Adapting Legacy Monolithic Applications into API Services for Web, Mobile and Other Clients

Walter Scarborough

CSESSP 2015

TACC
TEXAS ADVANCED COMPUTING CENTER
Powering Discoveries that Change the World

The Vision: from Portal to Platform

Background: XSEDE User Portal (XUP) stakeholders wanted to provide better support for mobile clients

Motivation: real time HPC job status updates via push notifications

TACC

Portal Design Tradeoffs

Client Layer

- Single Client vs. Multiple Clients

API Layer

- No API vs. Monolithic API vs. Microservice API

Data Layer

- Single Datasource vs. Multiple Datasources

TACC

XSEDE User Portal API

Over time, our monolithic API became difficult to maintain due to:

- Needing to support multiple endpoint versions
- Adding new datasources

A different design approach was needed for better complexity management.

TACC

From Monolithic to Microservices

Microservices aren't always the right solution, but they were the right solution for XUP.

- Service complexity can be reduced
- Services can evolve independently and at different speeds
- Services can be written by different teams in their language of choice
- Services can be reused in future APIs
- Suitable for incorporating a variety of separate datasources

Cyclomatic Complexity Comparison

	Monolithic XUP API	Microservice XUP API
Allocations Service	4	2.5
Profile Service	6	5
Token Service	2.2	2

Measurements taken using phploc v2.0.2

TACC

XSEDE User Portal Platform

- Full XUP (Java)
- Mobile
  - iOS (Swift / Objective C)
  - Android (Java)
  - Webapp (Javascript)

TACC

XSEDE User Portal API Infrastructure

- Nginx used as API gateway
- All services communicate via TCP
- All services run in docker containers
- All services documented via Swagger
- All services endpoints are versioned for compatibility
- Languages currently used: PHP, Python and Javascript (Node.js)

TACC

Future Work

- Move more legacy portal business logic into the API
- Improve software metrics gathering to identify complexity "hot spots"
- Explore Swagger client code generation
- Improve test coverage
- Continue adding continuous integration support

TACC

Questions?

wscarbor@tacc.utexas.edu