Constructing Workflow-Centric Traces in close to Real Time for the Hadoop File System

Neel Bhalla, Lexington High School
Adviser: Prof. Raja Sambasivan, Tufts University
Motivation

• We can’t debug distributed services – complicated, with 1000’s of nodes and services

• Machine-centric techniques are insufficient
  • E.g., GDB, GProf, perf. counters, strace, Dtrace

• To find performance issues in distributed systems we need to do tracing

Debugging services in a distributed system is a complicated problem
Workflow-Centric Tracing

- Every request involves a workflow
- Workflow: Structure & timing of work done to process them
- Structure: Order, concurrency & synchronization
- Facebook collects 1 Billion Traces/day

Challenge with tracing: real-time construction

Basic features:
- Trace point: Unique name / low-level params (e.g., CPU util., function vars)
- Context: request ID & logical clock
Problem: Trace Construction is Slow

• Existing tracing systems operate in mostly non-real time batch mode (e.g. Facebook ~ 1 day turnaround to find problems)
  • Data and computation intense

• Use cases
  • Anomaly Detection in distributed systems
  • Cyber Intrusion Detection
  • Failover Management
  • Performance Issues

• Developers use stream processing to query continuous data streams and react to important events (e.g. Apache Flink, Timely Dataflow)
Proposed System For Real Time Trace Reconstruction – **Altair**

- Several Non-Real Time Approaches
  - Google’s Dapper, Facebook’s Canopy, Brown’s X-Trace
- ETH’s Strymon [1] is a real-time stream processing system
  - It builds on Naiad Streaming Timely Dataflow
  - Traces are modelled as trees
- General infrastructure tracing frameworks (OpenTracing, X-Trace) represent traces as DAG (Directed Acyclic Graphs)
  - DAG can capture concurrency
- Strymon’s approach can be modified to incorporate DAG’s
- Developed real time tracing system for distributed systems - Altair

---

We developed a new system for processing traces called Altair

[1] Chothia, et. All, Online Reconstruction of Structural Information from Datacenter Logs, EuroSys ’17
Timely Dataflow

- Framework for writing dataflow programs

- Dataflow programming is a programming model in which the computation can be represented as a directed graph: The data flows along edges, while the computational logic in the vertices transforms it.

- The messages flowing along edges are annotated with timestamps.

Timely Dataflow used as the streaming framework

Reference: Naiad: A Timely Dataflow System
Evaluation: HDFS Tracing

- Mass Open Cloud runs OpenStack
- Access to 10 compute instances, Altair runs on 8 instances, 2 instances run Trace compression
- Instrumented HDFS and X-Trace server
  - HDFS (Hadoop Distributed File System) is distributed file system, used with MapReduce applications in datacenters
  - Performance of HDFS can directly affect the performance of jobs
- Event Test Data: 3000 Traces, ~350 graph nodes/traces, 0.525 Million event/Epoch
- Streaming simulator to generate event stream, replay and add anomalies, latency in event stream

Acknowledgement: Mass Open Cloud for their support
Results: Altair

The Altair approach is scalable with more than 99% parallelization.

Altair Events Processing Throughput

- Altair
- Amdhal Law 99% parallel portion (p=0.99)

- 0.23 Million events/second
- 0.06 Million events/second

Facebook collects 11 Million events/second (1 Billion Traces/day ~ (1000 nodes/trace))

Amdhal Law: Maximum expected improvement to a system when only part of the system in the parallelized

\[
\text{Speedup} = \frac{1}{(1-P) + \left(\frac{P}{N}\right)}
\]
Altair Use Cases

- Anomaly Detection in distributed systems
- Cyber Intrusion Detection
- Failover Management
- Performance Issues
Altair Use Case: Anomaly Detection

• Anomaly Detection application will be run continuously in two steps
  • The first step involves designing the Bloom filter. The design of the Bloom filter requires a representative set of graphs. This step is not real-time.
  • We are proposing graph clustering approach to extract template representative graphs that would be programmed into Bloom Filter
  • Second Step with Altair will run in real time
  • Any anomaly traces will be flagged by the Bloom Filter with little overhead

{N} epochs

• Bloom Filters
  • Probabilistic Data structures
  • "No" answers are always correct
  • Vector bit array (m) and hash functions (k)
Clustering to Extract Representative Workflows

- Use clustering to find unique representative flows in a sample of the event stream
- Trace Compression using feature vectors
  - MDL Score
  - String Edit Distance (Levenshtein Distance)
  - Hoffman Coding
- Feature Vectors provide insight into the flows
  - Used to extract performance metrics
  - Visualization
- Computationally Expensive
- Representative traces used to design Bloom Filters

Example

Representative Flow (DAG)
(9072 DAG)
Conclusions

• Developed an distributed tracing framework system based on timely dataflow model called Altair that can achieve real time performance

• Evaluated the Altair System on for Anomaly Detection use case

• Evaluation shows that Altair is highly scalable and can be adapted for high production environments
Acknowledgement

Thank you

• I want to thank
  • MIT, Primes for giving me the opportunity to participate in the research program
  • Prof. Raja Sambasivan, Tufts University for his invaluable guidance and spending time with me to assist me with research
  • Graduate students Emre Ates (BU) and Mania Abdi (Northeastern) for their feedback, knowledge and interactions
  • Mass Open Cloud for their access to cloud resources in support of this research

Questions
References

1. Canopy: An End-to-End Performance Tracing And Analysis System, Kaldor et al, SOSP '17, October 28, 2017, Shanghai, China


5. https://github.com/TimelyDataflow/timely-dataflow


