ICPP 2015 – Beijing, China-September 1-4

Resource Management in Cloud Computing Systems

FACULTY OF
ENGINEERING &
INFORMATION
TECHNOLOGIES

Albert Y. Zomaya | Professor and Director
Centre for Distributed and High Performance Computing
School of Information Technologies

THE UNIVERSITY OF SYDNEY

Centre for Distributed and High Performance Computing

› A 40+ member group. Past and current funding from the Australian Research Council, CISCO, ERICSSON, IBM, Microsoft, Sun, Smart Internet CRC, NICTA, DSTO and CSIRO.

› The Centre's mission is to establish **a streamlined research, technology exploration and advanced training program**. It will be a leading centre to undertake collaborative multi-disciplinary research in support of *distributed* and *high performance computing* and related industry to enable advances in information technology and other application domains.

› The Centre focuses currently on several themes which build on existing strengths at Sydney University:

- **Algorithmics and Data Mining**
- **Cloud Computing and Green ICT**
- **Internetworking**
- **Service Computing**
- **Distributed Computing Applications**

Outline

- **Resource Abundance in Clouds**
  - Source of inefficiency or opportunity of efficiency?
  - Inefficiency of current practices in resource management
- **Holistic Approach to Optimization of Cloud Efficiency**
  - Data center level efficiency
  - Individual node/resource level efficiency
  - Capturing trade-off between cost and performance
- **Conclusion**

Take Home Message

Source: http://www.flickr.com/photos/56104473@N04/5190273185/sizes/l/in/photostream/

› There is a need for different usage/application models for cloud computing environments

› Resource allocation in clouds involves a number of very complex issues that will be around for some time

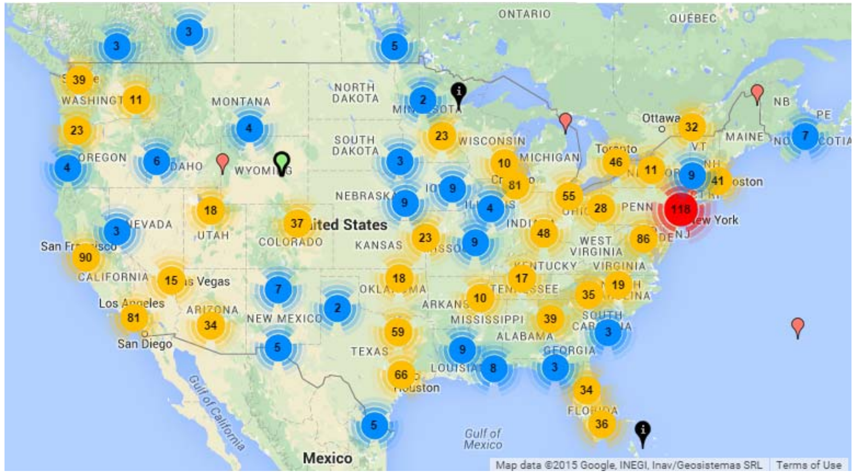
› A fertile research area with many directions

Gartner's Strategic Technology Trends for 2015

‘Efficiency’ of Resource Abundant Clouds

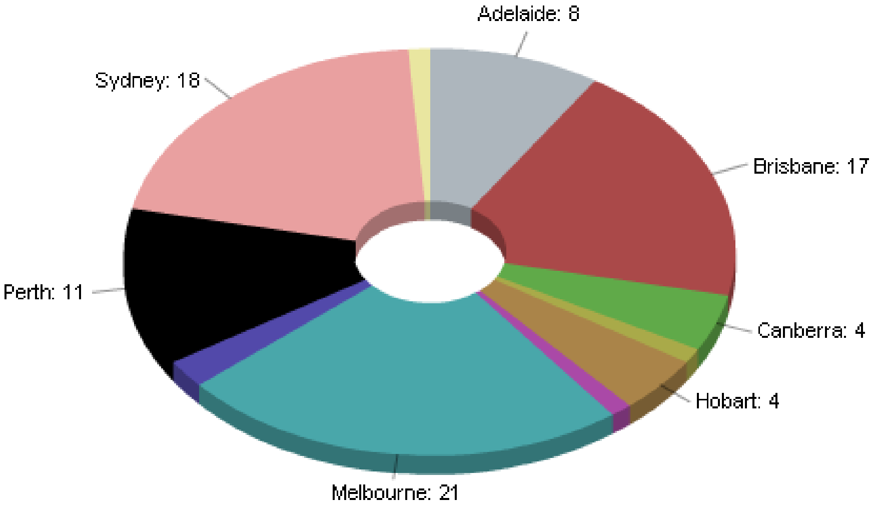
- Resource Efficiency ≠ Resource Utilization
- Definition of ‘Efficiency’
  - Minimum resource provisioning level
  - Maximum resource utilization
  - Meeting performance requirements (or high performance/throughput)
- Resource utilization
  - The number of active resources over time (system/data-center level)
  - The actual resource usage (e.g., CPU utilization)
- ***We aim to identify the minimum level of resource provisioning that maximizes resource utilization meeting performance requirements***

Efficiency of **‘Resource Abundant Clouds’**

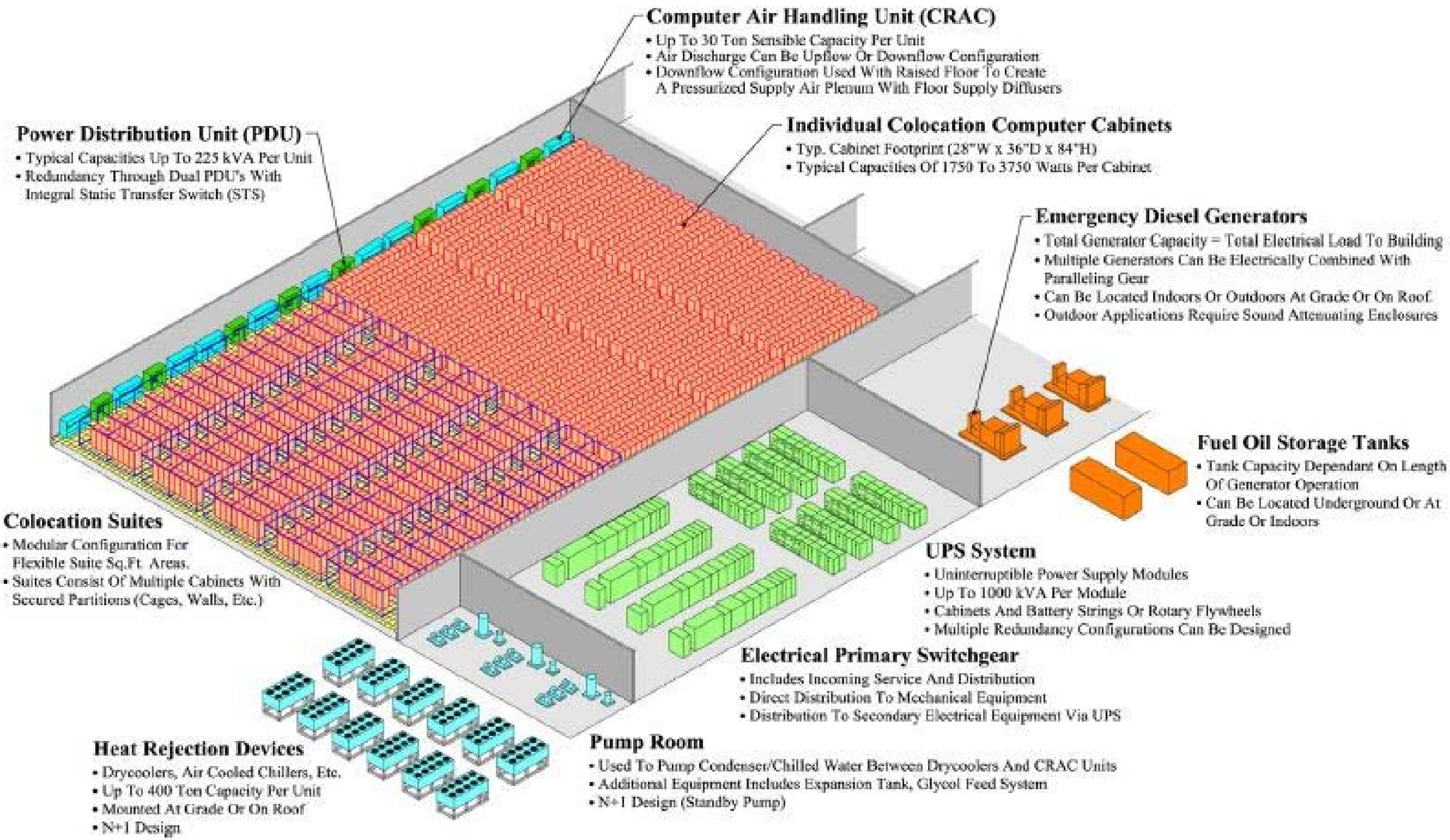


The National Security Administration (NSA) data center

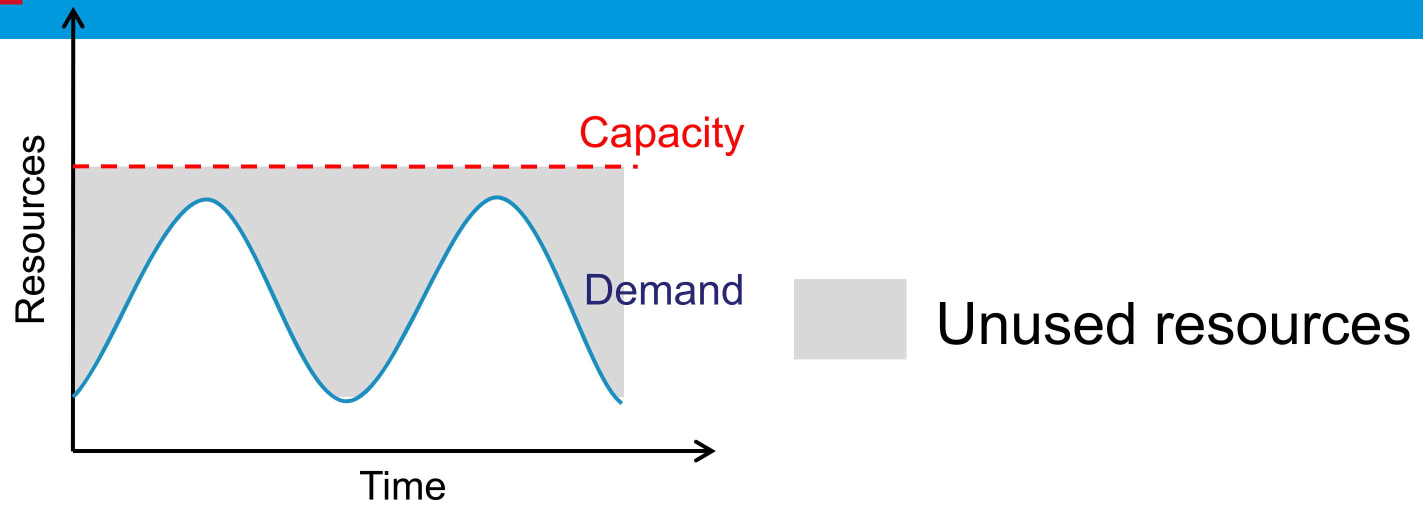
Google data center



Efficiency of **'Resource Abundant Clouds'**



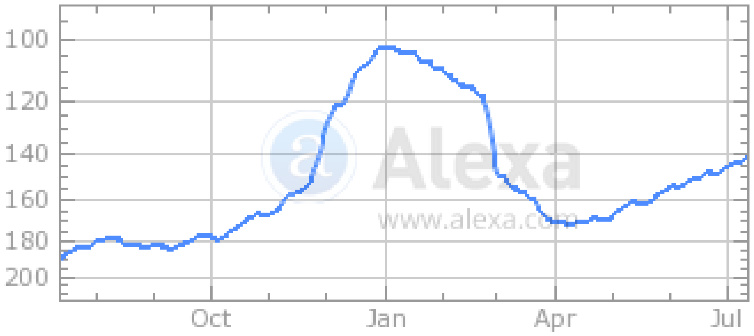
Inefficiency of Current Practices: Data Center Level



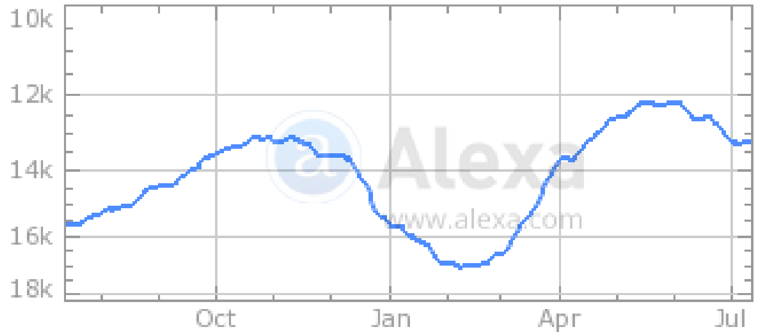
- Data center utilization is mostly below 10% [1] due to over-provisioning
- Idle servers still consume more than 50% of peak power draw [2]
- Average lifespan of servers is 3 years
- Energy costs are soaring
- Public cloud services are often charged by resource hours (partial hours are a source of cost inefficiency)

[1] Barroso, L. and Holzle, U. “The case for energy-proportional computing”, IEEE Computer, 40(12), pp. 33-37, 2007.
[2] Koomey, J. G. “Estimating total power consumption by servers in the U.S. and the world”, Lawrence Berkeley National Laboratory, Stanford University, 2007.

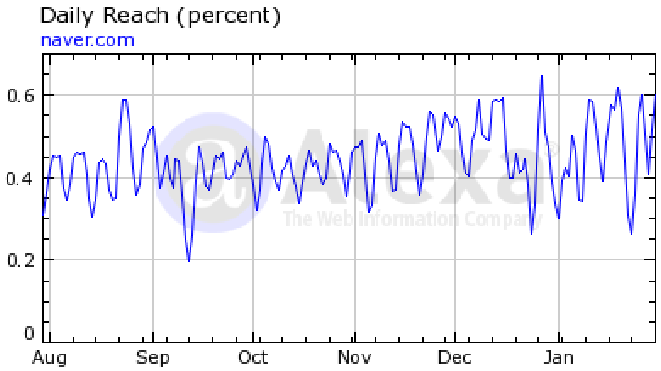
Inefficiency of Current Practices: Data Center Level



walmart.com

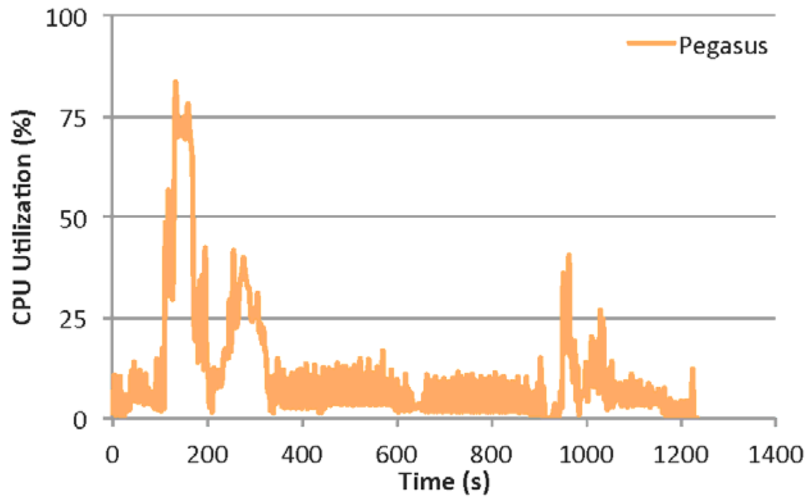


sydney.edu.au

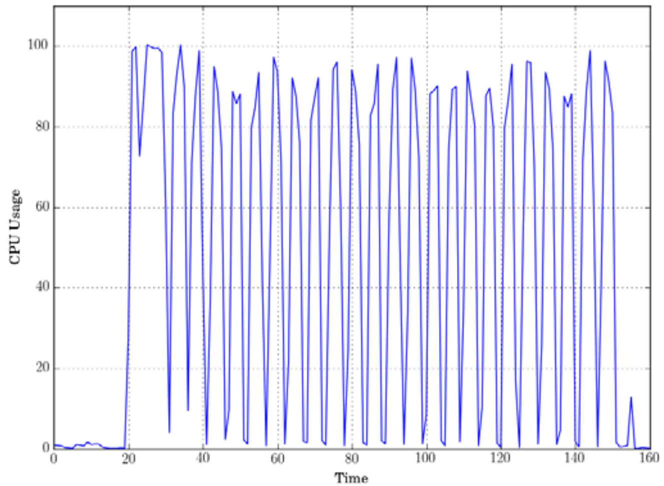


naver.com

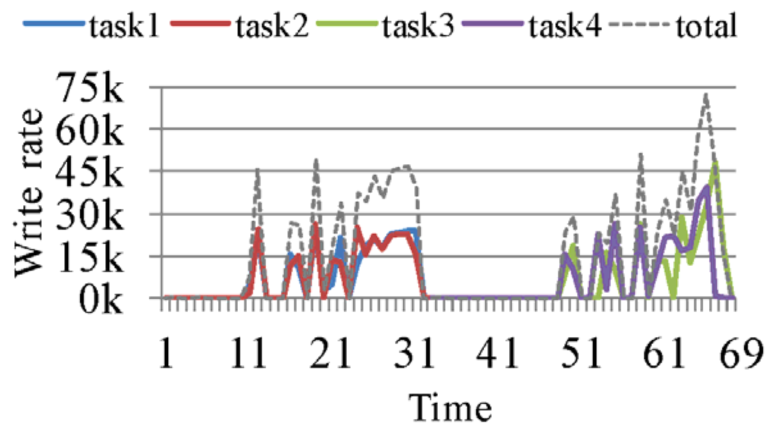
Inefficiency of Current Practices: Individual Resource Level



CPU utilization of scientific workflow

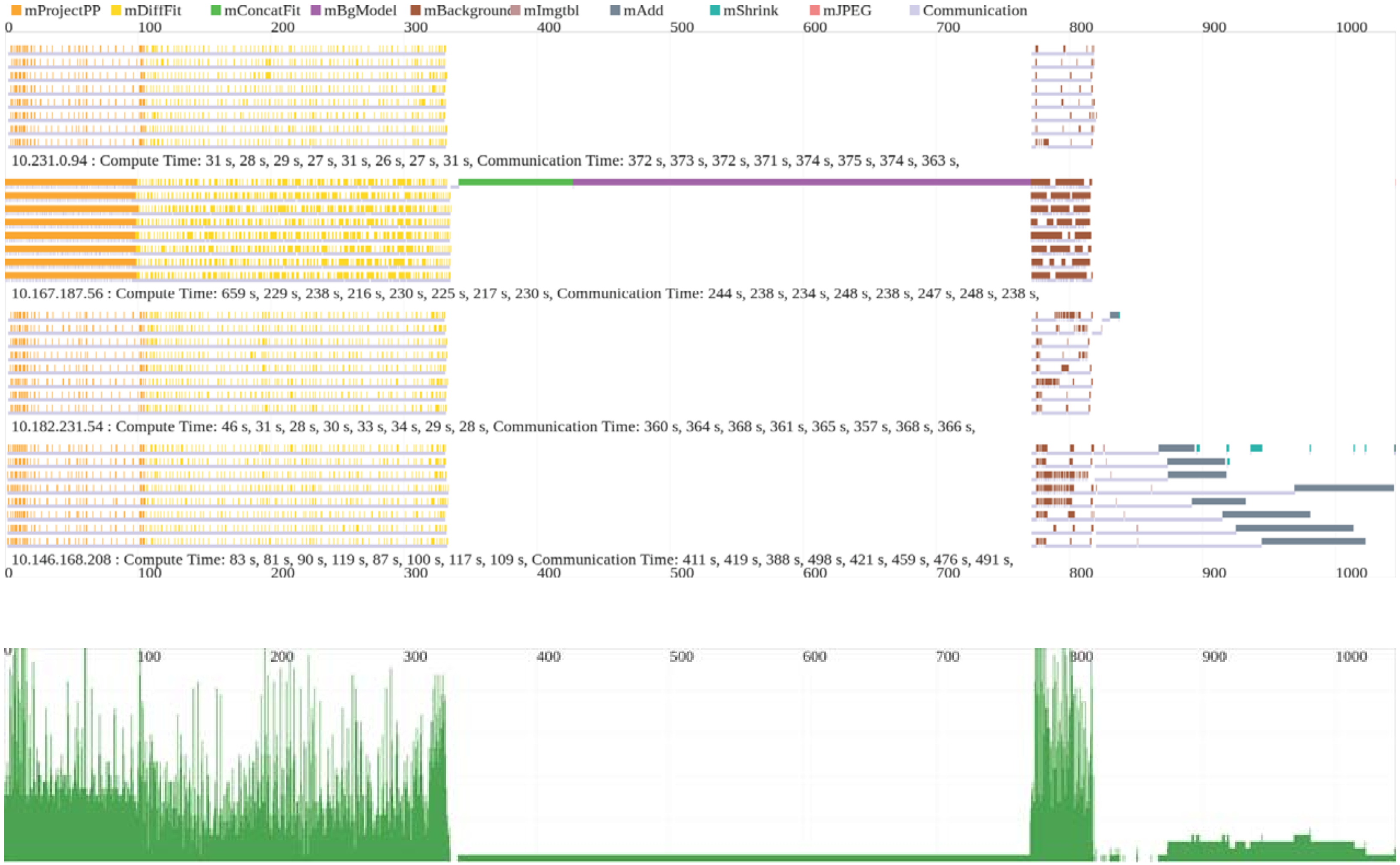


CPU utilization of MapReduce job



Write rate (I/O resource usage) of MapReduce job

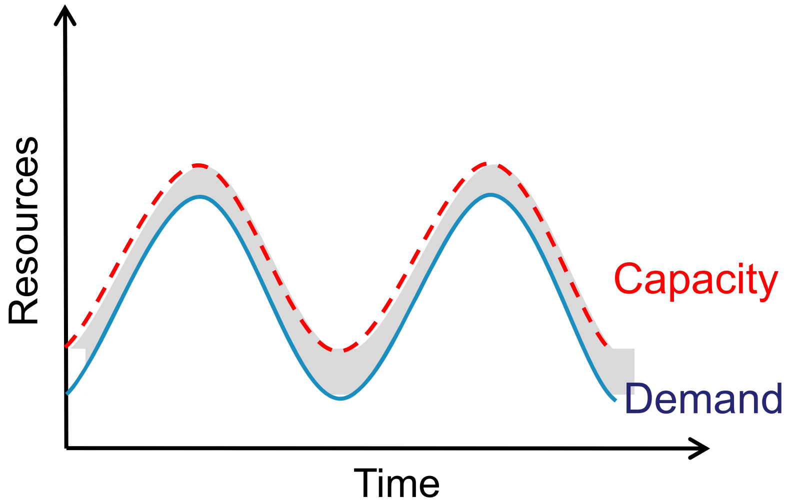
Inefficiency of Current Practices: individual resource level



Visualization of executing Montage astronomical scientific workflow

Ways to Improve Efficiency: Data Center Level

› Dynamic, adaptive resource provisioning by exploiting **elasticity** in the cloud



Real data center
in the cloud

Source: Energy Efficiency and Cloud Computing by D. Patterson in Microsoft Research Faculty Summit 2009

Optimizing Clouds

Source: http://www.flickr.com/photos/ibm_media/2071286721/

Optimizing the Efficiency of Clouds:
Our Solutions

› Resource Efficient Workflow Scheduling

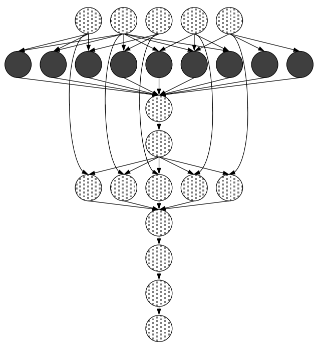
- Lee, Y. C. and Zomaya, A. Y., “Stretch Out and Compact: Workflow Scheduling with Resource Abundance,” in the ***Proceedings of the International Symposium on Cluster Cloud and the Grid (CCGRID)***, May 13-16, 2013.
- Lee, Y. C., Han, H. and Zomaya, A. Y., “On Resource Efficiency of Workflow Schedules,” in the ***Proceedings of the International Conference on Computational Science (ICCS)***, Jun. 10-12, 2014.
- Jiang, Q., Lee, Y. C. and Zomaya, A. Y., “Executing Large Scale Scientific Workflow Ensembles in Public Clouds,” in the ***Proceedings of the International Conference on Parallel Processing (ICPP)***, Sep 1-4, 2015.

› High Performance/Throughput Computing Applications

- HosseinyFarahabady, M.R., Lee, Y.C., Han, H., Zomaya, A.Y., “Randomized Approximation Scheme for Resource Allocation in Hybrid-Cloud Environment,” ***The Journal of Supercomputing*** 69(2): 576-592, 2014.
- Farahabady, M. H., Lee, Y. C. and Zomaya, A. Y., “Pareto-Optimal Cloud Bursting,” ***IEEE Transactions on Parallel and Distributed Systems*** , 25(10): 2670-2682, 2014.

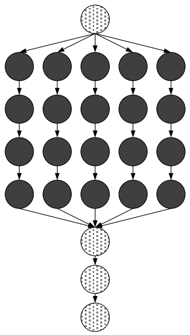
Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

- Many applications in science and engineering are becoming increasingly large-scale and complex
- These applications are often amalgamated in the form of workflows

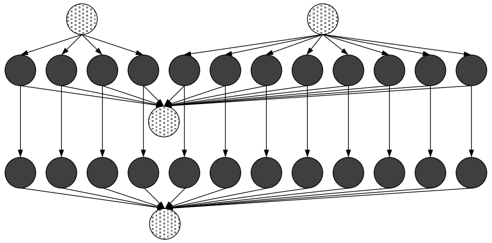


Montage:

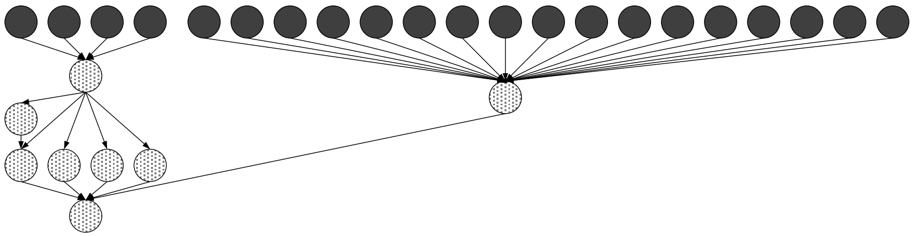
astronomical image mosaic engine



Epigenomics:
genome sequence processing



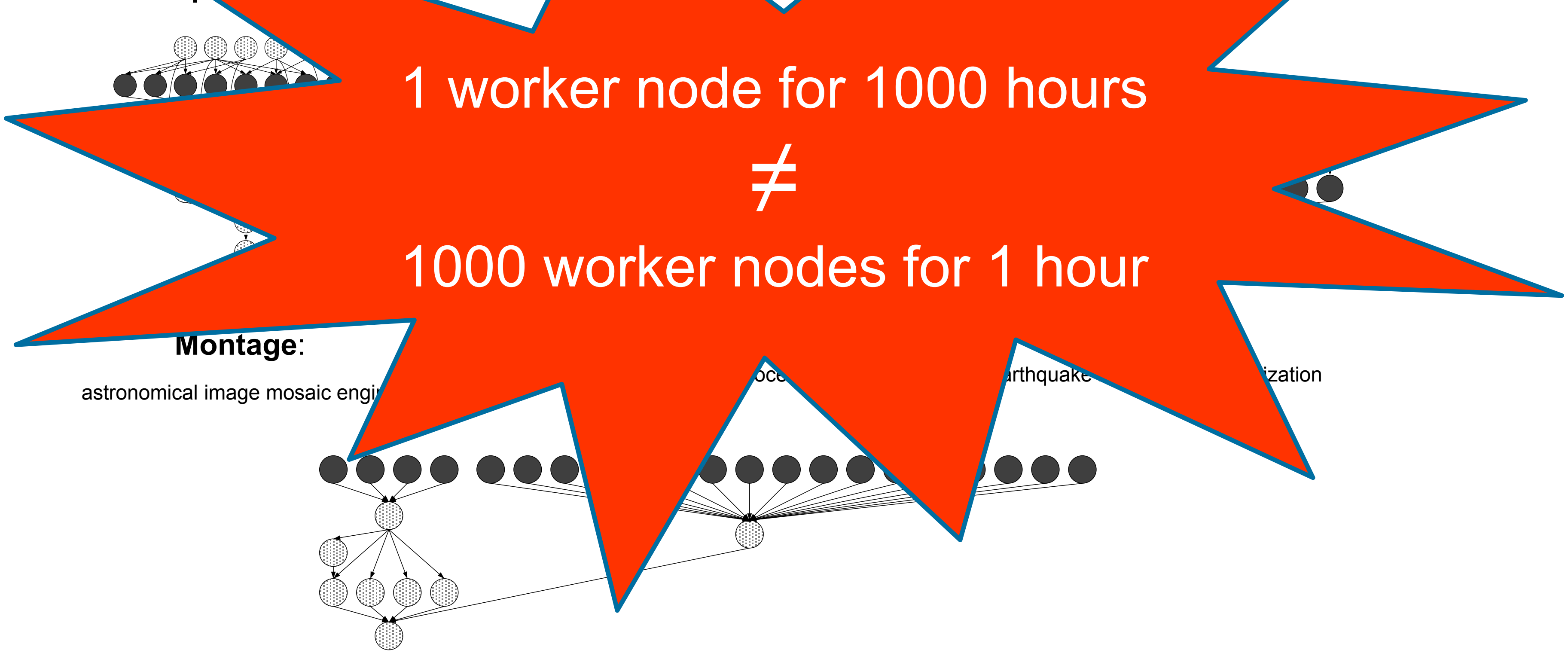
CyberShake:
earthquake hazards characterization



SIPHT:
Search for untranslated RNAs (sRNAs)

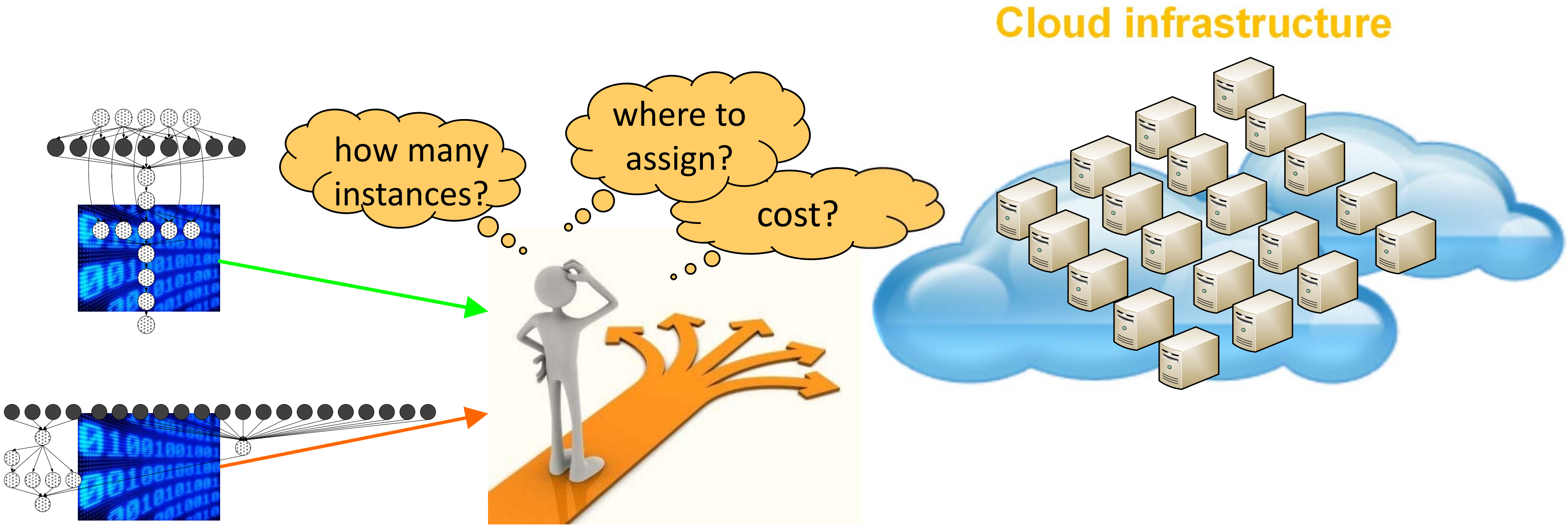
Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

- Many applications in science and engineering a… becoming increasingly la…ge-scale and complex
- These a…ns are ofte… …ama… …form of w… …ows

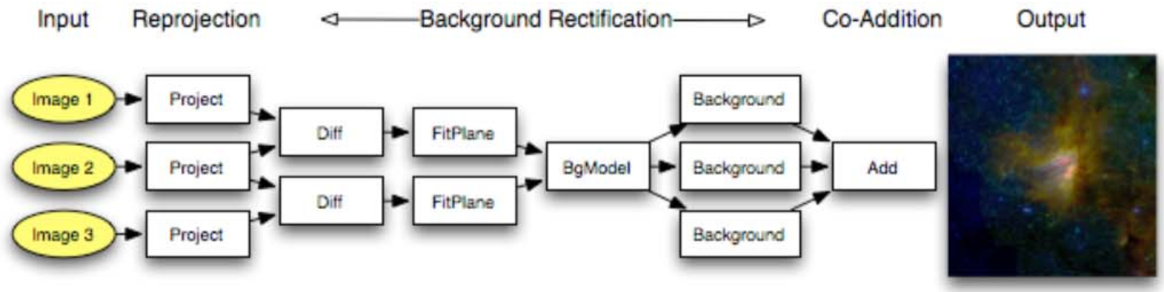


SIPHT:
Search for untranslated RNAs (sRNAs)

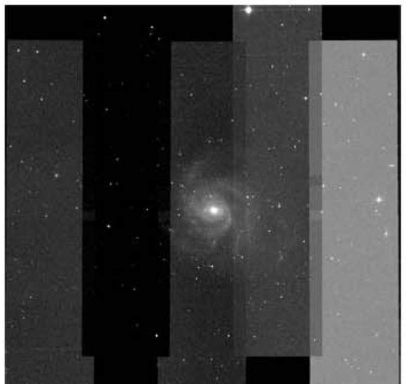
› Resource allocation and scheduling with abundant resources



- Running scientific workflows
  - Montage: an astronomical image mosaic engine
    - stitches together multiple input images to create custom mosaics of the sky
    - A 6.0 Degree Montage workflow contains 8,596 jobs, 1,444 input files with a total size of 4.0 GB and 22,850 intermediate files with a total size of 35GB.



Uncorrected m101 mosaic

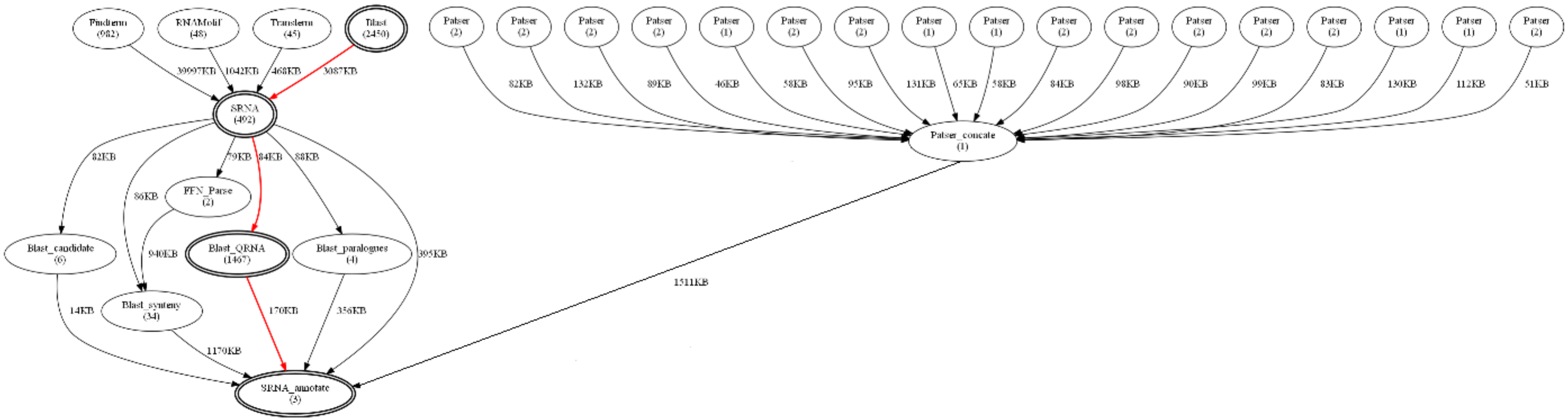


Corrected m101 mosaic

Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

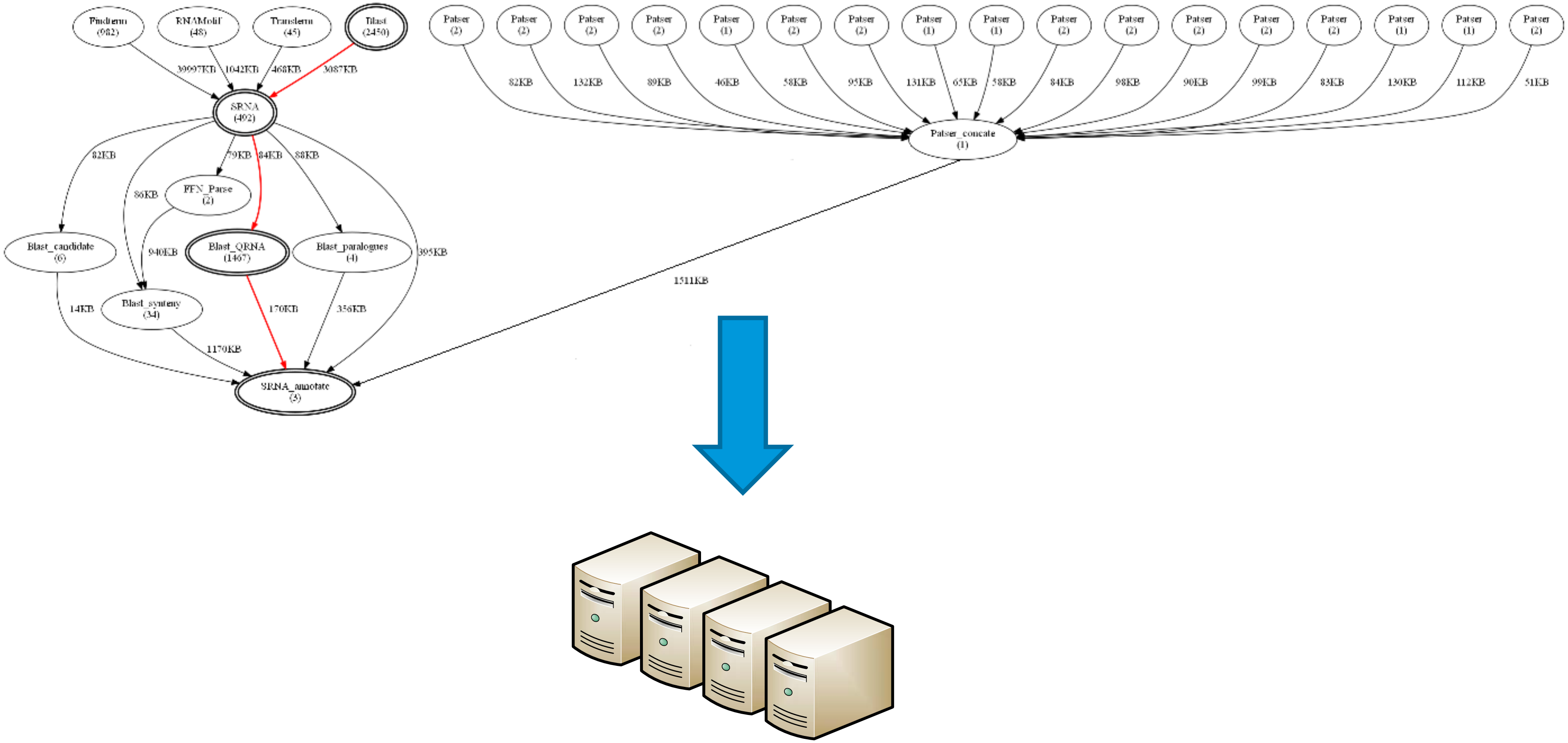
› Running scientific workflows

- *How many resources are needed for a given workflow application?*



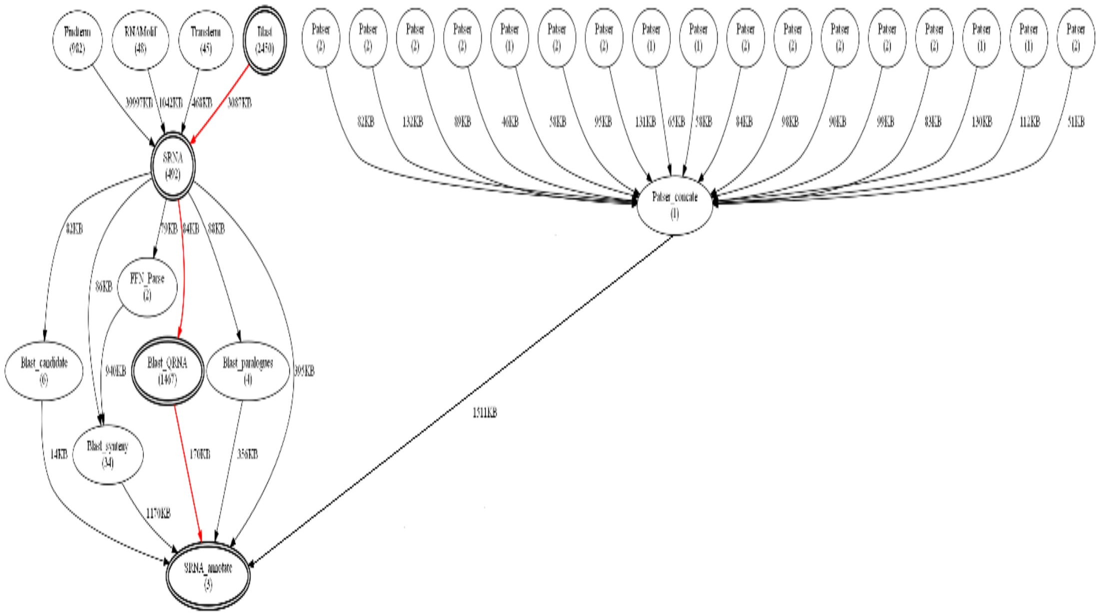
Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

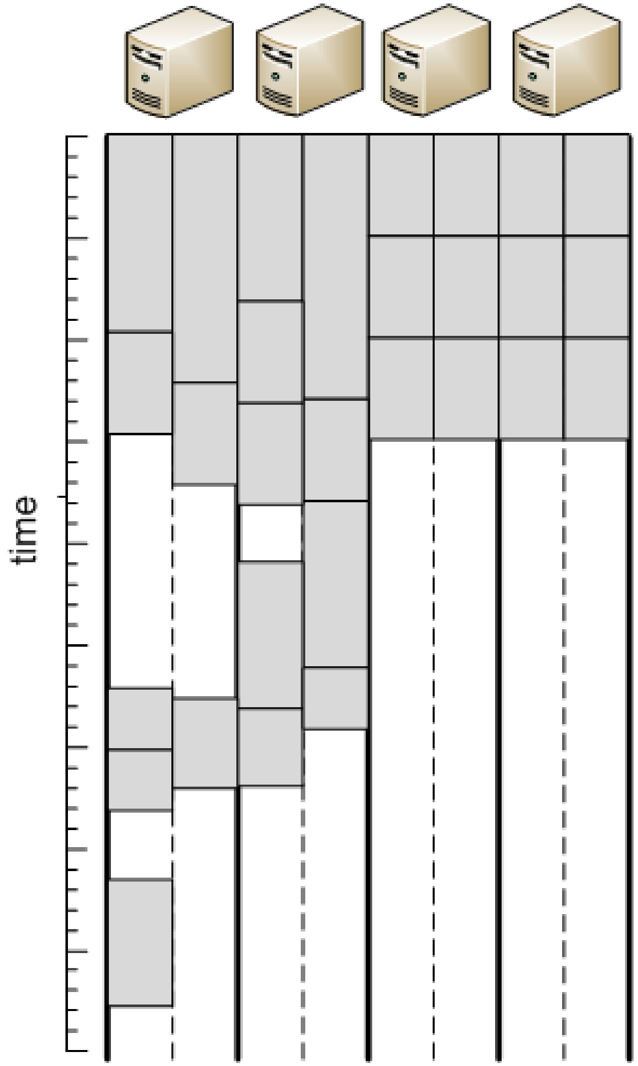
- Traditionally



Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

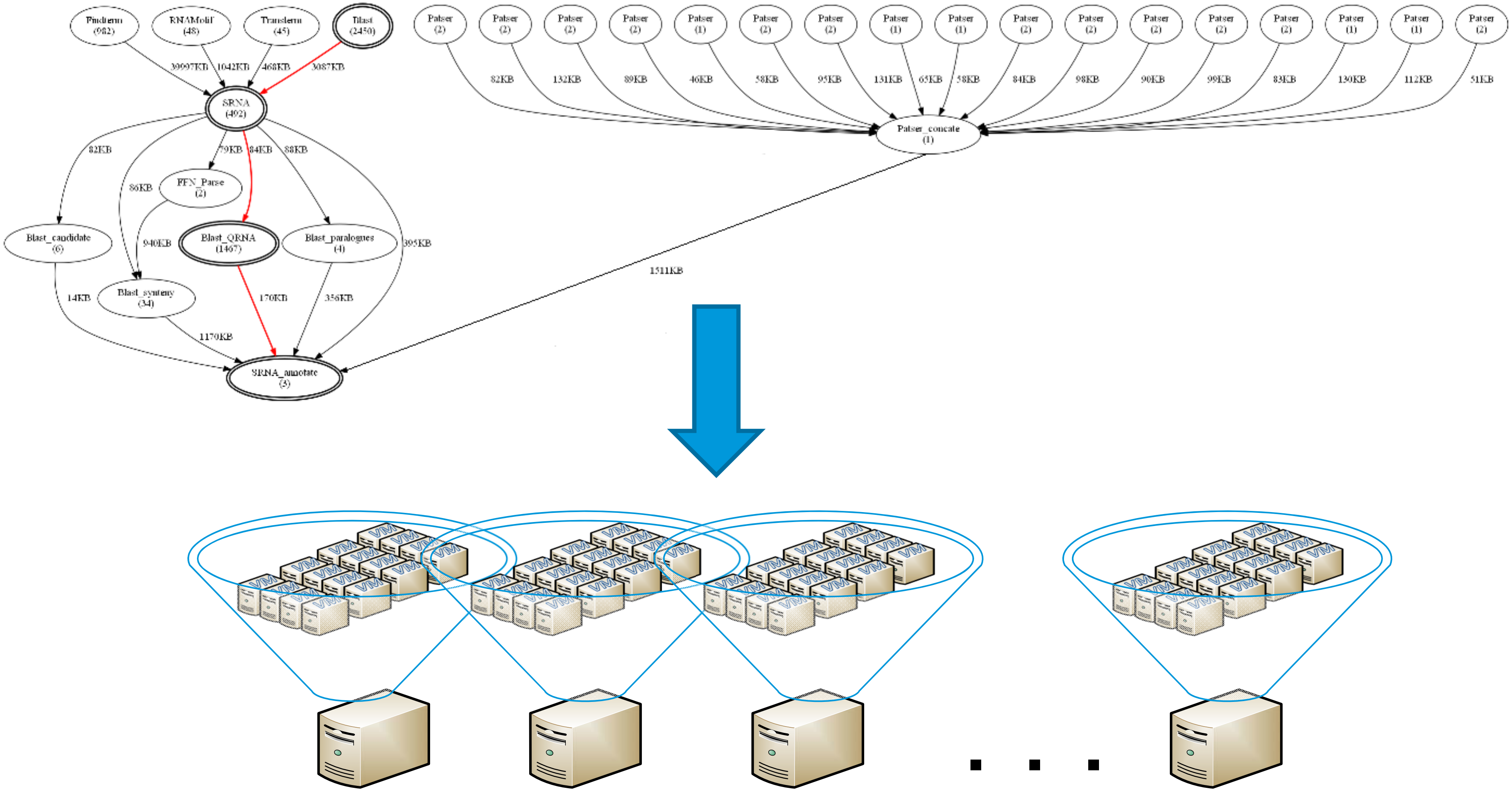
› Traditionally





Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

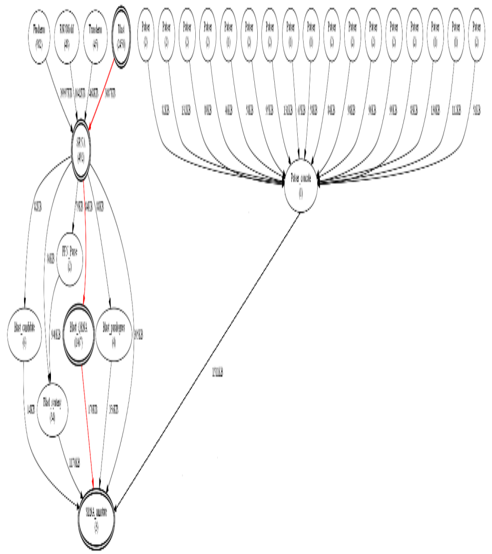
› Today

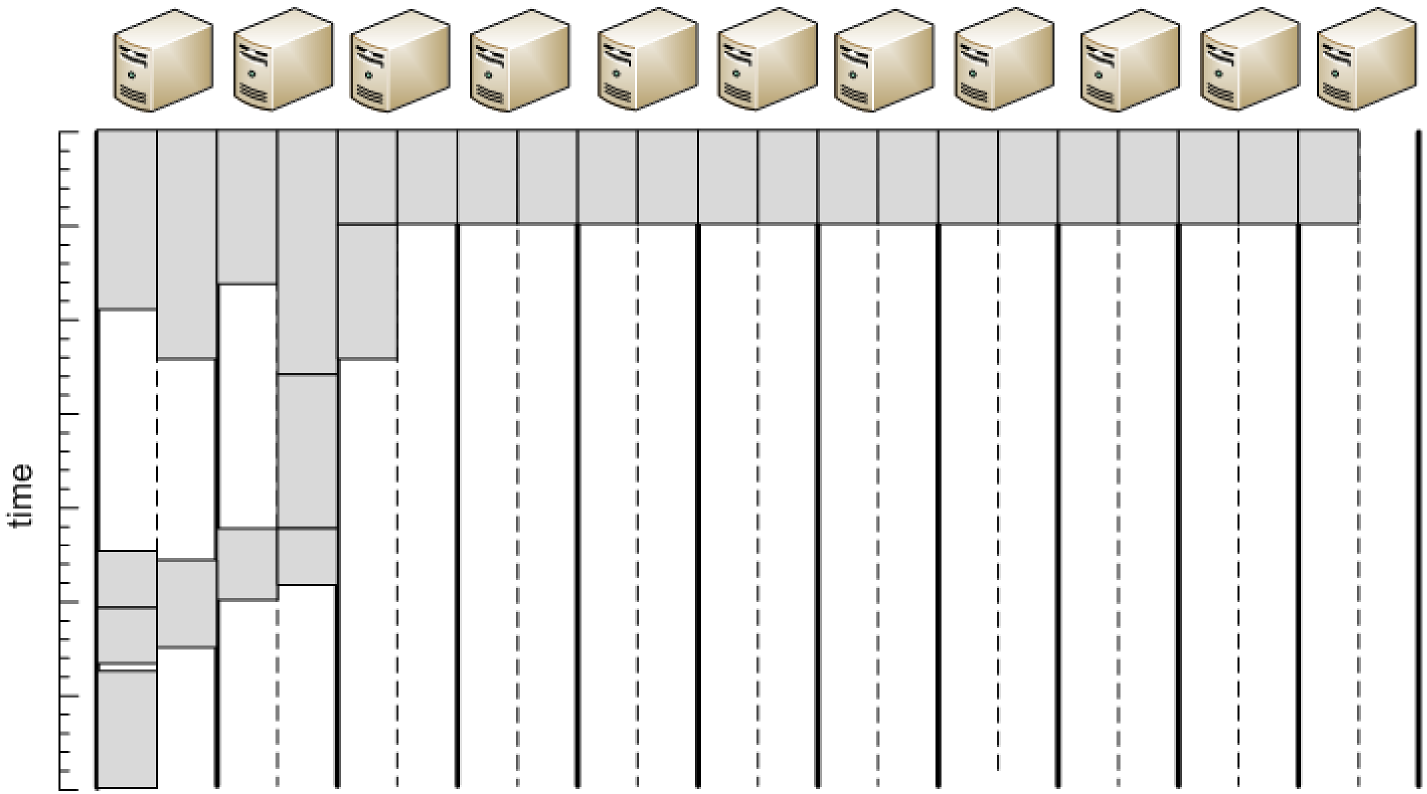


THE UNIVERSITY OF SYDNEY

Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

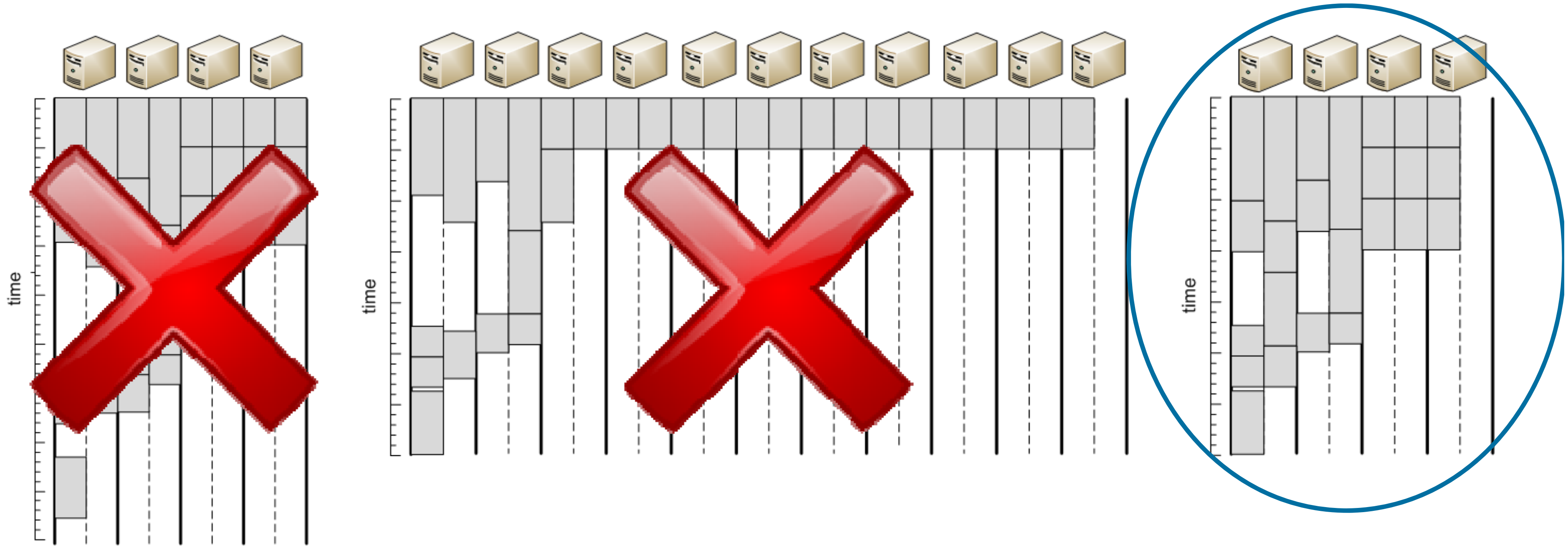
› Today





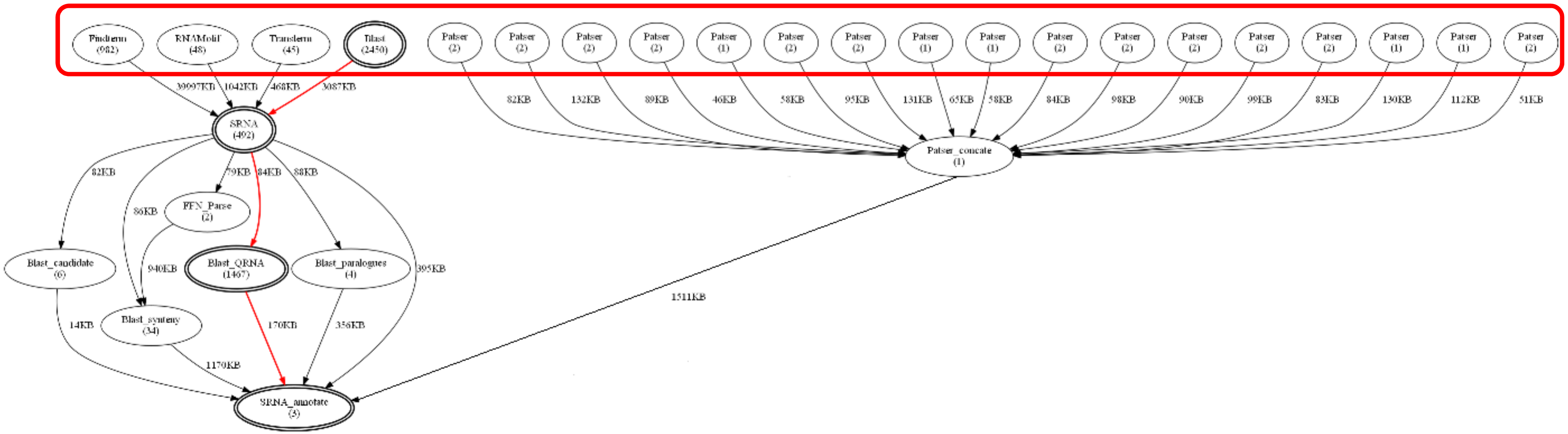
Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

› Resource efficient solution

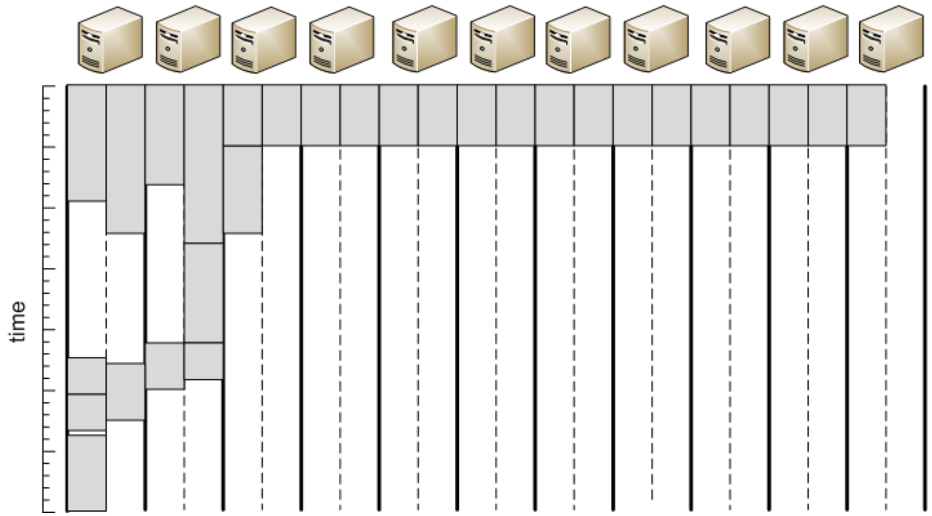


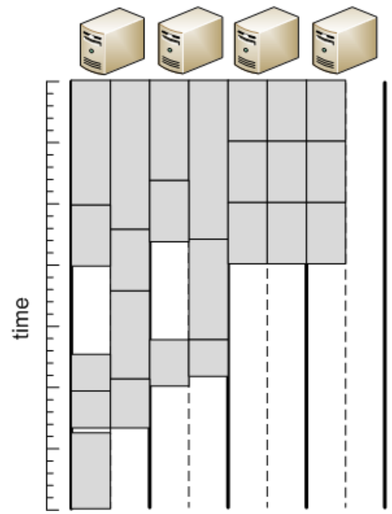
› Workflow scheduling with abundant resources

- How many resources are needed for a given workflow application?
- #resources used tends to be dominated by the (maximum) width of DAG



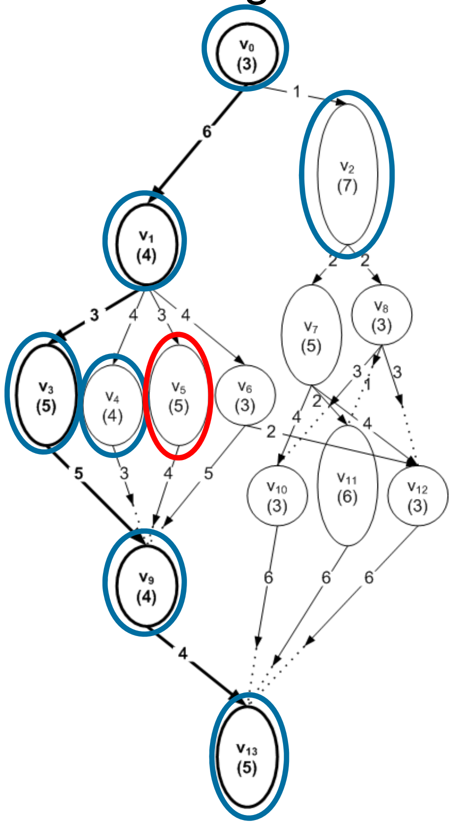
- Our solution (**stretch out and compact**)
  - CPF (Critical Path First): **stretch out** the schedule to preserve critical path length (the shortest possible time of completion) using as many resources
  - MER (Maximum Effective Reduction): **Compact** the schedule by rearranging tasks making use of idle/inefficiency slots present due to precedence constraints

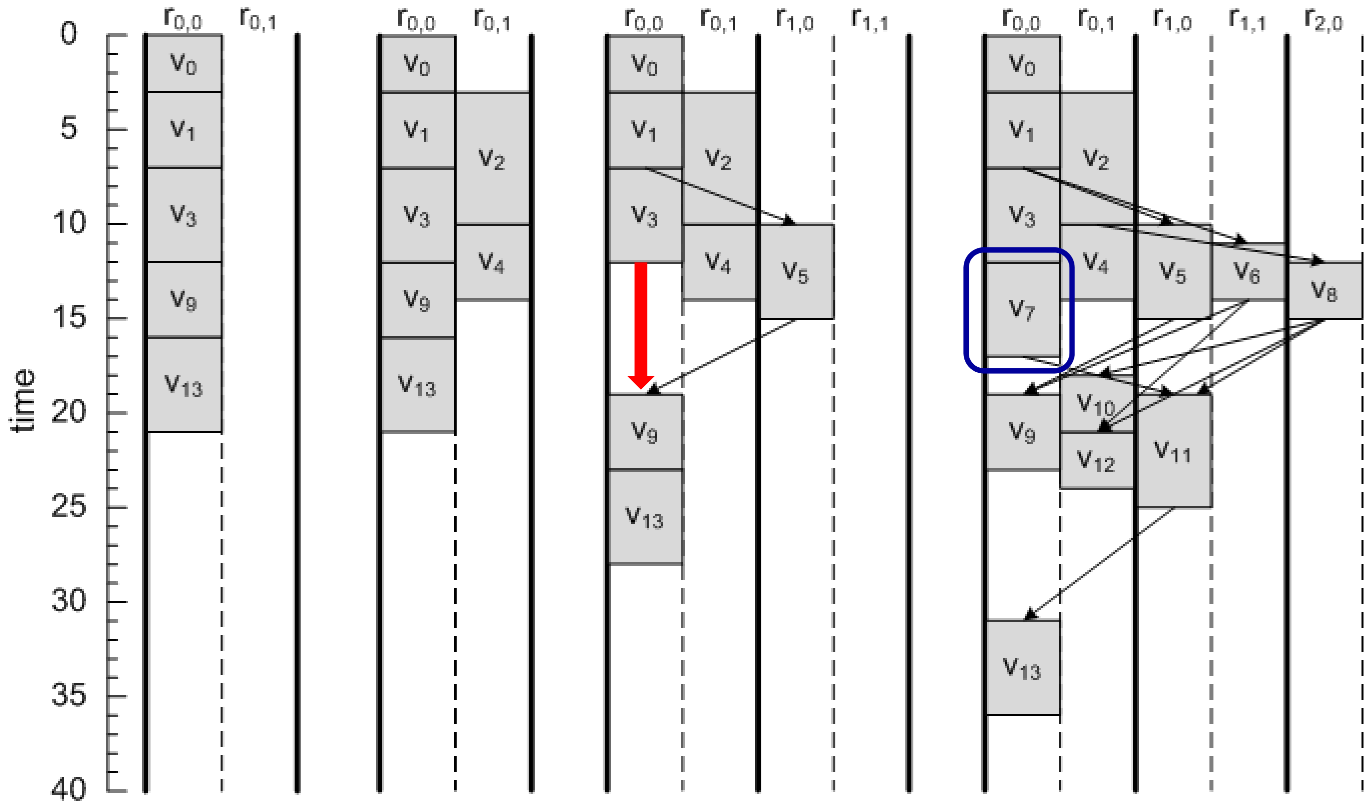




› Stretch out: Critical Path First (CPF)

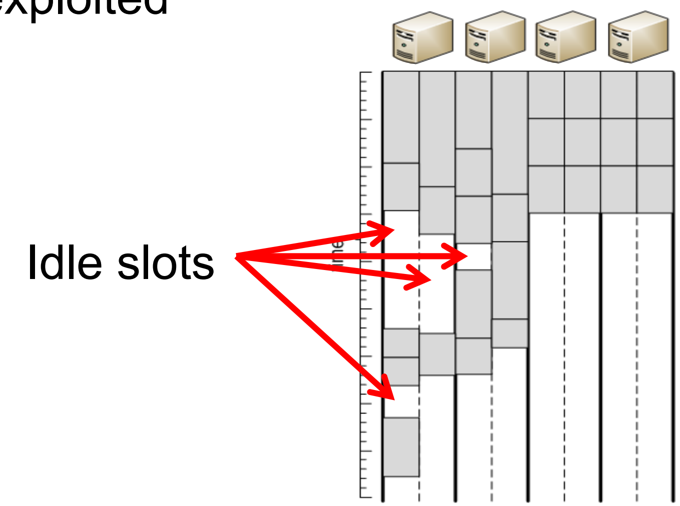
- Critical path length can be proactively preserved by **assigning all CP tasks on a particular resource (or CP resource) 'at the beginning'** and then scheduling remaining tasks





› Schedule compaction (Maximum Effective Reduction or MER)

- Makespan minimization and resource usage reduction are conflicting objectives
- Resource efficiency can be improved by resolving (or at least relieving) the conflict
- **How?**
  - The inefficiency in resource usage of workflow schedule (i.e., idle slots) should be better exploited



- Schedule compaction (Maximum Effective Reduction or MER)
  - The difference between resource usage reduction (RUR) and makespan increase (MI) in a resulting consolidated schedule as compared to the original output schedule

- $$Effective\ Reduction\ (ER) = \frac{(|R^0| - |R^*|)}{|R^0|} - \frac{(|ms^*| - |ms^0|)}{|ms^0|}$$

$|R^0|$: #resources used in the original schedule

$|R^*|$: #resources used in the consolidated schedule

ms^0 : the original makespan

ms^*: the makespan after consolidation

› Experimental Evaluation

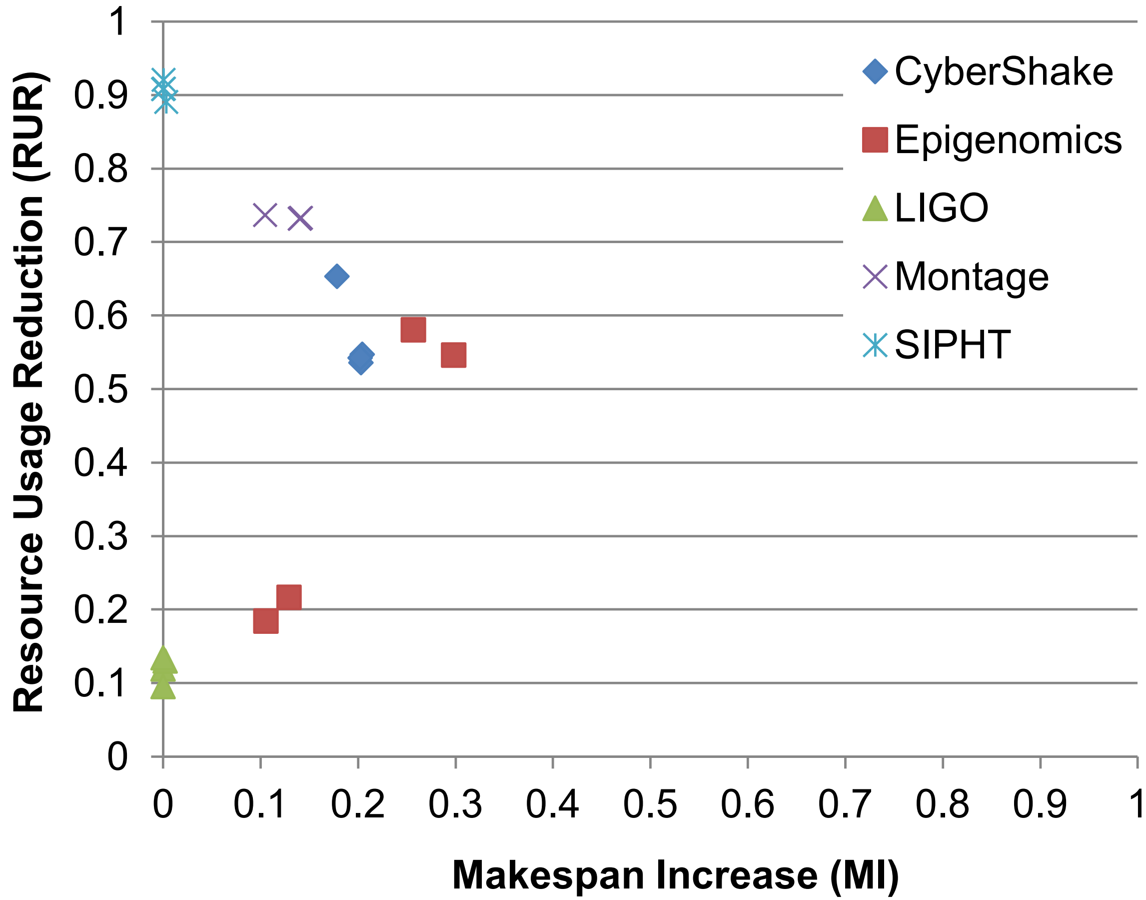
- Intel 40-core machine with 4 10-core Intel 2.4GHz Xeon processors
- Five real-world scientific workflows (50 - 6,000 tasks/job)
  - CyberShake, Epigenomics, LIGO, Montage and SIPHT

› Evaluation metrics

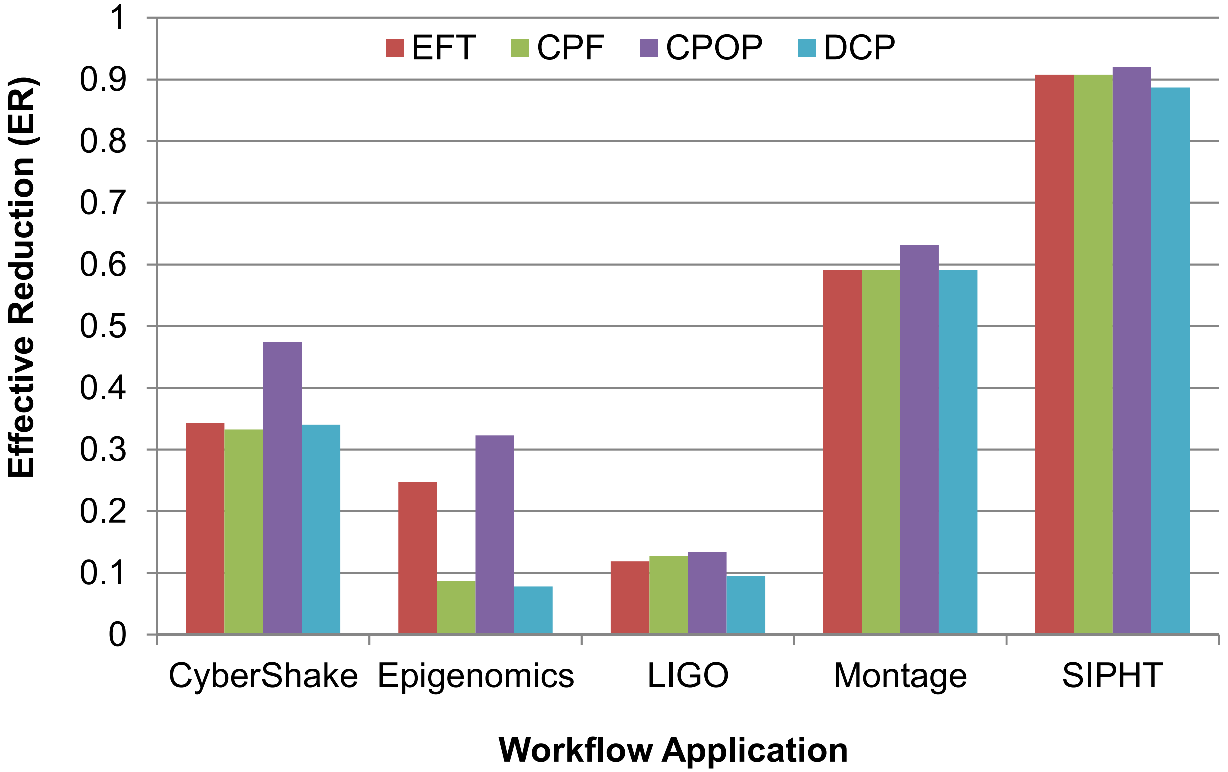
- Makespan
- #Resources used
- Algorithm running time

Optimizing the Efficiency of Clouds:
Resource Efficient Workflow Scheduling

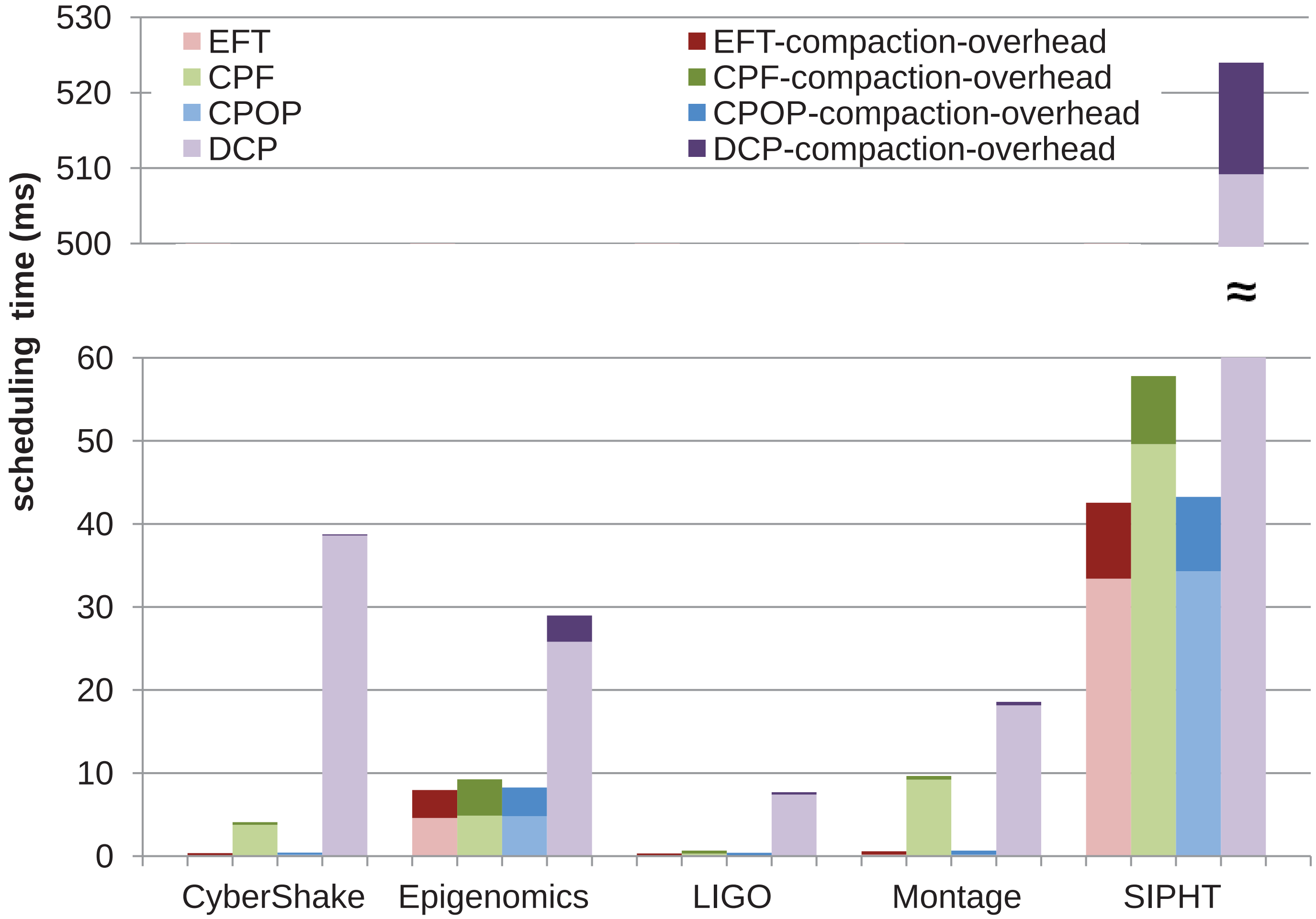
- Results: Makespan increase w.r.t resource usage reduction



› Results: effective reduction w.r.t. different apps and algorithms



› Results: scheduling time



Optimizing the Efficiency of Clouds:
Our Solutions

› Resource Efficient Workflow Scheduling

- Lee, Y. C. and Zomaya, A. Y., "Stretch Out and Compact: Workflow Scheduling with Resource Abundance," in the ***Proceedings of the International Symposium on Cluster Cloud and the Grid (CCGRID)***, May 13-16, 2013.
- Lee, Y. C., Han, H. and Zomaya, A. Y., "On Resource Efficiency of Workflow Schedules," in the ***Proceedings of the International Conference on Computational Science (ICCS)***, Jun. 10-12, 2014.
- Jiang, Q., Lee, Y. C. and Zomaya, A. Y., "Executing Large Scale Scientific Workflow Ensembles in Public Clouds," in the ***Proceedings of the International Conference on Parallel Processing (ICPP)***, Sep 1-4, 2015.

› High Performance/Throughput Computing Applications

- HosseinyFarahabady, M.R., Lee, Y.C., Han, H., Zomaya, A.Y., "Randomized Approximation Scheme for Resource Allocation in Hybrid-Cloud Environment," ***The Journal of Supercomputing*** 69(2): 576-592, 2014.
- Farahabady, M. H., Lee, Y. C. and Zomaya, A. Y., "Pareto-Optimal Cloud Bursting," ***IEEE Transactions on Parallel and Distributed Systems*** , 25(10): 2670-2682, 2014.

Optimizing the Efficiency of Clouds:
Executing Large-scale Workflow Ensembles

› Scientists need to run these workflows with different parameters repeatedly, or use a combination of different workflows to achieve an ultimate goal

› A ***workflow ensemble*** represents an entire scientific analysis as a set of interrelated but independent workflow applications

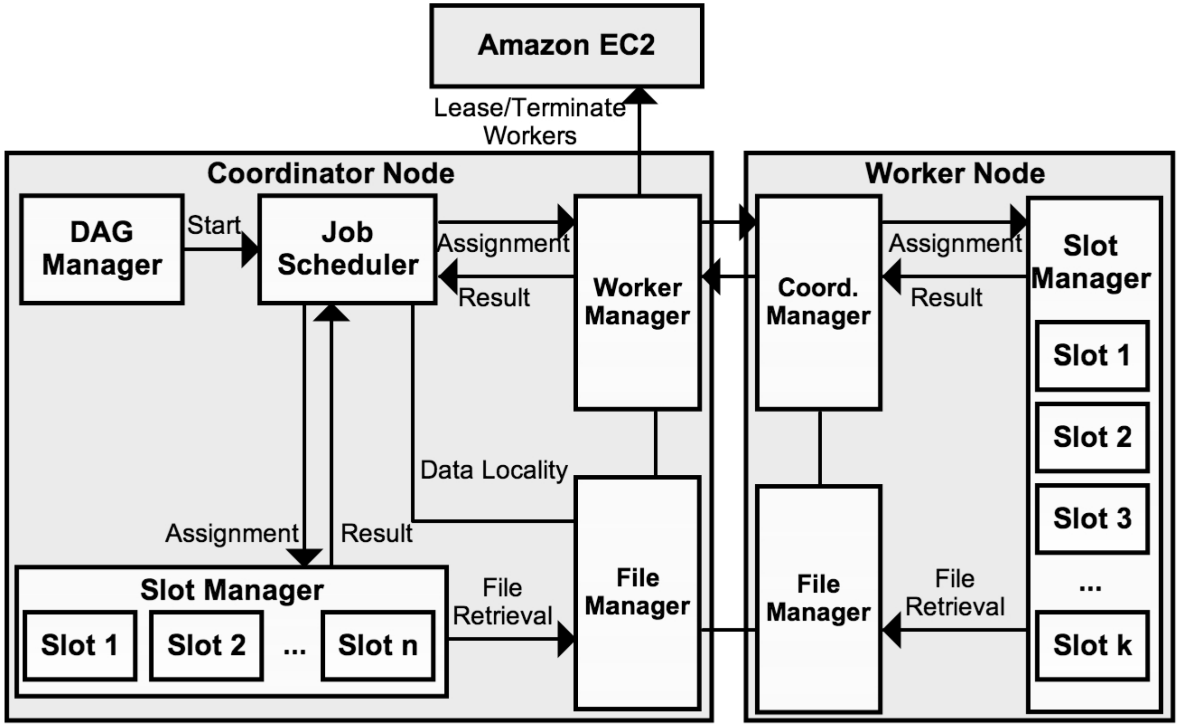
› **An ensemble of 200 6.0 degree Montage workflows**

- **1,717,200 jobs**
- **288,800 input files and 4,570,000 intermediate files, and**
- **Approximately 7 TB data footprint**

› We need an efficient “cloud-ready” workflow execution system for effectively dealing with resource allocation, data staging and execution coordination

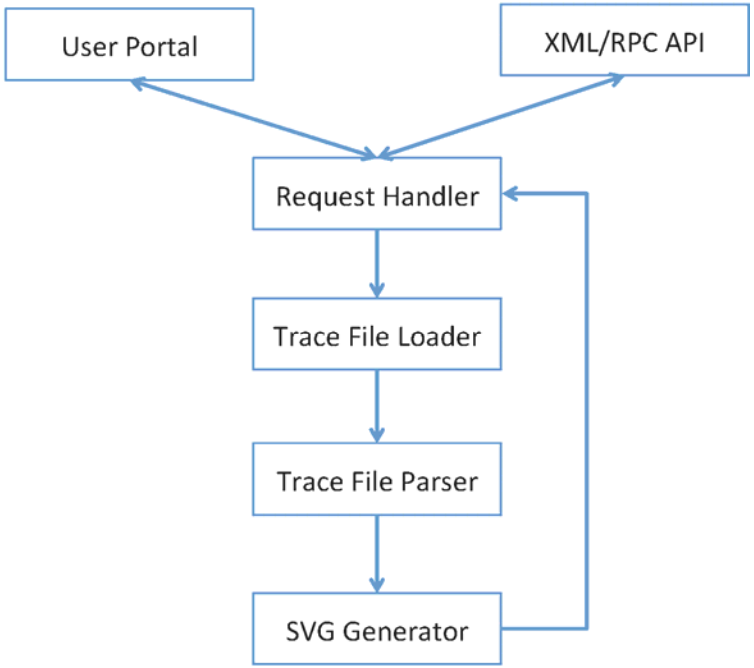
› DEWE (Distributed Elastic Workflow Execution)

- Open-source project supported by AWS Education Research Grant (https://bitbucket.org/lleslie/dwf/wiki/Home)

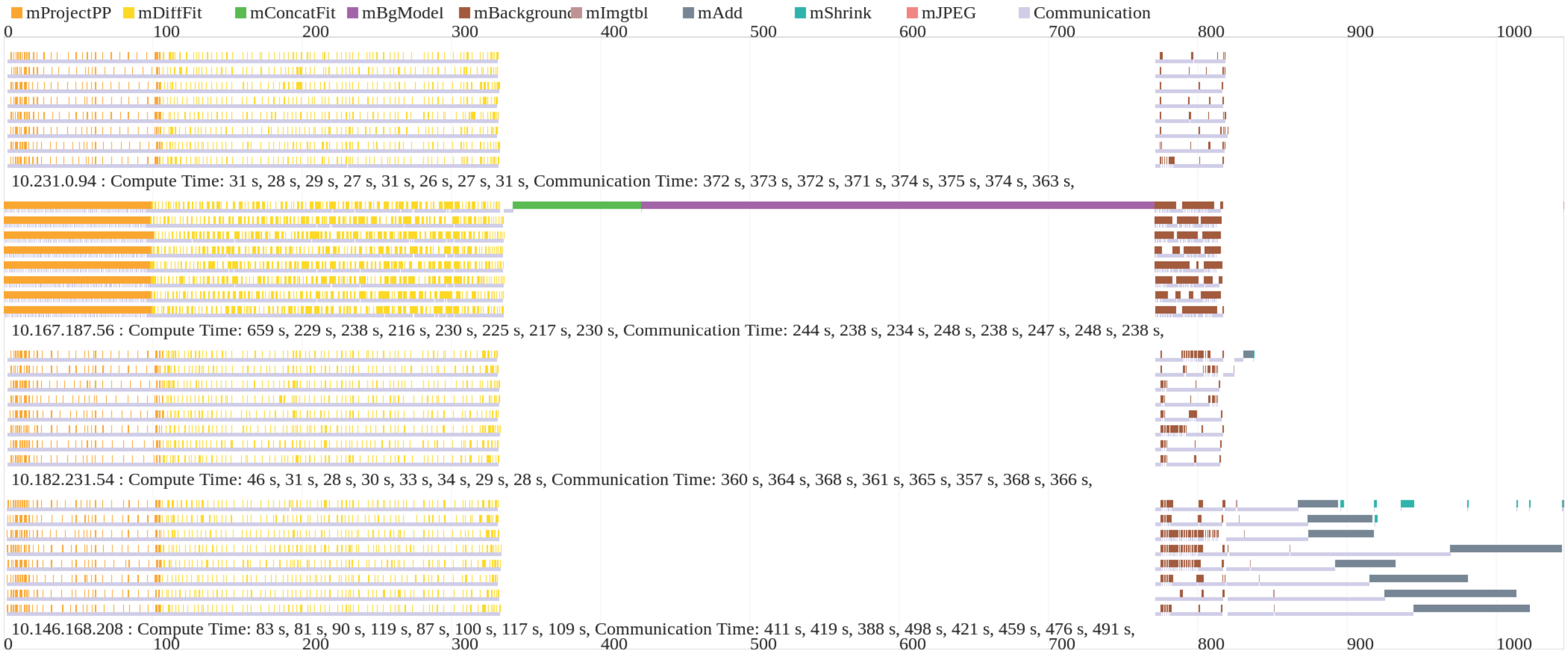


› DEWE (Distributed Elastic Workflow Execution)

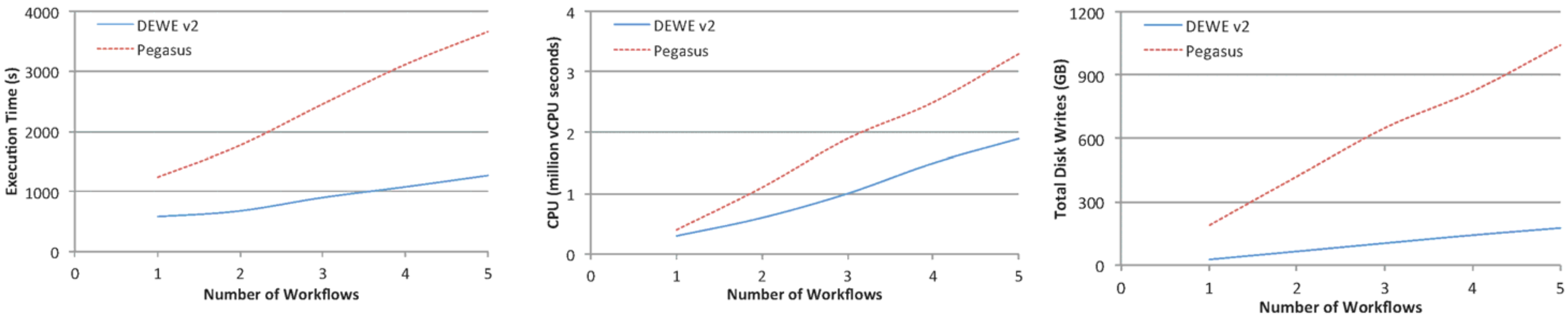
- The workflow visualization toolkit takes a workflow execution trace file as the input, and produces a scalable vector graph (SVG) or PDF representing the resource consumption status during the execution.



- DEWE (Distributed Elastic Workflow Execution)
  - The workflow visualization toolkit takes a workflow execution trace file as the input, and produces a scalable vector graph (SVG) or PDF representing the resource consumption status during the execution.



- DEWE vs. Pegasus (well-known workflow execution system)
  - Resource consumption of multiple 6.0 degree Montage workflows on Amazon EC2 c3.8xlarge instance



- DEWE evaluation
  - Node Performance Index *P* is used after profiling

$$P = \frac{W}{N * T}$$

W: the number of workflows

N: the number of worker nodes

T: the execution time needed for N workflows

Then, we can estimate the number of worker nodes needed to execute a large scale workflow ensemble with deadline constraints using the following formula:

$$N = \frac{W}{P * T}$$

› DEWE evaluation

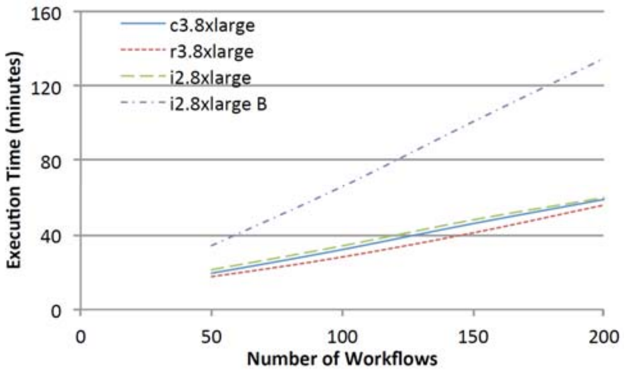
- Cluster configurations

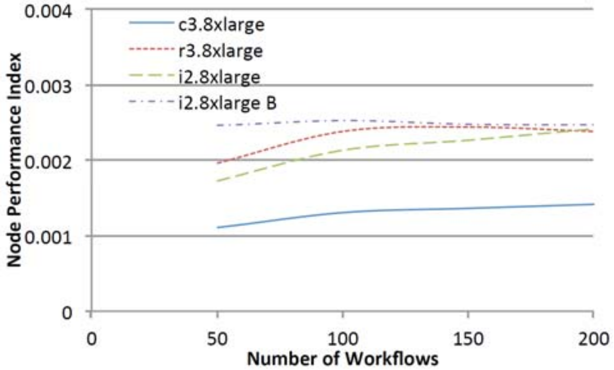
Cluster	#Nodes	#vCPUs	Memory (TB)	Storage (TB)	Price (USD/hr)
c3.8xlarge	40	1280	2.40	25.6	67.2
r3.8xlarge	25	800	6.10	16.0	70.0
i2.8xlarge	23	768	5.61	147.2	156.7
i2.8xlarge B	10	320	2.44	64.0	68.2

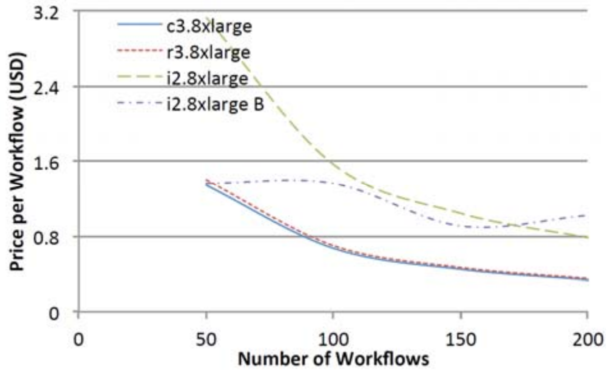
- Workflows
  - 50 - 200 6.0 degree Montage workflows
- Deadline constraint: 1 hour

› DEWE evaluation

- Results:
  - By adopting the pulling approach in our solution system, much of scheduling overhead can be removed as a majority of tasks in scientific workflows often exhibit homogeneity in their resource consumption pattern and acquiring a large number of homogeneous public cloud resources is easily possible.
  - **80% speed-up compared to Pegasus**
  - Cost and deadline compliance can be achieved







Optimizing the Efficiency of Clouds:
Our Solutions

› Resource Efficient Workflow Scheduling

- Lee, Y. C. and Zomaya, A. Y., "Stretch Out and Compact: Workflow Scheduling with Resource Abundance," in the ***Proceedings of the International Symposium on Cluster Cloud and the Grid (CCGRID)***, May 13-16, 2013.
- Lee, Y. C., Han, H. and Zomaya, A. Y., "On Resource Efficiency of Workflow Schedules," in the ***Proceedings of the International Conference on Computational Science (ICCS)***, Jun. 10-12, 2014.
- Jiang, Q., Lee, Y. C. and Zomaya, A. Y., "Executing Large Scale Scientific Workflow Ensembles in Public Clouds," in the ***Proceedings of the International Conference on Parallel Processing (ICPP)***, Sep 1-4, 2015.

› High Performance/Throughput Computing Applications

- HosseinyFarahabady, M.R., Lee, Y.C., Han, H., Zomaya, A.Y., "Randomized Approximation Scheme for Resource Allocation in Hybrid-Cloud Environment," ***The Journal of Supercomputing*** 69(2): 576-592, 2014.
- Farahabady, M. H., Lee, Y. C. and Zomaya, A. Y., "Pareto-Optimal Cloud Bursting," ***IEEE Transactions on Parallel and Distributed Systems*** , 25(10): 2670-2682, 2014.

› **Why cloud bursting?**

- Many organizations already operate their own computing facilities, called private clouds or data centres

- Multi-cloud model is practical and realistic in many scenarios:
  - Security is a major concern (compared to cloud sourcing)
  - Workloads exhibit different characteristics
  - Sporadic workload surges occur (a major source of over provisioning, inefficient resource usage)

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

› Tools for cloud bursting

EUCALYPTUS

openstack™
CLOUD SOFTWARE

OpenNebula.org
The Open Source Solution for Data Center Virtualization

ganeti
Cluster-based virtualization management software

Bright Computing
Cluster Manager

THE UNIVERSITY OF SYDNEY

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

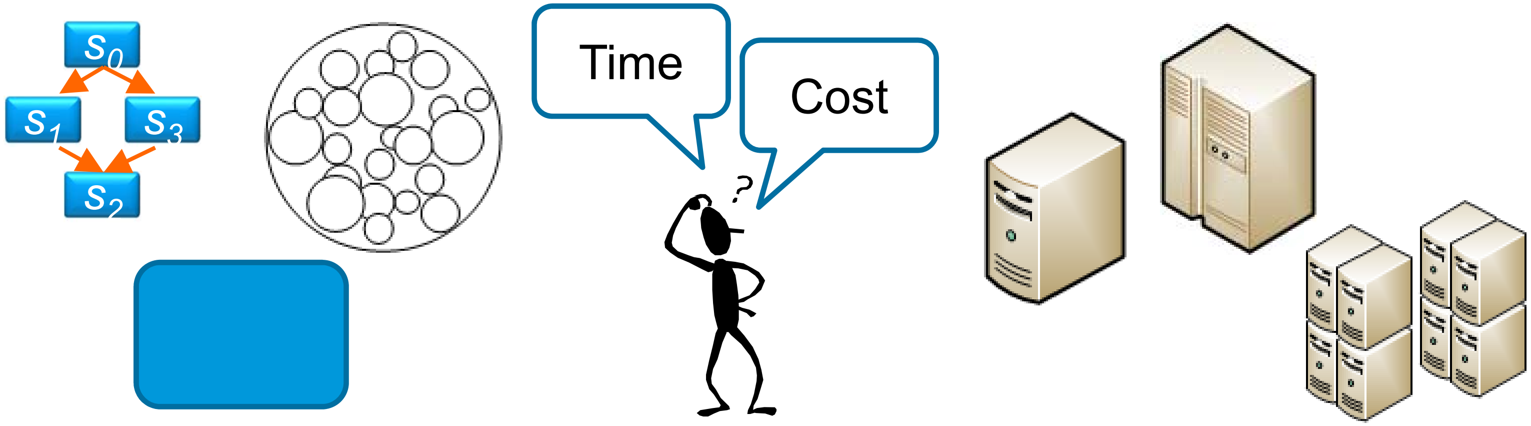
› Different users have a diverse set of applications possibly with different objectives, e.g., performance/time, cost, etc.

› Cloud providers offers a number of different services

- E.g., Standard, High-CPU, High-Memory, Compute Cluster, GPU Cluster

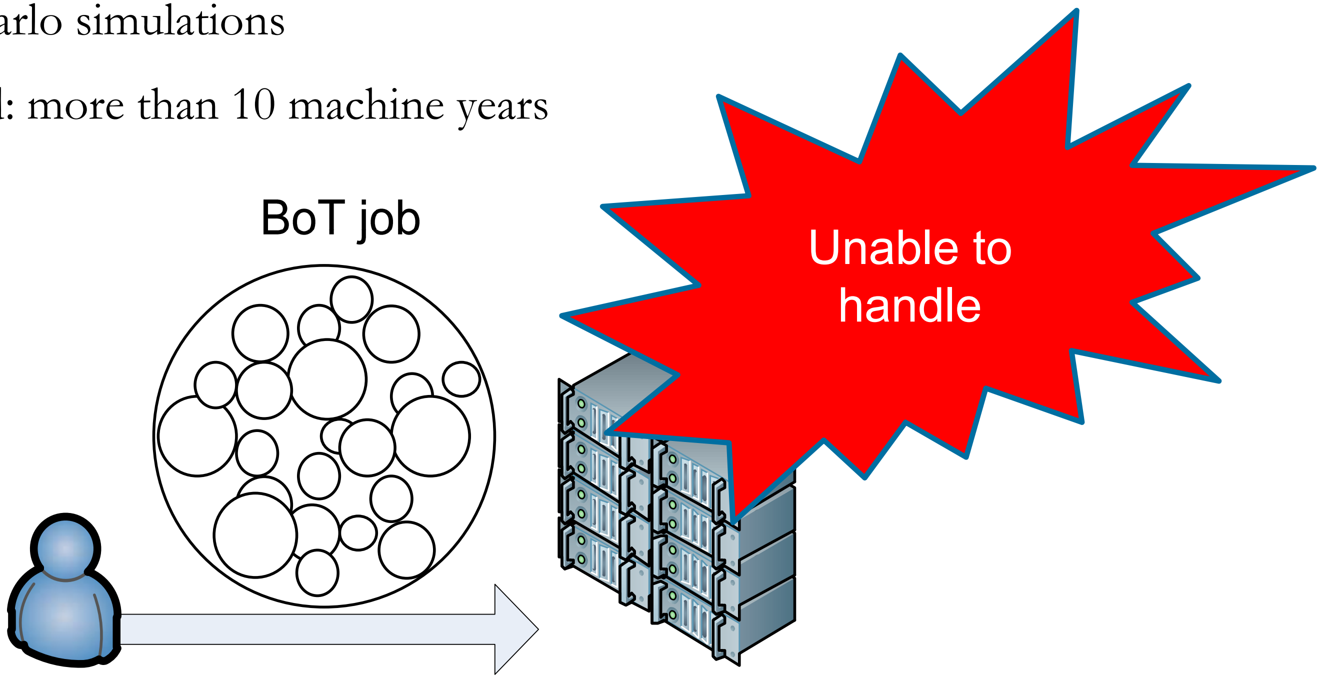
› Usage is typically charged by the hour

› Cost to performance ratio (cost efficiency) may vary significantly by scheduling and resource allocation

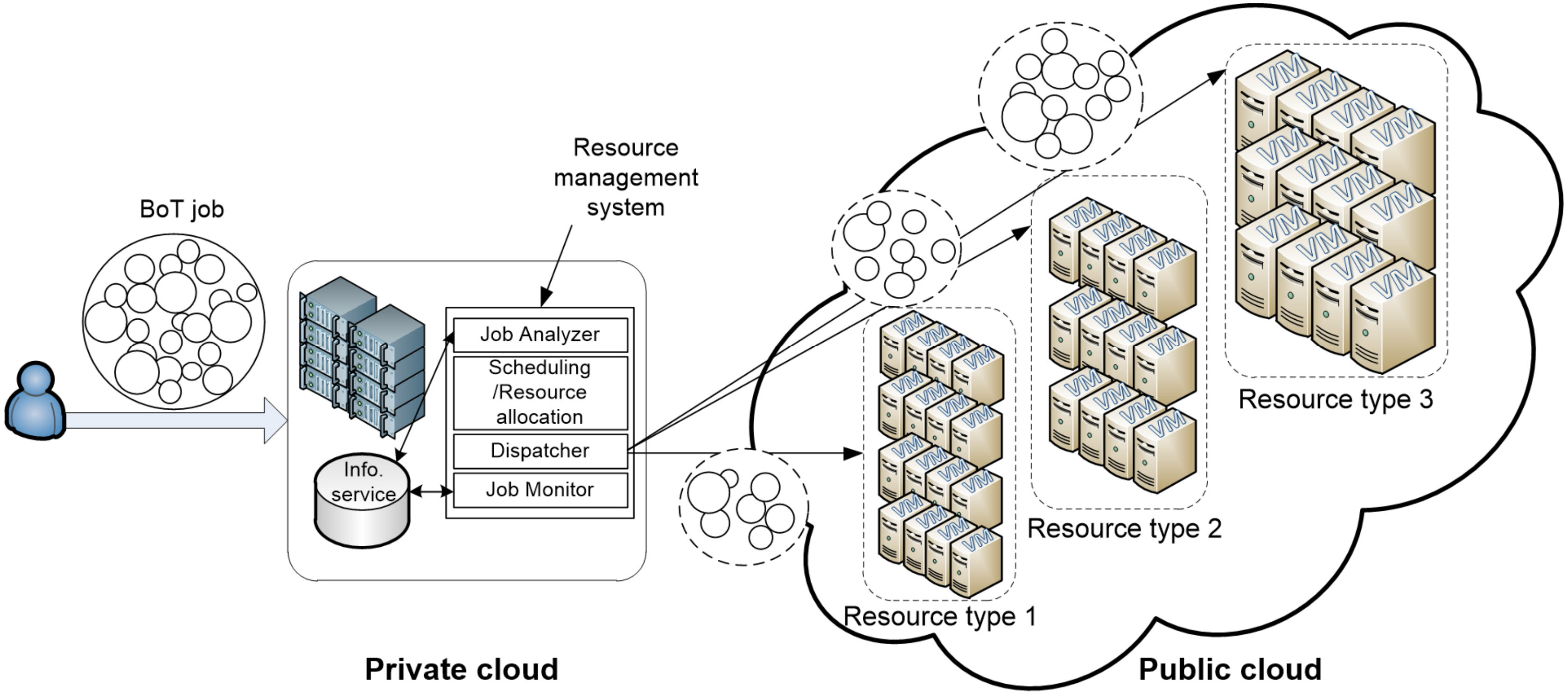


Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

- Private system often gets overwhelmed by resource requirement of bag-of-tasks (BoT) applications
  - BoT applications are common in science and engineering
    - Monte Carlo simulations
  - CycleCloud: more than 10 machine years



Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications



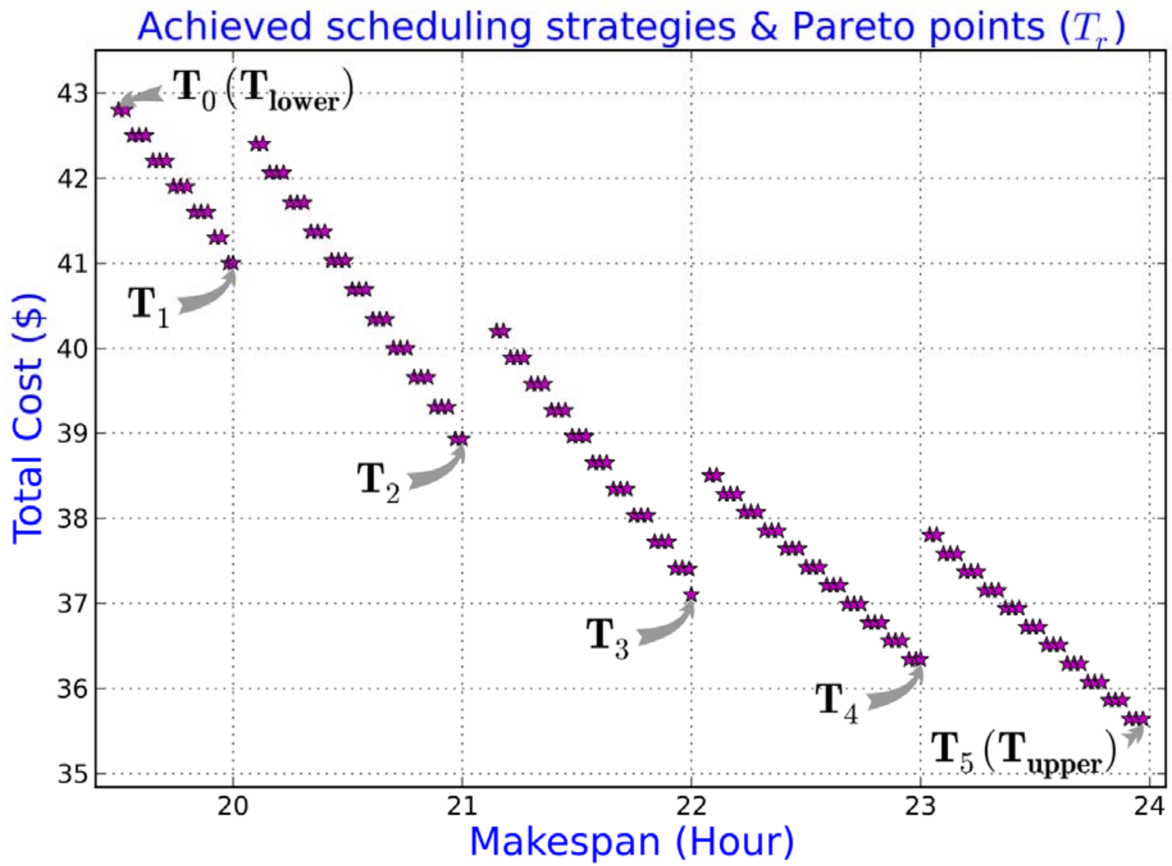
› **Cloud bursting with BoT applications**

- Multi-cloud model
  - Public and private cloud resources: $\langle s1, s2 \ldots sk\rangle$ and $\langle c1, c2 \ldots ck\rangle$
  - BoT application model
  - Set of *n* tasks
  - *Pi* : amount of time required to complete, unknown in advance
  - If task *j* run on machine *i*, it takes $Pj\ /s_i$ to finish.
- Objective function
  - User has two conflicting objectives of minimizing cost and maximizing performance (minimizing makespan)

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

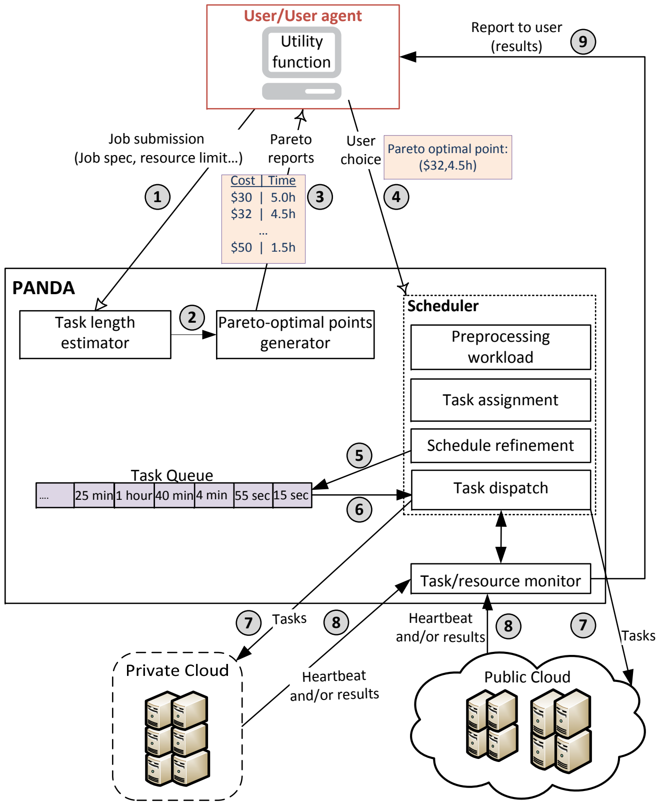
› Closer look to objective function

- Pareto optimality effectively captures the trade off between two conflicting objectives



Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

- **PANDA** (**PA**reto **N**ear-optimal **D**eterministic **A**pproximation)
  - A fully polynomial time approximation scheme (FPTAS) with input size *n* and approximation factor ε
- Four major steps
  - Pre-processing
    - Tasks are pre-processed for their lengths to be equalized
  - Task selection with trimming
    - Tasks are selected by solving subset sum problem
  - Task assignment
    - Each machine gets its workload (optimal #tasks)
  - Solution refinement
    - A task currently assigned to a slow resource is moved to a faster resource such that the time required by the faster resource does not incur any extra cost



Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

› Optimal task assignment: **integer programming**

$$\min z = \sum_{i \in \Gamma} L_i c_i \left\lceil \frac{x_i P}{s_i} \right\rceil \frac{x_i P}{s_i} + L_v c_v \left(\frac{x_v P}{s_v} \right)^2$$

$$s.t. \qquad \sum_{i \in \Gamma} L_i x_i + L_v x_v = n$$

$$x_v, x_i \in \mathbb{Z}^{\geq 0}$$

› Optimal solution for relaxed problem:

$$x_i = \frac{n L_i}{\alpha_i \sum_{j \in \Gamma \cup \{v\}} \frac{L_j^2}{\alpha_j}} \quad , \quad \forall i \in \Gamma \cup \{v\}$$

$$\alpha_i = L_i c_i P^2 / s_i^2, \text{ for } i \in \Gamma \cup \{v\}$$

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

- Pre-processing:
- Task selection:
- Task assignment
- Refinement

Algorithm 3: Approximate task assignment

input : $n, \epsilon, \boldsymbol{\pi} = \{P_1, P_2 \cdots P_n\}$, L_i, c_i, s_i; $\forall i \in \Gamma \cup \{v\}$

output: $\boldsymbol{\pi}_i$; a partition scheme of π while the sum of numbers in π_i approximates x_i^*;

begin

Set μ to a small real number, e.g., 1;

Let $n^* = \sum_{i=1}^{n} \frac{P_i}{\mu}$;

$x^* \leftarrow FindOptimal(n^*, \mu, L_i, c_i, s_i)$; // Algorithm 1

$Sort(x^*, descending)$;

$\boldsymbol{\pi}' \leftarrow \boldsymbol{\pi}$;

for $i = 1 \cdots k+1$ **do**

$\ell_0 \leftarrow \langle 0 \rangle$

for $j = 1 \cdots |\pi'|$ **do**

$\ell_j \leftarrow MergeList(\ell_{j-1}, \ell_{j-1} + P_j)$

$\ell_j \leftarrow Trim(\ell_j, \frac{\epsilon}{2|\pi'|})$ // Algorithm 2

Remove elements from ℓ_j for which the size is greater than $\frac{x_i^*}{1 - \frac{\epsilon}{2|\pi'|}}$

end

Let π_i^* be the nearest value to x_i^* in ℓ_j;

$\boldsymbol{\pi}_i \leftarrow$set of numbers whose sum is equal to π_i^*;

$\boldsymbol{\pi}' \leftarrow \boldsymbol{\pi}' - \boldsymbol{\pi}_i$;

end

Run the $refinement$ process in Algorithm 1.

end

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

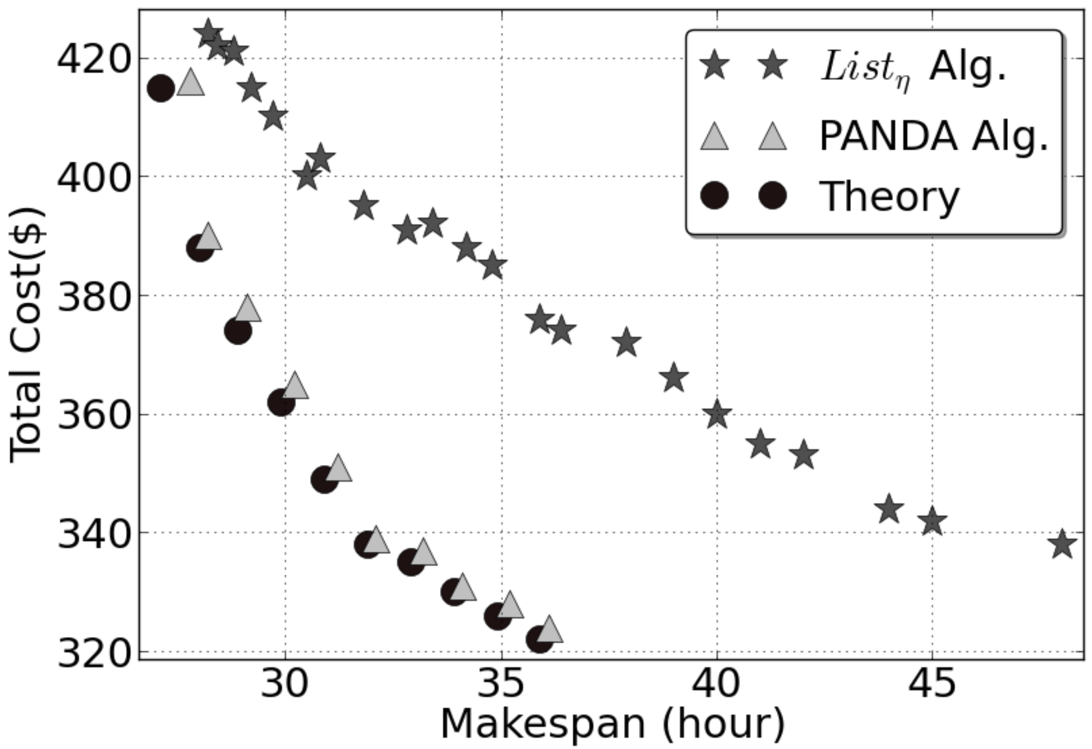
- **Experimental evaluation**
- We modeled ISOMAP as a real-world BoT application.
  - consists of tens of thousands of (CPU-intensive) tasks.
  - each task runs for seconds or up to tens of minutes.
  - Job sizes in million seconds (Ms): {1 Ms, 5 Ms, 10 Ms, 17 Ms}

- Multi-cloud setting

Cloud	Res. Type	Proc. Capacity	Hourly Cost
Amazon EC2 US East (VA)	m1.small	1	$0.080
	c1.medium	5	$0.165
	m1.large	4	$0.320
	c1.xlarge	20	$0.660
Private	4x10-core Xeon	10	$0.320

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

› Pareto frontier reached (1) theoretically, (2) by PANDA, and (3) by a modified List heuristic



L_i = 5, ε = 0.1, and job size = 10Ms (on m1.small)

Optimizing the Efficiency of Clouds: A Case for HPC/HTC applications

› Average values of makespan and total cost with respect to different sizes of BoT applications.

BoT size	$List_\eta$ ms(h)	$List_\eta$ cost	PANDA ms(h)	PANDA cost	Optimal ms(h)	Optimal cost
1M.s.	2.2	66.0	1.8	58.5	1.5	58.5
5M.s.	4.2	118.4	3.6	117.2	3.3	117.2
10M.s.	5.5	153.5	4.9	146.0	4.5	146.0
17M.s.	9.7	241.7	8.2	215.6	7.9	192.8

$L_i = 20$, $\varepsilon = 0.1$ on m1.small

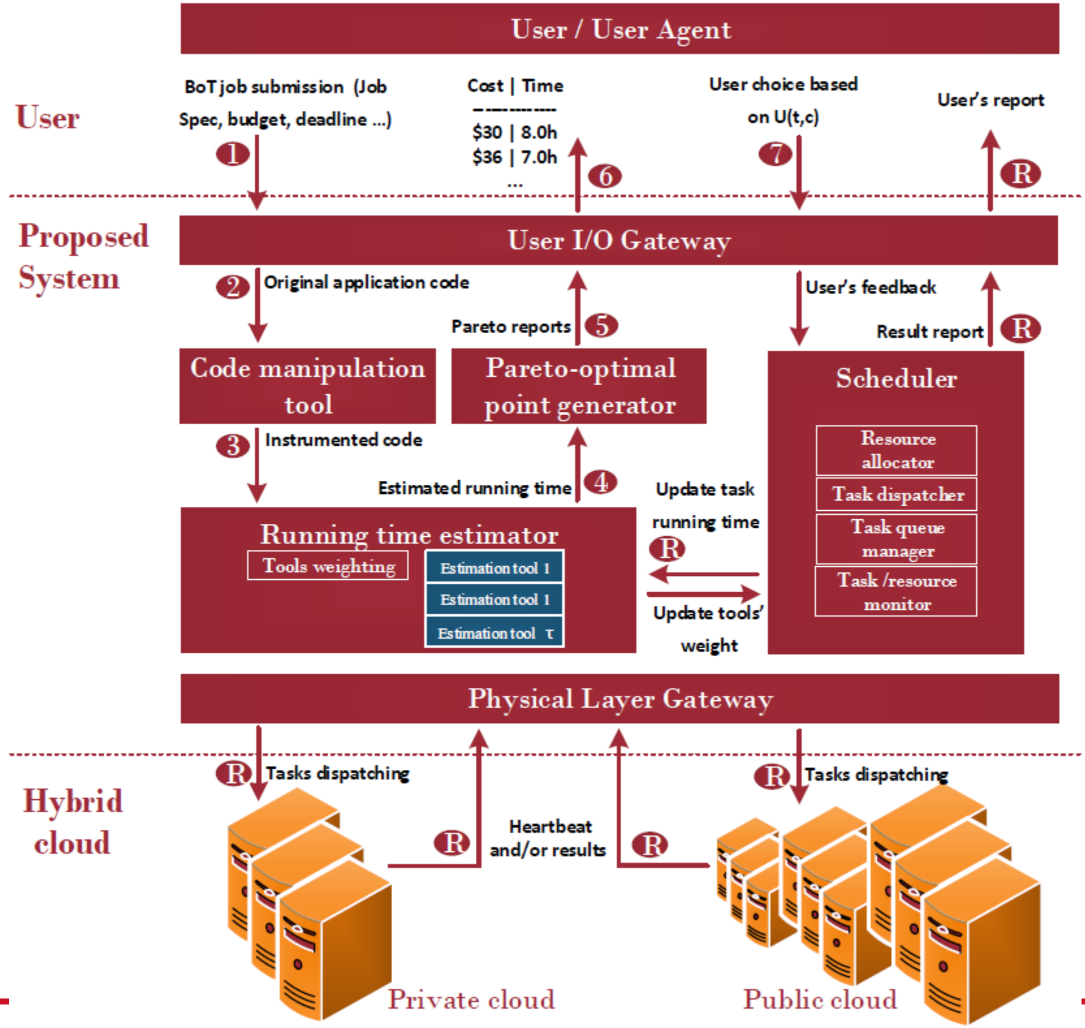
Unknown task execution times

- **PESU** (**P**areto **E**fficient **S**cheduling with **U**ncertainty)
  - We devise a dynamic resource allocation solution with a hybrid task running time estimation technique based on a feedback control mechanism

- Three phases
  - Estimation
    - estimates the execution time of each task using existing estimation techniques
  - Pareto-efficient point generation
    - Generates possible Pareto-efficient schedules
  - Resource allocation
    - Allocates resources for the selected Pareto-efficient point

Unknown task execution times

› PESU



Unknown task execution times

› Running time estimation

We use existing estimation techniques (e.g., ATOM, Pin, and Valgrind) in an iterative fashion

1. Add several breakpoints to each task
2. Assign an accurate weight to each tool by monitoring and comparing the actual running time of breaking points
3. Divide the whole time horizon into equal intervals
4. At the beginning of each interval, a monitoring phase happens:

- the actual revealed running time and the estimated running time are compared to evaluate the accuracy of each estimation tool.

Experimental evaluation: **Unknown task execution times**

- We modeled ISOMAP as a real-world BoT application.

Type (BoT size, Task Running Time)	No. Tasks $=10k \times 2^x$	Task Length $=2^x$(minitue)
LS (Large, Short)	x ~Wbl(1.7,2)	x ~U(0,3)
LL (Large, Long)	x ~Wbl(1.7,2)	x ~N(3.5,3)
LM (Large, Mixture)	x ~Wbl(1.7,2)	x ~N(1.8,3)

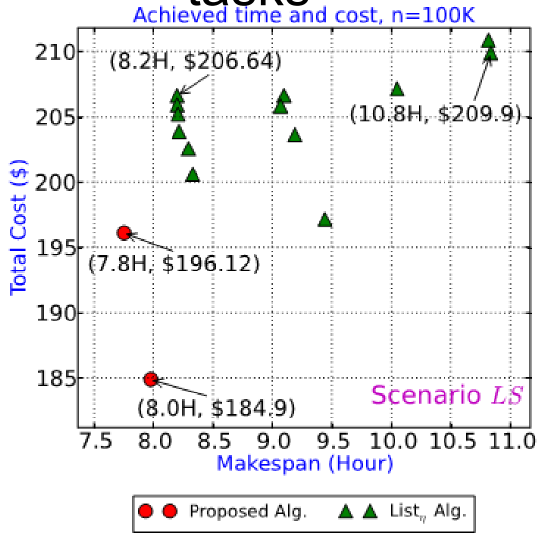
- Multi-cloud setting

Cloud	Res. Type	Proc. Capacity	Hourly Cost
Amazon EC2 US East (VA)	m1.small	1	$0.080
	c1.medium	5	$0.165
	m1.large	4	$0.320
	c1.xlarge	20	$0.660
Private	4x10-core Xeon	10	$0.320

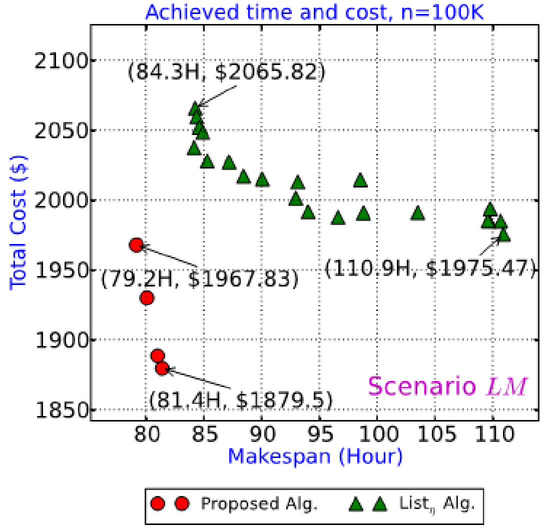
Results: **Unknown** task execution times

- Comparison of makespan and cost

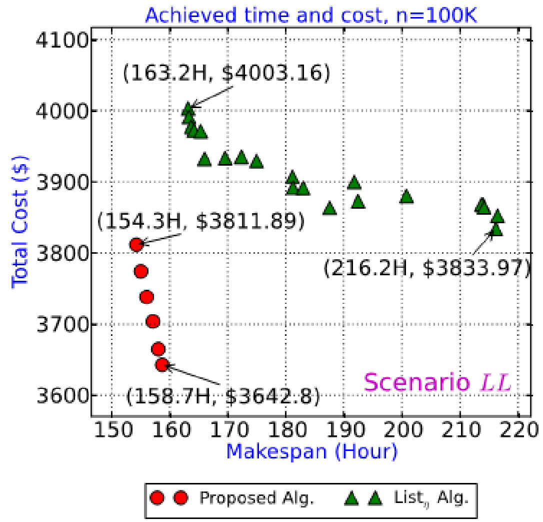
Short tasks



Short + Long tasks



Long tasks



Simple ideas, but hard to implement!!!!

Conclusion

- Today, with advances in **VM techniques** and the advent of **multi-/many-core processors**, resources are ever abundant
- Computing and data processing needs continuously increase
- Simply expanding resource capacity has resulted in poor resource utilization, i.e., average data center utilization is 10-30% or less
- Adaptive resource management for typical workloads in clouds are essential
  - Workflows: Maximization of resource utilization with min performance impact
  - HPC/HTC apps: Capturing trade-off between cost and performance

Sample of current research projects

- **Cost Efficiency of the Data Centre**
  - Cost reductions and profit increases (e.g. game theoretic methods)
  - Pay-as-you-go pricing, pricing dynamics
- **Implications of multi tenancy**
  - Resource virtualization → Resource contention (migrate VMs?)
  - Current SLAs: only availability (need to consider performance?)
- **Scheduling and resource allocation as a cost efficient solution (energy minimization**
  - Exploitation of application characteristics (e.g. data locality, latency, quality of service, execution time)
  - Explicit consideration of user experience/satisfaction
  - Map reducing applications, tuning Map reducible applications.
  - Hybrid clouds, cloud bursting for execution time, energy efficiency, pricing, privacy

Other recent work

- Rajiv Ranjan, Joanna Kolodziej, Lizhe Wang, Albert Y. Zomaya: Cross-Layer Cloud Resource Configuration Selection in the Big Data Era. **IEEE Cloud Computing** 2(3): 16-22 (2015)
- Lingfang Zeng, Bharadwaj Veeravalli, Albert Y. Zomaya: An integrated task computation and data management scheduling strategy for workflow applications in cloud environments. **J. Network and Computer Applications** 50: 39-48 (2015)
- Rajiv Ranjan, Lizhe Wang, Albert Y. Zomaya, Dimitrios Georgakopoulos, Xian-He Sun, Guojun Wang: Recent advances in autonomic provisioning of big data applications on clouds. **IEEE Trans. Cloud Computing** 3(2): 101-104 (2015)
- Lizhe Wang, Yan Ma, Albert Y. Zomaya, Rajiv Ranjan, Dan Chen: A Parallel File System with Application-Aware Data Layout Policies for Massive Remote Sensing Image Processing in Digital Earth. **IEEE Trans. Parallel Distrib. Syst.** 26(6): 1497-1508 (2015)

THE UNIVERSITY OF SYDNEY

Thank you