Programming the Cloud with TypeScript

Luke Hoban

QCON London
March 6th, 2019

Why am I interested in this topic?

Evolution of JavaScript

Browsers	**Node.js**	**??**
DOM APIs	OS APIs	AWS/Azure/etc APIs
SPA Frameworks	HTTP App/API Frameworks	Cloud App Frameworks

Infrastructure as **Code**

Infrastructure as ~~Code~~ **Text**

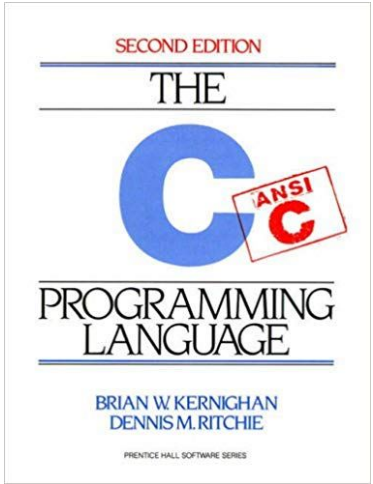
Infrastructure as **Software**

An Analogy

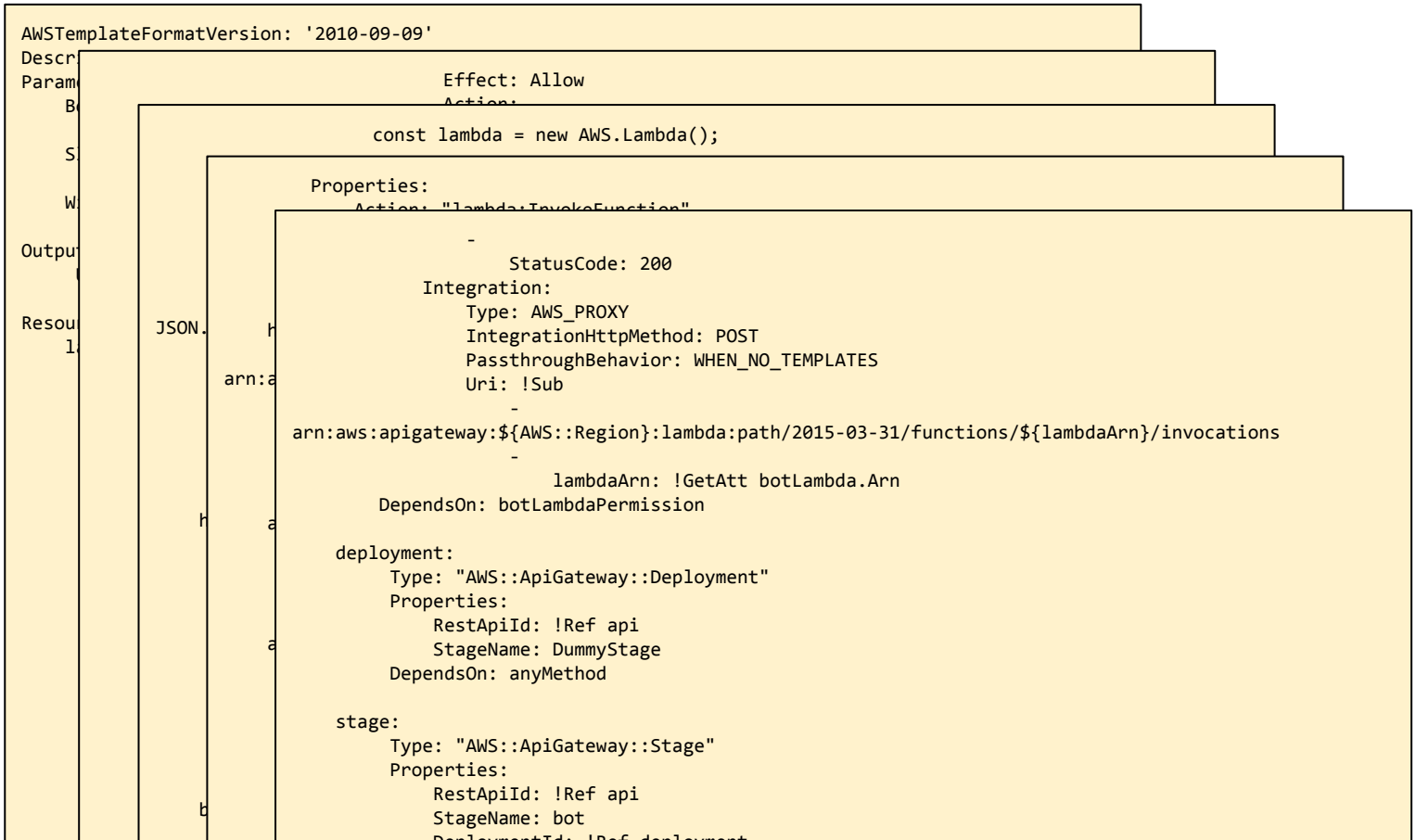
```
pushq   %rbp
movq    %rsp, %rbp
subq    $32, %rsp
leaq    71(%rip), %rdi
movl    $0, -4(%rbp)
movq    $13, -16(%rbp)
movq    -16(%rbp), %rsi
movb    $0, %al
callq   13
xorl    %ecx, %ecx
movl    %eax, -20(%rbp)
movl    %ecx, %eax
addq    $32, %rsp
popq    %rbp
retq
```

What's missing?

- Variables
- Loops
- Functions
- Abstraction
- C Standard Library
- Types



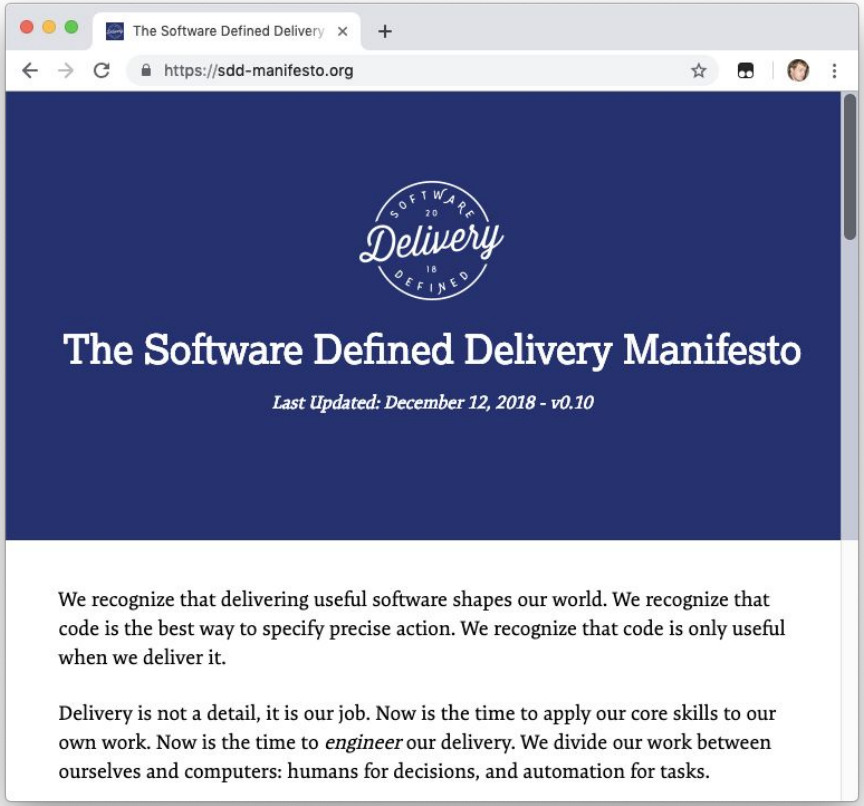
Infrastructure as Code



Demo

Infrastructure as Software

Other Similar Approaches

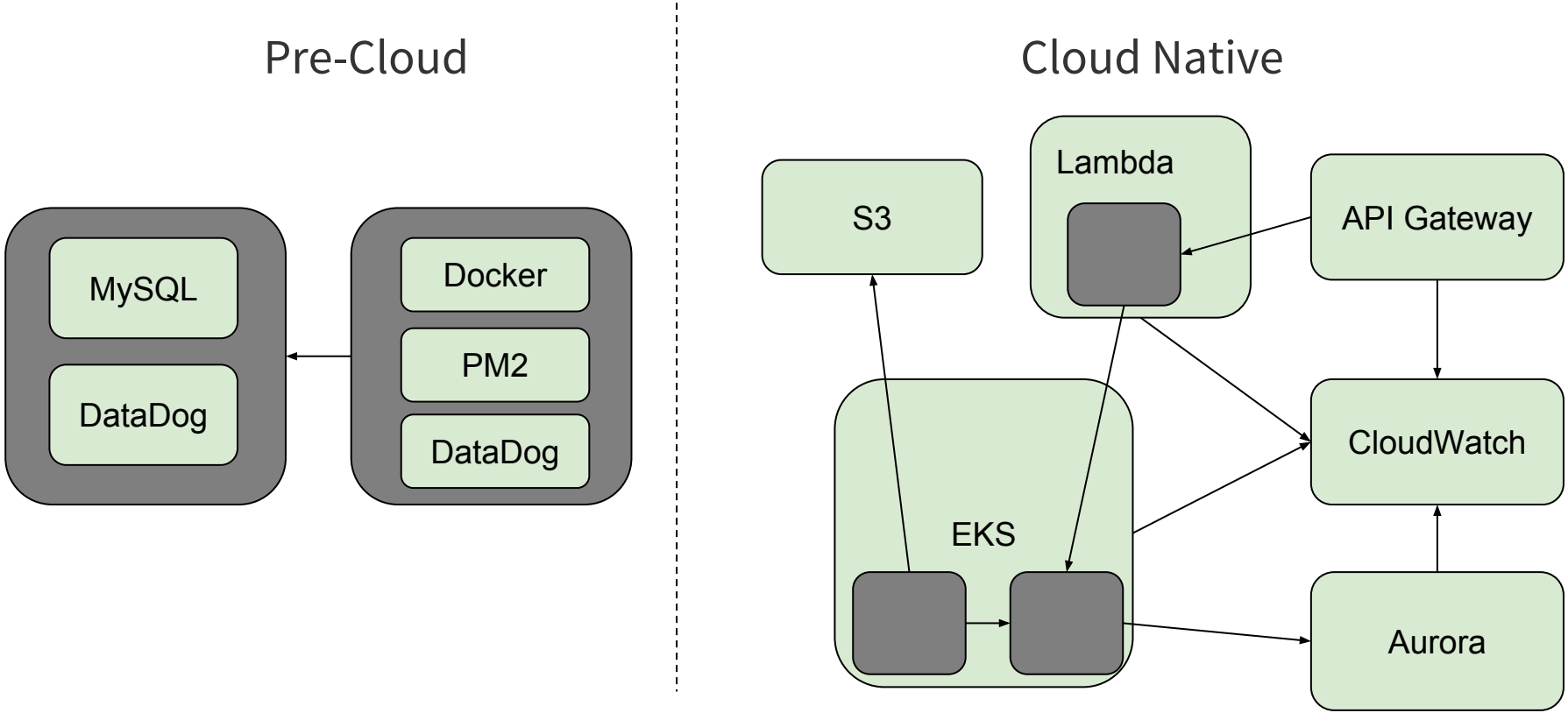


AWS CDK

Process Models

Page	Process	Stack
Transient	Finite lifetime	Lives “forever”
Script tags	ELF binaries	Desired State
Stateless	Largely stateless	Fundamentally Stateful

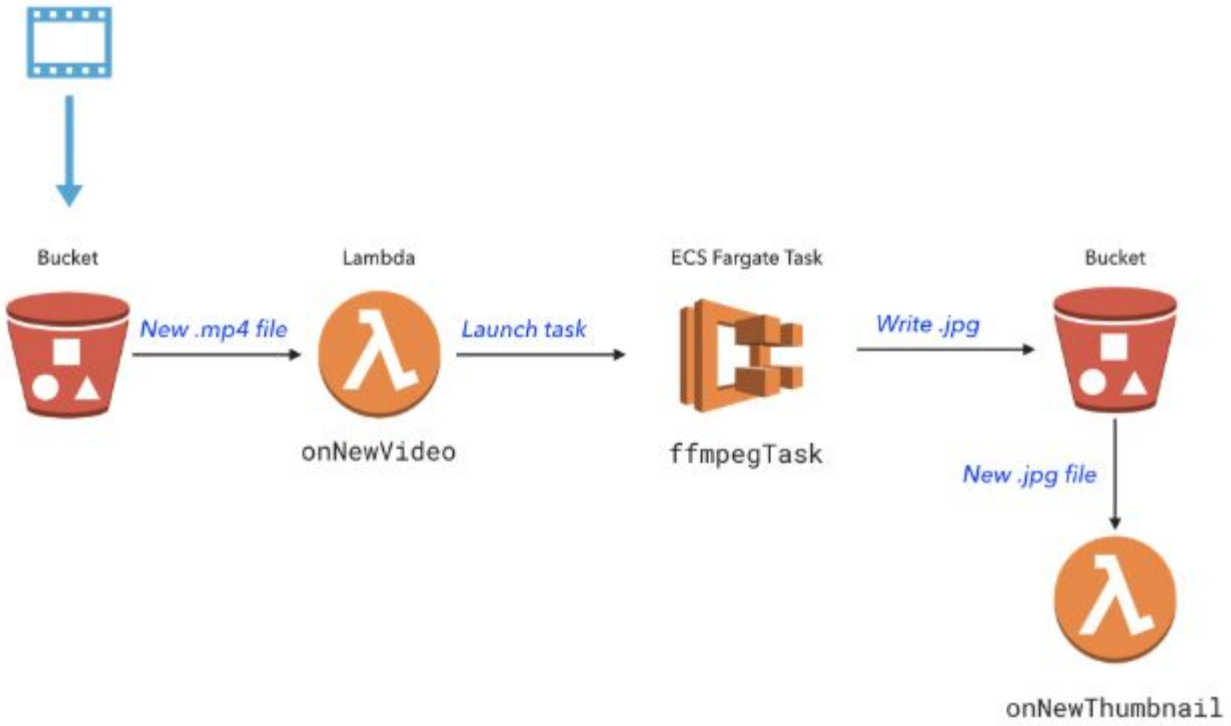
Implications of Managed Services



Demo

Breaking Down Barriers Between App and Infrastructure

Programming at the level of Architecture Diagrams



Demo

Programming at the level of Architecture Diagrams

Programming the Cloud

Continue the march of JavaScript from Browser to Server to Cloud

Apply Software Engineering to Cloud Infrastructure

Work at the right level of abstraction - raw infra or “architecture diagram”

Bridge the gap between App and Infra

A different kind of application model - “stacks” instead of processes

Thanks!

@lukehoban

lukehoban