



Conception d'Applications à Services

German Vega

Equipe Adèle
Equipe Adèle

1

Equipe Adèle
Equipe Adèle



Les tendances lourdes

- Le découplage (réutilisation, grain, abstraction)
 - ❖ Retirer du code la connaissance du contexte, et l'initiative
 - ❖ Ajout de méta données
 - Exprimer ce que fait le composant
 - Interface (signature)
 - La « sémantique » des fonctions : l'intention, les contraintes, le protocole d'utilisation
 - Les liens avec l'environnement
 - Les dépendance : les fonctions requises (nom, signature, caractéristiques)
 - Les caractéristiques (fonctionnelles, QoS, NF ...) de ces fonctions
 - ❖ Contrôle externe
 - Prendre les décisions à la place des composants
 - quels composants, quelles connexions, quand, pourquoi
 - On se charge de faire communiquer les composants (connexion, médiation, transformation)

06 novembre 2007

2

Découplage

- Découplage fonctionnel
 - ❖ Réduire les dépendances
- Découplage temporel
 - ❖ Dynamique (retarder les choix)
 - ❖ Auto XXX (revoir les choix)
- Découplage sémantique
 - ❖ Médiateurs sémantiques, multi mondes
 - ❖ Médiateur de procédé (protocoles d'utilisation)

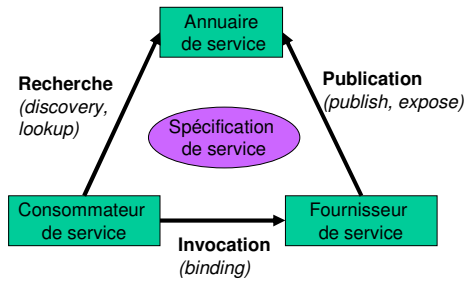
Les Services. Objectifs.

- Améliorer le découplage
 - ❖ Découplage fonctionnel
 - ❖ Réutilisation
- Améliorer la dynamique
 - ❖ Découplage temporel,
 - ❖ Adaptabilité aux conditions
- Améliorer l'évolutivité et la flexibilité
 - ❖ Découplage sémantique
- Améliorer la gestion du NF
 - ❖ Conteneurs dynamiques
 - ❖ Améliorer l'automatisation des applications

Mécanique de base des services

- Spécification = clé de voûte
 - ❖ standards
 - ❖ invariants

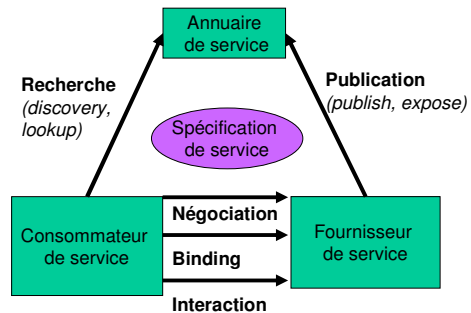
- Caractéristiques
 - ❖ Faible couplage
 - Liaison retardée
 - **Dynamisme**
 - Hétérogénéité



Mécanique de base des services

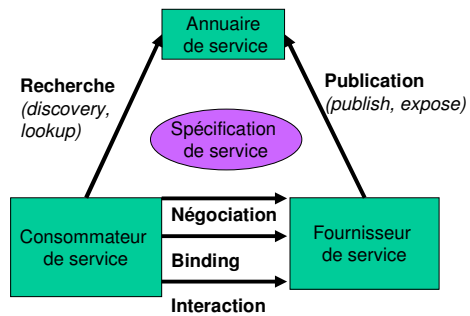
- Spécification = clé de voûte
 - ❖ standards
 - ❖ invariants

- Caractéristiques
 - ❖ Faible couplage
 - Liaison retardée
 - **Dynamisme**
 - Hétérogénéité

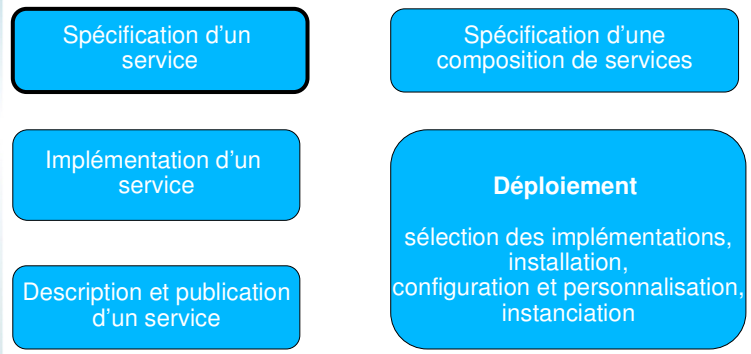


Mécanique de base des services

Publication	Statique /Dynamique	Centralisé /Repartie
Recherche	Statique /Dynamique	Centralisé /Repartie
Négociation	Statique /Dynamique	
Binding	Statique /Dynamique	Centralisé /Repartie
Interaction		Centralisé /Repartie



Activités dans la conception d'une application à services



■ Quels sont les besoins en termes d'outillage de génie logiciel pour supporter ces activités, et leurs acteurs?

Spécification d'un service

- Spécification fonctionnelle (au minimum)
 - ❖ Requête/réponse : signature des méthodes
 - ❖ Événementielle : format des événements
 - ❖ Data streaming : format des données
- Indépendante de la plateforme d'exécution ?
 - ❖ **Quel formalisme?**
- Au delà de la spécification fonctionnelle ...
 - ❖ Spécification d'un « contrat »
 - ❖ Spécification d'une « capability »
- Diverses spécifications en fonction du point de vue considéré.
 - ❖ Sécurité, performance, tolérance aux pannes, ...
 - ❖ **Quels formalismes?**
- **prescriptive ou descriptive?**

06 novembre 2007

9

Spécification d'un service

- Identifier les formalismes de spécification de services
- Abstraire les formalismes existantes, au niveau des plateformes et infrastructures à services spécifiques
- Formalisme généraliste ou formalisme spécifique au domaine ?
- Supporter la séparation des préoccupations, au niveau de la spécification d'un service
 - ❖ formalisme = langage = méta-modèle
 - ❖ préoccupation = formalisme = langage = méta-modèle
- Séparation de préoccupations => composition de préoccupations
- Abstraction des formalismes => projection vers une plateforme cible
 - ❖ Composition des préoccupations = composition des méta-modèles ?
 - ❖ Projection d'un formalisme = mapping méta-modèles ?

06 novembre 2007

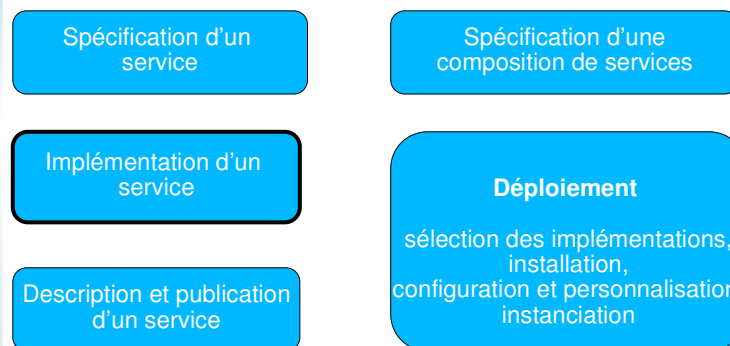
10

Description et publication d'un service

Interopérabilité des registres de services

- Peut-on vraiment spécifier et décrire un service de façon indépendante de la plateforme ?
- Besoin d'une phase explicite de description et publication
 - ❖ Dans le meilleur des cas toute l'information est déjà renseignée, et il s'agit simplement de publier cette information dans le(s) registre(s) de services de la plateforme cible
 - ❖ Dans le pire de cas, il faut ajouter des descriptions dépendantes de la plateforme

Activités dans la conception d'une application à services

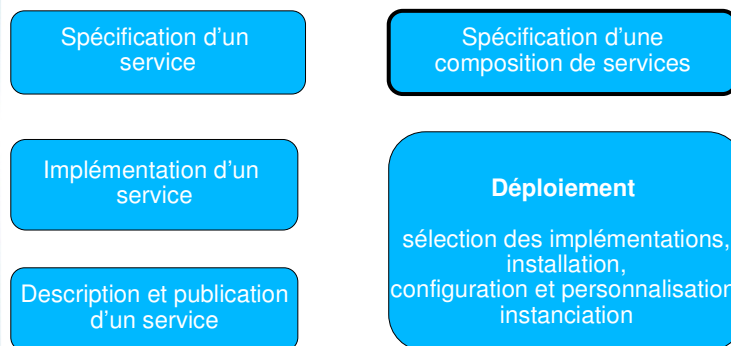


- Quels sont les besoins en termes d'outillage de génie logiciel pour supporter ces activités, et leurs acteurs?

Implémentation d'un service

- Prescriptive ou descriptive?
 - ❖ une implémentation est-elle conforme à une spécification?
 - ❖ par construction, par certification, par hasard?
- Indépendante de la plateforme d'exécution
 - ❖ solution à la iPOJO ☺
 - ❖ code métier en Java (ah, le bon vieux java ...du siècle dernier)
 - ❖ autres préoccupations
 - Expression déclarative
 - Génération de code
 - Gestion à l'exécution par un conteneur

Activités dans la conception d'une application à services



- Quels sont les besoins en termes d'outillage de génie logiciel pour supporter ces activités, et leurs acteurs?

Spécification d'une composition de services

■ La composition abstraite :

- ❖ Les services sont décrits en termes de leur « cababilities », indépendamment de leur implémentation
- ❖ Le découplage entre services et leur late binding permettent de différer la sélection d'une implémentation jusqu'au dernier moment

- ❖ Il est donc possible de spécifier une composition de services de façon abstraite

- ❖ Il est possible de concrétiser progressivement une composition de services
 - Sélection de services requis
 - Configuration et personnalisation des services choisis
 - Sélection et configuration même à l'exécution

Spécification d'une composition de services

■ Approches pour la composition de services:

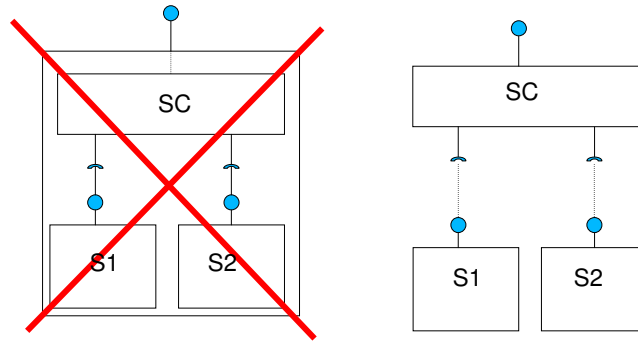
- ❖ Comment spécifier le comportement propre au composite?
- ❖ Comment spécifier les services requis?

- ❖ Une composition de services est un service?

- ❖ Sélection seulement parmi les services déployés ? ou éventuellement la sélection implique le déploiement (et instanciation) des services requis?
- ❖ Une composition de services est un mécanisme de conception ou de déploiement?

Spécification d'une composition de services

Approche « Un service n'est pas composé de services, un service requiert des services »



Spécification d'une composition de services

Approche « Un service n'est pas composé de services, un service requiert des services »

- Comment spécifier le comportement propre au composite?
 - ❖ Il n'y a pas vraiment de composites, l'implémentation d'un service peut requérir d'autres services
 - ❖ le code métier du « composite » est programmé dans le POJO
- Comment spécifier les services requis?
 - ❖ Variante 1: Utiliser l'API du registre de services
 - ❖ Variante 2: De façon déclarative (à la Service Binder)
- Une composition de services est un service?
 - ❖ L'assemblage de services n'existe pas vraiment
 - ❖ C'est qui importe est la résolution des dépendances
 - ❖ Un service peut dépendre d'autre services
 - ❖ La notion d'application est floue : un regroupement de services « mis ensemble »

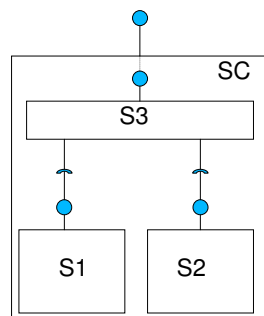
Spécification d'une composition de services

Approche « Un service n'est pas composé de services, un service requiert des services »

- Sélection seulement parmi les services déployés ? ou éventuellement la sélection implique le déploiement (et instanciation) des services requis?
 - ❖ Le déploiement et la résolution des dépendances sont orthogonaux
 - ❖ Le service reste dans l'état « dépendance non résolue »
 - ❖ Si quelqu'un (manuelle ou automatiquement ?) décide de déployer et instancier le « bon » service, la dépendance sera résolue
- Une composition de services est un mécanisme de conception ou de déploiement?
 - ❖ La composition n'est pas un mécanisme conceptuel de premier niveau (ni pour la conception, ni pour le déploiement)
 - ❖ Les dépendances entre services et leur résolution sont les mécanismes principaux

Spécification d'une composition de services

Approche « Composition structurelle orientée services »



- Il ne faut pas interpréter les boîtes comme des composants, mais comme des contraintes sur la sélection des services
- Il ne faut pas interpréter les traits entre les interfaces comme des connecteurs, mais comme des contraintes pour la résolution des dépendances
- Un service composite n'a pas de comportement propre, son comportement résulte de l'assemblage d'autres services : composite abstrait

a.k.a « et pourtant les composants c'était (presque) bien ou Service-ADL »

Spécification d'une composition de services

Approche « Composition structurelle orientée services »

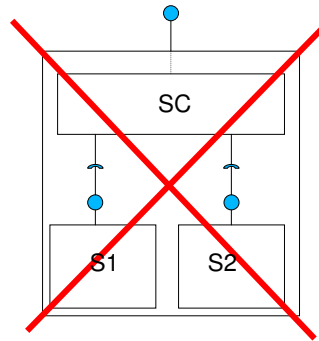
- Comment spécifier le comportement propre au composite?
 - ❖ Un service composite n'a pas de comportement propre, son comportement résulte de l'assemblage d'autres services : composite abstrait
- Comment spécifier les services requis?
 - ❖ De façon déclarative : dans l'ADL
- Une composition de services est un service?
 - ❖ Variante 1:
 - L'assemblage de services est une application
 - Une application n'est pas un service
 - Une application est composée d'autres applications
 - ❖ Variante 2 : Un assemblage de services est un service composite abstrait

Spécification d'une composition de services

Approche « Composition structurelle orientée services »

- Sélection seulement parmi les services déployés ? ou éventuellement la sélection implique le déploiement (et instanciation) des services requis?
 - ❖ Le déploiement et la résolution des dépendances sont interdépendants
 - ❖ L'assemblage de services exprime à la fois
 - Des contraintes sur la résolution des dépendances
 - Des contraintes sur la sélection des implémentations à déployer
 - Des contraintes sur l'instanciation des nouveaux services
- Une composition de services est un mécanisme de conception ou de déploiement?
 - ❖ La composition est un mécanisme de conception d'applications
 - ❖ La composition impose des contraintes sur le déploiement et la résolution des dépendances

Approche « Composition comportementale orientée services »



- Les dépendances sont explicites mais le flot de contrôle est implicite dans les composants
- Le composite n'a pas de comportement propre, et pourtant il faut spécifier les interactions entre les services requis
- Un langage d'orchestration permet de spécifier facilement le comportement du composite

Approche « Composition comportementale orientée services »

- Comment spécifier le comportement propre au composite?
 - ❖ Un service composite est spécifié comme une orchestration d'autres services
- Comment spécifier les services requis?
 - ❖ De façon déclarative
 - ❖ De préférence indépendamment du procédé métier
- Une composition de services est un service?
 - ❖ Variante à la BPEL:
 - Une orchestration de services est un service

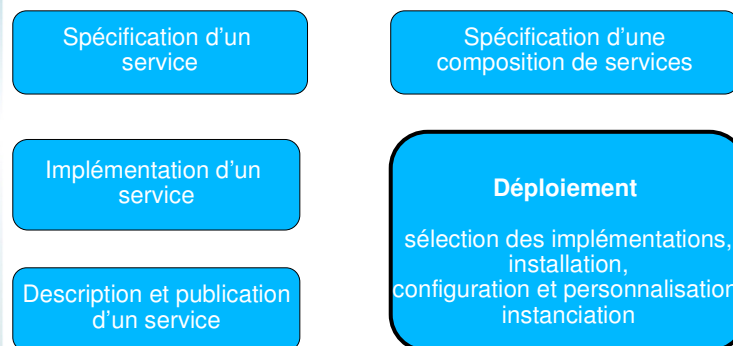
Spécification d'une composition de services

Approche « Composition comportementale orientée services »

- Sélection seulement parmi les services déployés ? ou éventuellement la sélection implique le déploiement (et instanciation) des services requis?
 - ❖ Ce n'est pas spécifié
 - ❖ Une orchestration est un service abstrait, tant que ses dépendances ne sont pas résolues, on peut utiliser la même stratégie que dans la composition structurelle

- Une composition de services est un mécanisme de conception ou de déploiement?
 - ❖ La orchestration est un mécanisme de conception d'applications
 - ❖ La orchestration n'impose pas des contraintes sur le déploiement et la résolution des dépendances

Activités dans la conception d'une application à services



- Quels sont les besoins en termes d'outillage de génie logiciel pour supporter ces activités, et leurs acteurs?



Les applications grande échelle

- Problématiques :
 - ❖ Très grand nombre de clients / passerelles / capteurs
 - ❖ Très forte évolutivité dynamique (clients, services, passerelles, ...)
 - ❖ Hypothèses de connexion (intermittence, hétérogénéité, initiative passerelle)
 - ❖ Intelligence répartie sur toute la chaîne
 - ❖ Vision globale et vision "granularité service"
- Le déploiement :
 - ❖ nouveau client, nouveau service, modification d'un contrat
- Architecture de déploiement :
 - ❖ serveur de déploiement + *repositories*
 - ❖ broker de déploiement
 - ❖ manager de déploiement sur la passerelle