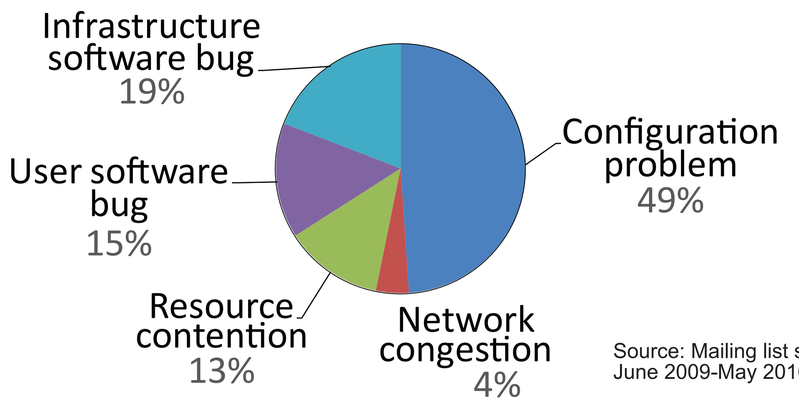
RIKA: PRACTICAL PEER-COMPARISON FOR PROBLEM LOCALIZATION

Soila Kavulya, Jiaqi Tan, Xinghao Pan, Rajeev Gandhi, Priya Narasimhan (CMU)

PROBLEM STATEMENT

- Diagnosis is challenging in large-scale systems
  - Complex node dependencies
  - Lots of information to sift through
- Goal: Automated problem diagnosis
  - Leverage instrumentation and statistical analysis
  - Analyze OS and application-level IO
- Target systems: Hadoop, and production VoIP system

Causes of problems in a big data system

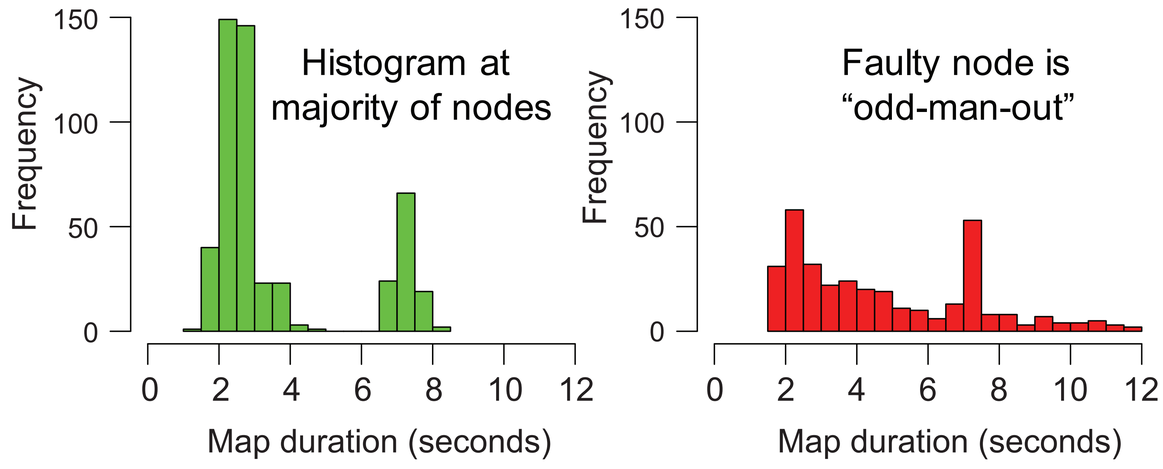


Source: Mailing list survey from June 2009-May 2010

ANOMALY DETECTION

- Group request flows based on peer properties
  - Compare statistics of states across peers
  - Application-level: Histograms of state durations
- Assume majority of request flows are correct
  - Compare histograms of state durations
  - Faulty node is "odd-man-out"

Map State Durations



PEER-COMPARISON FOR DIAGNOSIS

Rika (noun):
Swahili word for age-set or peers. Peers undergo rites of passage (birth, initiation, marriage) at similar times.

Rika: Practical Peer-comparison for Diagnosis

- Identifies peers (nodes, request flows) in system
- Diagnoses problems by identifying "odd-man-out"

Peer Similarity	Age-set	Diagnosis
Temporal	Similar ages	Events around same time
Spatial	Same geography	Events on same node
Phase	Birth→Initiation→…	Map→Shuffle→Reduce
Context	Same gender, clan	Same workload, h/w type

Extracting Peer Properties from Logs

Temporal similarity: timestamps **Context similarity: tasks**

2009-03-06 23:06:01,572 INFO org.hadoop.mapred.ReduceTask: attempt_200903062245_0051_r_000005_0 Scheduled 10 of 115 known outputs (0 slow hosts and 105 dup hosts)

Phase similarity: Map→Reduce

2009-03-06 23:06:01,612 INFO org.hadoop.mapred.ReduceTask: Shuffling 2 bytes (2 raw bytes) from attempt_200903062245_0051_m_000055_0 …from ip-10-250-90-207.ec2.internal

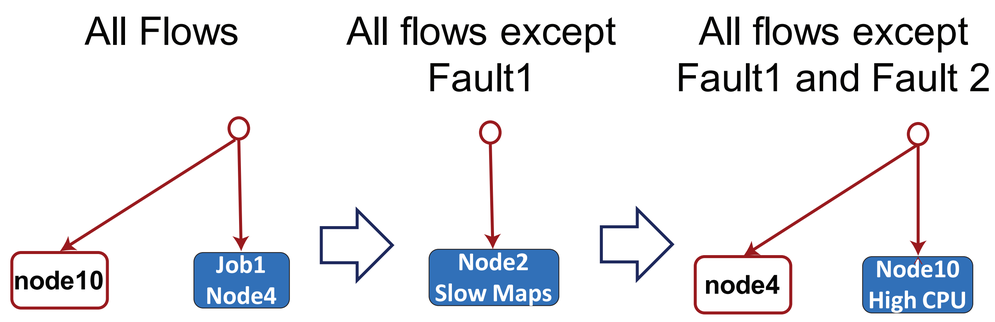
Spatial similarity: hostnames

PROBLEM LOCALIZATION

- Anomaly detection may indict multiple suspects
- Problem localization reduces false positives

Problem Localization Approach

1. Extract attributes from labeled flows
   - e.g., node names, node types
2. Localize problem using Bayesian algorithm
   - Identify attributes most correlated with problem
   - Rank multiple independent problems
3. Identify anomalous resource-usage metrics
   - Annotate requests with resource-metrics
   - Identify metrics most correlated with problem



Fault1: Job1, Node4,
Fault2: Node2, Slow maps
Fault3: node10 High CPU

Intel Science & Technology Center for Cloud Computing

Carnegie Mellon University · Georgia Tech · intel · PRINCETON UNIVERSITY · UC Berkeley