A Study of Long-Tail Latency in n-Tier Systems: RPC vs. Asynchronous Invocations

Qingyang Wang, Louisiana State University

Long-Tail Latency Problem
Web-facing applications encounter large response time fluctuations at moderate utilization (e.g., 50%)

Causes:
- Strong inter-tier dependency between thread-based servers through RPC calls in the long invocation chain
- Millibottlenecks occur in all system layers at moderate system utilization
- Millibottlenecks plus inter-tier dependency leads to Cross-Tier-Queue-Overflow, which in turn cause dropped packets and TCP retransmissions.

Solution:
- Asynchronous invocation between consecutive tiers in the long invocation chain
- Break the strong inter-tier dependency and Cross-Tier-Queue-Overflow

Benefits:
- Achieve predicable performance of n-tier web applications at moderate to high utilization
- Increase resource efficiency and save power of cloud data centers.

Results & Future Work
- Long-tail latency remains absent at system utilization levels as high as 83%, despite the same millibottlenecks.--Wang et al. ICDCS ‘17

Future works:
- Design profiling tools for asynchronous n-tier systems.
- Develop tools to facilitate the transforming RPC code to asynchronous code
- Run large-scale cloud experiments for validation

Push-back: Cross-tier-Queue-Overflow
An illustration example:

- t1=0 ms
- t2=50 ms
- t3=100 ms

A millibottleneck in downstream Tomcat ⇒ long queue in upstream Apache

Offline Analysis

An illustration example:

- t1=0 ms
- t2=50 ms
- t3=100 ms

A millibottleneck in downstream Tomcat ⇒ long queue in upstream Apache