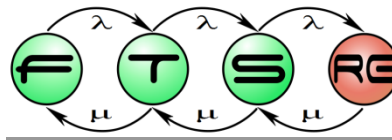


Mikroszolgáltatásokra épülő architektúra fejlesztésének és tesztelésének támogatása

Hallgató: Borlay Dániel

Konzulens: Szatmári Zoltán

Budapesti Műszaki és Gazdaságtudományi Egyetem
Hibatűrő Rendszerek Kutatócsoport



Probléma felvetés

- A vállalatok rengeteg különálló projektjének tesztelése sok erőforrást vesz el (főleg ha egymásra épülnek)
- Automatizált keretrendszer nélkül a fejlesztők és tesztelők kezelésben a verifikáció értékelésének a döntése
- Mikroszolgáltatások esetén ezek a problémák sokkal hangsúlyosabban jelentkeznek.
 - Sok apró szolgáltatás külön tesztelése
 - A teljes alkalmazás fordítása, integrálása, és tesztelése komplex feladat, de elkerülhetetlen

Feladat

- Bemutatni a mikroszolgáltatásokra épülő alkalmazásokat
- Megtervezni a folytonos integrációs keretrendszer működését, kitérve
 - Az elemi szolgáltatások, és
 - A teljes architektúra telepítésének kérdésére
- Megvalósítani a tervezett keretrendszert
- Értékelni a megvalósított keretrendszert

Feladat kifejtése

- Mikroszolgáltatások bemutatására egy minta alkalmazás készítése
 - Architektúra megtervezése
 - Telepítés, működés megtervezése és implementációja
- Mikroszolgáltatásokat támogató keretrendszer tervezése és elkészítése
 - Folytonos telepítést támogató keretrendszer tervezése, és elkészítése
 - Integrálás a meglévő minta alkalmazással

Continuous Integration/Deployment

- Folytonos integrálás
 - Metodológia
 - Napi több alkalmazás integrálás, fordítási, és tesztelési céllal
 - Gyakran artifactok az eredményei
 - Automatizált
- Folytonos telepítés
 - Folytonos integrációhoz képest az integrálás valamilyen dedikált helyre történik
- Mikroszolgáltatások esetén a különálló szolgáltatások miatt ez nélkülözhetetlen lehet.

Mikroszolgáltatás alapú architektúra

- Rendszertervezési minta
- Alkalmazás funkcióinak szétválasztása
 - Funkció, erőforrás, interfész szerint
- Jól definiált kommunikáció (Rest, SOAP, stb.)
- Külön fejlesztés
- Minimális logika a szolgáltatásokban
- Kicserélhetők a szolgáltatások
- Kevert technológiával is készíthető

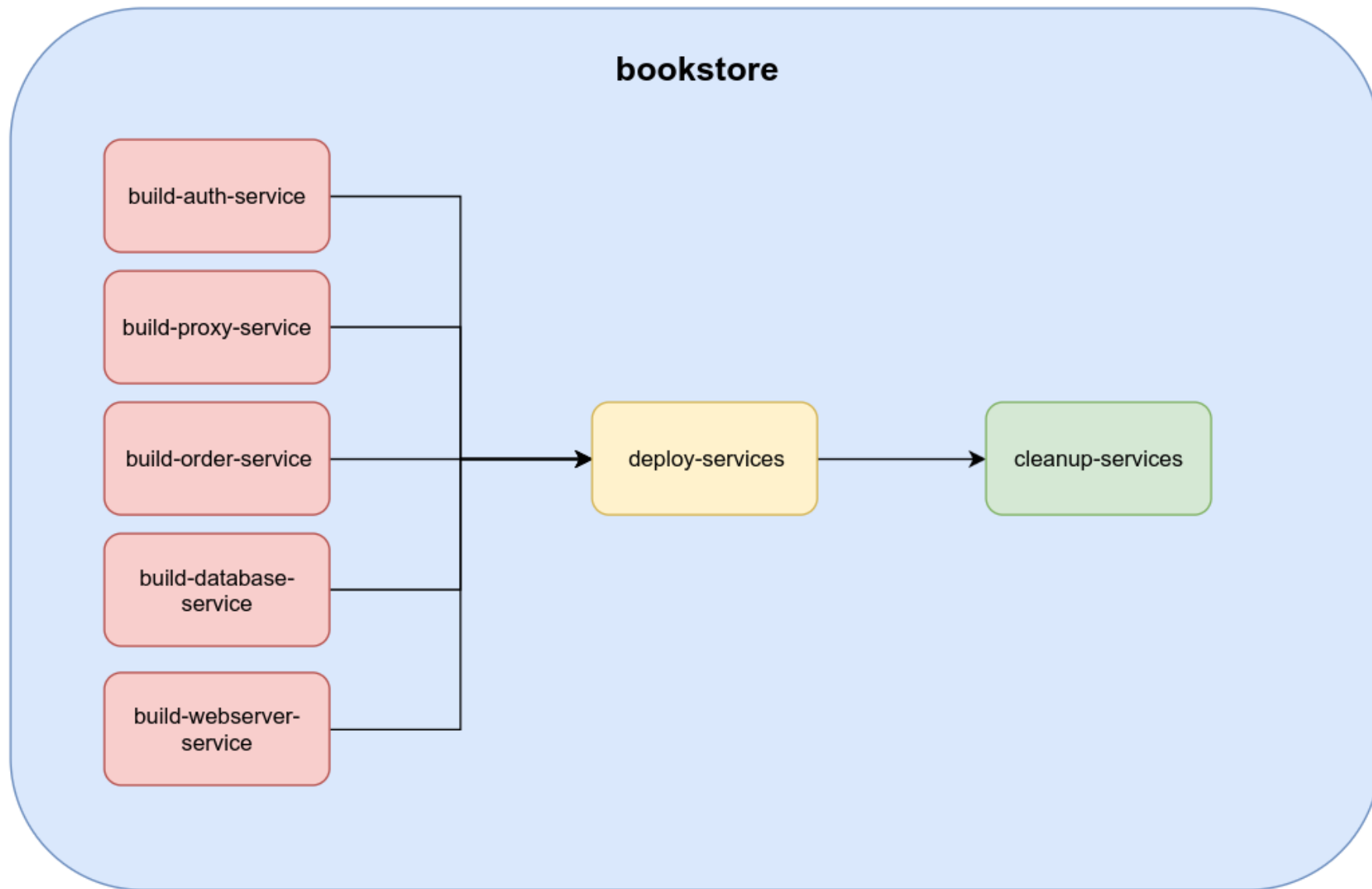
Technológiai áttekintés

- Számos technológiát kellett megismernem, hogy átfogó képet kapjak
- Mikroszolgáltatás építéséhez
 - Virtualizálás (Docker)
 - Hálózatkezelés (Consul, HAProxy, Ngnix)
 - Adatbáziskezelés (MySQL)
 - Webkiszolgálás (Apache webkiszolgáló)
 - Programozási nyelvek (Python, PHP, Java, Maven, Bash)
- Jenkins 2.0 (Continuous Integration keretrendszer)

Mikroszolgáltatást támogató keretrendszer

- Continuous Deployment elv átültetése a mikroszolgáltatások architektúrára kiélezve
 - Részek és egész fordítása, integrálása, tesztelése
- Tervezésnél absztrakt lépések meghatározása (Build, Deploy, Test)
- Implementáció
 - Teszt pipeline meghatározása
 - Alkalmazás integráció
 - Eredmények definiálása, és mentése

Keretrendszer



Keretrendszer



Jenkins > bookstore >

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Build Now](#)

[Delete Pipeline](#)

[Configure](#)

[Move](#)

[Full Stage View](#)

[GitHub](#)

[Pipeline Syntax](#)

[GitHub Hook Log](#)

Pipeline bookstore

This is the build pipe of the bookstore application.

Please find the artifacts of the application in the builder jobs.

Authentication: [Link](#)

Database: [Link](#)

Ordering: [Link](#)

Proxy: [Link](#)

Webserver: [Link](#)



[Recent Changes](#)

Stage View

Build History [trend —](#)

find X

- [#24](#) Dec 6, 2016 9:14 AM
- [#23](#) Dec 6, 2016 8:44 AM
- [#22](#) Dec 6, 2016 8:41 AM
- [#21](#) Dec 5, 2016 10:15 PM
- [#20](#) Dec 5, 2016 10:07 PM
- [#19](#) Dec 5, 2016 9:47 PM
- [#18](#) Dec 5, 2016 8:53 PM
- [#17](#) Dec 5, 2016 8:38 PM

Average stage times:

	Build	Deploy	Test	Cleanup
	3min 6s	6s	27s	46s
#24 Dec 06 09:14 No Changes	1min 40s	5s	2min 3s	46s
#23 Dec 06 08:44 No Changes	1min 36s	8s	13s failed	

Működés

- Új változtatás kerül a verziókezelőbe
- Elindul egy ellenőrző pipeline
 - Lefordulnak az alkalmazás részletei
 - Java program fordítás
 - Docker konténer építés
 - A lefordított konténereket elindítjuk, és kivárjuk az egymásra találás idejét (~ 20 sec)
 - Futtatjuk a teszteket, amik a teljes alkalmazás működését figyelik

Fejlesztési javaslatok

- Minta alkalmazás
 - Új funkciók hozzáadása
 - Kommunikáció finomítása
 - Virtualizáció bővítése
- Támogató keretrendszer
 - Verziókezelővel való összeintegrálás
 - Tesztek bővítése stabilitás, interfész és egység tesztelés terén
 - Ha a minta alkalmazás változik, a lépéseket bővíteni

Összefoglalás

- Megismertem a mikroszolgáltatásokra épülő architektúrát, és az ehhez felhasználható technológiákat
- Elkészítettem egy mint alkalmazást, amin keresztül megmutathatom a támogató keretrendszer működését
- Implementáltam egy folytonos telepítést támogató keretrendszert Jenkins alapokon

Kérdés

- Kérdés: Hogyan valósítható meg a szolgáltatások felderítése, ha a szolgáltatások nem ugyanazon a hálózaton futnak?
- Válasz: Ha minden szolgáltatás látható egymás számára, akkor a Consul, egy közös pont megragadásával (Proxy(k)) képes olyan változtatásokat eszközölni, ami elérhetővé teszi a szolgáltatásokat.

Kérdés

- Kérdés: A szolgáltatásokat a jelenlegi rendszer Docker képfájlokba csomagolja és telepíti. Mutassa be a csomagolás és telepítés menetét Docker konténerek nélkül egy választott felhő szolgáltatásban.
- Válasz:
 - Amazon felhő
 - Docker képfájl → Debian vagy RPM csomag
 - Docker run → SSH kapcsolat (install + szkriptek)

Kérdés

- Kérdés: Mi történik, ha leáll a HAProxy szolgáltatás? Hogyan tehető redundánssá?
- Válasz: Ha leáll a HAProxy szolgáltatás, az alkalmazás elérhetetlenné válik a felhasználó felé, kivéve, ha a webkiszolgáló szerver elérhető valamilyen IP címen vagy DNS bejegyzéssel. Több Proxy szolgáltatás párhuzamos futtatásával redundánssá tehető a rendszer, de megoldandó, hogy ne kapja meg minde a két szolgáltatás ugyan azokat a kéréseket.

Kérdés

- Kérdés: Hogyan valósítható meg a szolgáltatások felderítése, ha a szolgáltatások nem ugyanazon hálózaton futnak?
- Válasz: Ha nem ugyan azon hálózaton futnak a szolgáltatások egy közös pontot kell találni, amelyre kezdetben csatlakozhatnak a szolgáltatások, majd egy lista fenntartásával ez a funkció redundánssá tehető. (Ha minden szolgáltatás Consul szerver, akkor több csatlakozási pont is kialakulhat)