

Add instructor notes
here.

Performance Issues Using Distributed Tracing



Objective

Role of tracing in microservices

What is Spring Cloud Sleuth?

Adding Spring Cloud Sleuth to a project

Visualizing latency with Zipkin

Adding Zipkin to a solution

Role of Tracing in Microservices

Locating misbehaving components

Observe end to end latency

Understand actual,not specified behavior

What is Spring Cloud Sleuth

Spring Cloud Sleuth implements a distributed tracing solution for Spring Cloud, borrowing heavily from Dapper, Zipkin and Htrace

Sleuth should be invisible, and all your interactions with external systems should be instrumented automatically.

Glossary of Spring Cloud Sleuth Terms

1.Span

2.Trace

3.Annotation

- a. Client Sent
- b. Server Received
- c. Server Sent
- d. Client Received

Demo-Adding Spring Cloud Sleuth to a project

Visualizing latency with Zipkin

Zipkin is created by Twitter, OpenZipkin public fork.

Collects timing data

Shows service dependencies

Visualize latency for spans in trace

Many integrations, besides Spring

Demo-Adding Zipkin to a solution