

EVIDEN

čeps_{a.s.}

Elektrizující platforma

Použití kontejnerů v messaging a aplikační platformě ČEPS

David Potůček, Solution Architect, ČEPS

Petr Holan, Head of OneCloud, Eviden

Obsah

01

Co je cílem

02

Představení Operational Framework platformy

03

Týmová synergie

04

Shrnutí



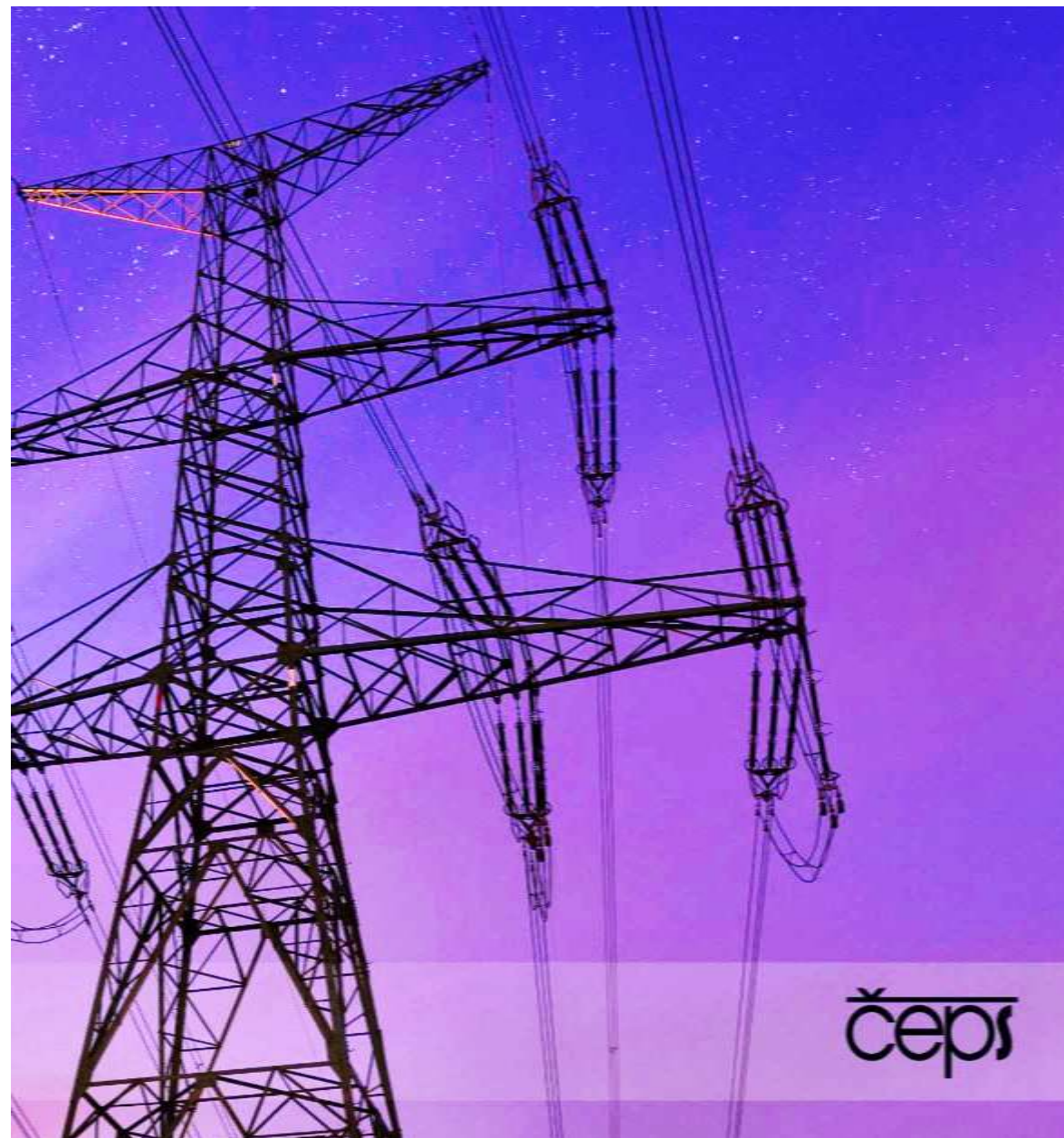
1 Co je cílem

Co je cílem

Základní předpoklady

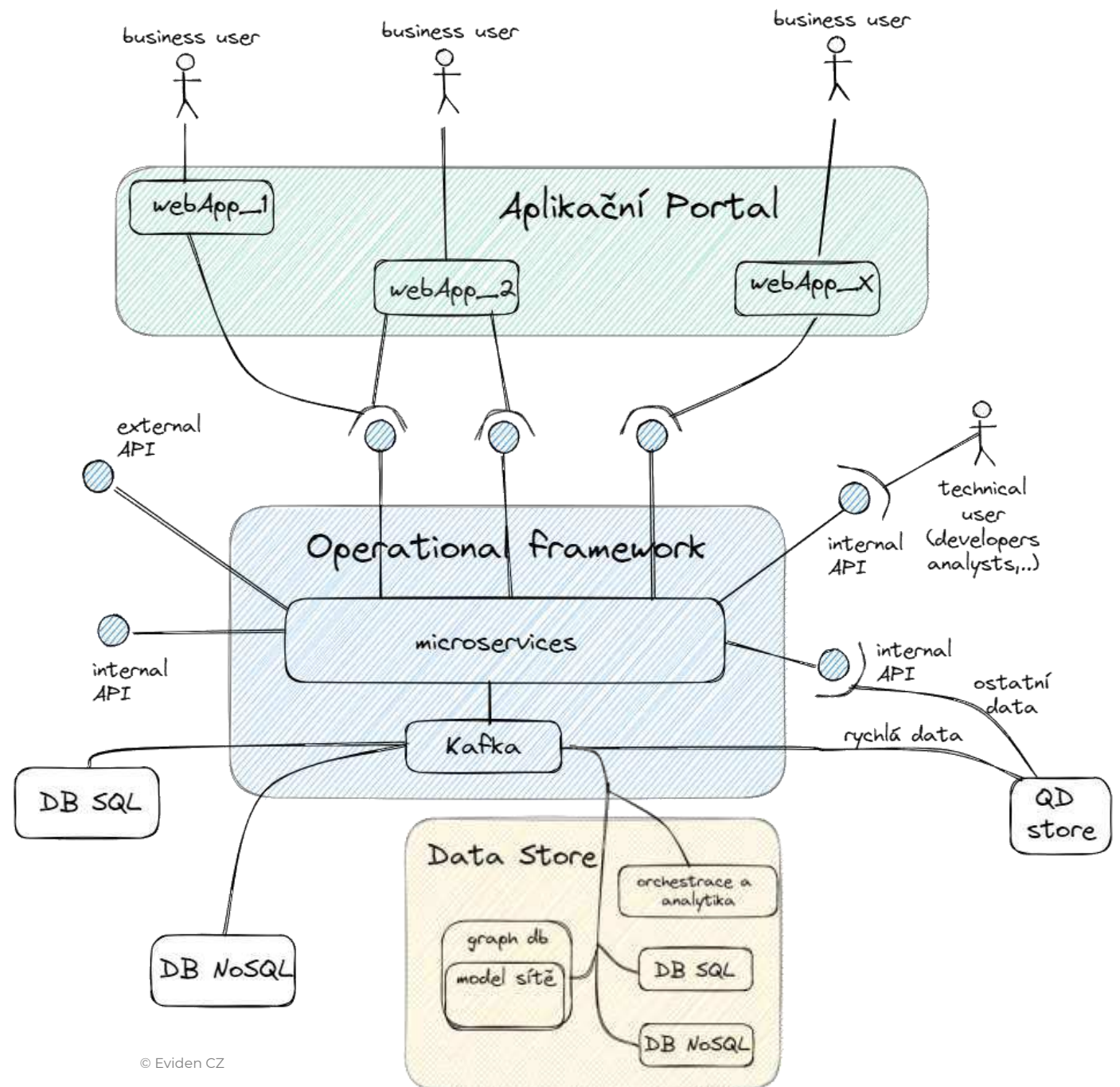
ČEPS

- Úvod do problematiky a zasazení do kontextu
- Specifika
 - Důraz na spolehlivost
 - Důraz na stabilitu
 - Dlouhodobé plánování
 - Technologická vzdělanost i na úrovni obchodu
- Softwarové vybavení firmy dnes
 - Microsoft, logika v DB, ...)
- Činnost dispečera



Co je cílem ČEPS HUB

- koncept tří vrstev ČEPS HUBu
 - Lightweight prezentace na portále
 - Mikroservisní integrační platforma
 - Databáze rychlé a bez vyšší logiky
- orchestrace dat a datových toků
 - Databáze
 - SQL, graph db, NoSQL (Mongo)



Co je cílem

Koncepce

- Integrovaný a messaging broker navržen jako interoperabilní s okolními informačními systémy interními i externími.
- Jádrem je produkt **Apache Kafka** pro robustní, bezpečnou a rychlou komunikaci jak v rámci interních komunikačních kanálů, tak i s okolím, realizující event-driven integrační a vývojový vzor.
- Plně integrované řešení v rámci všech modulů, které ho tvoří, s využitím všech příslušných relevantních částí platformy **Red Hat OpenShift**.
- Právě a zejména běhové prostředí této platformy založené na **kontejnerizační technologii** Kubernetes a modulární architektura umožňuje, že systém Operational Framework (OF) je škálovatelný, otevřený, rozšiřitelný a flexibilní.
- Vytvořeno jako **open-source** a tento licenční model se uplatní i u jednotlivých využívaných modulů třetích stran s výjimkou všech součástí komponent kontejnerizační platformy, kterou pokrývá licence **OpenShift Plus**, samozřejmě jsou řízené upgrady jednotlivých prvků, ze kterých je systém složen v rámci podpory.
- Neustálý přehled o aktuálním zatížení, sledování výkonu aplikací a možnost rychlé detekce a řešení případných chyb. Za tímto účelem jsou instalovány moduly umožňující dohled, ať již se jedná o tracing, logging či metriky výkonnosti.
- DevOps a **automatizace**, navrhnutí procesů a metodik, zajištění dokumentace a školení, rozvoj.



2 Představení Operational Framework platformy

Představení Operational Framework platformy

Popis modulů a jejich rolí v platformě 1

- **Messaging platforma – Kafka**
 - Komponenta zajišťující přenos zpráv v rámci OF a tvořící výkonné jádro platformy. Pro tuto funkcionalitu je použit open-source produkt Apache Kafka v podobě operátoru **Strimzi**.
- **Business integration services**
 - Sada mikroslužeb definovaných vlastností, funkcí a typů, tvořící softwarový framework, na kterém budou provozovány business úlohy.
- **Service Mesh – OpenShift Service Mesh**
 - Implementace service mesh řešení tvořící pozadí pro síť služeb, která se transparentně vrství na stávající distribuované aplikace. Výkonné funkce poskytují jednotný a efektivní způsob zabezpečení, připojení a monitorování služeb.
- **API Gateway – Tyk**
 - Komponenta pro správu konzumentů API a řízení API jako takových.
- **Log agregátor – OpenShift Logging**
 - Kombinuje produkty Elasticsearch a Kibana.

Představení Operational Framework platformy

Popis modulů a jejich rolí v platformě 2

- **Administrační klient**
 - Komponenta zajišťující správu existujících mikroslužeb, zobrazující jejich vazby a přístup konzumentů k nim, plus správu životního cyklu samotných služeb. Obsahuje procesy správy životního cyklu služeb, jejich evidence, schvalování a nasazování do provozu. Bez evidence služby v registračním modulu není možno službu nasadit do dalších prostředí.
- **Change Data Capture (CDC) – Debezium**
 - Open-source platforma pro sledování změn v databázích, umožňující reakci na tyto změny v reálném čase.
- **Red Hat Quay (INFRA)**
 - Registry a repository server, který slouží k ukládání, správě a distribuci kontejnerových obrazů.
- **IAM-IDP – Red Hat Single Sign-On (Keycloak)**
 - Jde o nástroj pro správu identit a přístupů (IAM), který poskytuje bezpečnostní vrstvu pro celou OF platformu. Zajišťuje autentizaci, autorizaci a správu uživatelských rolí a přístupových práv.
- **Scanner zdrojového kódu, Quality gate – SonarQube**
 - Nástroj pro statickou analýzu kódu, který pomáhá identifikovat a odstraňovat chyby, problémy s výkonem a bezpečnostní hrozby ve zdrojovém kódu aplikací v rámci build fáze.

Představení Operational Framework platformy

Popis modulů a jejich rolí v platformě 3

- **Secret management a úložiště kritických dat – HashiCorp Vault**
 - Jde o nástroj pro správu zabezpečených informací (secrets) a přístupových údajů. Poskytuje bezpečné ukládání a distribuci citlivých dat, jako jsou hesla, certifikáty nebo API klíče.
- **Systémy pro provozní monitoring OF – Prometheus, Grafana**
 - Soubor komponent pro zajištění sběru a vizualizace provozních logů a informací o stavu jednotlivých komponent.
 - Integrace na Zabbix
- **Systém správy verzí (VCS), Source code management (SCM), Nástroj na podporu vytváření a správy pipelines – GitLab, OpenShift GitOps (ArgoCD)**
 - Platforma OF využívá pro správu, ukládání a sdílení zdrojových kódů verzovací systém GitLab. Ten poskytuje vývojářům centralizované úložiště pro správu kódu, spolupráci a sledování změn včetně funkcí CI/CD. O deploy část se stará ArgoCD v podobě OpenShift GitOps.
- **Tracingový systém – Red Hat OpenShift distributed tracing**
 - Nástroj pro sledování a analýzu distribuovaných systémů. Jeho hlavním cílem je umožnit vývojářům monitorovat a diagnostikovat chování architektur mikroslužeb a dalších složitých systémů.

Představení OF platformy pro celkový kontext

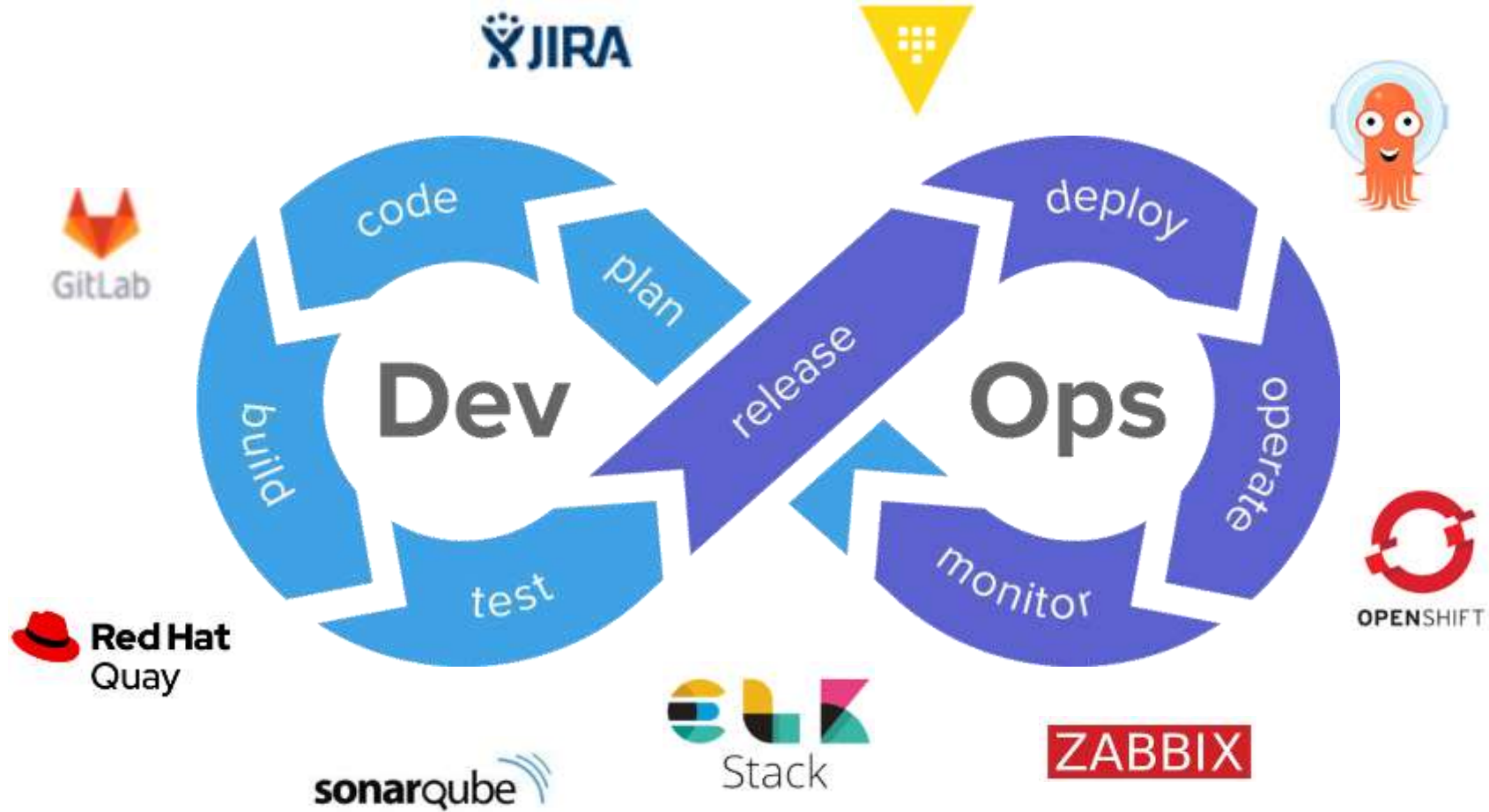
Shrnutí

- OpenShift Plus
- Nahraditelnost
 - Jakýkoliv element je v případě potřeby nahraditelný ekvivalentním
 - Včetně běhové platformy
- Open Source
 - Daný do souladu s Red Hat OpenShift Plus nabídkou



Představení Operational Framework platformy

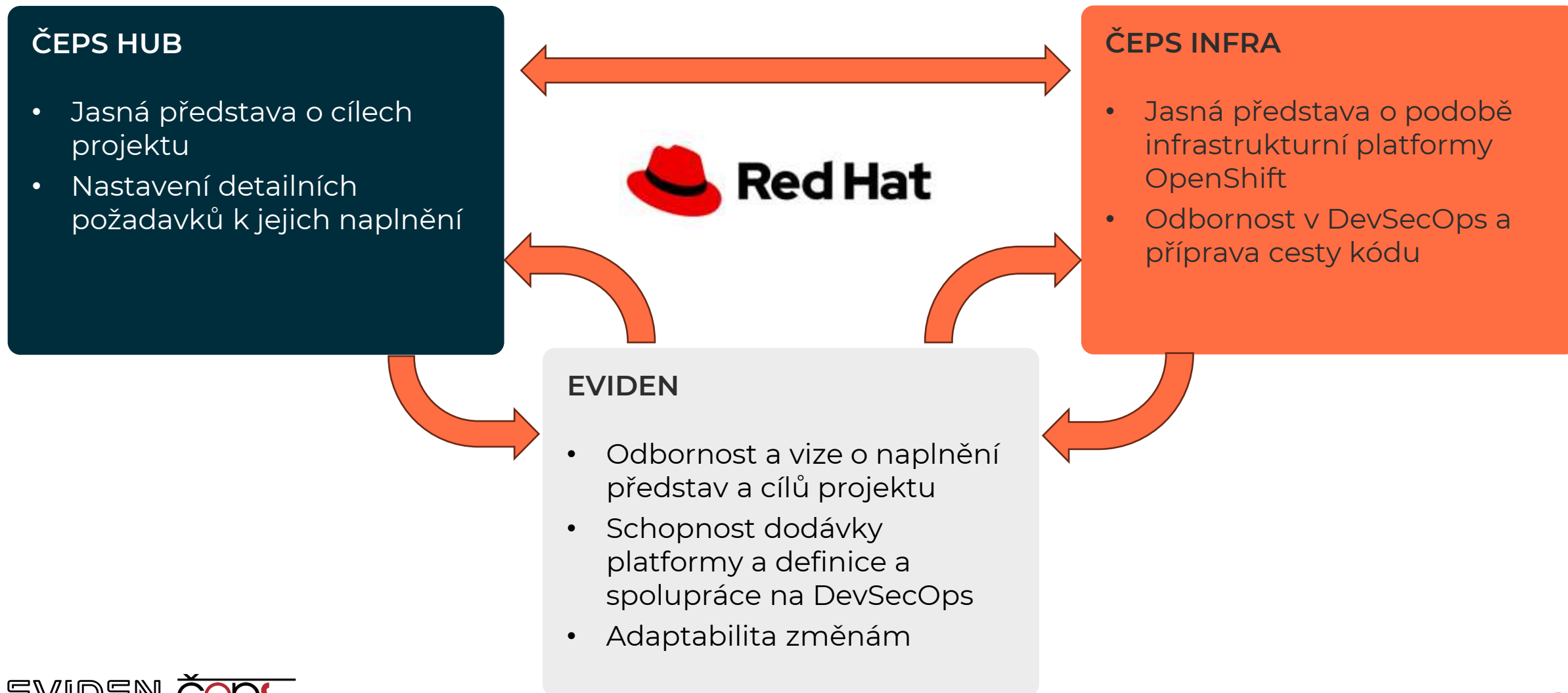
DevOps



3 Týmová synergie

Týmová synergie

Není to jen o technologiích



Týmová synergie

Je to o lidech

- **Komunikace**
 - Otevřenost
 - Bezpečné prostředí
 - Aktivní
- **Důvěra**
 - Spolehlivost
 - Empatie
 - Sdílení informací
- **Týmová práce**
 - Time management
 - Stále se učit
 - Neustálé hledání optimalizací
- **Orientace na zákazníka**
 - Adaptabilita
 - Optimalizace řešení
 - Navigace v dynamicky měnících se prioritách
- **Odbornost**
 - Best practices
 - Hardening
 - Sledování inovací
 - Školení

4 Shrnutí

Představení OF platformy pro celkový kontext

Shrnutí

- Co si odnést ...
 - ČEPS HUB obsahuje jednotné integrační messaging event driven rozhraní Operational Framework
 - ... které existuje v komplexní i v odlehčené verzi
 - ... založené na Kubernetes, kontejnerech a Red Hat OpenShift
 - ... jehož jádrem je **Kafka**
 - ... moduly této platformy jsou zaměnitelné a **open-source** či v licenci Red Hat
 - ... existují předpřipravené **templaty** a **dokumentace** pro procesy a postupy.
 - Děláme pokročilou kontejnerovou platformu a zatím se to daří !
 - Kdybyste nám chtěli poradit, co bychom mohli mít ještě lépe
 - ... nebo kdyby vás zajímalo víc a chtěli jste také takovou platformu, dejte vědět !



Děkujeme za pozornost

dpotucek@ceps.cz

petr.holan@eviden.com



Backup slides

Představení obecně vývoje pro OF - Mikroslužby

- **Samostatné služby**
 - Každá mikroslužba by měla být nezávislá a měla by mít vlastní databázi a prostředky. To zlepšuje škálovatelnost a zvyšuje odolnost systému.
- **API First**
 - Definovat rozhraní první (API) pro každou mikroslužbu, aby bylo jasné, jakým způsobem mohou ostatní služby komunikovat.
- **Autonomie**
 - Mikroslužby by měly být autonomní a schopné fungovat nezávisle. To znamená, že by neměly být silně závislé na ostatních službách.
- **Škálovatelnost**
 - Každou mikroslužbu lze škálovat nezávisle podle potřeby, což umožňuje efektivní využití zdrojů.
- **Monitorování a Logování**
 - Implementace robustních mechanismů pro sledování a logování, které umožní snadné odhalení a řešení problémů.
- **Bezstavovost (Statelessness)**
 - Mikroslužby by měly být navrženy tak, aby byly bezstavové, což zajistí škálovatelnost a odolnost.

Představení obecně vývoje pro OF - Event-Driven Pattern

- **Asynchronní komunikace**
 - Používání asynchronních mechanismů komunikace mezi službami, například pomocí front nebo publish-subscribe systémů.
- **Event Sourcing**
 - Ukládání a správa stavu aplikace prostřednictvím událostí, což usnadňuje sledování změn a obnovení stavu.
- **Event Bus**
 - Používání centralizovaného event busu pro komunikaci mezi mikroslužbami, což zjednodušuje a usnadňuje sledování událostí.
- **Idempotence**
 - Zajištění, že operace provedené v reakci na události jsou idempotentní, což minimalizuje potenciální problémy v případě opakovaného zpracování události.
- **Škálovatelnost**
 - Možnost škálovat event-driven systém podle potřeby, aby bylo možné zpracovat velké množství událostí.
- **Centralizovaná správa**
 - Použití centralizovaného řízení událostí pro sledování a správu celkového stavu systému.