Understanding the Dark Side of Big Data Clusters: An Analysis beyond Failures

Andrea Rosà[*], **Lydia Y. Chen**[†], **Walter Binder**[*]

[*]Università della Svizzera italiana, Faculty of Informatics, Switzerland
[†]IBM Research Zurich Lab, Cloud Server Technologies Group, Switzerland

IEEE/IFIP DSN 2015

June 23, 2015

This work has been supported by the Swiss National Science Foundation (project 200021_141002) and EU commission under FP7 GENiC project (608826).

Overview

- Extensive study on **unsuccessful executions**
  - Job and task terminations caused by internal and external events
- Analysis of Google cluster trace [1]
  - Related work on failure analysis:
    - Characteristics of job and task failures [2]
    - Failures and repairs of tasks and machines [3]
  - Other related work: [Reiss '12, Di '12, Liu '12]
- Our focus:
  1. **Performance impact**
  2. **Patterns and models**
  3. **Root causes**
- Limitation:
  - Black-box approach
  - Findings bounded to the trace

[1] J. Wilkes, More Google cluster data, Google research blog.

[2] X. Chen, C.-D. Lu, and K. Pattabiraman. Failure Prediction of Jobs in Compute Clouds: A Google Cluster Case Study. In *IEEE ISSRE*, 2014.

[3] P. Garraghan, P. Townend, and J. Xu. An Empirical Failure-Analysis of a Large-Scale Cloud Computing Environment. In *IEEE HASE*, 2014.

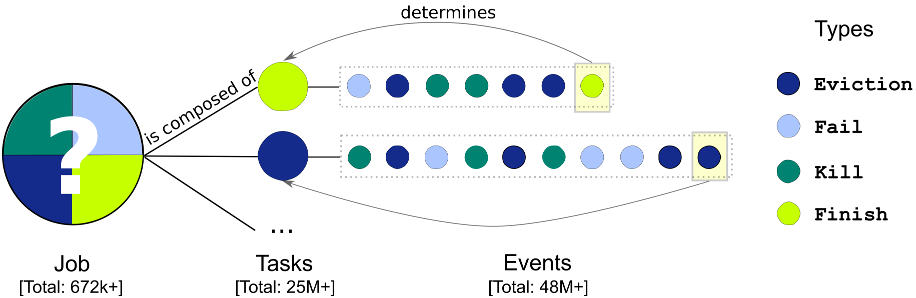
Trace Description

- Google cluster trace [1]
  - 29 days of workload
  - Heterogenous system and applications

[1] J. Wilkes, More Google cluster data, Google research blog. Nov 2011.

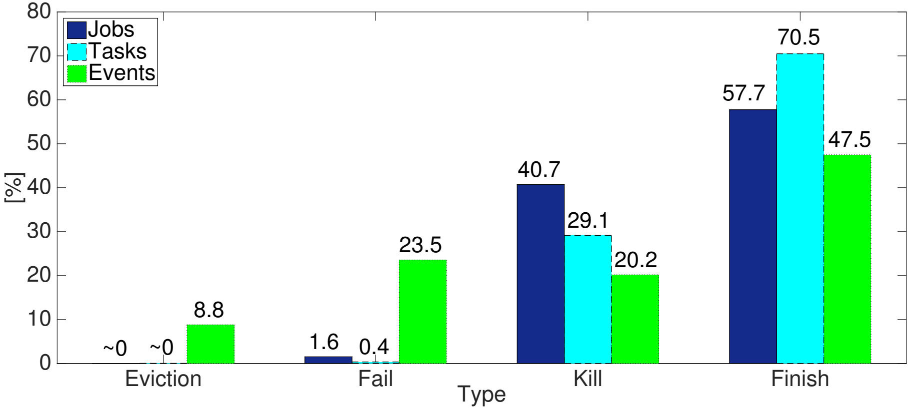
Trace Description

- Google cluster trace [1]
  - 29 days of workload
  - Heterogenous system and applications



[1] J. Wilkes, More Google cluster data, Google research blog. Nov 2011.

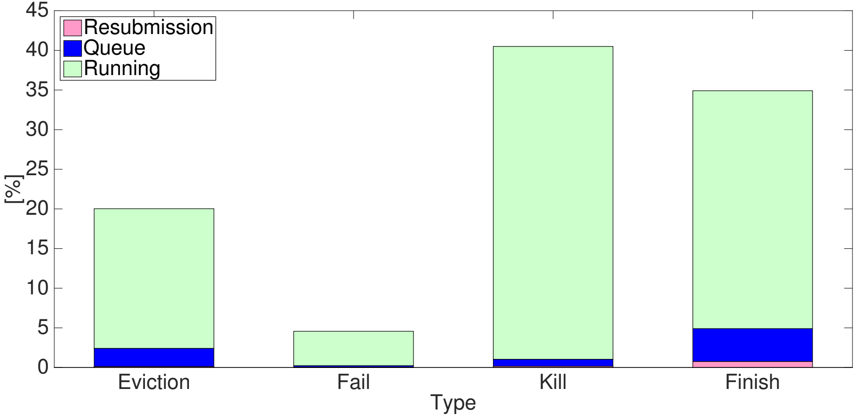
Motivations



- A lot of unsuccessful executions
- On jobs (42.3%), tasks (29.5%), events (52.5%)
- May lead to significant performance degradation

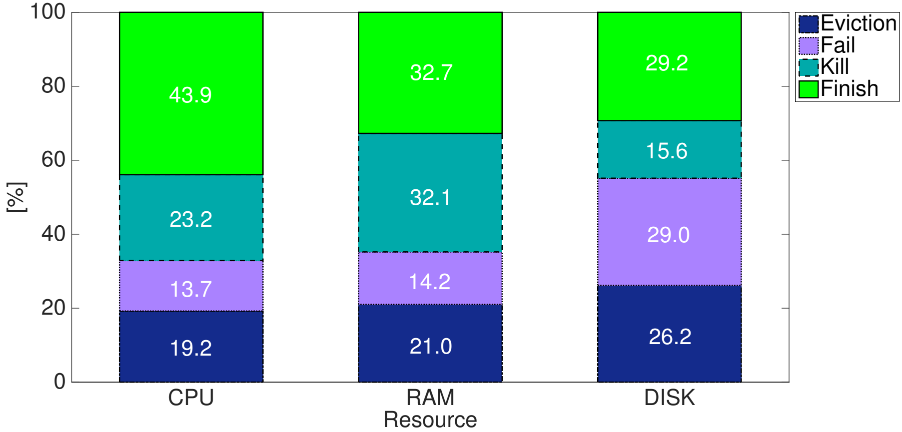
Outline

Time Waste



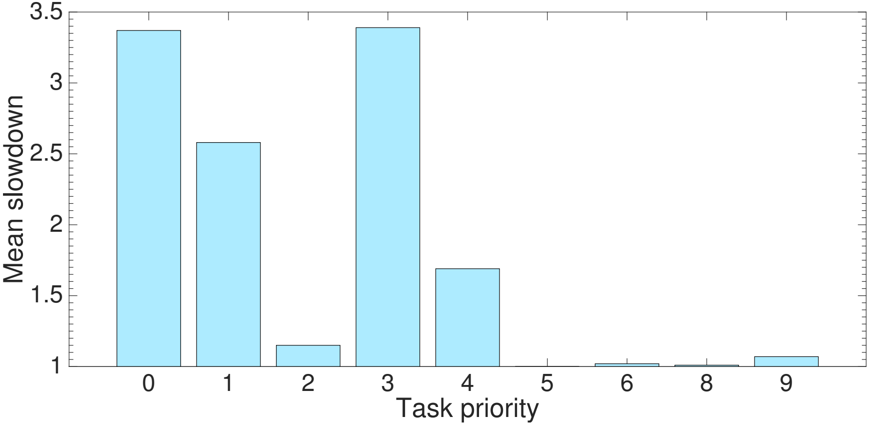
- Resubmission: termination to submission; Queue: submission to scheduling; Running: scheduling to termination/completion
- **More than 65% time wasted**
- Running time: 91%; Queue time: 8%

Resource Waste - Used Demand



- Used resources: AVG amount of resources used by tasks $\in [0, 1]$
- Resource demand $=$ amount of resources $\cdot$ running time
- **65% of used resource demand wasted**

Task Slowdown



- Slowdown $= \frac{\text{task running time}}{\text{running time last event}}$ [we consider `finish` tasks only]
- Priority: measure of the importance of tasks $\in [0, 11]$
- Low-priority tasks slowed down by 2.44X
- **Unsuccessful executions cause 2X slowdown on average**

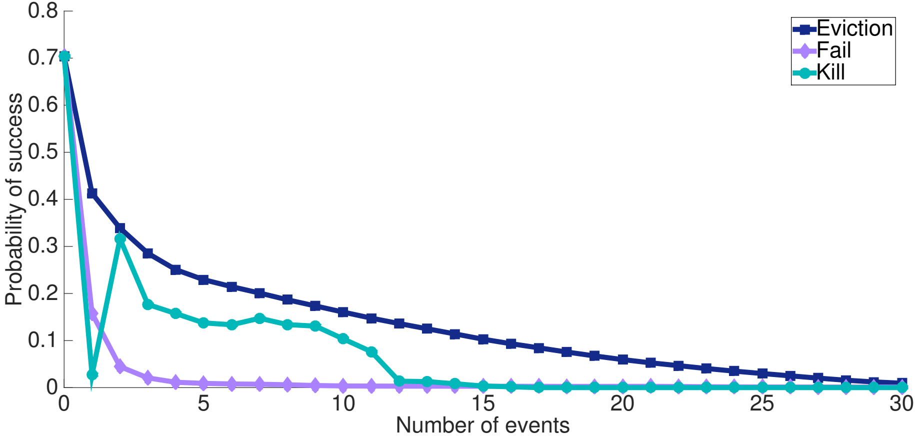
Outline

Task Types

Task type	Mean number of events				
	Overall (95^{th} p.)	Eviction	Fail	Kill	Finish
Eviction	2.372 (5)	2.094	0.259	0.004	0.015
Fail	3.130 (8)	0.350	2.700	0.020	0.060
Kill	2.516 (4)	0.302	1.175	1.023	0.016
Finish	1.094 (1)	0.061	0.008	0.011	1.014

- Types are determined by majority of events
- **Unsuccessful executions happen repeatedly on single tasks**
  - Up to: 828 `eviction`; 40393 `fail`.

Probability of Task Success



- Probability decreases with increasing number of events
- **Number of experienced events predicts probability of success**
  - Resource conservation policies can be applied to tasks based on number of events experienced

Dependencies between Jobs and Events

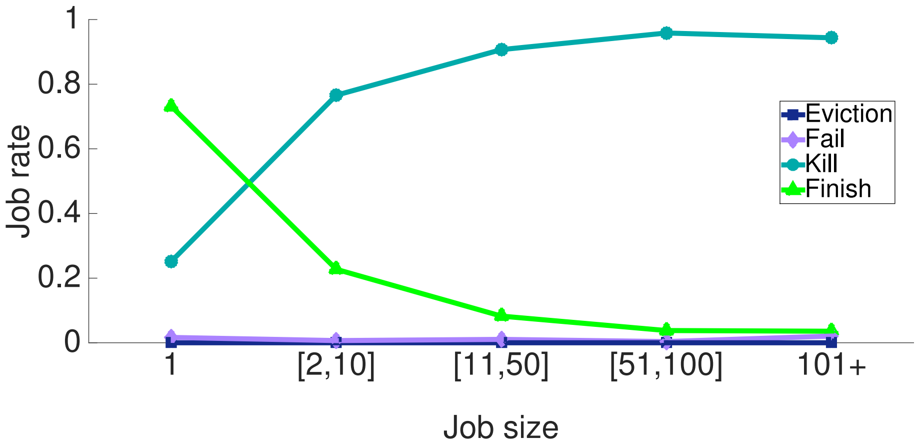
Event type (e)	Job type (j)			
	`Eviction`	`Fail`	`Kill`	`Finish`
`Eviction`	$4.04 \cdot 10^{-6}$	0.2706	0.7011	0.0283
`Fail`	0	0.5827	0.4159	0.0014
`Kill`	0	0.0637	0.9363	$2.46 \cdot 10^{-5}$
`Finish`	0	0.0053	0.9371	0.0576

- `Kill` events lead to `kill` jobs
- **Unsuccessful events lead to unsuccessful jobs**
- `Finish` events are not good predictors for `finish` jobs

Outline

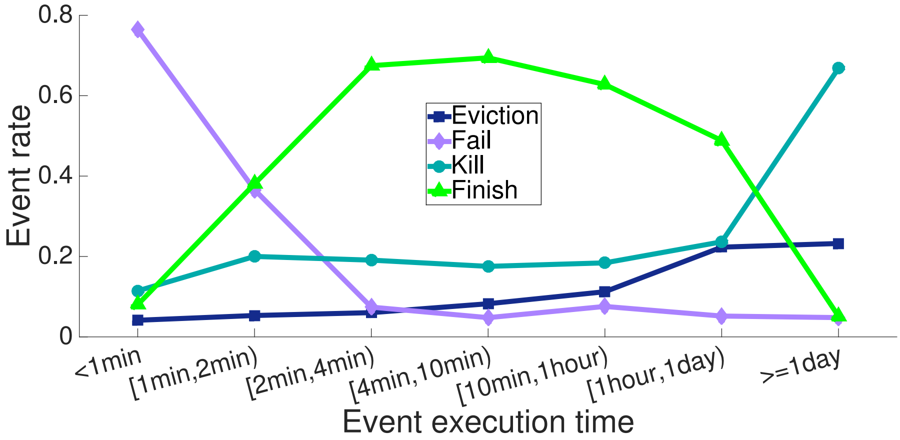
1. Performance Impact
    - Time waste
    - Resource waste
    - Task slowdown
2. Models and Patterns
    - Task types
    - Probability of task success
    - Dependencies between jobs and events
3. Root Causes
    - Jobs w.r.t. job size
    - Events w.r.t. execution time
    - Events w.r.t. machine concurrency

Root Causes for Jobs



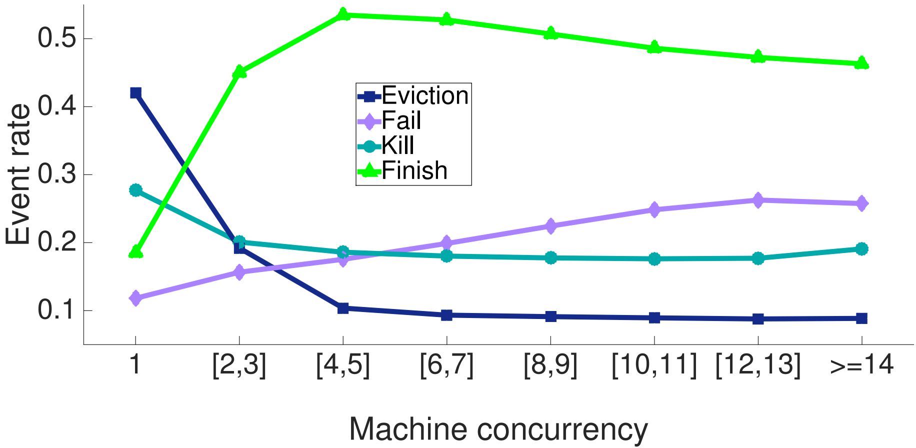
- Job size = # of tasks
- **High job size leads to unsuccessful jobs**
  - In contrast with latest trend of big-data applications

Root Causes for Events



- Many short-running tasks fail
- The scheduler tends to evict long-running tasks
  - **Preempting short-running tasks can mitigate the resource waste**

Root Causes for Events



- Machine concurrency = # tasks running on same machine
- `Fail` rate increases with machine concurrency
- **`Eviction` rate decreases with machine concurrency**

Conclusion

- We conducted an extensive analysis on **unsuccessful executions**
- **Key messages**:
  - Unsuccessful executions cause high performance degradation
    - Time waste: 65%
    - Resource waste: 65%
    - Task slowdown: 2X
  - Unsuccessful executions happen repeatedly on single tasks
  - Number of experienced events predicts probability of success
  - Unsuccessful events lead to unsuccessful jobs
  - High job size leads to unsuccessful jobs
  - Preempting short-running tasks can mitigate the resource waste
  - Eviction rate decreases with machine concurrency
- More results in the paper

Conclusion

- We conducted an extensive analysis on **unsuccessful executions**
- **Key messages**:
  - Unsuccessful executions cause high performance degradation
    - Time waste: 65%
    - Resource waste: 65%
    - Task slowdown: 2X
  - Unsuccessful executions happen repeatedly on single tasks
  - Number of experienced events predicts probability of success
  - Unsuccessful events lead to unsuccessful jobs
  - High job size leads to unsuccessful jobs
  - Preempting short-running tasks can mitigate the resource waste
  - Eviction rate decreases with machine concurrency

Andrea Rosà

andrea.rosa@usi.ch

http://www.inf.usi.ch/phd/rosaa

Università della Svizzera italiana (USI)
Faculty of Informatics

Università della Svizzera italiana

Faculty of Informatics