GILT

How Gilt Avoids Building a "Distributed Monolith"

Kevin Li
kli@gilt.com
Software Engineer @ Gilt
June 21, 2016

GILT

What is Gilt?

https://www.gilt.com/careers/open-positions/

GILT

What exactly is a Distributed Monolith?

Distributed systems coupled by **binary dependencies**

GILT

Common Examples of Binary Coupling

- Shared common models
- Shared “official” clients
- Shared platform dependencies

GILT

Wait a minute...those sounds great!

CS-100: **D**ont **R**epeat **Y**ourself!!!

MICROSERVICES

Think twice before applying DRY principles across distributed applications.

There are ways to achieve the same benefits of DRY without using dependencies.

GILT

Benefits of Microservices

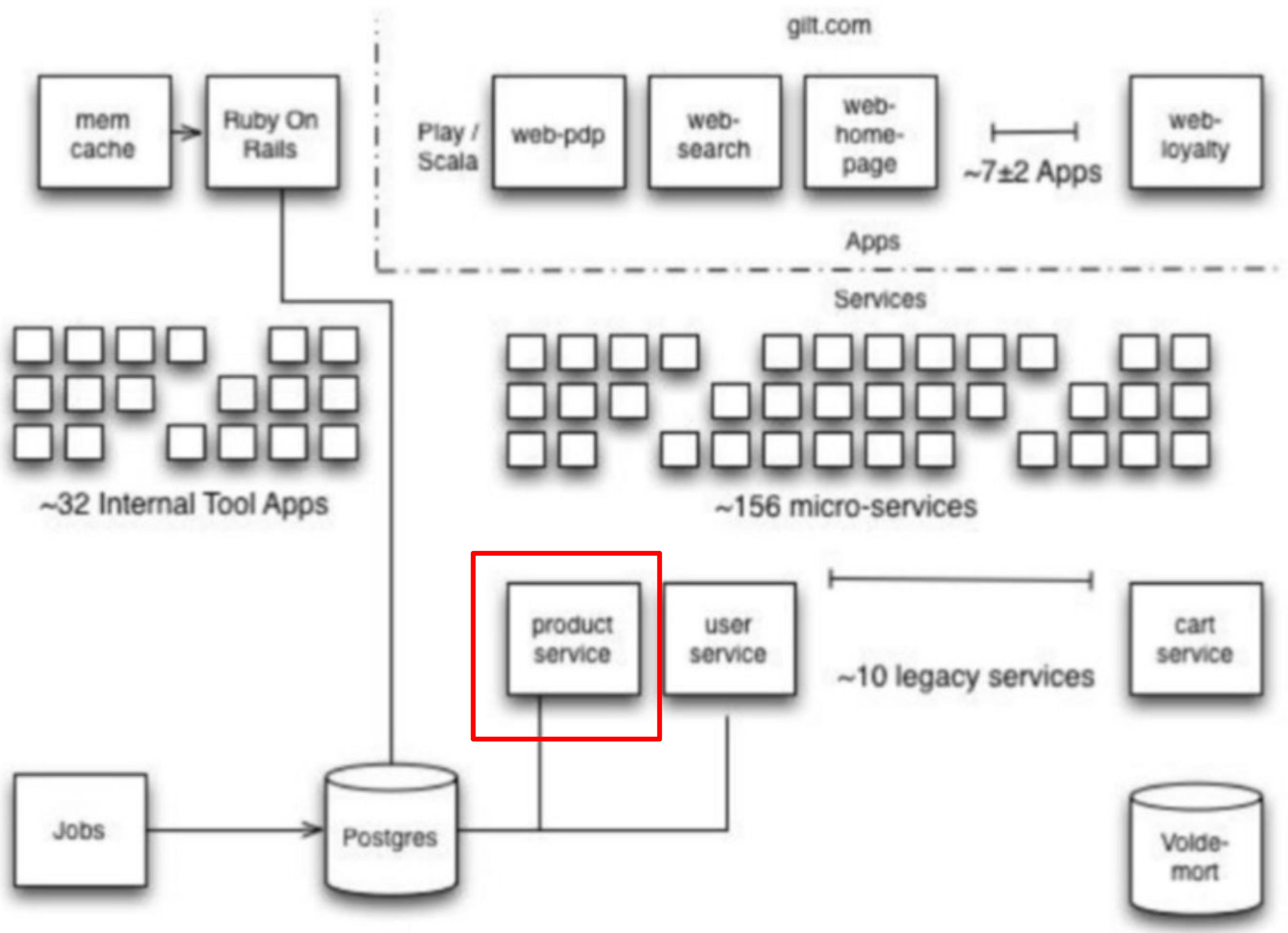
Independent scalability

Use the best tool for the job

Developer Velocity

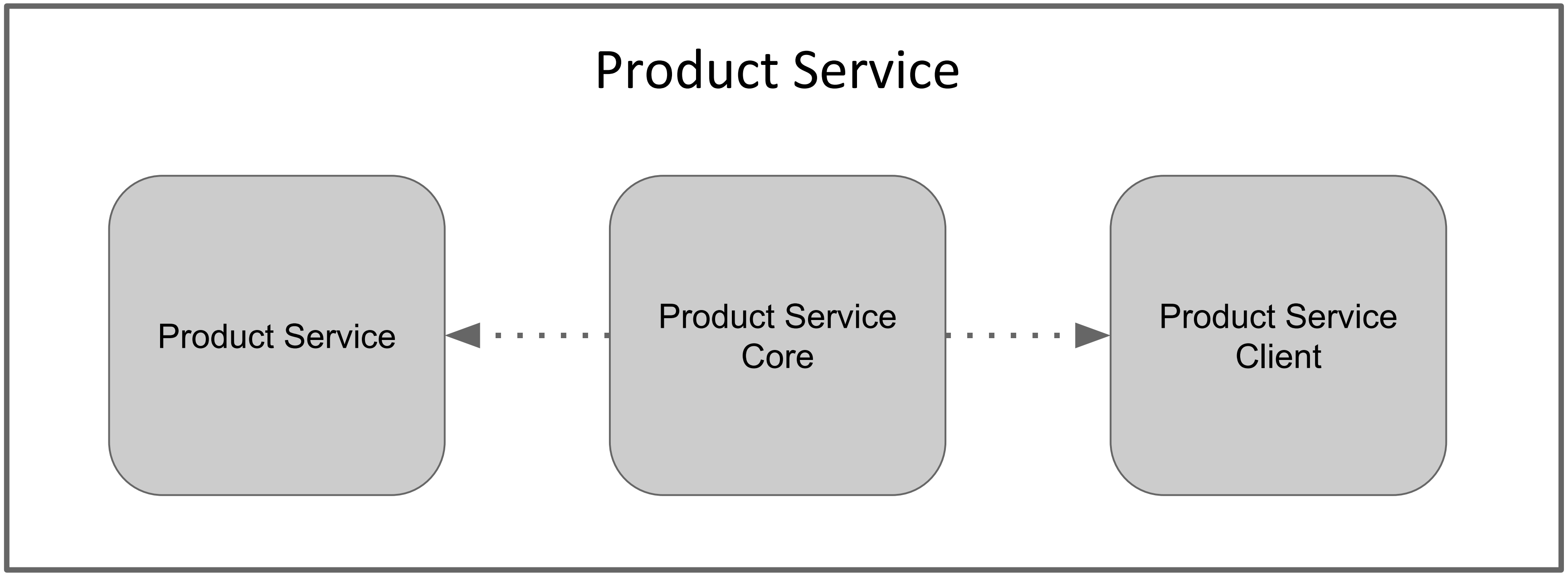
GILT

Gilt Microservice Landscape (circa 2014)



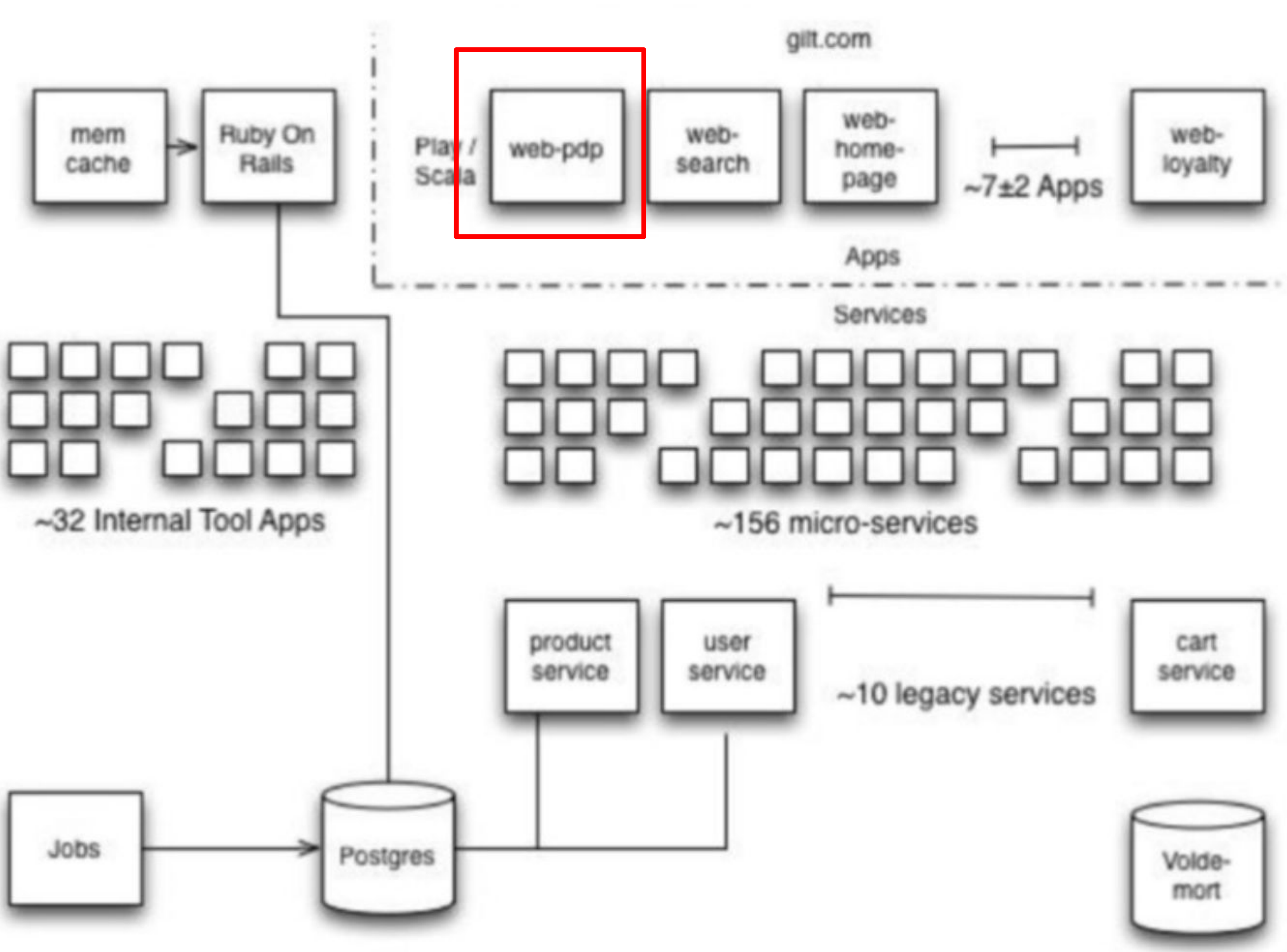
GILT

ClientServerCore Framework



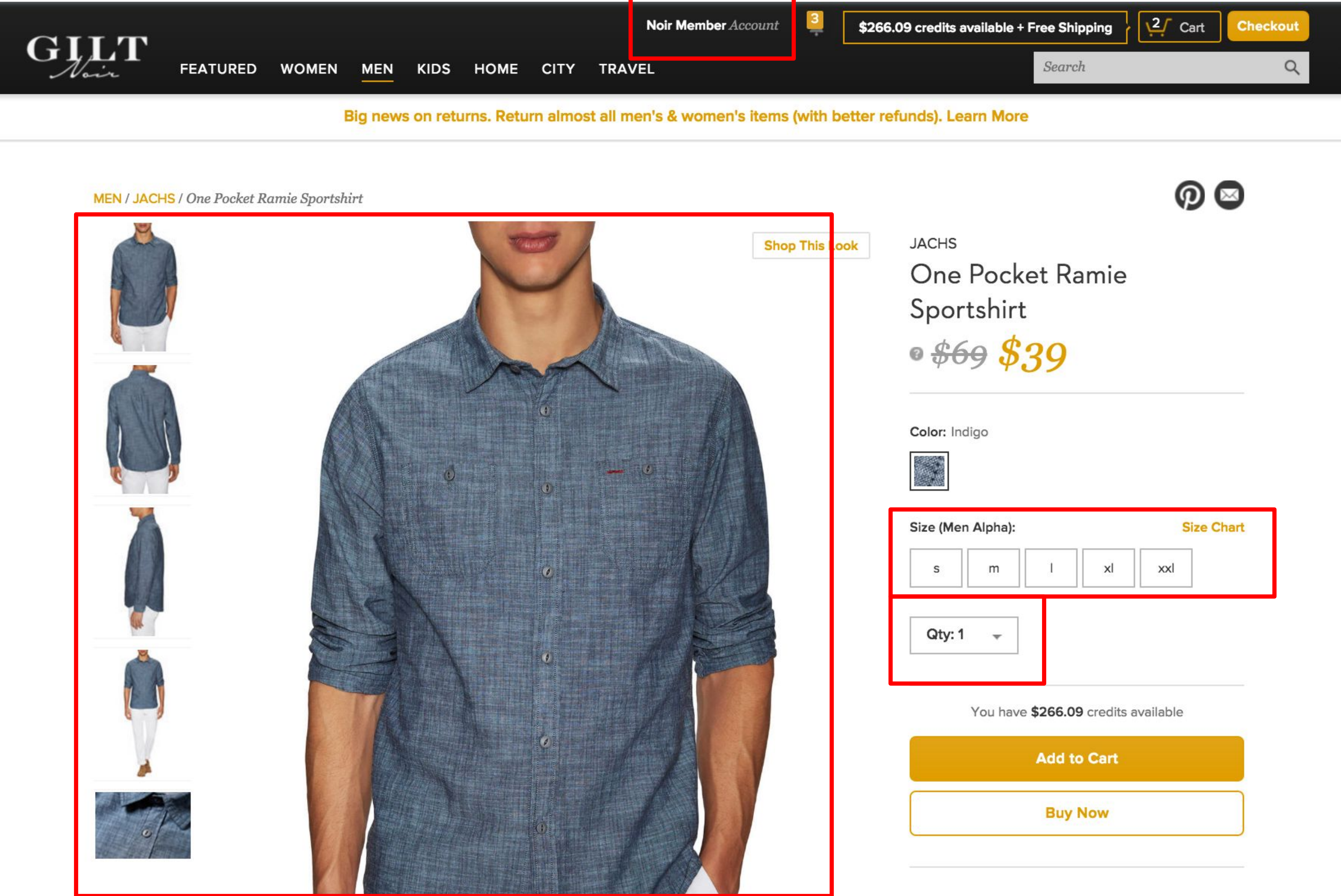
GILT

Gilt Microservice Landscape (circa 2013/14)



GILT

Product Detail Page



Noir Member *Account*

3

$266.09 credits available + Free Shipping

2 Cart

Checkout

FEATURED WOMEN MEN KIDS HOME CITY TRAVEL

Search

Big news on returns. Return almost all men's & women's items (with better refunds). Learn More

MEN / JACHS / *One Pocket Ramie Sportshirt*

Shop This Look

JACHS

One Pocket Ramie Sportshirt

~~$69~~ $39

Color: Indigo

Size (Men Alpha): Size Chart

s m l xl xxl

Qty: 1

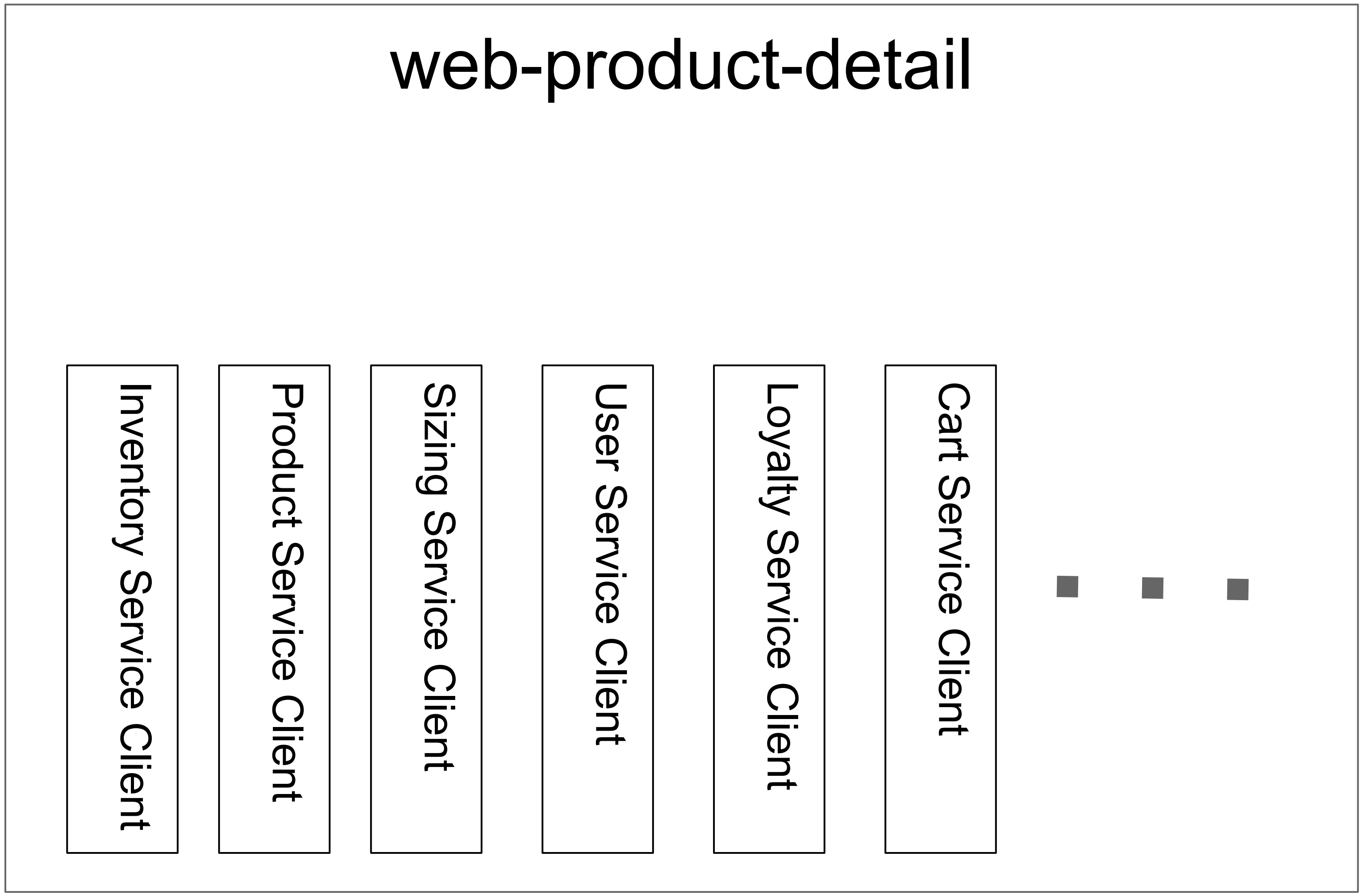
You have **$266.09** credits available

Add to Cart

Buy Now

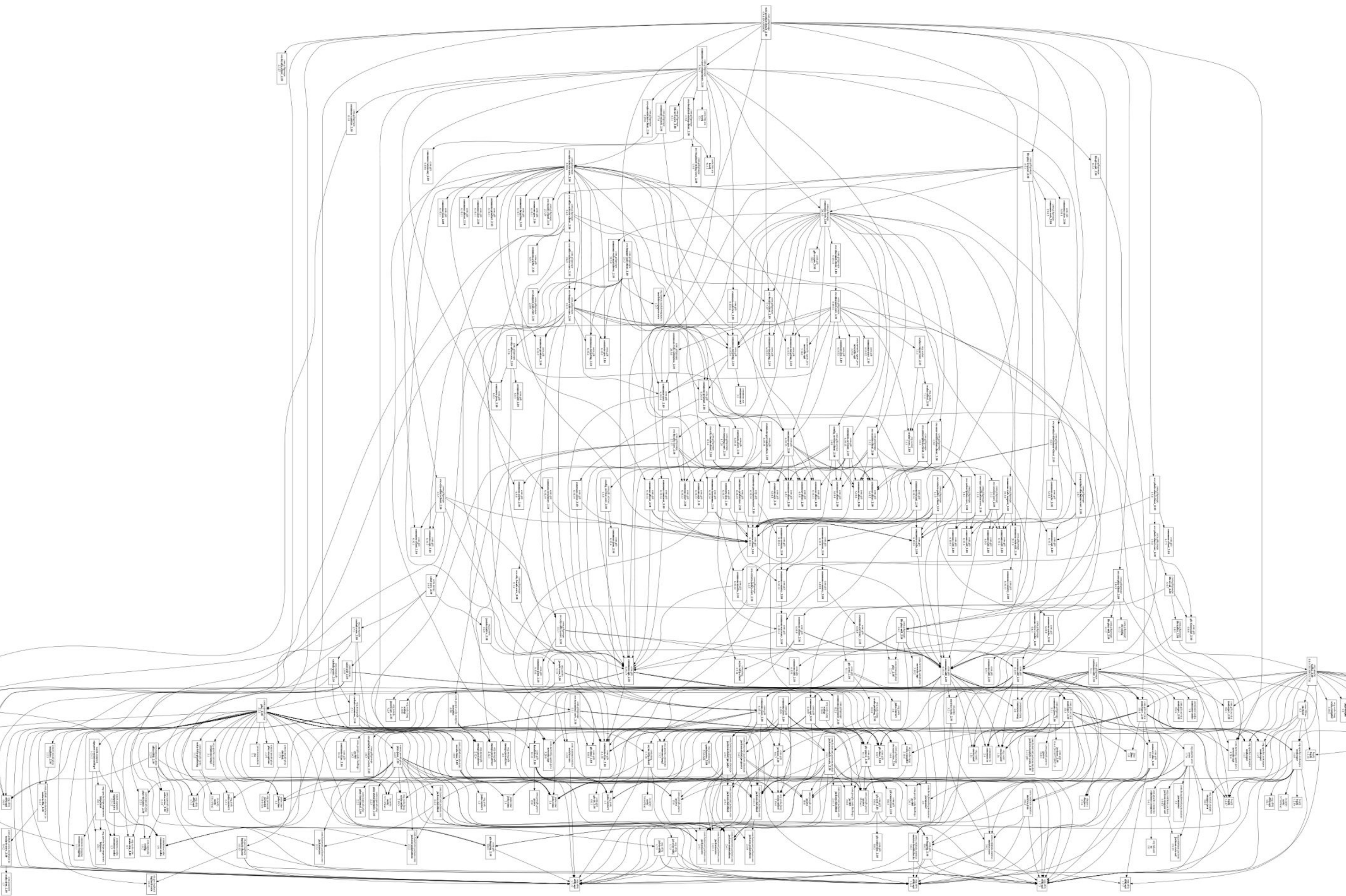
GILT

Product Detail Page



GILT

Product Detail Page Dependency Graph



GILT

It worked great...until

- Gilt scaled to 200+ microservices
- Different Scala Versions
- Different Java Versions
- Different Framework Versions

DEPENDENCY HELL!!!

Independent Scalability?

Arguably.

What if a more performant programming language could be used?

What if a different concurrency model or networking stack is faster?

GILT

Use the best tool for the job?

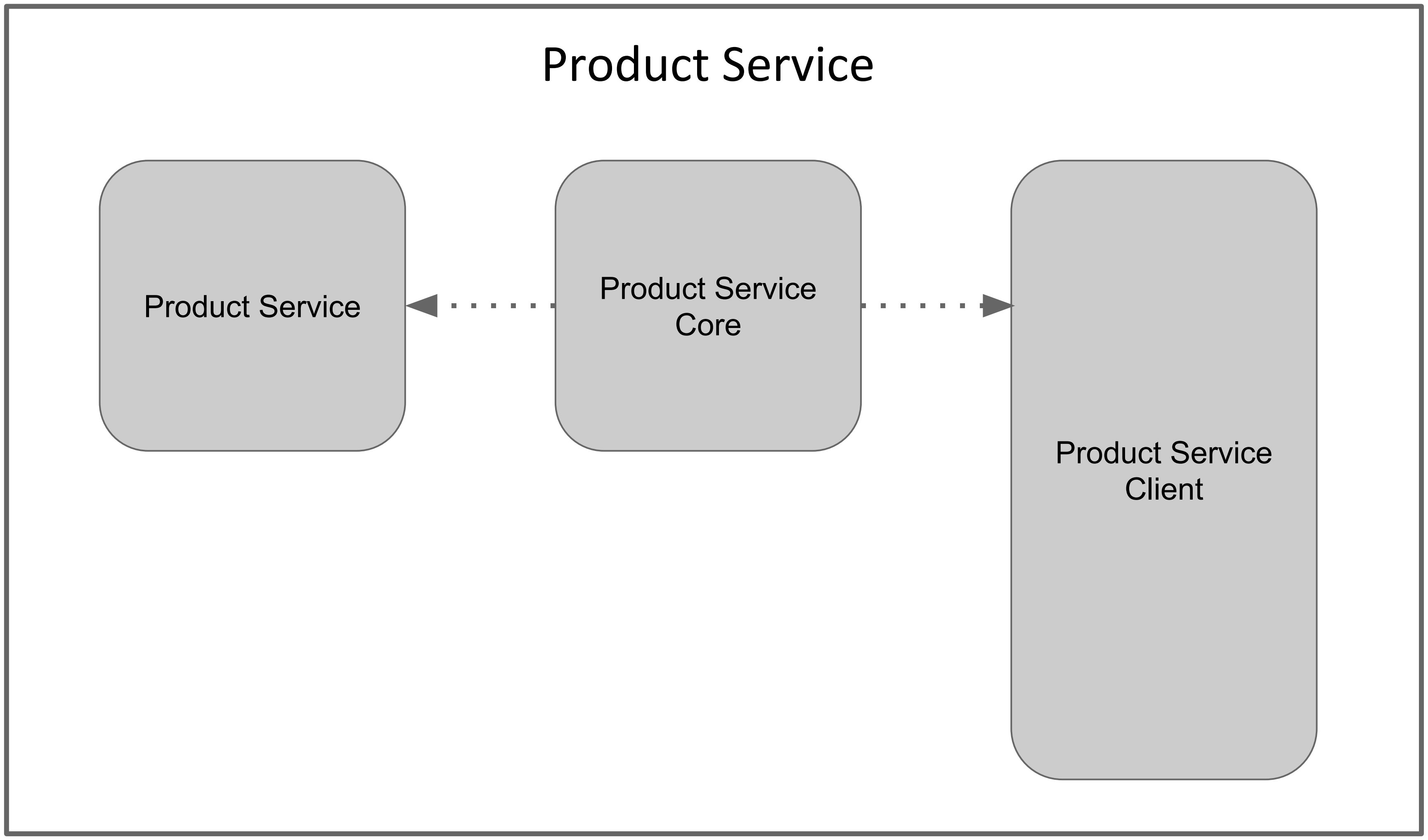
No.

Official Clients Unavailable.

“Rogue” clients are difficult to create and are fragile

GILT

Increasingly "Fat" Client



Developer Velocity?

No.

Cannot independently upgrade dependencies

Takes months to upgrade dependencies across the organization.

GILT

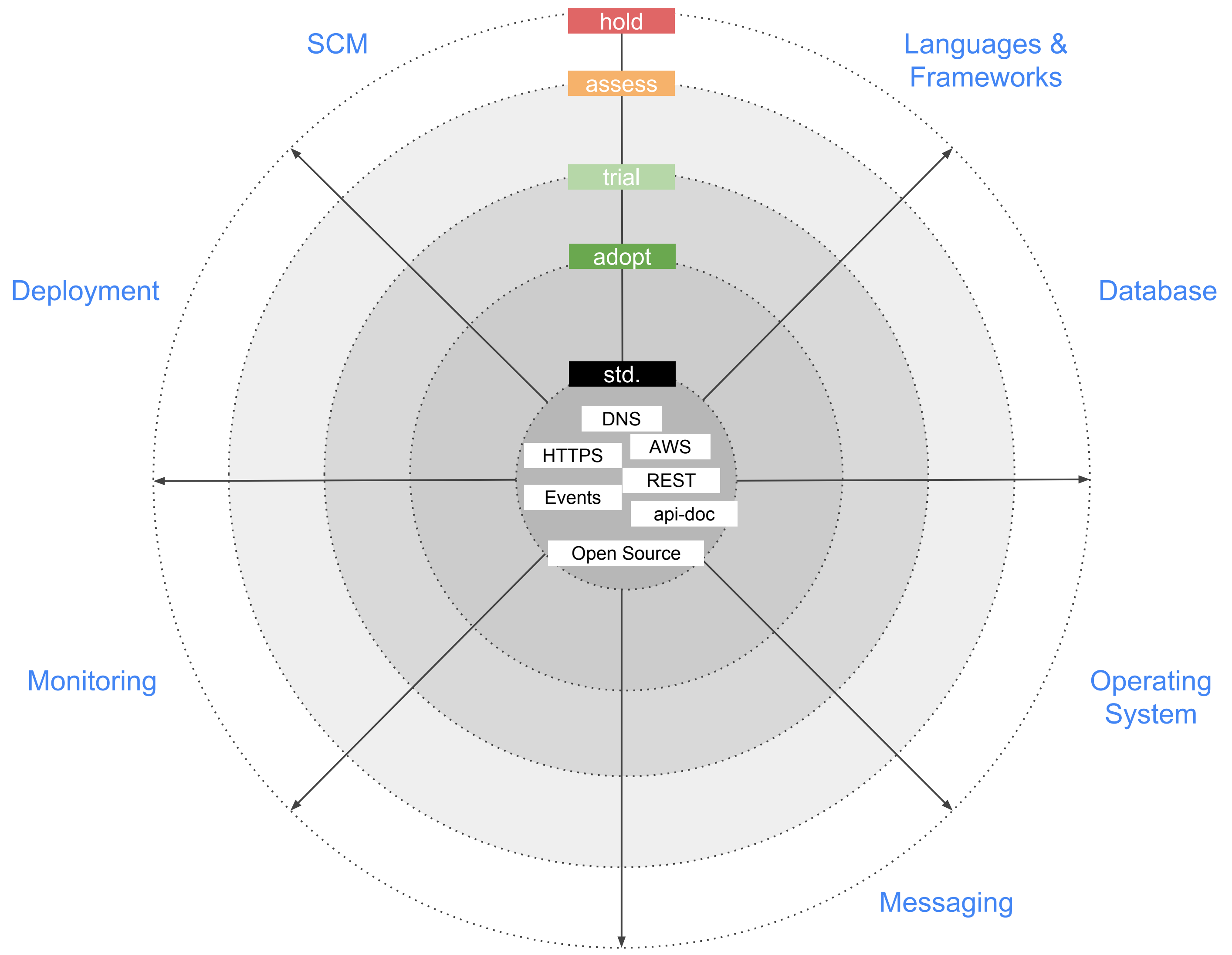
The Solution

Depend on **Standards**, not binaries:

https://github.com/gilt/standards

GILT

Gilt’s Tech Radar



GILT

But...doesn’t that affect developer velocity?

I don’t want to rewrite a client everytime I need to talk to a service. And I sure as hell don’t want to copy/paste the same client multiple times.

GILT

Schema-first development

With a well defined schema, we can write code generators to create models, serializers and clients for us.

Turns out copy & paste is fine if you’re copying **versioned and programmatically generated code!**

GILT

Swagger

“The World's Most Popular Framework for APIs.”

API schema can be defined in Json or YAML

Wide range of supported languages

https://github.com/zalando/play-swagger

www.swagger.io

GILT

apidoc

Open Source and created by Gilt’s ex-CTO Mike Bryzek

Schema defined in api.json file (swagger also supported)

Client generation for: Play, Node, Ning, Ruby, Android, Go, (Swift in the works)

Convention over configuration

Excellent Play Framework Support

apidoc

GILT

Apidoc Demo!

https://github.com/yuanl1/hot-potato

GILT

How is this better than WSDL?

- Doesn’t hide the fact that you’re going over the wire
- Client is completely restful
- No remote Objects

Summary

- Think twice before adding a binary dependency
- Depend on standards, not binaries
- API first / Schema first development
- Prefer code generation over adding an dependency

Thank You

Thanks to Mike Bryzek for creating and open sourcing apidoc

Thank to Ben Christensen for inspiring this talk with his excellent presentation: https://www.youtube.com/watch?v=-czp0Y4Z36Y

Thanks to Sean Sullivan for critiquing and providing some slides