**OVERVIEW**

Diagnosing performance problems in distributed systems is difficult. Root cause of many performance changes can be localized by:

- Comparing path/timing of requests (i.e., their flow) from both periods
- Identifying changes in the flows and ranking them by their contribution to the performance change
- Can diagnose 57% of all problems observed in a prototype system

Comparison of flows implemented in a diagnosis toolset

- Used to diagnose problems in Ursa Minor & Google services
- Also contains mechanisms to browse request flows
  - Helps identify consistently slow behavior and design problems

**SPECTROSCOPE**

- Toolset for diagnosing problems in distributed systems
- Operates on categories, which contain same-structure requests
- Allows for visualizing categories, browsing request flows, and comparing request flows

**COMPARING REQUEST FLOWS:**

<table>
<thead>
<tr>
<th>Precursor</th>
<th>Mutation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read 20 µs Cache 100 µs Start 000 µs</td>
<td>Read 20 µs Cache 100 µs Start 000 µs</td>
</tr>
<tr>
<td>Reply 100 µs</td>
<td>Reply 100 µs</td>
</tr>
</tbody>
</table>

Rank: 1
Requests: 7000
Type: Structural

<table>
<thead>
<tr>
<th>Precursor</th>
<th>Mutation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write 10 µs Start 000 µs End 000 µs</td>
<td>Write 10 µs Start 000 µs End 000 µs</td>
</tr>
<tr>
<td>Reply 100 µs</td>
<td>Reply 100 µs</td>
</tr>
</tbody>
</table>

Rank: 2
Requests: 5000
Type: Response-time

- Uses statistical tests and heuristics to compare flows between a non-problem and problem period and identify categories containing:
  - Response-time mutations: Requests that have changed in cost
  - Structural mutations: Requests that changed in the path they take
  - Candidate precursors: How the mutations might have been serviced during the non-problem period
- Root cause of performance change automatically localized by:
  - Comparing structural mutations to their precursors
  - ID'ing edges of response-time mutations with higher cost
  - ID'ing low-level parameters that best separate mutations/precursors

**BROWSING REQUEST FLOWS**

- Presents a PDF of response times and allows ranges to be selected
- Shows categories that fall in selected ranges with statistical info

- Toolset for diagnosing problems in distributed systems
- Operates on categories, which contain same-structure requests
- Allows for visualizing categories, browsing request flows, and comparing request flows

**END-TO-END TRACES**

End-to-end traces utilized to browse and compare flows

- Yields request-flow graphs, showing structure and timing of individual requests
- Captures low-level parameters, e.g. client and function parameters
- Adds minimal overhead and is used in production systems

**DIAGNOSIS OF REAL PROBLEMS**

- Spectroscope used to diagnose problems in Ursa Minor & Google
  - A few are described below

**PERIODIC SPIKES**

- Example of: How comparing flows can show that the problem is not due to the distributed system, but due to external factors
- Periodic spikes in run times seen in regression tests
- Comparing request flows revealed no mutations
- This led developers to suspect the client
- Visualization of request intra-arrival times shows client sends requests at a slower rate when spikes are observed
- Currently suspect this is due to externally initiated backup activity

**CREATE BEHAVIOR**

- Example of: Diagnosing a performance degradation over time
  - Comparing first 1000 requests to last 1000 showed:
    - Degradation from growing loop between metadata server & storage node

**INTER-CLUSTERS PERFORMANCE AT GOOGLE**

- Example of: Ruling out distributed system as the root cause
  - Loadtests run on a distributed service in two different datacenters differed in performance, but developers expected it to be similar
  - Spectroscope revealed many response-time mutations
    - Both within service and in its dependencies
    - Suggested problem was pervasive to the slower loadtest’s datacenter
  - Root cause: Problem w/shared bigtable in slower datacenter