

Lumos: Provenance-guided Automatic Online Debugging

General Examination - Jingyuan (Leo) Chen

Committee: Prof. Amit Levy (Advisor), Prof. Ravi Netravali, Prof. Mae Milano

Abstract

Monitoring and debugging production distributed applications is difficult. The myriad of interactions between concurrent components in a large-scale system can cause transient and non-deterministic bugs that escape thorough off-line testing and verification. Therefore, it is crucial to record the provenances of bugs at runtime to zoom into potential responsible program states for efficient root cause analysis. Existing tools fall short of automatically exposing coherent provenances at low overhead without human intervention. In this talk, I will present Lumos, a dataflow-guided distributed tracing framework that automatically exposes application-level provenances for a bug under investigation. Lumos leverages dataflow analysis to identify program states that may affect the symptom of the incident, and utilizes distributed tracing to expose them with causally related fine-grained tracepoints, while incurring low overhead on the in-production system.

Reading List:

Book: Principles of Computer System Design by Jerome H. Saltzer and M. Frans Kaashoek.

Papers:

1. [Exokernel: An operating system architecture for application-level resource management](#)
2. [The Scalable Commutativity Rule: Designing Scalable Software for Multicore Processors](#)
3. [R2: An Application-Level Kernel for Record and Replay](#)
4. [ODR: Output-deterministic replay for multicore debugging](#)
5. [Improving software diagnosability via log enhancement](#)
6. [Failure sketching: A technique for automated root cause diagnosis of in-production failures](#)
7. [Pivot tracing: Dynamic causal monitoring for distributed systems](#)
8. [A brief survey of program slicing](#)
9. [Fabric: a platform for secure distributed computation and storage](#)
10. [Verdi: a framework for implementing and formally verifying distributed systems](#)