

Microservice-Powered Applications

It worked for Voltron, it can work for you!



Bryan Soltis – Kentico Technical Evangelist

PLATINUM SPONSORS



GOLD SPONSORS



Asynchrony Labs

deckcommerce



GREENFIELD CREATIVE



CENTRIQ TRAINING



Progress



TEKsystems



Microsoft

SILVER SPONSORS



Conference Notebook & Party

Promotional Items & Attendee Lanyards

Attendee Swag Bags

Media



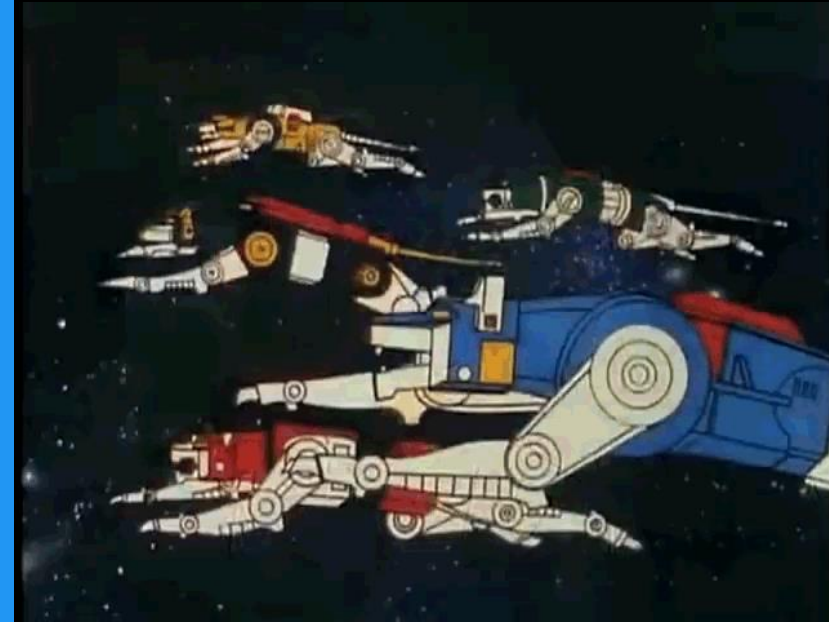
Voltron?

- Originally aired in 1984 (all others don't count)
- Based on Planet Arus (Castle of Lions)
- Originally a great robot split into 5 lions by Witch Haggar
 - Black Lion (Lightning) – Keith
 - Red Lion (Magma) - Lance
 - Blue Lion (Water) – Sven / Princess Allura
 - Green Lion (Wind) - Pidge
 - Yellow Lion (Sand) – Hunk
- Enemies
 - Witch Haggar
 - Emperor Zeppo
 - Prince Lotor
 - Robeasts!



What are micro services?

- Small, independent processes
- Communicate using language-agnostic APIs
- Decoupled building blocks for larger applications
- Remove single points of failure
- Best of breed services
- Allow for unique integrations/capabilities



Identify the bad guys

- Complex, restrictive content management
- Technology lock-in
- Inflexible, non-scalable platforms
- Ineffective, static search
- Complex integrations, processes
- Boring, static user experience



How are companies using them?

- Integrating systems
 - Microsoft Azure
 - Amazon AWS
- Scalable platforms
 - Netflix
 - Uber
- Remove technology monoliths
 - Groupon
- Serve multiple services
 - Amazon
- Deploy changes easier
 - Ebay





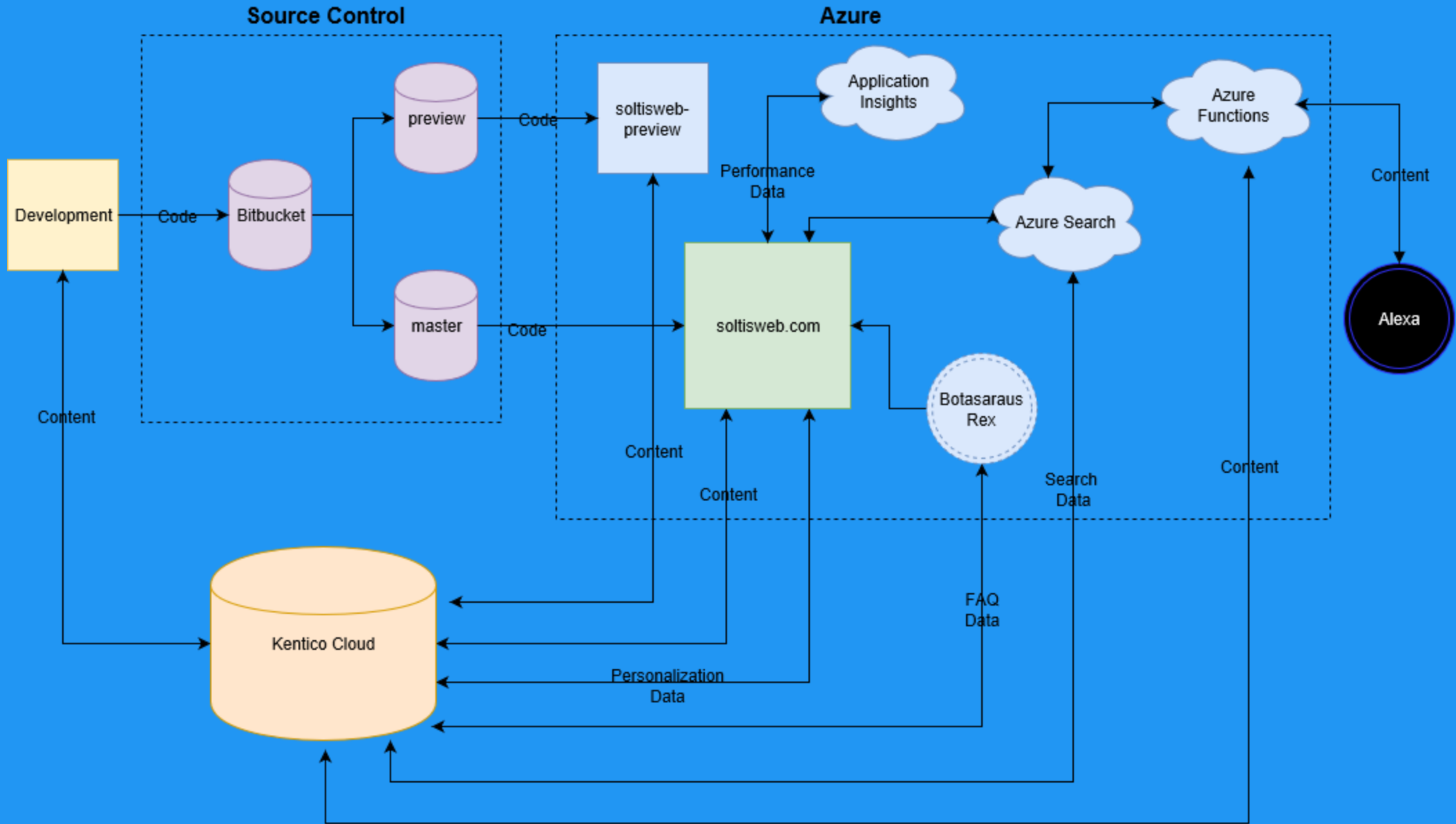
Assembling my team

- Kentico Cloud – SaaS Content Repository
- Azure App Services – Web Hosting
- Azure Search – SaaS Search
- Azure Functions – Kentico Cloud / Search Integration / Alexa
- Azure Bot Service – Interactive FAQs
- Application Insights – Performance reporting



Let's see it in action







PROS

- Functionality isolation
- Ability to choose services / systems
- Easily swap out services, if needed
- Blend multiple platforms / languages
- Smaller components



CONS

- Information barriers
- More calls between systems
- Larger, more complex systems
- Increased latency / processing time
- Multiple support channels



Learn more

- Why a microservices approach to building applications?
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-overview-microservices>
- Design patterns for microservices
<https://azure.microsoft.com/en-us/blog/design-patterns-for-microservices/>
- Kentico Cloud
<https://kenticocloud.com/>
- Azure Functions
<https://azure.microsoft.com/en-us/services/functions/>
- Azure Search
<https://azure.microsoft.com/en-us/services/search/>
- Azure Bot Service
<https://azure.microsoft.com/en-us/services/bot-service/>



Want to talk more?



Let's chat over
a beer at the
Social event!

Or maybe a
few...

Bryan Soltis

E-mail: bryans@kentico.com
Skype: kentico_bryans
Twitter: bryan_Soltis
GitHub: github.com/bryansoltis

devnet.kentico.com
facebook.com/KenticoCMS
twitter.com/kentico
linkedin.com/company/kentico-software

