

Modernizing an Enterprise

Ray Krueger

Ray Krueger

Chief Architect

Hyatt Hotels and Resorts

Ray Krueger

Chief **Technology** Architect

Hyatt Hotels and Resorts

About Hyatt

- Formed in 1957
- Over 90,000 employees
- Corporate Headquarters here in Chicago
- 652 properties in 53 countries
- 12 Brands
- #47 Fortune 100 Best Companies to Work For
- #49 Forbes America's Best Employers

Chief Architect

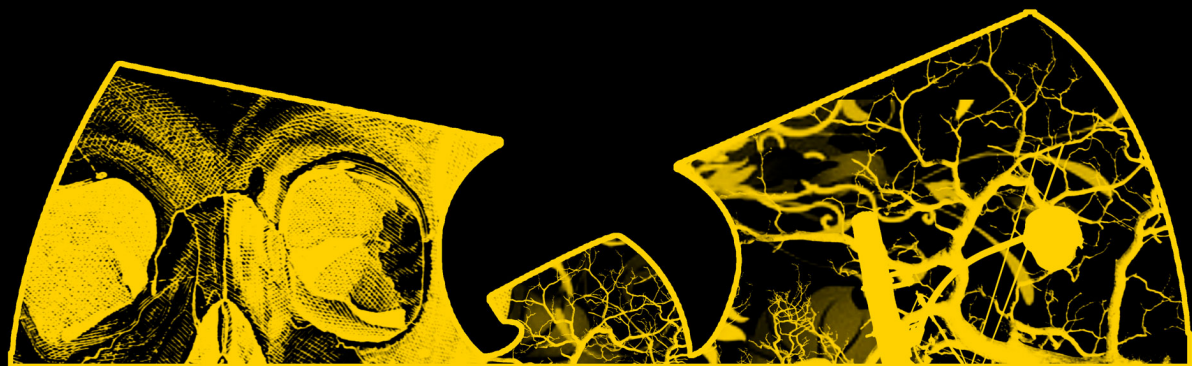
What the hell is that?

A Secret

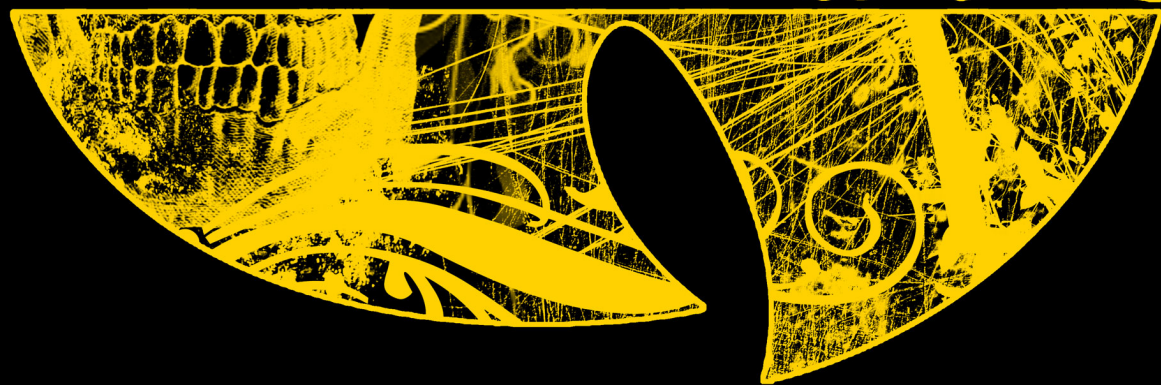


Where are we coming from?

Technology as a cost center



WU-TANG



So Many Datacenters

Work Harder not Smarter

Several Monoliths

No Real Standards

Where are we now?



Datacenter Consolidation

Technology as a Differentiator

Open Source First

Vendors where necessary

Standards as Source Code

Adopt an IETF style RFC Process

AUTOMATION

CAPSLOCK

Rundeck

The first domino

Ansible

Jenkins

Testing

Docker



Docker



Oh look, a container ship, how original.

Docker



Docker



Containers Everywhere?

No. Not yet.

Why Docker?

HYPE TRAIN WOO WOO

Static Binary

Common Build and Packaging

In order to get here



There's that damn container ship again.

You have to start here



..and here



..and here



A better analogy?



Infrastructure Concerns

Logging, Monitoring, Persistence,
even Operating Systems

Empowering Operations

How containers get to production



Frameworks

API Gateway

Netflix Zuul

Bellhop

HYATT® BELLHOP

UX Design System XL

FOUNDATIONS

PATTERNS

ELEMENTS

UTILITIES

MODULES

STANDARDS

RESOURCES

Bellhop is Hyatt.com's internal design system. This includes the global styling and interactions patterns used on Hyatt.com. This system also supports customizations for each of the Hyatt brands and for our umbrella Master Brand.

This is not a static system. Rather, it is a living, learning system. By driving this system off of the same code used in our production web applications, any changes can be rolled out across the site quickly and be reflected here immediately. No waiting for updates and no need to keep multiple copies in sync.

By using a shared style and pattern library, we can:

- Build pages and systems **quickly**. Focusing on new features, instead of repeating the same basic styling.
- Offer our users a **consistent** User Experience, instead of encountering slightly different looks and patterns with each section or page.
- Begin to **reuse** and repurpose code and ideas across the entire site, instead of spending time reinventing the same wheel.



Credits

Our pattern and style library is influenced by the work of:

- [Rizzo by Lonely Planet](#)
- [MailChimp's Pattern Library](#)
- [Github's Primer](#)
- [Pattern Lab](#)

Code

Source code for all the patterns, styles and documentation can be found in [Bitbucket](#). Pull requests are welcome and encouraged.

Bellhop

FOUNDATIONS

PATTERNS

ELEMENTS

[Buttons](#)
[Calendar](#)
[Checkboxes](#)
[Copy](#)
[Dividers](#)
[Headings](#)
[Links](#)
[Lists](#)
[Messaging](#)
[Navigation](#)
[Radio Buttons](#)
[Select Lists](#)
[Stepper](#)
[Tables](#)
[Text Inputs](#)

BUTTONS

CTA

Primary button for submissions.

SIGN IN

```
<button type="submit" class="button">Sign In</button>
```

SHOP CTA

Primary and secondary buttons for transactions.

BOOK NOW

```
<button type="submit" class="button-shop">Book Now</button>
```

Sparkplug

Service Framework

Service Oriented Architecture

Microservices

Nanoservices

Picoservices

Sparkplug

Service Framework

Minibar

Web Framework

Where are we going?



Legacy Systems

MOAR Experiments

MOAR Data



Event Driven Architecture

CONTAINERS



EVERYWHERE



CHICAGO

INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2016

goto; conference



Please

Remember to rate this session

Thank you!