MODULE WS

Présentation des services Web

1.

Présentation des architectures distribuées

Architectures ?

- Architecture centralisée : « client/serveur »
  - 1 serveur / N clients
- Architecture distribuée : « peer to peer »
  - N serveurs / N clients

Rappel sur l'architecture centralisée

- Client/serveur : distinction stricte entre le rôle de client et le rôle de serveur
  - 1 client effectue une requête pour un service donné sur un serveur et attend une réponse
  - 1 serveur reçoit une demande de service, la traite et retourne une réponse au client

Rappel sur l'architecture centralisée

- Caractéristiques du client :
  - Actif
  - Connecté à un serveur
  - Envoie des requêtes à un serveur
  - Attend et traite les réponses du serveur
  - Interagit avec un utilisateur final (par exemple avec une IHM)

Rappel sur l'architecture centralisée

- Caractéristiques du serveur :
  - Passif
  - A l'écoute des requêtes clients
  - Traite les requêtes et fournit une réponse
  - Pas d'interaction directe avec les utilisateurs finaux

Rappel sur l'architecture centralisée

- Exemples d'architecture centralisée client/serveur :
  - Consultation de pages web (envoi de requêtes HTTP depuis un navigateur à un serveur pour consulter les pages)
  - Gestion des mails (client pour envoyer et recevoir les mail, serveur pour la gestion : SMTP, POP, IMAP)

Rappel sur l'architecture centralisée

- Découpage en couches :
  - **Présentation** : affichage, dialogue avec un utilisateur final
  - **Service** : traitements, règles de gestion et logique applicative
  - **Données** : DAO, persistance des données

Rappel sur l'architecture centralisée

- Découpage en couches :
  - La **répartition** de ces couches entre les rôles de client et de serveur permet de distinguer entre les différents types d'architecture client/serveur
    - 2 tiers
    - 3 tiers
    - N tiers

Architectures distribuées

- Relations d'égal à égal :
  - Pas de connaissance globale du réseau
  - Pas de coordination globale des nœuds
  - Chaque nœud ne connait que les nœuds constituant son voisinage
  - Toutes les données sont accessibles depuis n'importe quel nœud
  - Les nœuds sont volatiles

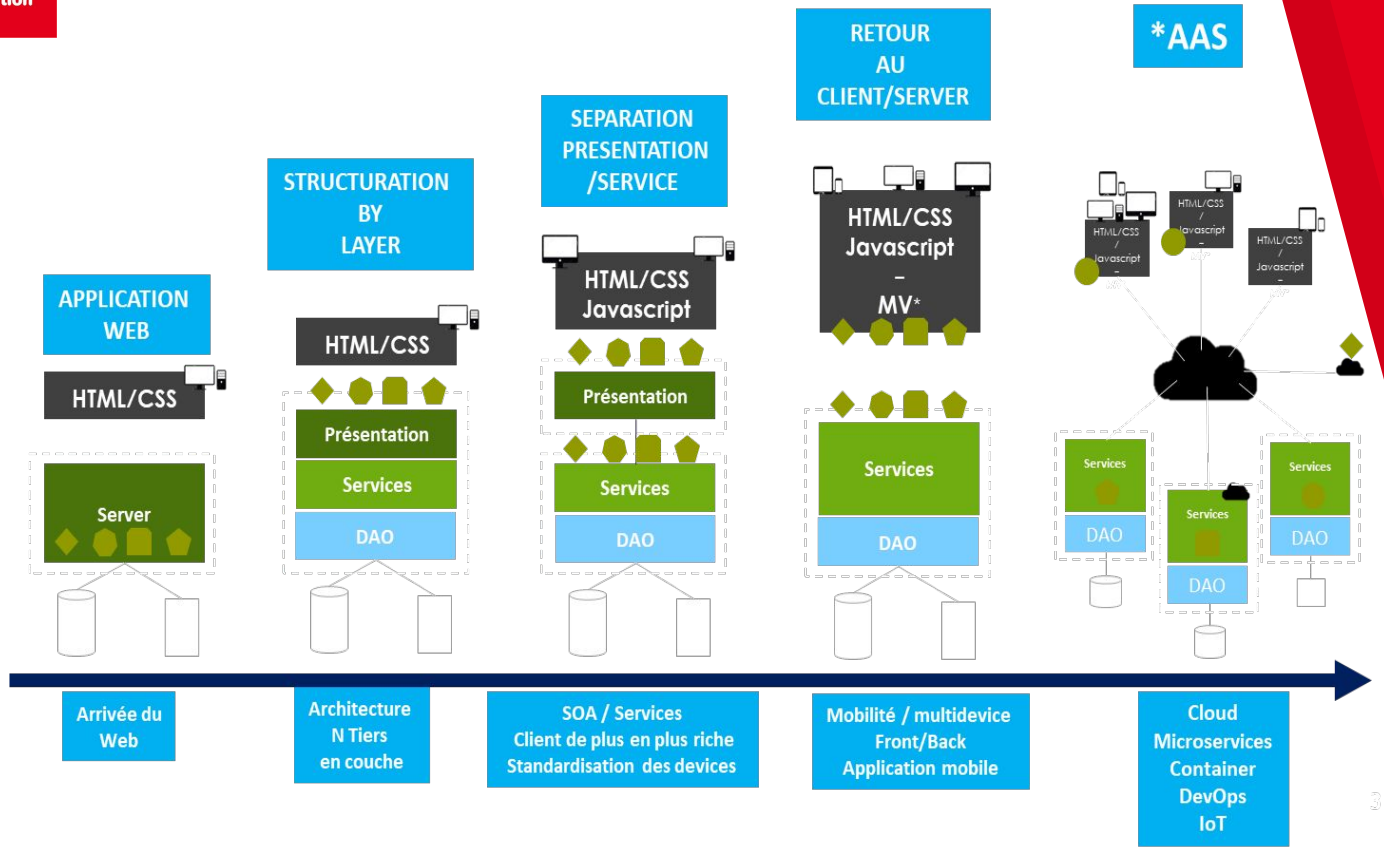
Architectures distribuées

- Avantages :
  - Plus adapté à la montée en charge (scalabilité)
  - Meilleure robustesse en cas de panne (réplication, pas se SPOF : « single point of failure »)

Architectures distribuées

- Inconvénients :
  - Problématiques spécifiques
    - Concurrence
    - Fragmentation des données
    - Gestion de la réplication
    - …

Evolution des architectures au cours du temps

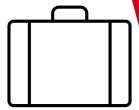


2.

Positionnement des Web services

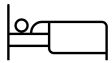
Les Services Web

- Exemple d'une agence de voyage :
  - Un produit « voyage » = une combinaison de plusieurs produits



    - Réservation des billets de transport

    - Réservation des nuits d'hôtel



    - Réservation des locations de voiture

    - …

Les Services Web

- Exemple d'une agence de voyage :
  - La construction d'un produit « voyage » est le résultat d'informations récupérées auprès de différents fournisseurs :
    - Compagnies aériennes
    - Compagnies ferroviaires
    - Loueurs de voiture
    - Chaînes hôtelières

Les Services Web

- Exemple d'une agence de voyage :
  - Une application de réservation de voyage sollicite d'autres applications réparties pour satisfaire la demande !
  - 2 types de sollicitations :
    - Transformation : adaptation du dialogue en fonction du profil utilisateur
    - Agrégation : appel à des applications proposées par des partenaires ou fournisseurs

Les Services Web

- Exemple d’une agence de voyage :
  - Pour réaliser cela, on s’appuie des Services Web !

Les Services Web

- Un Service Web, c'est quoi ?
  - Fonction distante mise à disposition sur un réseau accessibilité
  - Infrastructure souple pour des échanges entres des systèmes distribués hétérogènes
  - Localisable à partir de registres
  - Couplage faible

Les Services Web

- Un Service Web, c'est quoi ?
  - Répond à la problématique B2B (SOA -> architecture orientée service)
  - Un service résout un problème donnée
  - Combinaison possible pour résoudre des problèmes complexes

Les Services Web

- Idée générale
  - Un client a un besoin
  - Pour un besoin, plusieurs services et donc fournisseurs peuvent exister (avec ses propres caractéristiques)
  - Le client choisit un fournisseur pour pouvoir utiliser son service (celui qui correspond à son besoin et qui est compatible avec ses exigences (coût, performance, …)

Les Services Web

- Pourquoi peut-on avoir besoin de Services Web ?
  - Besoin d'interopérabilité dans des environnements applicatifs distribués
  - Echanges sur des protocoles standards (HTTP, SMTP, …)
  - Echanges entre des systèmes hétérogènes (environnements différents, langages différents)

Les Services Web

- Les usages
  - Assemblage de composants faiblement couplés
  - Définition indépendantes mais interaction
  - Adapté pour les applications orientées messages
  - Asynchronisme

Les Services Web

- Les acteurs
  - Le client : celui qui invoque le service web
  - Le fournisseur : celui qui fournit et met à disposition le service web
  - L'annuaire : celui qui détient et partage les informations sur les services web

Les Services Web

- Les acteurs
  - Le fournisseur :
    - Serveur d'application (par exemple JEE)
    - Expose un ou plusieurs services (EJB, servlets, enveloppés d'une couche « service »)

Les Services Web

- Les acteurs
  - L'annuaire :
    - Déclaration dans un annuaire = publication

3.

Approches SOAP et REST

Les types de service web

- Services web de type **SOAP**

- Services web de type **REST**

Les types de service web

- Services web de type **SOAP**
  - SOAP est un protocole
  - SOAP = Simple Object Access Protocol
  - Initialement conçu pour que des applications développées avec différents langages sur différentes plateformes puissent communiquer

Les types de service web

- Services web de type **SOAP**
  - Protocole = règles imposées qui augmentent la complexité et les coûts
  - Mais, standards qui assurent la conformité et sont privilégiés pour certaines applications en entreprise

Les types de service web

- Services web de type **SOAP**
  - Les standards de conformité intégrés incluent la sécurité, l'atomicité, la cohérence, l'isolement et la durabilité (ACID), et un ensemble de propriétés qui permet d'assurer des transactions de base de données fiables

Les types de service web

- Services web de type **SOAP**
  - Les principales spécifications :
    - WS-Security : standardise la manière dont les messages sont sécurisés et transférés via des identifiants uniques appelés jetons
    - WS-ReliableMessaging : standardise la gestion des erreurs entre les messages transférés par le biais d'une infrastructure informatique non fiable

Les types de service web

- Services web de type **SOAP**
  - Les principales spécifications :
    - WS-Adressing : ajoute les informations de routage des paquets en tant que métadonnées dans des en-têtes SOAP, au lieu de les conserver plus en profondeur dans le réseau
    - WSDL (Web Services Description Language) : décrit la fonction d'un service web ainsi que ses limites

Les types de service web

- Services web de type **SOAP**
  - Lorsqu'une requête de données est envoyée à une API SOAP, elle peut être gérée par n'importe quel protocole de couches de l'application : HTTP, SMTP, TCP, ...

Les types de service web

- Services web de type **SOAP**
  - Les messages SOAP doivent être envoyés sous la forme d'un document XML
  - Une fois finalisée, une requête destinée à une API SOAP ne peut pas être mise en cache par un navigateur (pas possible d'y accéder plus tard sans la renvoyer vers l'API)

Les types de service web

- Services web de type **REST**
  - REST **n'est pas un protocole**
  - REST est un ensemble de principes architecturaux adapté aux besoins des services web et applications mobiles légers
  - La mise en place de ces recommandations est laissée à l'appréciation des développeurs

Les types de service web

- Services web de type **REST**
  - L'envoi d'une requête à une API REST se fait généralement par le protocole HTTP
  - À la réception de la requête, les API développées selon les principes REST (appelées API ou services web RESTful) peuvent renvoyer des messages dans différents formats : HTML, XML, texte brut, JSON

Les types de service web

- Services web de type **REST**
  - Le format JSON (JavaScript Object Notation) est le plus utilisé pour les messages : léger, lisible par tous les langages de programmation et les humains
  - Les API RESTful sont plus flexibles et plus faciles à mettre en place

4.

Liens avec la SOA

Architecture orientée Service (SOA)

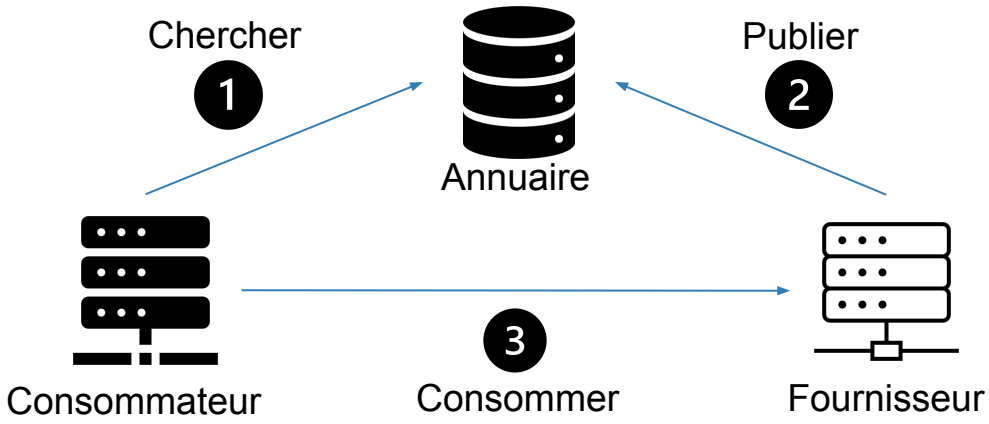
- SOA : Modèle d'interaction applicative qui met en œuvre des services
  - Un client demande un service à un fournisseur

Architecture orientée Service (SOA)

- Objectifs :
  - Réutilisabilité
  - Interopérabilité
  - Réduction du couplage entre les systèmes

Architecture orientée Service (SOA)

- Paradigme SOA
  - **Chercher, publier, consommer**



Architecture orientée Service (SOA)

- Service : composant logiciel distribué exposant des fonctionnalités à forte valeur ajoutée d'un domaine métier
  - Contrat standardisé
  - Couplage lâche
  - Abstraction
  - Réutilisabilité

Architecture orientée Service (SOA)

- Service : composant logiciel distribué exposant des fonctionnalités à forte valeur ajoutée d'un domaine métier
  - Découvrable
  - Autonomie
  - Sans état
  - Composable

Architecture orientée Service (SOA)

- SOA peut s’appuyer sur les web services ! (WSOA)