Designing Continuous Delivery Into Your Platform

John Simone - Heroku
@j_simone
Continuous Delivery

Who haz it?
The Easy Bits
The Easy Bits

Version your code
The Easy Bits

Version your code

Control your deployments
The Easy Bits

Version your code

Control your deployments

Deploy behind feature flags
The Easy Bits

Version your code
Control your deployments
Deploy behind feature flags
Favor slow rollouts
The Tricky Parts
The Tricky Parts

Version your configuration
The Tricky Parts

Version your configuration

Planning your database migration
The Tricky Parts

Version your configuration
Planning your database migration
Having the ability to rollback
The Tricky Parts

Version your configuration
Planning your database migration
Having the ability to rollback
Canary deployments
Well... Damn
Well... Damn

Version your environment

Stateless and disposable app containers

No snowflake servers
There is no magic bullet
Just a new perspective: Build it into your development platform
What is a development platform?
What is a development platform?
How your code gets integrated and deployed
What is a development platform?

How your code gets integrated and deployed

How your environments are managed
What is a development platform?

How your code gets integrated and deployed

How your environments are managed

The container your code runs in
What is a development platform?

How your code gets integrated and deployed

How your environments are managed

The container your code runs in

Your operational data pipeline
What is a development platform?

How your code gets integrated and deployed
How your environments are managed
The container your code runs in
Your operational data pipeline
What is a development platform?

How your code gets integrated and deployed

How your environments are managed

The container your code runs in

Your operational data pipeline

It’s not new: you already have one
Application Versioning
Application Versioning

Ability to ensure deployments only happen via CI
Application Versioning

Ability to ensure deployments only happen via CI

100% repeatable deployments
Application Versioning

Ability to ensure deployments only happen via CI

100% repeatable deployments

Method of rollback built in
Environment
Environment

Disposable containers for code execution
Environment

Disposable containers for code execution
Recreate them on deployment/restart
Environment

Disposable containers for code execution
Recreate them on deployment/restart
Origin state only, everything else is ephemeral
Environment

Disposable containers for code execution
Recreate them on deployment/restart
Origin state only, everything else is ephemeral
Destroy them often
Configuration
Configuration

Mechanism for versioning
Configuration

Mechanism for versioning

Built into the container and immutable
Configuration

Mechanism for versioning

Built into the container and immutable

Changing it means a deployment
Configuration

Mechanism for versioning
Built into the container and immutable
Changing it means a deployment
Feature flagging is at the application layer
Putting it together
Putting it together

Visibility of changes
Putting it together

Visibility of changes

Visibility of the current state
Putting it together

Visibility of changes

Visibility of the current state

Deal with applications not infrastructure
Shameless Recruiting Plug
Shameless Recruiting Plug

Heroku is hiring:
all manner of engineers
product managers
technical account managers

jobs.heroku.com
salesforce.com/dreamjob
Questions?
Questions?

John Simone
Questions?

John Simone
Build & Packaging team - Heroku
Questions?

John Simone

Build & Packaging team - Heroku

@j_simone
Questions?

John Simone
Build & Packaging team - Heroku
@j_simone
heroku.com