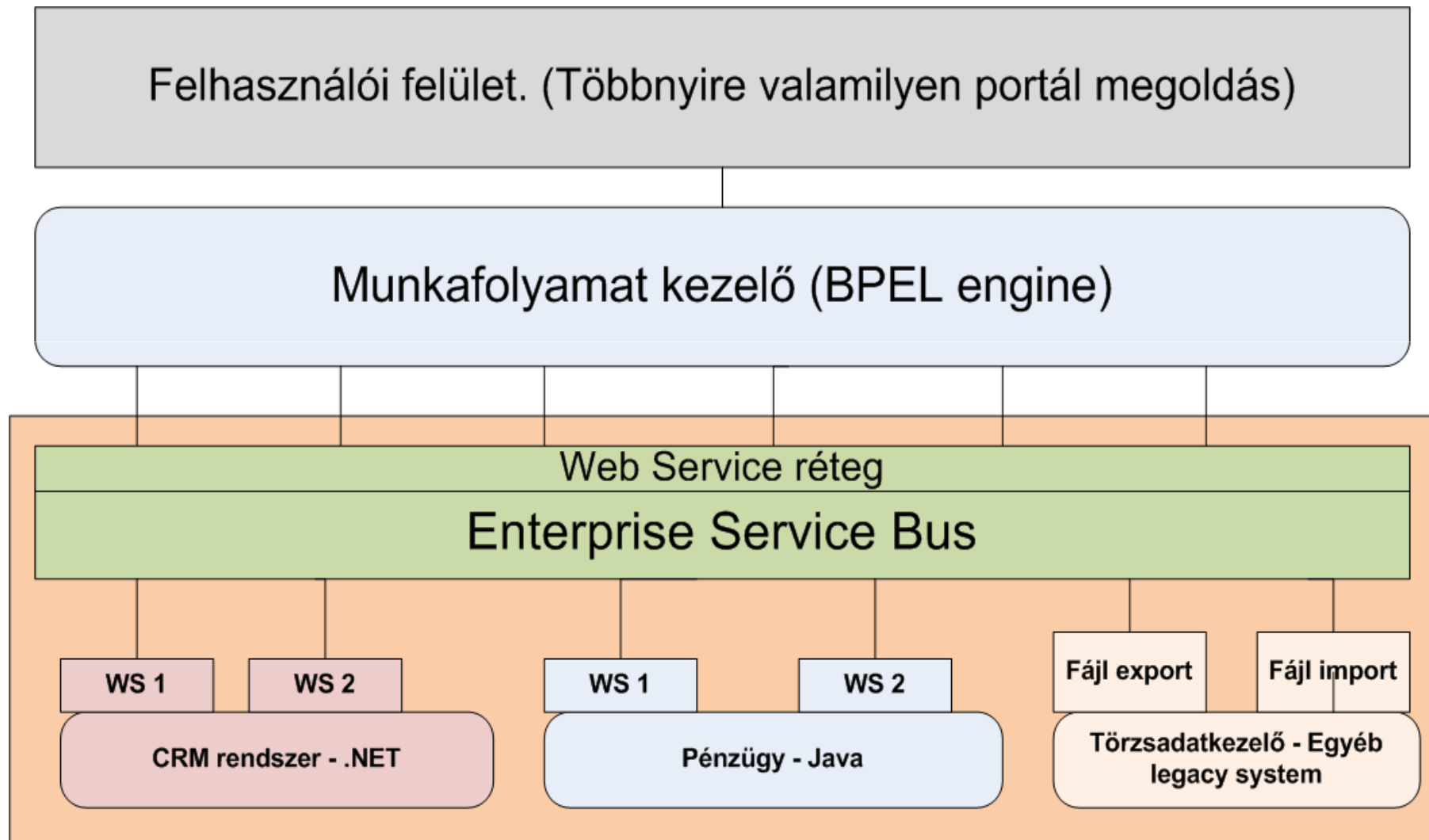


SOA a gyakorlatban

avagy rendszerintegrációk ESB alapon

Klasszikus SOA architektúra



Rendszerintegráció ESB-vel

- Általános célok
 - Különböző platformon futó és különböző protokollokon kommunikáló alkalmazások összekötése
 - A partneralkalmazások elől el kell fedni a technológiai különbségeket és az implementációs részleteket
 - Lehetőség szerint a partnerrendszerek natív interfészein keresztül kommunikáljanak egymással, akkor is, ha ezek közvetlenül nem kapcsolhatók össze egymással a technológiai különbségek miatt
 - Az integráció szabványos platformon legyen implementálva (pl. JBI)

Eszközök

- Apache Servicemix
 - Servicemix 3.3 ill Servicemix 4
- Fejlesztőkörnyezet
 - Eclipse IDE
 - Build: Apache Maven
- Fejlesztői és automatizált tesztelés
 - Unit tesztek mock végpontokkal
 - soapUI használata Web Service-ek tesztelésére

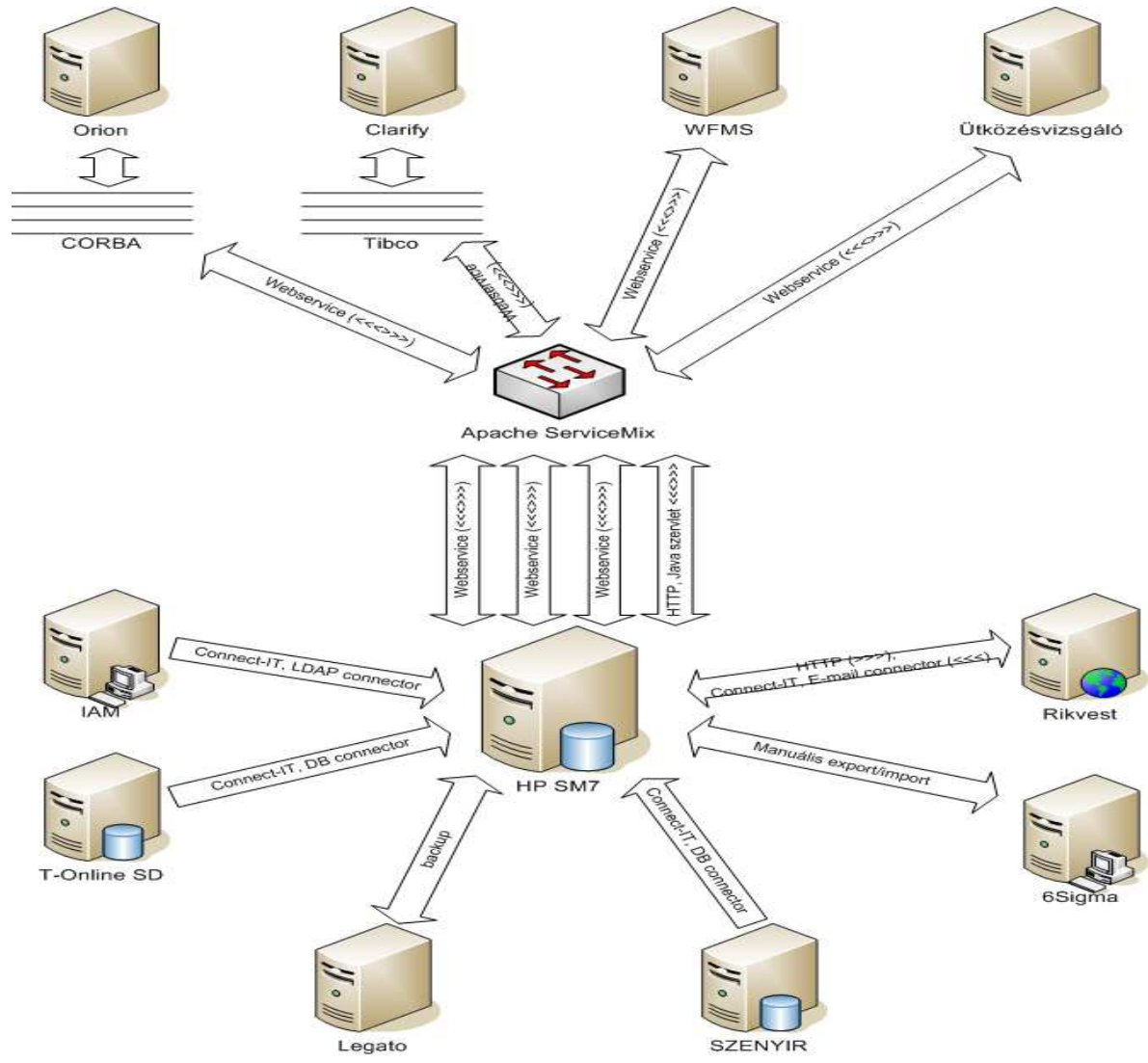
Az Enterprise Service Bus

- Normalized Message Router
- JBI komponensekből felépülő szolgáltatási végpontok
 - Bean, HTTP, HTTP-SOAP, JMS, CXF, Saxon Proxy, EIP, Camel, Scripting, File, FTP, stb.
- Consumer – provider végpontok közötti message exchange
- Enterprise Integration Patterns
 - Message Routing: Content Based Router, Content Enricher, Message Filter, Recipient List, stb.
- Routolás, üzenet transzformáció: Java, Scripting, XSLT, WS Notification
- Támogatott protokollok: File, FTP, DB, HTTP/S, JMS, SOAP, TCP, Corba

Egyéb jellemzők

- Spring integráció
- JMX management
- Hotdeploy
- Service Unitokból felépülő Service Assembly-k a telepíthető artifact-ok.
- Futtatható önálló alkalmazásként, Java EE szerverre telepítve, vagy egyéb Java alkalmazásba ágyazva.
- Open Source, Java Standard-eken alapuló architektúra

Gyakorlati alkalmazás



Egy Service Assembly logikai felépítése

