Android, iOS, and Web
Best Friends Forever: The BFF Pattern

Janani Subbiah, Detroit Labs

Who am I?

Software Engineer

Android apps/backend systems

From India

Agenda

What is a BFF?

Advantages of using a BFF

“No free lunch” with using a BFF

What can help?

When do I need a BFF?

Who..What..BFF?

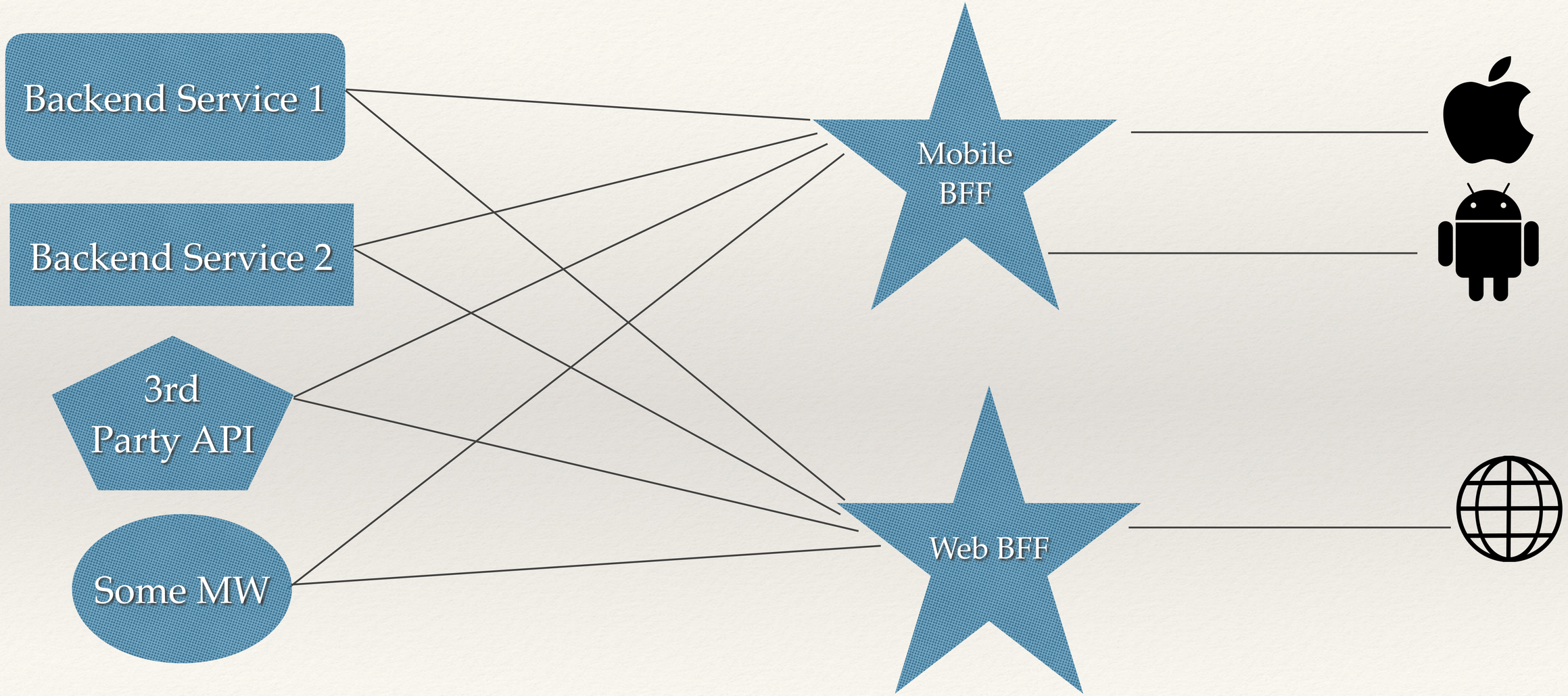
Backends For Frontends

Backend For Frontends

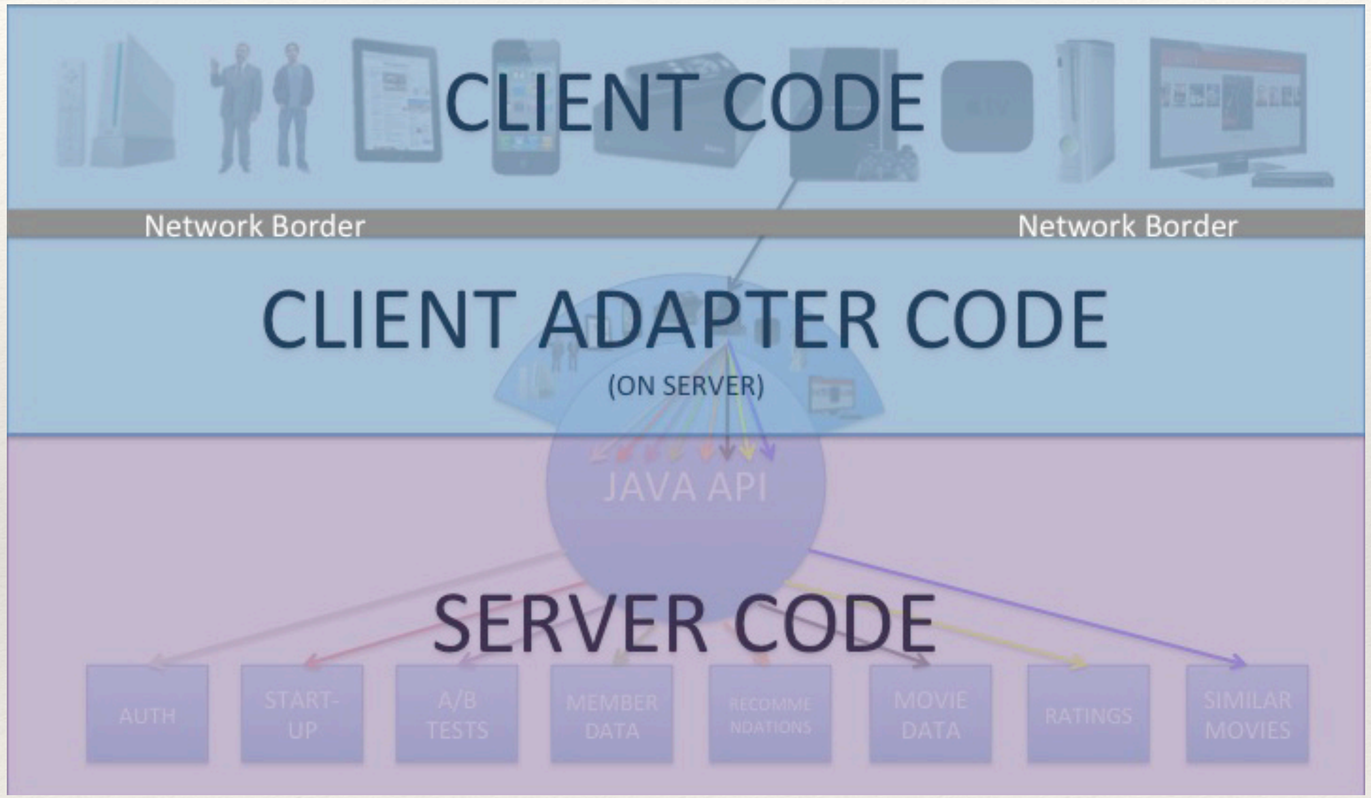
A service(s) that sits in between various IT systems and the UIs

A middleware specifically **tailored for the UI(s)**

Backends For Frontends

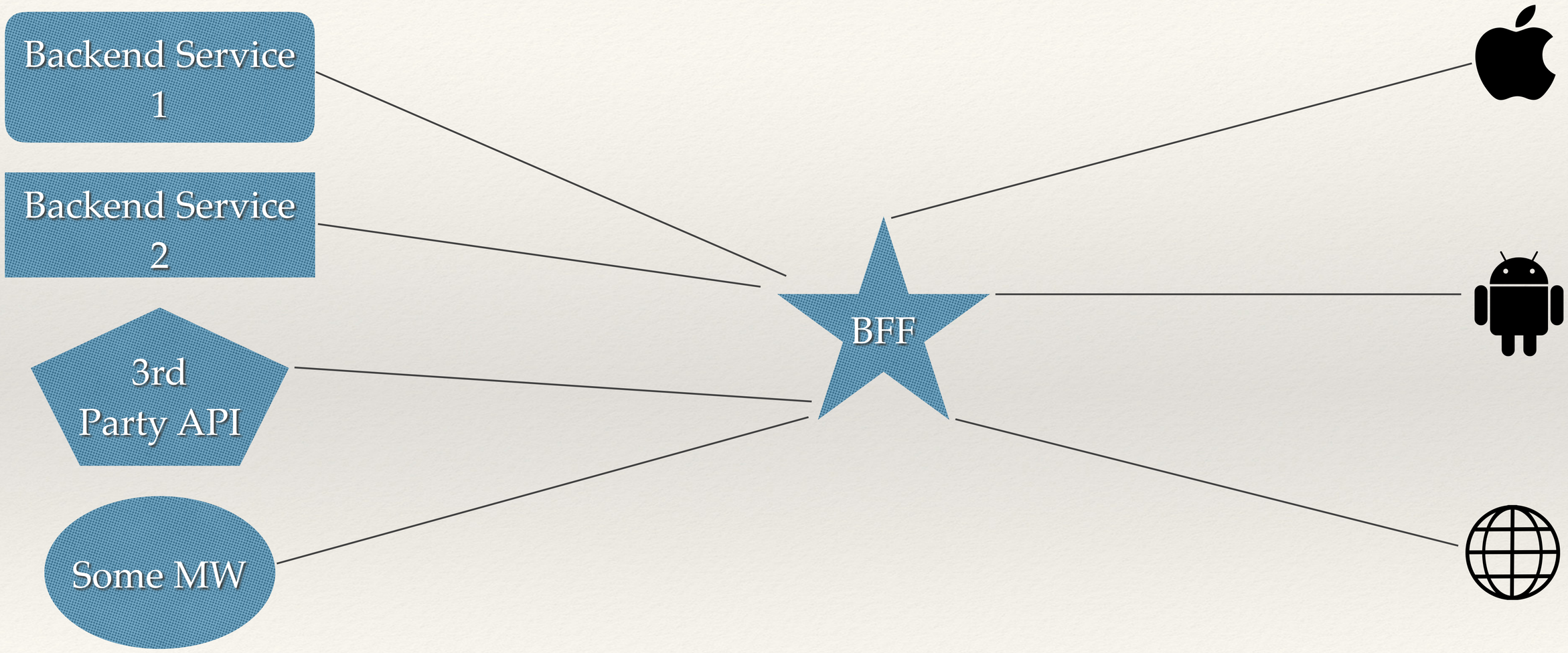


The Netflix Approach



Netflix: Embracing the differences inside the Netflix API redesign

Backend For Frontends



YAY BFF!

Orchestrator

Between multiple systems and the UIs

UI specific APIs

Cut out what is not required

Add what is required

Combine multiple downstream services

Caching

Caches responses

Less load on downstream services

Improved response time

Responses even if downstream services are down

Less code duplication

Common logic pulled up into the BFF

Easy to take updates

Consistency

If something is wrong, it is wrong in all the places

One place to fix them all

Consistency across UIs

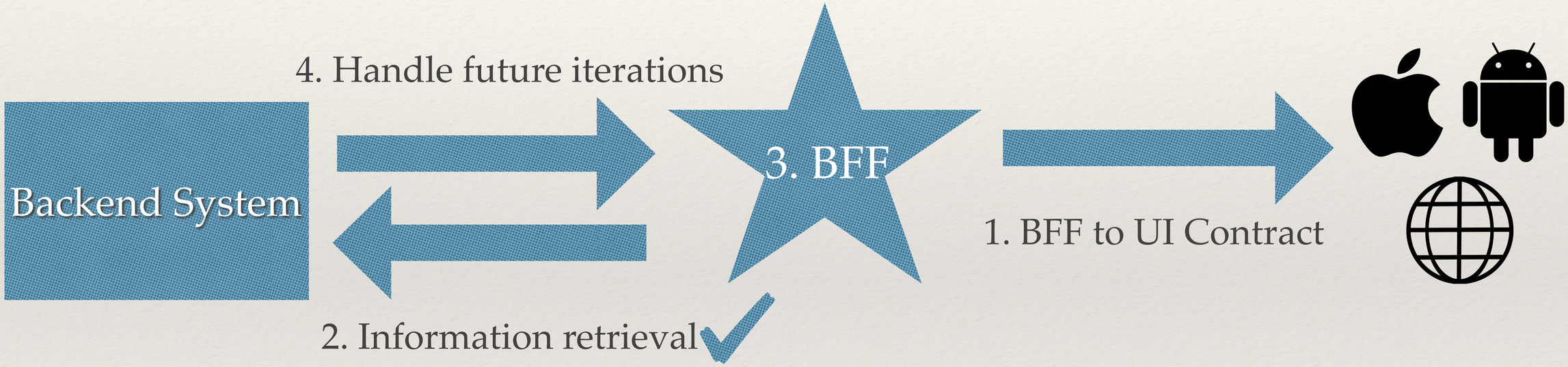
All UIs expect the same response

UI specific features

Backend systems DO NOT have to worry about UI specific features

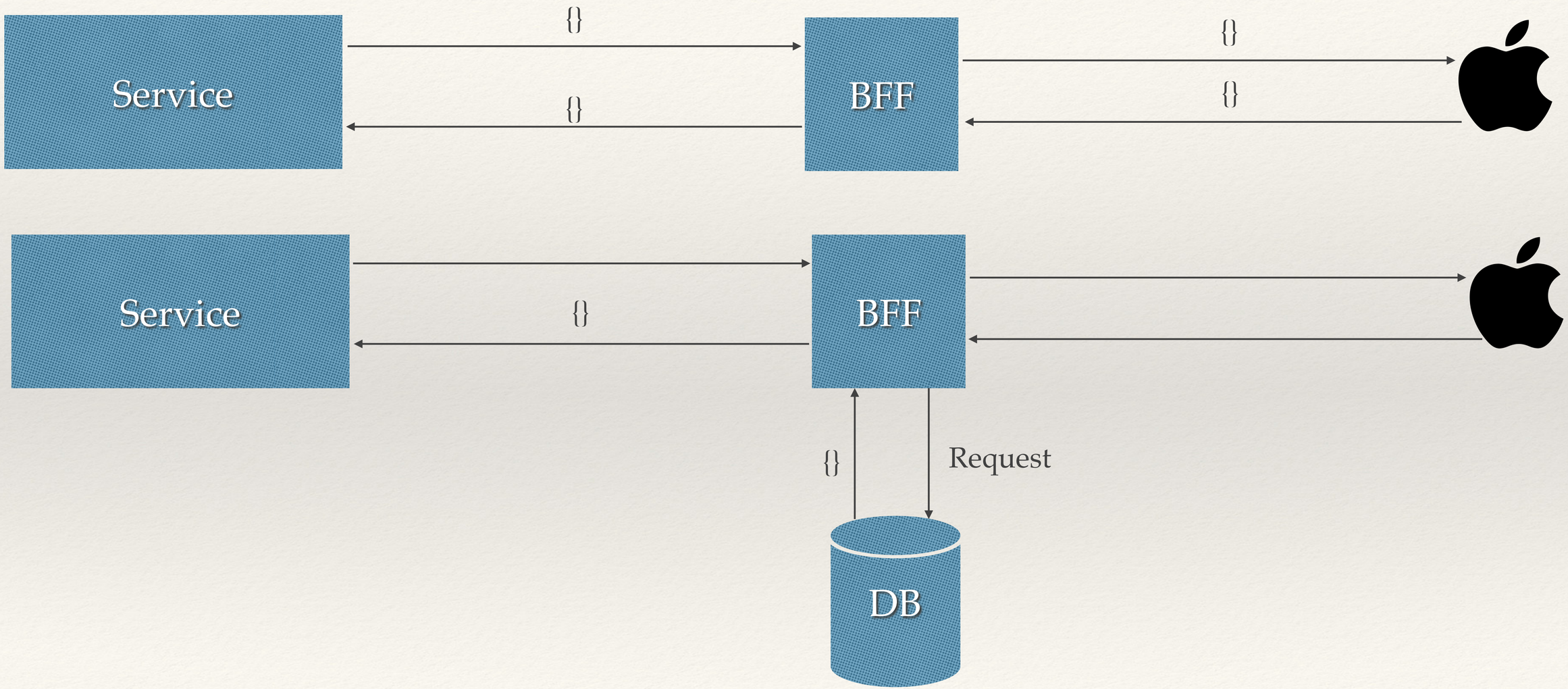
Push notifications

Getting ahead



The.....Downside

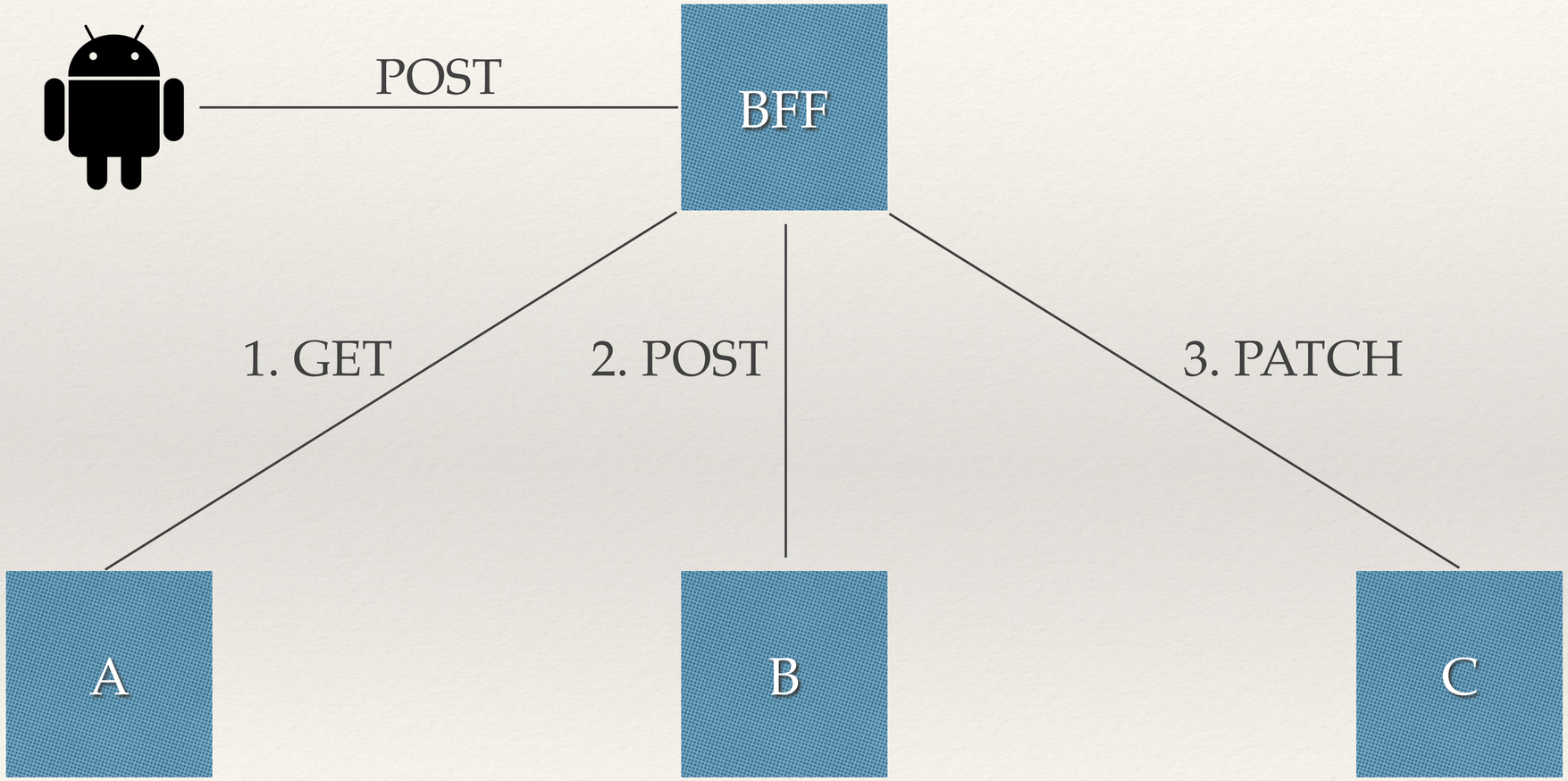
No Database Storage



An additional service

+1 deployment, maintenance, update

Hidden failures

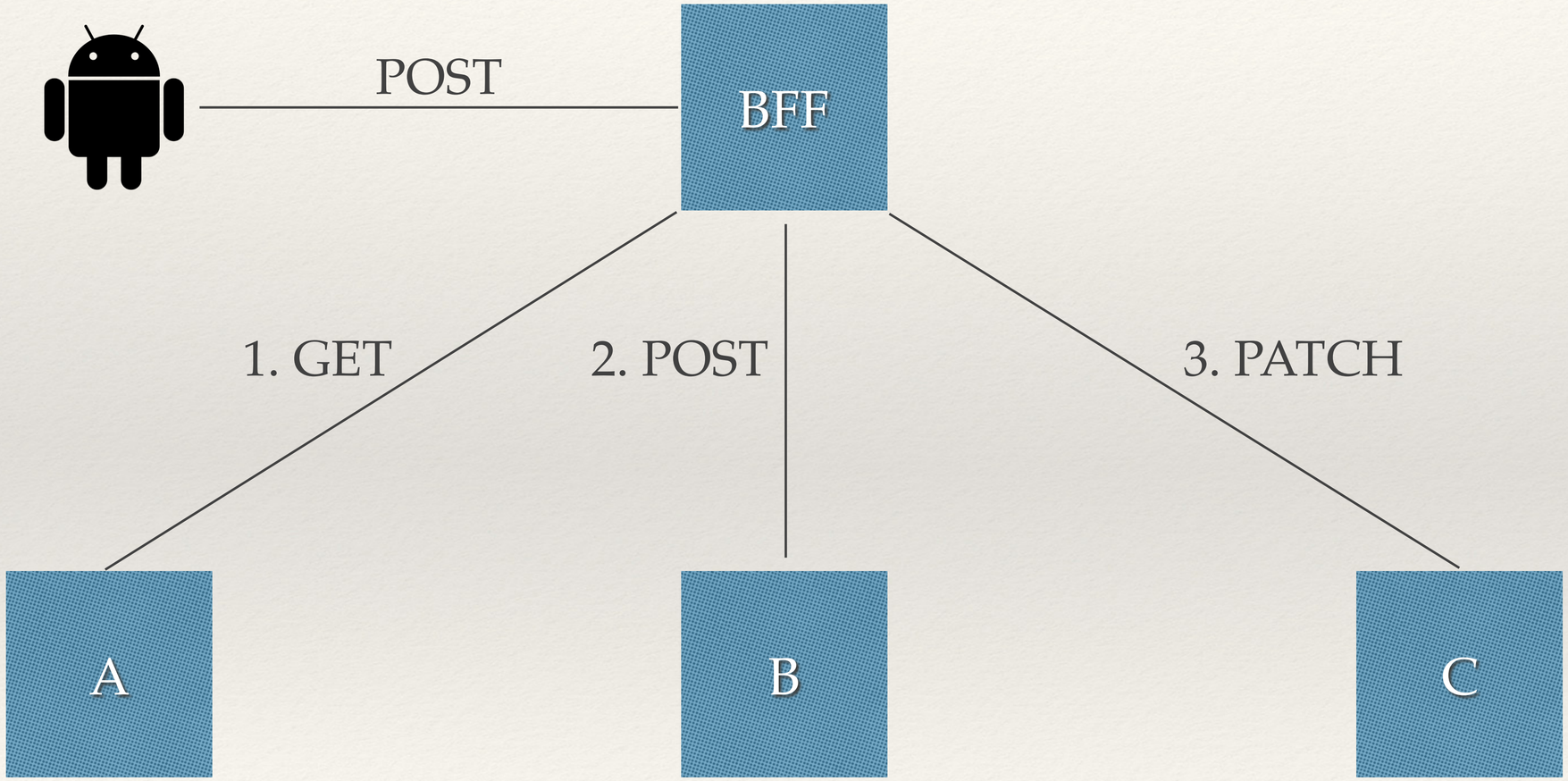


PATCH to C fails

=>

POST to the UI fails

HTTP verbs mimicry



Hard to maintain a 1:1 mapping

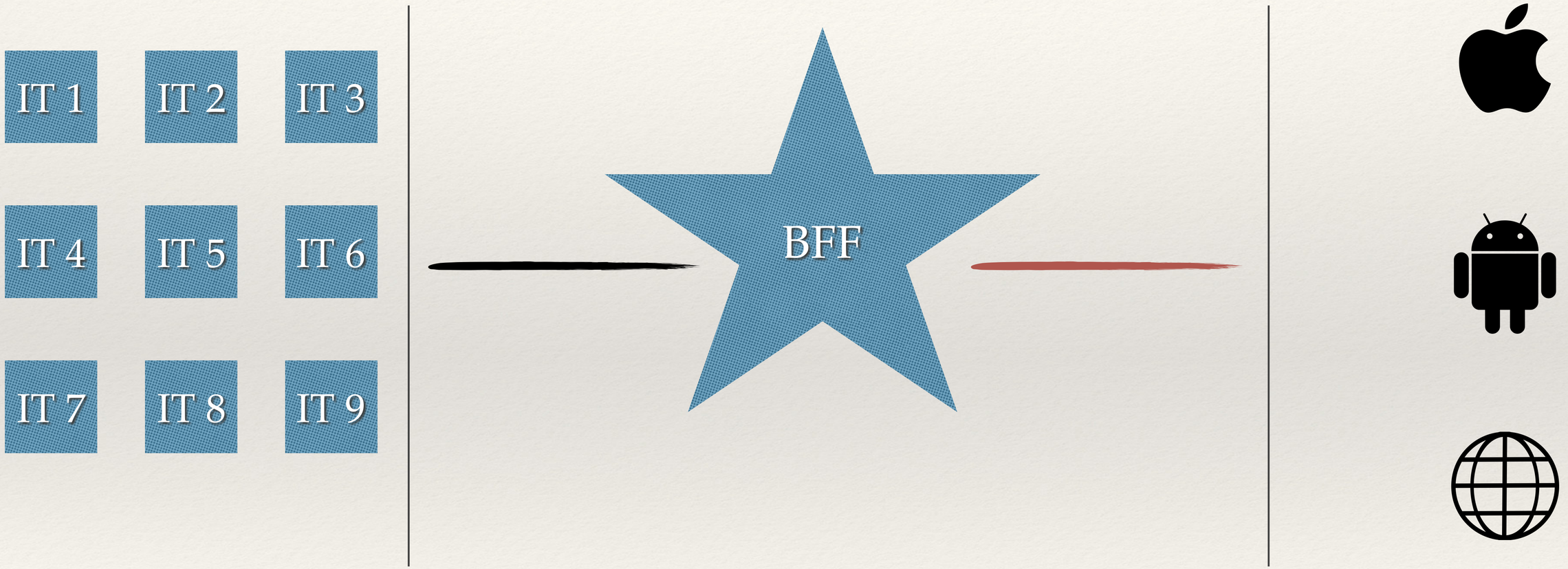
Web vs mobile

Web apps have specific needs

Cross Origin Resource Sharing

Cross Site Request Forgery

BFF Team abstraction



What helps?

Charles: Proxying tool

Proxying downstream APIs to BFF

Proxying BFF to UI

Rewrite rules

Map Local rules

Documentation

Swagger for the API specs

Documentation Github repo

Screenshots

Sample responses

Links to Swagger APIs

Charles rewrite and map local rules

Logging

Logging all requests and responses

Splunk

Error Messages

Meaningful error messages

What happened? Where did it happen?

Integration testing

Periodic testing sessions

Mostly once/sprint

Run through demo scenarios

When do I need a BFF?

Multiple UIs with complicated data retrieval logic

A lot of systems the UI(s) need to talk to

My is not for everyone

but everyone will mostly needs a BFF especially if you are building a multi platform app!

Credits/References

Bryan Kelly & Chelsey Baker, Detroit Labs

Netflix: Embracing the differences inside the Netflix API redesign

ThoughtWorks: BFF @ SoundCloud

Questions?