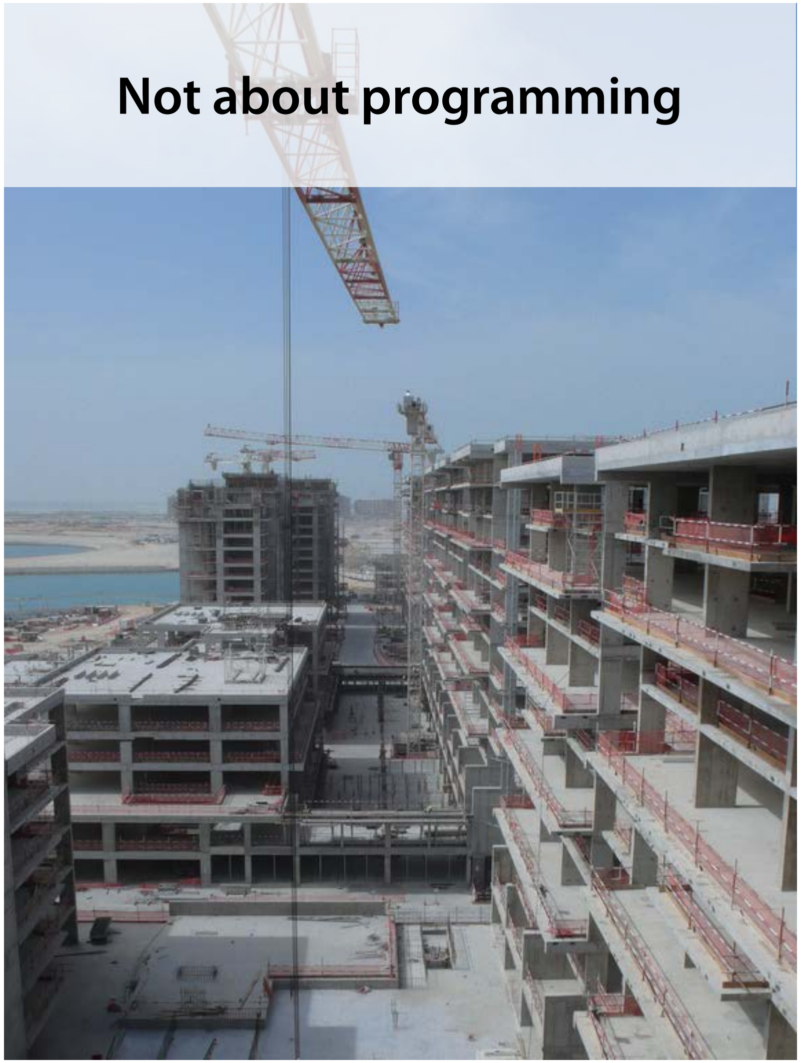
Building RESTful Systems: Introduction

Howard Dierking

http://codebetter.com/howarddierking

pluralsight
hardcore developer training

About This Course



Not about programming

About architecture and design

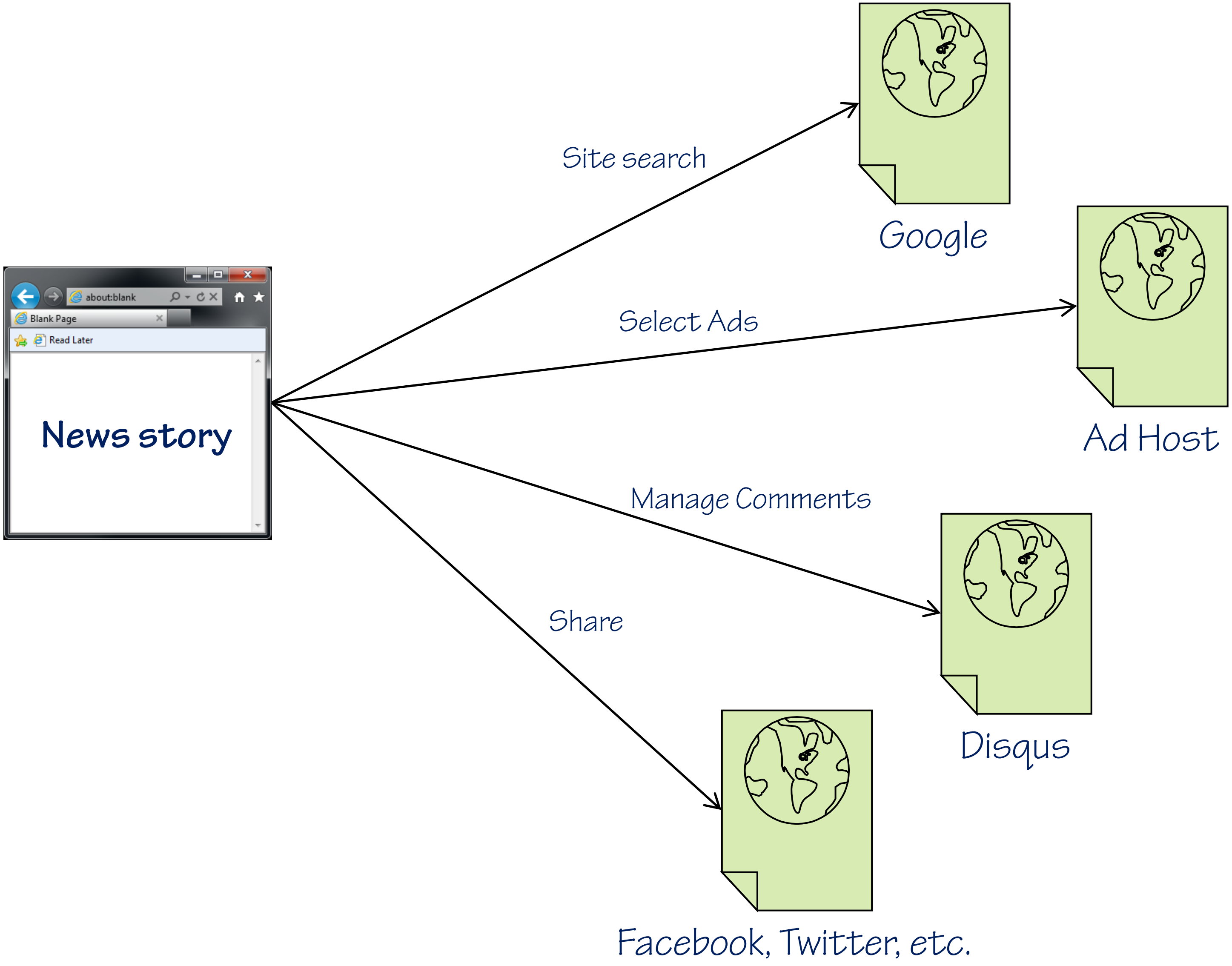
Overview

- REST Drivers
- REST Properties
- Definition of REST
- REST and Other Technologies/Architectural Styles
- REST and Richardson’s Maturity Model

REST Drivers

- Heterogeneous Interoperability
- Devices
- The Cloud

Heterogeneous Interoperability



News story

Site search

Google

Select Ads

Ad Host

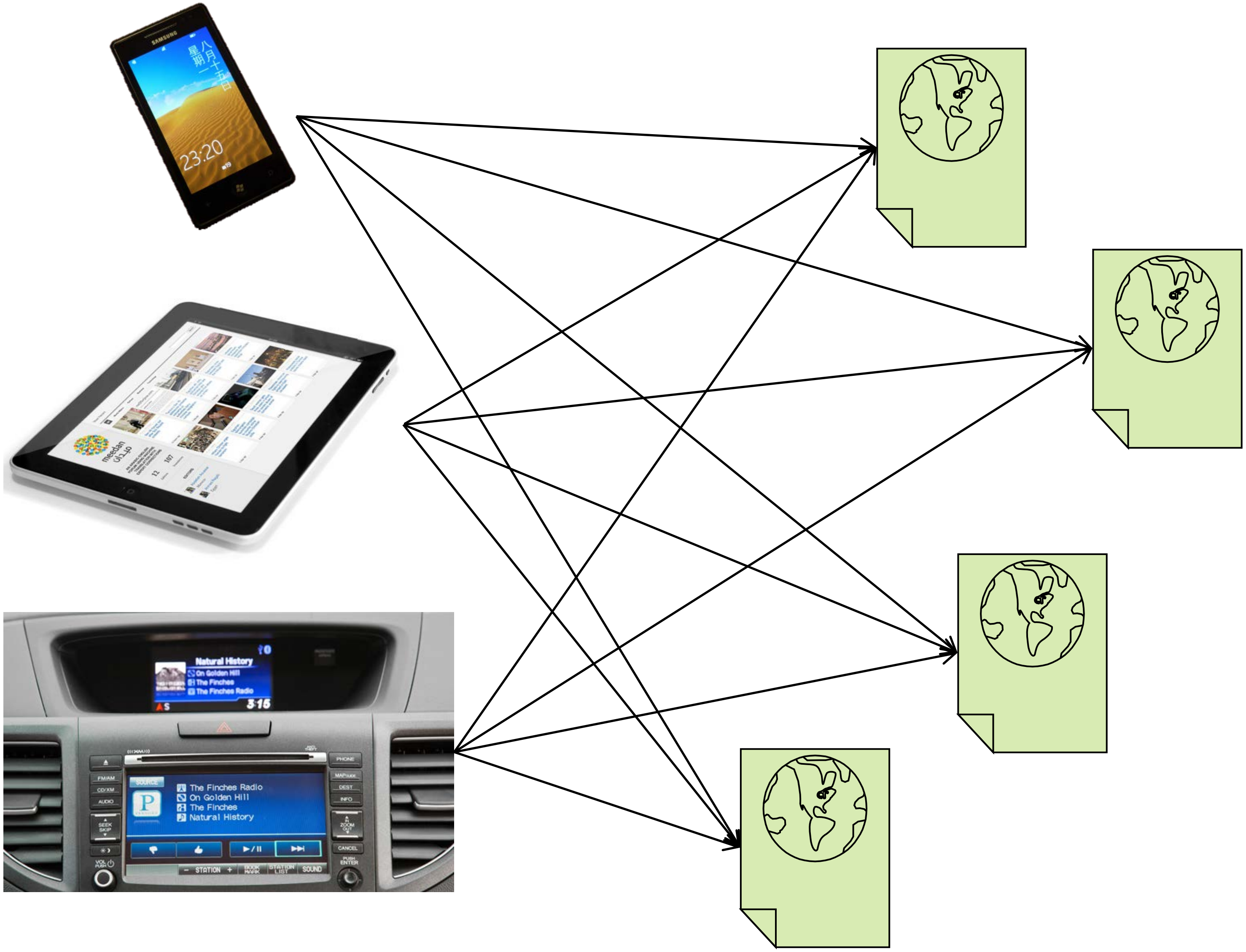
Manage Comments

Disqus

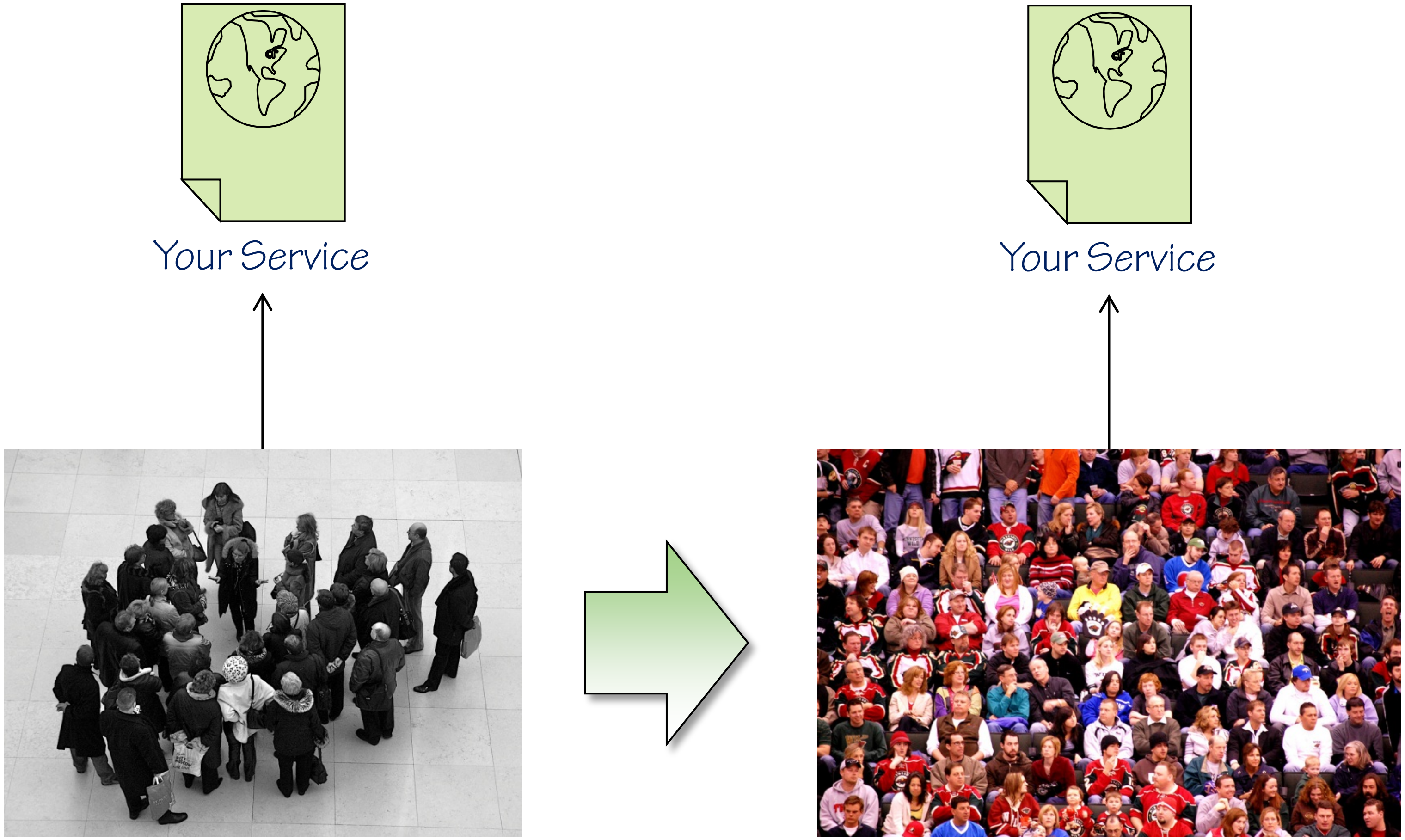
Share

Facebook, Twitter, etc.

Devices


SAMSUNG
八月十五日
星期一
23:20
meedan
Natural History
On Golden Hill
The Finches
The Finches Radio
3:15
SOURCE
The Finches Radio
On Golden Hill
The Finches
Natural History
STATION
SOUND

The Cloud



Your Service

Your Service

“It is critical to build a scalable architecture in order to take advantage of a scalable infrastructure”

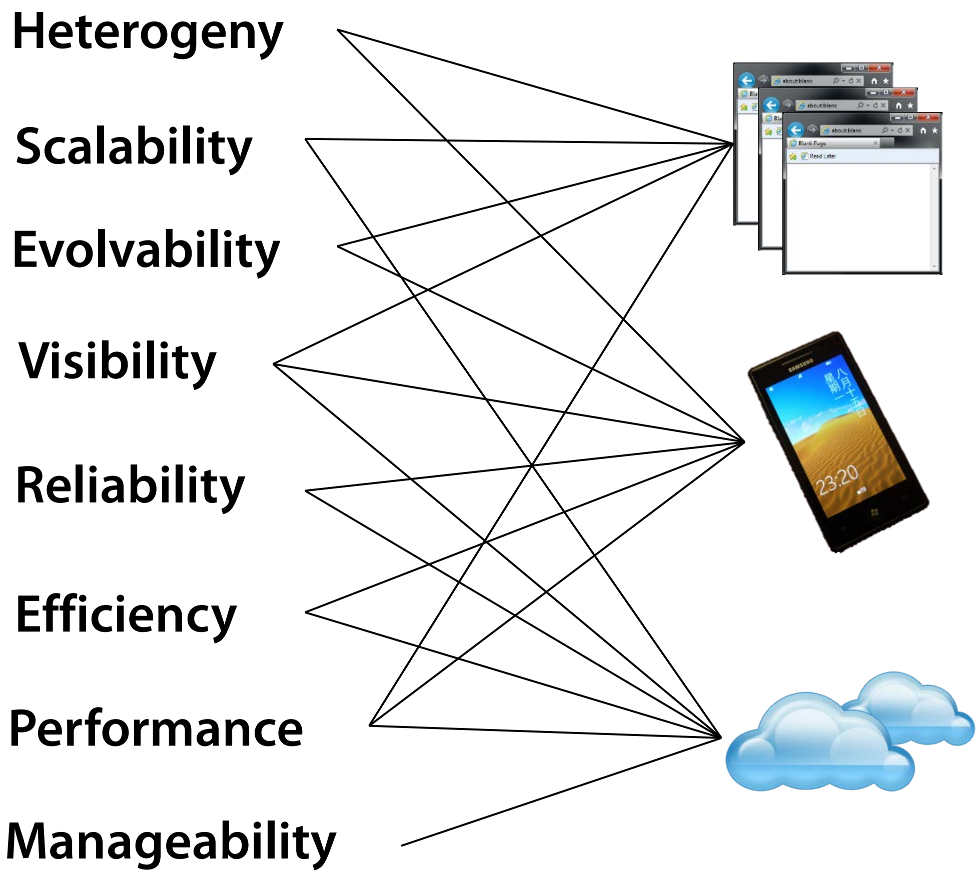
The Cloud

- Many companies do not and will not have Facebook-sized scale requirements
- The major appeal of the cloud is getting out of the business of maintaining infrastructure

Properties of REST

- Heterogeny
- Scalability
- Evolvability
- Visibility
- Reliability
- Efficiency
- Performance
- Manageability

Properties of REST



Heterogeny

Scalability

Evolvability

Visibility

Reliability

Efficiency

Performance

Manageability

What REST is NOT

- **RPC**
  - REST is not a way to call methods over a network without the overhead of SOAP and WSDL
- **HTTP**
  - While most current RESTful systems are built using HTTP, an architecture implemented on top of HTTP is not inherently RESTful
- **URIs**
  - Largely because of the frameworks on which they are built, many RESTful systems have clean URLs (“cool URLs”). However, this is not a requirement for REST
  - Hyper-focus on URIs can actually make designs non-RESTful

What is REST?

Architectural Style

- Stands for “Representational state transfer”
- First introduced in 2000 by Roy Fielding as a part of his doctoral dissertation
- Intended for long-lived network-based applications

REST and SOAP

- Many people think of REST as “something that isn’t SOAP”
- SOAP (and WSDL) is an implementation detail for RPC-style systems

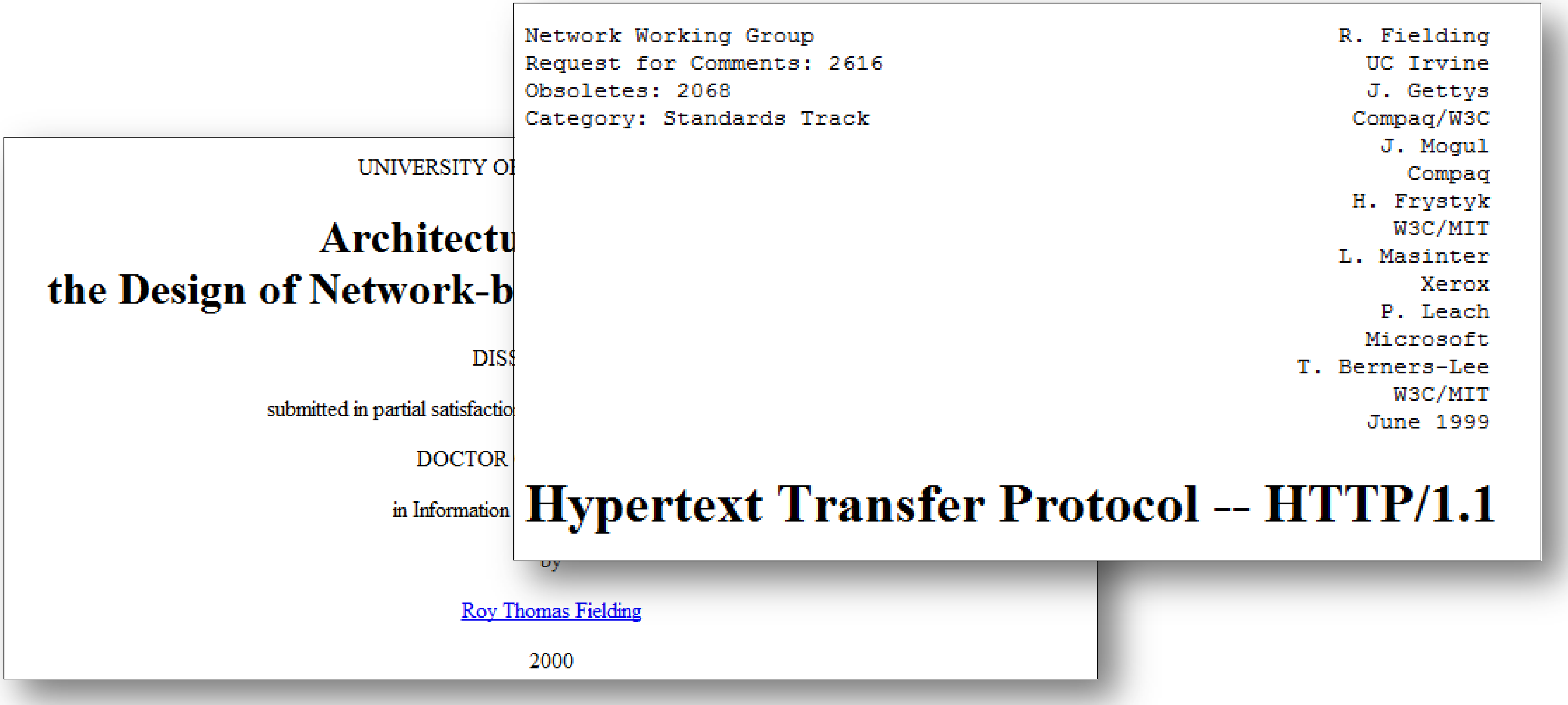
- REST is juxtaposed with RPC

REST and RPC

RPC	REST
Contract is a service and its operations	Contract is the uniform interface
Action semantics are specified out of band	Actions semantics are specified by the uniform interface and state transitions are specified by hypermedia controls embedded in representations
Error semantics are specified out of band	Error semantics are specified by the uniform interface
Limited caching support	Caching supported by all intermediaries which understand the uniform interface
Client and server share ownership of the URL namespace	Server owns the URL namespace
Inputs and outputs are tied to underlying runtime types	Inputs and outputs are tied to the media type specification
Can tunnel through multiple protocols	Tied to the uniform interface of the supporting protocol

REST and HTTP

- Most RESTful systems in use today are built on top of HTTP
- HTTP provides the necessary controls to build RESTful systems
- This is not a coincidence...



UNIVERSITY OI

Architectu
the Design of Network-b

DISS

submitted in partial satisfactio

DOCTOR

in Information

by

Roy Thomas Fielding

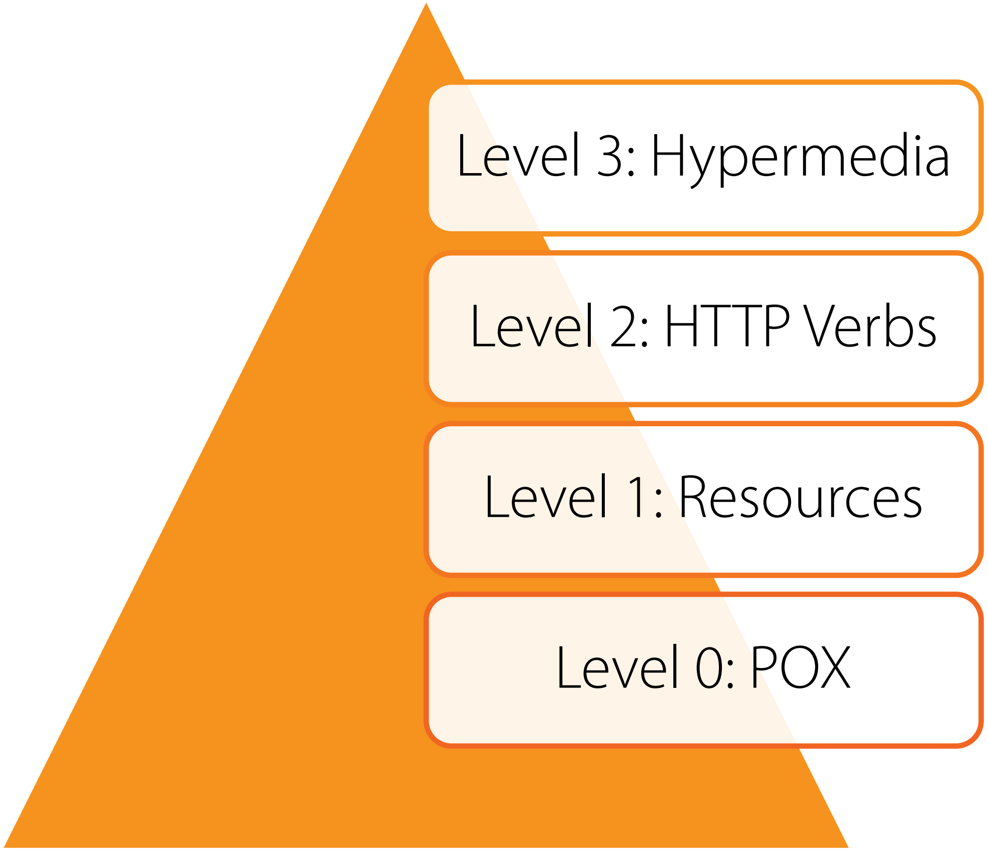
2000

```
Network Working Group                                      R. Fielding
Request for Comments: 2616                                   UC Irvine
Obsoletes: 2068                                              J. Gettys
Category: Standards Track                                   Compaq/W3C
                                                              J. Mogul
                                                                Compaq
                                                            H. Frystyk
                                                               W3C/MIT
                                                           L. Masinter
                                                                 Xerox
                                                              P. Leach
                                                             Microsoft
                                                        T. Berners-Lee
                                                               W3C/MIT
                                                             June 1999
```

Hypertext Transfer Protocol -- HTTP/1.1

Richardson's Maturity Model

- The path *towards* REST was described by Leonard Richardson and has been referenced in quite a few books blogs
- Only Level 3 in this model can actually be considered REST



Level 3: Hypermedia

Level 2: HTTP Verbs

Level 1: Resources

Level 0: POX

Summary

- **Why REST?**
  - Drivers
  - Benefits
- **Definition**
- **REST and Other Technologies/Architectural Styles**

References

- Fielding, Roy Thomas. *Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation, University of California, Irvine, 2000.
- HTTP/1.1 Specification
  - http://www.w3.org/Protocols/rfc2616/rfc2616.html