Microservices architektura a kdy ji použít

Tomáš Herceg

Microsoft Most Valuable Professional (ASP.NET)
Microsoft Regional Director

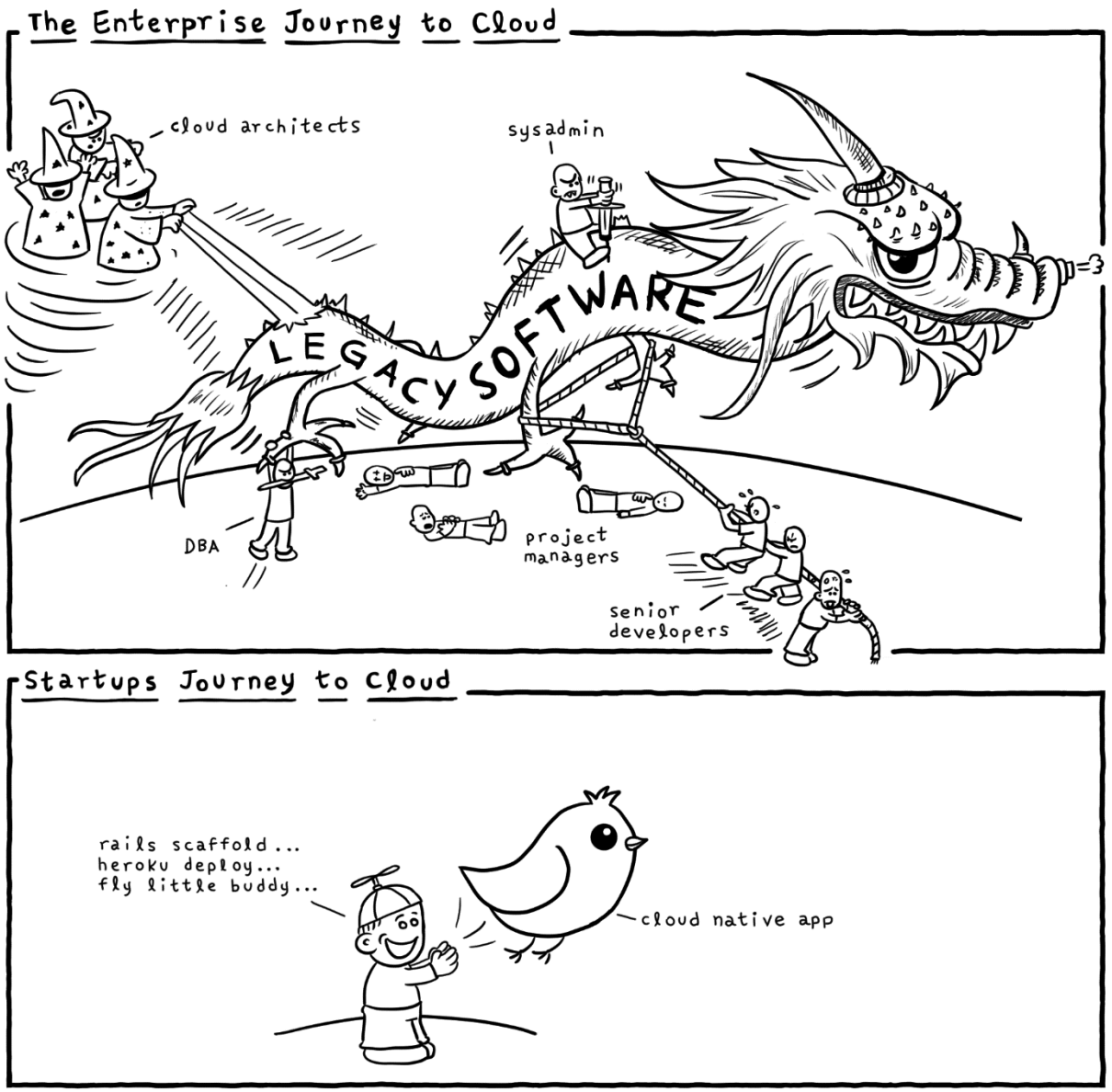
tomas.herceg@riganti.cz

@hercegtomas

tomasherceg.com/blog

riganti

DOTVVM



Daniel Stori {turnoff.us}
Thanks to Michael Tharrington

Microservices architektura

- Malé nezávislé služby
- Samostatné nasazování a správa
- Mohou využívat různé programovací jazyky
- Řeší jednu konkrétní oblast funkcí
  - Pozor na nanoservices
- Mají svůj vlastní storage
- 2012 – Netflix
  - „Fine-grained Service Oriented Architecture“

Monolit

- Kompiluje a nasazuje se jako jeden celek
- Může být modulární
  - Vertikální dělení
- Může být rozdělen do vrstev (DAL, BL, Presentation Layer)
  - Horizontální dělení

- Jedna sada technologií
- Jeden mindset
- Jeden (velký) tým

Nevýhody monolitů

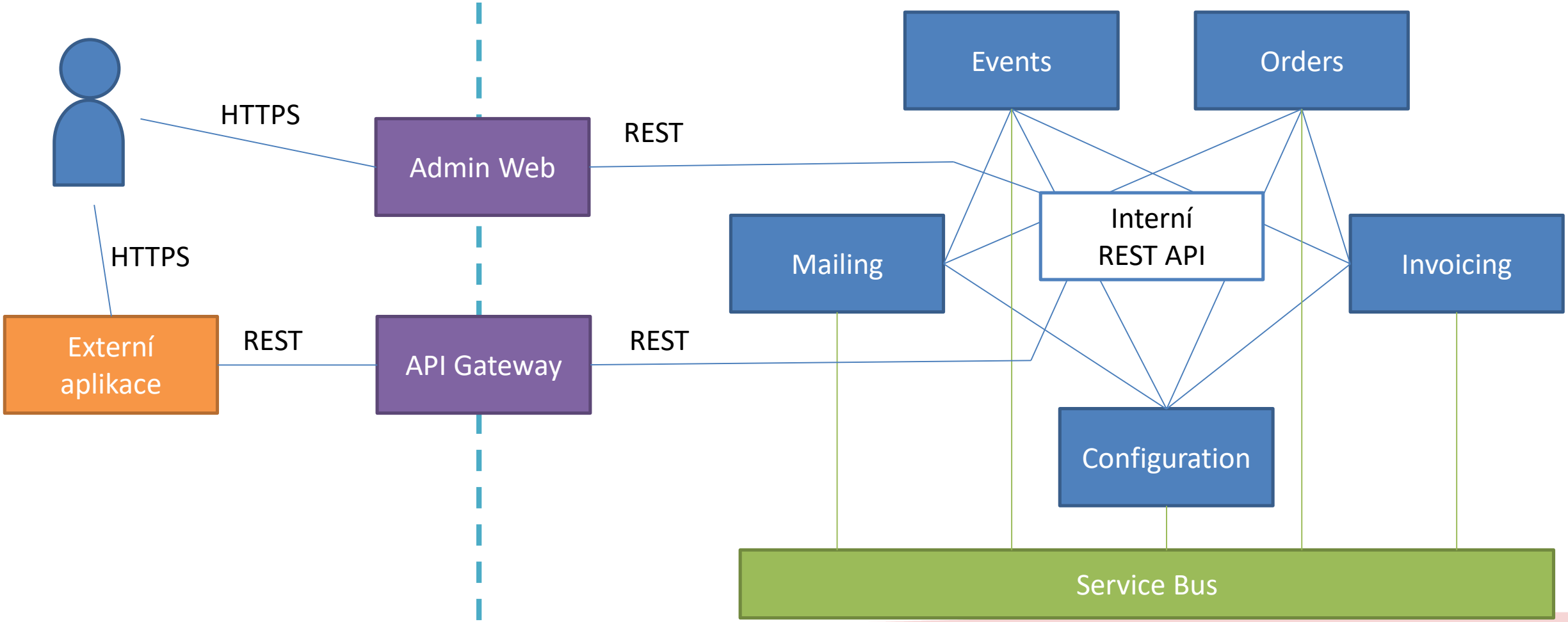
- Velká codebase
- Pomalé buildy (někdy)
- Omezené možnosti škálování
- Obvykle nelze nasazovat několikrát denně
  - Jak často nasazujete?
  - Bojíte se i při nasazení opravy překlepu na tlačítku, že se něco rozbije?

Určité projekty samy tíhnou k microservices

- Začalo to jako obyčejná webová aplikace
- Přibyly různé cron-joby a pravidelně spouštěné utility
- Integrační connectory a další nástroje
  - Vlastní databáze, pravidelná synchronizace dat

- Nemusí se nasazovat vše najednou
  - Bez infrastruktury pro CI/CI a testů je občas problém to uhlídat

Jak to v praxi může vypadat?



DEMO

Ukázka microservices aplikace

Výhody microservices

- Služby mají jasné ohraničení zodpovědnosti
  - Snazší orientace, ostatní služby jsou black box
- Jednoduché komunikační rozhraní
  - Typicky REST API a publish/subscribe
- Je možné je relativně snadno nahradit
  - Postupná modernizace aplikace

Výhody microservices

- O každou službu se může starat jiný tým
  - Každý nemusí rozumět všemu
- Každou službu lze nasazovat samostatně
  - Méně chyb
- Výpadek jedné služby ne vždy ohrozí celý systém
  - Většinou to dopadne lépe

Nevýhody microservices

- Mnohem komplexnější
- Vyžaduje nový přístup, nástroje a postupy
  - Testování
  - Kontinuální integrace
  - Automatizované (a chytré) nasazování
- Obtížnější on-premise hosting

Kdy microservices můžete použít?

- Pokud máte více týmů a potřebujete rozdělit odpovědnosti
- Pokud máte heterogenní řešení
  - Různé technologie, frameworky
- Pokud máte hodně customizované řešení
  - Stovky zákazníků, mnoho úprav pro jednotlivé zákazníky
- Pokud máte velké problémy se škálováním
  - A potřebujete každou komponentu škálovat samostatně

Jaké technologie využít?

- Azure Functions
  - Ideální pro jednodušší scénáře
  - Snadná integrace se storage
- Azure Container Service
  - Kubernetes, Docker Swarm, DC/OS
  - IaaS
- Azure Service Fabric
  - PaaS

Další informace a materiály

- eShopOnContainers
  - https://github.com/dotnet-architecture/eShopOnContainers

- 17:15 – 18:30 Úvod do Kubernetes a Azure Service Fabric

Update Days

Legacy aplikace

Konference a workshop se zaměřením na modernizaci legacy aplikací

28. - 30. května 2018
Microsoft, Praha

legacyapp.updatedays.cz

Update Conference Prague @update_conf update_conf

Update Conference Prague 2018

.NET - Cloud - Security

22nd - 23rd November 2018
Clarion Congress Hotel Prague

Carl FRANKLIN
Richard CAMPBELL

Authors of .NET Rocks!

updateconference.net

Update Conference Prague @update_conf update_conf

Update Conference Prague 2018

.NET - Cloud - Security

22nd - 23rd November 2018

Clarion Congress Hotel Prague

Julie LERMAN

The leading authority on Entity Framework

updateconference.net

Update Conference Prague @update_conf update_conf

Q&A

Tomáš Herceg

Microsoft Most Valuable Professional (ASP.NET)
Microsoft Regional Director

tomas.herceg@riganti.cz

@hercegtomas

tomasherceg.com/blog

riganti

DOTVVM