(IaaS) Cloud Benchmarking:
Approaches, Challenges, and Experience

HPDC 2013

Alexandru Iosup

Parallel and Distributed Systems Group
Delft University of Technology
The Netherlands

Our team: **Undergrad** Nassos Antoniou, Thomas de Ruiter, Ruben Verboon, … **Grad** Siqi Shen, Nezih Yigitbasi, Ozan Sonmez **Staff** Henk Sips, Dick Epema, Alexandru Iosup **Collaborators** Ion Stoica and the Mesos team (UC Berkeley), Thomas Fahringer, Radu Prodan (U. Innsbruck), Nicolae Tapus, Mihaela Balint, Vlad Posea (UPB), Derrick Kondo, Emmanuel Jeannot (INRIA), Assaf Schuster, Orna Ben-Yehuda (Technion), Ted Willke (Intel), Claudio Martella (Giraph), …

Lectures at the Technion Computer Engineering Center (TCE), Haifa, IL

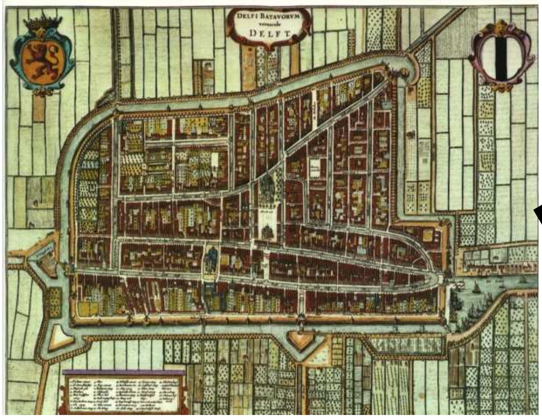
IaaS Cloud Benchmarking	May 7
Massivizing Online Social Games	May 9
Gamification in Higher Education	May 27
Lectures at IBM Haifa, Intel Haifa	June 2,3
Scheduling in IaaS Clouds	Actually, HUJI June 5

A TU Delft perspective on Big Data Processing and Preservation	**June 6**	**10am Taub 337**

Grateful to Orna Agmon Ben-Yehuda, Assaf Schuster, Isaac Keslassy.

Also thankful to Bella Rotman and Ruth Boneh.

(TU) Delft – the Netherlands – Europe



founded 13th century
pop: 100,000

founded 1842
pop: 13,000

pop: 16.5 M

(We are here) אנחנו כאן

The Parallel and Distributed Systems Group at TU Delft

VENI

Alexandru Iosup

Grids/Clouds
P2P systems
Big Data
Online gaming

Dick Epema

Grids/Clouds
P2P systems
Video-on-demand
e-Science

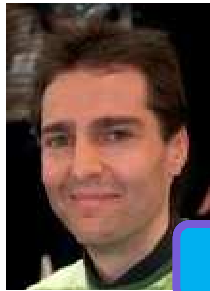
VENI

Ana Lucia Varbanescu

HPC systems
Multi-cores
Big Data
e-Science

Henk Sips

HPC systems
Multi-cores
P2P systems



VENI

Johan Pouwelse

P2P systems
File-sharing
Video-on-demand

Home page

- www.pds.ewi.tudelft.nl

Publications

- see PDS publication database at publications.st.ewi.tudelft.nl

Not This Presentation, but Relevant

PDS Work on OpenCL vs OpenMP

CPU-unfriendly programming

Porting from CUDA leaves marks

Memory access row-v-column, Local mem, mem copy

Granularity and tiling

Fine-grained can lead to poor cache locality on CPU

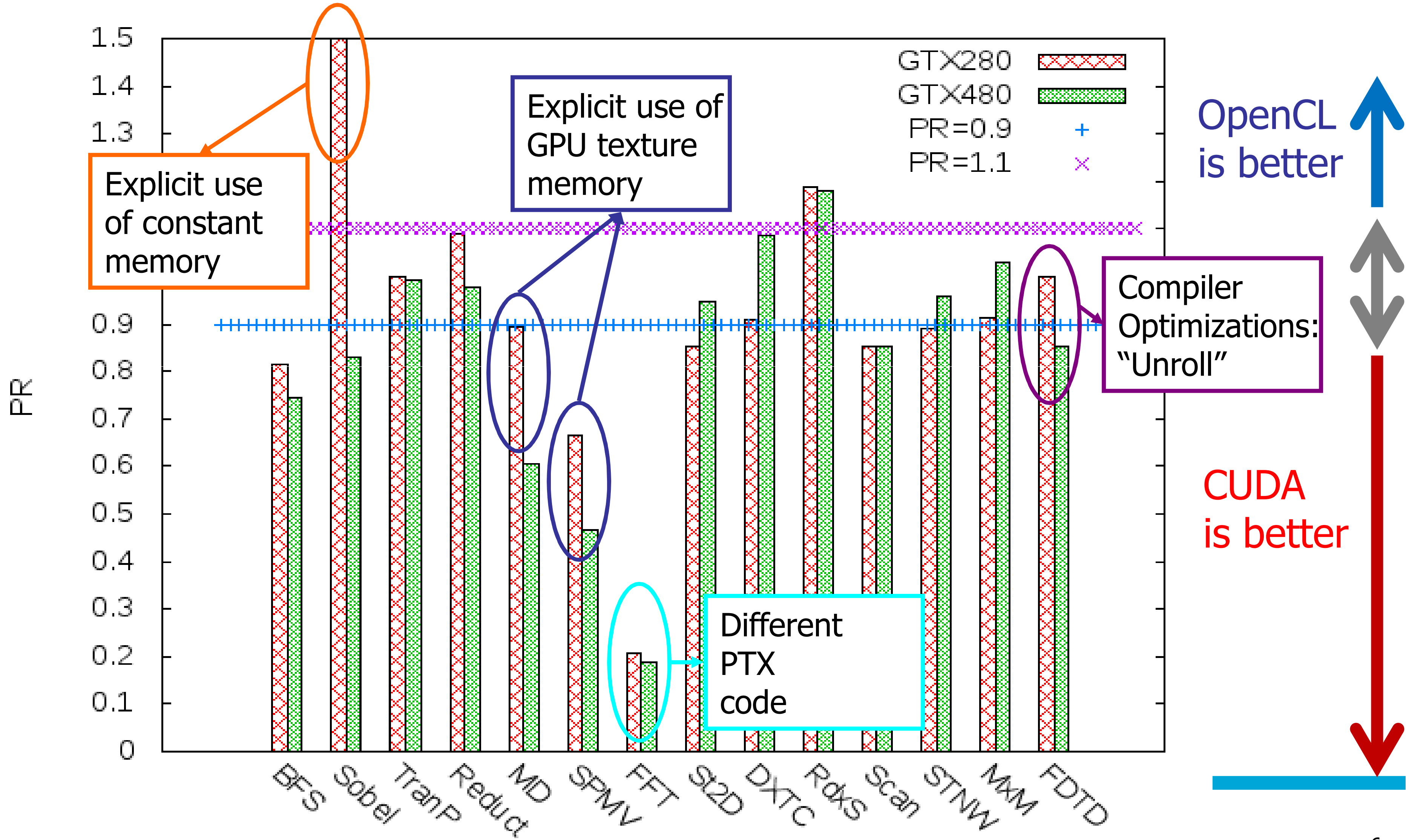
OpenCL compilers need maturing

AMD 2.5 vs Intel 1.1 compilers very different in implicit vectorization, default optimizations, etc.

Shen et al. Performance Gaps between OpenMP and OpenCL for Multi-core CPUs ICPP2012W

Not This Presentation, but Relevant

PDS Work on OpenCL vs CUDA

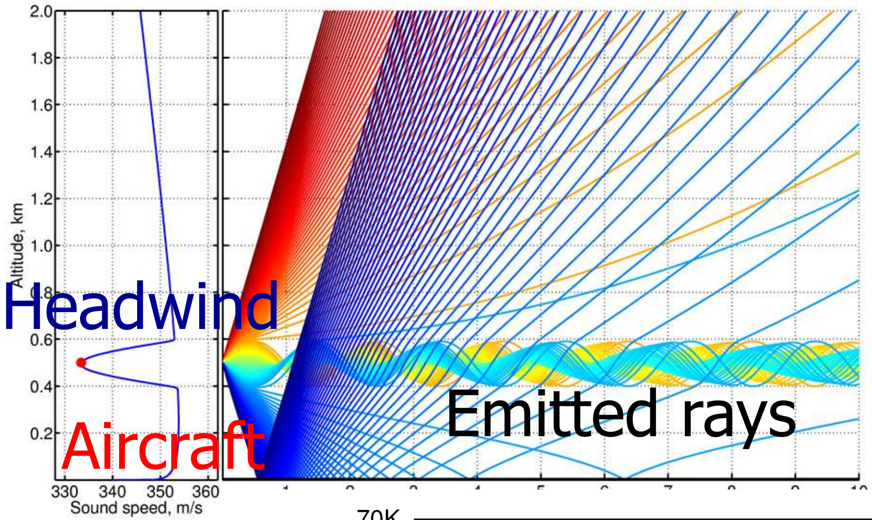


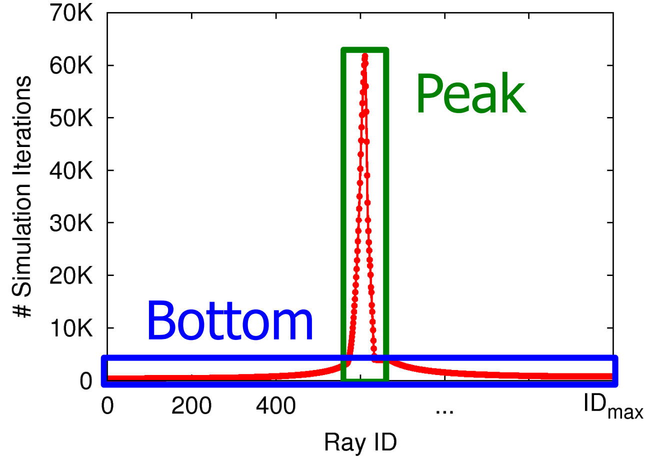
Fang et al. A Comprehensive Performance Comparison of CUDA and OpenCL, ICPP'11

Not This Presentation, but Relevant

Imbalanced Workloads on Fused Archi

- Acoustic ray-tracing
- Fused architecture
  - Task + Data parallelism
  - Divide the whole workloads into
    - A bottom part (on the GPU)
    - A peak part (on the multi-core CPU)
  - multi-core CPU(s) **and** GPU(s)



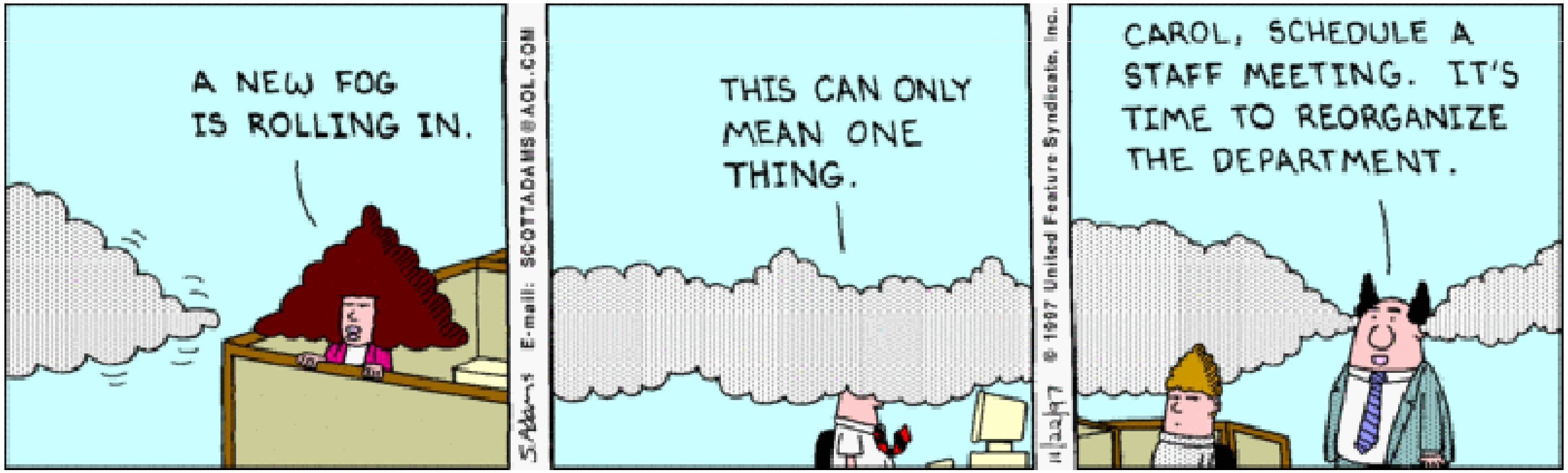


- Experimental results
  - **10x better performance than traditional**
  - Auto-tuned soft real-time approaching hard real-time: ~30 ms

Shen et al. . Glinda: A Framework for Accelerating Imbalanced Applications on Heterogeneous Platforms. CF'13.

What is Cloud Computing?
1. A Cloudy Buzzword

- 18 definitions in computer science (ECIS'10). NIST has one. Cal has one. We have one.
- "We have redefined cloud computing to include everything that we already do." Larry Ellison, Oracle, 2009



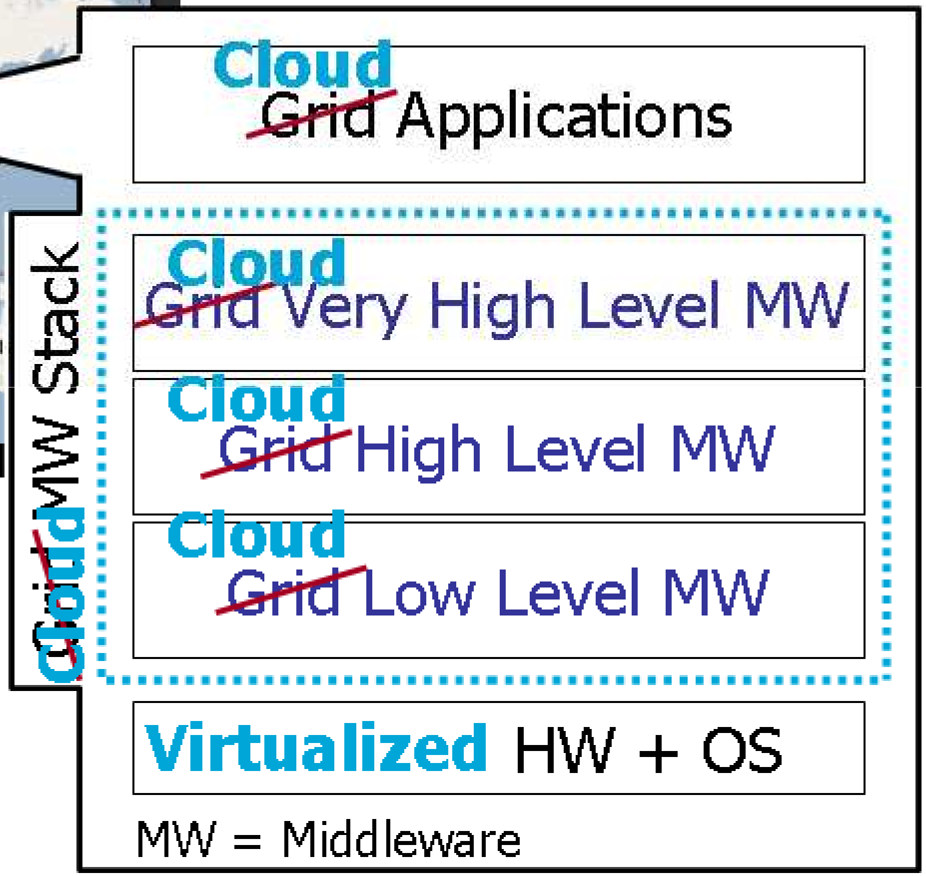
Source: http://dilbert.com/strips/comic/1997-11-22/

What is Cloud Computing?
2. A Descendant* of the Grid Idea

Source: http://royal.pingdom.com/2008/04/11/map-of-all-google-data-center-locations/

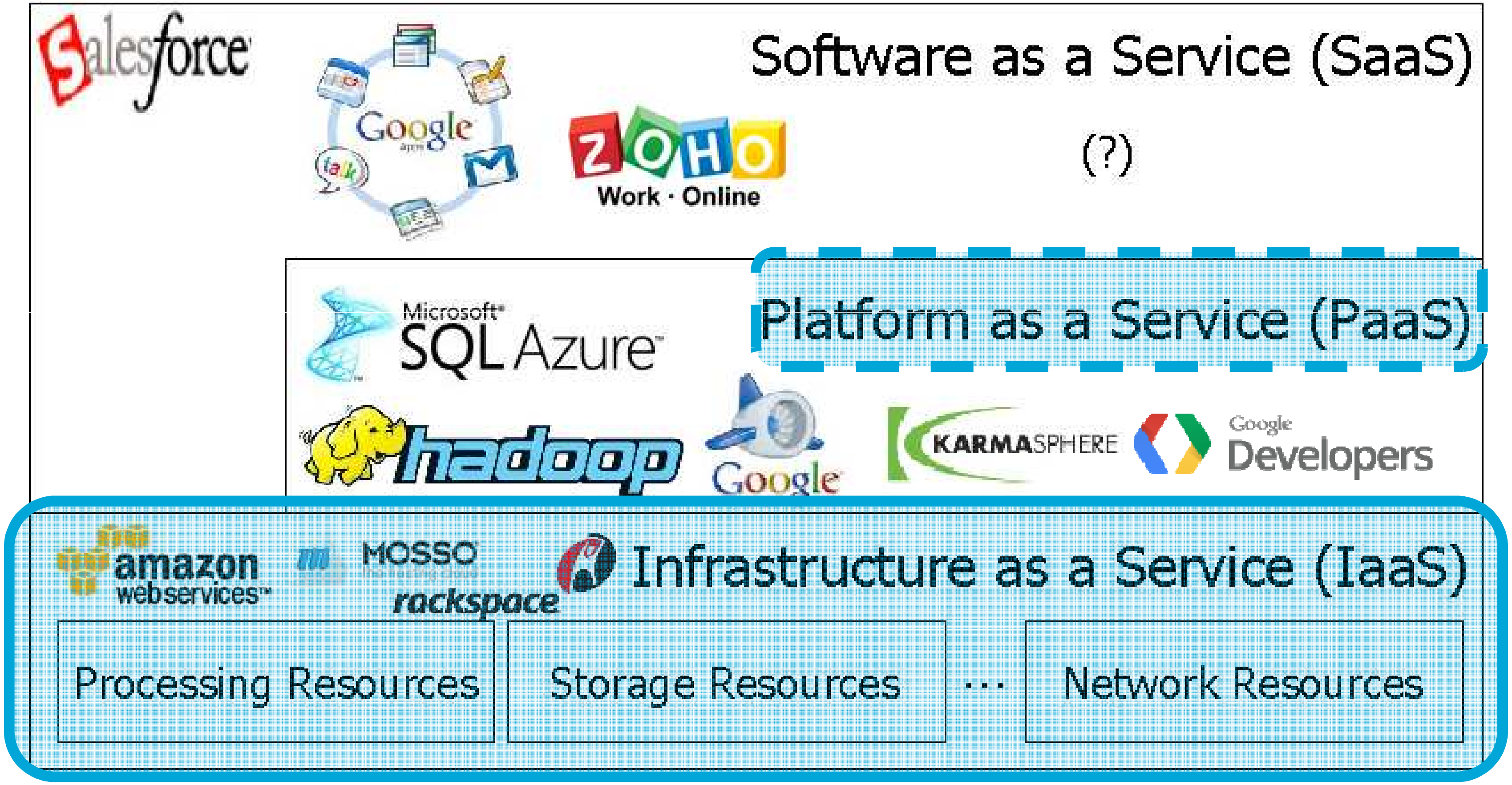
"A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities [+ for] nontrivial QoS." I. Foster, 1998 + 1999

* Subset.

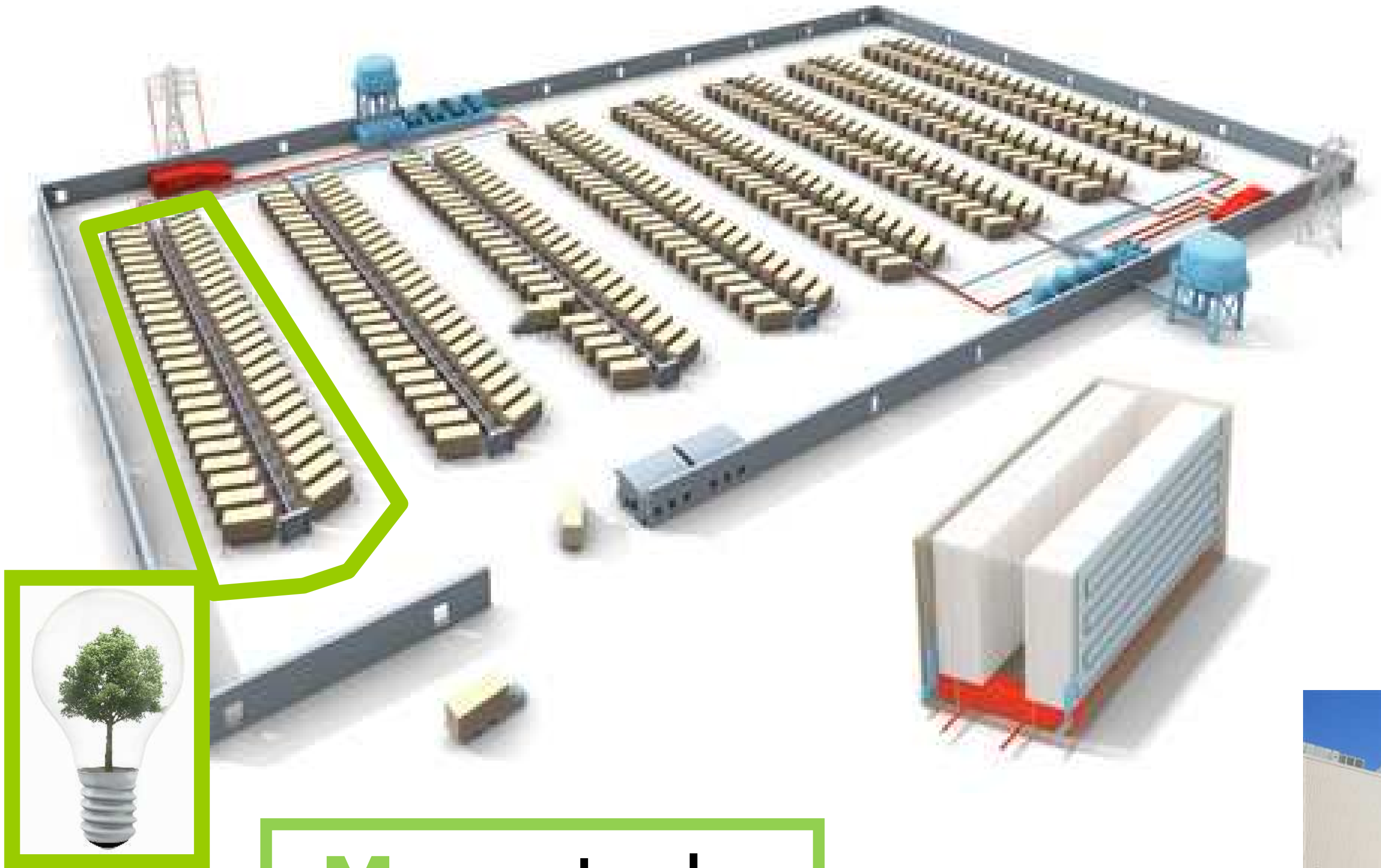


What is Cloud Computing?
3. A Useful IT Service

"Use only when you want! Pay only for what you use!"



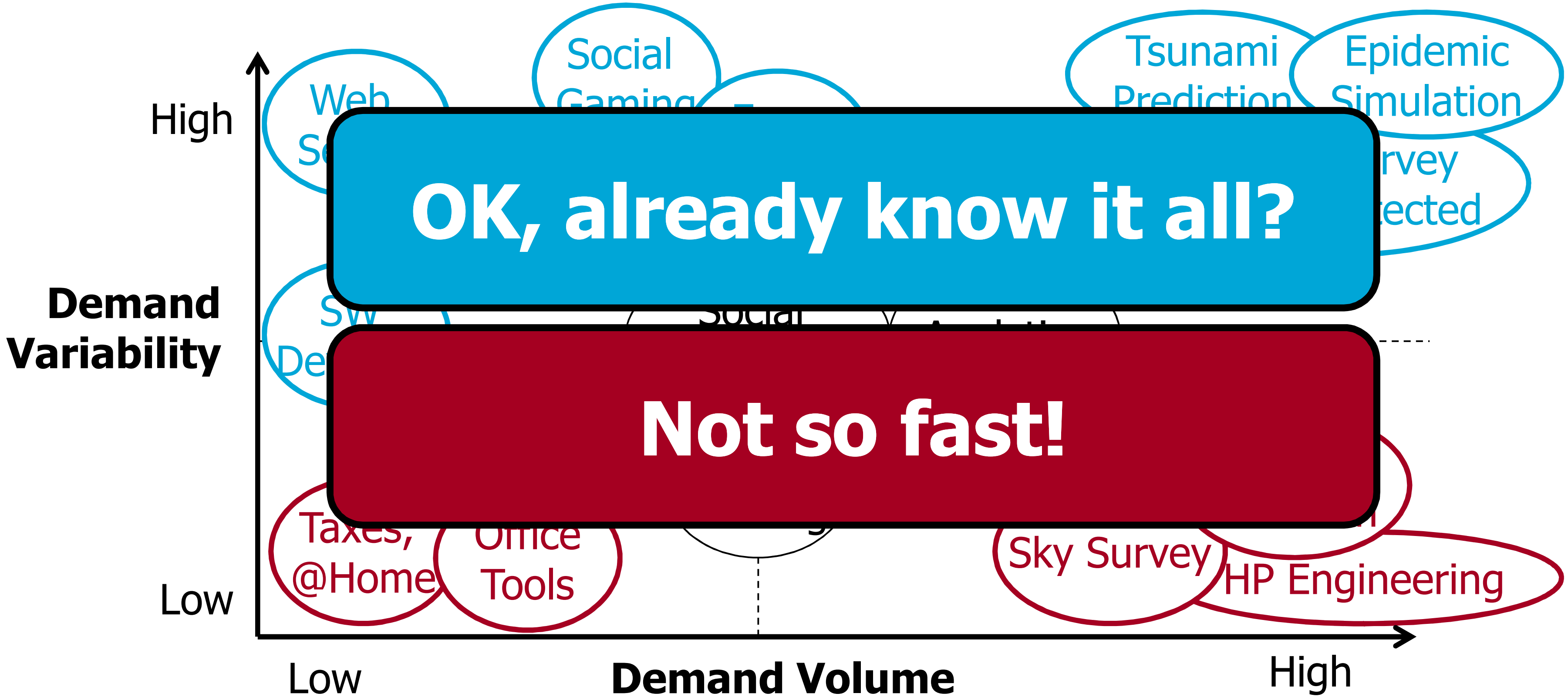
IaaS Cloud Computing



Many tasks

UptimeInstitute
GREEN ENTERPRISE IT AWARD 2010

Which Applications Need Cloud Computing? A Simplistic View…



OK, already know it all?

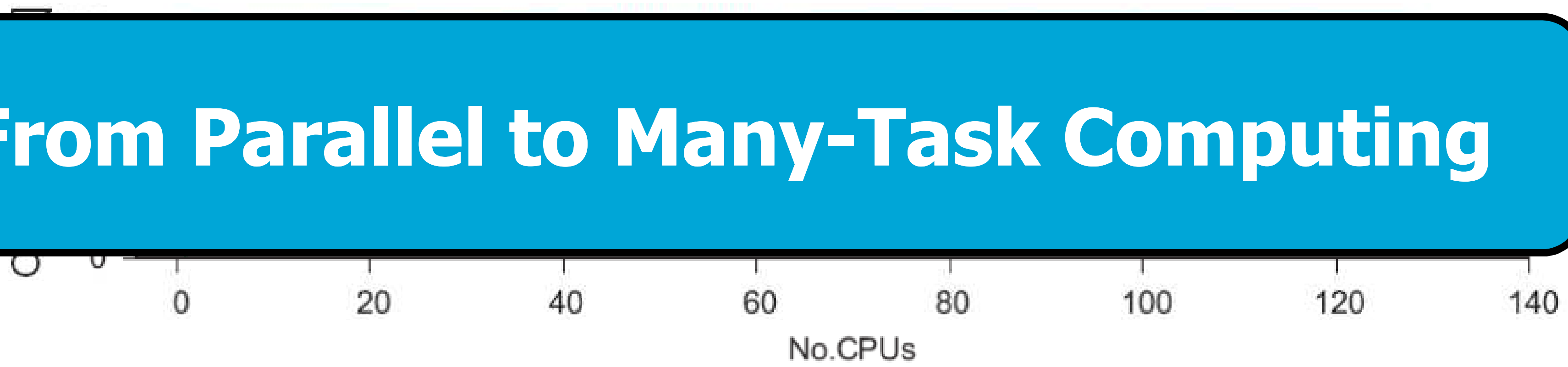
Not so fast!

After an idea by Helmut Krcmar

What I Learned From Grids

* The past

- Average job size is 1 (that is, there are **no** **[!]** **tightly-coupled, only conveniently parallel jobs**)



From Parallel to Many-Task Computing

A. Iosup, C. Dumitrescu, D.H.J. Epema, H. Li, L. Wolters, How are Real Grids Used? The Analysis of Four Grid Traces and Its Implications, Grid 2006.
A. Iosup and D.H.J. Epema, Grid Computing Workloads, IEEE Internet Computing 15(2): 19-26 (2011)

What I Learned From Grids

* The past

- NMI Build-and-Test Environment at U.Wisc.-Madison:
  112 hosts, >40 platforms (e.g., X86-32/Solaris/5, X86-64/RH/9)
- Serves >50 **grid middleware packages**: Condor, Globus, VDT, gLite, GridFTP, RLS, NWS, INCA(-2), APST, NINF-G, BOINC …

Two years of functionality tests ('04-'06): over 1:3 runs have at least one failure!

(1) Test or perish!
(2) For grids, reliability is more important than performance!

A. Iosup, D.H.J.Epema, P. Couvares, A. Karp, M. Livny, Build-and-Test Workloads for Grid Middleware: Problem, Analysis, and Applications, CCGrid, 2007.

What I Learned From Grids

* The past

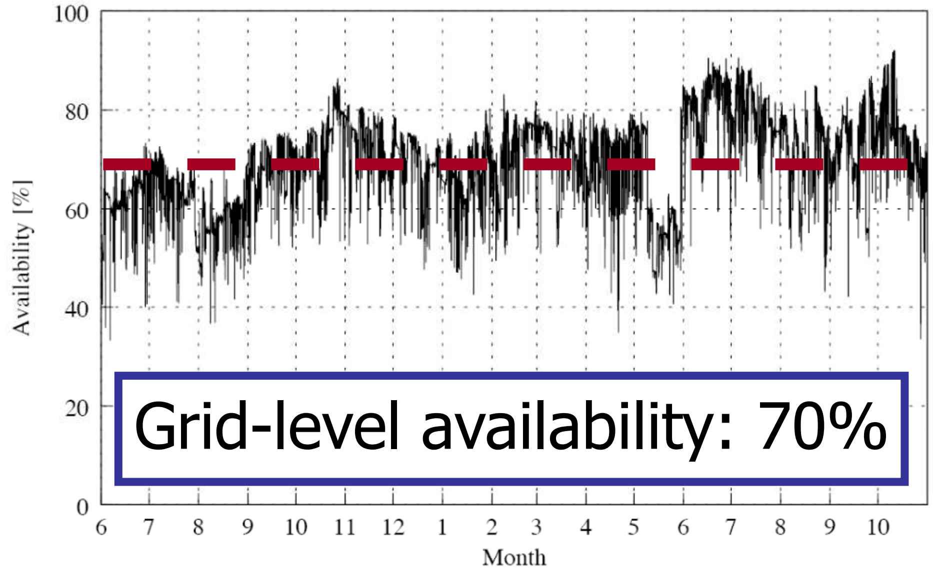
- Server
  - 99.99999% reliable

Grids are unreliable infrastructure

BNL-LCG2	0 vs 1 (0.00%)
INFN-T1	19066 vs 6042 (75.94%)
NIKHEF-ELPROD	5994 vs 22270 (21.21%)
RAL-LCG2	21631 vs 22391 (49.14%)
Taiwan-LCG2	18254 vs 9246 (66.38%)
USCMS-FNAL-WC1	101542 vs 8623 (92.17%)
pic	12851 vs 6627 (65.98%)
TOTAL	495281 vs 167668 (74.71%)

CERN LCG jobs
74.71% successful
25.29% unsuccessful

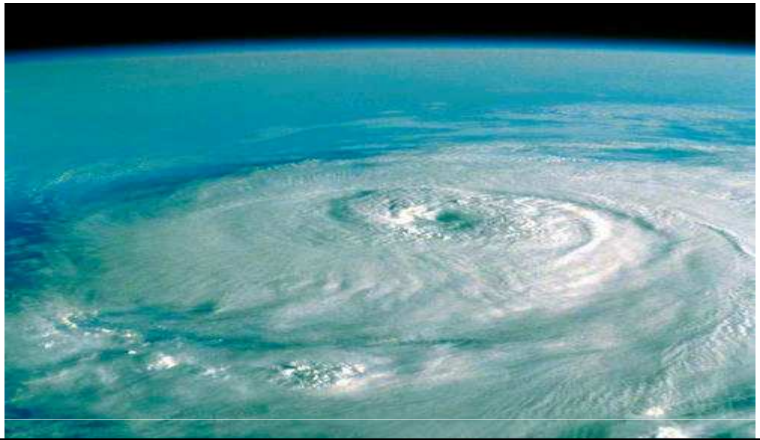
Source: dboard-gr.cern.ch, May'07.



A. Iosup, M. Jan, O. Sonmez, and D.H.J. Epema, On the Dynamic Resource Availability in Grids, Grid 2007, Sep 2007.

What I Learned From Grids, Applied to IaaS Clouds

or



We just don't know!

- "The path to abundance"
- On-demand capacity
- Cheap for short-term tasks
- Great for web apps (EIP, web crawl, DB ops, I/O)

- "The killer cyclone"
- Performance for scientific applications (compute- or data-intensive)
- Failures, Many-tasks, etc.

This Presentation: Research Questions

Q0: What are the workloads of IaaS clouds?

Q1: What is the performance of production IaaS cloud services?

Q2: How variable is the performance of widely used production cloud services?

Q3: How do provisioning and allocation policies affect the performance of IaaS cloud services?

Q4: What is the performance of production graph-processing platforms? (ongoing)

Ot… mance
of…

But … this is Benchmarking = process of quantifying the performance and other non-functional properties of the system

TU Delft University…

Why IaaS Cloud Benchmarking?

- Establish and share best-practices in answering important questions about IaaS clouds

- Use in procurement
- Use in system design
- Use in system tuning and operation
- Use in performance management

- **Use in training**

SPEC Research Group (RG)

The Research Group of the Standard Performance Evaluation Corporation

* The present

Mission Statement

- Provide a **platform for** collaborative research efforts in the areas of computer benchmarking and quantitative system analysis
- Provide metrics, tools and benchmarks for evaluating early prototypes and research results as well as full-blown implementations
- Foster interactions and collaborations btw. industry and academia

TU Delft
Delft University of Technology

Find more information on: *http://research.spec.org*

Current Members (Dec 2012)

* The present

SPEC RG Cloud Working Group is looking for new members!

http://research.spec.org/working-groups/rg-cloud-working-group.html

Find more information on: *http://research.spec.org*

PEDCA, a New FP7-REGIONS Project

- Create a Pan-European Data Center Alliance
  - Higher-education target
  - Industry target
  - Looking for active European countries

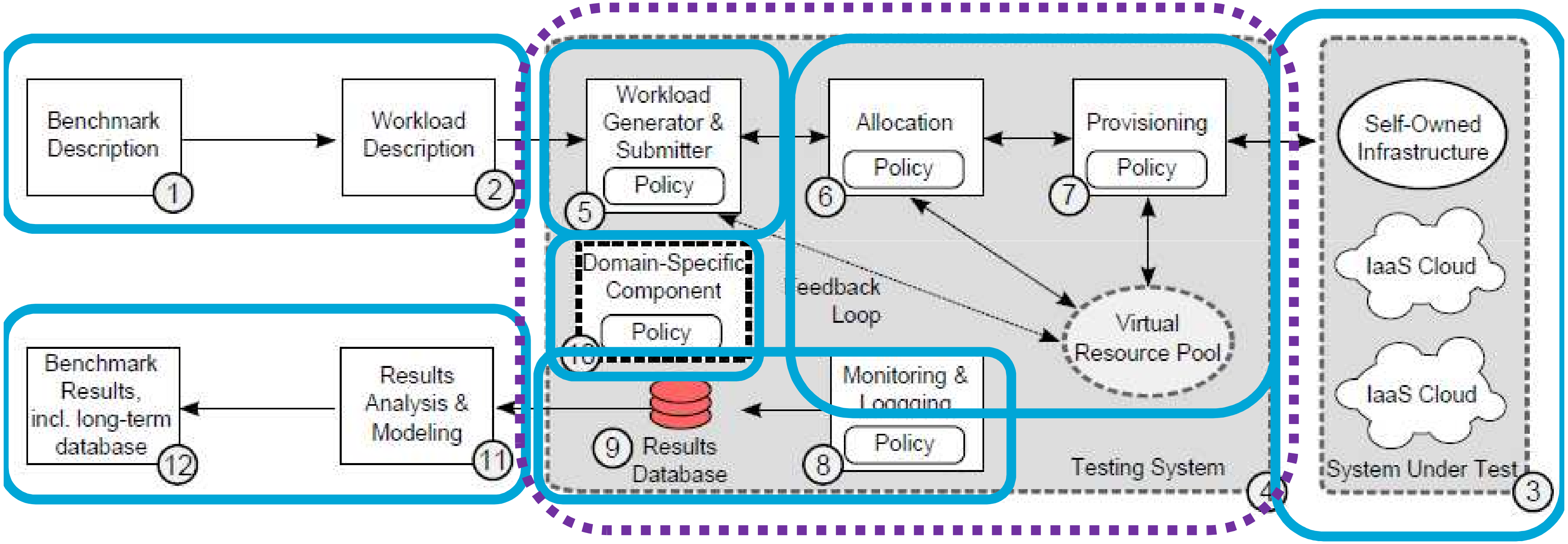
- EU FP7-REGIONS Project
  - 2M EUR
  - 18 months, starting July 2013
  - Transnational cooperation between regional research-driven clusters: DE, NL, UK (lead)

Agenda

1. An Introduction to IaaS Cloud Computing
2. Research Questions or Why We Need Benchmarking?
3. **A General Approach and Its Main Challenges**
4. IaaS Cloud Workloads (Q0)
5. IaaS Cloud Performance (Q1) and Perf. Variability (Q2)
6. Provisioning and Allocation Policies for IaaS Clouds (Q3)
7. Big Data: Large-Scale Graph Processing (Q4)
8. Conclusion

A General Approach for IaaS Cloud Benchmarking

* The present



Approach: Real Traces, Models, and Tools + Real-World Experimentation (+ Simulation)

* The present

- Formalize real-world scenarios
- Exchange real traces
- Model relevant operational elements
- Develop calable tools for meaningful and repeatable experiments
- Conduct comparative studies
  - Simulation only when needed (long-term scenarios, etc.)

Rule of thumb:

Put 10-15% project effort into benchmarking

10 Main Challenges in 4 Categories*

* The future

* List not exhaustive

- **Methodological**
  1. Experiment compression
  2. Beyond black-box testing through testing short-term dynamics and long-term evolution
  3. Impact of middleware

- **Workload-related**
  1. Statistical workload models
  2. Benchmarking performance isolation under various multi-tenancy workloads

- **System-Related**
  1. Reliability, availability, and system-related properties
  2. Massive-scale, multi-site benchmarking
  3. Performance isolation, multi-tenancy models

- **Metric-Related**
  1. Beyond traditional performance: variability, elasticity, etc.
  2. Closer integration with cost models

Read our article

Iosup, Prodan, and Epema, IaaS Cloud Benchmarking: Approaches, Challenges, and Experience, MTAGS 2012. (invited paper)

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

IaaS Cloud Workloads: Our Team

Alexandru Iosup
TU Delft

BoTs
Workflows
Big Data
Statistical modeling

Dick Epema
TU Delft

BoTs
Grids

Mathieu Jan
TU Delft/INRIA

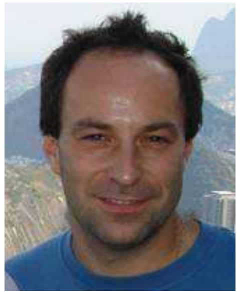
BoTs
Statistical modeling

Ozan Sonmez
TU Delft

BoTs

Thomas de Ruiter
TU Delft

MapReduce
Big Data
Statistical modeling



Radu Prodan
U.Isbk.

Workflows

Thomas Fahringer
U.Isbk.

Workflows

Simon Ostermann
U.Isbk.

Workflows

What I'll Talk About

IaaS Cloud Workloads (Q0)

1. BoTs
2. Workflows
3. Big Data Programming Models
4. MapReduce workloads

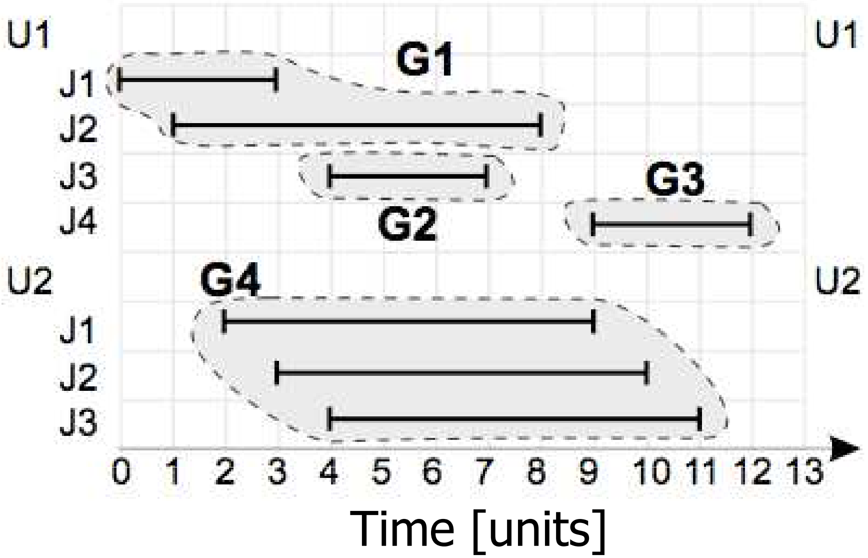
What is a Bag of Tasks (BoT)? A System View

BoT = set of jobs sent by a user…

$$W_u = \{J_i | user(J_i) = u\}$$

…that is submitted at most Δs after the first job

$$ST(J') \leq ST(J) + \Delta$$



- Why Bag of *Tasks*? From the perspective of the user, jobs in set are just tasks of a larger job
- A single useful result from the complete BoT
- Result can be combination of all tasks, or a selection of the results of most or even a single task

Iosup et al., The Characteristics and Performance of Groups of Jobs in Grids, Euro-Par, LNCS, vol.4641, pp. 382-393, 2007.

Q0

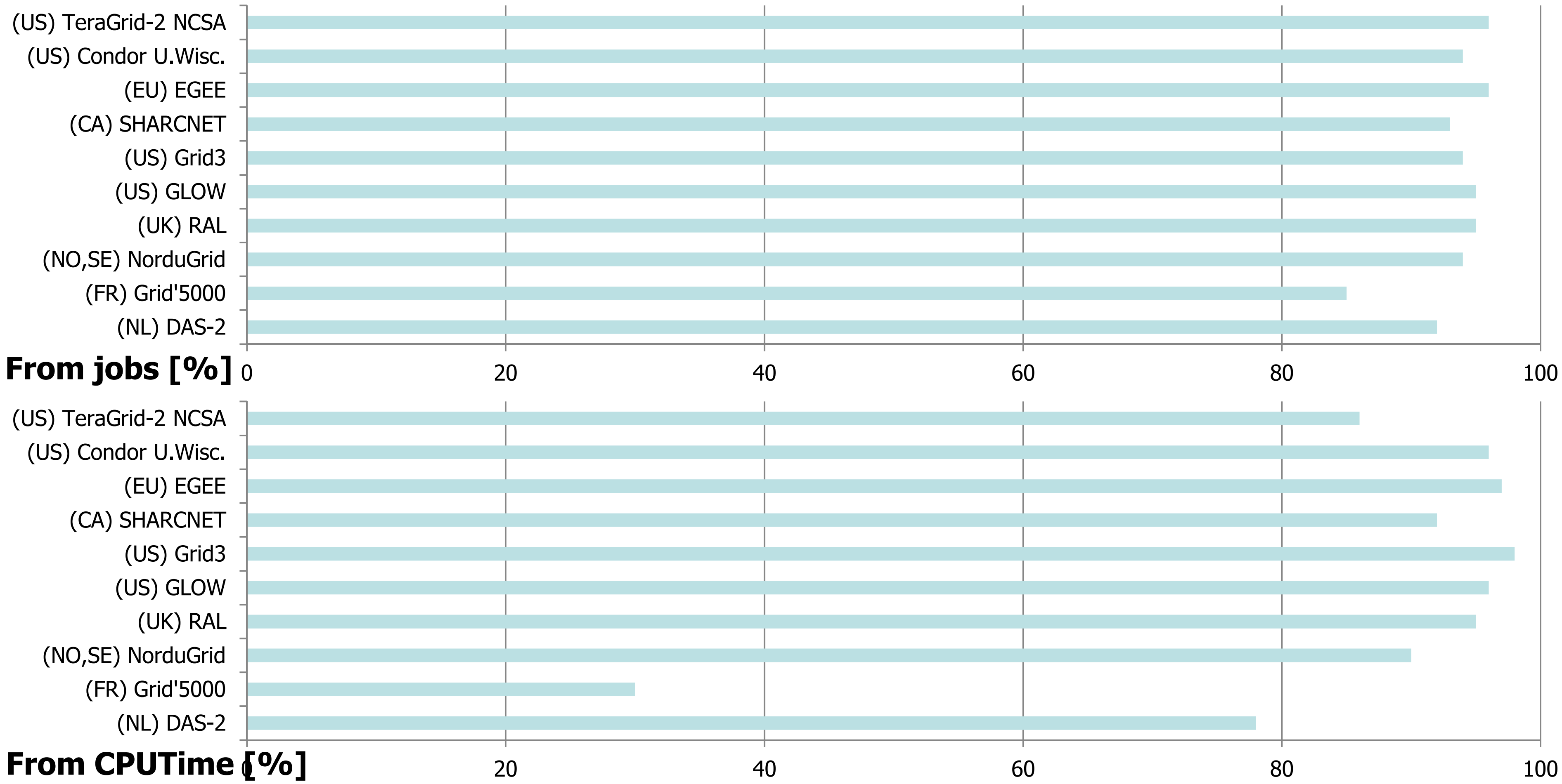
Applications of the BoT Programming Model

- Parameter sweeps
  - Comprehensive, possibly exhaustive investigation of a model
  - Very useful in engineering and simulation-based science

- Monte Carlo simulations
  - Simulation with random elements: fixed time yet limited inaccuracy
  - Very useful in engineering and simulation-based science

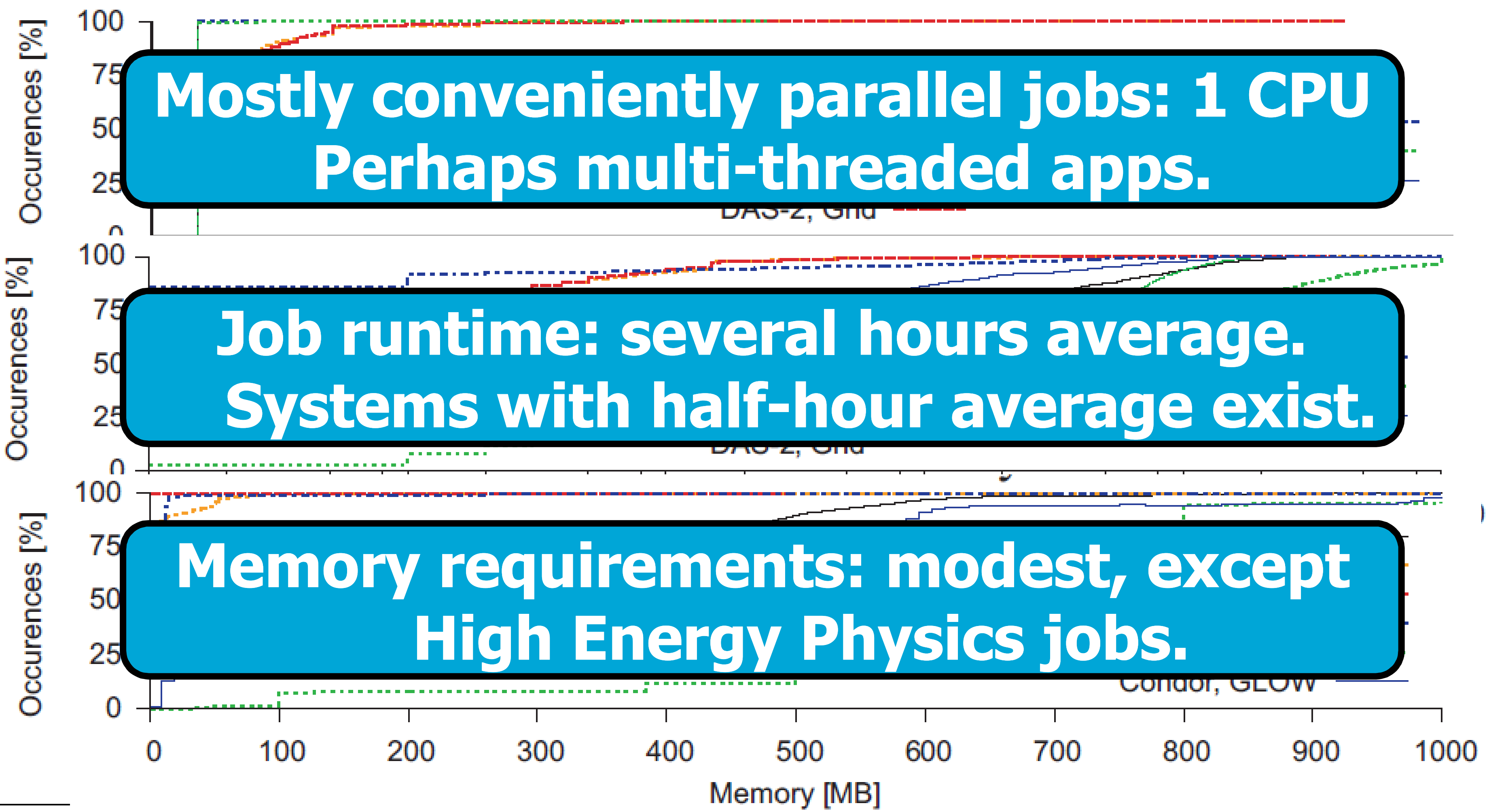
- Many other types of batch processing
  - Periodic computation, Cycle scavenging
  - Very useful to automate operations and reduce waste

BoTs Are the Dominant Programming Model for Grid Computing (Many Tasks)



Iosup and Epema: Grid Computing Workloads.
IEEE Internet Computing 15(2): 19-26 (2011)

BoTs by Numbers: CPUs, Runtime, Mem



Mostly conveniently parallel jobs: 1 CPU
Perhaps multi-threaded apps.

Job runtime: several hours average.
Systems with half-hour average exist.

Memory requirements: modest, except
High Energy Physics jobs.

Iosup et al., The Grid Workloads Archive, FGCS, 2008.

Iosup and Epema, Grid Computing Workloads, IEEE Internet Computing, 2011.

Actual numbers.

BoTs by numbers: I/O, Files, Remote Sys

T-12 part	I/O [KOps] Total	Rd	Wr	Wr %	I/O Traffic [MB] Total	Rd	Wr %
[illegible]	[illegible]	[illegible]	[illegible]	20%	469	174	63%
[illegible]	[illegible]	[illegible]	[illegible]	20%	144	114	21%
[illegible]	[illegible]	[illegible]	[illegible]	3%	161	130	19%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	389	33	92%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	330	31	91%

I/O: modest, except HEP

Rd:Wr varies widely

I/O,HEP: 65MBps/experiment

Upper bound for typical sci.apps.

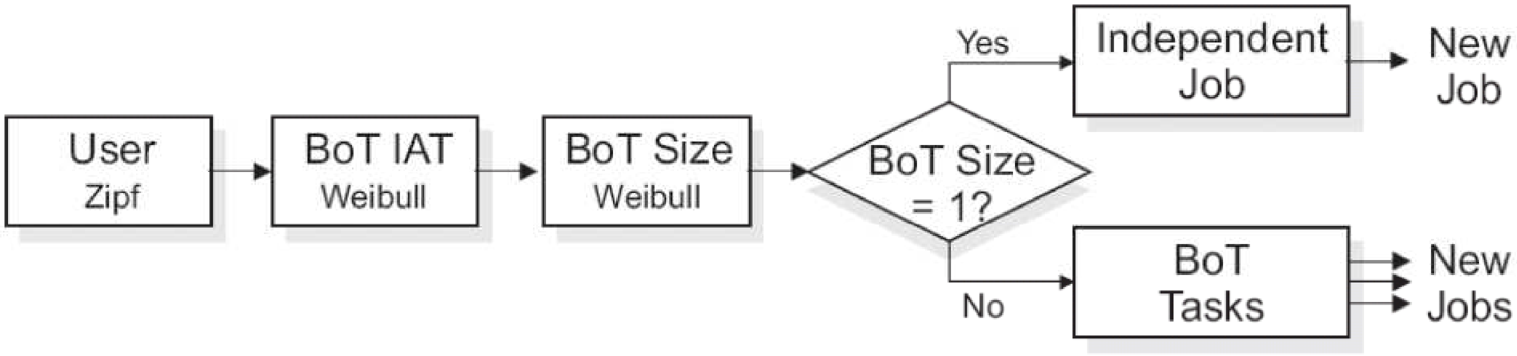
T-12 part	File Transfer [MB] Total	In	In / Out %		Remote Sys. Calls [MB] Total	In	In / Out %	
t1	10,865	8,259	76%	24%	28	16	59%	41%
t2	1,736	1,542	89%	11%	71	28	40%	60%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	58%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	44	40	91%	9%

Remote Sys.: small Xfers, latency important

Netw: 2-10GB, input mostly

Iosup and Epema, Grid Computing Workloads, IEEE Internet Computing, 2011.

BoT Workload Model

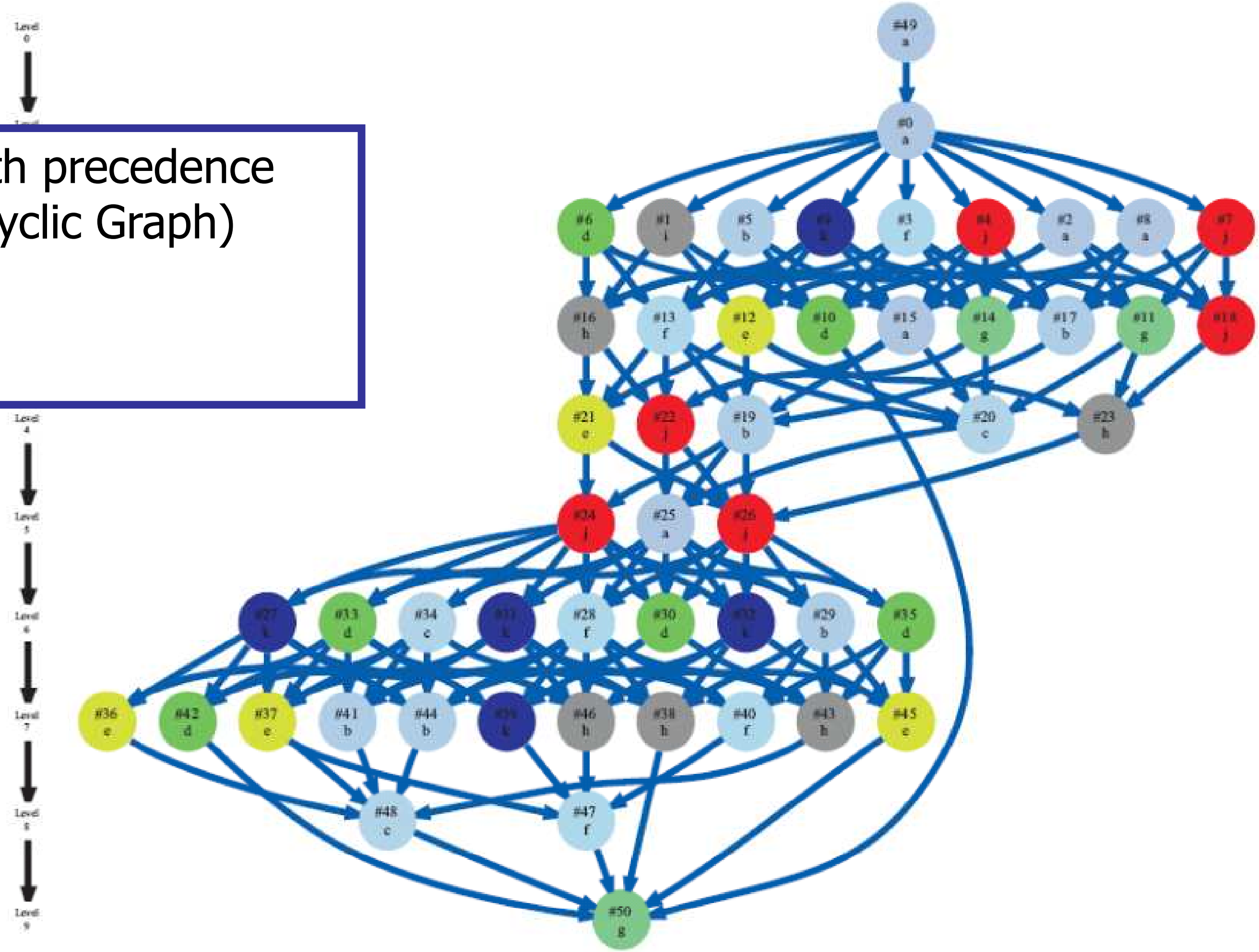


- Single arrival process for both BoTs and parallel jobs
- Validated with 7 grid workloads

A. Iosup, O. Sonmez, S. Anoep, and D.H.J. Epema. The Performance of Bags-of-Tasks in Large-Scale Distributed Systems, HPDC, pp. 97-108, 2008.

What is a Wokflow?

WF = set of jobs with precedence
(think Direct Acyclic Graph)



Applications of the Workflow Programming Model

- Complex applications
  - Complex filtering of data
  - Complex analysis of instrument measurements

- Applications created by non-CS scientists*
  - Workflows have a natural correspondence in the real-world, as descriptions of a scientific procedure
  - Visual model of a graph sometimes easier to program

- Precursor of the MapReduce Programming Model (next slides)

*Adapted from: Carole Goble and David de Roure, Chapter in "The Fourth Paradigm", http://research.microsoft.com/en-us/collaboration/fourthparadigm/

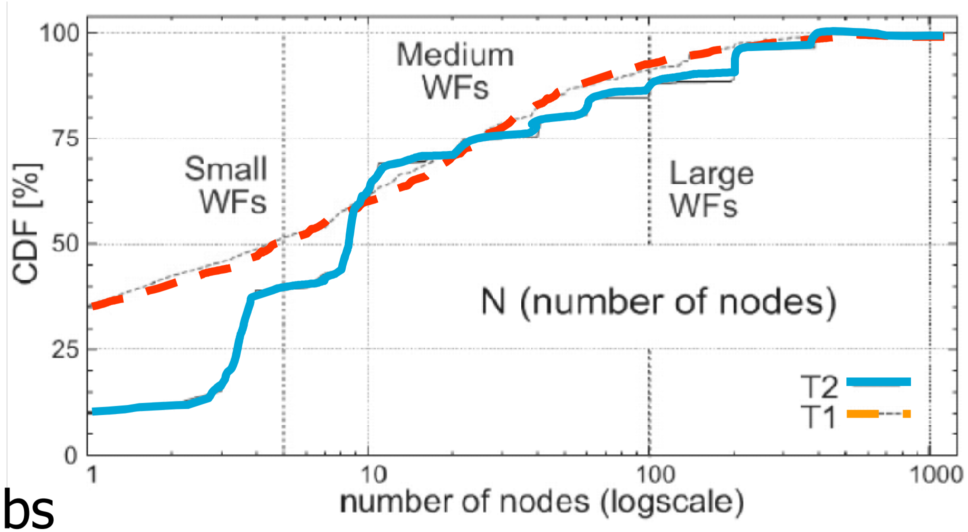
Q0

Workflows Exist in Grids, but Did No Evidence of a Dominant Programming Model

- Traces

Trace	Source	Duration	Number of WFs	Number of Tasks	CPUdays
T1	DEE	09/06-10/07	4,113	122k	152
T2	EE2	05/07-11/07	1,030	46k	41

- Selected Findings

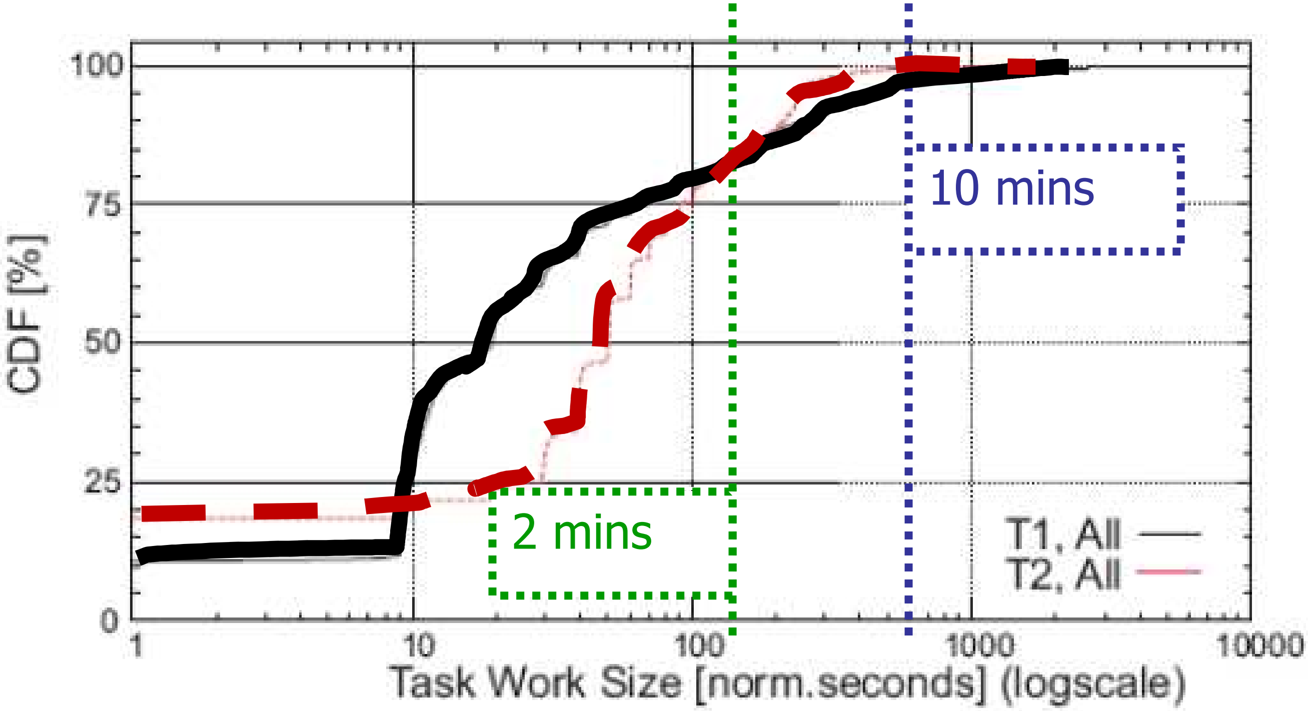


  - Loose coupling
  - Graph with 3-4 levels
  - Average WF size is 30/44 jobs
  - 75%+ WFs are sized 40 jobs or less, 95% are sized 200 jobs or less

Ostermann et al., On the Characteristics of Grid Workflows, CoreGRID Integrated Research in Grid Computing (CGIW), 2008.

Q0

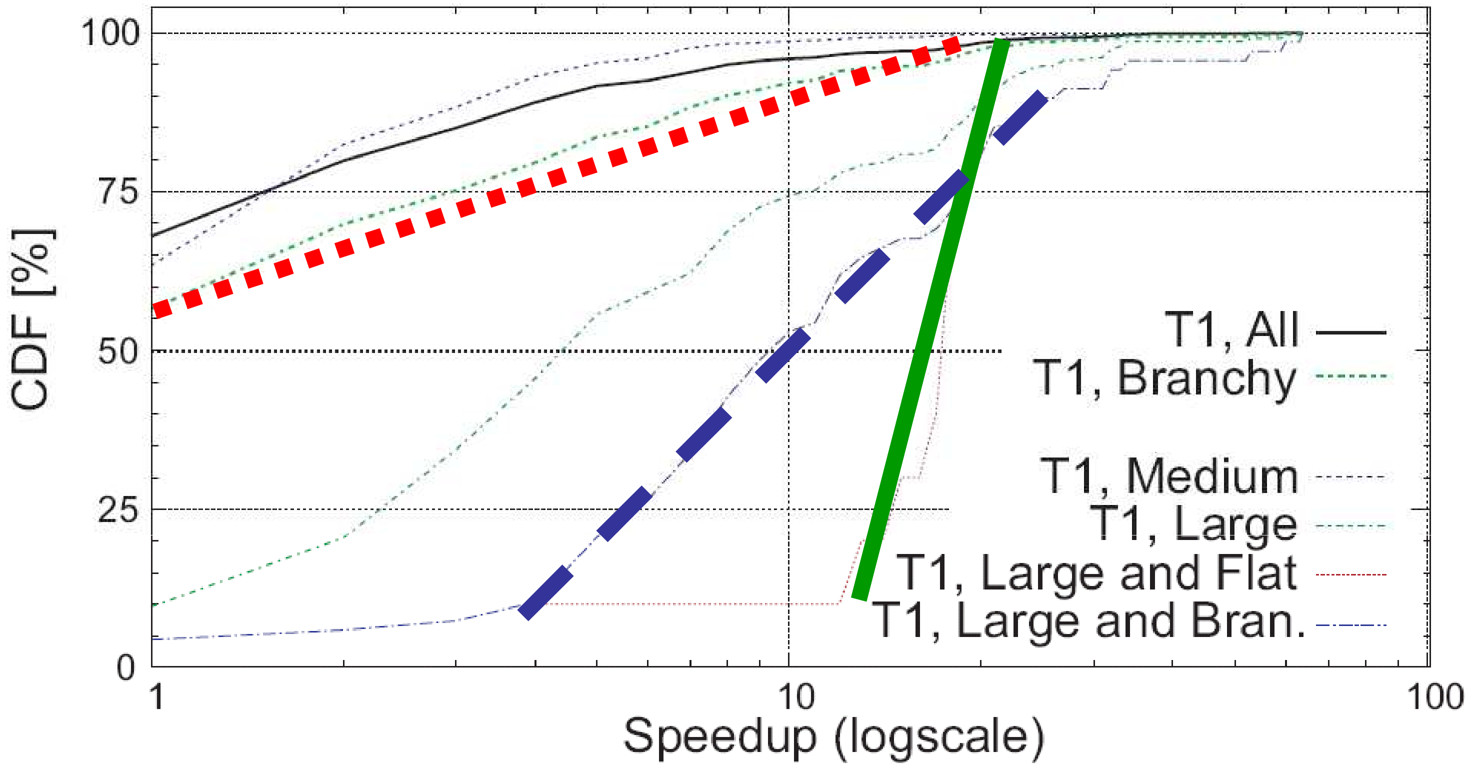
Workflows: Intrinsic Characteristics
Task Work Size



- >80% WFs take <2 minutes on 1000-SI2k machine
- >95% WFs take <10 minutes on 1000-SI2k machine

Ostermann et al., On the Characteristics of Grid Workflows, CoreGRID Integrated Research in Grid Computing (CGIW), 2008.

Workflows: Environment-Related Characteristics



- **Workflow class matters**: better SU for "easier" classes
- Large-and-Flat "easier" than Large-and-Branchy
- Large-and-Branchy "easier" than Branchy (o/head)

The Three "V"s of Big Data

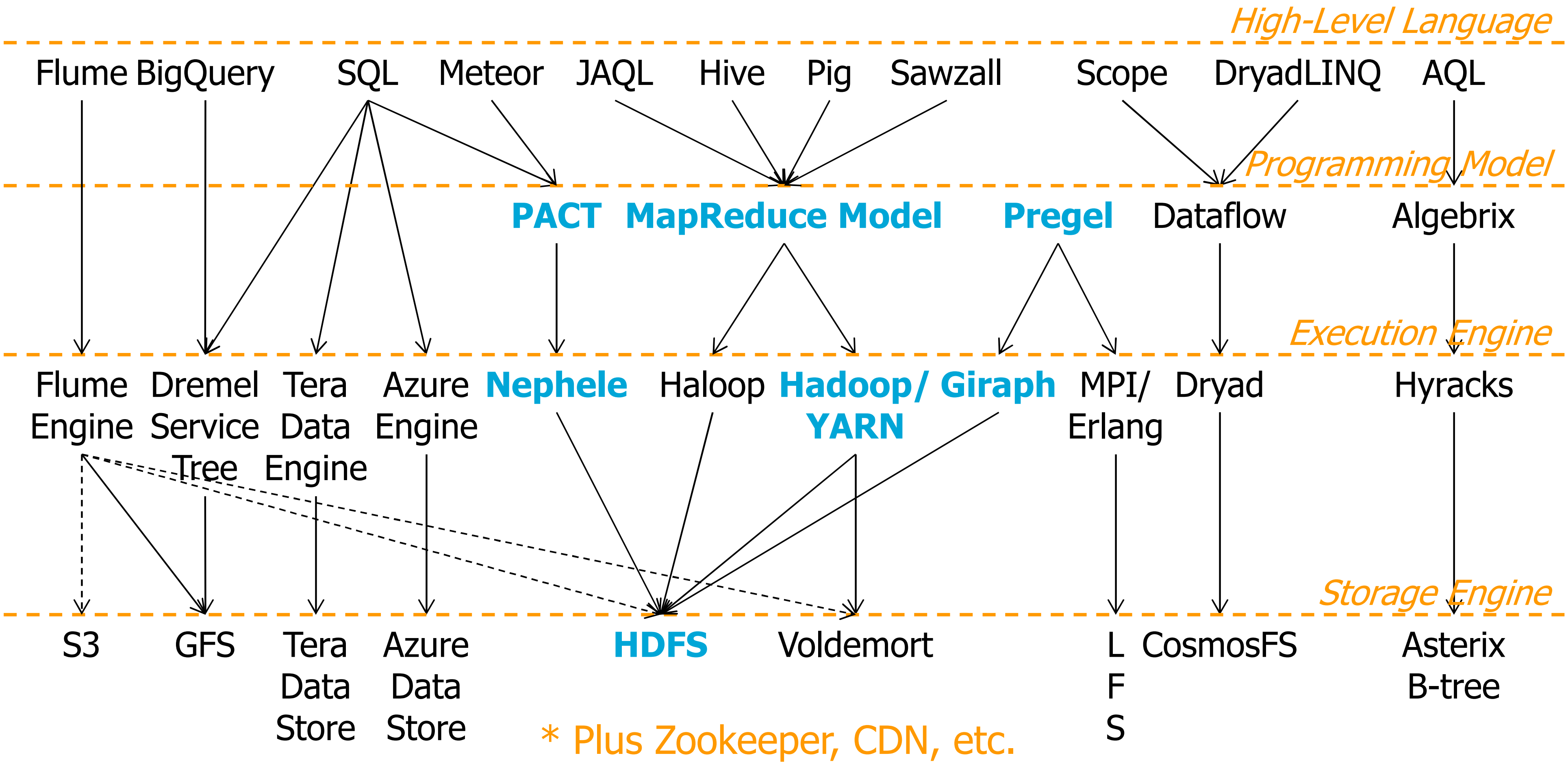
- Volume
  - More data vs. better models
  - Data grows exponentially
  - Analysis in near-real time to extract value
  - Scalable storage and distributed queries
- Velocity
  - Speed of the feedback loop
  - Gain competitive advantage: fast recommendations
  - Identify fraud, game cheating, predict customer churn faster
- Variety
  - The data can become messy: text, video, audio, etc.
  - Difficult to integrate into applications

Too big, too fast, does not comply with traditional DB

Adapted from: Doug Laney, "3D data management", META Group/Gartner report, Feb 2001. http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

Q0

Ecosystems of Big-Data Programming Models

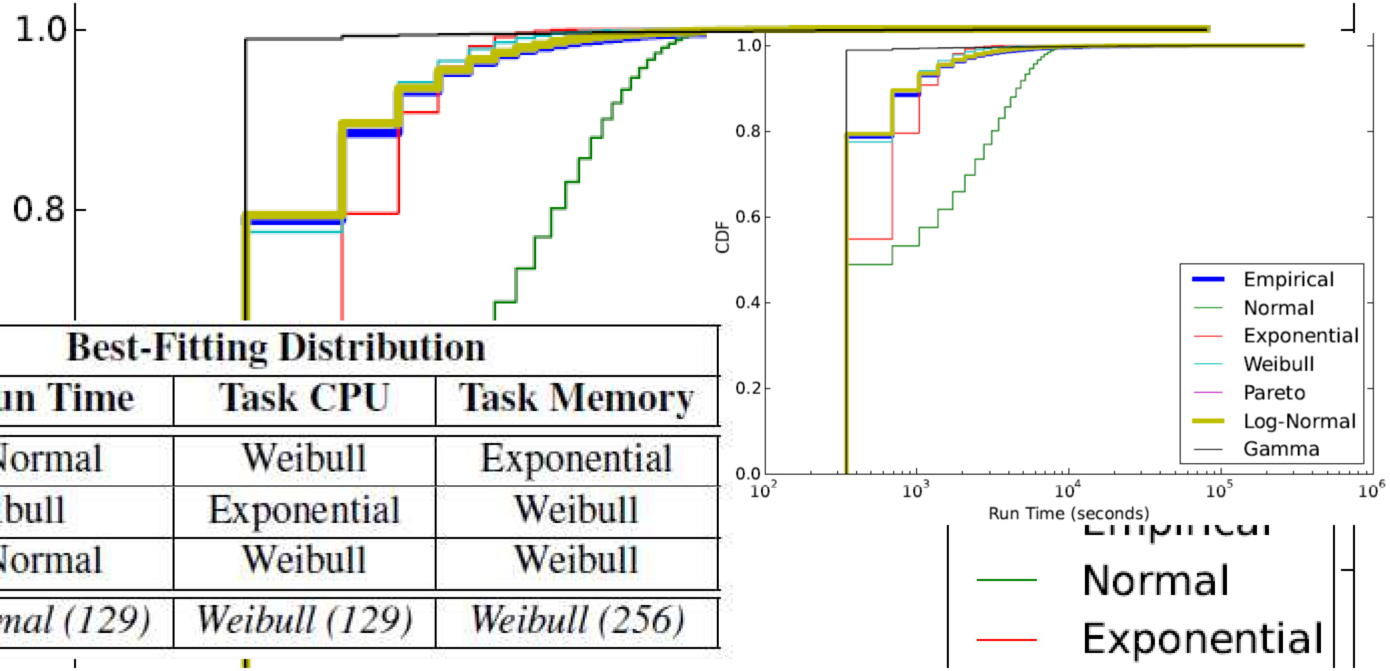


Adapted from: Dagstuhl Seminar on Information Management in the Cloud,
http://www.dagstuhl.de/program/calendar/partlist/?semnr=11321&SUOG

Q0

Our Statistical MapReduce Models

- Real traces
  - Yahoo
  - Google
  - 2 x Social N

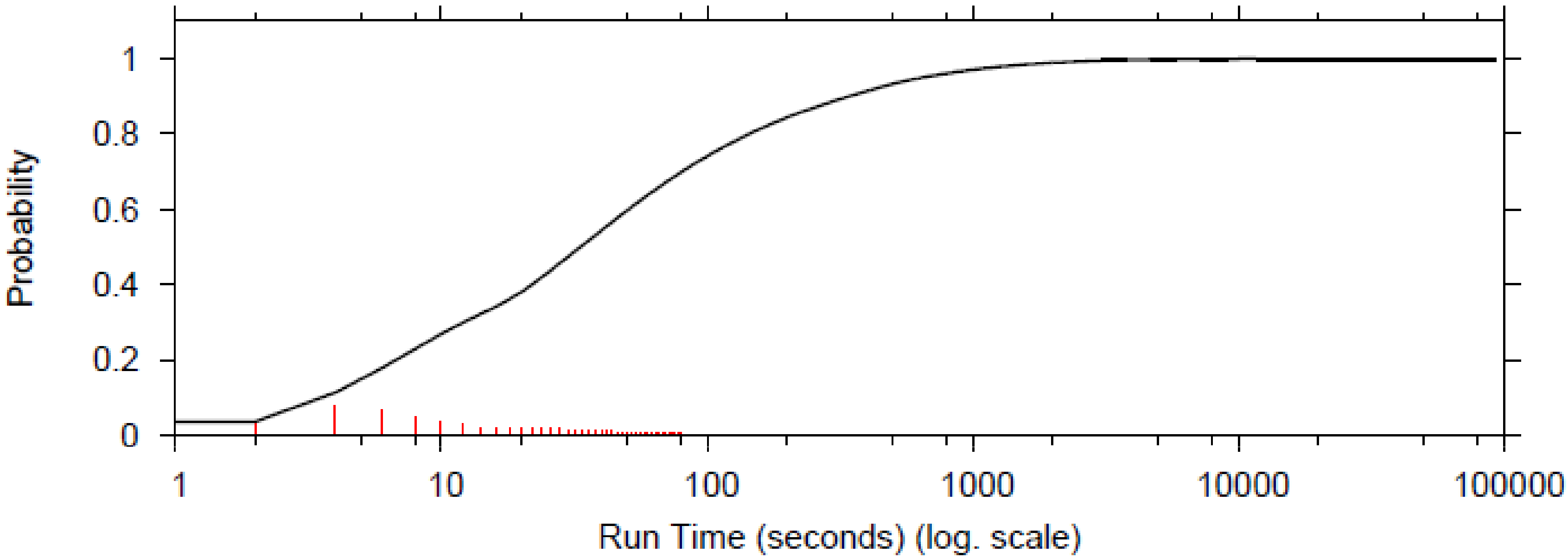


		Best-Fitting Distribution		
Job	**Task Count**	**Task Run Time**	**Task CPU**	**Task Memory**
1	1	Log-Normal	Weibull	Exponential
2	128	Weibull	Exponential	Weibull
3	128	Log-Normal	Weibull	Weibull
Overall Best Fit		*Log-Normal (129)*	*Weibull (129)*	*Weibull (256)*

Model	Tasks	Correlation	Map/Reduce Modeled	Sign. Level	Indirect Distr. Sel.
Complex Model	Indirect	Run time – Disk	Separately	0.05	Best fits
Relaxed Complex Model	Indirect	Run time – Disk	Separately	0.02	All fits
Safe Complex Model	Direct	Run time – Disk	Separately	0.05	–
Simple Model	Direct	–	Together	0.05	–

de Ruiter and Iosup. A workload model for MapReduce. MSc thesis at TU Delft. Jun 2012. Available online via TU Delft Library, http://library.tudelft.nl .

MR tasks: Runtime, I/O

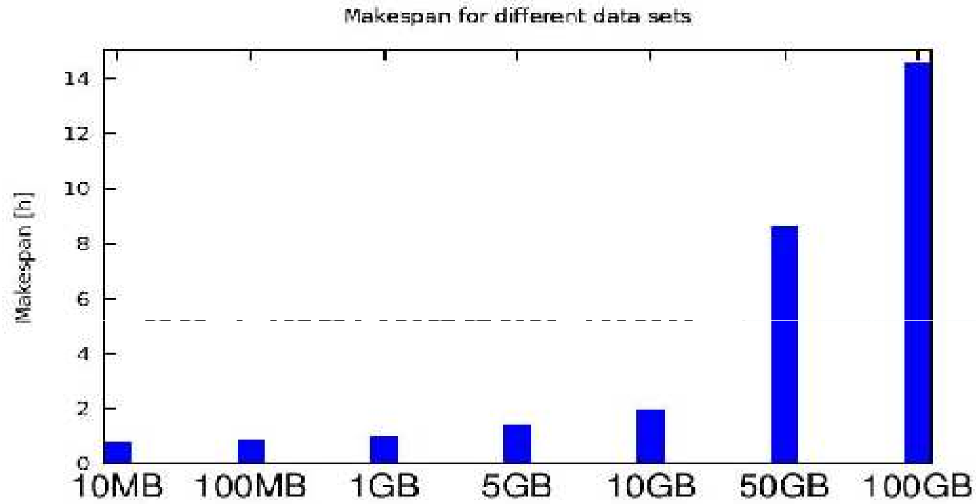


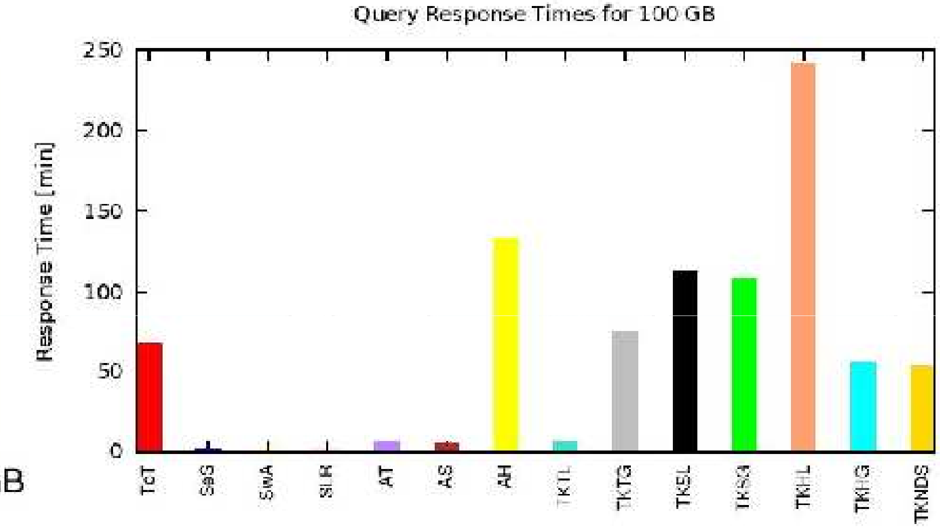
- Job runtime median: 30s to 3 minutes
- Job runtime mean: 2.5 minutes to 45 minutes
- Data intensive?
  Strong correlation runtime—disk operations

Our Real-World MapReduce Workflow

Q0

- BTWorld (2009—ongoing)
  - 1,000s of trackers, 100Ms users, 30M+ shared files





- Non-trivial algorithms
  - SQL aggregations, joins, selections, projections
- Execution plan important

```
SELECT *
FROM scrapes
NATURAL JOIN (
  SELECT tracker
  FROM TKT_local
  GROUP BY tracker
  ORDER BY MAX(sessions) DESC
  LIMIT k);
```

TKT: Top-K trackers, by # users

Hegeman, Ghit, Capota, Hidders, Epema, Iosup. The BTWorld Use Case for Big Data Analytics with MapReduce, 2013.

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

IaaS Cloud Performance: Our Team

Alexandru Iosup
TU Delft

Performance
Variability
Isolation
Multi-tenancy
Benchmarking

Dick Epema
TU Delft

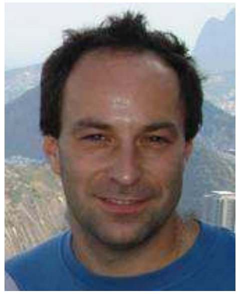
Performance
IaaS clouds

Nezih Yigitbasi
TU Delft

Performance
Variability

Athanasios Antoniou
TU Delft

Performance
Isolation



Radu Prodan
U.Isbk.

Benchmarking

Thomas Fahringer
U.Isbk.

Benchmarking

Simon Ostermann
U.Isbk.

Benchmarking

What I'll Talk About

IaaS Cloud Performance (Q1)

1. Previous work
2. Experimental setup
3. Experimental results
4. Implications on real-world workloads

Some Previous Work
(>50 important references across our studies)

Virtualization Overhead

- Loss below 5% for computation [Barham03] [Clark04]
- Loss below 15% for networking [Barham03] [Menon05]
- Loss below 30% for parallel I/O [Vetter08]
- Negligible for compute-intensive HPC kernels [You06] [Panda06]

Cloud Performance Evaluation

- Performance and cost of executing a sci. workflows [Dee08]
- Study of Amazon S3 [Palankar08]
- Amazon EC2 for the NPB benchmark suite [Walker08] or selected HPC benchmarks [Hill08]
- CloudCmp [Li10]
- Kosmann et al.

Production IaaS Cloud Services

Q1

- **Production IaaS cloud:** lease resources (infrastructure) to users, operate on the market and have active customers

Name	Cores (ECUs)	RAM [GB]	Archi. [bit]	Disk [GB]	Cost [$/h]
Amazon EC2					
m1.small	1 (1)	1.7	32	160	0.1
m1.large	2 (4)	7.5	64	850	0.4
m1.xlarge	4 (8)	15.0	64	1,690	0.8
c1.medium	2 (5)	1.7	32	350	0.2
c1.xlarge	8 (20)	7.0	64	1,690	0.8
GoGrid (GG)					
GG.small	1	1.0	32	60	0.19
GG.large	1	1.0	64	60	0.19
GG.xlarge	3	4.0	64	240	0.76
Elastic Hosts (EH)					
EH.small	1	1.0	32	30	£0.042
EH.large	1	4.0	64	30	£0.09
Mosso					
Mosso.small	4	1.0	64	40	0.06
Mosso.large	4	4.0	64	160	0.24

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Our Method

Q1

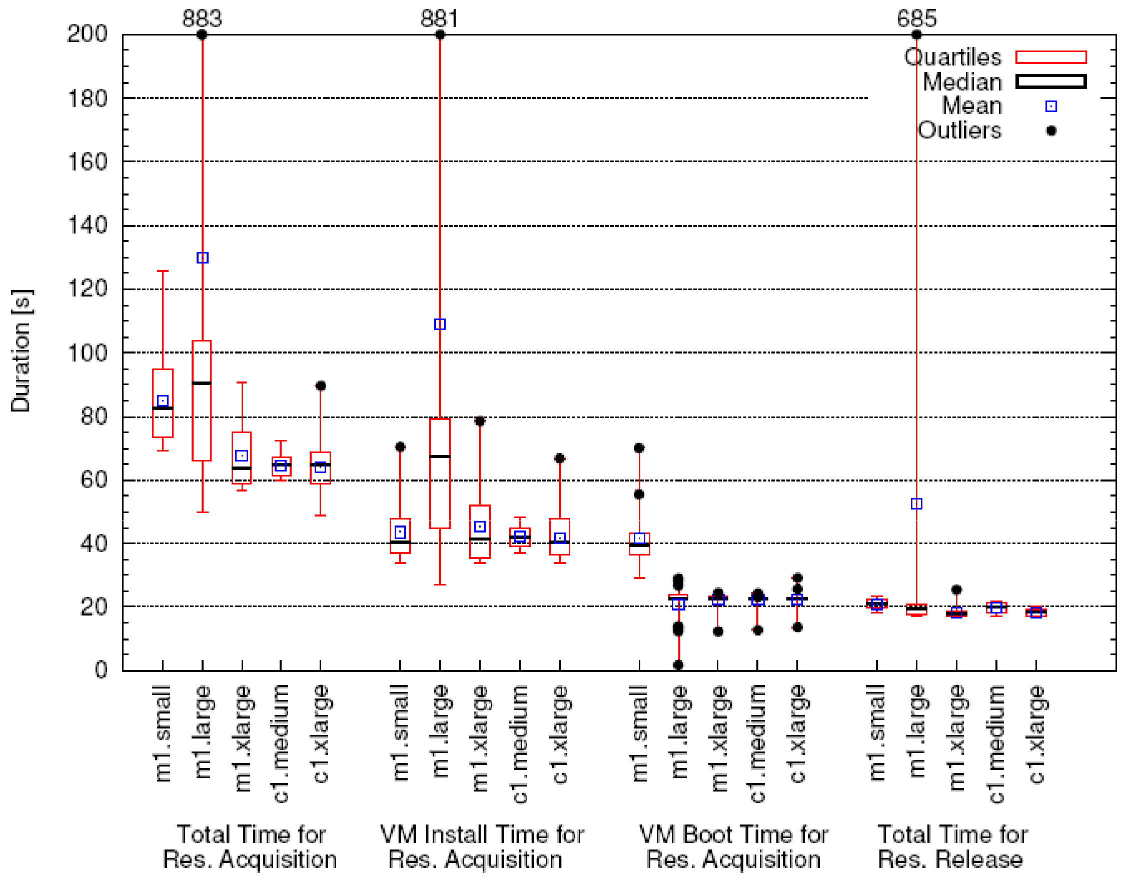
- Based on general performance technique: model performance of individual components; system performance is performance of workload + model [Saavedra and Smith, ACM TOCS'96]
- Adapt to clouds:
  1. Cloud-specific elements: resource provisioning and allocation
  2. Benchmarks for single- and multi-machine jobs
  3. Benchmark CPU, memory, I/O, etc.:

Type	Suite/Benchmark	Resource	Unit
SI	LMbench/all [24]	Many	Many
SI	Bonnie/all [25], [26]	Disk	MBps
SI	CacheBench/all [27]	Memory	MBps
MI	HPCC/HPL [28], [29]	CPU	GFLOPS
MI	HPCC/DGEMM [30]	CPU	GFLOPS
MI	HPCC/STREAM [30]	Memory	GBps
MI	HPCC/RandomAccess [31]	Network	MUPS
MI	HPCC/b_{eff}(lat.,bw.) [32]	Comm.	μs, GBps

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Single Resource Provisioning/Release

Q1

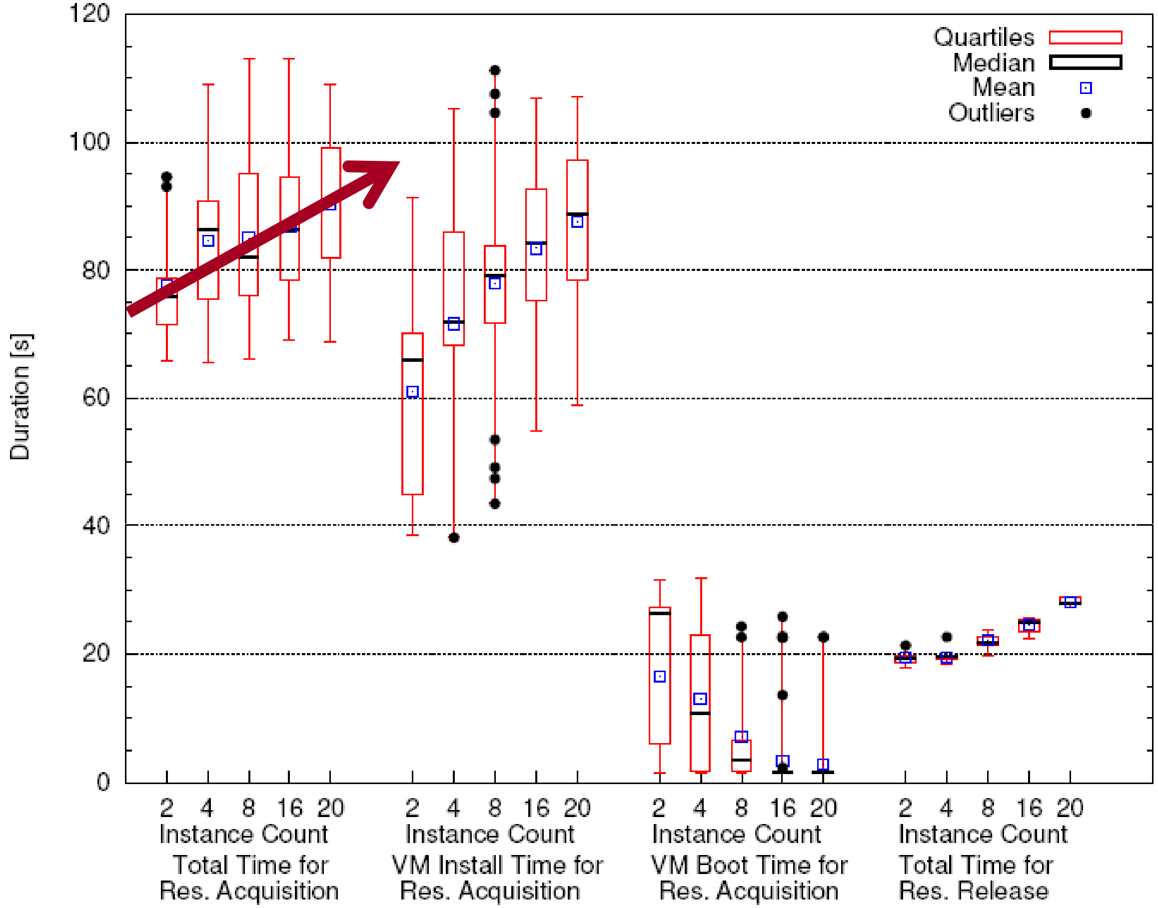


- Time depends on instance type
- Boot time non-negligible

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Multi-Resource Provisioning/Release

Q1



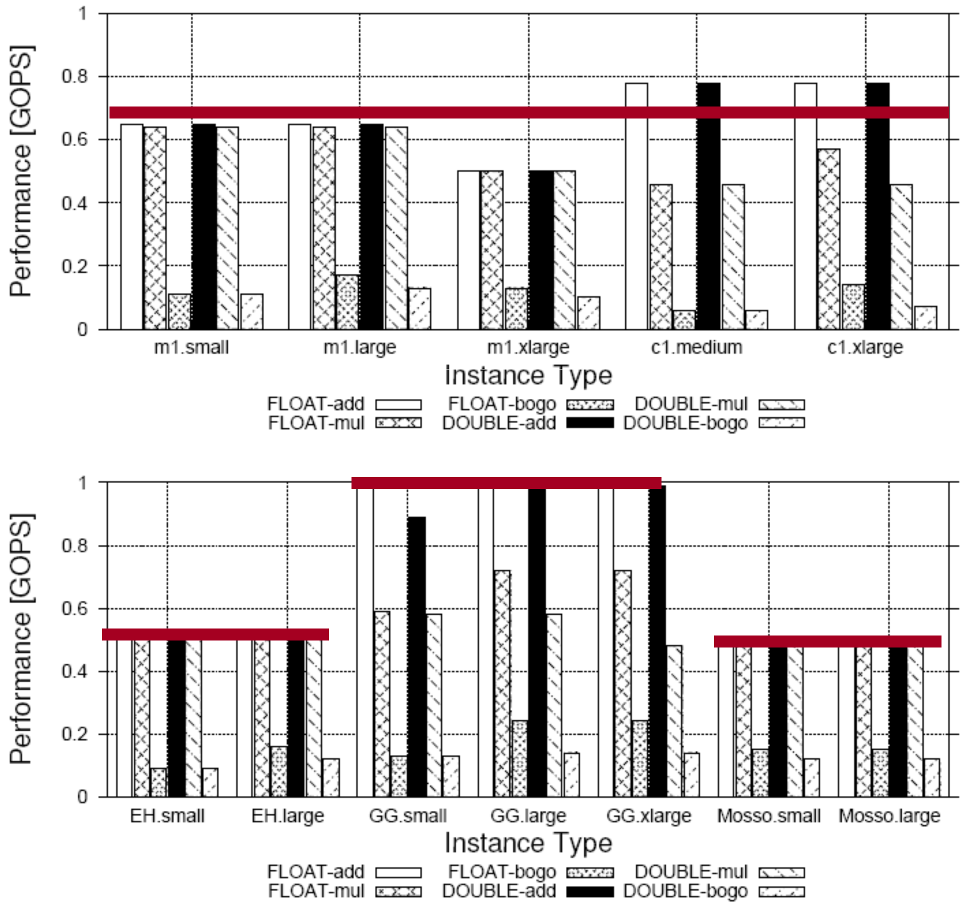
- Time for ***multi-***resource increases with number of resources

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

CPU Performance of Single Resource

Q1

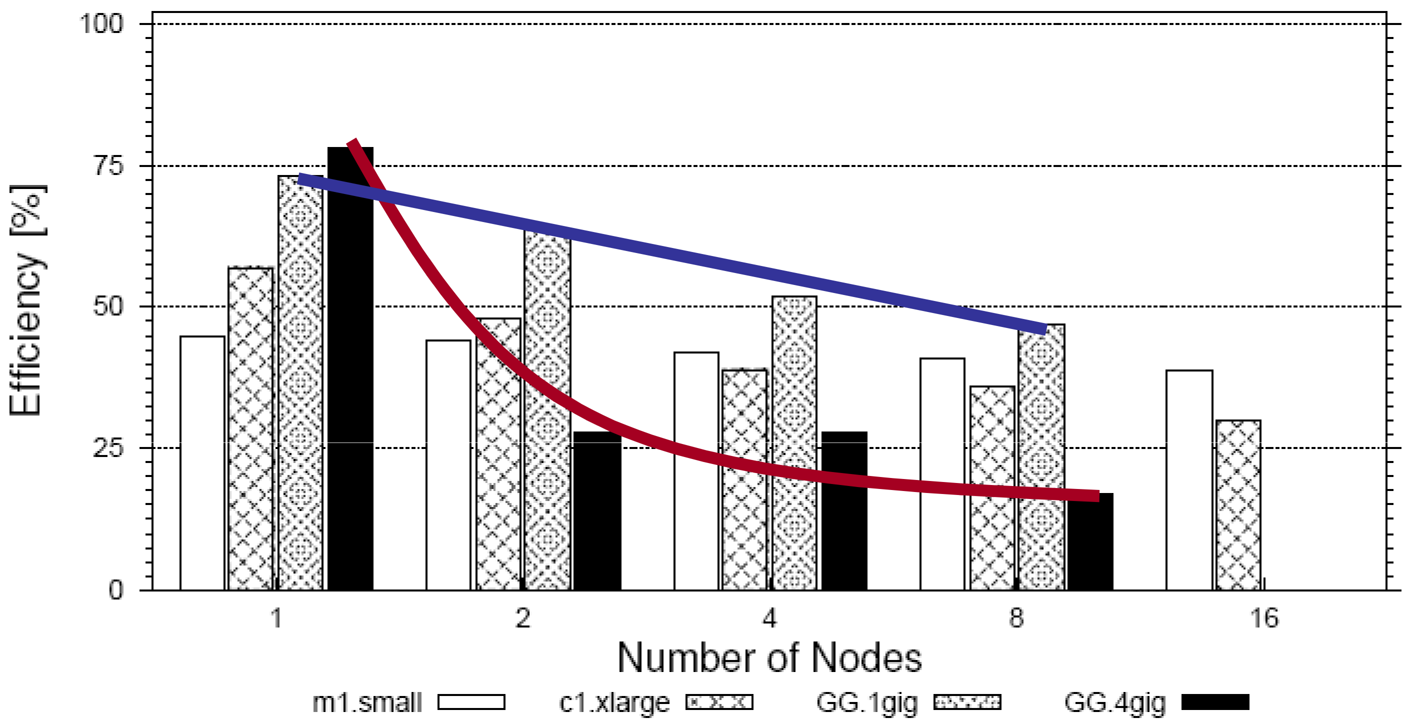
- ECU definition: "a 1.1 GHz 2007 Opteron" ~ 4 flops per cycle at full pipeline, which means at peak performance one ECU equals 4.4 gigaflops per second (GFLOPS)
- Real performance 0.6..0.1 GFLOPS = ~1/4..1/7 theoretical peak



Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

HPLinpack Performance (Parallel)

Q1

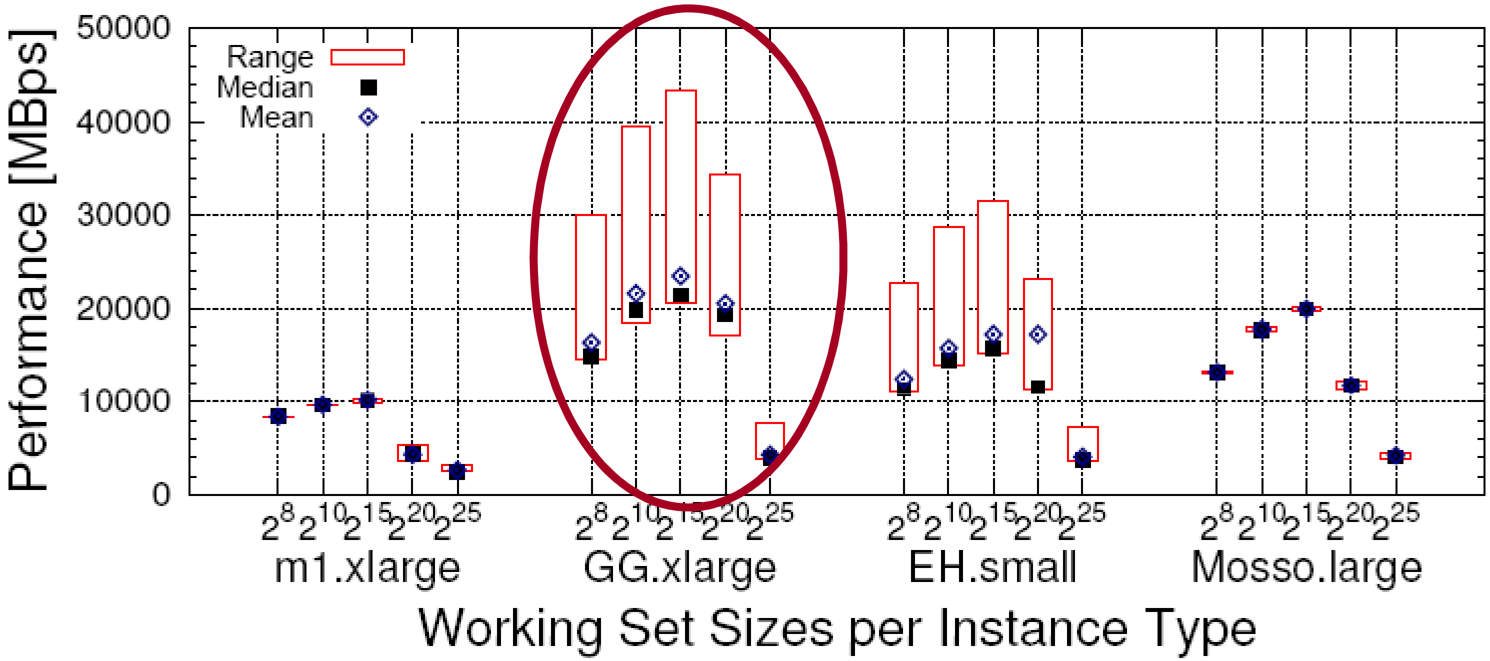


- Low efficiency for parallel compute-intensive applications
- Low performance vs cluster computing and supercomputing

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Performance Stability (Variability)

Q1
Q2



- High performance variability for the best-performing instances

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Summary

Q1

- Much lower performance than theoretical peak
  - Especially CPU (GFLOPS)
- Performance variability
- Compared results with some of the commercial alternatives (see report)

Implications: Simulations

Q1

- Input: real-world workload traces, grids and PPEs
- Running in
  - Original env.
  - Cloud with source-like perf.
  - Cloud with measured perf.
- Metrics
  - WT, ReT, BSD(10s)
  - Cost [CPU-h]

Trace ID, Source (Trace ID in Archive)	Trace Time [mo.]	Number of Jobs	Number of Users	System Size Sites	System Size CPUs	Load [%]
Grid Workloads Archive [13], 6 traces						
1. DAS-2 (1)	18	1.1M	333	5	0.4K	15+
2. RAL (6)	12	0.2M	208	1	0.8K	85+
3. GLOW (7)	3	0.2M	18	1	1.6K	60+
4. Grid3 (8)	18	1.3M	19	29	3.5K	-
5. SharcNet (10)	13	1.1M	412	10	6.8K	-
6. LCG (11)	1	0.2M	216	200+	24.4K	-
Parallel Workloads Archive [16], 4 traces						
7. CTC SP2 (6)	11	0.1M	679	1	0.4K	66
8. SDSC SP2 (9)	24	0.1M	437	1	0.1K	83
9. LANLO2K (10)	5	0.1M	337	1	2.0K	64
10. SDSC DS (19)	13	0.1M	460	1	1.7K	63

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Implications: Results

Q1

Trace ID	Source env. (Grid/PPI) AWT [s]	AReT [s]	ABSD (10s)	Cloud (real performance) AReT [s]	ABSD (10s)	Total Cost [CPU-h,M]	Cloud (source performance) AReT [s]	ABSD (10s)	Total Cost [CPU-h,M]
DAS-2	432	802	11	2,292	2.39	2	450	2	1.19
RAL	13,214	27,807	68	131,300	1	40	18,837	1	6.39
GLOW	9,162	17,643	55	59,448	1	3	8,561	1	0.60
Grid3	-	7,199	-	50,470	3	19	7,279	3	3.60
SharcNet	31,017	61,682	242	219,212	1	73	31,711	1	11.34
LCG	-	9,011	-	63,158	1	3	9,091	1	0.62
CTC SP2	25,748	37,019	78	75,706	1	2	11,351	1	0.30
SDSC SP2	26,705	33,388	389	46,818	2	1	6,763	2	0.16
LANL O2K	4,658	9,594	61	37,786	2	1	5,016	2	0.26
SDSC DS	32,271	33,807	516	57,065	2	2	6,790	2	0.25

- Cost: Clouds, real >> Clouds, source
- Performance:
  - AReT: Clouds, real >> Source env. (**bad**)
  - AWT,ABSD: Clouds, real << Source env. (**good**)

Iosup et al., Performance Analysis of Cloud Computing Services for Many Tasks Scientific Computing, (IEEE TPDS 2011).

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

IaaS Cloud Performance: Our Team

Alexandru Iosup
TU Delft

Performance
Variability
Isolation
Multi-tenancy
Benchmarking

Dick Epema
TU Delft

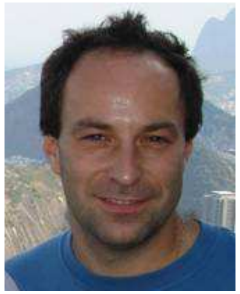
Performance
IaaS clouds

Nezih Yigitbasi
TU Delft

Performance
Variability

Athanasios Antoniou
TU Delft

Performance
Isolation



Radu Prodan
U.Isbk.

Benchmarking

Thomas Fahringer
U.Isbk.

Benchmarking

Simon Ostermann
U.Isbk.

Benchmarking

What I'll Talk About

IaaS Cloud Performance Variability (Q2)

1. Experimental setup
2. Experimental results
3. Implications on real-world workloads

Production Cloud Services

Q2

- **Production cloud:** operate on the market and have active customers

- **IaaS/PaaS: Amazon Web Services (AWS)**
  - EC2 (Elastic Compute Cloud)
  - S3 (Simple Storage Service)
  - SQS (Simple Queueing Service)
  - SDB (Simple Database)
  - FPS (Flexible Payment Service)

- **PaaS: Google App Engine (GAE)**
  - Run (Python/Java runtime)
  - Datastore (Database) ~ SDB
  - Memcache (Caching)
  - URL Fetch (Web crawling)

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Q2

Our Method [1/3]
Performance Traces

- CloudStatus*
  - Real-time values and weekly averages for most of the AWS and GAE services

- Periodic performance probes
  - Sampling rate is under 2 minutes

* www.cloudstatus.com

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Our Method [2/3]
Analysis

Q2

1. Find out whether variability is present
   - Investigate several months whether the performance metric is highly variable

2. Find out the characteristics of variability
   - Basic statistics: the five quartiles (Q_0-Q_4) including the median (Q_2), the mean, the standard deviation
   - Derivative statistic: the IQR (Q_3-Q_1)
   - CoV > 1.1 indicate high variability

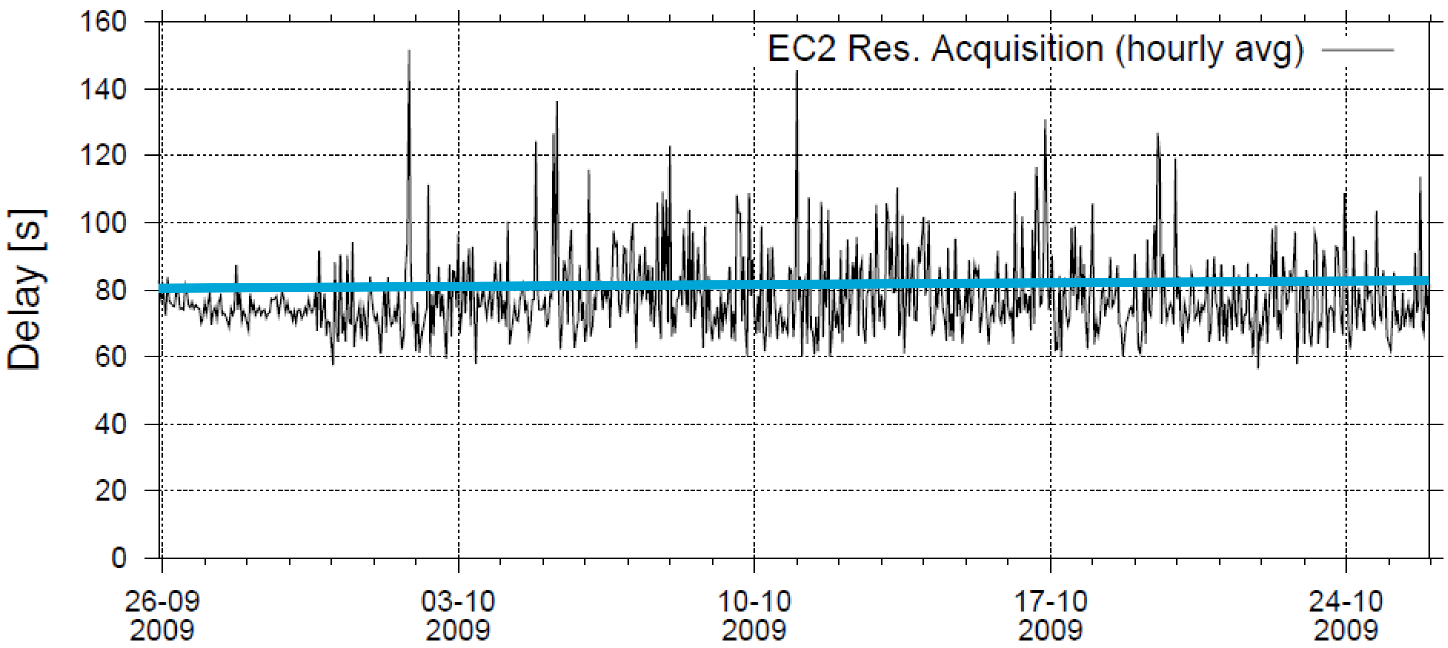
3. Analyze the performance variability time patterns
   - Investigate for each performance metric the presence of daily/monthly/weekly/yearly time patterns
   - E.g., for monthly patterns divide the dataset into twelve subsets and for each subset compute the statistics and plot for visual inspection

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Our Method [3/3]
Is Variability Present?

Q2

- **Validated Assumption:** The performance delivered by production services is variable.

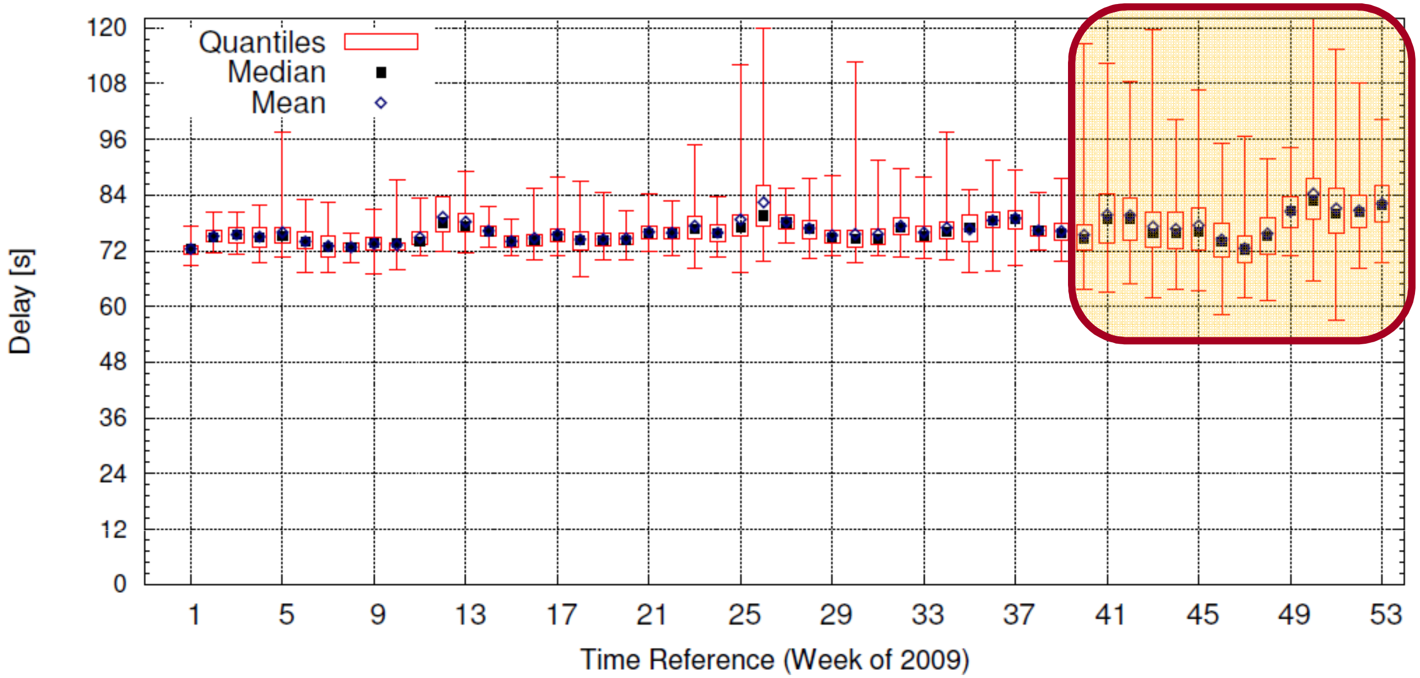


Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

AWS Dataset (1/4): EC2

Variable Performance

Q2

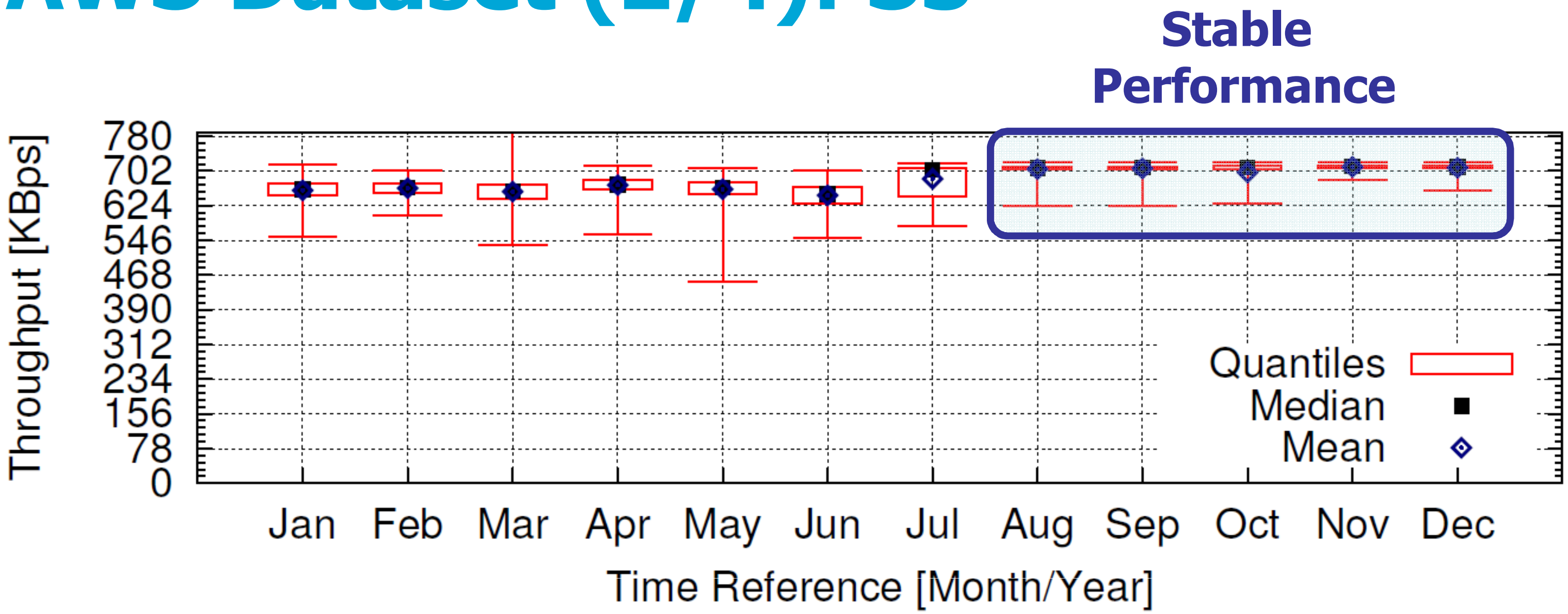


- **Deployment Latency [s]:** Time it takes to start a small instance, from the startup to the time the instance is available
- Higher IQR and range from week 41 to the end of the year; possible reasons:
  - Increasing EC2 user base
  - Impact on applications using EC2 for auto-scaling

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Q2

AWS Dataset (2/4): S3

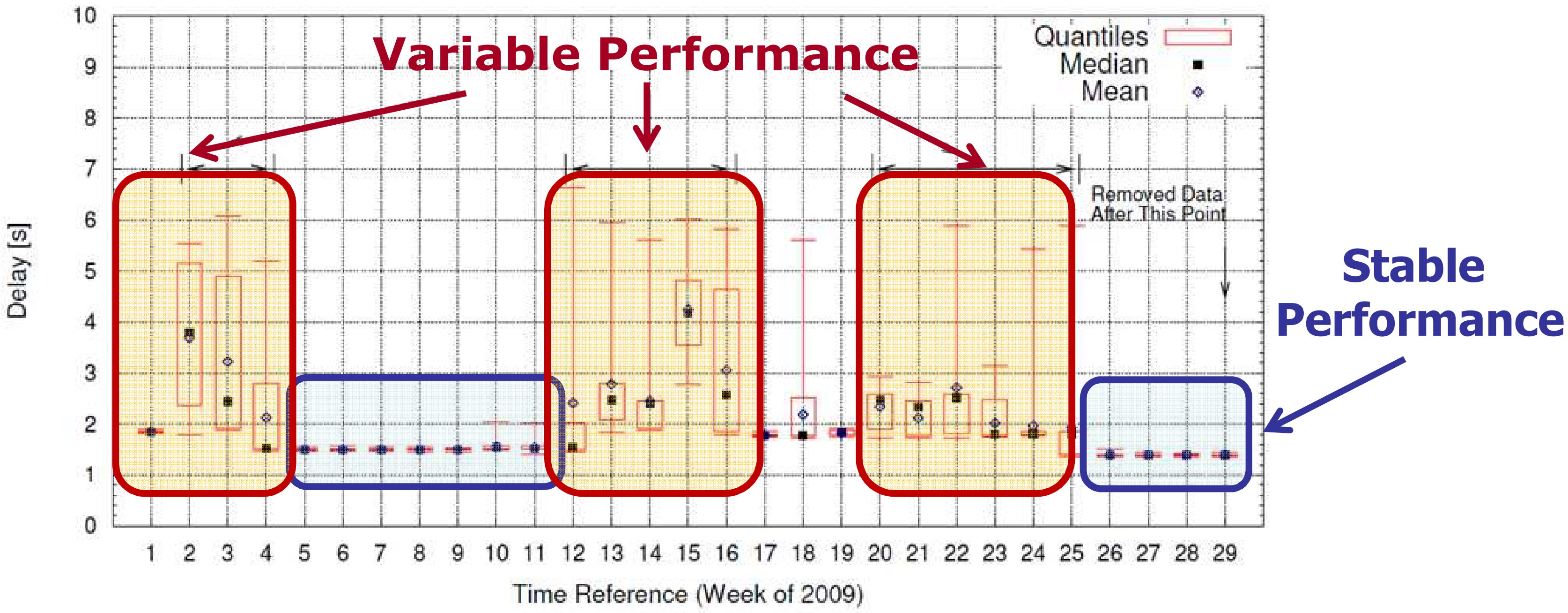


- **Get Throughput [bytes/s]:** Estimated rate at which an object in a bucket is read
- The last five months of the year exhibit much lower IQR and range
  - More stable performance for the last five months
  - Probably due to software/infrastructure upgrades

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

AWS Dataset (3/4): SQS

Q2



- **Average Lag Time [s]:** Time it takes for a posted message to become available to read. Average over multiple queues.
- Long periods of stability (low IQR and range)
- Periods of high performance variability also exist

AWS Dataset (4/4): Summary

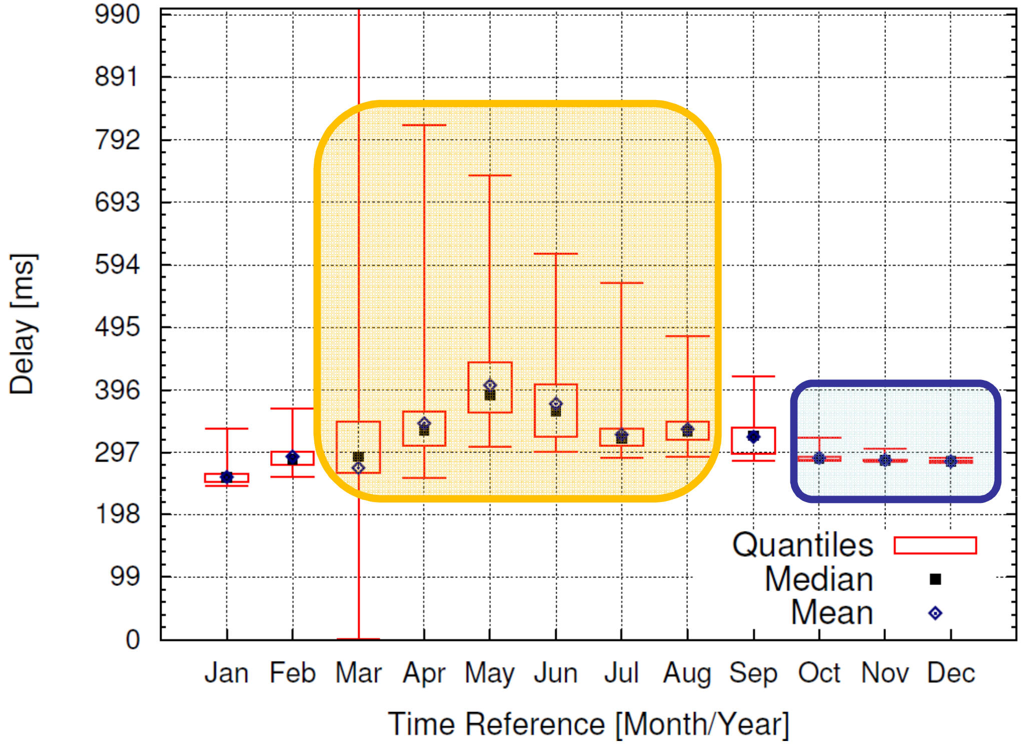
Q2

- **All services exhibit time patterns in performance**
- EC2: periods of special behavior
- SDB and S3: daily, monthly and yearly patterns
- SQS and FPS: periods of special behavior

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

GAE Dataset (1/4): Run Service

Q2

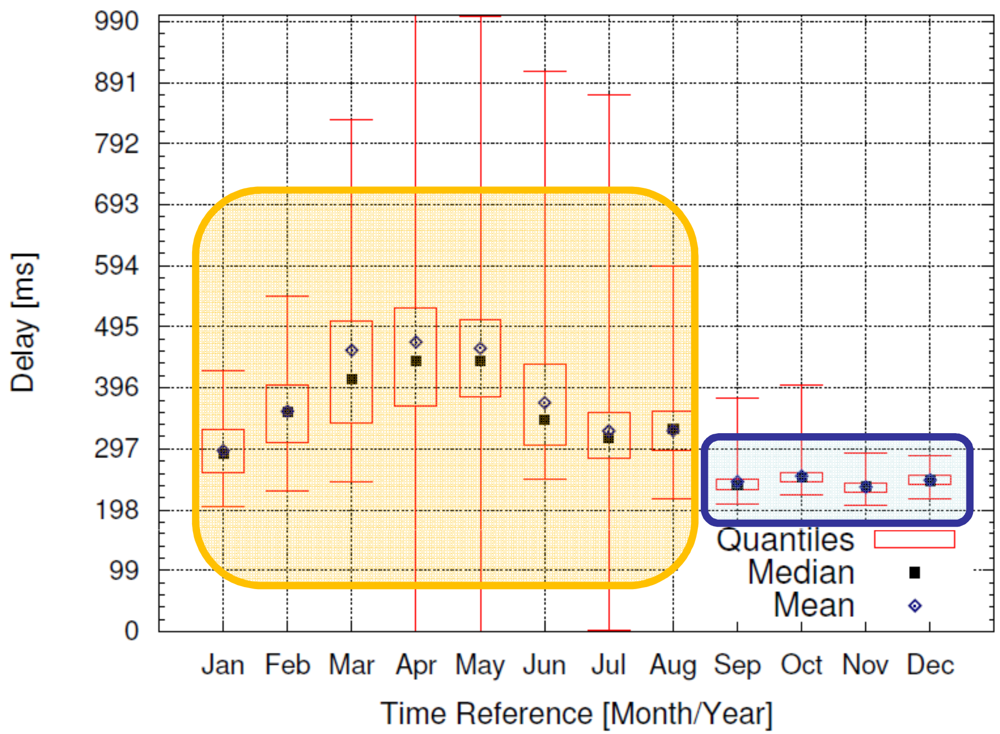


- **Fibonacci [ms]:** Time it takes to calculate the 27th Fibonacci number
- Highly variable performance until September
- Last three months have stable performance (low IQR and range)

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

GAE Dataset (2/4): Datastore

Q2

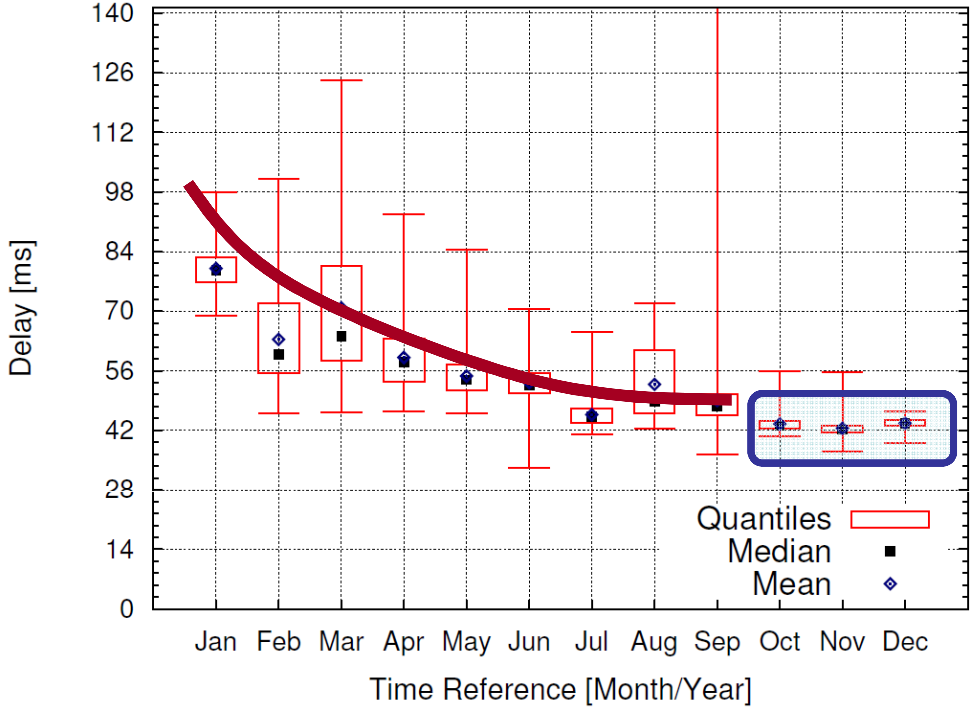


- **Read Latency [s]:** Time it takes to read a "User Group"
- Yearly pattern from January to August
- The last four months of the year exhibit much lower IQR and range
  - More stable performance for the last five months
  - Probably due to software/infrastructure upgrades

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

GAE Dataset (3/4): Memcache

Q2



- **PUT [ms]:** Time it takes to put 1 MB of data in memcache.
- Median performance per month has an increasing trend over the first 10 months
- The last three months of the year exhibit stable performance

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

GAE Dataset (4/4): Summary

Q2

- All services exhibit time patterns

- Run Service: daily patterns and periods of special behavior
- Datastore: yearly patterns and periods of special behavior
- Memcache: monthly patterns and periods of special behavior
- URL Fetch: daily and weekly patterns, and periods of special behavior

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Experimental Setup (1/2): Simulations

Q2

- Trace based simulations for three applications
- **Input**
  - GWA traces
  - Number of daily unique users
  - Monthly performance variability

Application	Service
Job Execution	GAE Run
Selling Virtual Goods	AWS FPS
Game Status Maintenance	AWS SDB/GAE Datastore

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Experimental Setup (2/2): Metrics

Q2

- Average Response Time and Average Bounded Slowdown
- Cost in millions of consumed CPU hours
- **Aggregate Performance Penalty** -- APP(t)

$$\frac{P(t)}{P_{ref}} \times \frac{U(t)}{\max U(t)}$$

  - Pref (Reference Performance): Average of the twelve monthly medians
  - P(t): **random** value sampled from the distribution corresponding to the current month at time t (*Performance is like a box of chocolates, you never know what you're gonna get ~ Forrest Gump*)
  - max U(t): max number of users over the whole trace
  - U(t): number of users at time t
  - **APP—the lower the better**

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Q2

Grid & PPE Job Execution (1/2): Scenario

- Execution of compute-intensive jobs typical for grids and PPEs on cloud resources
- **Traces**

Trace ID, Source (Trace ID in Archive)	Trace			System		
		Number of		Size		Load
	Mo.	Jobs	Users	Sites	CPUs	[%]
Grid Workloads Archive [17], 3 traces						
1. RAL (6)	12	0.2M	208	1	0.8K	85+
2. Grid3 (8)	18	1.3M	19	29	3.5K	-
3. SharcNet (10)	13	1.1M	412	10	6.8K	-
Parallel Workloads Archive [18], 2 traces						
4. CTC SP2 (6)	11	0.1M	679	1	430	66
5. SDSC SP2 (9)	24	0.1M	437	1	128	83

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Grid & PPE Job Execution (2/2): Results

Q2

- All metrics differ by less than 2% between cloud with stable and the cloud with variable performance
- Impact of service performance variability is low for this scenario

Trace ID	Cloud with Stable Performance ART [s]	Stable Performance ABSD (10s)	Stable Performance Cost	Cloud with Variable Performance ART [s]	Variable Performance ABSD (10s)	Variable Performance Cost
RAL	18,837	1.89	6.39	18,877	1.90	6.40
Grid3	7,279	4.02	3.60	7,408	4.02	3,64
SharcNet	31,572	2.04	11.29	32,029	2.06	11.42
CTC SP2	11,355	1.45	0.29	11,390	1,47	0.30
SDSC SP2	7,473	1.75	0.15	7,537	1.75	0.15

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Selling Virtual Goods (1/2): Scenario

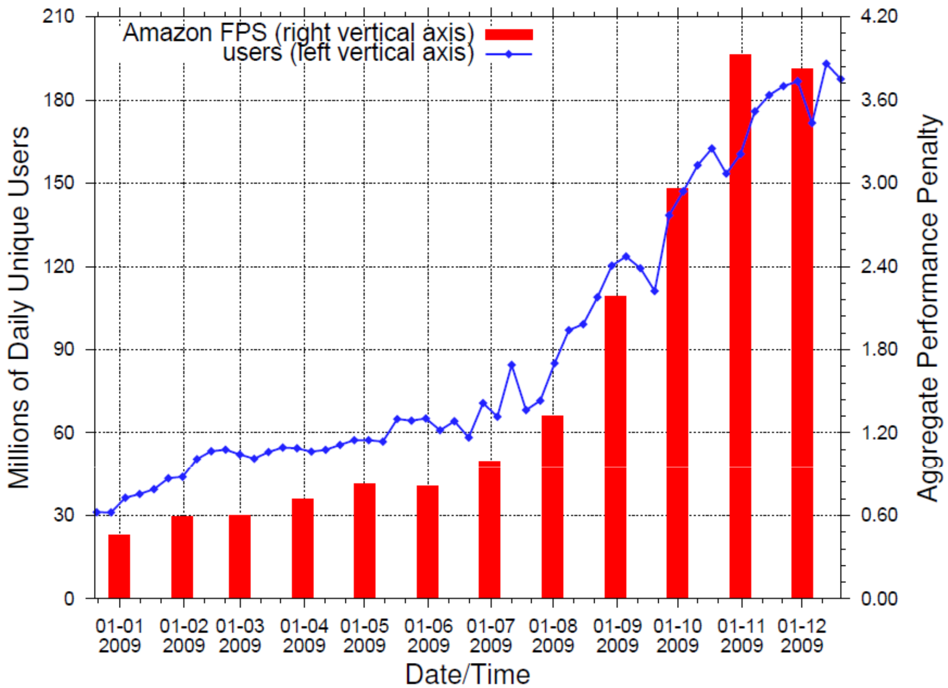
- Virtual good selling application operating on a large-scale social network like Facebook
- Amazon FPS is used for payment transactions
- Amazon FPS performance variability is modeled from the AWS dataset
- **Traces:** Number of daily unique users of Facebook*

* Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Selling Virtual Goods (2/2): Results

Q2

- Significant cloud performance decrease of FPS during the last four months + increasing number of daily users is well-captured by **APP**



- APP metric can trigger and motivate the decision of switching cloud providers

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Game Status Maintenance (1/2): Scenario

Q2

- Maintenance of game status for a large-scale social game such as Farm Town or Mafia Wars which have millions of unique users daily

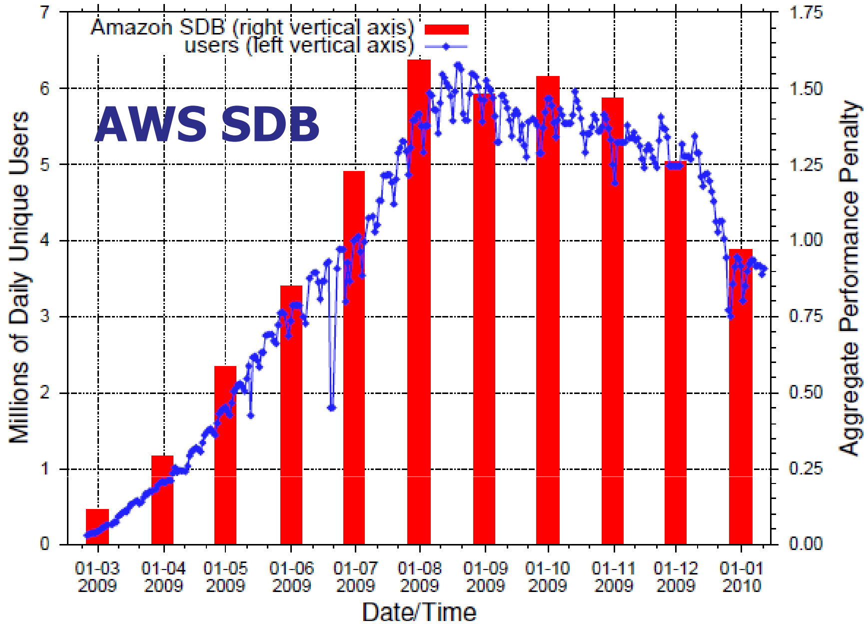
- AWS SDB and GAE Datastore

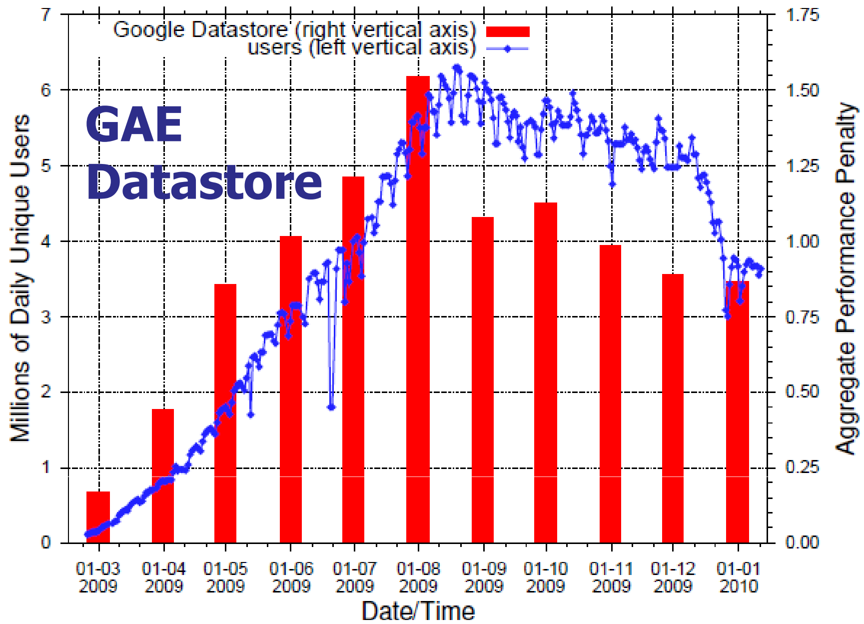
- We assume that the number of database operations depends linearly on the number of daily unique users

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Game Status Maintenance (2): Results

Q2





- Big discrepancy between SDB and Datastore services
- **Sep'09-Jan'10:** APP of Datastore is well below than that of SDB due to increasing performance of Datastore
- APP of Datastore ~**1** => no performance penalty
- APP of SDB ~**1.4** => %40 higher performance penalty than SDB

TU Delft
Delft University of Technology

Iosup, Yigitbasi, Epema. On the Performance Variability of Production Cloud Services, (IEEE CCgrid 2011).

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

IaaS Cloud Policies: Our Team

Alexandru Iosup
TU Delft

Provisioning
Allocation
Elasticity
Utility
Isolation
Multi-Tenancy

Dick Epema
TU Delft

Provisioning
Allocation
Koala

Bogdan Ghit
TU Delft

Provisioning
Allocation
Koala

Athanasios Antoniou
TU Delft

Provisioning
Allocation
Isolation
Utility

Orna Agmon-Ben Yehuda
Technion
Elasticity, Utility

David Villegas
FIU/IBM
Elasticity, Utility

What I'll Talk About

Provisioning and Allocation Policies for IaaS Clouds (Q3)

1. Experimental setup
2. Experimental results
3. Some links on Portfolio Scheduling
4. Some links on Elastic MapReduce

Q3

Provisioning and Allocation Policies*

* For User-Level Scheduling

- Provisioning

Policy	Class	Trigger	Adaptive
Startup	Static	–	–
OnDemand	Dynamic	QueueSize	No
ExecTime	Dynamic	Exec.Time	Yes
ExecAvg	Dynamic	Exec.Time	Yes
ExecKN	Dynamic	Exec.Time	Yes
QueueWait	Dynamic	Wait Time	Yes

- Allocation

Policy	Queue-based	Known job durations
FCFS	Yes	No
FCFS-NW	No	No
SJF	Yes	Yes

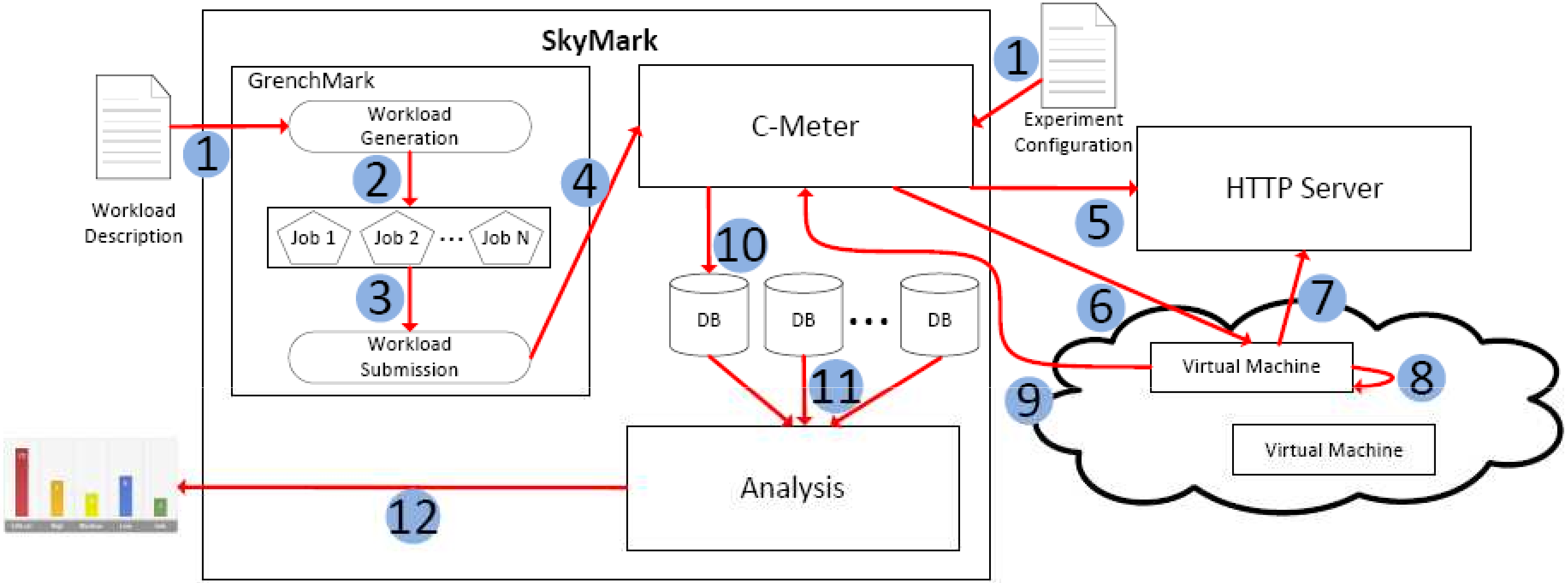
- Also looked at combined Provisioning + Allocation policies

The SkyMark Tool for IaaS Cloud Benchmarking

Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid 2012

Experimental Tool: SkyMark

Q3



Provisioning and Allocation policies steps 6+9, and 8, respectively

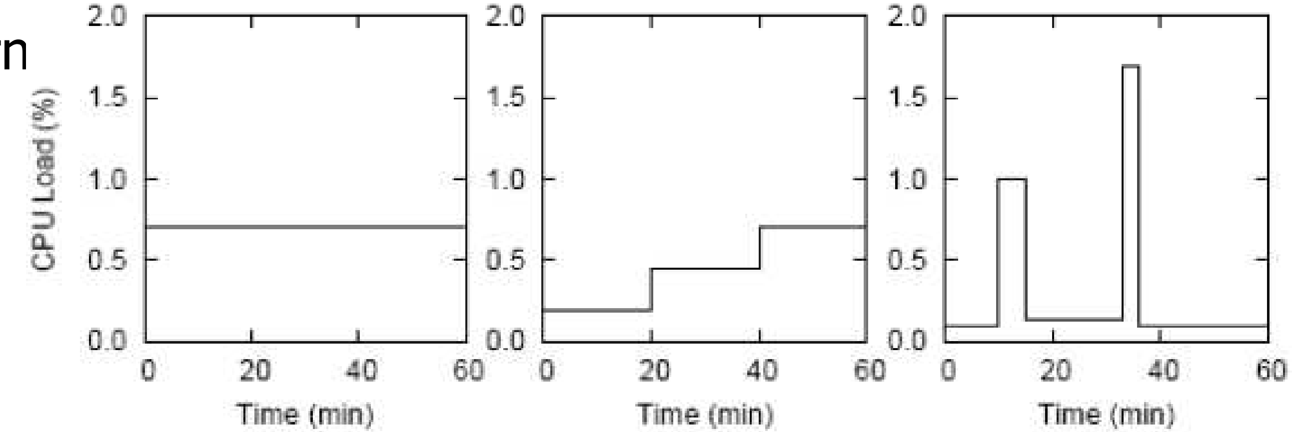
Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, PDS Tech.Rep.2011-009

Experimental Setup (1)

Q3

- Environments
  - DAS4, Florida International University (FIU)
  - Amazon EC2
- Workloads
  - Bottleneck
  - Arrival pattern

Workload Unit	CPU	Memory	I/O	Appears in
WU1	X			WL1
WU2		X		WL2,WL4
WU3			X	WL3,WL4



Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid2012 + PDS Tech.Rep.2011-009

Experimental Setup (2)

Q3

- **Performance Metrics**
  - Traditional: Makespan, Job Slowdown
  - Workload Speedup One (SU1)
  - Workload Slowdown Infinite (SUinf)

$$SU_1(W) = \frac{MS(W)}{\sum_{i \in W} t_R(i)}$$

$$SU_\infty(W) = \frac{MS(W)}{\max_{i \in W} t_R(i)}$$

- **Cost Metrics**
  - Actual Cost (Ca)
  - Charged Cost (Cc)

$$C_a(W) = \sum_{i \in leased\ VMs} t_{stop}(i) - t_{start}(i)$$

$$C_c(W) = \sum_{i \in leased\ VMs} \lceil t_{stop}(i) - t_{start}(i) \rceil$$

- **Compound Metrics**
  - Cost Efficiency (Ceff)
  - Utility

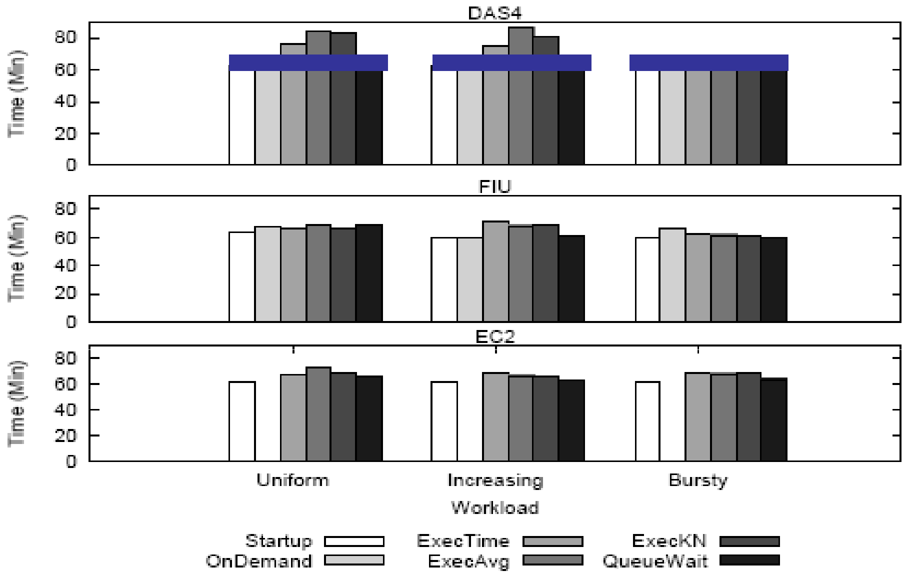
$$C_{eff}(W) = \frac{C_c(W)}{C_a(W)}$$

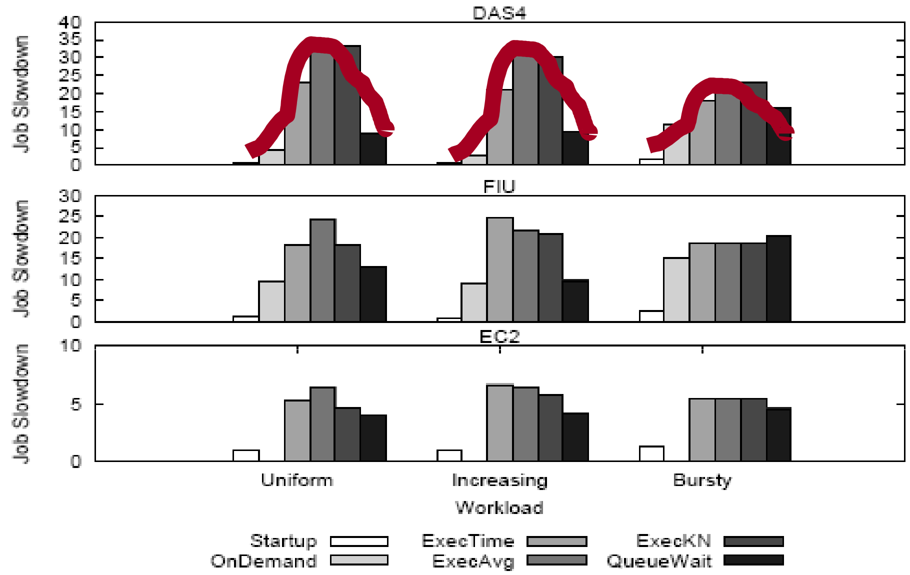
$$U(W) = \frac{SU_1(W)}{C_c(W)}$$

Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid 2012

Performance Metrics

Q3



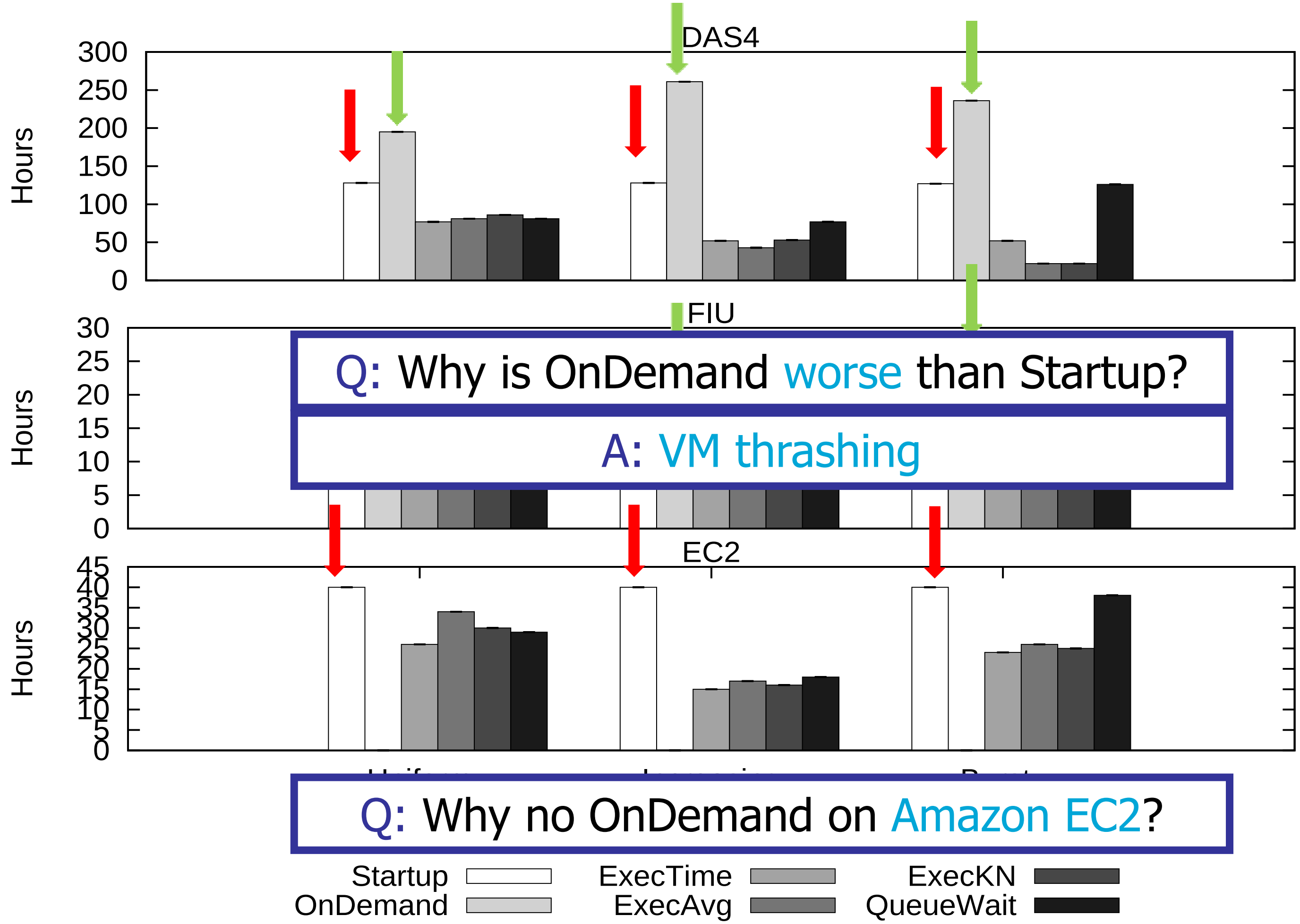


- Makespan very similar
- Very different job slowdown

Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid 2012

Cost Metrics

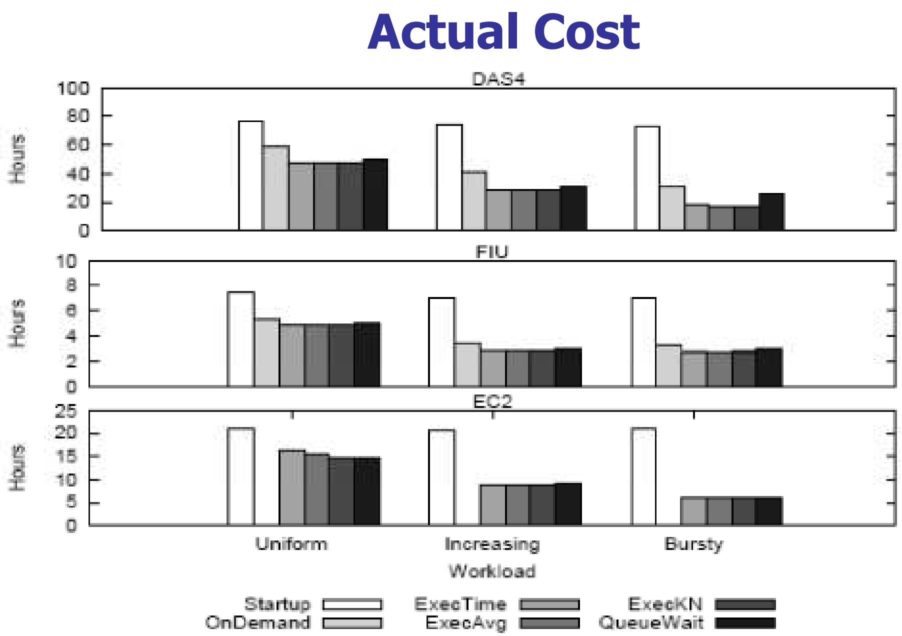
Charged Cost (C_c)

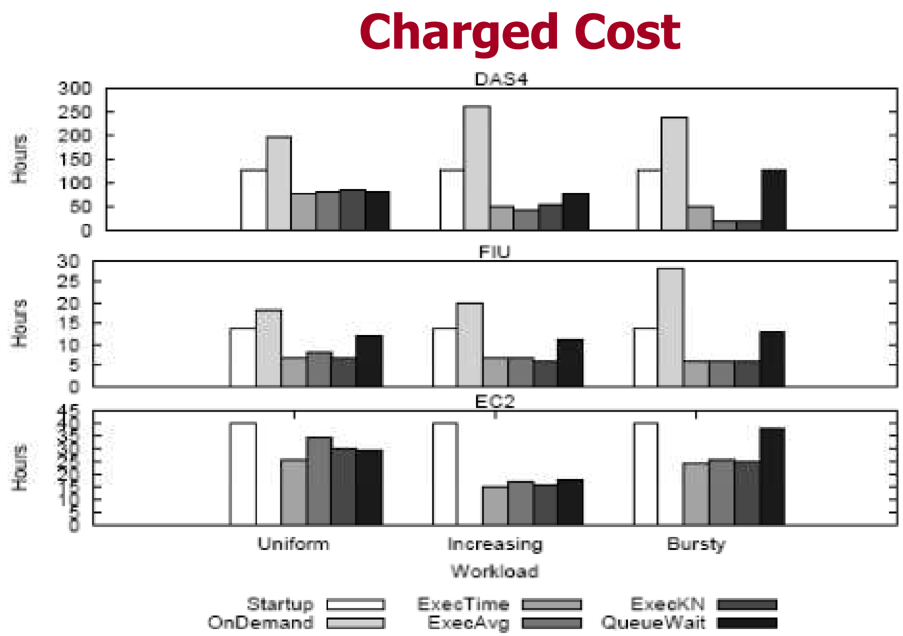


TUDelft
Delft University of Technology

Cost Metrics

Q3



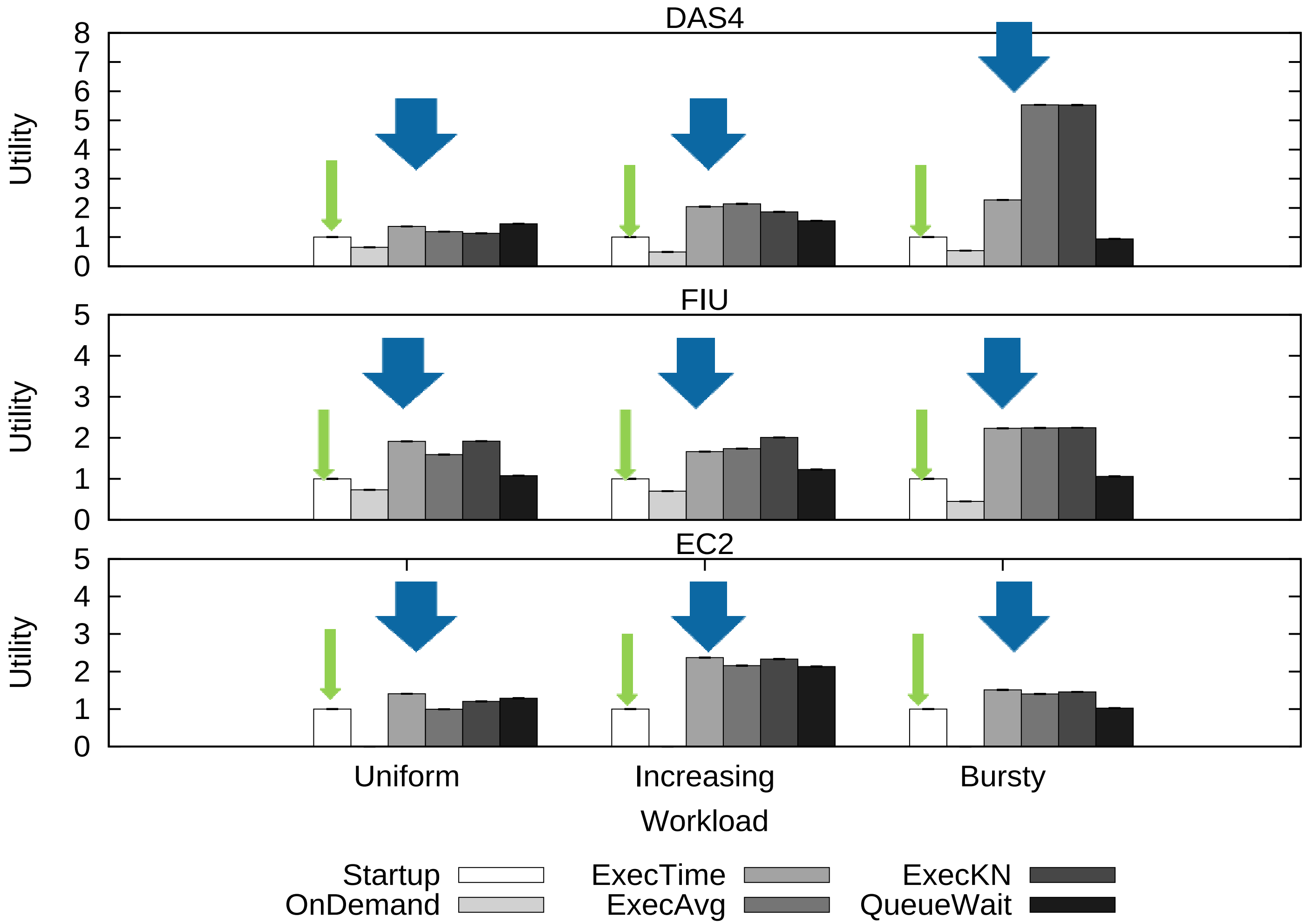


- Very different results between actual and charged
  - Cloud charging function an important selection criterion
- All policies better than Startup in actual cost
- Policies much better/worse than Startup in charged cost

Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid 2012

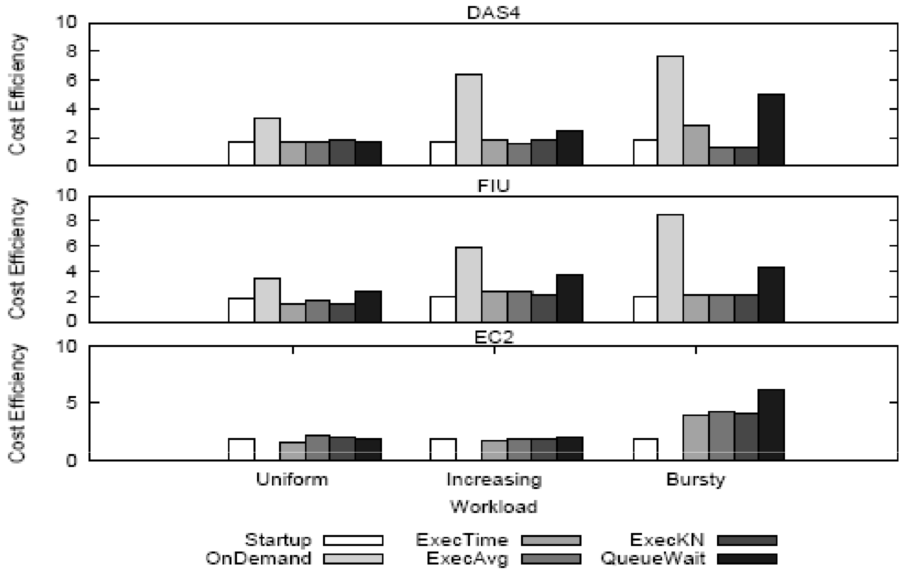
Compound Metrics (Utilities)

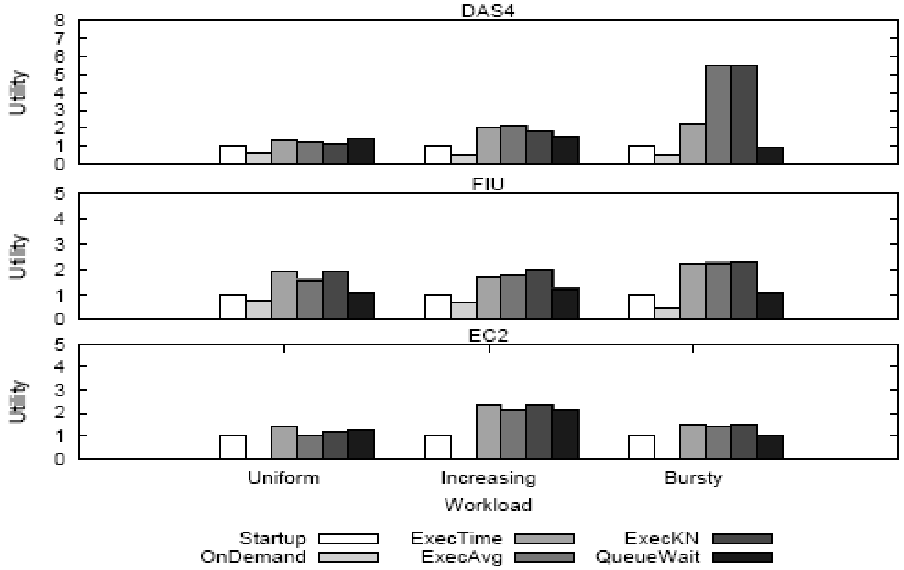
Utility (*U*)



Compound Metrics

Q3





- Trade-off Utility-Cost still needs investigation
- **Performance or Cost, not both:**
  the policies we have studied improve one, but not both

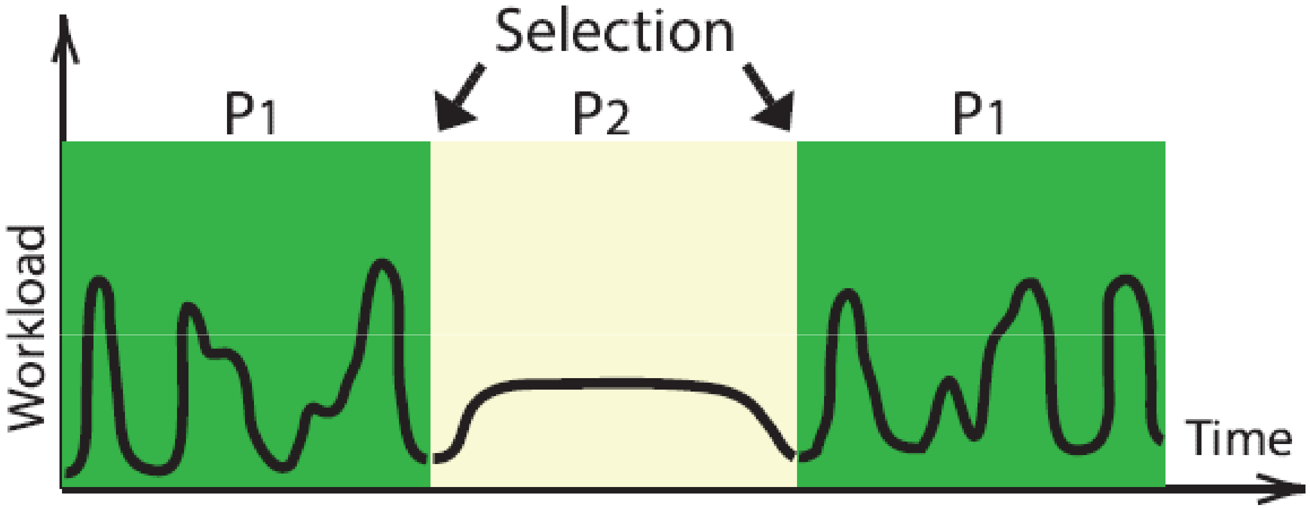
Villegas, Antoniou, Sadjadi, Iosup. An Analysis of Provisioning and Allocation Policies for Infrastructure-as-a-Service Clouds, CCGrid 2012

Why Portfolio Scheduling?

- **Data centers increasingly popular**
  - Constant deployment since mid-1990s
  - Users moving their computation to IaaS clouds
  - Consolidation efforts in mid- and large-scale companies
- **Old scheduling aspects**
  - Hundreds of approaches, each targeting specific conditions—which?
  - No one-size-fits-all policy
- **New scheduling aspects**
  - New workloads
  - New data center architectures
  - New cost models

- **Developing a scheduling policy is risky and ephemeral**
- **Selecting a scheduling policy for your data center is difficult**

What is Portfolio Scheduling? In a Nutshell, for Data Centers



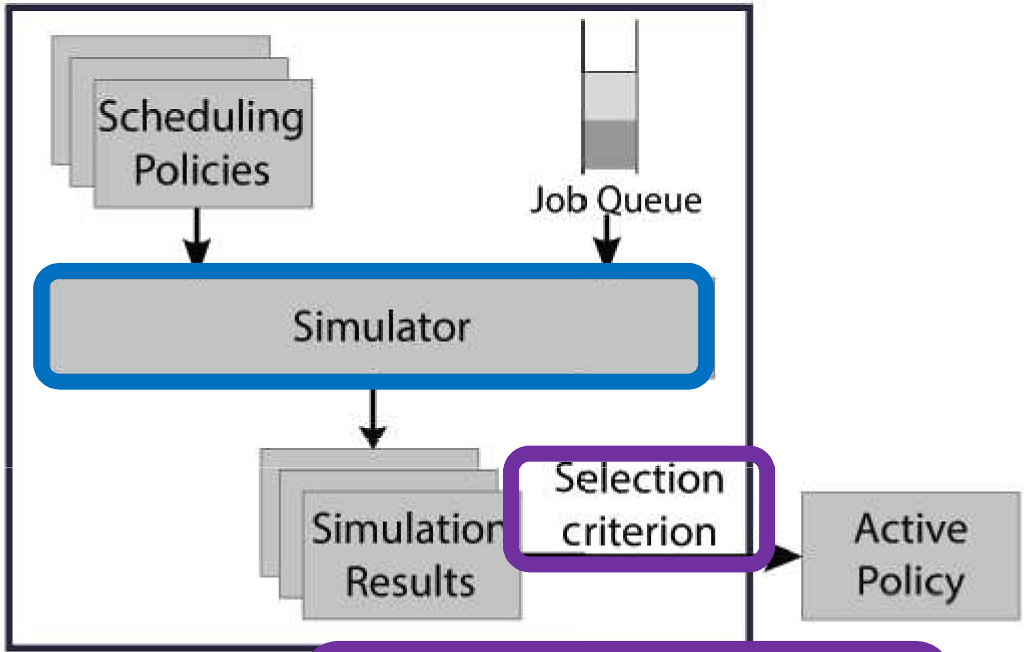
- Create a set of scheduling policies
  - Resource provisioning and allocation policies
- Online selection of the active policy, at important moments
  - Periodic selection, in this work
- Same principle for other changes: pricing model, system, …

Portfolio Scheduling Components
Selection

- Periodic execution

- Simulation-based selection
- Utility function

- Alternatives simulator
  - Expert human knowledge
  - WL sample in real env.
  - Mathematical analysis
- Alternatives utility function
  - Well-known and exotic functions



$$U = \kappa \cdot \left(\frac{R_J}{R_V}\right)^{\alpha} \cdot \left(\frac{1}{S}\right)^{\beta}$$

R_J: Total Runtime of Jobs
R_V: Total Runtime of VMs
S: Slowdown

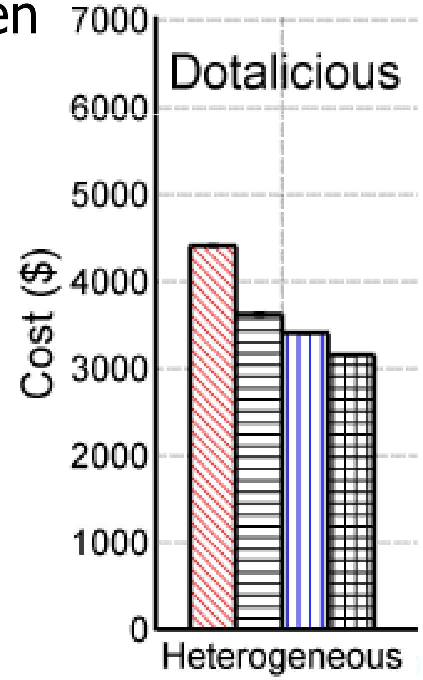
$\alpha=\beta=1$
K=100

Deng, Verboon, Iosup. A Periodic Portfolio Scheduler for Scientific Computing in the Data Center. JSSPP'13.

Agmon Ben-Yehuda, Schuster, Sharov, Silberstein, Iosup. ExPERT: pareto-efficient task replication on grids and a cloud. IPDPS'12.

Portfolio Scheduling for Online Gaming
(also for Scientific Workloads)

- **CoH =** <u>C</u>loud-based, <u>o</u>nline, <u>H</u>ybrid scheduling
  - Intuition: keep rental cost low by finding good mix of machine configurations and billing options
  - Main idea: **portfolio scheduler** = run *both* solver of an Integer Programming Problem and various heuristics, then pick best schedule at deadline
  - Additional feature: Can use **reserved cloud instances**



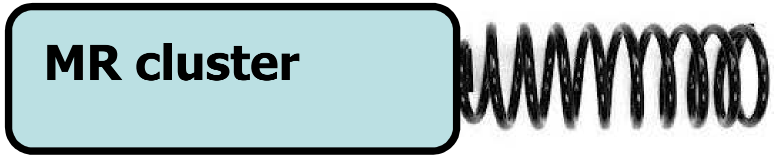
- Promising early results, for **Gaming** (and scientific) workloads

Trace	#jobs	average runtime [s]
Grid5000	200,450	2728
LCG	188,041	8971
DotaLicious	109,251	2231

Shen, Deng, Iosup, and Epema. Scheduling Jobs in the Cloud Using On-demand and Reserved Instances, EuroPar'13.

Ad: Resizing MapReduce Clusters

- Motivation:
  - Performance and data isolation
  - Deployment version and user isolation
  - Capacity planning : efficiency—accuracy trade-off



- Constraints:
  - Data is big and difficult to move
  - Resources need to be released fast

- Approach:
  - Grow / shrink at processing layer
  - Resize based on resource utilization
  - Policies for provisioning and allocation

TU Delft
Delft University of Technology

Ghit and Epema. Resource Management for Dynamic MapReduce Clusters in Multicluster Systems. MTAGS 2012. Best Paper Award.

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

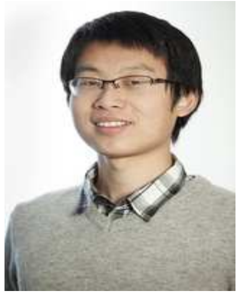
Big Data/Graph Processing: Our Team

Alexandru Iosup
TU Delft

Cloud Computing
Gaming Analytics
Performance Eval.
Benchmarking
Variability

Ana Lucia Varbanescu
UvA

Parallel Computing
Multi-cores/GPUs
Performance Eval.
Benchmarking
Prediction



Yong Guo
TU Delft

Cloud Computing
Gaming Analytics
Performance Eval.
Benchmarking

Marcin Biczak
TU Delft

Cloud Computing
Performance Eval.
Development

Graphitti

http://www.pds.ewi.tudelft.nl/graphitti/

Consultant for the project.
Not responsible for issues related to this work. Not representing official products and/or company views.

Claudio Martella
VU Amsterdam
All things Giraph

Ted Willke
Intel Corp.
All things graph-processing

What I'll Talk About

Q4

How well do graph-processing platforms perform? (Q4)

1. Motivation
2. Previous work
3. Method / Bechmarking suite
4. Experimental setup
5. Selected experimental results
6. Conclusion and ongoing work

Why "How Well do Graph-Processing Platforms Perform?"

Q4

- Large-scale graphs exists in a wide range of areas: social networks, website links, online games, etc.

- Large number of **platforms** available to developers
  - Desktop: Neo4J, SNAP, etc.
  - Distributed: Giraph, GraphLab, etc.
  - Parallel: too many to mention

Q4

Some Previous Work

Graph500.org: BFS on synthetic graphs

Performance evaluation in graph-processing (limited algorithms and graphs)

- Hadoop does not perform well [Warneke09]
- Graph partitioning improves the performance of Hadoop [Kambatla12]
- Trinity outperforms Giraph in BFS [Shao12]
- Comparison of graph databases [Dominguez-Sal10]

Performance comparison in other applications

- Hadoop vs parallel DBMSs: grep, selection, aggregation, and join [Pavlo09]
- Hadoop vs High Performance Computing Cluster (HPCC): queries [Ouaknine12]
- Neo4j vs MySQL: queries [Vicknair10]

Problem: Large differences in performance profiles across different graph-processing **algorithms** and **data sets**

Guo, Biczak, Varbanescu, Iosup, Martella, Willke.
How Well do Graph-Processing Platforms Perform?
An Empirical Performance Evaluation and Analysis

Graphitti

Q4

Our Method

A benchmark suite for
performance evaluation of graph-processing platforms

1. Multiple Metrics, e.g.,
   - Execution time
   - Normalized: EPS, VPS
   - Utilization

2. Representative graphs with various characteristics, e.g.,
   - Size
   - Directivity
   - Density

3. Typical graph algorithms, e.g.,
   - BFS
   - Connected components

Benchmarking suite
Data sets

Q4

Graphs	# V	# E	d ($\times 10^{-5}$)	$\bar{\text{D}}$	Size	Directivity
Amazon	262.1 K	1.2 M	1.8	4.7	18 MB	directed
WikiTalk	2.4 M	5.0 M	0.1	2.1	87 MB	directed
KGS	293.3 K	16.6 M	38.5	112.9	210 MB	undirected
Citation	3.8 M	16.5 M	0.1	4.4	297 MB	directed
DotaLeague	61.2 K	50.9 M	2,719.0	1,663.2	655 MB	undirected
Synth	2.4 M	64.2 M	2.2	53.6	964 MB	undirected
Friendster	65.6 M	1.8 B	0.1	55.1	31 GB	undirected

SNAP

Graph500
http://www.graph500.org/

The Game Trace Archive
http://gta.st.ewi.tudelft.nl/

Guo, Biczak, Varbanescu, Iosup, Martella, Willke.
How Well do Graph-Processing Platforms Perform?
An Empirical Performance Evaluation and Analysis

Graphitti

Q4

Benchmarking Suite
Algorithm classes

1. General Statistics (STATS: # vertices and edges, LCC)
2. Breadth First Search (BFS)
3. Connected Component (CONN)
4. Community Detection (COMM)
5. Graph Evolution (EVO)

Q4

Benchmarking suite
Platforms and Process

- Platforms

YARN

Giraph

- Process
    - Evaluate baseline (out of the box) and tuned performance
    - Evaluate performance on fixed-size system
    - Future: evaluate performance on elastic-size system
    - Evaluate scalability

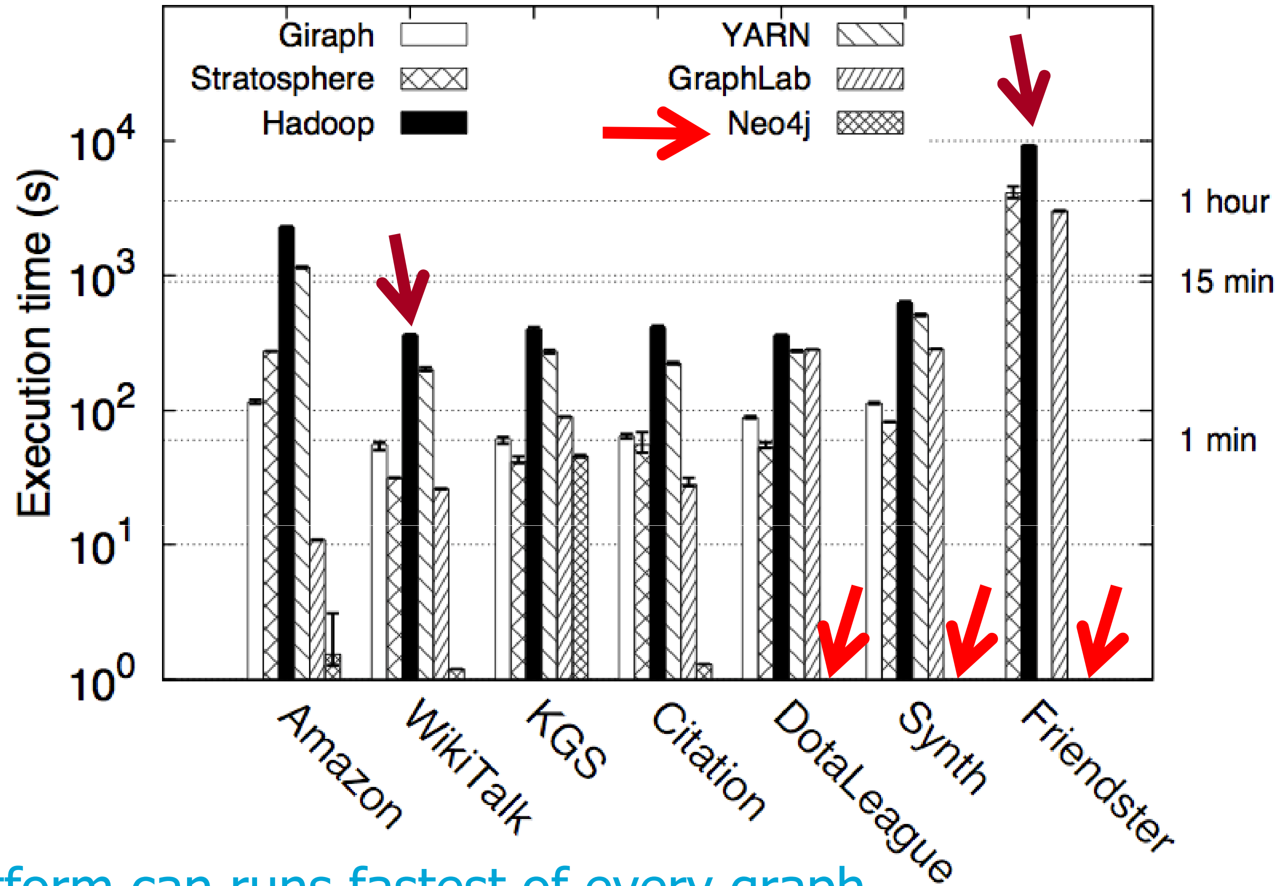
Guo, Biczak, Varbanescu, Iosup, Martella, Willke.
How Well do Graph-Processing Platforms Perform?
An Empirical Performance Evaluation and Analysis

Experimental setup

- Size
  - Most experiments take 20 working nodes
  - Up to 50 working nodes
- DAS4: a multi-cluster Dutch grid/cloud
  - Intel Xeon 2.4 GHz CPU (dual quad-core, 12 MB cache)
  - Memory 24 GB
  - 10 Gbit/s Infiniband network and 1 Gbit/s Ethernet network
  - Utilization monitoring: Ganglia
- HDFS used here as distributed file systems

BFS: results for all platforms, all data sets

Q4



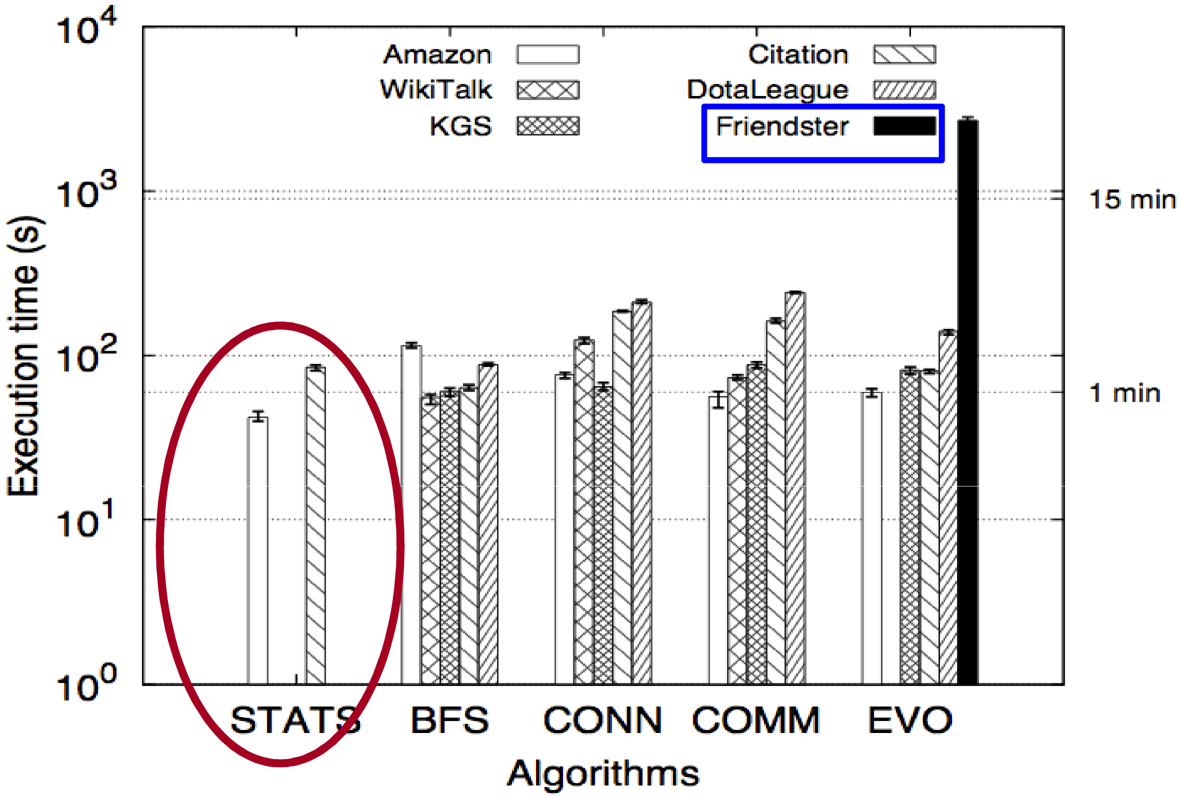
- No platform can runs fastest of every graph
- Not all platforms can process all graphs
- Hadoop is the worst performer

Guo, Biczak, Varbanescu, Iosup, Martella, Willke. How Well do Graph-Processing Platforms Perform? An Empirical Performance Evaluation and Analysis

Graphitti

Giraph: results for all algorithms, all data sets

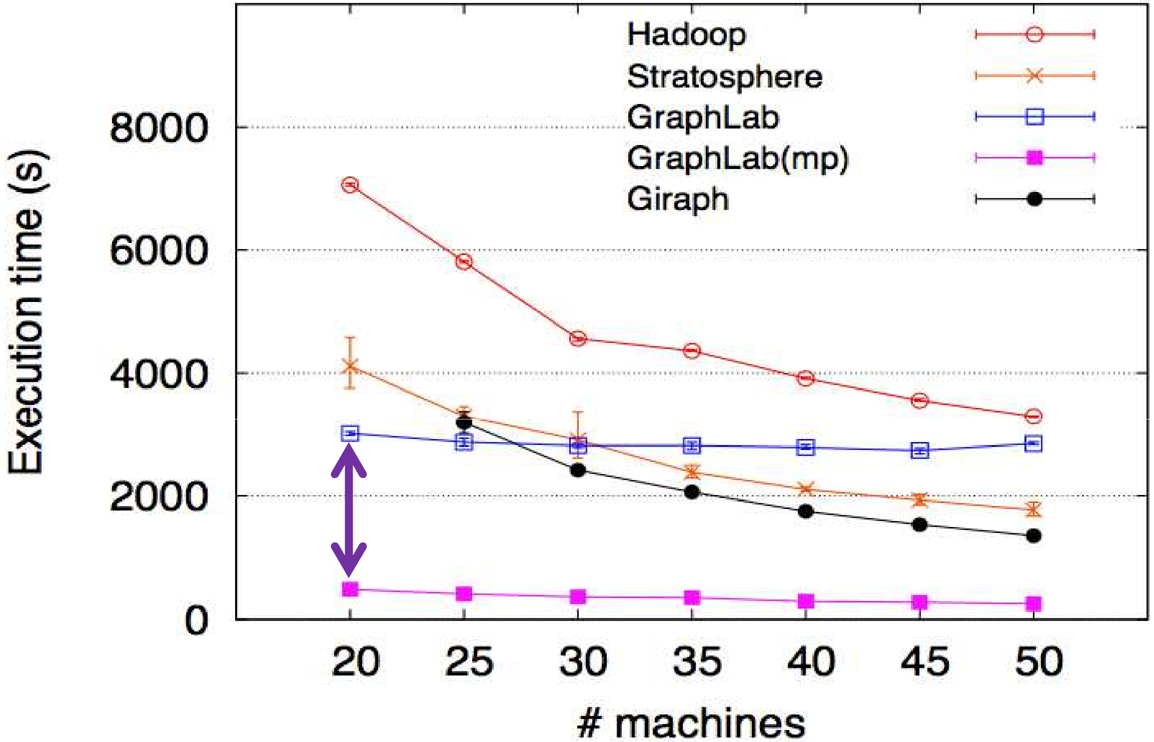
Q4



- Storing the whole graph in memory helps Giraph perform well
- Giraph may crash when graphs or messages become larger

Horizontal scalability: BFS on Friendster (31 GB)

Q4



- Using more computing machines can reduce execution time
- Tuning needed for horizontal scalability, e.g., for GraphLab, split large input files into number of chunks equal to the number of machines

Guo, Biczak, Varbanescu, Iosup, Martella, Willke. How Well do Graph-Processing Platforms Perform? An Empirical Performance Evaluation and Analysis

Graphitti

Q4

Additional Overheads
Data ingestion time

- Data ingestion
  - Batch system: one ingestion, multiple processing
  - Transactional system: one ingestion, one processing

- Data ingestion matters even for batch systems

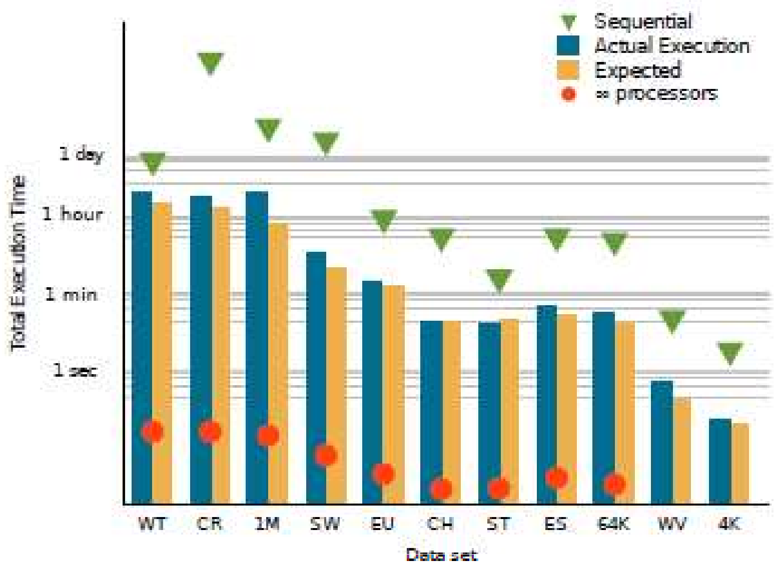
	Amazon	DotaLeague	Friendster
HDFS	1 second	7 seconds	5 minutes
Neo4J	4 hours	6 **days**	**n/a**

Guo, Biczak, Varbanescu, Iosup, Martella, Willke.
How Well do Graph-Processing Platforms Perform?
An Empirical Performance Evaluation and Analysis

Graphitti

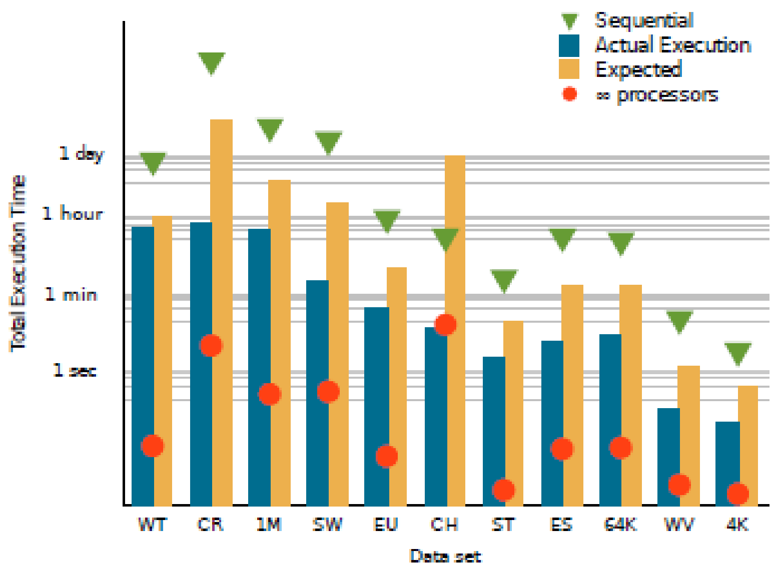
GPUs vs CPUs: All-Pairs Shortest Path

Pender and Varbanescu. MSc thesis at TU Delft. Jun 2012. TU Delft Library, http://library.tudelft.nl .

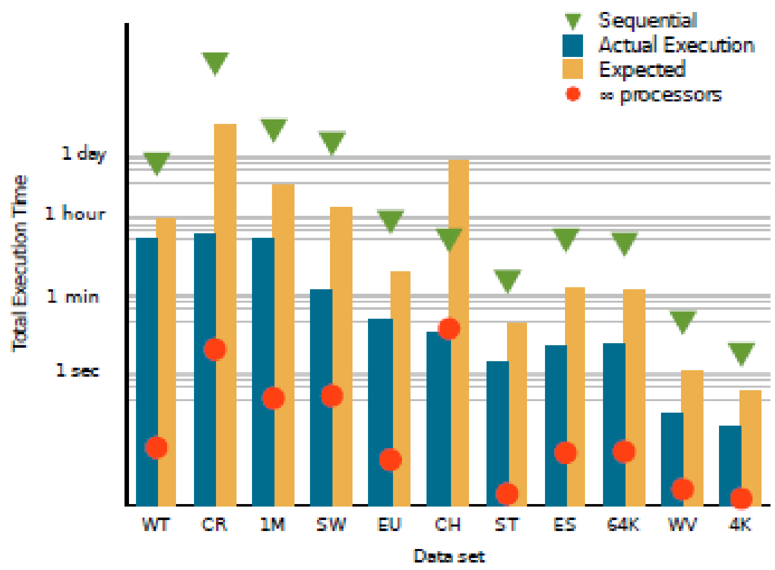


(a) Intel Xeon E5620

	Dataset
WT	Wikipedia Talk Network
CR	California Road Network
1M	Graph 1M
SW	Stanford Web Graph
EU	EU Email Communication Network
CH	Chain 100K
ST	Star 100K
ES	Epinions Social Network
64K	Graph 64K
WV	Wikipedia Vote
4K	Graph 4K



(c) Nvidia Tesla C2050/ C2070

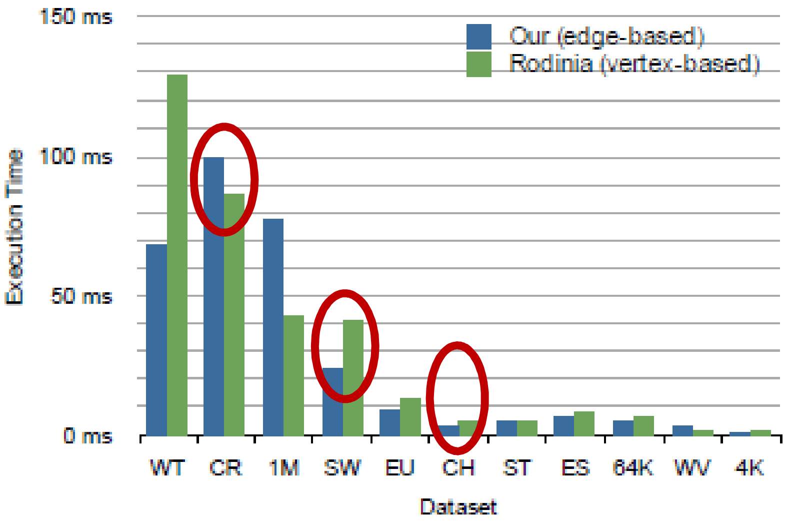


(d) Nvidia GeForce GTX480

Delft University of Te

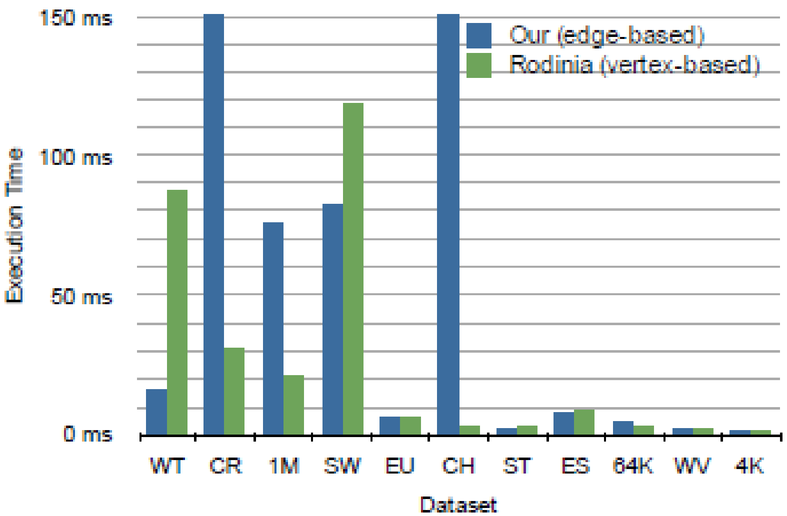
GPUs vs CPUs: BFS vs Data Format, E/V-based

Pender and Varbanescu. MSc thesis at TU Delft. Jun 2012. TU Delft Library, http://library.tudelft.nl .

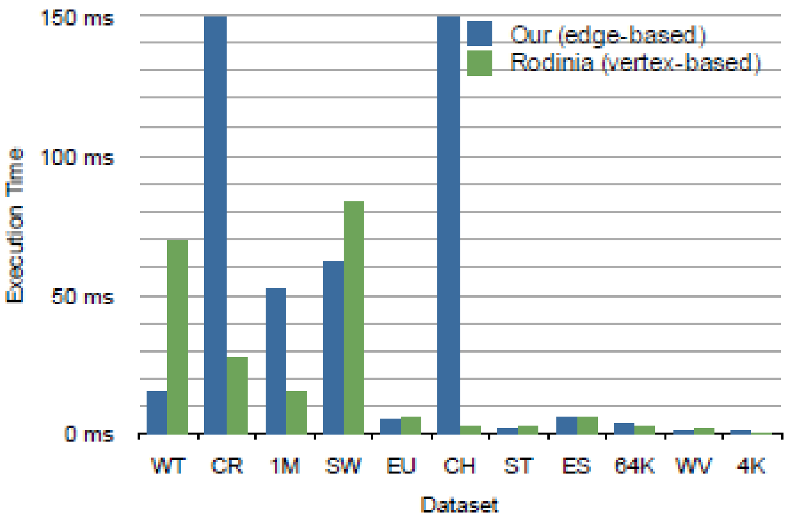


(a) Intel Xeon E5620

	Dataset
WT	Wikipedia Talk Network
CR	California Road Network
1M	Graph 1M
SW	Stanford Web Graph
EU	EU Email Communication Network
CH	Chain 100K
ST	Star 100K
ES	Epinions Social Network
64K	Graph 64K
WV	Wikipedia Vote
4K	Graph 4K



(c) Nvidia Tesla C2050/ C2070



(d) Nvidia GeForce GTX480

Conclusion and ongoing work

Q4

- Performance is f(Data set, Algorithm, Platform, Deployment)
- Cannot tell yet which of (Data set, Algorithm, Platform) the most important (also depends on Platform)
- Platforms have their own drawbacks
- Some platforms can scale up reasonably with cluster size (horizontally) or number of cores (vertically)

- Ongoing work
  - *Benchmarking* suite
  - Build a performance boundary model
  - Explore performance variability

http://bit.ly/10hYdIU

Guo, Biczak, Varbanescu, Iosup, Martella, Willke.
How Well do Graph-Processing Platforms Perform?
An Empirical Performance Evaluation and Analysis

Graphitti

Agenda

1. An Introduction to IaaS Cloud Comput
2. Research Questions or Why We Need B
3. A General Approach and Its Main Challe
4. **IaaS Cloud Workloads (Q0)**
5. **IaaS Cloud Performance (Q1) & Perf. Variability (Q2)**
6. **Provisioning & Allocation Policies for IaaS Clouds (Q3)**
7. **Big Data: Large-Scale Graph Processing (Q4)**
8. Conclusion

Agenda

1. An Introduction to IaaS Cloud Computing
2. Research Questions or Why We Need Benchmarking?
3. A General Approach and Its Main Challenges
4. IaaS Cloud Workloads (Q0)
5. IaaS Cloud Performance (Q1) and Perf. Variability (Q2)
6. Provisioning and Allocation Policies for IaaS Clouds (Q3)
7. **Conclusion**

~~Conclusion~~ Take-Home Message

- **IaaS cloud benchmarking: approach + 10 challenges**

- **Put 10-15% project effort in benchmarking = understanding how IaaS clouds really work**
  - Q0: Statistical workload models
  - Q1/Q2: Performance/variability
  - Q3: Provisioning and allocation
  - Q4: Big Data, Graph processing

- **Tools and Workload Models**
  - SkyMark
  - MapReduce
  - Graph processing benchmarking suite

http://www.flickr.com/photos/dimitrisotiropoulos/4204766418/

Thank you for your attention! Questions? Suggestions? Observations?

More Info:

HPDC 2013

- http://www.st.ewi.tudelft.nl/~iosup/research.html
- http://www.st.ewi.tudelft.nl/~iosup/research_cloud.html
- http://www.pds.ewi.tudelft.nl/

Do not hesitate to contact me…

Alexandru Iosup

A.Iosup@tudelft.nl
http://www.pds.ewi.tudelft.nl/~iosup/ (or google “iosup”)
Parallel and Distributed Systems Group
Delft University of Technology

WARNING: Ads

www.pds.ewi.tudelft.nl/ccgrid2013

Dick Epema, General Chair
Delft University of Technology Delft

Thomas Fahringer, PC Chair
University of Innsbruck

Delft, the Netherlands
May 13-16, 2013

Paper submission deadline:
November 22, 2012

If you have an interest in novel aspects of performance, you should join the SPEC RG

- **Find a new venue to discuss your work**
  - Exchange with experts on how the performance of systems can be measured and engineered
  - Find out about novel methods and current trends in performance engineering
  - Get in contact with leading organizations in the field of performance evaluation
- Find a new group of potential employees
- Join a SPEC standardization process
- **Performance in a broad sense:**
  - *Classical performance metrics:* Response time, throughput, scalability, resource/cost/energy, efficiency, elasticity
  - *Plus dependability in general*: Availability, reliability, and security