

Architecture Orientée service SOA

Enseignante: Bènène Fradi



Objectifs du module

- Sensibiliser l'apprenant des défis de l'interopérabilité
- Maîtriser les concepts liés de services web et technologies liées
- Construire et déployer des services web et leurs clients
- Familiariser l'apprenant avec le style d'architecture SOA
- Maîtriser un outil de mise en œuvre SOA

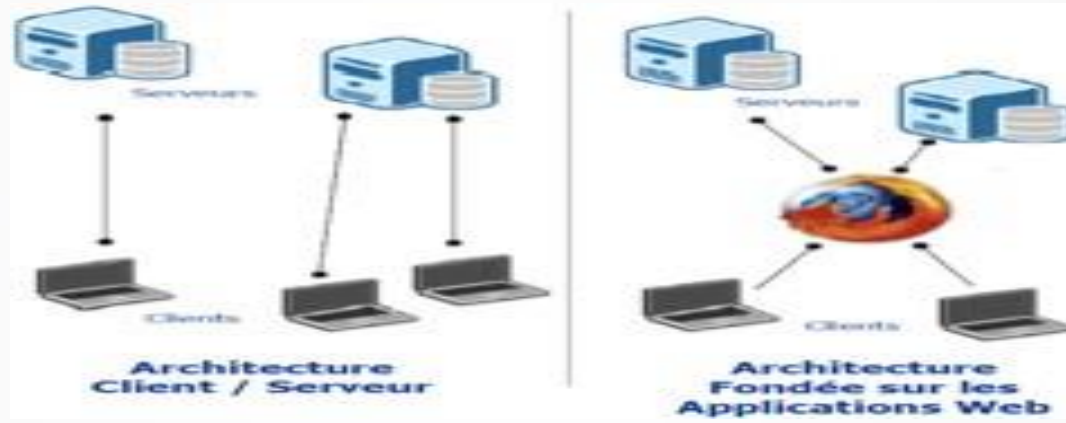
Chapitre 3



Introduction aux architecture SOA

► Architecture distribuée

- Architecture où le traitement des données des applications est distribué sur plusieurs machines en réseau
 - **Exemples** : Architectures client-serveur, N-Tiers, Web



- Limites dues à leurs technologies de base :
 - Utilisation de composants provenant d'un **même constructeur**,
 - Utilisation d'un **langage de programmation spécifique**,
 - **Complexité des technologies** utilisées
 - Incapacité de répondre au besoin d'**interopérabilité**
 - Evolution des AD **vers SOA**



SOA et Interopérabilité



- L'**interopérabilité** exprime le besoin d'échanger des données entre 2 systèmes distribués et **éventuellement hétérogènes** pour un besoin d'**intégration**
- **SOA**, proposée par Gartner Group (2003-2008), se prête comme solution au problème d'interopérabilité, en répondant aux questions :
 - Quel **transport de communication** (TCP/IP ? quelle version?) ?
 - Quel **protocole applicatif** (HTTP, FTP, SFTP...) ?
 - Quel est le **format de données** à adopter ? (HTML,XML,,,) ?
 - ...

Enjeux de la SOA



- Elle est devenue une solution incontournable pour gagner d'autres enjeux :
 - **Métier** : Produire des **SI ouverts** (à la fois interopérables + évolutifs)
 - Proposer une solution utilisant des **standards** et assurant un **couplage faible**
 - **Métier** : Réduire le **Time to Market** (le temps entre le besoin et la mise en production)
 - Proposer des moyens pour réduire le temps du cycle projet
 - **Technique** : **Fédérer les technologies**
 - **Financier** : Maîtriser les **coûts** et les **délais**

► Architecture SOA



- Architecture orientée service (Service Oriented Architecture)
- Style d'**architecture distribuée** qui permet de fournir ou consommer un processus métier en tant que **service**
- Offre des services **réutilisables** et **interopérables** via des interfaces standards (construites autour de XML)
- Plusieurs partenaires peuvent communiquer et échanger des données dans le contexte de SOA **indépendamment des Plateformes et langages**

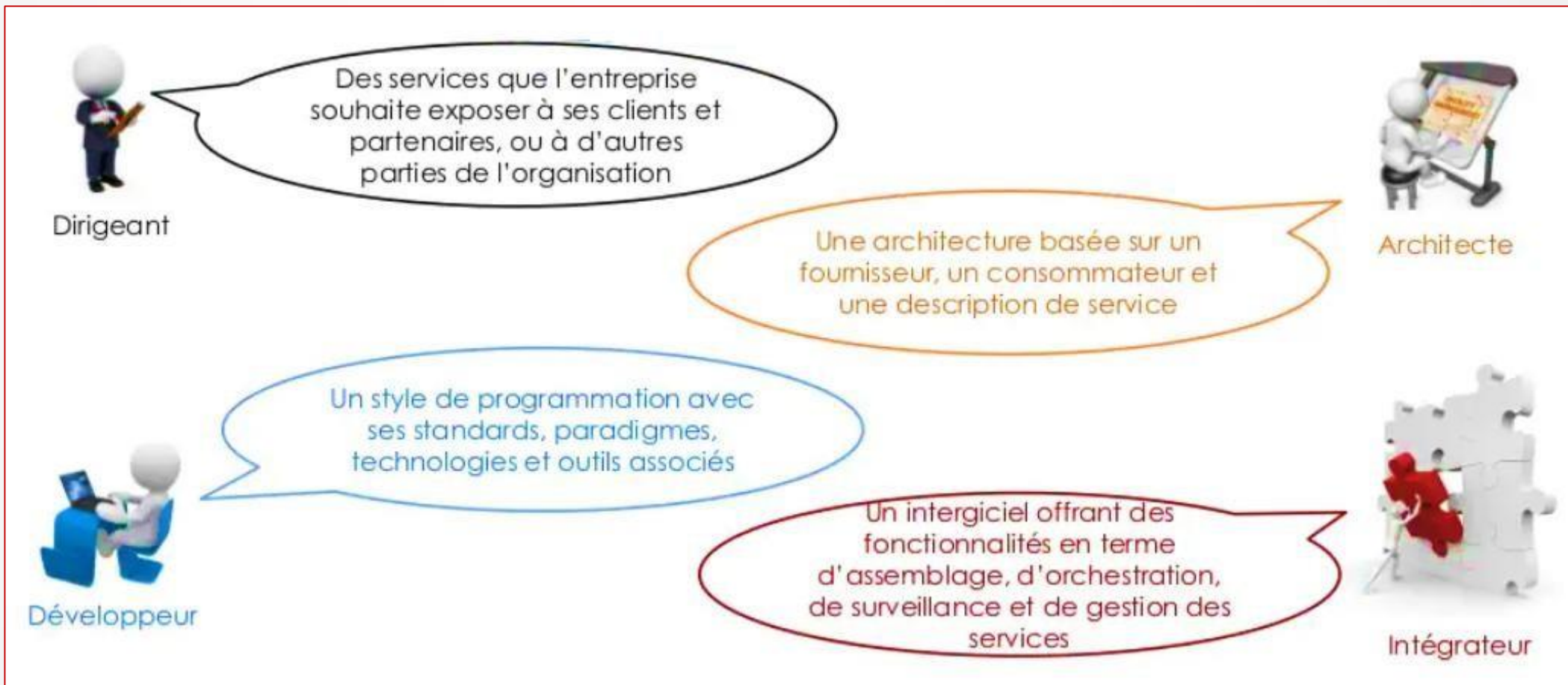




Technologies d'implémentation de SOA

- L'architecture SOA peut être implémentée par différentes Technologies :
 - **CORBA**,
 - **APIs : RMI ou JMS** (combinées aux Interfaces Java),
 - **Services Web** (protocole SOAP)
- Les **Services Web** demeurent la technologie émergente pour l'implémentation de l'architecture SOA puisqu'ils sont :
 - multiplateformes,
 - multilangages,
 - relativement faciles à implémenter
 - Disposent de standards : WSDL, BPEL...

Besoins...



Définition SOA



1ère définition : SOA est un paradigme pour l'organisation et l'utilisation de services distribués. Elle permet d'offrir des services, découvrables, invocables et composables.

2ème définition : SOA est un ensemble de composants qui peuvent être invoqués, et dont les descriptions d'interface peuvent être publiées et découvertes.

3ème définition : SOA est un style d'architecture qui permet la réorganisation et l'encapsulation des fonctionnalités d'un système d'information en un ensemble de services faiblement couplés appartenant à la fois au niveau métier et au niveau technique de l'entreprise. Les services, munis d'un contrat d'utilisation et d'une interface de description, seront publiés dans des registres de services afin qu'ils puissent être invoqués par des clients distants.

Définition SOA



- “ L’architecture orientée service constitue un **style d’architecture** basée sur le principe de séparation de l’activité métier en une série **de services**”.
- “ Ces services peuvent **être assemblés et liés** entre eux selon le principe de couplage lâche pour exécuter l’application désirée. ”

Gartner - Septembre 2005

- Objectifs: Décomposer une fonctionnalité en un ensemble de fonctions basiques(services) fournies par des composants.
- Décrire finement le schéma d’interaction entre ces services.

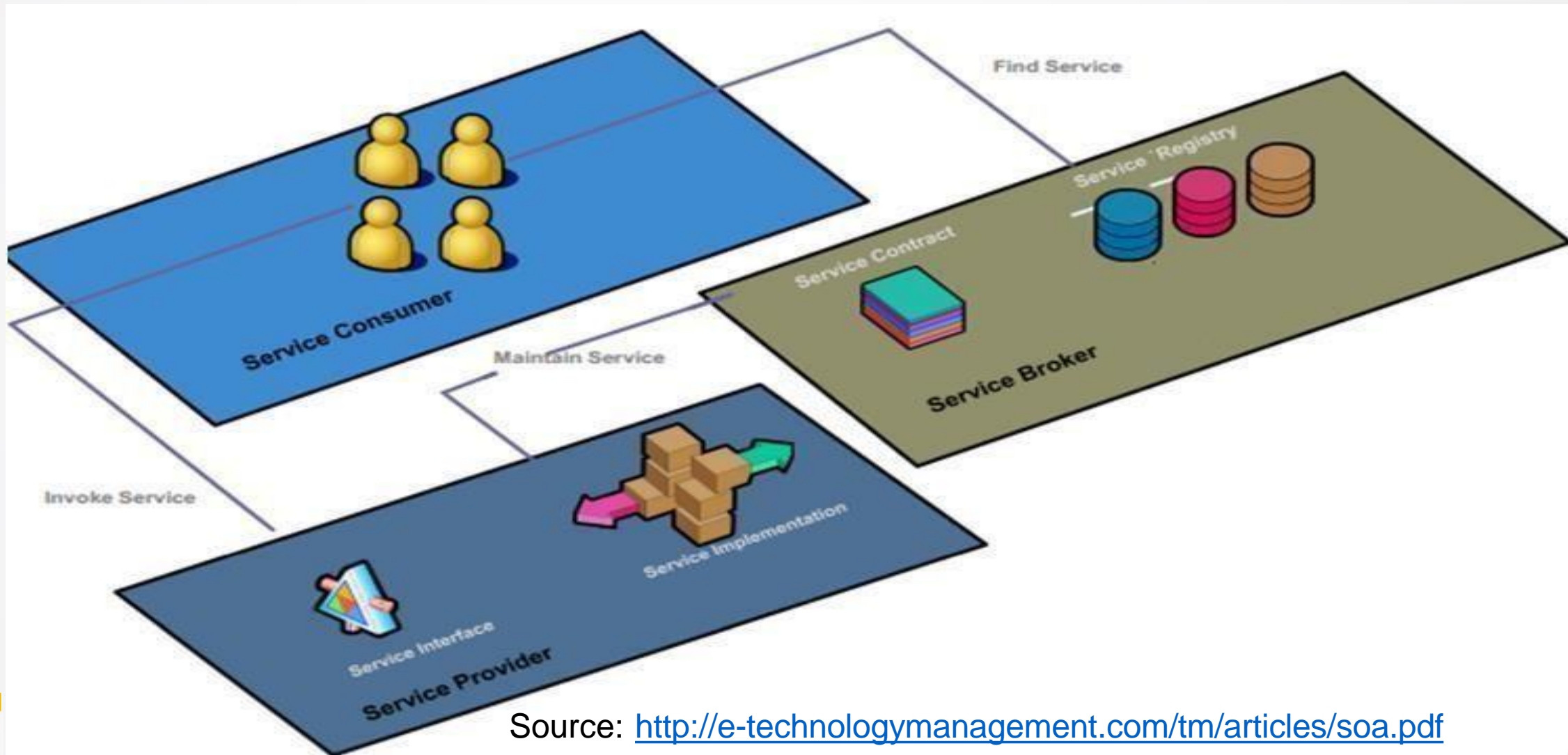


Éléments SOA



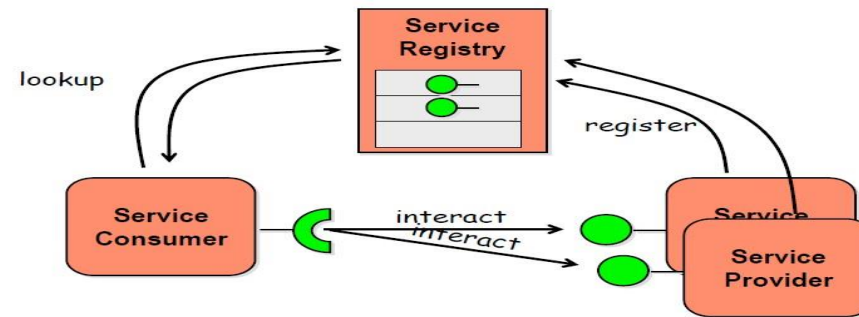
- **Le fournisseur de service** crée le service Web, puis publie son interface ainsi que les informations d'accès au service, dans un annuaire de services Web.
- **L' annuaire de service** rend disponible l'interface du service ainsi que ses informations d'accès, pour n'importe quel demandeur potentiel de service(peut être public ou privé).
- **Le consommateur de service** accède à l'annuaire de service pour effectuer une recherche afin de trouver les services désirés. Ensuite, il se lie au fournisseur pour invoquer le service.

▶ Eléments SOA



Source: <http://e-technologymanagement.com/tm/articles/soa.pdf>

Paradigme SOA



■ Fournisseur de service :

- **Fournit** un service accessible via une adresse
- **publie** son contrat dans le registre de services
- et **exécute** les requêtes des consommateurs (un **Proxy** et un cache peuvent être utilisés du côté consommateur pour délocaliser le traitement et réduire le nombre d'appels réseau)

■ Consommateur de service : application, service...

- **Cherche** le service dans le registre (son adresse)
- Se **lie dynamiquement** au service (binding)
- **Invoke** le service via une requête conforme au contrat

■ Registre de services : Annuaire des contrats de services

- Le **Contrat** décrit le **format d'échange** (format des requête/réponse, les pré et post conditions du service et sa QoS, ex: temps de réponse)

Le contrat est renouvelable par demande de nouveau bail à partir du registre

Interaction

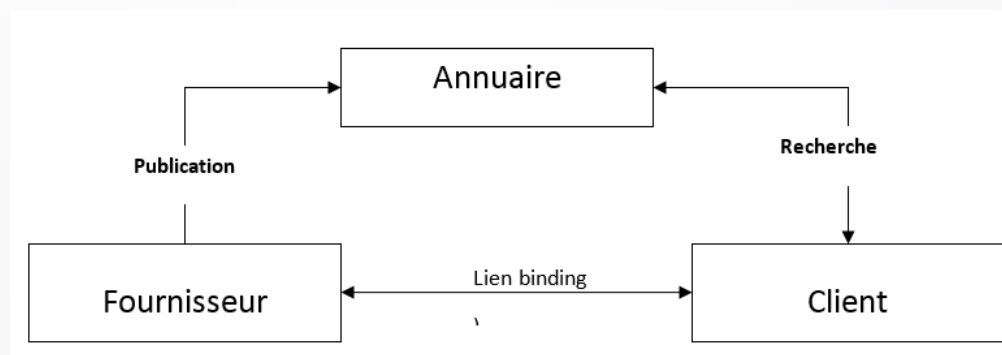


Les interactions de base entre ces trois composants incluent les opérations suivantes :

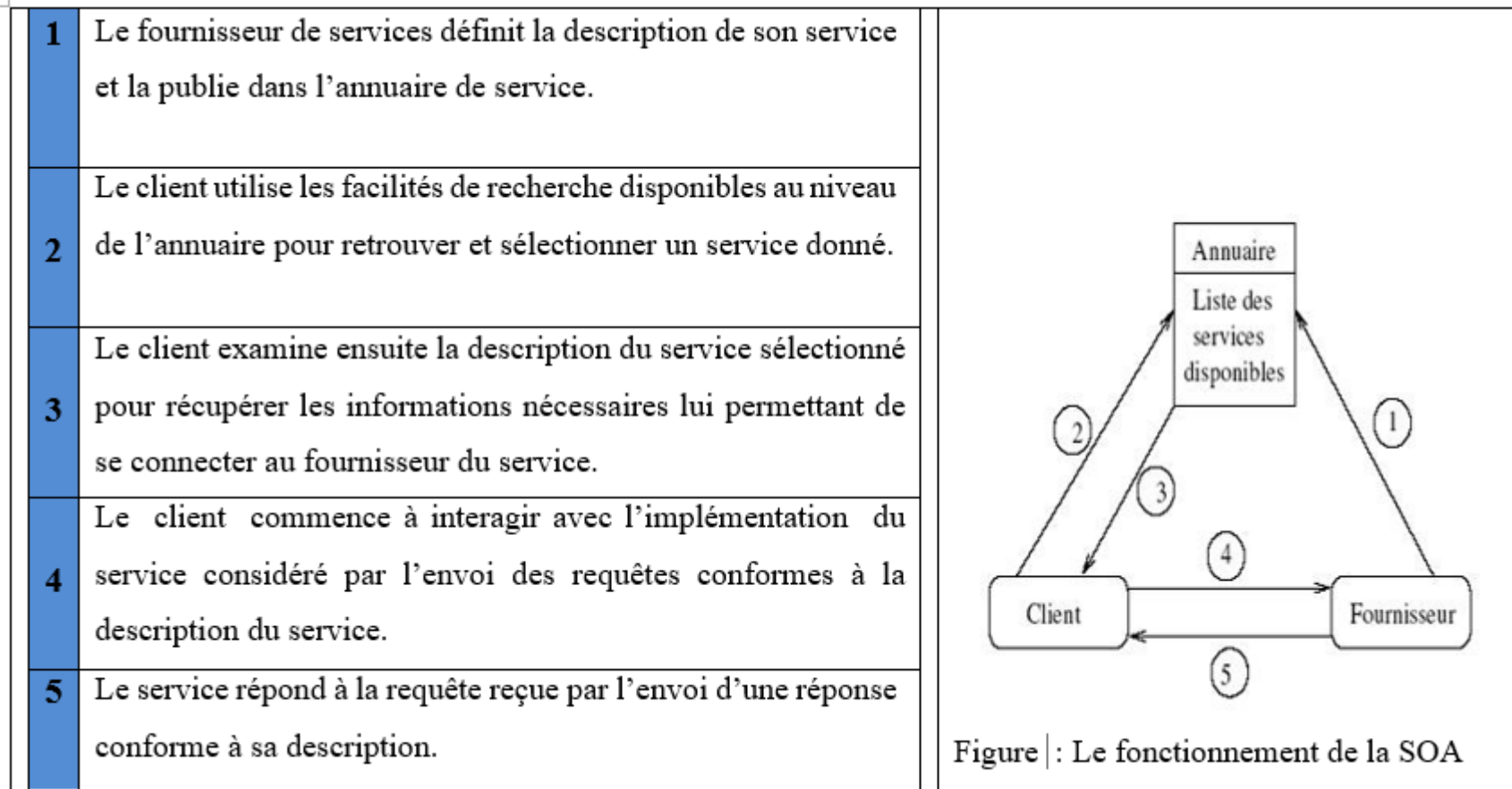
Publication : opération réalisée par le fournisseur de service, qui consiste à enregistrer le service dans l'annuaire pour le rendre accessible aux clients.

Recherche : opération réalisée par le client, elle consiste à rechercher un service dans l'annuaire.

liens (binding) d'opérations : réponse de l'annuaire à une requête de recherche émise par un client, elle consiste à trouver le service répondant à la requête du client.



► Fonctionnement SOA



Cycle de vie des services dans la SOA



Le cycle de vie des services est défini en quatre phases :

Construction : la phase de construction inclut le développement et le test de l'implémentation du service, la définition de la description de l'interface et la définition de la description d'implémentation.

Il y a trois possibilités pour fournir une implémentation d'un service par : (i) la création de nouveaux services, (ii) la transformation des applications existantes en services, (iii) la composition de nouveaux services et applications existantes,

la composition de nouveaux services à partir des services existants implique l'ordonnancement et l'orchestration de flux des messages entre les programmes directement ou bien avec la technique de Workflow ,

Déploiement : la phase de déploiement inclut la publication de la description du service dans un service d'enregistrement

Exécution : dans cette phase le consommateur du service peut trouver la description du service et invoquer toutes les opérations définies dans le service.

Gestion : cette phase couvre la gestion et l'administration de l'application du service : sécurité, disponibilité,...etc.



Apports de SOA



- Améliorer la rapidité ainsi que la productivité des développements.
- Une réutilisabilité possible des services.
- De meilleures possibilités d'évolution.
- Une maintenance facilitée.
- Couplage faible entre les services.
- Architecture basée sur des standards ouverts.
- L'indépendance par rapport aux aspects technologiques.
- Une modularité permettant de remplacer facilement un service par un autre.

► Architecture SOA ou microservices ?



- L'architecture SOA est indépendante de la technologie
 - Permet une interopérabilité transparente entre les différents services.
 - Un service web est une application **monolithique** :
 - développée en un seul bloc;
 - déployée d'une manière unitaire dans un serveur d'application.
- L'architecture micro-services est un modèle d'architecture lié à une application,
 - Se compose d'un ensemble de petits services **indépendants** (langages, architecture, etc).
 - Chaque micro-service:
 - est gérée par une équipe;
 - est séparément déployé;
 - s'exécute dans un processus qui lui est propre.



SOA et Services Web



- Les services Web sont une **technologie d'implémentation de la SOA**
- Les services Web constituent la meilleure solution disponible (**standardisée**)
 - Sont définis par des standards : OASIS, W3C, IETF et WS-I



Service Web



- **Selon W3C :**

Application logicielle identifiée par un URI, dont les interfaces et le binding peuvent être définis, décrits et découverts par des artifacts XML, et supportent des interactions directs avec d'autres applications logicielles utilisant des messages à base de XML via des protocoles à base d'Internet

- **Composant logiciel :**

- **modulaire** implémenté dans **n'importe quel langage**
- déployé sur **n'importe quelle plateforme**
- enveloppé dans une couche de **standards dérivés du XML**
- ne nécessitant pas une configuration réseau particulière
- **publié, localisé et invoqué** de n'importe quel point du Web
- **accessible** via des **protocoles standards** Internet

► Définition d'un Service web



- Un service Web est:
 - une « unité logique applicative » accessible en utilisant les protocoles standard d'Internet.
 - Un API fournissant des données et des services à d'autres applications.
 - Il s'appuie sur un ensemble de standards (HTTP, WSDL, SOAP et UDDI)
- Les SW interagissent à travers l'échanges de messages

► Les web services

- Les Web Services sont des composants web basés sur Internet (HTTP) qui exécutent des tâches précises et qui respectent un format spécifique (XML).
- Ils permettent aux applications de faire appel à des fonctionnalités à distance en simplifiant ainsi l'échange de données.



Schéma d'appel d'un service web

- Les Web Services permettent aux applications de dialoguer à travers le réseau, indépendamment de
 - leur plate-forme d'exécution
 - et de leur langage d'implémentation.
- Ils s'inscrivent dans la continuité d'initiatives telles que
 - CORBA (*Common Object Request Broker Architecture*, de l'OMG) en apportant toutefois une réponse plus simple, s'appuyant sur des technologies et standards reconnus et maintenant acceptés de tous.



Rappel Services web

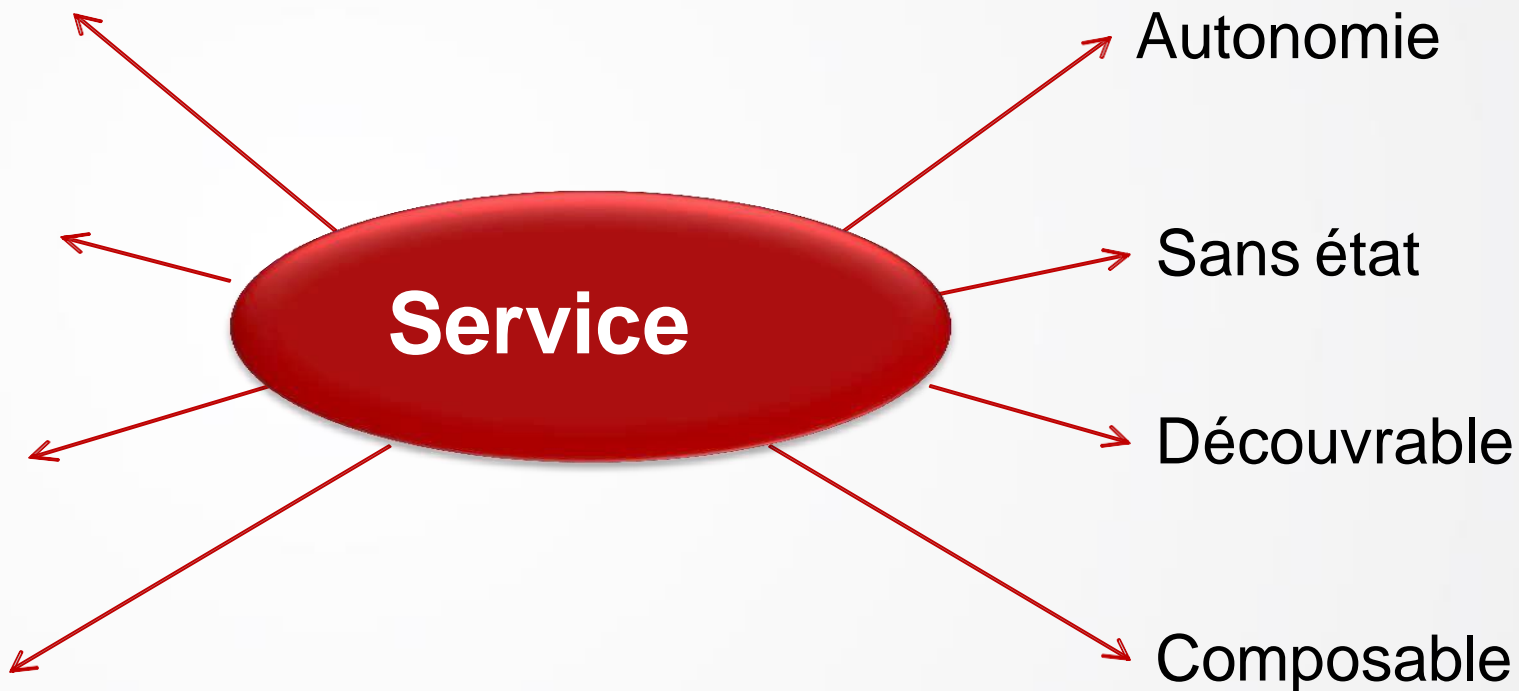


Contrat standardisé

Couplage lâche

Abstraction

Réutilisabilité





Utilité..



- Faire interagir des composants hétérogènes, distants, et indépendants avec un protocole standard (ex: SOAP)
- Plus concrètement:
 - ✓ Permettre à une application de trouver automatiquement via le web le service dont elle a besoin et d'échanger des données avec lui.
- Les Web Services ont été conçus pour intégrer la dimension d'Internet, et la standardisation des échanges.
- Ils permettent d'interconnecter : Différentes entreprises, Différents matériels,
Différentes applications et Différents clients.



Types de services Web



- **Etendus** : utilisent les standards **UDDI** (annuaire)/ **WSDL** (contrat) / **SOAP** (consommation)
- **REST** (Representational State Transfer) utilisent :
 - directement **HTTP** au lieu d'une enveloppe SOAP
 - un **URI** pour nommer et identifier une ressource
 - les méthodes HTTP (**POST**, **GET**, **PUT** et **DELETE**) pour effectuer les opérations de base CRUD
- **Mais** :
 - REST n'est pas un remplacement générique de SOAP (ne couvre qu'une partie des besoins)
 - REST utilise WADL (Web Application Description Language) pour décrire les Contrats, et qui est non standardisé : initiative isolée de SUN

► Les composants d'un service



- 1 L'interface :** L'interface de service permet de définir les modalités d'accès au service (nom, les données d'entrée et de sortie des opérations publiques). Ceci pour encapsuler l'implémentation et l'exécution physique de service. La fonctionnalité du service est exposée par l'interface aux clients.
- 2 Le contrat :** Le contrat de service joue un rôle majeur : il détaille les conditions d'utilisation du service sous forme de pré et post conditions, protocoles, et contraintes.
Un contrat de service est un document organisé en plusieurs parties qui sont :
 - L'identification des parties
 - La description des fonctions du service.
 - La description de l'interface du service.
 - La description de la qualité du service.
 - La description du cycle de vie du service et du contrat.
 - La description des termes de l'échange (s'il y'en a).
- 3 La logique d'affaire :** se compose des règles de service, les exécutions et les résultats d'un service dans la réalité. Cette partie est présentée particulièrement dans les services métiers.
- 4 Les données :** Un service peut également inclure des données.
- 5 L'implémentation :** L'implémentation de service fournit physiquement la logique d'affaire exigée et les données appropriées. C'est la réalisation technique qui accomplit le contrat de service.

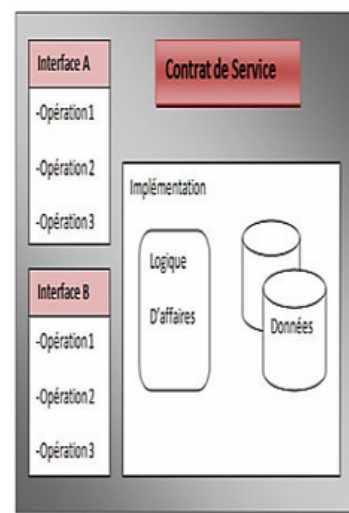
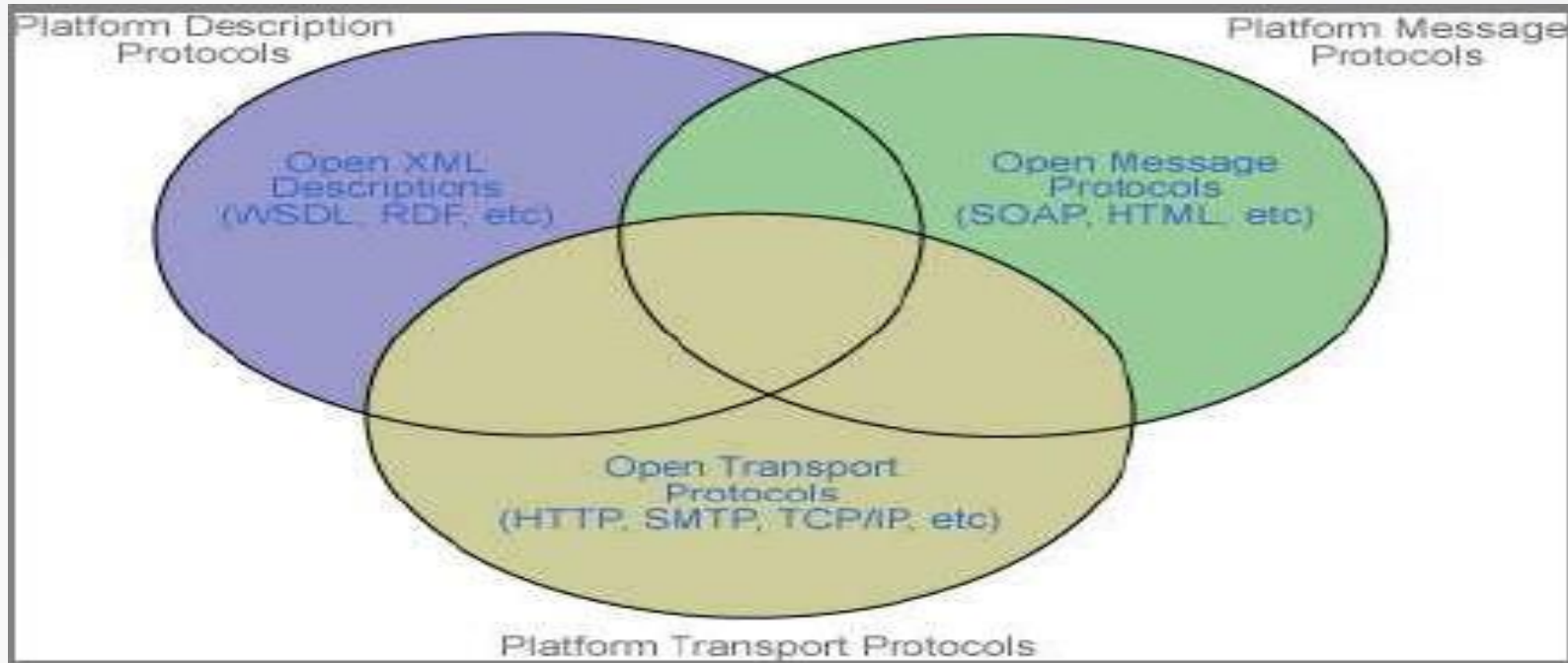


Figure 3 : Composants de service.

Protocoles des services Web



- Protocole de **description** des services (méthodes, signatures...) : **WSDL...**
- Protocole pour l'écriture des **messages** : **SOAP...**
- Protocole de **transport** pour faire circuler les données sur Internet (**http**, **SMTP**, **FTP**, **TCP/IP**, **POP3**, **NNTP...**)



Les composants de l'architecture de référence



L'architecture des services Web est une architecture qui définit les éléments globaux qui assurent l'interopérabilité des services Web.

L'architecture de référence comporte trois composants de base qui sont:

- **Le fournisseur de service** : c'est le propriétaire du service Web. Il représente l'environnement d'hébergement et d'exécution du service. Il est constitué de trois couches de bases:
 - La couche de données** : contient une ou plusieurs bases de données
 - La couche applicative** : c'est la plateforme de développement qui assure l'exécution du service Web
 - La couche de description** : elle expose les fonctionnalités du service via un fichier WSDL

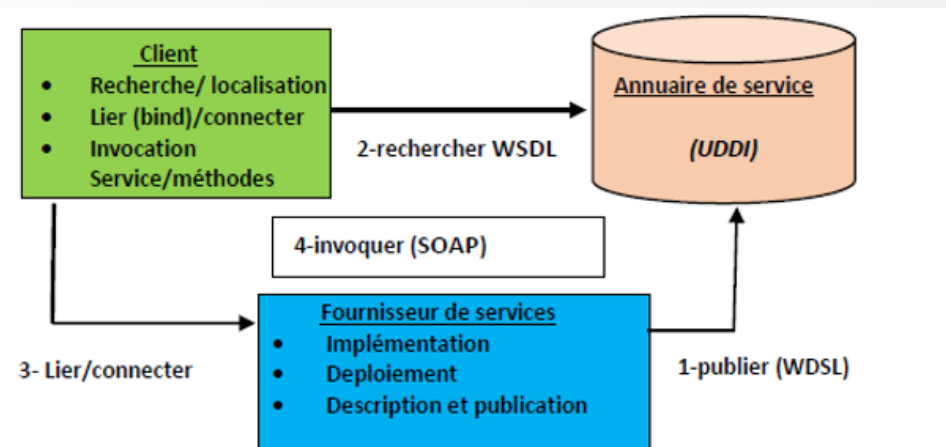


Figure : Architecture de référence des services Web.

- **Le client** : Le client est défini comme le consommateur du service. Il peut accéder à ce dernier en échangeant avec le fournisseur des messages SOAP. Techniquement, le client peut être une simple application Windows ou Web, comme il peut être un autre service Web.
- **L'annuaire des services** : L'annuaire des services est un registre de description qui offre aux fournisseurs le moyen de publier et d'indexer leurs services Web sur le réseau, il permet, en outre, aux clients de rechercher les services publiés.

▶ Etapes d'exécution des Services Web dans une architecture de référence

Les principales étapes d'exécution des services Web dans l'architecture de référence sont :

Etape 1 : définition et description du service Web

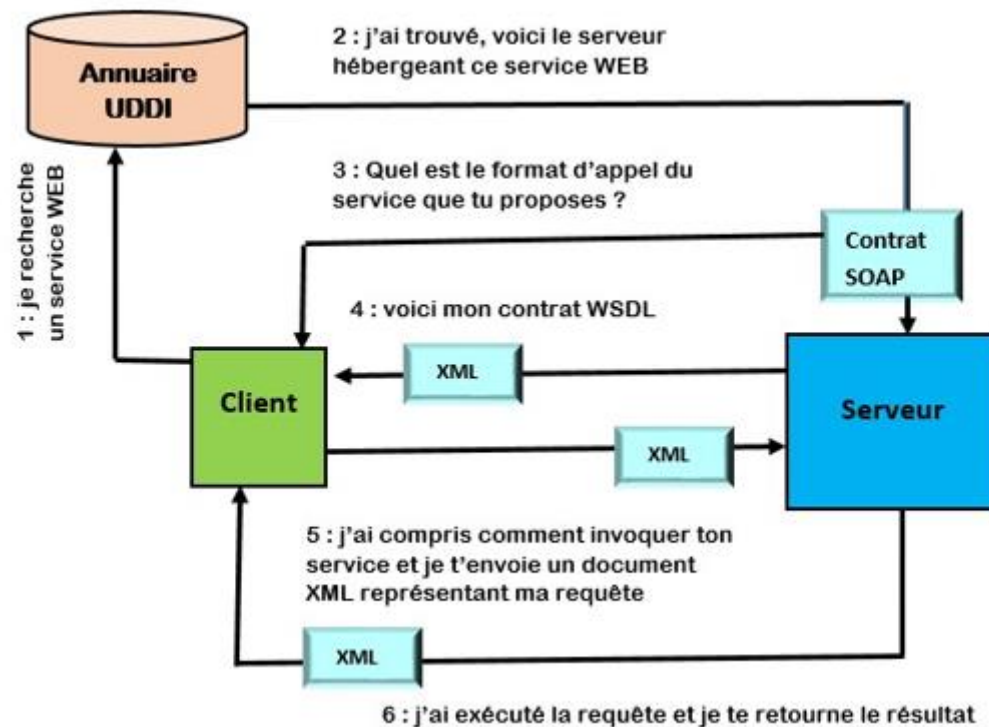
La définition est faite en WSDL au sein du fournisseur de services Web.

Etape 2 : publication du service Web

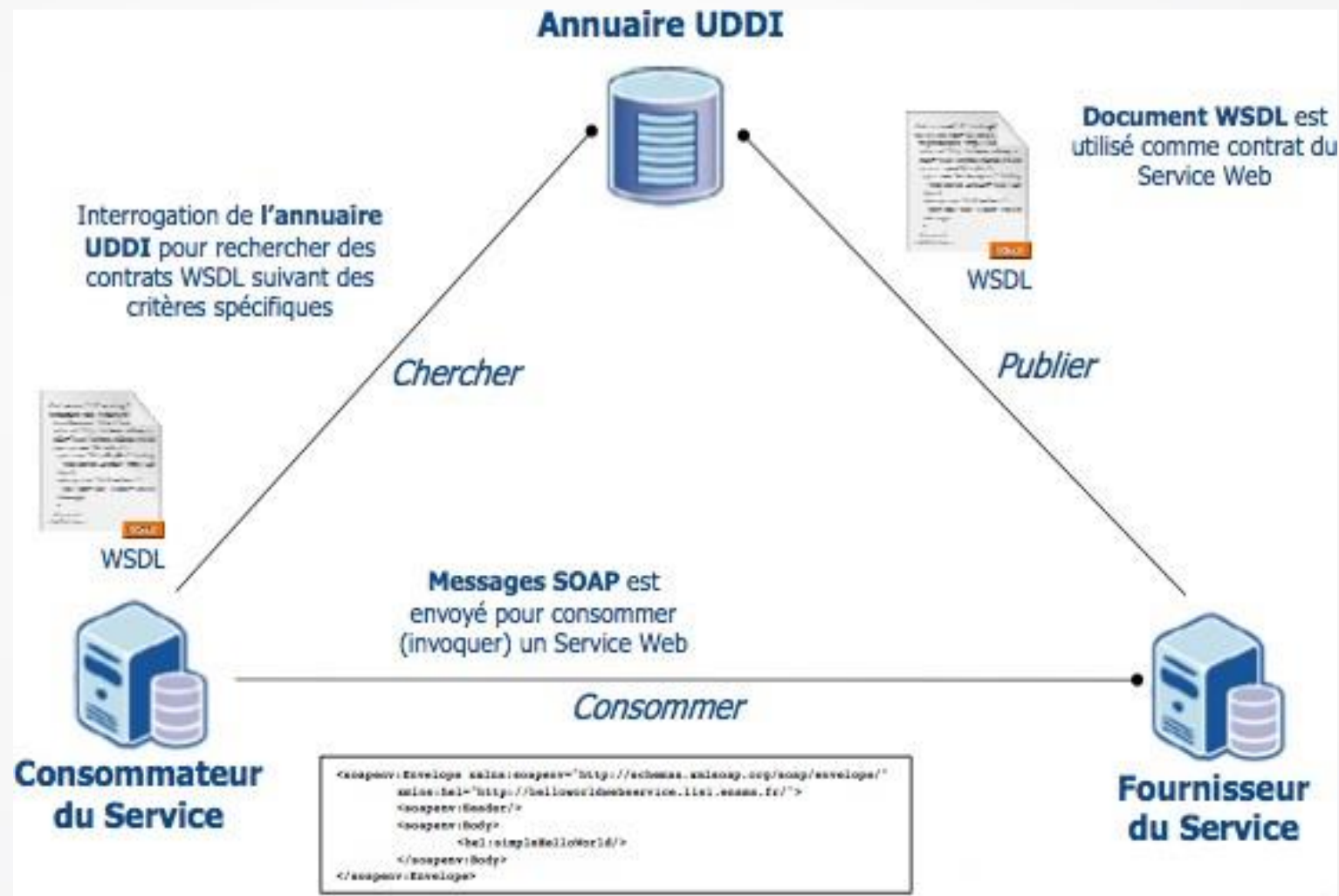
publier le service Web dans un annuaire dédié UDDI afin de le rendre accessible aux clients.

Etape 3 : recherche du service Web

le client se connecte, sur un annuaire UDDI pour effectuer une recherche de service Web .



Services Web étendus





Etapes d'exécution des Services Web dans une architecture de référence



Les principales étapes d'exécution des services Web dans l'architecture de référence sont :

Etape 4: récupération des informations de description du service et enregistrement du client

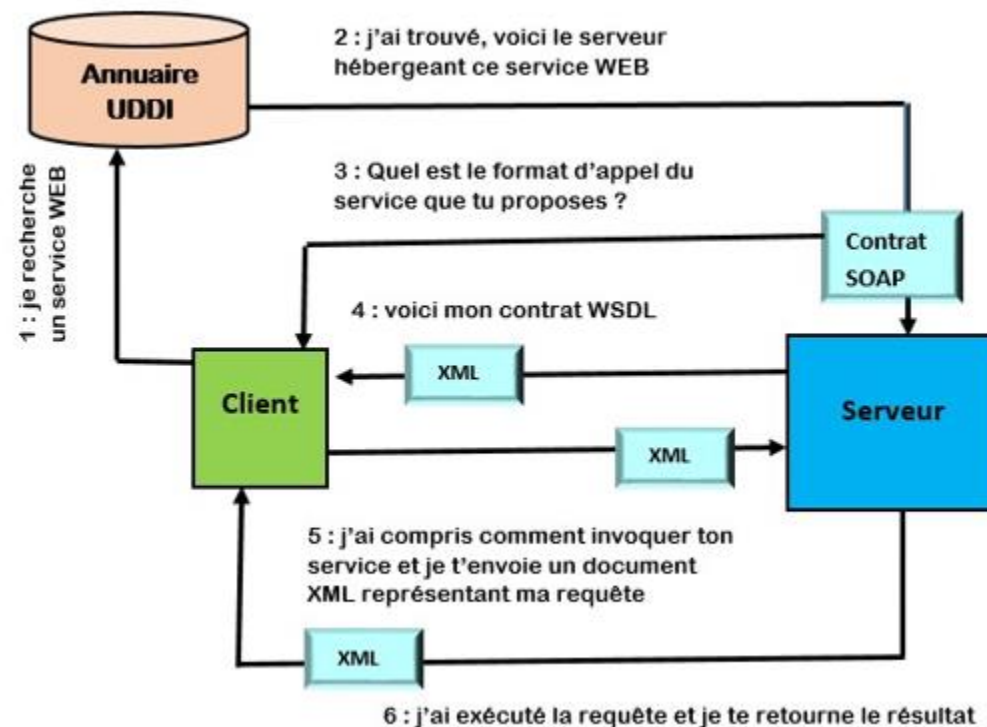
le service Web trouvé par le client, récupère de l'annuaire UDDI la description du service au format WSDL puis s'enregistre auprès du fournisseur associé au service Web.

Etape 5 : connexion au service Web

permet d'assurer la communication entre le composant demandeur du service et le fournisseur de service à travers des Wrappers (listener et proxy). Pour cela, le proxy du composant demandeur émet une requête SOAP au composant fournisseur du service. Le protocole HTTP véhicule le message SOAP jusqu'au listener du fournisseur du service.

Etape 6 : réponse du service Web

Dans cette étape le service Web du fournisseur renvoie sa réponse au demandeur sous





Etapes d'exécution des Services Web dans une architecture de référence



Etape 4: récupération des informations de description du service et enregistrement du client

le service Web trouvé par le client, récupère de l'annuaire UDDI la description du service au format WSDL puis s'enregistre auprès du fournisseur associé au service Web.

Etape 5 : connexion au service Web

permet d'assurer la communication entre le composant demandeur du service et le fournisseur de service à travers des Wrappers (listener et proxy). Pour cela, le proxy du composant demandeur émet une requête SOAP au composant fournisseur du service. Le protocole HTTP véhicule le message SOAP jusqu'au listener du fournisseur du service.

Etape 6 : réponse du service Web

Dans cette étape le service Web du fournisseur renvoie sa réponse au demandeur sous la forme d'un document XML via SOAP et HTTP.



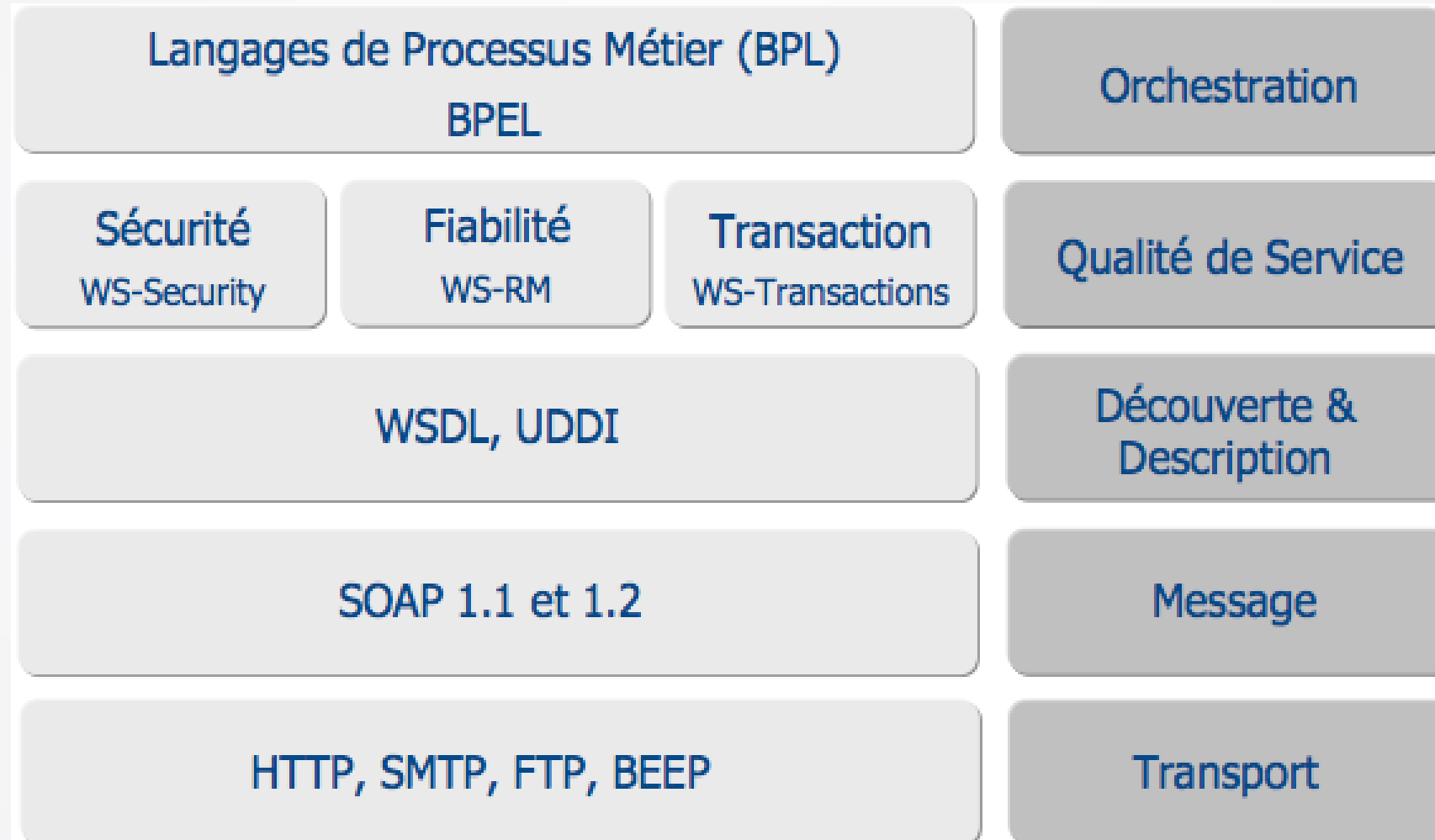
Concepts des web services



- Le concept des Web Services s'articule actuellement autour des trois concepts suivants :
 - SOAP (*Simple Object Access Protocol*)
 - est un protocole d'échange inter-applications indépendant de toute plate-forme, basé sur le langage XML.
 - Un appel de service SOAP est un flux ASCII encadré dans des balises XML et transporté dans le protocole HTTP.
 - WSDL (*Web Services Description Language*)
 - donne la description au format XML des Web Services en précisant les méthodes pouvant être invoquées, leurs signatures et le point d'accès (URL, port, etc.).
 - UDDI (*Universal Description, Discovery and Integration*)
 - normalise une solution d'annuaire distribué de Web Services, permettant à la fois la publication et l'exploration (recherche) de Web Services.
 - UDDI se comporte lui-même comme un Web service dont les méthodes sont appelées via le protocole SOAP.



Pile des standards des services Web





Références



- 1 <http://fr.wikipedia.org/wiki/Paradigme>
- 2 <http://design-patterns.fr/introduction-a-la-programmation-orientee-objet>
- 3 <http://fr.wikipedia.org/wiki/Middleware>
- 4 http://blog.xebia.fr/2009/04/29/soa-du-__composant-au-service-lautonomie