Bus de services

ou Enterprise Service Bus (ESB)

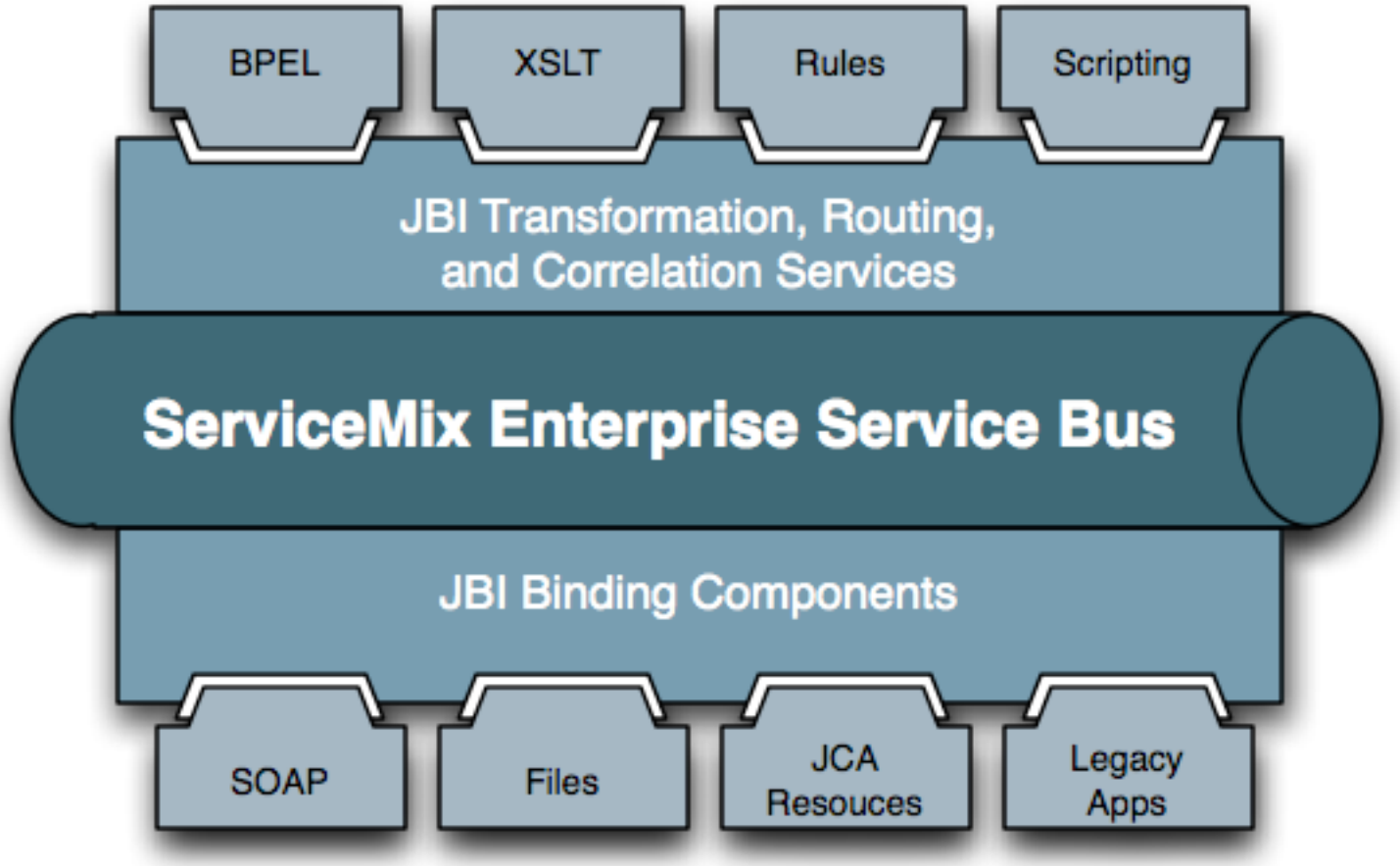
Architecture orientée services

- Architecture à base de composants
- Encapsulation de code métier dans ces composants
  - Permet un accès programmé aux fonctionnalités métier
  - Permet de séparer l'implémentation du métier de l'organisation des briques les unes avec les autres
- Dans une implémentation basique
  - Les services (en particulier les adresses) sont relativement fixes
    - Malgré l'aspect "loose coupling" des services web
  - Difficile à gérer dans le cas où on a beaucoup de services interagissant avec d'autres services

Bus de services

- Serveur ou ensemble de serveurs
  - Hébergeant des services
  - Servant d'intermédiaire dans la transmission des messages entre services
    - Cette couche intermédiaire donne plus de souplesse dans le traitement des messages
    - Permet d'abstraire le routage des messages
  - Les services ne sont pas forcément web
    - Dans le sens où un service peut envoyer un message à un autre service sans passer par un protocole réseau

Exemple d'ESB



http://servicemix.apache.org

ESB en Java: JBI

- Objectif: améliorer l'intégration au niveau service
- JBI: API de normalisation de message
  - Cœur (payload) du message: XML
  - Ensemble de headers contenant des métadonnées
  - Des attachements (*c.f.* XOP/MTOP)
- Ressemble à une API pour des services Web
  - Simplifiés
  - Sans l'aspect "web"

Composants d'un bus de services

- Des composants de traitement des messages:
  - Implémentation de besoins métier (e.g. bout de programme Java)
  - Fabrication d'un service comme un assemblage de services (e.g. process BPEL)
- Des composants de transformation de message:
  - Passage d'un schéma à un autre (interopérabilité à haut niveau)
  - Routage
- Des composants d'interfaçage
  - Interaction avec "l'extérieur" du service
    - Bindings SOAP, protocoles réseaux
    - Interactions avec le système environnant

Composants d'interfaçage

Binding components

- Permet de convertir ou de créer un message normalisé (xml+headers):
  - en/à partir d'un autre format pour l'intégration au SI
    - Message SOAP, format propriétaire …
  - De l'envoyer via un protocole approprié
    - SOAP sur HTTP, XML brut sur HTTP, fichier envoyé par FTP …
- Effectue la normalisation du message
  - Extraction des headers SOAP par exemple

Composants de traitement de message

Service Engine

- Service (web) implémentant du code métier
  - Implémentation basée e.g. JAX-WS
  - EJB, POJO, Javascript, Ruby …
- Processus Métier (e.g. BPEL)
- Gestion sous forme événementielle
  - Ensemble de règles logiques déclenchées à la réception du message (e.g. Drools)

Composants de routage

Service Engine

- Composant servant à sélectionner le composant auquel est destiné un message
  - Par du code java, javascript, ruby, …
  - Par des règles (Drools)

⇒Routage dynamique pouvant dépendre du contenu des messages

Composants de transformation

Service Engine

- Normalisation des messages
  - Simplification du code
  - Les données peuvent différer
    - Dans leur présentation
    - Dans leur structure
- Transformation du contenu des messages
  - XSLT/XQuery
  - Code Java, Javascript …

Schémas d'échange de messages

Message Exchange Patterns

- Basé sur les schémas d'échanges WSDL:
  - One-way
  - Reliable One-way
  - Request-Response
  - Request-Optional response

Routeur de message

Normalized Message Router

- Sert à adresser et à transporter les messages
- Services
  - consumers (clients)
  - providers (services)
    - Implémente une interface WSDL
  - Communique avec le bus à travers un "delivery channel"
- Messages
  - Message exchange → transmission de message

Points d'accès, routage et NMR

- Points d'accès
  - Externes
    - utilisés via des composants d'interfaçage
  - Internes
    - Interagissant directement avec le NMR
- Routage
  - Explicite
  - Implicite, en fonction du type de service
    - Résolution par le NMR
  - Dynamique, via des adresses transmises dans les messages
    - e.g.: callback

Routage et service connections

- Certaines informations de routage peuvent être fournies lors de la mise en places des composants
  - Service connection
    - Côté "consumer": le service à utiliser (type ou endpoint explicite)
    - Côté "provider": le endpoint correspondant

Appel à un service et routage

- Routage implicite:
  - Endpoint correspondant à la partie consumer d'une connection → message dirigé vers la partie provider de la connection
- Routage explicite:
  - Endpoint correspondant à un service réel du NMR
    - Routage par le composant consumer

Déploiement

- Service Unit
  - Correspond à 1 composant
  - Comprend des fichiers de configurations
    - e.g. quelle feuille de style XSLT pour un composant xsl
  - Des fichiers de déploiement
    - Décrit les services fournis et utilisés par ce SU
    - Les noms de services définis ici peuvent être utilisés par le NMR pour le routage implicite
- Service Assembly
  - Ensemble de "service units"
  - Configuration
    - Des connexions entres les services units
    - Des dépendences vis à vis des services externes

ServiceMix 4: OSGi

- Modèle de déploiement alternatif
- Bundle OSGi
  - Contenant 1 ou plusieurs composants
  - En général du même type
    - POJO CXF
    - Queues/Topics JMS
    - Etc
- Fichier de configuration type "Spring"
  - Reprend les informations fournies dans les déploiements JBI

Références

- Standard JBI
  - http://jcp.org/aboutJava/communityprocess/final/jsr208/index.html
- Javadoc JBI
  - http://servicemix.apache.org/javadocs.html
  - + intégrée au standard JBI
- Page de Servicemix
  - http://servicemix.apache.org