Experimenting on Architectures for High Performance Computing

Lucas Nussbaum lucas.nussbaum@loria.fr

École ARCHI 2017









General Outline

- HPC architectures
- Experimentation and reproducible research in Computer Science
- Grid'5000: a Large-Scale Instrument for Parallel and Distributed Computing Experiments

HPC Architectures

Several slides taken from Giovanni Erbacci (CINECA) and Jack Dongarra. Thanks!

Advanced School on PARALLEL COMPUTING

Computational Sciences

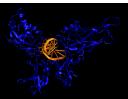
Computational science (with theory and experimentation), is the "third pillar" of scientific inquiry, enabling researchers to build and test models of complex phenomena



Ouick evolution of innovation:

- · Instantaneous communication
- Geographically distributed work
- Increased productivity
- More data everywhere
- · Increasing problem complexity
- Innovation happens worldwide







Computational Sciences today

Multidisciplinary and multiscale problems

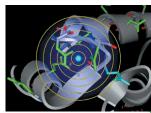
Coupled applications

- Full simulation of engineering systems
- Full simulation of biological systems
- Astrophysics
- Materials science
- Bio-informatics, proteomics, pharmaco-genetics
- Scientifically accurate 3D functional models of the human body
- Biodiversity and biocomplexity
- Climate and Atmospheric Research
- Energy
- Digital libraries for science and engineering



Complex mathematical models

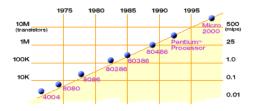






Moore's Law



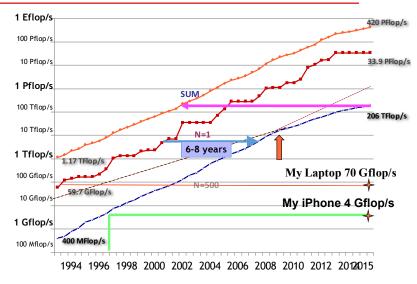


- Empirical law which states that the complexity of devices (number of transistors per square inch in microprocessors) doubles every 18 months.
- Gordon Moore, INTEL co-founder, 1965
- It is estimated that Moore's Law will still hold in the near future but applied to the number of cores per processor





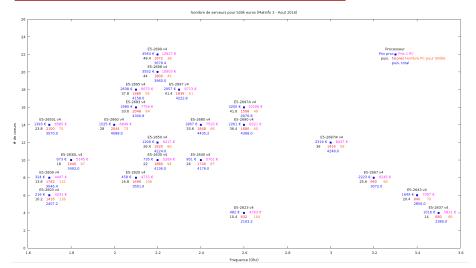
Performance Development of HPC over the Last 24 Years from the Top500



Hardware

- Tradeoff between:
 - Performance (FLOPS)
 - Cost
 - ★ Purchase
 - ★ Operation (energy consumption ~> cooling)

Intel CPUs (Broadwell-EP, dual-CPU)



(also: Turbo-Boost)

by Gérald Monard (dir. mésocentre EXPLOR)

Communication layer

- Key performance factor: latency
 - Many small, blocking messages during computations
- Specific networking technologies:
 - Myrinet (ancient, 28% of TOP500 in 2005)
 - InfiniBand (main vendor: Mellanox)
 - Intel Omni-Path (since 2015)
 - Support for collective communications in hardware

10G-Ethernet vs InfiniBand

RTT between two nodes (using Intel MPI Benchmarks' PingPong)

10G-Ethernet

InfiniBand FDR (56 Gb/s)

#bvtes	#repetitions	t[usec]	Mbytes/sec	#hvtes	#repetitions	t [usec]	Mbytes/sec
0	1000	17.17	0.00	0	1000	1.19	0.00
1	1000	15.74	0.06	1	1000	1.22	0.78
2	1000	15.06	0.13	2	1000	1.23	1.55
4	1000	15.21	0.25	4	1000	1.23	3.10
8	1000	15.40	0.50	8	1000	1.24	6.15
16	1000	15.10	1.01	16	1000	1.25	12.16
32	1000	15.05	2.03	32	1000	1.27	24.07
64	1000	15.59	3.92	64	1000	1.32	46.08
128	1000	15.72	7.77	128	1000	1.96	62.20
256	1000	18.61	13.12	256	1000	2.07	118.17
512	1000	23.91	20.42	512	1000	2.21	221.30
1024	1000	28.55	34.20	1024	1000	2.53	386.29
2048	1000	49.10	39.78	2048	1000	3.12	625.68
4096	1000	61.99	63.02	4096	1000	3.71	1052.75
8192	1000	61.75	126.53	8192	1000	5.11	1527.79
16384	1000	62.36	250.56	16384	1000	6.74	2319.08
32768	1000	92.19	338.98	32768	1000	9.37	3334.06
65536	640	164.89	379.04	65536	640	14.48	4315.46
131072	320	298.79	418.35	131072	320	25.00	4999.99
262144	160	485.18	515.28	262144	160	45.54	5489.57
524288	80	860.18	581.27	524288	80	86.79	5761.11
1048576	40	1310.61	763.00	1048576	40	169.26	5907.99
2097152	20	2428.78	823.46	2097152	20	336.05	5951.48
4194304	10	4810.80	831.46	4194304	10	667.61	5991.54

Accelerators

- GPGPU (General-purpose computing on graphics processing units), since 2001
 - ◆ E.g. Nvidia Tesla for the HPC market
 - P100: 3500 cores @ 1.3 GHz, 8-10 TFLOPS SP, 4-5 TFLOPS DP
- Xeon Phi (Intel MIC architecture Many Integrated Cores)
 - Knights Corner (2013)
 - ★ PCIe card, 60 cores, 240 threads, 1 1.2 TFLOPS
 - ★ Virtualization layer to run standard x86 code
 - KNL Knights Landing (2016)
 - ★ 64-72 cores (x4 threads), 2.6 3.4 TFLOPS
 - ★ Host processor (CPU) optionally with integrated Omni-Path adapter

Storage

- Parallel file systems
 - Lustre, GPFS, BeeGFS, etc.
- ► Local storage on nodes, sometimes
- NVM Express (NVMe) ≈ SSD on PCIe

Outside of the Intel world

- China leading TOP500 with Sunway TaihuLight System
 - Using 40 960 custom-made Sunway SW26010 CPUs (1.4 GHz)
 - (Because U.S. banned Intel from supplying Xeon chips to top 4 China supercomputing centers)
- Some IBM Blue Gene/Q (IBM A2 processor, Power architecture)
- Some interest for ARM CPUs
 - Because of higher energy efficiency (e.g. big.LITTLE)

Challenge: efficient use in software

- HPC software: runtime, libraries, applications, tightly coupled with hardware
- A lot of legacy code (often in FORTRAN)
- With a lot of domain-specific knowledge
- Difficult to adjust to new architectures
- Also, speedup limited by the sequential part (Amdahl's Law)
 - Harder to make use of many slower cores



Experimentation and Reproducible Research in Computer Science

Validation in (Computer) Science

- Two classical approaches for validation:
 - Formal: equations, proofs, etc.
 - Experimental, on a scientific instrument
- Often a mix of both:
 - In Physics
 - In Computer Science
- Quite a lot of formal work in Computer Science
- But also quite a lot of experimental validation
 - ◆ Distributed computing, networking ~ testbeds (IoT-LAB, Grid'5000)
 - Language/image processing → evaluations using large corpuses



Validation in (Computer) Science

- Two classical approaches for validation:
 - Formal: equations, proofs, etc.
 - Experimental, on a scientific instrument
- Often a mix of both:
 - In Physics
 - In Computer Science
- Quite a lot of formal work in Computer Science
- But also quite a lot of experimental validation
 - ◆ Distributed computing, networking ~ testbeds (IoT-LAB, Grid'5000)
 - ◆ Language/image processing → evaluations using large corpuses





(Poor) state of experimentation in CS

- 1994: survey of 400 papers¹
 - among published CS articles in ACM journals, 40%-50% of those that require an experimental validation had none
- ▶ 1998: survey of 612 papers²
 - too many papers have no experimental validation at all
 - too many papers use an informal (assertion) form of validation
- 2009 update: situation is improving³

¹Paul Lukowicz et al. "Experimental Evaluation in Computer Science: A Quantitative Study". In: *Journal of Systems and Software* 28 (1994), pages 9–18.

²M.V. Zelkowitz and D.R. Wallace. "Experimental models for validating technology". In: *Computer* 31.5 (May 1998), pages 23–31.

³Marvin V. Zelkowitz. "An update to experimental models for validating computer technology". In: *J. Syst. Softw.* 82.3 (Mar. 2009), pages 373–376.

(Poor) state of experimentation in CS (2)

Most papers do not use even basic statistical tools

Papers published at the Europar conference⁴

Year	Tot. papers	With error bars	Percentage			
2007	89	5	5.6			
2008	89	3	3.4			
2009	86	2	2.4			
2010	90	6	6.7			
2011	81	7	8.6			
2007-2011	435	23	5.3			

- 2007: Survey of simulators used in P2P research⁵
 - Most papers use an unspecified or custom simulator

⁴Study carried out by E. Jeannot.

⁵S. Naicken et al. "The state of peer-to-peer simulators and simulations". In: *SIGCOMM Comput. Commun. Rev.* 37.2 (Mar. 2007), pages 95–98.

State of experimentation in other sciences

- 2008: Study shows lower fertility for mices exposed to transgenic maize
 - AFSSA report⁶:
 - ★ Several calculation errors have been identified
 - ★ led to a false statistical analysis and interpretation

⁶Opinion of the French Food Safety Agency (Afssa) on the study by Velimirov et al. entitled "Biological effects of transgenic maize NK603xMON810 fed in long-term reproduction studies in mice"

State of experimentation in other sciences

- 2008: Study shows lower fertility for mices exposed to transgenic maize
 - AFSSA report⁶:
 - ★ Several calculation errors have been identified
 - ★ led to a false statistical analysis and interpretation
- 2011: CERN Neutrinos to Gran Sasso project: faster-than-light neutrinos
 - 2012: caused by timing system failure

⁶Opinion of the French Food Safety Agency (Afssa) on the study by Velimirov et al. entitled "Biological effects of transgenic maize NK603xMON810 fed in long-term reproduction studies in mice"

State of experimentation in other sciences

- 2008: Study shows lower fertility for mices exposed to transgenic maize
 - AFSSA report⁶:
 - ★ Several calculation errors have been identified
 - ★ led to a false statistical analysis and interpretation
- 2011: CERN Neutrinos to Gran Sasso project: faster-than-light neutrinos
 - 2012: caused by timing system failure
- Solution
 Solution</p
- But some errors are properly identified
 - Stronger experimental culture in other (older?) sciences?
 - ★ Long history of costly experiments, scandals, . . .

⁶Opinion of the French Food Safety Agency (Afssa) on the study by Velimirov et al. entitled "Biological effects of transgenic maize NK603xMON810 fed in long-term reproduction studies in mice"

Reproducible Research movement

- Originated mainly in computational sciences (Computational biology, data-intensive physics, etc.)
- Explores methods and tools to enhance experimental practices
 - Enable others to reproduce and build upon one's work
- Several different motivations

Reproducible Research movement

- Originated mainly in computational sciences (Computational biology, data-intensive physics, etc.)
- Explores methods and tools to enhance experimental practices
 - Enable others to reproduce and build upon one's work
- Several different motivations

Do The Right Thing TM

- Fundamental basis of the scientific method
- ► K. Poppler, 1934: non-reproducible single occurrences are of no significance to science
- Increases transparency, reduces rejection of the scientific community (climate, GMO)



This may be an interesting contribution but:

► This average value must hide something

- This average value must hide something
- ► As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not

- This average value must hide something
- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- ► That can't be true, I'm sure they removed some points

- This average value must hide something
- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- ► That can't be true, I'm sure they removed some points
- Why is this graph in logscale? How would it look like otherwise?

- This average value must hide something
- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- ► That can't be true, I'm sure they removed some points
- ▶ Why is this graph in logscale? How would it look like otherwise?
- The authors decided to show only a subset of the data. I wonder what the rest looks like

- This average value must hide something
- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- ► That can't be true, I'm sure they removed some points
- ▶ Why is this graph in logscale? How would it look like otherwise?
- The authors decided to show only a subset of the data. I wonder what the rest looks like
- There is no label/legend/... What is the meaning of this graph? If only I could access the generation script

I thought I used the same parameters but I'm getting different results!

- I thought I used the same parameters but I'm getting different results!
- The new student wants to compare with the method I proposed last year

- I thought I used the same parameters but I'm getting different results!
- ► The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember

- I thought I used the same parameters but I'm getting different results!
- ► The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- ► The damned fourth reviewer asked for a major revision and wants me to change figure 3 :(

Frustration as an author

- I thought I used the same parameters but I'm getting different results!
- ► The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- ► The damned fourth reviewer asked for a major revision and wants me to change figure 3 :(
- Which code and which data set did I use to generate this figure?

Frustration as an author

- I thought I used the same parameters but I'm getting different results!
- ► The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- ► The damned fourth reviewer asked for a major revision and wants me to change figure 3 :(
- Which code and which data set did I use to generate this figure?
- It worked yesterday!

Frustration as an author

- I thought I used the same parameters but I'm getting different results!
- ► The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- ► The damned fourth reviewer asked for a major revision and wants me to change figure 3 :(
- Which code and which data set did I use to generate this figure?
- It worked yesterday!
- 6 months later: why did I do that?

Accelerate your research, increase your impact

- Makes it easier to base on your previous work
- Makes it easier for others to base on your work
 - More visibility, more collaborations
 - More citations
 Sharing Detailed Research Data Is Associated with Increased Citation Rate⁷

⁷Heather A. Piwowar et al. "Sharing Detailed Research Data Is Associated with Increased Citation Rate". In: *PLoS ONE* 2.3 (Mar. 2007), e308. DOI: 10.1371/journal.pone.0000308. URL: http://dx.plos.org/10.1371/journal.pone.0000308.

Because you might be forced to

- NSF policy on the dissemination and sharing of research results
- ► H2020 Open Research Data Pilot⁸ (for 20% of H2020):
 - 1. participating projects are required to deposit the research data described above, preferably into a research data repository. [...]
 - 2. as far as possible, projects must then take measures to enable for third parties to access, mine, exploit, reproduce and disseminate (free of charge for any user) this research data.

At the same time, projects should provide information via the chosen repository about tools and instruments at the disposal of the beneficiaries and necessary for validating the results, for instance specialised software or software code, algorithms, analysis protocols, etc. Where possible, they should provide the tools and instruments themselves.

Nothing at ANR yet?

⁸European Commission. *Guidelines on Open Access to Scientific Publications and Research Data in Horizon 2020.* Dec. 2013. URL: http://ec.europa.eu/research/participants/data/ref/h2020/grants_manual/hi/oa_pilot/h2020-hi-oa-pilot-guide_en.pdf.

Different types of experimental reproducibility⁹

 Replications that vary little or not at all with respect to the reference experiment

same method, environment, parameters \rightarrow same result

- Also called Replicability
- Replications that do vary but still follow the same method as the reference experiment

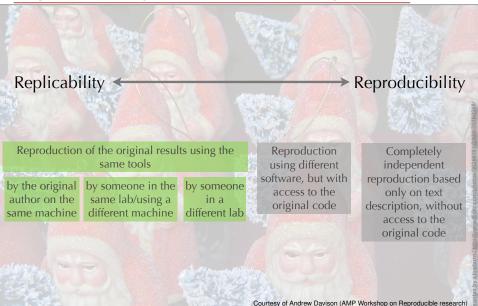
same method, but different {env., params} \rightarrow same conclusion

- Example: different testbed
- Replications that use different methods to verify the reference experiment results

different method \rightarrow same conclusion

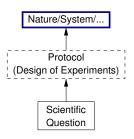
⁹Omar S. Gómez et al. "Replications types in experimental disciplines". In: *ESEM'10*. 2010.

Reproducibility: what are we talking about?



Author

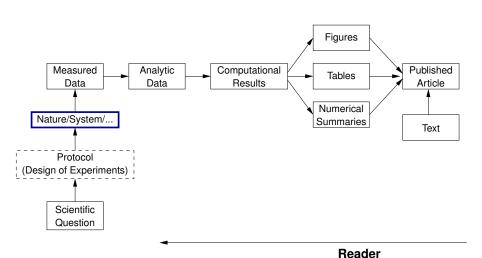
Published Article



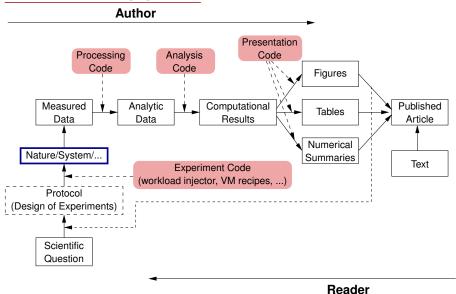
Reader

Inspired by Roger D. Peng's lecture on reproducible research, May 2014
Improved by Arnaud Legrand

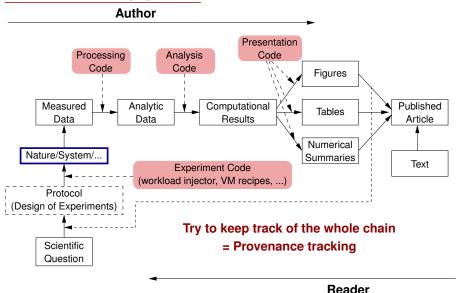
Author



Inspired by Roger D. Peng's lecture on reproducible research, May 2014



Inspired by Roger D. Peng's lecture on reproducible research, May 2014
Improved by Arnaud Legrand

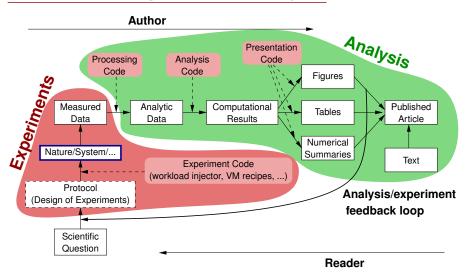


Inspired by Roger D. Peng's lecture on reproducible research, May 2014
Improved by Arnaud Legrand

Reproducible research challenges

- Better descriptions of each step
 - Executable descriptions?
 - Efficient/optimal descriptions?
- ► Facilitate/automate provenance tracking
 - ♦ ~ move burden away from experimenter
 - Testbeds or experiment management tools with built-in support for provenance collection?
- Ensure that provenance data is sufficient/complete
- Provide sustainable/durable/dependable long-term storage
 - Stable infrastructure
 - Open, standard formats
- Keep stable references between article, code, data

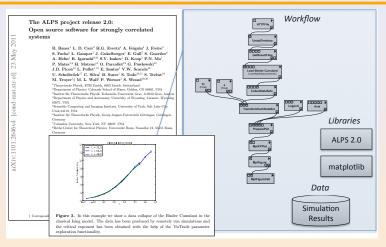
Solutions for reproducible analysis



Note: Analysis is generally not very domain-specific

Vistrails: a workflow engine for provenance tracking

An Provenance-Rich Paper: ALPS2.0



[Bauer et al., JSTAT 2011]

Chronicing computations in real-time

VCR computation platform Plugin = Computation recorder

```
Regular program code
figure1 = plot(x)
save(figure1,'figure1.eps')
> file /home/figure1.eps saved
>
```

Chronicing computations in real-time

VCR computation platform Plugin = Computation recorder

```
Program code with VCR plugin
```

```
repository vcr.nature.com
verifiable figure1 = plot(x)
```

- > vcr.nature.com approved:
- > access figure1 at https://vcr.nature.com/ffaaffb148d7

Word-processor plugin App

LaTeX source

\includegraphics{figure1.eps}

LaTeX source with VCR package

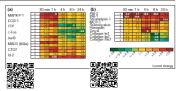
\includeresult{vcr.thelancet.com/ffaaffb148d7}

Permanently bind printed graphics to underlying result content

Research Paper Analysis of replicative senescence Shelton et al. 943

Figure 3

Time course of serum stimulation. (a) Early passage (E: PD30) or late passage (L: PD89) BJ cultures were held in 0.5% serum for 2 days, then stimulated with 10% FBS. RNA levels from cultures at the indicated time points (Cy5 channel) were compared with the uninduced starting culture (Cy3 channel). Positive values indicate higher expression in induced cells; negative values indicate lower expression in induced cells. Question marks indicate that there was insufficient signal for detection. A complete listing of serum-responsive genes from this analysis is provided in Supplementary material. (b) The serum-responsiveness of select senescence-regulated genes in early passage (PD30) BJ fibroblasts.



senescence response appears to overlap substantially with gene expression patterns observed in activated fibroblasts during wound healing [24–26], MCP-1, Gro-6, II.-1]b and II.-15 are strong effectors of macrophage and neutrophil III.-15 are strong effectors of macrophage and neutrophil Toil (Til-4) in senescent fibroblasts confirms the overall immune response behavior as resonecence. Til-4 is an III.-1 receptor homolog and is implicated in the activation of the gene regulatory protein NF-643, a function proposed to part of the innate immune response [29]. The induction of II.-15 at senescence is also consistent with an innate interest of the confirmation of the confirmation of the contraction of the confirmation of the confirmation of the part of the confirmation of the confirmation of the contraction of the confirmation of the confirmation of the contraction of the confirmation of the confirmation of the contraction of the confirmation of the

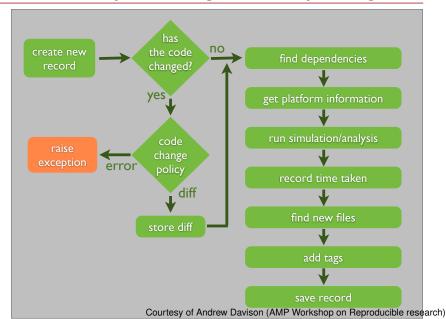
Deficiencies in the response of senescent cells to serum stimulation have been reported, and include an inability to induce the expression of e.for mRNA [31] and markers of late GI and Sphase [32]. In response to serum, expression of inflammatory chemokines, matrix-degrading proteases and their modulators is induced in early-passage difficulties and fibroblasts, and expression of matrix collagens is reduced. This transient busts of activity may represent the natural response of these cells in wound repair [24], Id-2 transcripts were hyper-induced in serum-stimulated sensener. Ifforoblasts, which may be linked to the ability of cytokines to induce Id-2 curvession I331. The levels of mRNAs for states overlap substantially with those in telomereinduced senescence (W.F., D.N.S., R. Allsopp, S. Lowe, and G. Ferbeyre, unpublished observations) and thus are likely to use many of the same activation processes.

The pattern of gene expression at senescence varies substantially in different cell types. Although the expression of matrix and structural proteins, such as the collagens, keratins and auxiliary factors, is repressed in RPE cells. inflammatory regulators are not induced, in contrast to dermal fibroblasts. Physiologically, this would make sense as an acute inflammatory response in a tissue critical for normal vision would be likely to have deleterious consequences. However, as the RPE layer has a central role in the deposition and maintenance of extracellular matrix in the retina, decrements in the ability of senescent RPE cells to maintain appropriate expression patterns, as evidenced by decreased expression of collagens, keratins, aggrecan, transglutaminase and so on, would be predicted to have adverse affects on retinal architecture. Dysfunction of the RPE cell layer is considered to be a substantial factor in the development of age-related macular degeneration [36].

Surprisingly, early-passage HUVECs overexpress many of the markers associated with senescence in dermal fibroblasts, such the pro-inflammatory molecules IL-15 and Tlr-4. In very low serum, vascular endothelial cells are

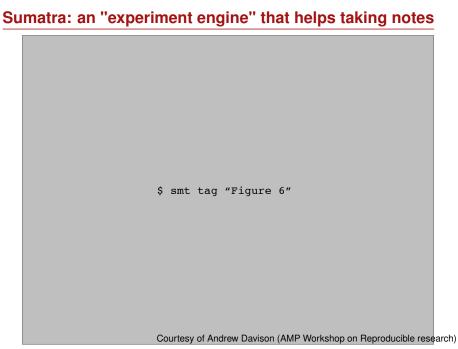
urtesy of Matan Gavish and David Donoro (AMP Workshop on Reproducible research)

Sumatra: an "experiment engine" that helps taking notes

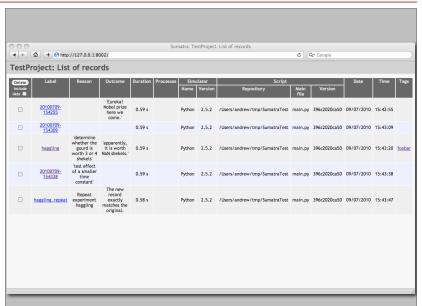


Sumatra: an "experiment engine" that helps taking notes

\$ smt comment 20110713-174949 "Eureka! Nobel prize here we come."

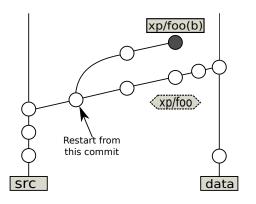


Sumatra: an "experiment engine" that helps taking notes



Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

Git + Org-mode workflow¹⁰



- Track link between code, experiments and results using Git branches
- Integrates with Org-mode for litterate programming

¹⁰Luka Stanisic et al. "An Effective Git And Org-Mode Based Workflow For Reproducible Research". In: *SIGOPS Oper. Syst. Rev.* 49.1 (Jan. 2015), pages 61–70.

Sweave: literate programming with LaTeX and R

Sweave Example 1

```
\documentclass[a4paper]{article}
\title{Sweave Example 1}
\author{Friedrich Leisch}
\begin{document}
\maketitle
In this example we embed parts of the
examples from the \texttt{kruskal.test} help
page into a \LaTeX{} document:
<<>>=
data(airquality)
library(ctest)
kruskal.test(Ozone ~ Month, data = airquality)
which shows that the location parameter of
the Ozone distribution varies significantly
from month to month. Finally we include a
boxplot of the data:
\begin{center}
<<fig=TRUE,echo=FALSE>>=
boxplot(Ozone ~ Month, data = airquality)
\end{center}
\end{document}
```

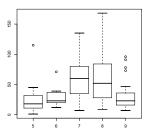
Friedrich Leisch May 21, 2007

In this example we embed parts of the examples from the kruskal.test help page into a LATpX document:

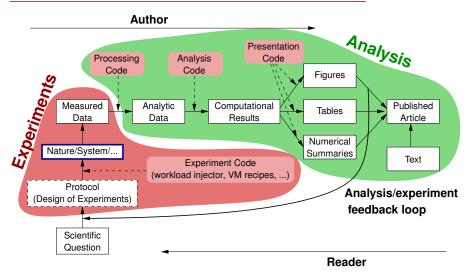
```
> data(airquality)
> library(ctest)
> kruskal.test(Ozone ~ Month, data = airquality)

Kruskal-Wallis rank sum test
```

data: Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value = 6.901e-06
which shows that the location parameter of the Ozone distribution varies sigmificantly from mouth to month. Finally we include a boxplot of the data:

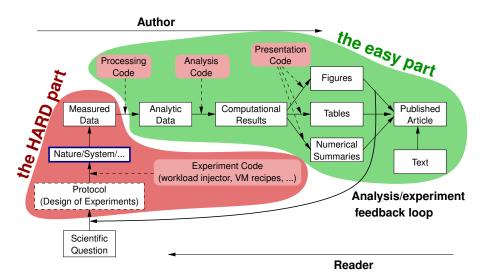


Solutions for reproducible experiments



Note: Experiments is generally quite domain-specific

The Distributed Computing point-of-view



The Distributed Computing point-of-view

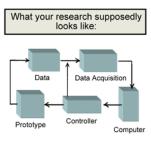


Figure 1. Experimental Diagram

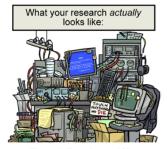


Figure 2. Experimental Mess

- Rely on large, distributed, hybrid, prototype hardware/software
- Measure execution times (makespans, traces, ...)
- Many parameters, very costly and hard to *reproduce*

Similar issues in e.g. Wireless Sensor Networks research

Experimental environment management

Experimental environment management

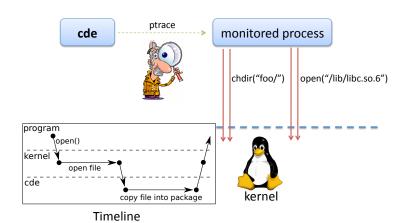
- ► How to describe/provide the software environment used?

 *I used OpenMPI on Debian ®
- Obvious solution: virtual machines

Yes, but:

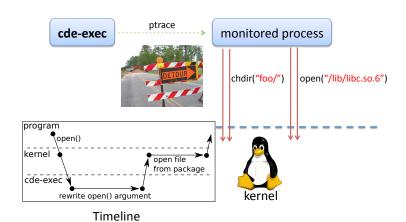
- ◆ Only provides the final result, not the logic behind each change
 → easy to forget why/when something was customized
- No synthetic description: the full image must be provided
- ◆ Cannot really be used as a basis for future experiments
 (≈ object vs source code, preferred form for making modifications)

Creating a package with cde



¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

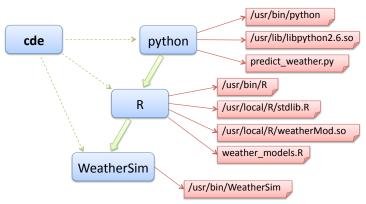
Executing a package with cde-exec



¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

Creating a package with cde

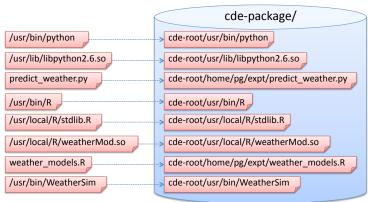
cd /home/pg/expt/
cde python predict_weather.py



¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

Creating a package with cde

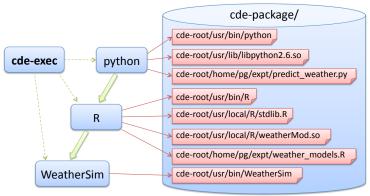
cd /home/pg/expt/
cde python predict_weather.py



¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

Executing a package with cde-exec

cd cde-package/cde-root/home/pg/expt/
cde-exec python predict_weather.py



¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

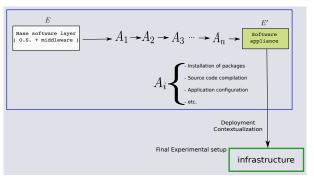
But:

- Does not provide the preferred form for making modifications
- Execution is slower (2% 30%) due to ptrace

¹¹Philip J. Guo and Dawson Engler. "CDE: Using System Call Interposition to Automatically Create Portable Software Packages". In: USENIX ATC. 2011.

Kameleon: reproducible software appliances¹²

- Using recipes (high-level description)
 - Similar to cfengine, Puppet, Chef in the sysadmin world



- ▶ Persistent cache to allow re-generation without external resources (Linux distribution mirror) ~> self-contained archive
- Supports LXC, Docker, VirtualBox, qemu, Kadeploy images, etc.

¹²Cristian Camilo Ruiz Sanabria et al. "Reproducible Software Appliances for Experimentation". In: *TRIDENTCOM*'2014.

Improving description and control of experiments

- Legacy way of performing experiments: shell commands
 - time-consuming
 - error-prone
 - details tend to be forgotten over time
- Promising solution: automation of experiments

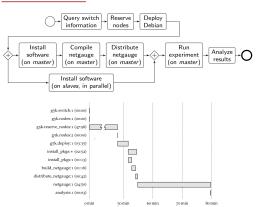


Tools for automation of experiments

- Several projects around Grid'5000 (but not specific to Grid'5000):
 - Expo (Cristian Ruiz)
 - Execo (Mathieu Imbert)
 - ◆ XPFlow (Tomasz Buchert)
- Others, for other scientific domains:¹³
 - Plush/Gush (PlanetLab)
 - OMF, NEPI (Wireless testbeds)
- Features:
 - Ease scripting of experiments in high-level languages (Ruby, Python)
 - Provide useful and efficient abstractions :
 - ★ Testbed management
 - ★ Local & remote execution of commands
 - Data management
 - Engines for more complex processes

¹³Tomasz Buchert et al. "A survey of general-purpose experiment management tools for distributed systems". In: *Future Generation Computer Systems* 45 (2015), pages 1–12.

XPFlow¹⁴



```
engine.process :exp do |site, switch|
    s = run q5k.switch, site, switch
    ns = run a5k.nodes. s
    r = run q5k.reserve nodes,
        :nodes => ns. : time => '2h'.
        :site => site. :type => :deploy
    master = (first of ns)
    rest = (tail of ns)
    run q5k.deplov,
        r, :env => 'squeeze-x64-nfs'
    checkpoint :deployed
    parallel :retry => true do
        forall rest do |slave|
            run :install pkgs, slave
        end
        sequence do
            run :install pkgs. master
            run :build netgauge, master
            run :dist netgauge,
                master, rest
        end
    end
    checkpoint :prepared
    output = run :netgauge, master, ns
    checkpoint :finished
    run :analysis, output, switch
end
```

Experiment description and execution as a Business Process Workflow

Supports parallel execution of activities, error handling, snapshotting, built-in logging, etc.

soon: automatic provenance collection

¹⁴Tomasz Buchert et al. "A workflow-inspired, modular and robust approach to experiments in distributed systems". In: *CCGRID*'2014.

Other related issues and initiatives

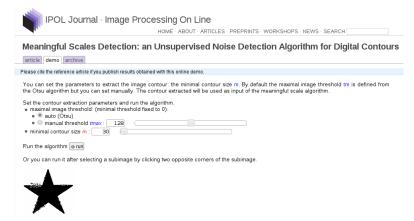
Preserving data and software

- No, your homepage is not a durable storage solution
 - Half-life of URLs in IEEE Computer and CACM: four years¹⁵
 - Y2K crisis: in 1999, 40% of companies had either lost or thrown away the original source code for their systems
 - Code Spaces (Git/SVN project hosting in AWS) hacked: all data lost
- Solutions exist:
 - Articles: ArXiv, HAL
 - Data: Zenodo/OpenAire (CERN, EU-funded), ISAAC (CINES), figshare (Cloud-based)
 - Software: Software Heritage (need to consider execution environment, interdependences, software evolution → more complex than books/articles/data)

¹⁵Diomidis Spinellis. "The Decay and Failures of Web References". In: *Commun. ACM* 46.1 (Jan. 2003), pages 71–77.

Online journals, companion websites

- Host code, allow execution (sometimes)
- Example: IPOL Journal Image Processing On Line¹⁶



Others: DAE, RunMyCode, etc.

¹⁶http://www.ipol.im/ (demo)

Evaluation campaigns & challenges

- ► Evaluate several algorithms against each other, on a given set of inputs
- Events co-hosted with conferences
- Examples in the language/signal processing community:
 - ♦ Music Information Retrieval Evaluation Exchange (MIREX)
 - Signal Separation Evaluation Campaign (SiSEC)
 - CHiME Speech Separation and Recognition Challenge
 - Shared Task on Parsing of morphologically-rich languages (SPMRL)

Artifacts evaluation / reproducibility committees

 Authors can submit an archive with the material needed to reproduce their results, and get a "Reproducible" stamp on their paper

¹⁷http://www.artifact-eval.org/

¹⁸http://ctuning.org/cm/wiki/index.php?title=Reproducibility

¹⁹http://db-reproducibility.seas.harvard.edu/

Artifacts evaluation / reproducibility committees

- Authors can submit an archive with the material needed to reproduce their results, and get a "Reproducible" stamp on their paper
- Questions:
 - How easy is it to use the provided artifact? (Easy to reuse)
 - Does the artifact help to reproduce the results from the paper? (Consistent)
 - What is the percentage of the results that can be reproduced? (Complete)
 - Does the artifact describe and demonstrate how to apply the presented method to a new input? (Well documented)

¹⁷http://www.artifact-eval.org/

¹⁸http://ctuning.org/cm/wiki/index.php?title=Reproducibility

¹⁹http://db-reproducibility.seas.harvard.edu/

Artifacts evaluation / reproducibility committees

- Authors can submit an archive with the material needed to reproduce their results, and get a "Reproducible" stamp on their paper
- Questions:
 - ♦ How easy is it to use the provided artifact? (Easy to reuse)
 - ◆ Does the artifact help to reproduce the results from the paper? (Consistent)
 - What is the percentage of the results that can be reproduced? (Complete)
 - Does the artifact describe and demonstrate how to apply the presented method to a new input? (Well documented)
- Introduced in several conferences:
 - ◆ Software engineering, programming languages¹⁷: ESEC/FSE 2011, ECOOP 2013, OOPSLA 2013, SAS 2013, PLDI 2014, ISSTA 2014, HSCC 2014
 - Compilation, parallel computing¹⁸: CGO 2015, PPoPP 2015
 - Databases: SIGMOD 2008¹⁹, VLDB 2013

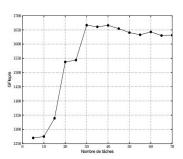
¹⁷http://www.artifact-eval.org/

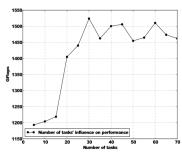
¹⁸http://ctuning.org/cm/wiki/index.php?title=Reproducibility

¹⁹http://db-reproducibility.seas.harvard.edu/

Realis @ COMPAS 2013 and 2014

- ► COMPAS: Conférence en Parallélisme, Architecture et Système
 - French-speaking, mostly for PhD students
- Realis: test reproducibility of papers submitted to COMPAS
 - Participating authors submit their experimentation description
 - Each author reproduces the experiments from another article
 - ★ Get the identical results, without contacting the authors
 - ★ Evaluate the quality (flexibility, robustness) of the approach
- Most results were reproduced (but none without contacting the authors)





Conclusions

- Reproducible research
 - A way to improve our daily work, with immediate benefits
 - An opportunity to think about our practices
 - A research field of its own
- Many solutions and tools are now ready for use
- Monthly online seminars: https://github.com/alegrand/RR_webinars

Grid'5000:

a Large-Scale Instrument for Parallel and Distributed
Computing Experiments

Distributed computing: a peculiar field in CS

- Performance and scalability are central to results
 - But depend greatly on the environment (hardware, network, software stack, etc.)
 - Many contributions are about fighting the environment
 - ★ Making the most out of limited resources
 - ★ Handling performance imbalance ~ load balancing
 - ★ Handling faults ~ fault tolerance
 - ★ Hiding complexity ~ abstractions: middlewares, runtimes

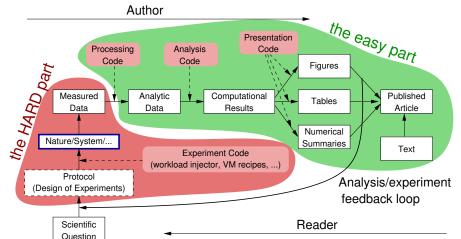
Distributed computing: a peculiar field in CS

- Performance and scalability are central to results
 - But depend greatly on the environment (hardware, network, software stack, etc.)
 - Many contributions are about fighting the environment
 - ★ Making the most out of limited resources
 - ★ Handling performance imbalance ~ load balancing
 - ★ Handling faults ~ fault tolerance
 - ★ Hiding complexity ~ abstractions: middlewares, runtimes
- Validation of most contributions require experiments
 - Very little formal validation
 - Even for more theoretical work → simulation (SimGrid, CloudSim)

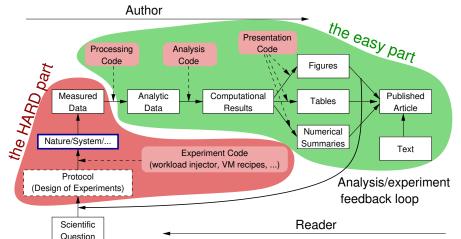
Distributed computing: a peculiar field in CS

- Performance and scalability are central to results
 - But depend greatly on the environment (hardware, network, software stack, etc.)
 - Many contributions are about fighting the environment
 - ★ Making the most out of limited resources
 - ★ Handling performance imbalance ~ load balancing
 - ★ Handling faults ~ fault tolerance
 - ★ Hiding complexity ~ abstractions: middlewares, runtimes
- Validation of most contributions require experiments
 - Very little formal validation
 - Even for more theoretical work → simulation (SimGrid, CloudSim)
- But experimenting is difficult and time-consuming, but often neglected
 - How could we perform better experiments?
 - Very similar to (not computational) biology or physics

What's an experiment: the research pipeline



What's an experiment: the research pipeline



Grid'5000 mission: a large-scale, shared testbed to support high-quality, reproducible experiments

The Grid'5000 testbed

- One of the world-leading testbeds for distributed computing
 - ♦ 8 sites, 30 clusters, 840 nodes, 8490 cores
 - Dedicated 10-Gbps backbone network
 - Various HPC networks and accelerators
 - 550 users and 100 publications per year



The Grid'5000 testbed

- One of the world-leading testbeds for distributed computing
 - 8 sites, 30 clusters, 840 nodes, 8490 cores
 - Dedicated 10-Gbps backbone network
 - Various HPC networks and accelerators
 - 550 users and 100 publications per year



- ► A meta-grid, meta-cloud, meta-cluster, meta-data-center:
 - Used by CS researchers in HPC / Clouds / Big Data / Networking
 - To experiment in a fully controllable and observable environment
 - Design goals:
 - Support high-quality, reproducible experiments
 - ★ On a large-scale, shared infrastructure

Landscape – cloud & experimentation

- ► Public cloud infrastructures (AWS, Azure, Google, etc.)
 - No information/guarantees on placement, multi-tenancy, real performance
- Private clouds: Shared observable infrastructures
 - © Monitoring & measurement
 - ② No control over infrastructure settings
 - → Ability to understand experiment results
- On-demand clouds dedicated observable infrastructures (BonFIRE)
 - © Limited ability to alter infrastructure
- ► Bare-metal as a service, fully reconfigurable infrastructure (Grid'5000)
 - © Control/alter all layers, including virtualization technology, operating system, networking

Outline

- Introduction
- Discovering resources from their description
- Reconfiguring the testbed to meet experimental needs
- Monitoring experiments, extracting and analyzing data

Discovering resources from their description

- ▶ Describing resources ~ understand results
 - Covering nodes, network equipment, topology
 - Machine-parsable format (JSON) → scripts
 - Archived (State of testbed 6 months ago?)

```
"processor": {
  "cache l2": 8388608,
  "cache l1": null,
  "model": "Intel Xeon".
  "instruction set": ""
  "other description": ""
  "version": "X3440",
  "vendor": "Intel".
  "cache lli": null,
  "cache lld": null.
  "clock speed": 2530000000.0
"uid": "graphene-1",
"type": "node".
"architecture": {
  "platform type": "x86 64".
  "smt size": 4,
  "smp size": 1
"main memory": {
  "ram size": 17179869184,
  "virtual size": null
"storage devices": [
    "model": "Hitachi HDS72103",
    "size": 298023223876.953.
    "driver": "ahci".
    "interface": "SATA II".
    "rev": "JPFO"
    "device": "sda"
```

Discovering resources from their description

- ▶ Describing resources ~ understand results
 - Covering nodes, network equipment, topology
 - Machine-parsable format (JSON) → scripts
 - ◆ Archived (State of testbed 6 months ago?)
- Verifying the description
 - Avoid inaccuracies/errors → wrong results
 - Could happen frequently: maintenance, broken hardware (e.g. RAM)
 - Our solution: g5k-checks
 - Runs at node boot (or manually by users)
 - Acquires info using OHAI, ethtool, etc.
 - Compares with Reference API

```
"processor": {
 "cache l2": 8388608,
  "cache l1": null,
  "model": "Intel Xeon".
  "instruction set": ""
  "other description": ""
  "version": "X3440",
  "vendor": "Intel".
  "cache lli": null,
 "cache lld": null.
 "clock speed": 2530000000.0
"uid": "graphene-1",
"type": "node".
"architecture": {
  "platform type": "x86 64".
 "smt size": 4,
 "smp size": 1
"main_memory": {
 "ram size": 17179869184,
  "virtual size": null
"storage devices": [
   "model": "Hitachi HDS72103",
    "size": 298023223876.953.
    "driver": "ahci".
    "interface": "SATA II",
   "rev": "JPFO"
    "device": "sda"
```

Discovering resources from their description

- ▶ Describing resources ~ understand results
 - Covering nodes, network equipment, topology
 - Machine-parsable format (JSON) → scripts
 - ◆ Archived (State of testbed 6 months ago?)
- Verifying the description
 - Avoid inaccuracies/errors → wrong results
 - Could happen frequently: maintenance, broken hardware (e.g. RAM)
 - Our solution: g5k-checks
 - ★ Runs at node boot (or manually by users)
 - ★ Acquires info using OHAI, ethtool, etc.
 - ★ Compares with Reference API
- Selecting resources
- ◆ OAR database filled from Reference API

 oarsub -p "wattmeter='YES' and gpu='YES'"

 oarsub -l "cluster='a'/nodes=1+cluster='b' and

 eth10g='Y'/nodes=2,walltime=2"

```
"processor": {
 "cache l2": 8388608,
  "cache l1": null,
  "model": "Intel Xeon".
  "instruction set": ""
  "other description": ""
  "version": "X3440",
  "vendor": "Intel".
  "cache lli": null,
 "cache lld": null.
 "clock speed": 2530000000.0
"uid": "graphene-1",
"type": "node".
"architecture": {
  "platform type": "x86 64".
 "smt size": 4,
 "smp size": 1
"main_memory": {
 "ram size": 17179869184,
 "virtual size": null
"storage devices": [
   "model": "Hitachi HDS72103",
    "size": 298023223876.953,
    "driver": "ahci".
   "interface": "SATA II".
   "rev": "JPFO"
    "device": "sda"
```

Outline

- Introduction
- Discovering resources from their description
- Reconfiguring the testbed to meet experimental needs
- Monitoring experiments, extracting and analyzing data

Reconfiguring the testbed

- Typical needs:
 - Install specific software
 - Modify the kernel
 - Run custom distributed middlewares (Cloud, HPC, Grid)
 - Keep a stable (over time) software environment

Reconfiguring the testbed

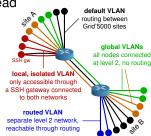
- ▶ Typical needs:
 - Install specific software
 - Modify the kernel
 - Run custom distributed middlewares (Cloud, HPC, Grid)
 - Keep a stable (over time) software environment
- Likely answer on any production facility: you can't
- Or:
 - Install in \$HOME, modules → no root access, handle custom paths
 - Use virtual machines → experimental bias (performance), limitations
 - Containers: kernel is shared → various limitations

Reconfiguring the testbed

- Operating System reconfiguration with Kadeploy:
 - Provides a Hardware-as-a-Service cloud infrastructure
 - Enable users to deploy their own software stack & get root access
 - Scalable, efficient, reliable and flexible:
 200 nodes deployed in ~5 minutes
- Customize networking environment with KaVLAN
 - Protect the testbed from experiments (Grid/Cloud middlewares)
 - Avoid network pollution

◆ By reconfiguring VLANS ~ almost no overhead





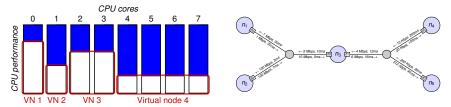
Creating and sharing Kadeploy images

- When doing manual customization:
 - Easy to forget some changes
 - Difficult to describe
 - The full image must be provided
 - Cannot really serve as a basis for future experiments (similar to binary vs source code)
- ► Kameleon: Reproducible generation of software appliances
 - Using recipes (high-level description)
 - ◆ Persistent cache to allow re-generation without external resources (Linux distribution mirror) ~> self-contained archive
 - Supports Kadeploy images, LXC, Docker, VirtualBox, qemu, etc.

http://kameleon.imag.fr/

Changing experimental conditions

- Reconfigure experimental conditions with Distem
 - Introduce heterogeneity in an homogeneous cluster
 - Emulate complex network topologies



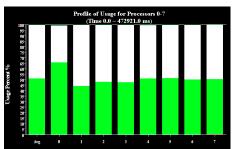
http://distem.gforge.inria.fr/



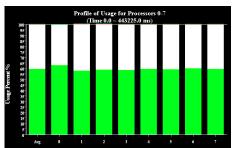


Testing Charm++ load balancing with Distem





RefineLB



total run time: 473s Average CPU usage: 51% total run time: 443s Average CPU usage: 59%

- Every 2 minutes, 1/8 of the nodes are downclocked for 2 minutes
- On the figure, node 0 has been downclocked
- Visible improvement thanks to load balancing

Ensuring consistent hardware configuration

- Many hardware performance settings can have a huge impact
 - BIOS and firmware versions
 - ★ Horror story: older disk firmware version on one node caused 10% performance drop
 - Disks read/write cache
 - CPU P-states, C-states, hyperthreading, turbo-boost
 - On Grid'5000: enabled by default, documentation on how to change them
 - NUMA settings: node interleaving, snoop mode
- Goal: uniform configuration inside clusters; some settings defined testbed-wide
- Regression tests to ensure this

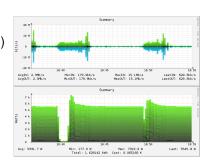
Outline

- Introduction
- Discovering resources from their description
- Reconfiguring the testbed to meet experimental needs
- Monitoring experiments, extracting and analyzing data

Monitoring experiments

Goal: enable users to understand what happens during their experiment

- System-level probes (usage of CPU, memory, disk, with Ganglia)
- Infrastructure-level probes
 - Network, power consumption
 - Captured at high frequency (≈1 Hz)
 - Live visualization
 - ♦ REST API
 - Long-term storage



Conclusions

- Grid'5000: a testbed for high-quality, reproducible research on HPC, Clouds, Big Data and Networking
- With a unique combination of features
 - Description and verification of testbed
 - Reconfiguration (hardware, network)
 - Monitoring
 - Support for automation of experiments
- Try it yourself!
 - Free account through the Open Access program http://www.grid5000.fr/open-access

More: https://www.grid5000.fr

Bibliography

- Resources management: Resources Description, Selection, Reservation and Verification on a Large-scale Testbed. http://hal.inria.fr/hal-00965708
- Kadeploy: Kadeploy3: Efficient and Scalable Operating System Provisioning for Clusters. http://hal.inria.fr/hal-00909111
- ► KaVLAN, Virtualization, Clouds deployment:
 - Adding Virtualization Capabilities to the Grid'5000 testbed. http://hal.inria.fr/hal-00946971
 - Enabling Large-Scale Testing of laaS Cloud Platforms on the Grid'5000 Testbed. http://hal.inria.fr/hal-00907888
- Kameleon: Reproducible Software Appliances for Experimentation. https://hal.inria.fr/hal-01064825
- Distem: Design and Evaluation of a Virtual Experimental Environment for Distributed Systems. https://hal.inria.fr/hal-00724308
- XP management tools:
 - ♦ A survey of general-purpose experiment management tools for distributed systems. https://hal.inria.fr/hal-01087519
 - XPFlow: A workflow-inspired, modular and robust approach to experiments in distributed systems. https://hal.inria.fr/hal-00909347
 - Using the EXECO toolbox to perform automatic and reproducible cloud experiments. https://hal.inria.fr/hal-00861886
 - Expo: Managing Large Scale Experiments in Distributed Testbeds. https://hal.inria.fr/hal-00953123
- Kwapi: A Unified Monitoring Framework for Energy Consumption and Network Traffic. https://hal.inria.fr/hal-01167915
- Realis'2014: Reproductibilité expérimentale pour l'informatique en parallélisme, architecture et système. https://hal.inria.fr/hal-01011401