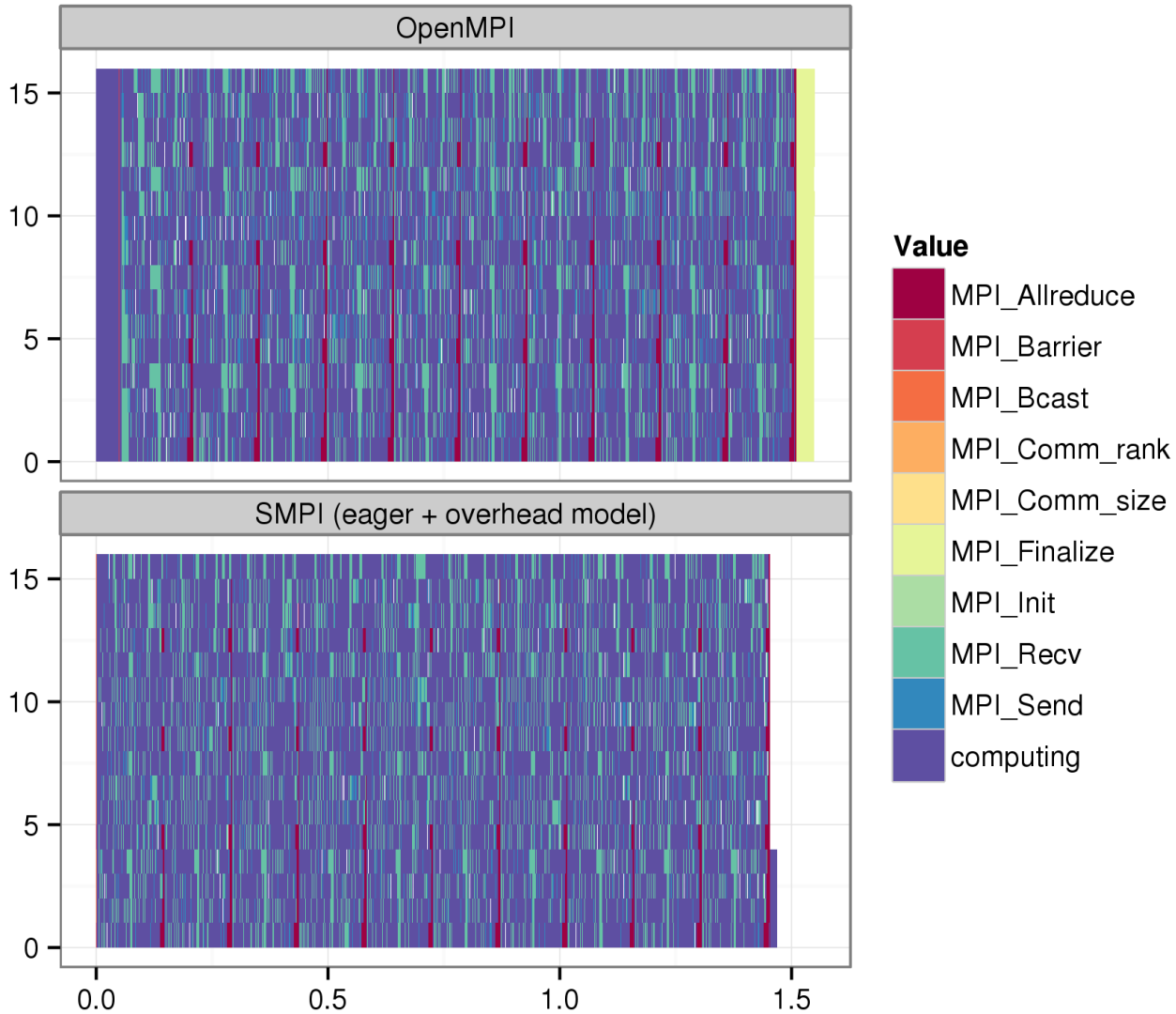


Synchronous mode ($k > S_d$)



(The SimGrid default model captures these effects, and much more)

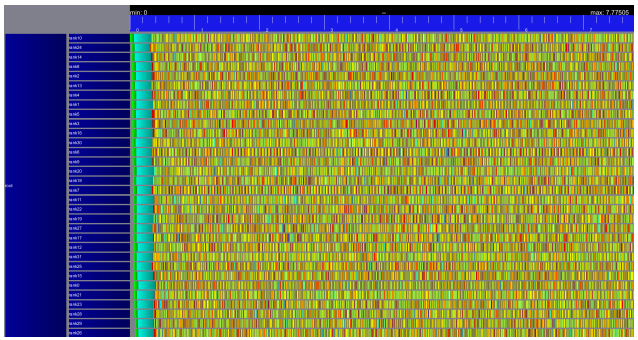
And this just works!

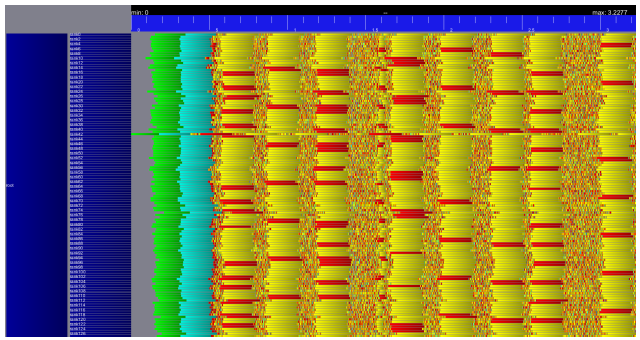


- Sweep3D: Simple (but not trivial) application predicted in all details
- Graphène (16 procs), OpenMPI, TCP, Gigabit Ethernet without overfitting FX :)

Reality is sometimes . . . surprising

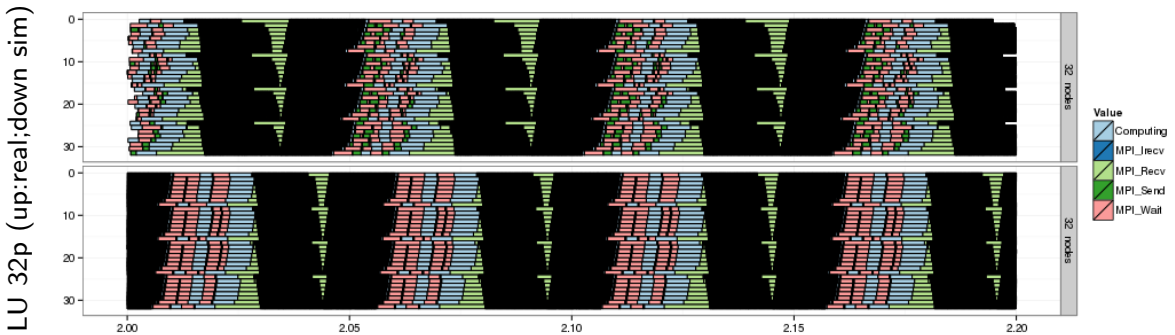
Hardware sucks

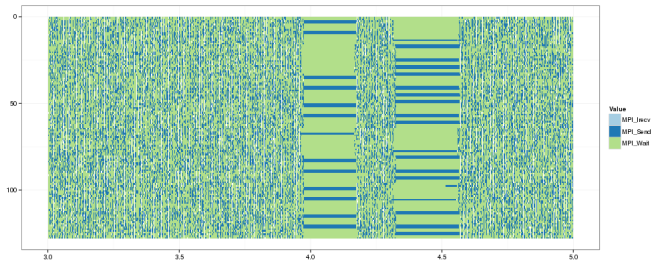




- BigDFT on Graphene
- Harware Bug (?)
- Packet drops; Timeouts

TCP sucks





Congestion $\rightsquigarrow$ slows down
Speed $= 0 \rightsquigarrow$ timeout, and reset

We can incorporate these effects in our models (and others)

- **But don't you want to fix reality?**
- We were modeling to understand, that's a huge victory!

Conclusions on the SONGS project

- Goal: Computational Science of Computer Systems
  - Systems are too large, dynamic and complex for a reductionist approach
  - We need models to understand, and simulations to predict
- **Realistic models** of Modern Systems (DataGrids, P2P, Clouds & HPC)
- **Efficient Methodology**: Planning, Simulation and Analysis (with Open Science)

The project is Fully on Track

- Work factorization is really effective ($\leadsto$ productivity gain)
- Many, many results. In all 8 WP.
- WP4: Sufficient models to predict MPI applications
  - Many dark areas remain, but this is unpreceded
  - Non-trivial but correct predictions; Reality sometimes worse than Simulation :)
- WP8: Open Science opens a brave new world

There is much more to discover during the second half

- If things remain the same, all fixed goals should be reached
- The project is attracting many external contributors
- Are we experiencing the emergence of the vivid research community we need?

Take Away Messages

SimGrid will prove helpful to your research

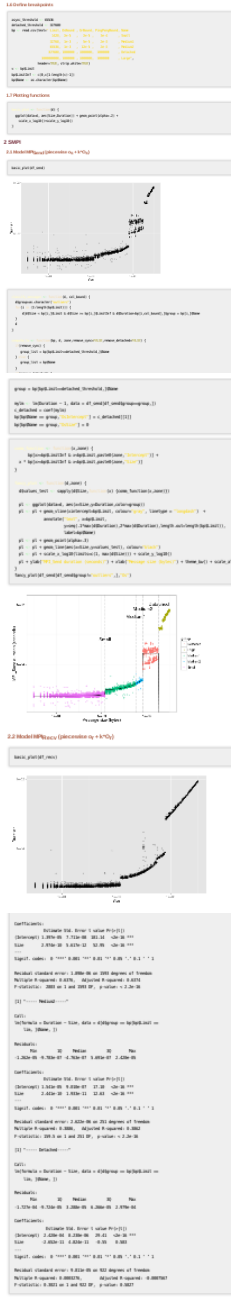
- Versatile: Used in several communities (scheduling, GridRPC, HPC, P2P, Clouds)
- Accurate: Model limits known thanks to validation studies
- Sound: Easy to use, extensible, fast to execute, scalable to death, well tested
- Open: User-community much larger than contributors group/ GPL
  120 publications (110 distinct authors, 5 continents), 4 PhD/25+commiters,5+ unaffiliated
- Around since over 10 years, and ready for at least 10 more years

Welcome to the Age of (Sound) Computational Science

- Discover: `http://simgrid.gforge.inria.fr/`
- Learn: 101 tutorials, user manuals and examples
- Join: user mailing list, #simgrid on irc.debian.org

  We even have some open positions ;)

Other finding of the project: Open Science



The Devil is in the Details *vs.* Reproducibility Graal

- Experiment description (environment / protocol) not trivial (déluge de données)
- Very sensible experiments: impact macro of micro errors
- Statistical post-processing more and more advanced

But that works, too!

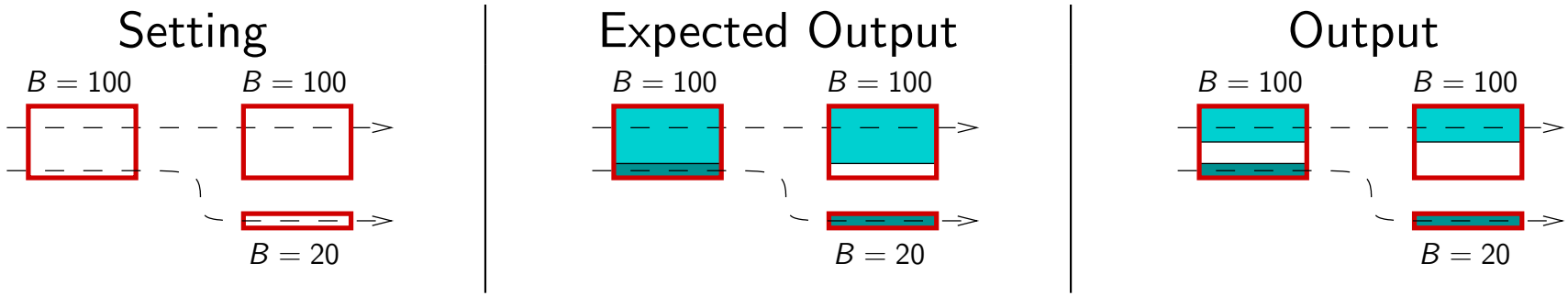
- Grid'5000 is very precious: hardware, but also know-how
- Our tools (YMMV): git + org-mode + R
- *Computational scientists* already use them, btw

We just need to convince our community ;)

- *I found the results section of this paper to be pretty weak.*
- *If less accurate models drive the user to the same conclusions (as Fig. 8 indicates), why we need more complex models?*

Invalidating Simulators from the Litterature

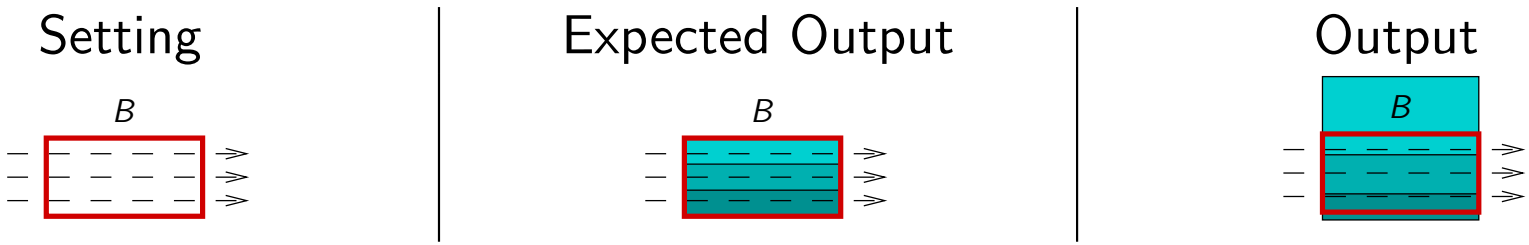
Naive flow models documented as wrong



Known issue in Narses (2002), OptorSim (2003), GroudSim (2011).

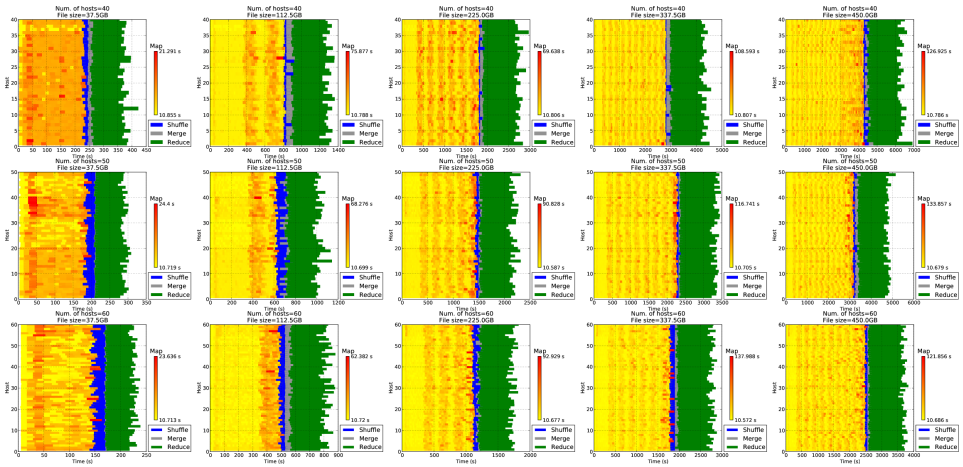
Validation by general agreement

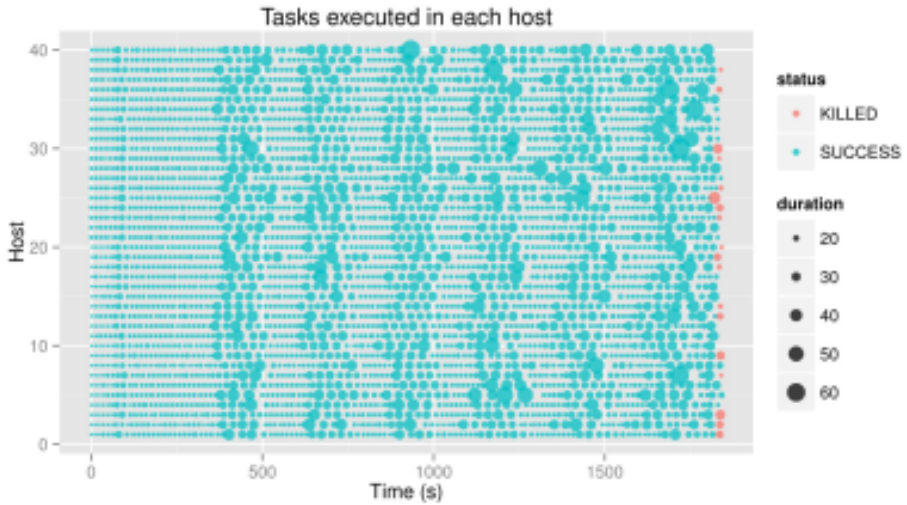
> *"Since SimJava and GridSim have been extensively utilized in conducting cutting edge research in Grid resource management by several researchers, bugs that may compromise the validity of the simulation have been already detected and fixed."* CloudSim, ICPP'09

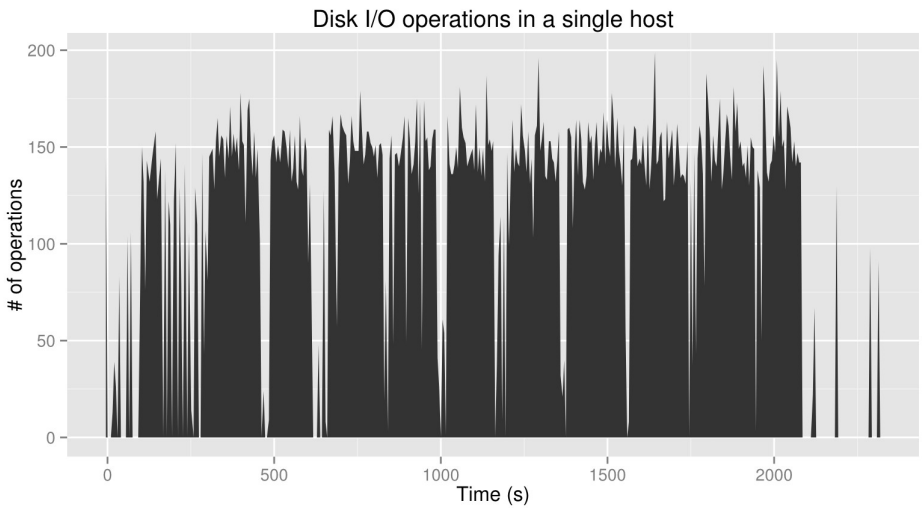


Buggy flow model (GridSim 5.2, Nov. 25, 2010). Similar issues with naive packet-level models.

MapReduce on Grid'5000







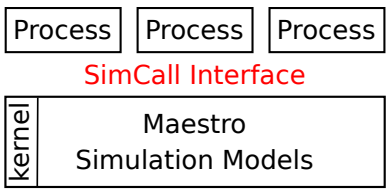
- Huge CPU slowdown
- Due to the IDE disks
  Does not happen in SATA

Can be modeled, but you have to know

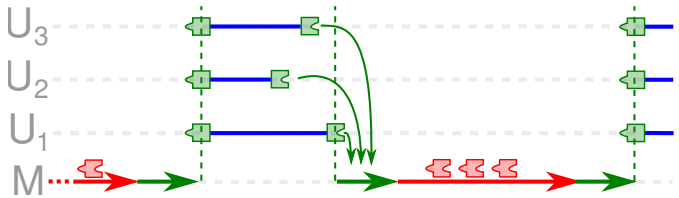
SimGrid is an Operating Simulator

OS-like internal design, isolating user processes with **simcalls**

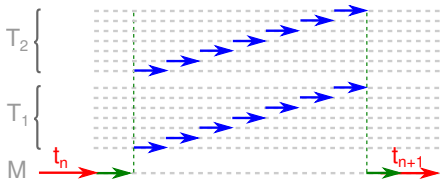
Functional View



Temporal View



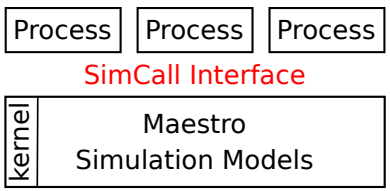
Implementation



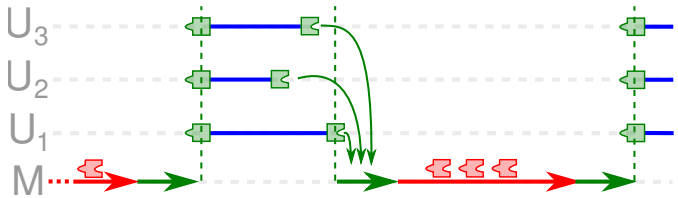
SimGrid is an Operating Simulator

OS-like internal design, isolating user processes with **simcalls**

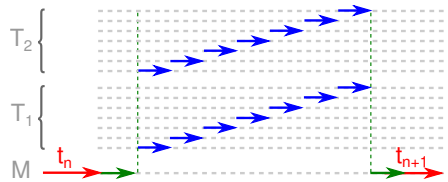
Functional View



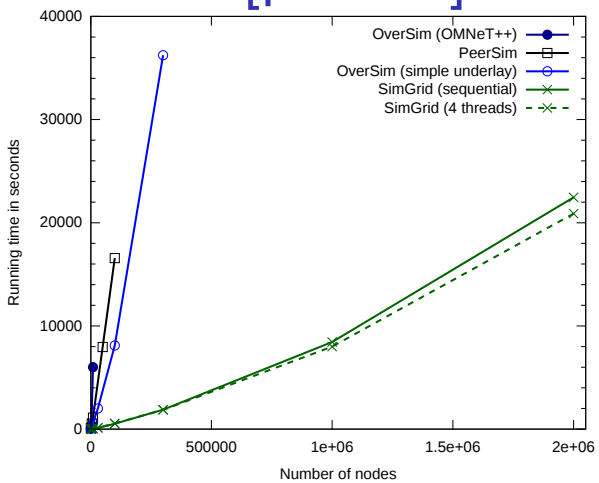
Temporal View



Implementation



Efficient [parallel] simulation



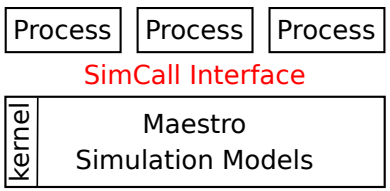
dPeerSim: 2LP $\rightsquigarrow$ 4h / 16LP $\rightsquigarrow$ 1h

(but only 47s in sequential PeerSim, and 5s with SimGrid :)

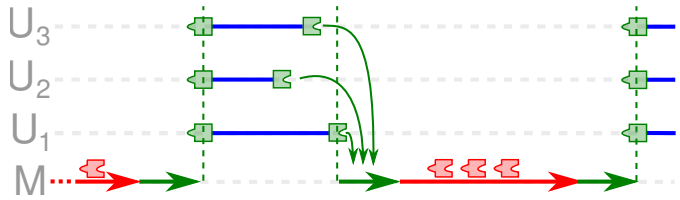
SimGrid is an Operating Simulator

OS-like internal design, isolating user processes with **simcalls**

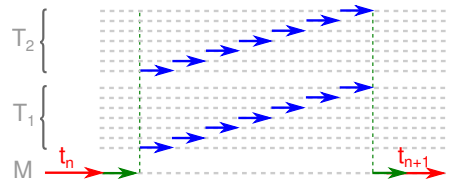
Functional View



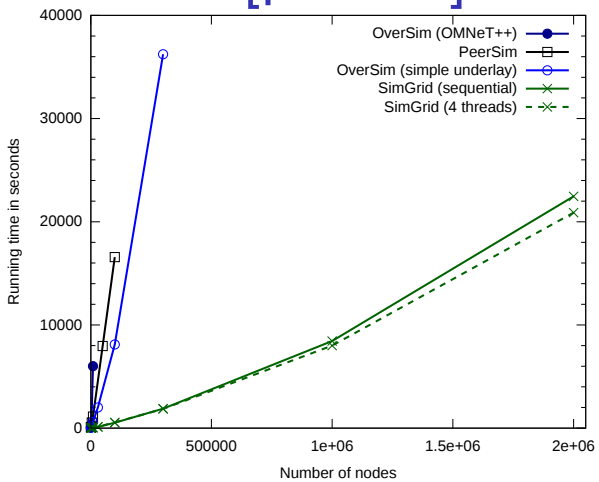
Temporal View



Implementation



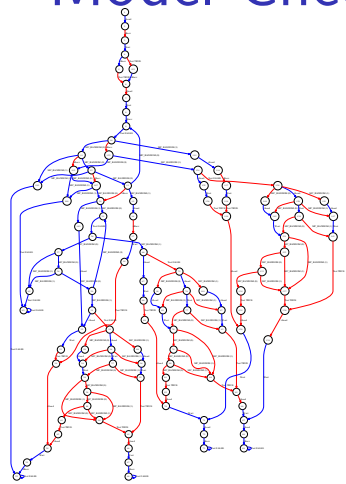
Efficient [parallel] simulation



dPeerSim: 2LP $\rightsquigarrow$ 4h / 16LP $\rightsquigarrow$ 1h

(but only 47s in sequential PeerSim, and 5s with SimGrid :)

Model-Checking (Safety & Liveness)



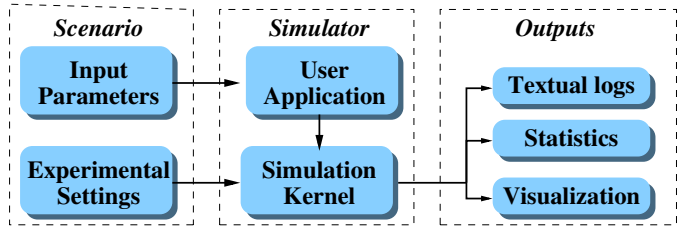
Exhaustive Chord

(2 processes)

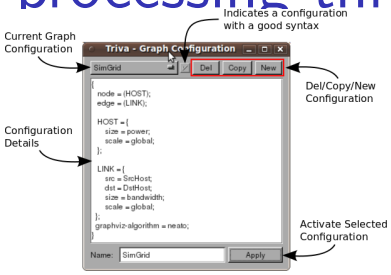
- Aims at bug finding, not assessment
- System State Equality
- + DPOR Reduction
- Soon more parallelism
- Soon statistical MC

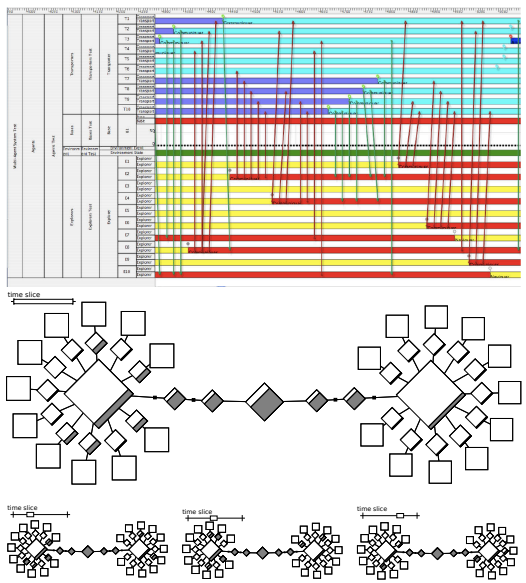
Toward an Integrated Scientific Workflow

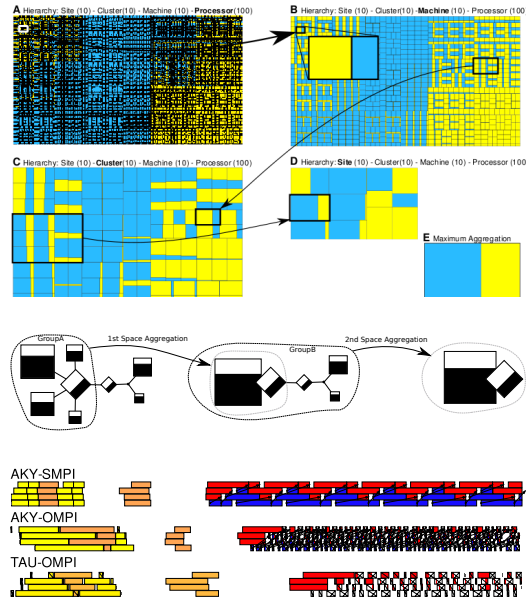
1. Prepare the experimental scenarios
2. Launch thousands of simulations
3. Post-processing and result analysis



Post-processing through Vizualization







Platform and Workload Generation