

Pattern: Exception tracking

Context

You have applied the Microservice architecture pattern ([../microservices.html](#)). The application consists of multiple services and service instances that are running on multiple machines. Errors sometimes occur when handling requests. When an error occurs, a service instance throws an exception, which contains an error message and a stack trace.

Problem

How to understand the behavior of an application and troubleshoot problems?

Forces

- Exceptions must be de-duplicated, recorded, investigated by developers and the underlying issue resolved
- Any solution should have minimal runtime overhead

Solution

Report all exceptions to a centralized exception tracking service that aggregates and tracks exceptions and notifies developers.

Resulting Context

This pattern has the following benefits:

- It is easier to view exceptions and track their resolution

This pattern has the following drawbacks:

- The exception tracking service is additional infrastructure

Related patterns

- Log aggregation ([application-logging.html](#)) - exceptions should be logged as well as reported to a tracking service

[Tweet](#)

[Follow @MicroSvcArch](#)