PROBLEM LOCALIZATION IN HADOOP USING DRACO
S. P. Kavulya, J. Tan, K. Joshi (AT&T), M. Hiltunen (AT&T), S. Daniels (AT&T), R. Gandhi, P. Narasimhan (Carnegie Mellon University)

PROBLEM STATEMENT

- Localizing problems is challenging in large systems
  - Complex node dependencies
  - Lots of information to sift through
- Goal: Automated problem localization
  - Identify application and infrastructure-level problems
  - Use unmodified logs in production systems
- Target: Hadoop - open-source MapReduce framework

CONSTRUCT FLOWS FROM HADOOP LOGS

Hadoop Log Snippet

```
Timestamps
Task properties
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)

2009-03-06 23:06:01,612 INFO org.hadoop.mapred.ReduceTask: Shuffling 2 bytes (2 raw bytes) from attempt_200903062245_0051_r_000005_0 from ip-10-250-90-207.ec2.internal
```

Extract Features from Logs

Flow Schema (JSON)

MapReduce: {
  "events": [{"Map": {
    "primary-key": "MapID",
    "join-key": "MapID",
    "next-event": "Shuffle"}, ...

CONSTRUCT FLOWS FROM HADOOP LOGS

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

PROBLEM LOCALIZATION APPROACH

ANOMALY DETECTION

- Label task flows as successful or anomalous
- Use task exceptions to identify failed tasks
- Use survival analysis to identify slow tasks
- Why survival analysis?
  - Can cope with incomplete tasks using censoring
  - Supports regression (e.g., scale duration by I/O size)

1. Extract attributes from labeled flows
2. Localize problem using Bayesian algorithm
3. Identify anomalous resource-usage metrics

Causality: Map Reduce

Database Records

```
<time=20090306230601, task=ReduceTask, taskid=attempt_2009x_r_y
```

Stitch causal traces

```
Block Read → Map → Shuffle → Reduce → Block Write
```

Causes of problems in a Hadoop system

- Infrastructure software bug: 19%
- User software bug: 15%
- Resource contention: 13%
- Network congestion: 4%

Source: Mailing list survey from June 2009-May 2010

Map Task Durations

- Anomalous Tasks
  - Prob(Duration>3) < 0.05

Problem Duration (seconds)

- Task Duration (seconds)
  - Probability(Duration > x)

Fault1:
  - Job1
  - Node4

Fault2:
  - Slow maps
  - Node2

Fault3:
  - High CPU
  - Node10

1. Extract attributes from labeled flows
2. Localize problem using Bayesian algorithm
3. Identify anomalous resource-usage metrics

NoSQL Database

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