

Ir Denis Voituren

# **Service Oriented Architecture**

## **Agenda**

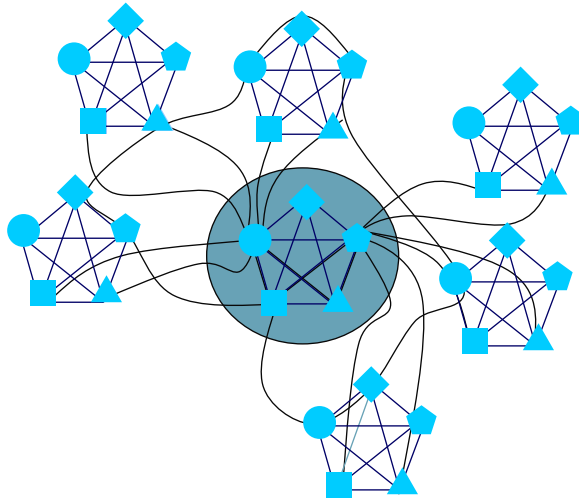
- Vue d'ensemble
- Technologies utilisées
- Développement d'un service web
- Utilisation d'un service web
- Cycle de vie
- Démonstrations

## **Vue d'ensemble**

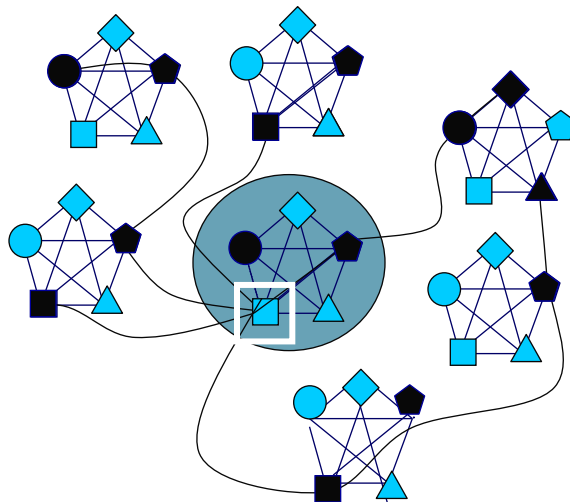
### **Les affaires aujourd'hui**

- CEO (chief executive officer)
  - Améliorer le service clientèle
  - Révision de la chaîne d'approvisionnement
  - Accélérer la vitesse de prise de décision
  
- CIO (chief information officer)
  - Réorienter l'architecture IT
  - Connexion avec un nombre limité d'éléments externes

## Sociétés et processus métiers



## Sociétés et processus métiers



# Incitant

- Les Sociétés, fournisseurs, partenaires, et clients doivent être capables de travailler ensemble
  - Beaucoup plus rapidement
  - À travers internet
  - Et le risque doit être nul

# Historique

Technology paradigm	Mainframe (1970 - today)	Client / Server (1985 - today)	Component (1995 - today)	SOA (2005 - ?)
Business needs	<ul style="list-style-type: none"> <li>■ Core Processes</li> <li>■ Collaboration</li> <li>■ Automatio</li> <li>■ Agility</li> </ul>	<ul style="list-style-type: none"> <li>■ Automated core processes</li> <li>■ Collaboration</li> <li>■ Automation</li> <li>■ Agility</li> </ul>	<ul style="list-style-type: none"> <li>■ Automated departmental processes</li> <li>■ Hard-Wired integration</li> <li>■ Collaboration</li> <li>■ Automation</li> <li>■ Agility</li> </ul>	<ul style="list-style-type: none"> <li>■ Virtual company</li> <li>■ Collaboration</li> <li>■ Automation</li> <li>■ Agility</li> </ul>
Applications	<ul style="list-style-type: none"> <li>■ Custom applications</li> <li>■ Core financials</li> </ul>	<ul style="list-style-type: none"> <li>■ Integrated standard software (ERP)</li> </ul>	<ul style="list-style-type: none"> <li>■ Composite applications (CRM, SCM, extended ERP)</li> </ul>	<ul style="list-style-type: none"> <li>■ Orchestrated services</li> <li>■ Modelling</li> </ul>
Key middleware	<ul style="list-style-type: none"> <li>■ OS</li> </ul>	<ul style="list-style-type: none"> <li>■ RDBMS, MOM</li> </ul>	<ul style="list-style-type: none"> <li>■ EAI, application server, first-generation Web services</li> </ul>	<ul style="list-style-type: none"> <li>■ ESB; BPM engines, Web services</li> </ul>
Topology	<ul style="list-style-type: none"> <li>■ Centralized</li> <li>■ 1 - 2-tier</li> </ul>	<ul style="list-style-type: none"> <li>■ Distributed</li> <li>■ 2 - 3-tier</li> </ul>	<ul style="list-style-type: none"> <li>■ Distributed</li> <li>■ 3 - N-tier</li> </ul>	<ul style="list-style-type: none"> <li>■ Distributed</li> <li>■ N-tier</li> </ul>
Networking	<ul style="list-style-type: none"> <li>■ Dumb terminal</li> <li>■ SNA</li> </ul>	<ul style="list-style-type: none"> <li>■ Ethernet</li> </ul>	<ul style="list-style-type: none"> <li>■ Internet / http</li> </ul>	<ul style="list-style-type: none"> <li>■ Internet</li> <li>■ SOAP</li> </ul>

From Pierre Audoin Consultants, 2006. Service Oriented Architecture (SOA) Germany 2006, Vision, Markets and Strategies 2005-2009

## Applications distribuées

- Modèle Client/Serveur
  - Peut extensible
  - Pas sécurisé
  
- Modèle d'objet distribué
  - Composants : Package et interopérabilité
  - Remoting : Appel des méthodes à distance
  - COM, CORBA, Java RMI et EJB
    - N'est pas pensé pour Internet
    - Interopérabilité : pauvre/pas réellement de standard

## Applications distribuées

- Architecture 3-tiers
  - Possibilité de créer des applications web extensibles

# Portails

The screenshot shows a web portal with the following sections:

- Mail:** Check Voicemail, Email, Calendar, Address Book, Contact, Logout, Colors.
- Calendar:** Today On Events, Tuesday, January 11, 10:52PM, News, World, Desktop Calendar, Full AOL Insurance, Billing, Billing - My First Sale, New! Taxes, Step by Step, Listen - Musical E-Cards.
- Finance:** My Stocks, Charles Schwab, Full Executive Stock News, Symbol, Price, Change, News, Dow, S&P 500, NASDAQ, NYSE, CBOE, NYSE, NYSE, NYSE, NYSE.
- Ads:** Autos, Business, Computers, Education, Entertainment, Family, Games, Health, Investing, Lifestyle, Real Estate, Relationships, Sports, Travel.
- News:** Top Stories | Photos, ZNet Tech News, Business News, Sports News, Daily Enough News.
- Other Svcs:** Today's News, Free E-Card, Stock Quotes, Yellow Pages, Free ZIP, Free E-Card Download, Store Albums, Classifieds, Maps & Directions, Horoscopes, More...
- Weather:** Forecast for Wed Jan 12, Allentown, MA, 35° - 41°, Seattle, WA, 38° - 41°, Seattle King County Int'l, 30° - 42°.

## Limite du portail

- Pas de standard pour exposer les différentes fonctionnalités
- L'intégration coûte chère
- N'est pas pensé pour être utilisé en dehors du but initial fixé
- Problème ?
  - HTML est un langage de présentation
  - On ne peut redéfinir les fonctionnalités d'une manière fiable
  - Possibilité de problème lors de l'affichage à l'écran

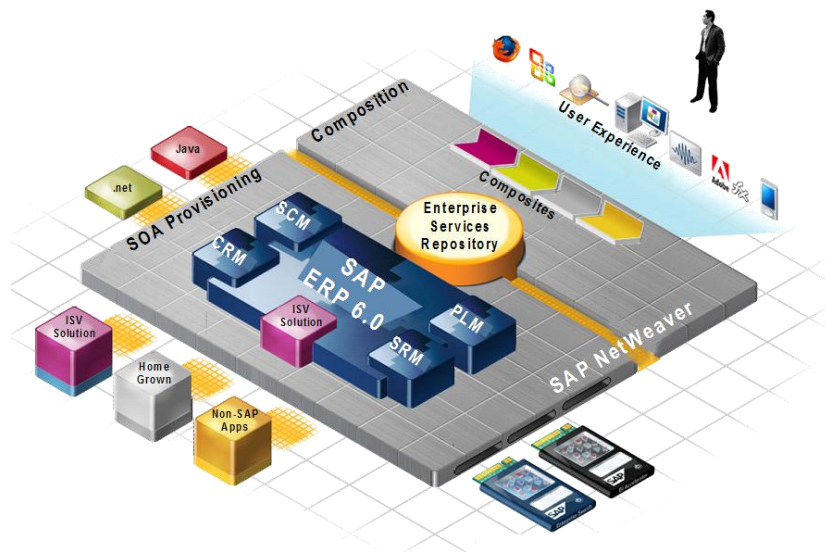
## Solution : SOA

- Architecture orientée services
  - Modèle d'interaction applicative qui met en œuvre des services
  - Le service est une action exécutée par un *Fournisseur* à l'attention d'un *Client*
  - Très souvent basé sur les *services web*

## Service Web

- Offre des fonctionnalités à un client
  - À travers internet ou un intranet
  - Une URL programmable
  - Des fonctions que vous pouvez appeler à travers internet
- Basé sur des standards W3C
  - HTTP, XML, SOAP, WSDL, UDDI, etc.
- Peut être implémenté dans n'importe quel langage et sous n'importe quelle plate-forme

# Business Process Platform



## Service Web

- Un service web combine à merveille les meilleurs fonctionnalités des applications distribuées et des portails en éliminant les mauvais éléments
  - Fournit un mécanisme qui permet d'appeler des méthodes à distance
  - Utilise des standards web (e.g. HTTP, XML) pour l'appel de méthodes

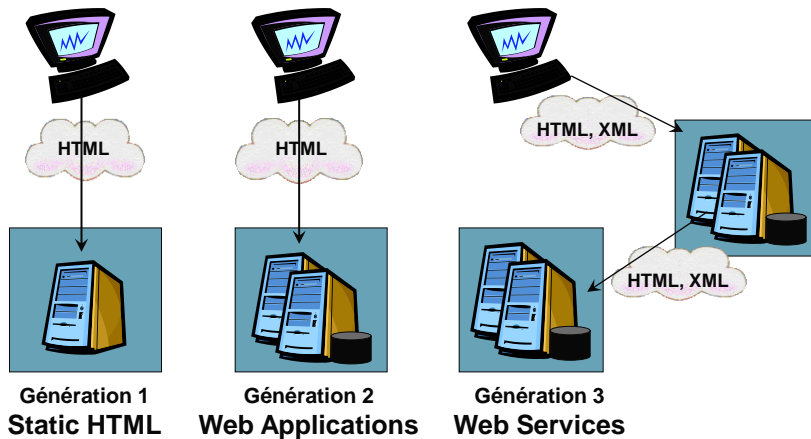
## **Service Web**

- Les services web autorisent l'interconnexion entre :
  - Différentes sociétés
  - Différents appareils
  - Applications
  - Différents clients
    - Et pas seulement au niveau des navigateurs
- Distribution et intégration d'applications logiques

## **Service Web**

- Nouvelle philosophie pour le développement
- Offre des applications comme des services
- Internet de " Troisième génération "

## Evolution du Web



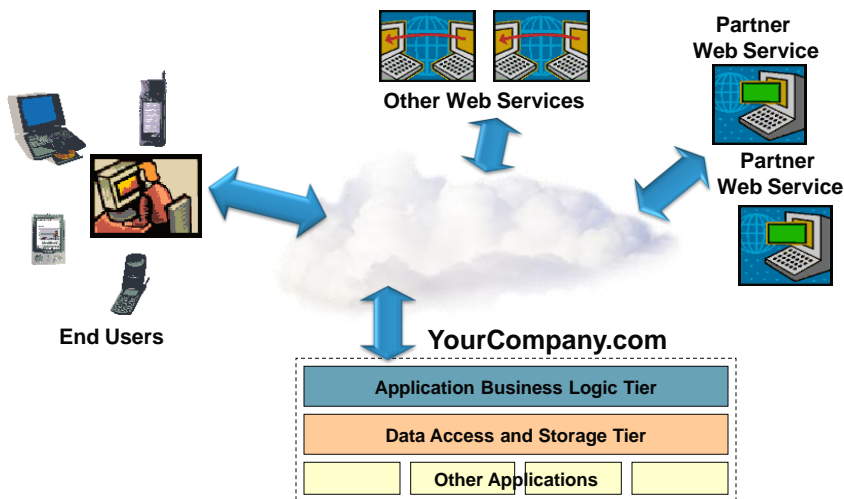
## Avantages

- Pour tout le monde
  - Augmente la puissance des applications existantes
  - "Construire et acheter" les décisions de développement
  - Diminuer le temps et le coût du développement
- Entreprises
  - Impératif d'intégration
  - Dynamique, facilite la relation B2B
- Nouvelle philosophie commerciale basée sur le web
  - Plus personnalisée
  - Nouveaux services/nouveau type revenu
  - Soyez "partout" plutôt que sur une seule cible

## Exemple : planifier un voyage

- Aller sur le site web
- S'identifier via OpenID
- Trouver le vol concerné
  - Les informations concernant votre profil (place, companies, etc.) n'ont plus besoin d'être réintroduites
  - Le système peut trouver la place la moins chère en fonction des disponibilités de votre agenda
- Paiement du ticket avec carte
- Les informations concernant votre nouveau voyage sont automatiquement introduites dans votre agenda
- Les informations complémentaires (voiture, hôtel, etc.) sont automatiquement ajoutées à votre agenda
- Le jour du voyage, recevez les informations concernant le vol via email, téléphone portable, etc.

## Modèle d'application



## Exemple de service web

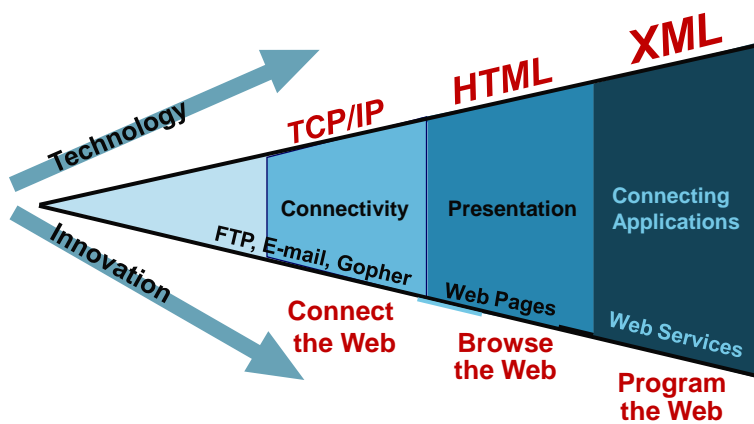
- E-commerce : commande de livres, fournitures de bureau, autres produits
- Envoi de courrier : UPS, FedEx
- Temps
- Carte routière
- Redirection du téléphone, personnalisation des messages, etc.

## Exemples SAP



# Technologies

## Technologies



## Services Web

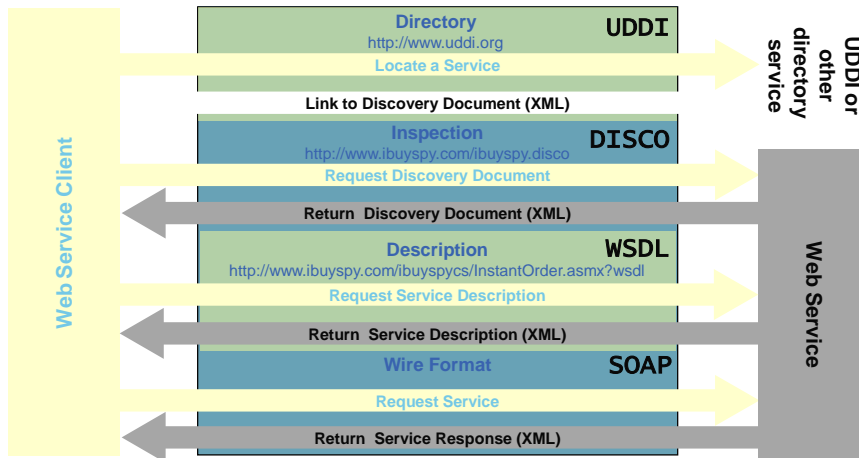
Directory: Publish & Find Services:	UDDI
Inspection: Find Services on server:	DISCO
Description: Formal Service Descriptions:	WSDL
Wire Format: Service Interactions:	SOAP
Universal Data Format:	XML
Ubiquitous Communications:	Internet

**Simple, Open, Broad Industry Support**

## Couches

- Discovery
  - Les Directory offrent la possibilité au client de localiser les web services intéressants
    - UDDI
  - Les Inspection offrent la possibilité de localiser les documents qui traitent des web services à une adresse (URL) donnée
    - DISCO
  - Une Description permet de définir le format des méthodes fournies par un web service
    - WSDL

## Différentes couches



## Web Service Wire Format

- Format d'envoi d'un message
  - HTTP-GET
  - HTTP-POST
  - SOAP
  
- Le SOAP fournit une solide interface HTTP/XML
  - Support étendu pour les types de données

## XML - Base

- XML a été étudié pour représenter et envoyer des données structurées
  - En HTML
    - `<p>Jan 15, 2000</p>`
  - En XML
    - `<OrderDate>Jan 15, 2000</OrderDate>`
- Le XML n'a pas été créé pour afficher ou formater les données
  - Le langage HTML et XML sont tous les deux dérivés du SGML

## Syntaxe XML

- Le XML est composé de balises et d'attributs (ou propriétés)

```
<root>  
<orders orderid="10643" customerid="ALFKI"  
employeeid="6" orderdate="1997-08-25T00:00:00"  
requireddate="1997-09-22T00:00:00"  
shippeddate="1997-09-02T00:00:00" />  
</root>
```

## Schéma XML

- Le schéma XML décrit la structure d'un document XML
  - Le schéma XML décrit les spécifications des balises et des propriétés
  - Le schéma XML décrit également les contraintes liées au contenu
  - Le schéma XML et les DTD sont liés

## SOAP

- Le SOAP est un puissant protocole pour l'échange d'informations dans un environnement d'application distribuée et/ou hétérogène
  - Il autorise l'interopérabilité entre les différentes plateformes
- interopérabilité
  - OS, Object, le langage de programmation est neutre
  - Indépendant du matériel
  - Indépendant de tout protocole
- Travail sur les infrastructures existantes d'internet

## SOAP

- Objectif principal : “ne pas inventer de nouvelle technologie”
- Basé sur des standards Internet
  - SOAP  $\approx$  HTTP + XML
  - Cautionné par le W3C
- Les spécifications SOAP définissent :
  - Le format des messages SOAP
  - Comment envoyer un message
  - Comment recevoir une réponse
  - Encodage des données

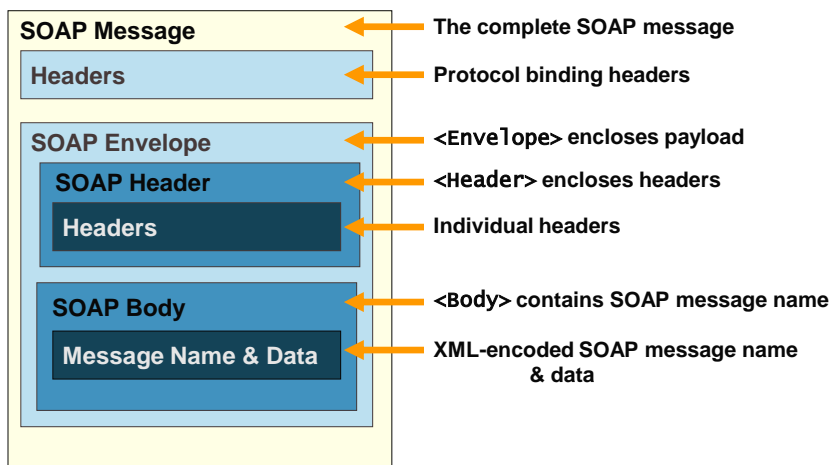
## SOAP

- Une requête SOAP est une requête HTTP POST

```
POST /webCalculator/Calculator.asmx HTTP/1.1
Content-Type: text/xml
SOAPAction: "http://tempuri.org/Add"
Content-Length: 386

<?xml version="1.0"?>
<soap:Envelope ...>
  ...
</soap:Envelope>
```

## Structure d'un message



## Sécurité

- Construit sur la sécurité HTTP
  - HTTPS
  - X.509 certificates
- Les développeurs choisissent quelle méthode ils utilisent
- Firewall-friendly
- Type safe

## Support Industriel

- DevelopMentor Inc.
- Digital Creations
- IONA Technologies PLC
- Jetform
- ObjectSpace Inc.
- Rockwell Software Inc.
- SAP
- Compaq
- Microsoft
- Rogue Wave Software Inc.
- Scriptics Corp.
- Secret Labs AB
- UserLand Software Inc.
- Zveno Pty. Ltd.
- IBM
- Hewlett Packard
- Intel

## WSDL

- Web Services Description Language
- Utilisation des schémas XML pour décrire les Web Services
  1. Définition de l'interface de service
  2. Définition de l'implémentation des services
    - Adresse d'où les Web Service peuvent être appelées
- Séparation entre les messages abstraits et concrets

## WSDL

- Ouvert – disponible pour les autres espaces de nom, très extensible
- Capable d'importer d'autres schémas et WSDL
- Fournit la "recette" pour les Web Services
- Offre l'interface et le détail de l'implémentation

## WDSL - Exemple

### Démonstration



## Inspection : DISCO

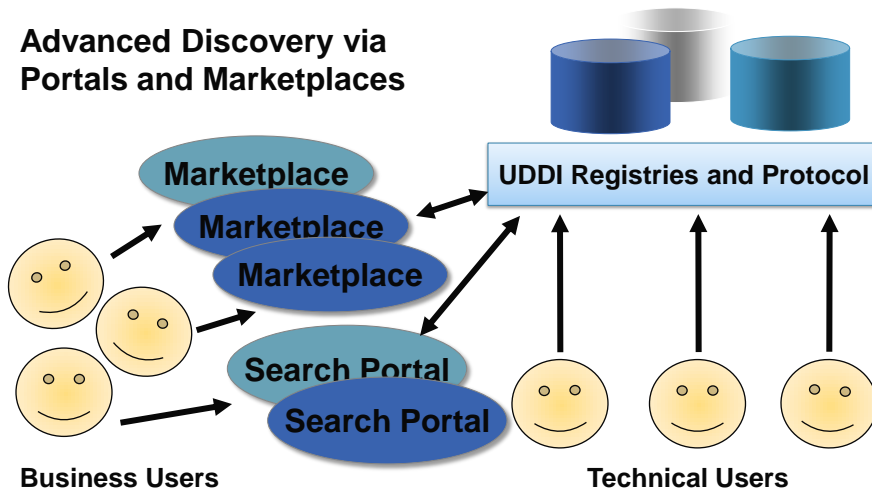
- L'Inspection vous donne la possibilité de trouver un Web Services sur un site donné
  - Un site internet publie un document DISCO, lequel retourne l'URL de la description WSDL
  - Le document DISCO peut contenir des références à d'autres sites ou d'autres documents DISCO
  - La recherche dynamique offre une palette de lieux où l'on peut rechercher les services

## UDDI

- UDDI = Universal Description, Discovery, and Integration
- Mécanisme standard pour la publication et la localisation des sociétés et des services qu'elles proposent.
- Spécifications
  - Schéma pour les services et leurs descriptions
  - API pour la publication et la recherche
  - Développé sur des standards industriels (XML, HTTP, TCP/IP, SOAP)
  - Applicable sur des services XML ou non-XML
- Implémentation
  - Instance de spécification publique ou privée

## Vision

### Advanced Discovery via Portals and Marketplaces



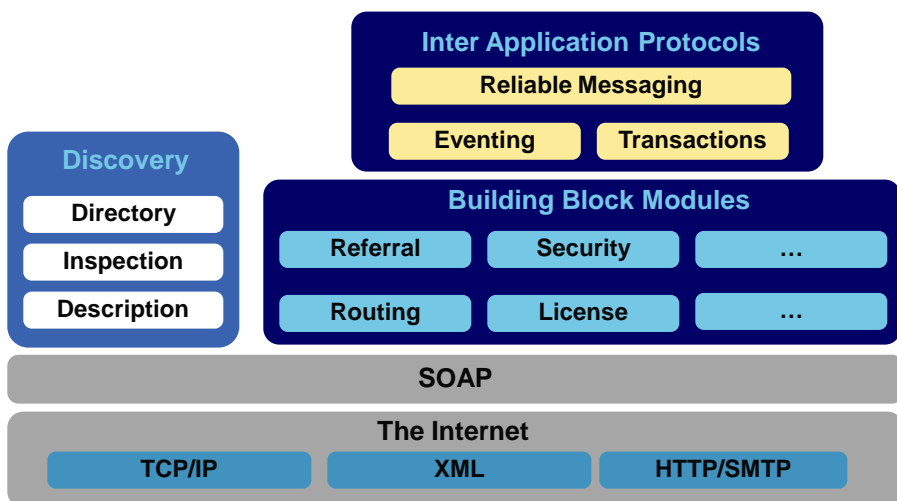
## Providers, Services et Bindings

- Providers
  - Exemples: Service comptable, Corporate Application Server
  - Nom, Description, Informations sur les contacts
  - Information d'identification et de catégorisation
- Services
  - Exemples: Service d'achat, Service de paiement
  - Nom, Description(s)
  - Information de catégorisation
- Bindings
  - Description(s), points d'accès (Access Point ), paramètres
  - Exemples: Access Point (<http://...>) pour les Web Services

## UDDI

- Neutre en terme de protocole – comme un registry, il peut pointer sur tout
- Utilisation de Globally Unique Identifiers (GUIDs)
- Séparation entre l'interface et l'implémentation

## Global Services Architecture

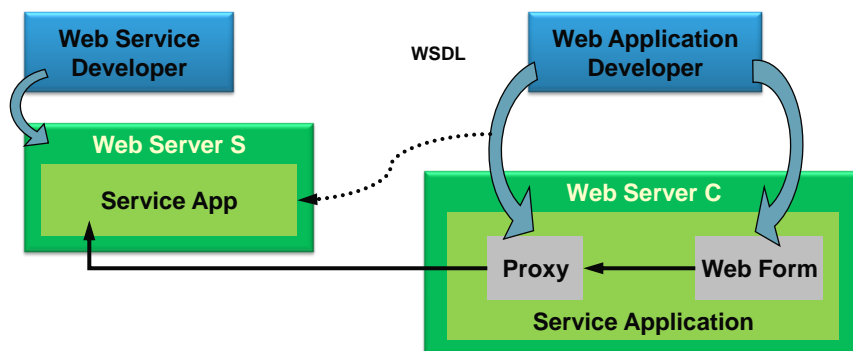




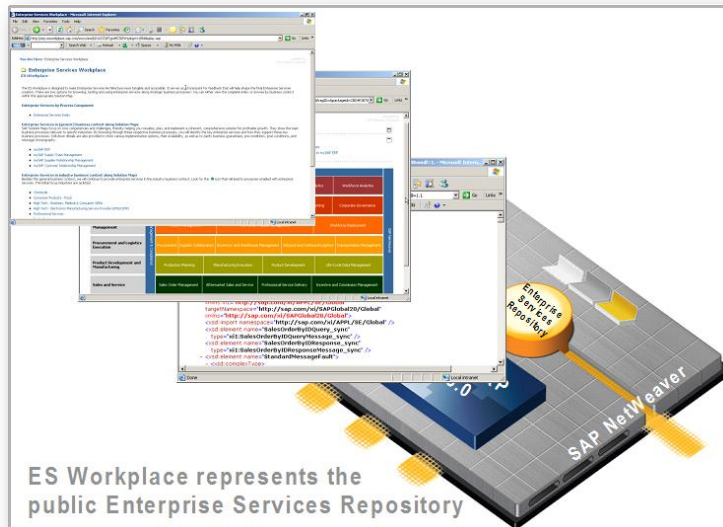
## Vue d'ensemble

- Localisation du Web Services désiré
  - UDDI
  - DISCO
- Obtenir des détails concernant le Web Services
  - WSDL
- Création d'un proxy qui représente le Web Services
  - Le Proxy a les mêmes méthodes/arguments/valeur de retour que le Web Services

## Vue d'ensemble



## Vue d'ensemble



## Vue d'ensemble

- Accès aux Web Services via l'URL
  - HTTP request/response
- Peut envoyer une requête WSDL via URL
- Peut appeler via :
  - HTTP-GET
  - HTTP-POST
  - HTTP-SOAP

## Via HTTP-GET

### ■ HTTP-GET

```
http://localhost/MathService.asmx/Multiply?a=11&b=11
```

### ■ Le Résultat est un document XML

```
<?xml version="1.0" encoding="utf-8" ?>  
<int xmlns="http://www.mysite.com/MathService/">  
  121  
</int>
```

## Via HTTP-POST

### ■ HTTP-POST

```
POST /MathService.asmx/Multiply HTTP/1.1  
Host: localhost  
Content-Type: application/x-www-form-urlencoded  
Content-Length: length
```

```
a=11&b=11
```

### ■ Le Résultat est un document XML

```
<?xml version="1.0" encoding="utf-8" ?>  
<int xmlns="http://www.mysite.com/MathService/">  
  121  
</int>
```

## Via HTTP-SOAP

- Grammaire XML pour
  - WebMethod, Méthode, paramètre, résultat
- Gère tous les types de données et classes standard .NET
- Class and structure
  - Sérialisation au format XML

## Type Marshalling

- Utilisation de HTTP-GET ou HTTP-POST
  - types primitifs
    - E.g. integer, string, double, byte, ...
  - Types Enum
  - Tableau de type primitif et d'Enum
  - Transmission par valeur uniquement

## Type Marshalling

- Utilisation du SOAP
  - Types Primitive
  - Types Enum
  - Classes and Structure
  - DataSet
  - XmlNode
  - Transmission par valeur et par référence

## Utilisation

- Requête avec des paramètres "WSDL"
  - Description WSDL formel du Web Services
  - Basé sur la grammaire XML

<http://localhost/MathService.asmx?WSDL>

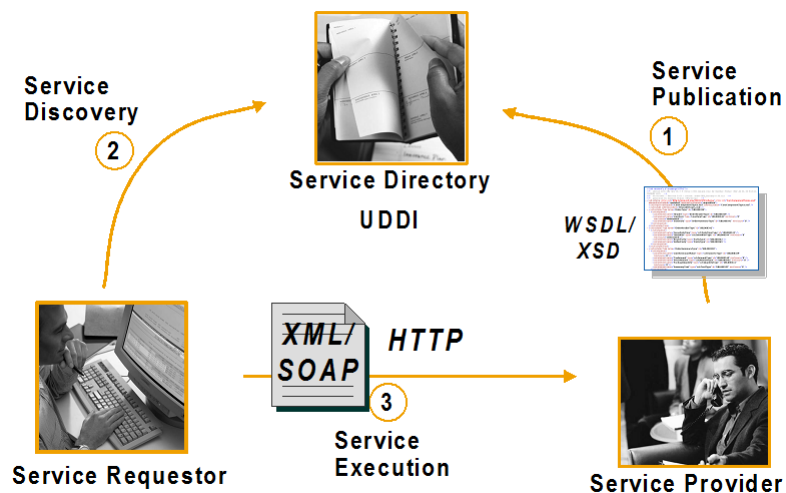
## Création d'un proxy en C#

- Utilisation du `wsdl.exe` pour générer une classe Proxy

```
wsdl http://localhost/MathService.asmx?WSDL
```

- Créer `MathService.cs`
- Le Proxy scèle l'URL du Web Services dans le constructeur

## Cycle complet

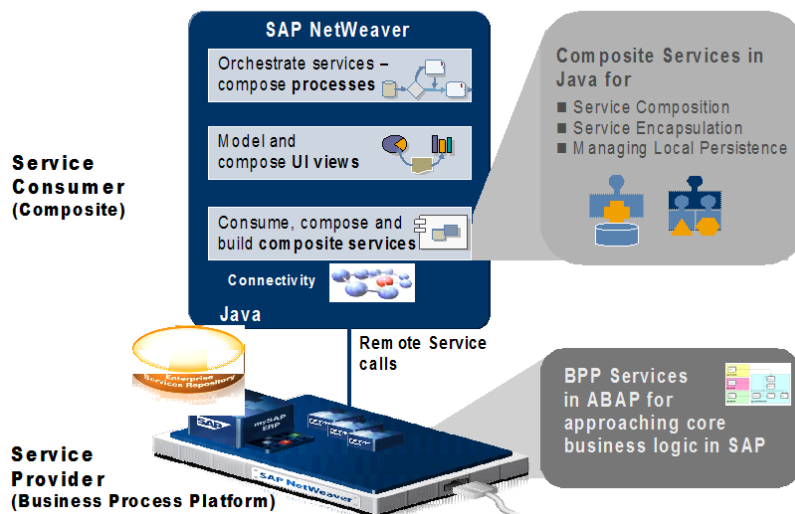


# Demonstration

- Utilisation de MathService.asmx



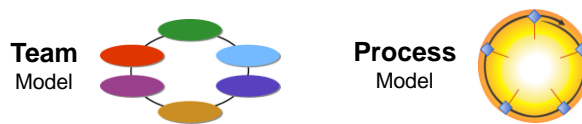
# Développement sous SAP



# Cycle de vie

## Modèles et disciplines

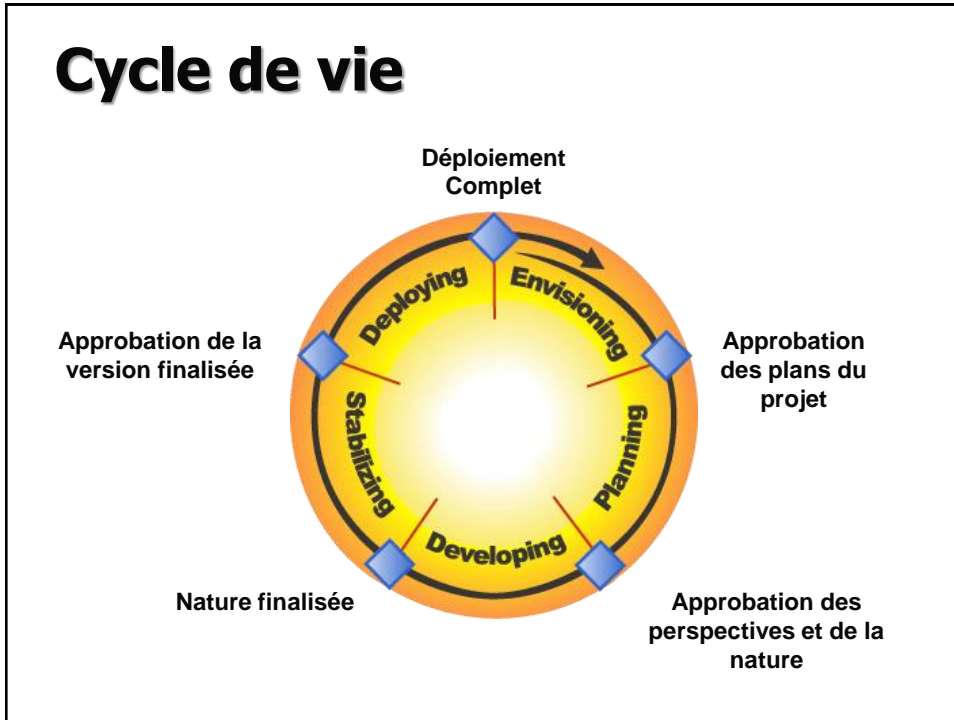
### Models



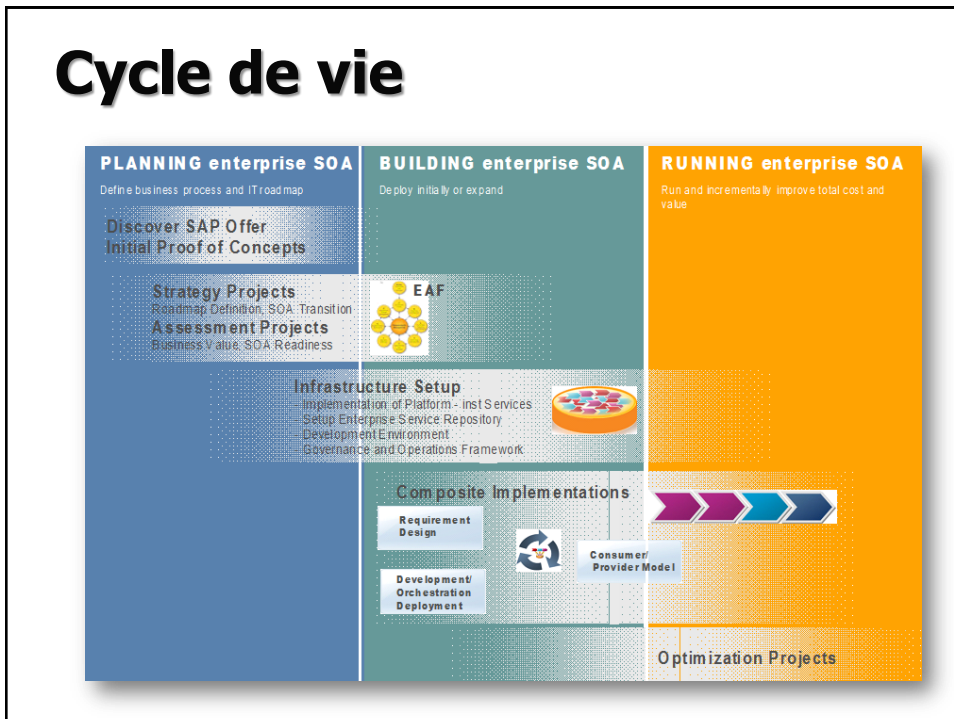
### Disciplines



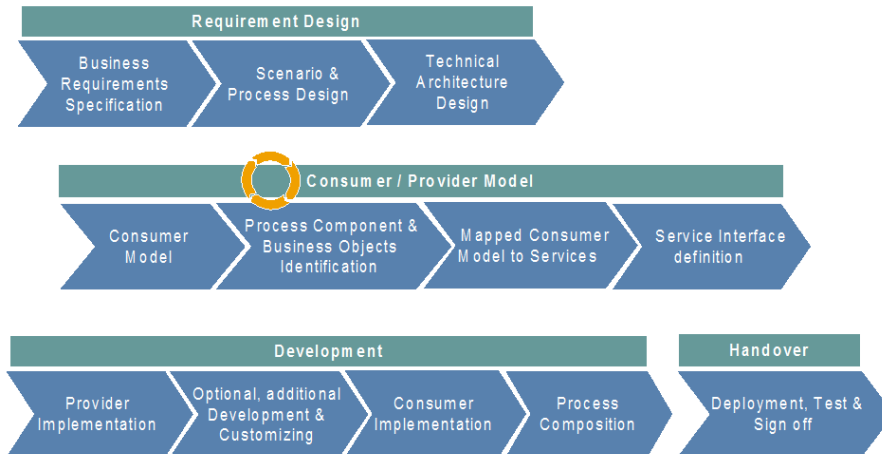
# Cycle de vie



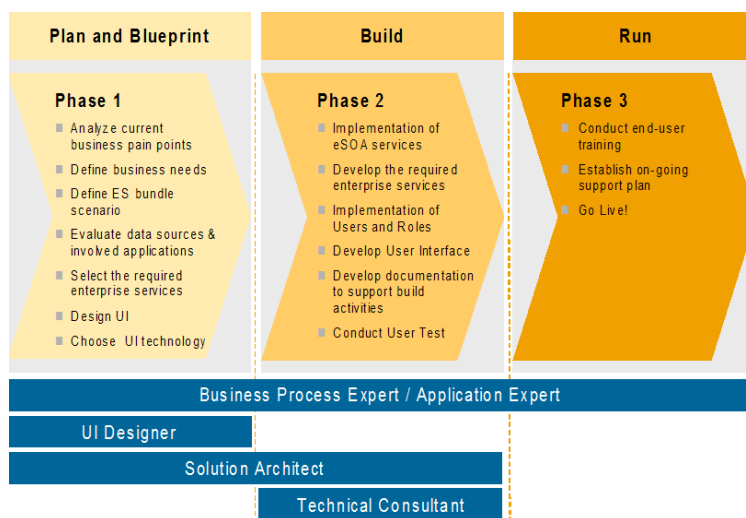
# Cycle de vie



# Cycle de vie



# Phases et rôles



# Implémentation

## 1. Requirement Design

- Business requirement specification
- Scenario & process design
- Technical architecture



## 3. Development / Handover

- Provider implementation
- Optional, additional development & customizing
- Consumer implementation
- Process composition
- Deployment



## 2. Consumer / Provider Model

- Consumer model
- Identified process Components and business objects
- Mapped consumer model to services
- Service interface definition



- Supporting IT Organization
- Governance and Operations
  - Organizational aspects
  - New roles and tasks
  - New implementation methodology

# Conclusions

## Ressources

- WSDL Specification  
<http://www.w3.org/TR/wsdl>
- A Quick Introduction to WSDL  
[http://msdn.microsoft.com/library/default.asp?url=/library/enus/soap/hm/soap\\_overview\\_72b0.asp?frame=true](http://msdn.microsoft.com/library/default.asp?url=/library/enus/soap/hm/soap_overview_72b0.asp?frame=true)
- UDDI  
<http://www.uddi.org>  
<http://uddi.microsoft.com>
- HailStorm  
<http://www.microsoft.com/net/hailstorm.asp>
- Building Web Services with SOAP and ASP.NET  
<http://msdn.microsoft.com/msdnmag/issues/01/02/WebComp/webcomp.asp>