

ID 11

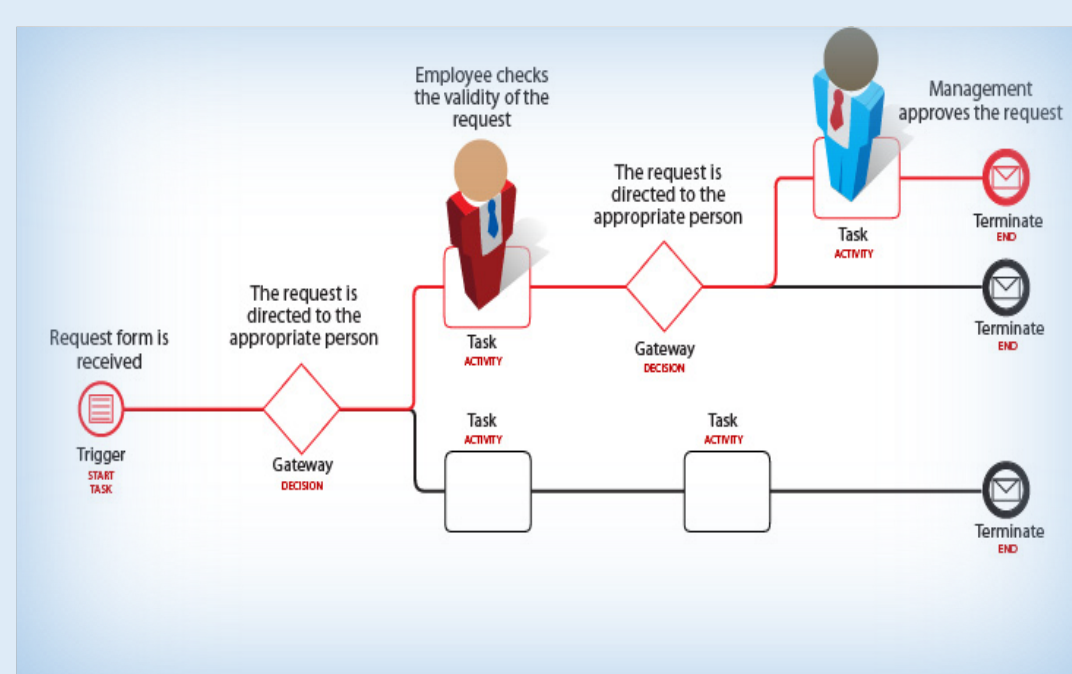
Business Process Representation as RESTful Resources

Co znamená pojem "business process"?

Business process je souhrn činností, transformujících (pomocí lidí a nástrojů) souhrn vstupů do souhrnu výstupů (zboží nebo služeb), přičemž tyto výstupy jsou určeny pro jiné lidi nebo procesy.

Příklady:

- schválení dovolené
- zpracování tiketu
- zahájení projektu
- ukončení projektu
- zahájení studia
- ukončení studia



Pro formální popis, implementaci a kontrolu business procesu se používají pomocné systémy (**business process engines** - BPEs):

- Bonita
- ProcessMaker
- SAP
- a další

Problémy:

- BPEs jsou často proprietární
- Implementace business procesu není přenosná mezi různými systémy
- Náhrada jednoho systému jiným je příliš drahá
- Změny procesů za jejich běhu jsou velmi obtížné

Řešení:

- Přidat ještě jednu úroveň abstrakce

Požadované vlastnosti:

- Jednoduchá integrace s již existujícími řešení
- Možnost jednoduše vyměnit jednu implementaci za jinou
- Možnost jednoduše vyměnit jeden systém za jiný
- Možnost mít variabilní části
- Možnost měnit variabilní části za běhu procesu

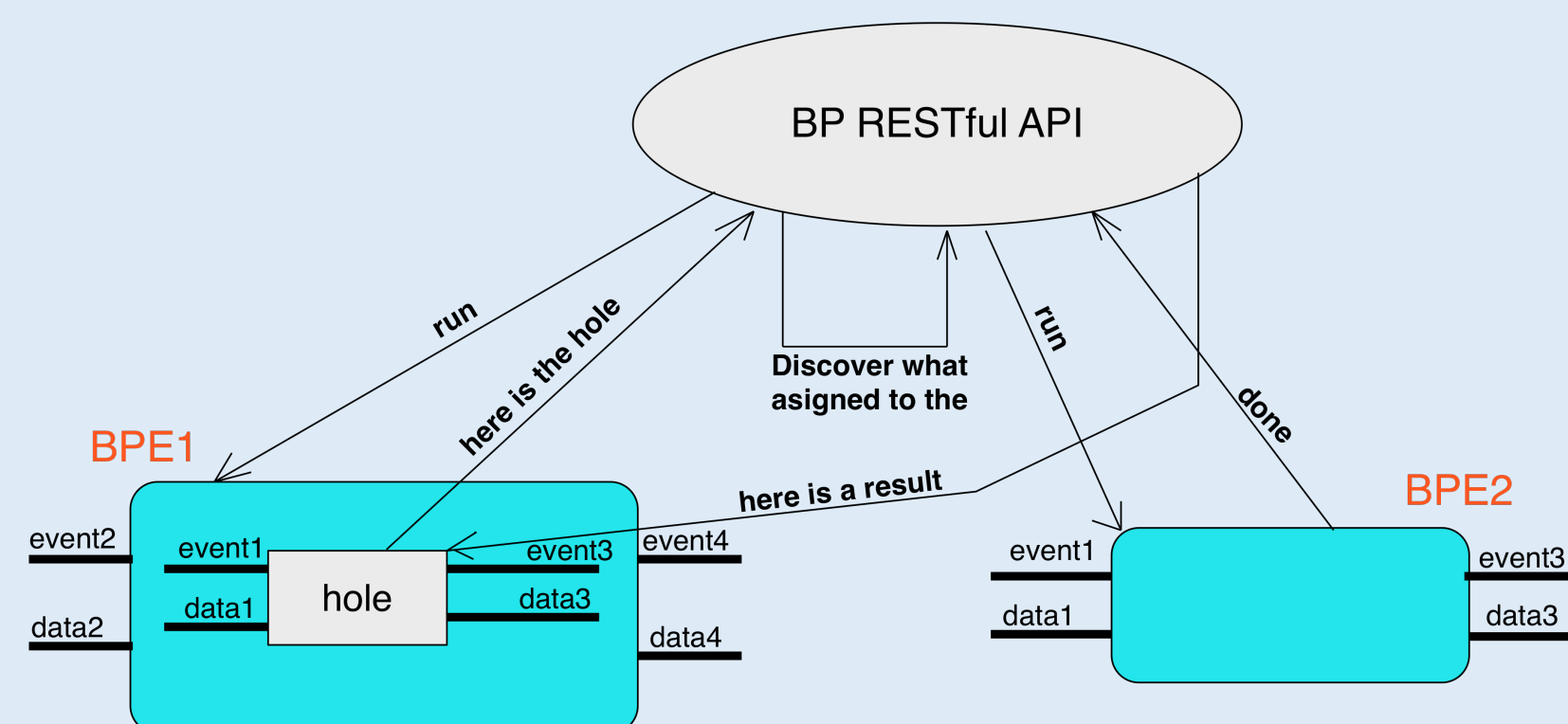
Jak na to?

Potřebujeme API. REST? SOAP? Něco jiného?

WWW je úspěšný projekt známý po celém světě. A ten využívá **REST** architekturní styl.



Jak to funguje



What is REST?

REST =
Representational
State Transfer

- Systém je rozdělen na zdroje
- Každý zdroj má unikátní identifikaci
- Každý zdroj má svoji vlastní reprezentaci
- Změna systému je prováděna přes změnu reprezentace nějakého zdroje

Co jsou zdroje v našem systému?

- *Template* - abstrakce druhé úrovně (representuje šablonu procesu, kde není známo, co je přiřazeno ke které hole)
- *Template holes* - popis variabilních částí
- *Pattern* - abstrakce první úrovně (representuje šablonu procesu, kde ke každé hole, může být přiřazen konkrétní proces)
- *Pattern holes* - variabilní části, ke kterým může být přiřazen konkrétní proces
- *Instance* - samotný běžící proces (běží na nějakém BPE)
- *Instance holes* - variabilní části v běžícím procesu (subprocesy)

Výhody:

- Využijeme sílu již existujících modelovacích nástrojů
- Uživatelé nemusí měnit prostředí
- Jednoduchá možnost orchestrace
- Možnost využití distribuovaného prostředí