Failure Analysis of Jobs in Compute Clouds: A Google Cluster Case Study

Xin Chen, and **Karthik Pattabiraman**

University of British Columbia (UBC)

Charng-Da Lu, Unaffiliated

Compute Clouds

- Infrastructure as a Service

Compute Clouds

Data & Storage Clouds

- Access to computational resources.
- Increasing cloud adoption in the scientific community.

Application Failures

- High failure rate in cloud clusters
- Isolation of resources not guaranteed
- Resources and power wasted in failures

```
Application application_1392853856445_0900 failed 2 times due to AM
Container for appattempt_1392853856445_0900_000002 exited with exitCode: 143 due to:
Current usage: 337.6 MB of 1 GB physical memory used; 2.2 GB of 2.1 GB virtual memory
```

Pervious Studies on Failures

- System Failures
  - HPC [Martino et al., DSN 14'], [El-Sayed et al., DSN 13']
  - Cloud hardware reliability [Vishwanath et al., SoCC 10']
- Application Failures
  - Hadoop [Kavulya et al., CCGrid 10'], [Ren et al., IISWC 12']

No published application failure study on a generic production cloud with heterogeneous workloads

Research Question

- What are the characteristics of job failures in a production compute cloud?

- Technical Challenges
  - A large number of heterogeneous applications
  - Different types of failures
  - Different factors contributing to failures

- Other challenges
  - Few data-sets of production clouds, missing information

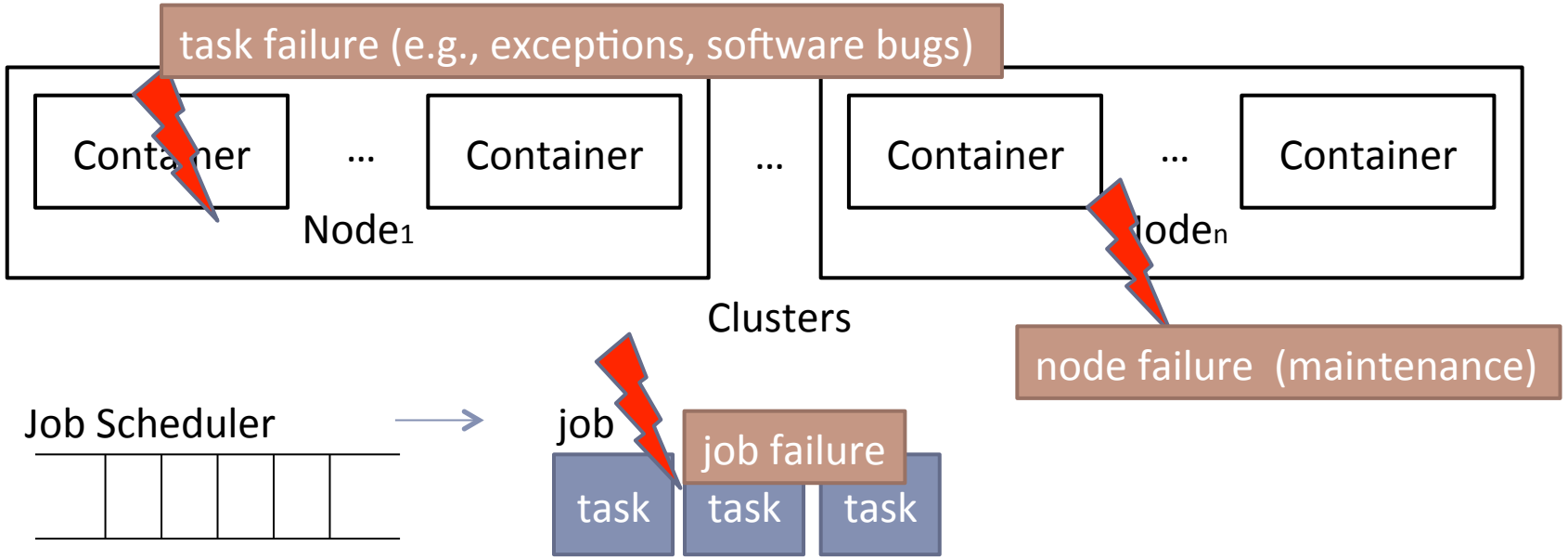
Dataset used in this paper

- **Google cluster workload traces [Wilkes2012]**
  - Originally released for job scheduling studies
  - Publicly available:
    - https://code.google.com/p/googleclusterdata/
  - One month data on production cluster of 1,2500 nodes
  - Includes both failure data and periodic resource usage data

- **Hides important information such as nature of jobs, users, spatial locations of tasks etc. due to privacy reasons**
  - Limited in the kinds of studies we can do
  - Root causes of failures is not provided

First paper to analyze job & task failures in Google cluster data

Google Clusters: Failures

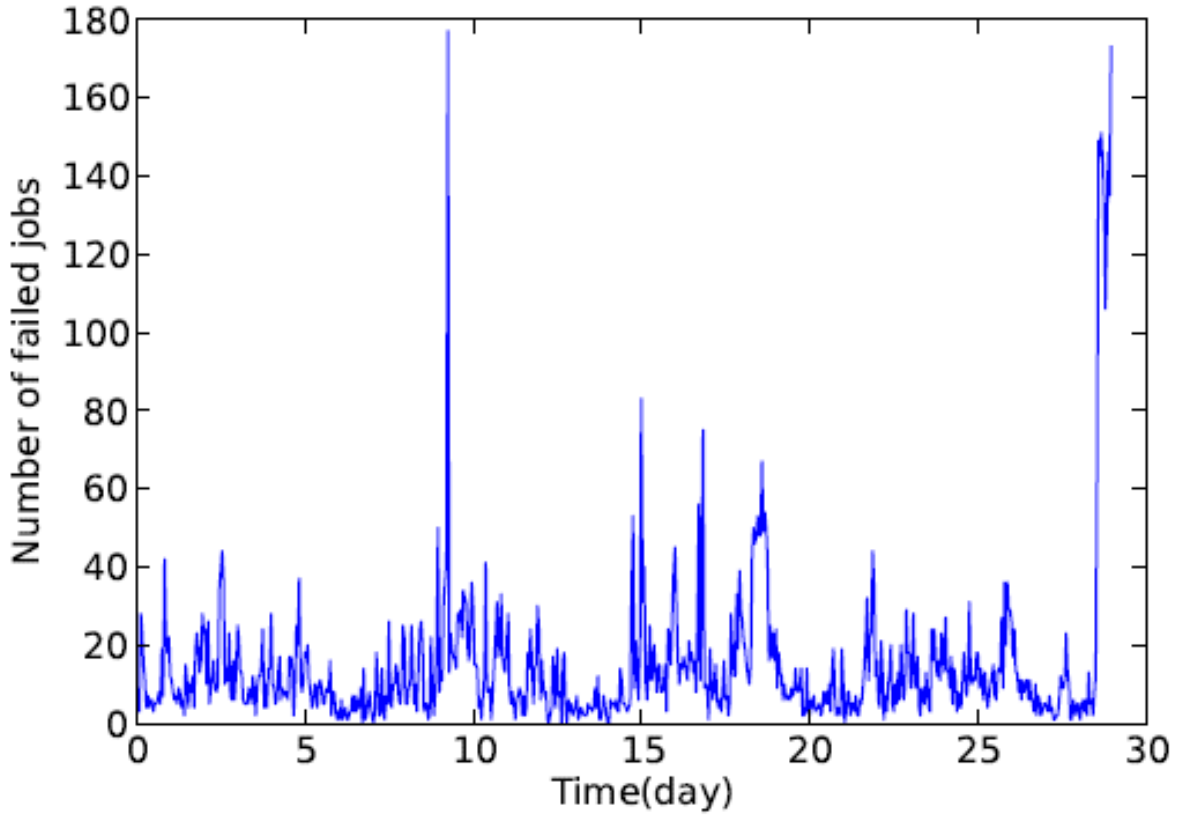


- Production jobs (e.g., web services)
- Batch jobs

- Around 680 users
- 670,000 jobs
- 48 million tasks
- 12,500 nodes for 1 month

- Records we use
  - Job failures, task failures, and node failures
  - Other attributes and usage of jobs, tasks and nodes

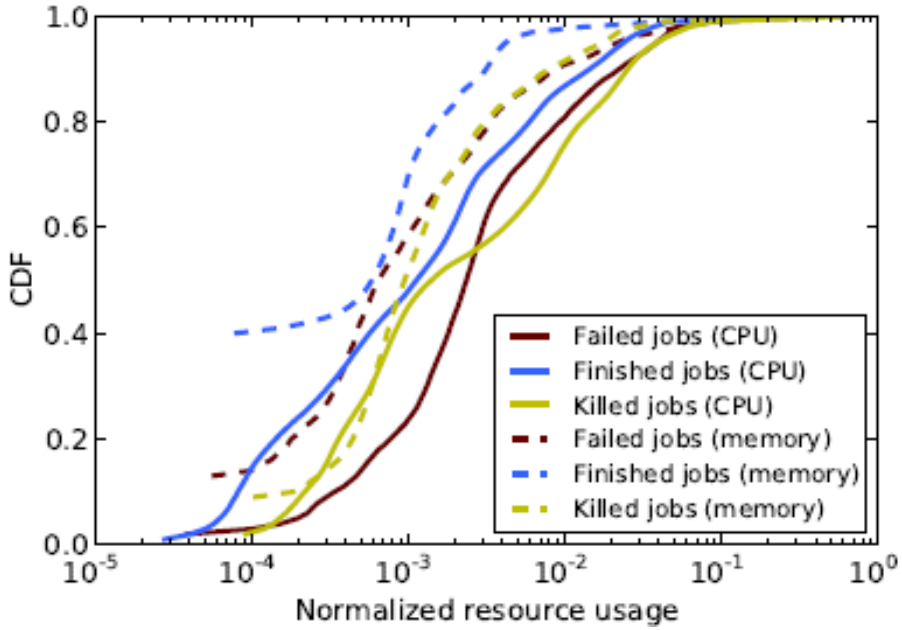
Job Failures: Google Data



- An average of 14.6 jobs fail in an hour > 10,000 job failures
- Failed jobs constitute about 1.5% of the total jobs (670,000)

Why study job failures ?

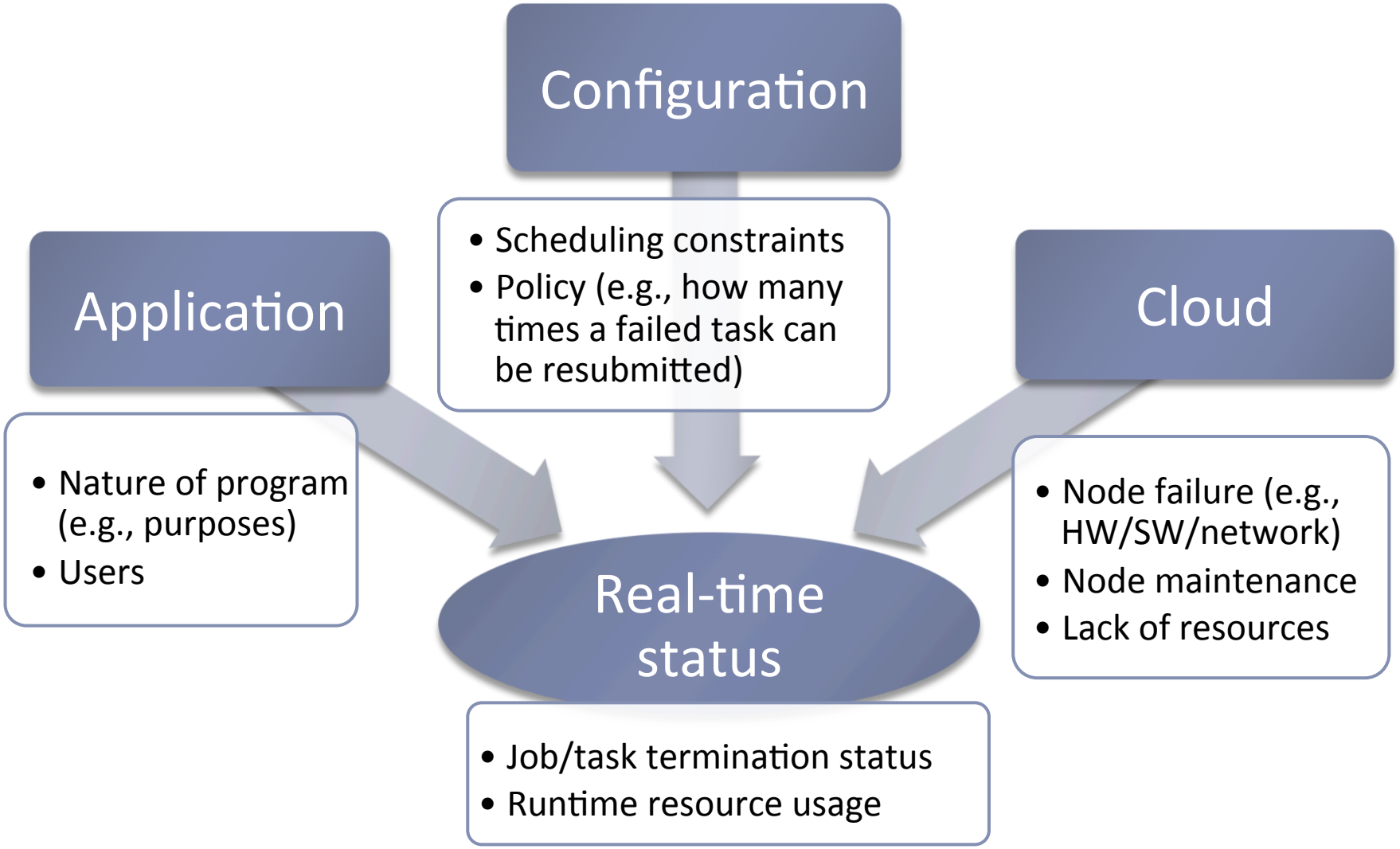
- Normalized CPU or memory (done by Google)



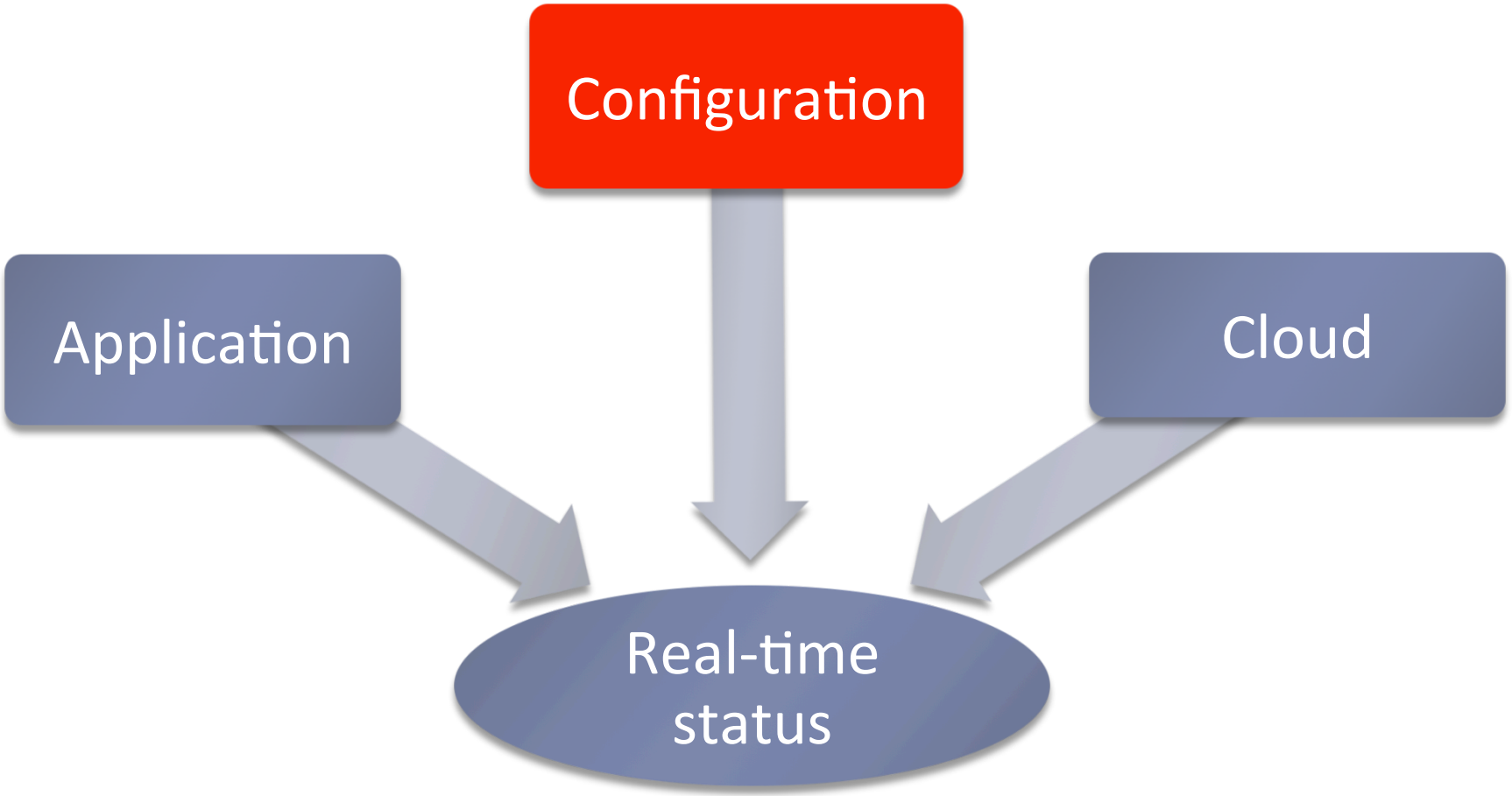
- Overall usage: failed jobs Vs. finished jobs

  CPU – 2.5X memory – 6.6X

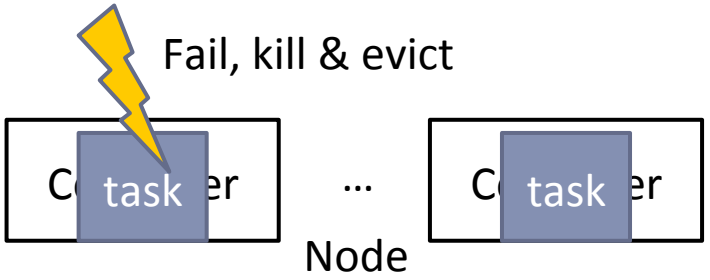
Factors leading to Cloud Application Failures



Factors leading to Cloud Application Failures

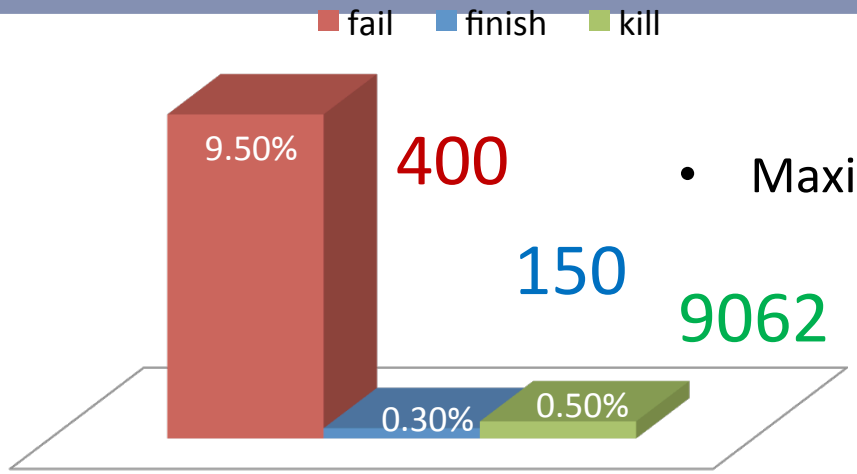


Configuration Factor: Task Resubmissions



- Task resubmission

Frequent task resubmissions may waste resources and energy, particularly in failed and killed jobs.



400

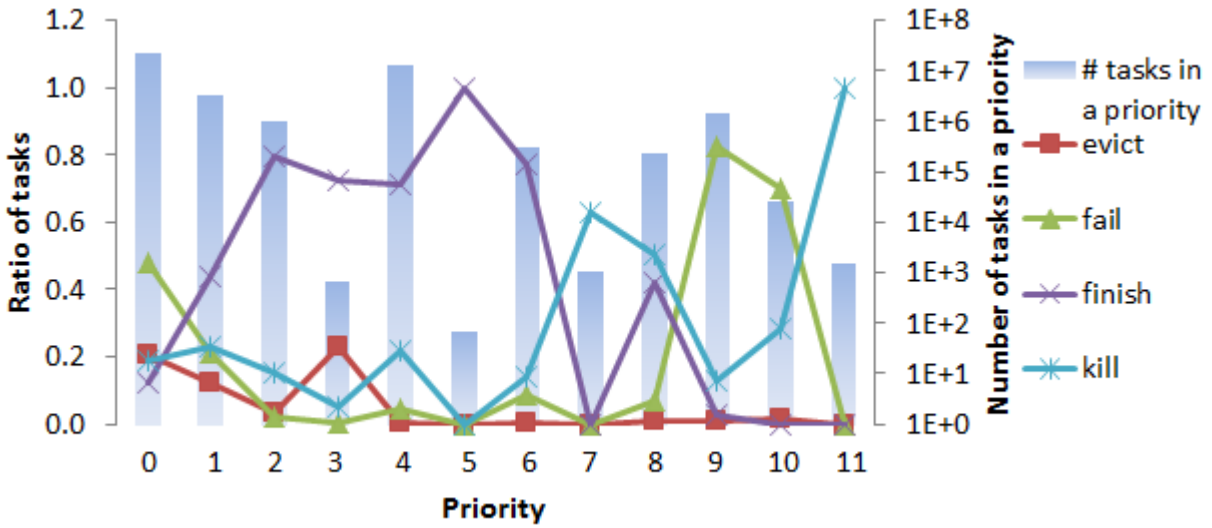
150

9062

- Maximum resubmissions

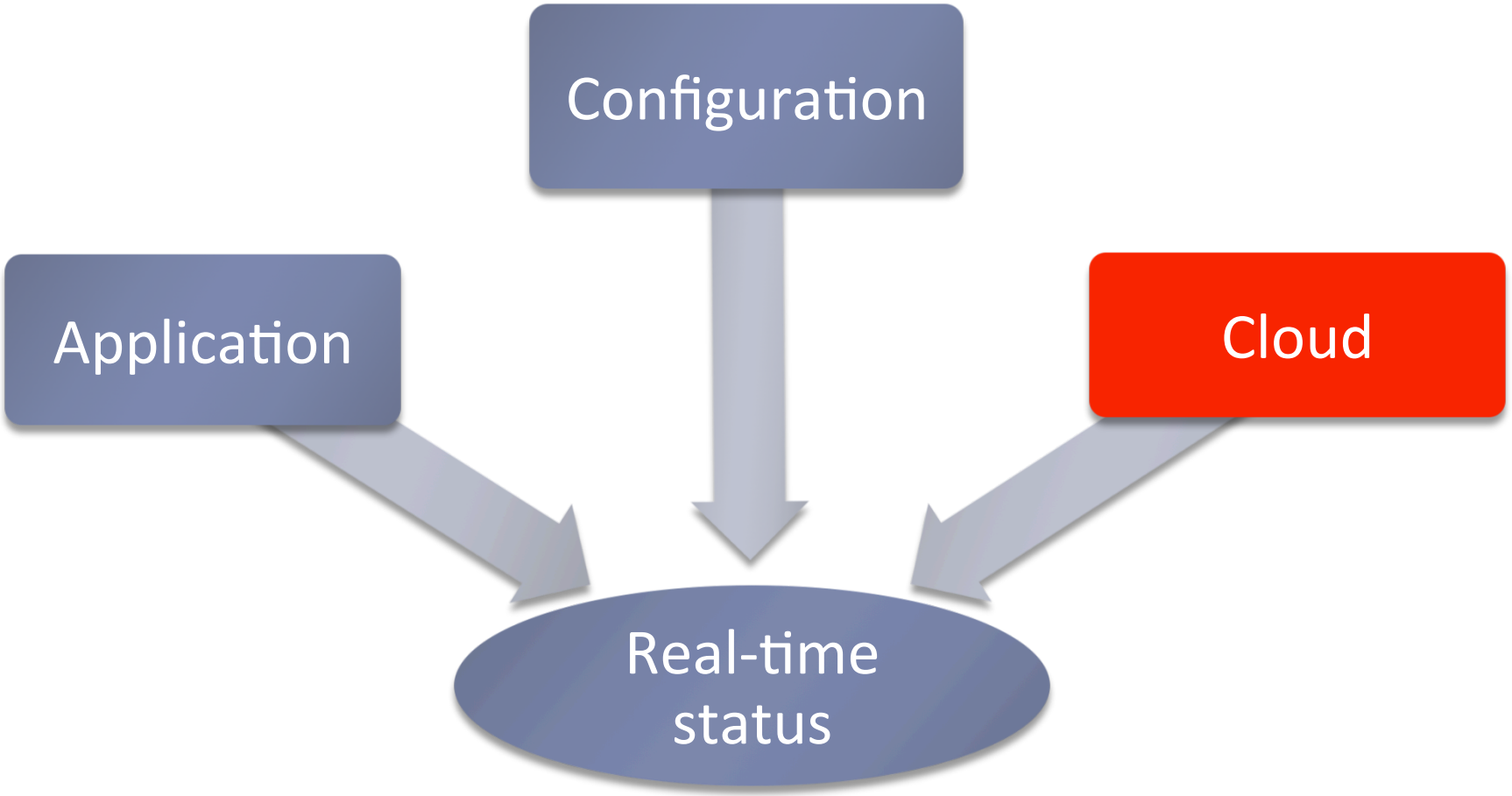
Configuration Factor: Priority

Priority determines the nodes assigned to the task.

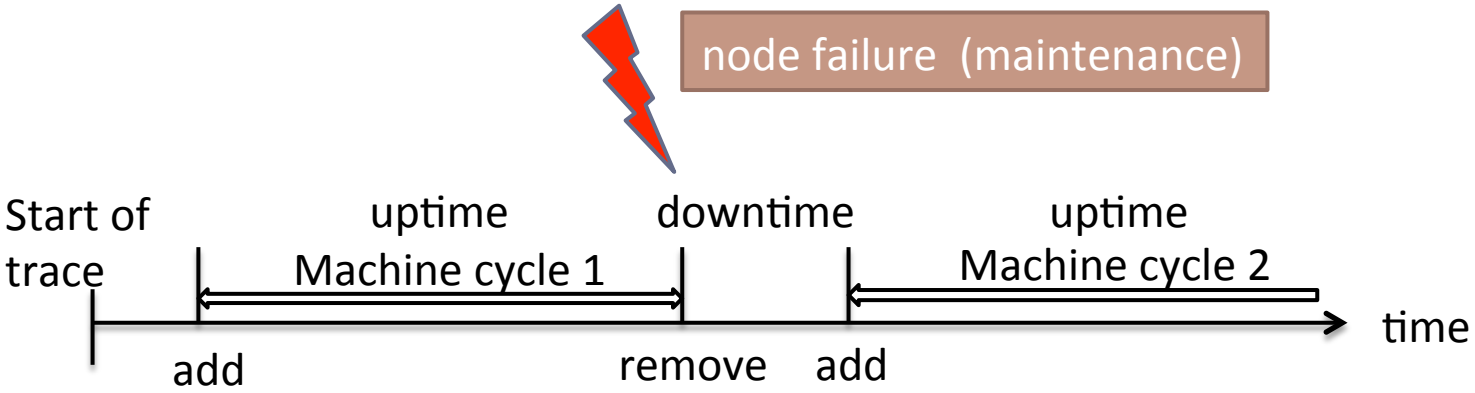


- Low-priority and high-priority jobs experience high failure rate
  - Result holds even when disregarding resubmissions

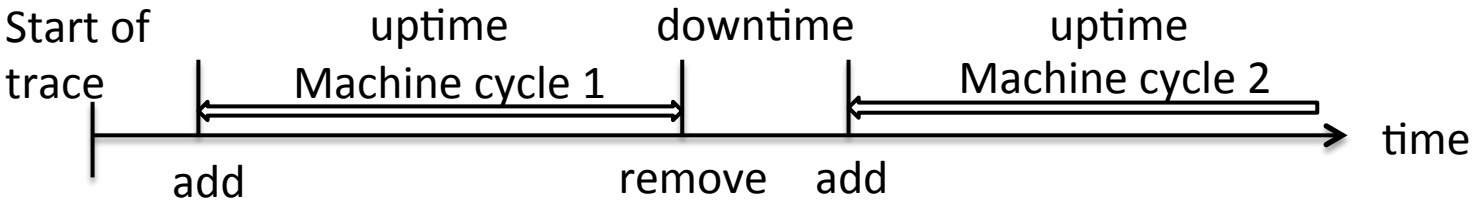
Factors leading to Cloud Application Failures



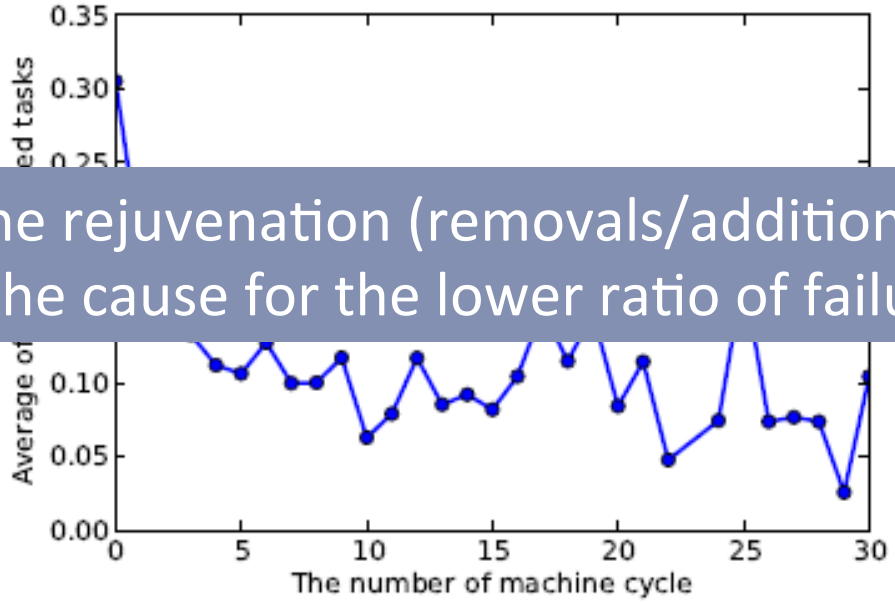
Cloud Factor: Node Failure



Cloud Factor: Node Failure (Cont.)

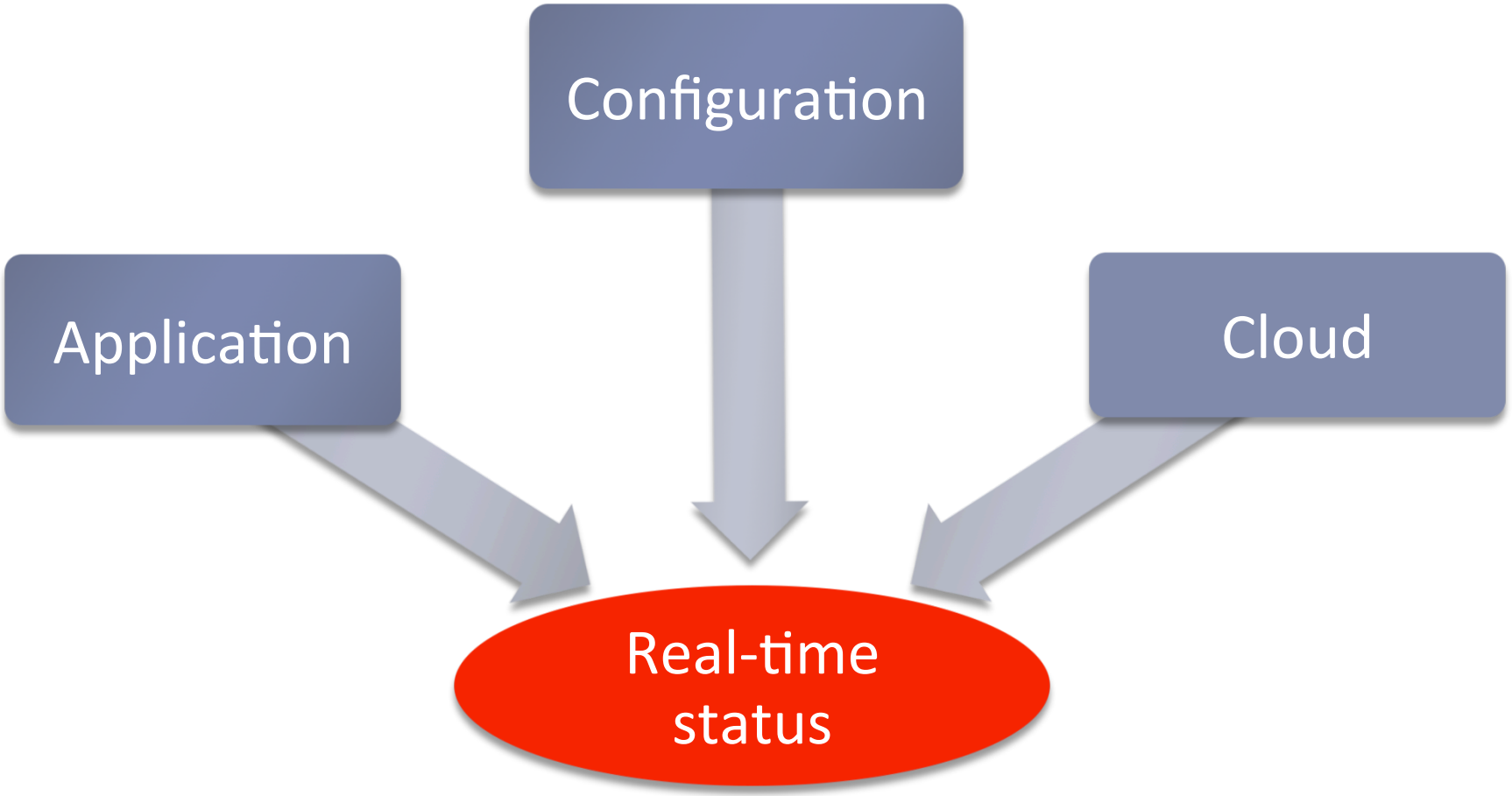


- Average of failed task ratio VS number of machine cycles



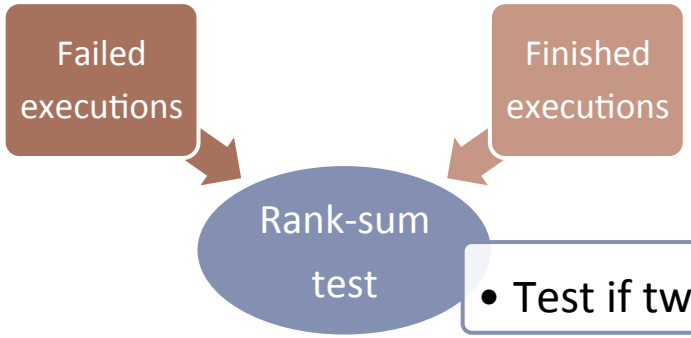
Machine rejuvenation (removals/additions) may be the cause for the lower ratio of failures

Factors leading to Cloud Application Failures



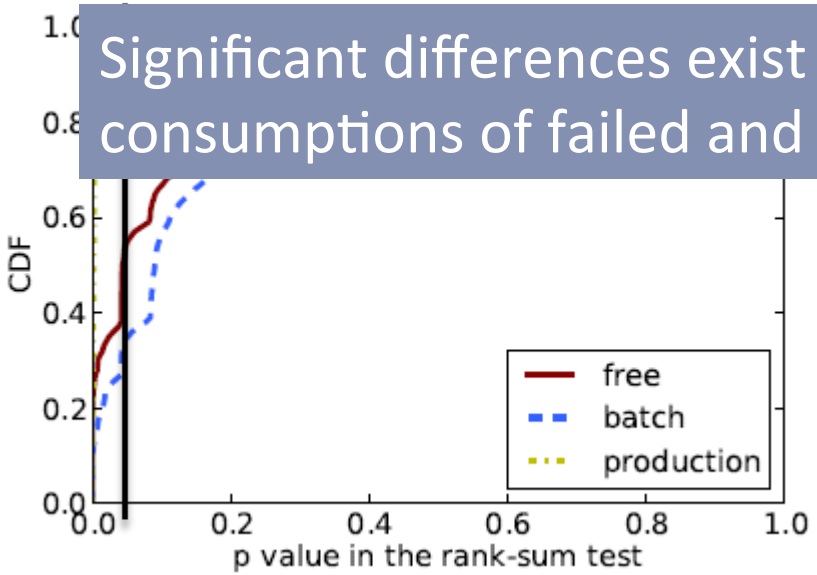
Status Factor: Resource Usage

- Distinctions in the task resource usages



- Test if two samples significantly differ

- CPU usage



Significant differences exist between the resource consumptions of failed and finished tasks of same job

0.05

- Batch: 34.8%
- Production: 93.2%

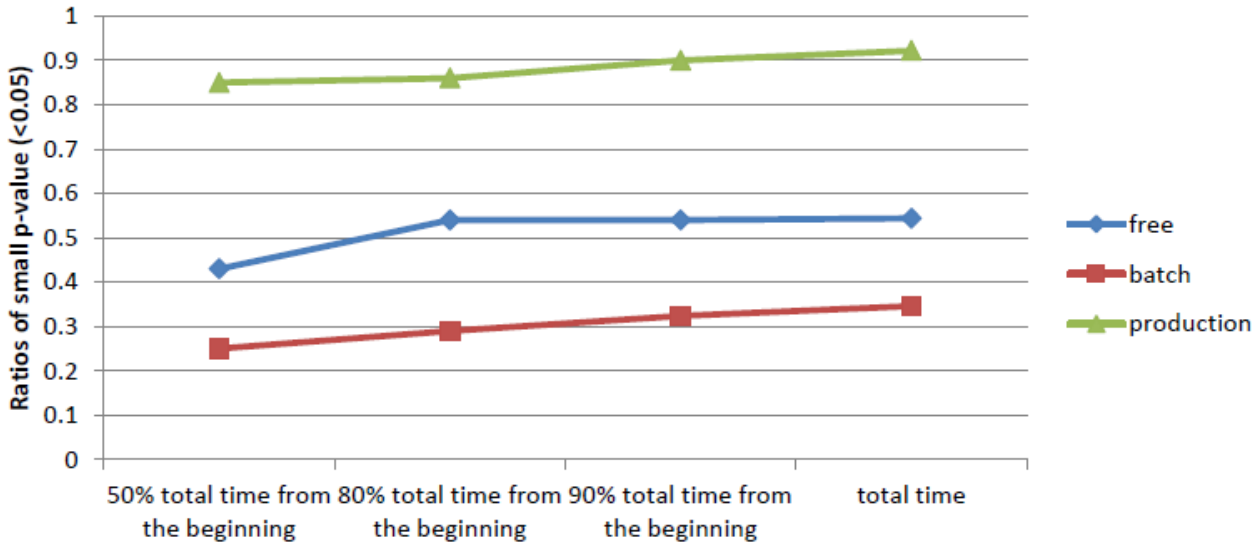
* Free: low priority batch

Early Failure Manifestation

- Differences between failed and finished executions manifest much earlier than the termination.

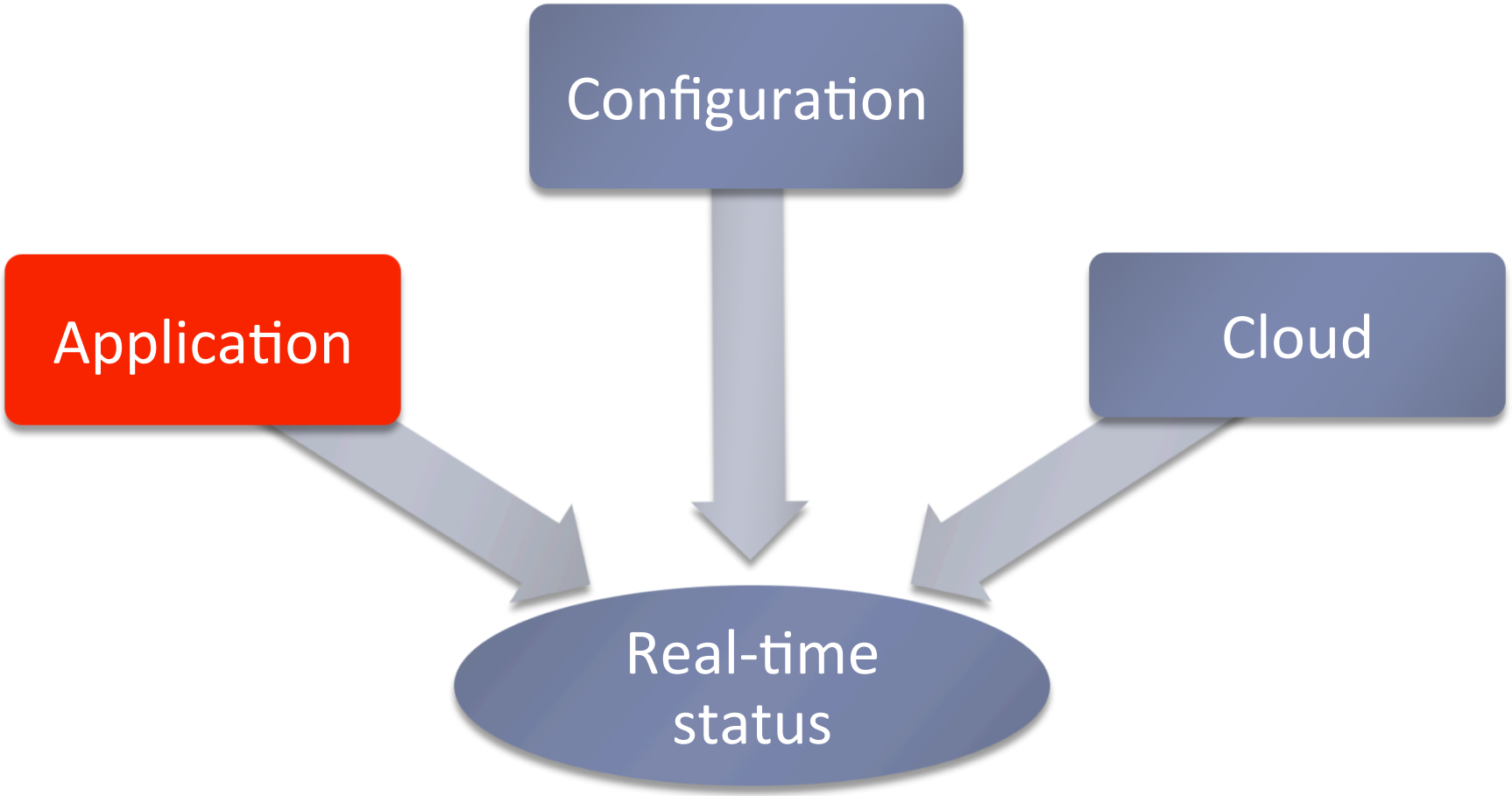
Statistical test

- Test if two samples significantly differ



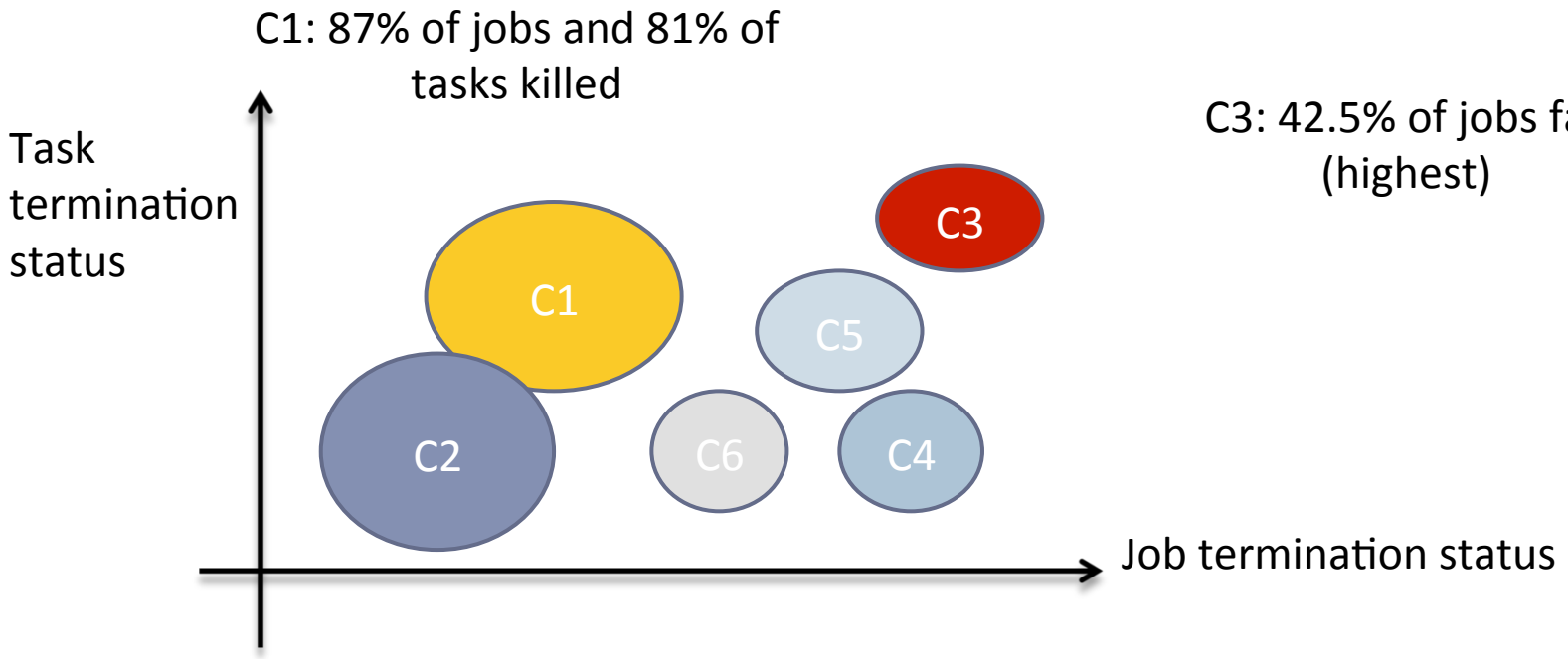
- The differences in resource consumption are significant even halfway into the job ➔ potential for failure prediction

Factors leading to Cloud Application Failures



Application Factor: Users

- K-means clustering on termination status (fail, finish, kill, evict)



- Correlations between failures and attributes help identify features to indicate high ratios of failures.

clusters)

Summary of Findings

- **Significant resource consumption due to failed jobs**
- **Job and task failures manifest differently**
  - High number of task resubmissions in failed jobs
  - Both low and high priority jobs - 3 times as many failures
  - Node maintenance and update improve reliability
- **Differences in resource consumption exist**
  - Many of the jobs have significant differences between failed and finished task submissions
  - Differences manifest even halfway into a long job's execution
- **User profiles can be clustered into 6 dominant groups**

Implications

Failure Prediction

Scheduling updates

Anomaly Detection

- Early failure prediction at **infrastructure provider** level
  - A lot of resource usage by failed jobs
  - Over submitted task executions
  - Significant potential for early prediction
- Removals or updates of containers (rejuvenation)
- User based clustering used for anomaly detection

Threats to Validity

- Internal threats
  - Anonymized names of users and applications
  - No information on root causes
  - Normalized resource usage

- External threats
  - Limited to Google clusters

Related Work on Google Failure Data

- [Di et al., ICPP 13']
  - Job-specific information and the termination statuses of tasks.
  - Our paper: unique job IDs, and correlation between the clusters of failures with user profiles
- [Guan et al., SRDS 13']
  - Very low average correlations of raw resource usage to failures.
  - Our paper: much higher correlations and more significant differences between failures and successful terminations
- [Garraghan et al., HASE 14']
  - The node and task failures' statistical distributions
  - Our paper: Job and task failures
  - Do not use job and cloud system attributes to understand the correlations between job failures and attributes.

Conclusion

- Cloud applications require high reliability
- Failure characterization study of Google data
  - Factors: application, cloud, configuration, and real-time status.
  - Implications for prediction, scheduling and anomaly detection
- Future work
  - To analyze a more comprehensive set of failures in a wider range of cloud systems
  - To perform comprehensive failure prediction [RSDA'14]

Contact me for the data/questions: karthikp@ece.ubc.ca