



## Travaux de recherches dans les architectures orientées services

Tarak Chaari

Maître assistant à l'institut supérieur d'électronique  
et de communication

[tarak.chaari@redcad.org](mailto:tarak.chaari@redcad.org)

# Votre interlocuteur

 **Tarak CHAARI**

 **Maître assistant à l'ISECS**



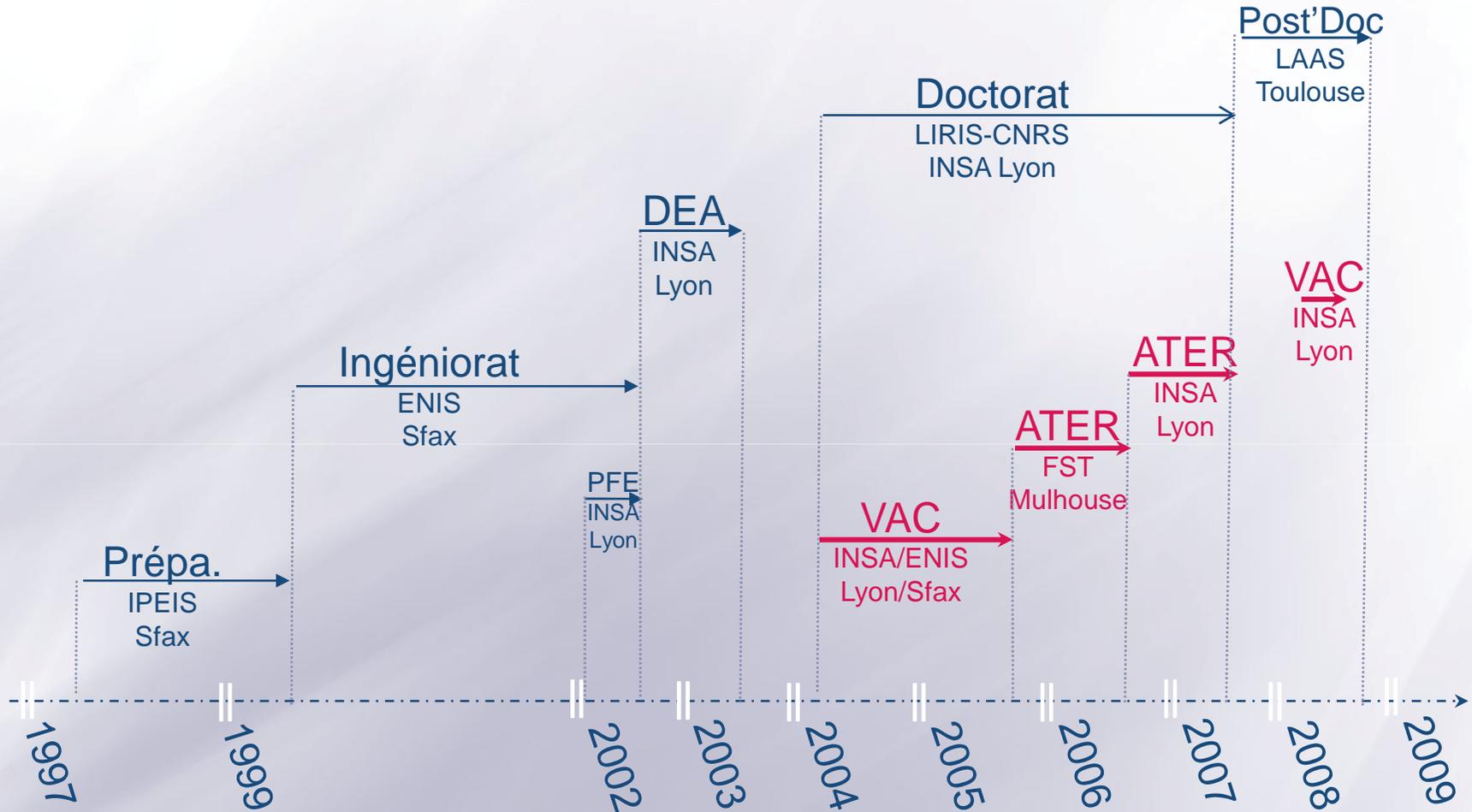
 **Membre de l'unité de recherche RedCAD**



 **Enseignement: Ingénierie des systèmes d'information**

 **Recherche: l'adaptation dans les environnements dynamiques**

# Cursus universitaire



- Formation & Recherche
- Enseignement
- VAC Vacataire

# Présentation générale

## Le nom de l'intervention cours

- Travaux de recherche dans les architectures orientées services : ADL & Adaptation

## Volume horaire

- 18 heures
- Cours

## Objectifs

- Avoir une idée sur les travaux de recherche liés aux architectures orientées services
- Adaptation et adaptabilité

# Plan

1. Introduction aux architectures orientées services (SOA)
2. Travaux de recherche liés aux SOA
3. Sémantique dans les SOA
4. L'adaptation au contexte dans les SOA



# Besoins des Applications Web

## Le Web classique

- Conçu pour les applications à interactions humaines
- Partage d'information
- E-commerce de type B2C

## Extensions nécessaires

- Accès à l'information depuis des programmes
- Processus de e-commerce B2B automatisés
- Interopérabilité entre applications
- Plate-forme pour le calcul distribué

# Principes des services Web

- Proposer une plate-forme de traitement distribuée autour d'Internet/Intranet
- Permettre l'interopérabilité totale entre composants
- Offrir un modèle de composants simple
  - Modules avec interfaces de fonctions
  - Paramètres structurés et diversifiés en place des URL longues contraignantes (ASCII, plats,  $\leq 255$ )
- Permettre de découvrir dynamiquement les services
  - Annuaire de services et service de recherche
- Offrir des outils d'intégration de composants
- A l'échelle de l'Internet ou de l'Intranet

# Exemple d'applications

## Diffusion d'information

- horaires, incidents, états de stocks
- cours de bourse, entrées d'annuaires, etc.

## Contrôle d'information

- autorisation de cartes de crédits
- authentification d'un client
- Mise à jour de logiciels

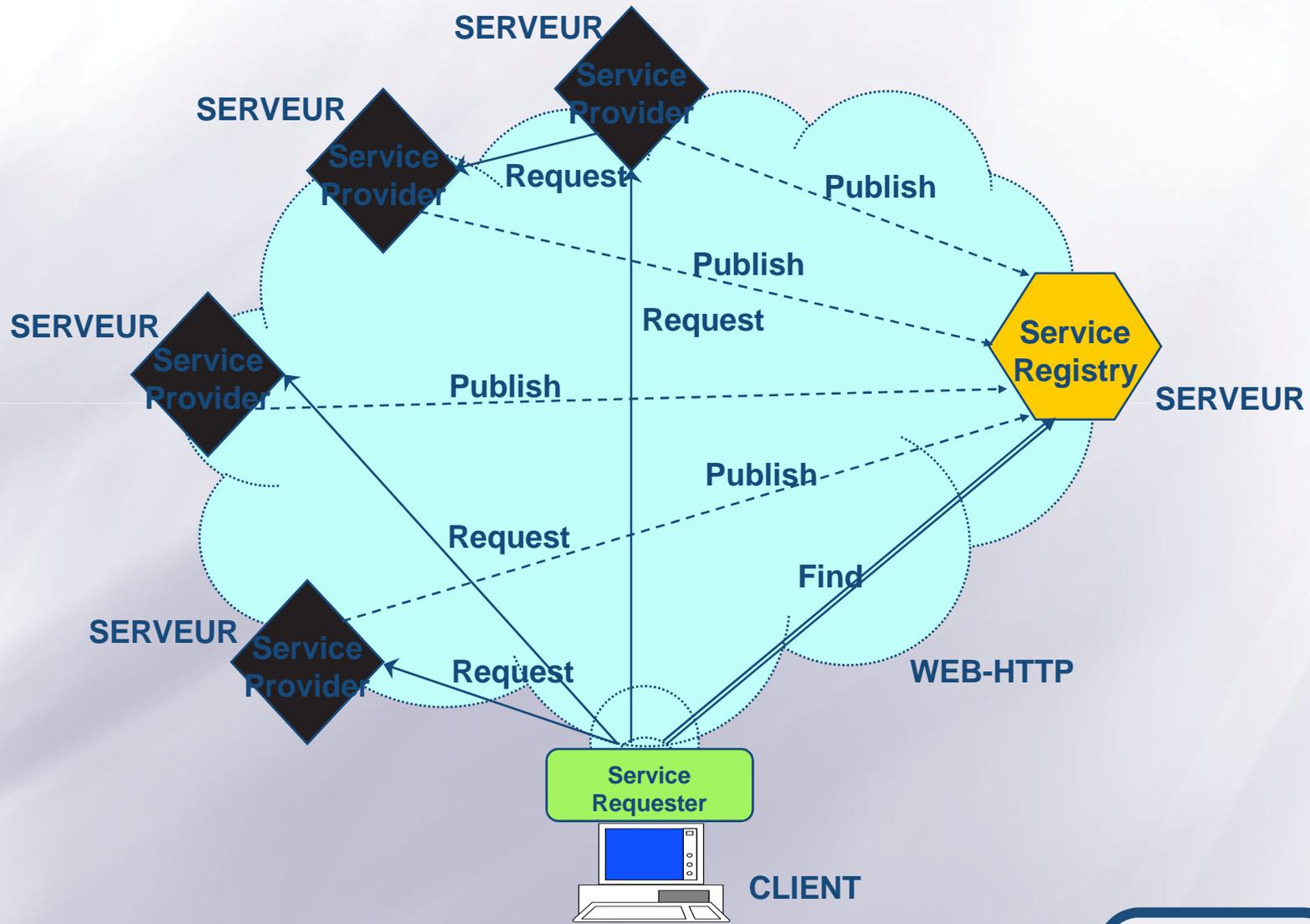
## Documentation automatique

- présentation, marketing, veille technologique
- librairie électronique, manuels, maintenance, etc.
- communication et "knowledge management"

## Commerce électronique

- présentation, sélection, transaction, médiation, etc.

# Architecture Web Services



# Qu'est ce qu'un Web Service ?

## ☰ Définition [W3C]

- Un Web service est un composant logiciel identifié par une URI, dont les interfaces publiques sont définies et appelées en XML.
- Sa définition peut être découverte par d'autres systèmes logiciels.
- Les services Web peuvent interagir entre eux d'une manière prescrite par leurs définitions, en utilisant des messages XML portés par les protocoles Internet.

## ☰ Exposition

- Langage WSDL utilisé pour décrire le service
- Similaire à IDL mais basé sur XML

## ☰ Activation

- Protocole Web au-dessus de HTTP (RPC XML, SOAP)
- Autres protocoles possibles ...

# Les Composants

## Service Provider (Fournisseur de service)

- Application s'exécutant sur un serveur et comportant un module logiciel accessible par Internet en XML

## Service Registry (Annuaire de service)

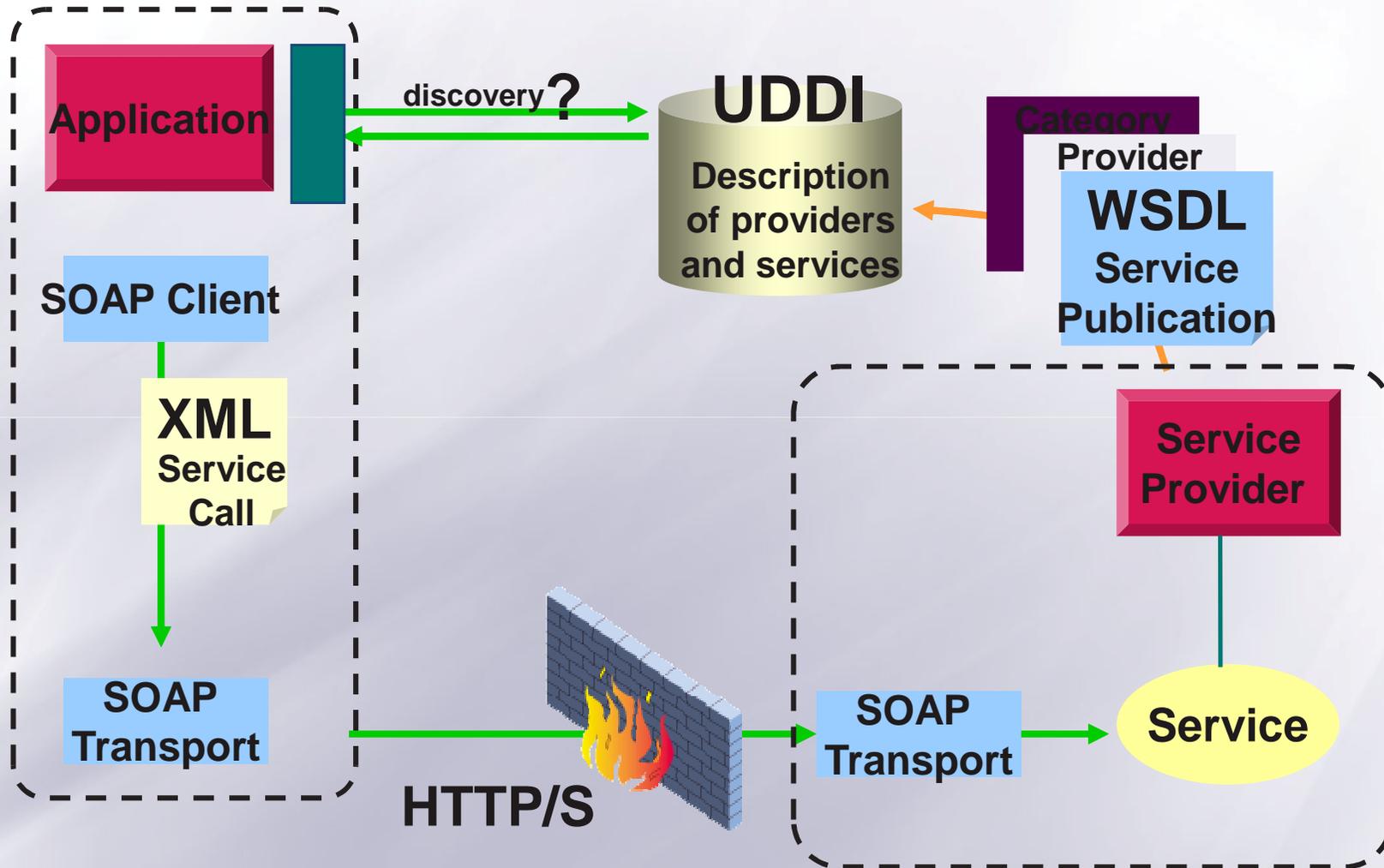
- Annuaire des services publiés par les providers (UDDI)
- Géré sur un serveur niveau application, entreprise ou mondial

## Service Requester (Demandeur de service)

- Application cliente se liant à un service et invoquant ses fonctions par des messages XML (SOAP)



# Vue d'ensemble



# Description des services: WSDL

## Elément Type

- Types des paramètres (schéma XML)

## Elément Message

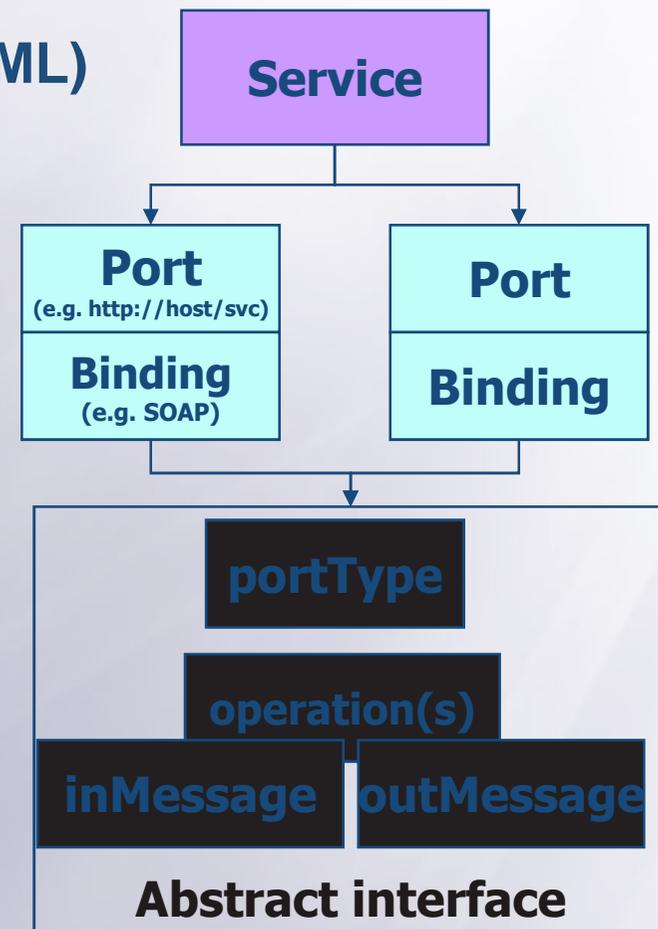
- Appel et retour d'opération

## Elément Port type

- Groupe d'opération

## Elément Binding

- URL de l'opération
- Type de protocole



# Description en WSDL

```
<definitions name = "... " xmlns: ...>  
  <types>  
    <!--Définition des types de données; basés sur ceux des schémas --> ...  
  </types>  
  <message>  
    <!--Déclaration des messages (entrées et sorties)--> ... </message>  
  <portType>  
    <!--Déclaration des opérations (par association des messages)--> ...  
  </portType>  
  <binding>  
    <!--Définition de la liaison WSDL – SOAP (noms d'actions et codages)-->  
  </binding>  
  <service name= " ... " >  
    <!--Déclaration des ports (groupes d'opérations et protocoles d'accès)--  
    >... </service>  
</definitions>
```



# Exemple: Service Web GoogleSearch (les types)

```
<xsd:complexType name="GoogleSearchResult">
  <xsd:all>
    ...
    <xsd:element name="estimatedTotalResultsCount" type="xsd:int"/>
    <xsd:element name="resultElements" type="typens:ResultElementArray"/>
    ...
  </xsd:all>
</xsd:complexType>

<xsd:complexType name="ResultElementArray">
  <xsd:complexContent>
    <xsd:restriction base="soapenc:Array">
      <xsd:attribute ref="soapenc:arrayType" wsdl:arrayType="typens:ResultElement[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="ResultElement">
  <xsd:all>
    <xsd:element name="summary" type="xsd:string"/>
    <xsd:element name="URL" type="xsd:string"/>
    <xsd:element name="title" type="xsd:string"/>
    <xsd:element name="cachedSize" type="xsd:string"/>
    ...
  </xsd:all>
</xsd:complexType>
```



# Exemple: Service Web GoogleSearch (les messages)

```
<message name="doGoogleSearch">  
  <part name="key"      type="xsd:string"/>  
  <part name="q"        type="xsd:string"/>  
  <part name="start"    type="xsd:int"/>  
  <part name="maxResults" type="xsd:int"/>  
  ...  
</message>
```

```
<message name="doGoogleSearchResponse">  
  <part name="return"   type="typens:GoogleSearchResult"/>  
</message>
```

```
<message name="doGetCachedPage">  
  <part name="key"      type="xsd:string"/>  
  <part name="url"      type="xsd:string"/>  
</message>
```

```
<message name="doGetCachedPageResponse">  
  <part name="return"   type="xsd:base64Binary"/>  
</message>
```

```
<message name="doSpellingSuggestion">  
  <part name="key"      type="xsd:string"/>  
  <part name="phrase"   type="xsd:string"/>  
</message>
```

```
<message name="doSpellingSuggestionResponse">  
  <part name="return"   type="xsd:string"/>  
</message>
```

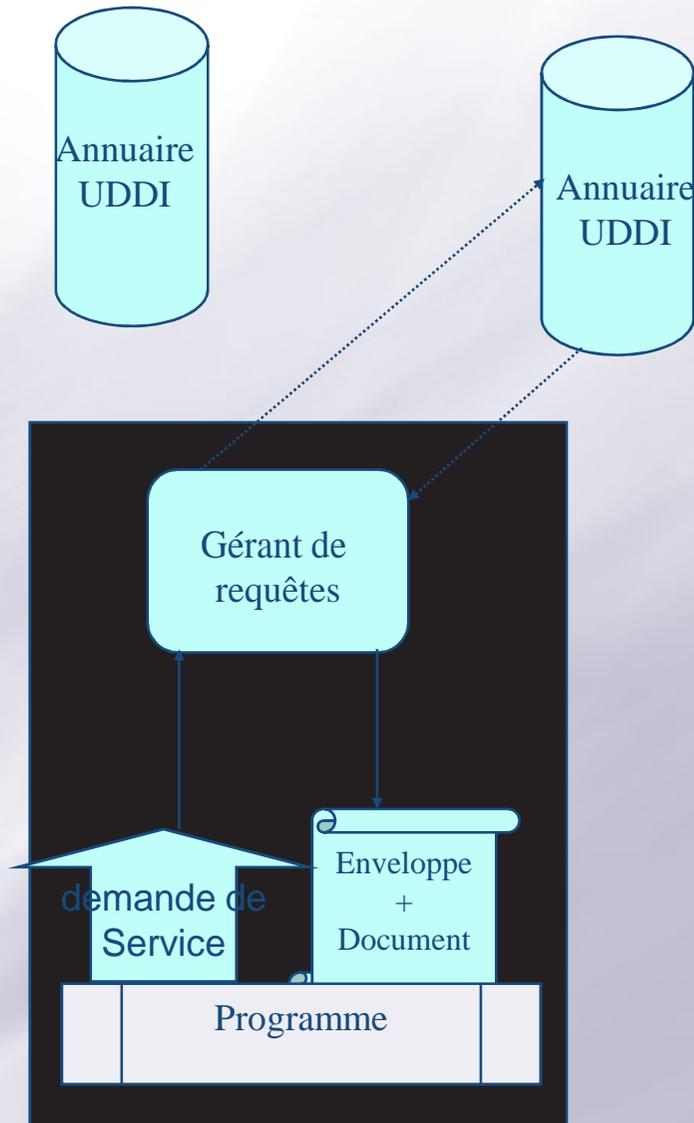


# Exemple: Service Web GoogleSearch (bindings & ports)

```
<binding name="GoogleSearchBinding" type="typens:GoogleSearchPort">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
  ...
  <operation name="doGoogleSearch">
    <soap:operation soapAction="urn:GoogleSearchAction"/>
    <input>
      <soap:body use="encoded"
        namespace="urn:GoogleSearch"
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
    </input>
    <output>
      <soap:body use="encoded"
        namespace="urn:GoogleSearch"
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
    </output>
  </operation>
</binding>

<!-- Endpoint for Google Web APIs -->
<service name="GoogleSearchService">
  <port name="GoogleSearchPort" binding="typens:GoogleSearchBinding">
    <soap:address location="http://api.google.com/search/beta2"/>
  </port>
</service>
```

# Annuaire des services: UDDI



☐ **Universal Description, Discovery and Integration**

☐ **Annuaire des services**

- décrit par un document
- WSDL, spec. EJB, autre ...

☐ **Accessible en SOAP**

☐ **Fonctions**

- Enregistrer votre société
- Enregistrer des services
- Enregistrer des opérations
- Découvrir des services
- ...

# Principaux fournisseurs

## IBM UDDI Registry

- Un registre UDDI avec des fonctionnalités de recherche
- [www-3.ibm.com/services/uddi/](http://www-3.ibm.com/services/uddi/)

## Microsoft UDDI Business Registry (UBR)

- [uddi.microsoft.com/default.aspx](http://uddi.microsoft.com/default.aspx)

## SAP UDDI Business Registry

- Public pour l'instant

## Systinet Registry

- Support complet de UDDI V3
- [www.systinet.com/products/sr/overview](http://www.systinet.com/products/sr/overview)

## Oracle Application Server UDDI Registry

- [otn.oracle.com/tech/webservices/htdocs/uddi/](http://otn.oracle.com/tech/webservices/htdocs/uddi/)

# Client Web de Recherche

The screenshot shows a Mozilla Firefox browser window titled "UDDI Browser - Mozilla Firefox". The address bar contains "http://soapclient.com/uddisearch.html" and the search engine is set to "UDDI Registry". The page content includes a navigation menu with links like "Home", "SOAP Tools", "UDDI Browser", "Resources", "Source Code", "RFCs", "News Reader", "SOAP Interop", and "Bookmarks". Below the menu is a search form titled "UDDI Browser" with fields for "Operator" (set to "Microsoft"), "Search For" (set to "hotel"), and "in" (set to "Business Names"). A "Search" button and a link to "Advanced Search" are also present.

**What is this?**

Universal Description, Discovery, and Integration (UDDI) Service is an industry-wide effort to bring a common standard for business-to-business(B2B) integration. It defines a set of standard interfaces for accessing a database of web services. As a pioneer of web service implementations, we provide this web interface to the live registries.

**How to Browse/Search?**

- Select an operator from the dropdown menu.
- Type in a string you want to searching for, and select one of the categories.
- The result page contains the business descriptions and a list of services provided. Click on the service key to see the service details.
- One of the dill-down paths is  
**Business->Services->Service Detail->Bindings->Binding Detail->tModel Detail**

**Where to Find Category Codes?**

Some UDDI searches use numerical category codes, you can find them at the following pointers:

**UDDI Links**

- UDDI Resources
- UDDI.org
- XML Cover Pages
- IBM UDDI
- MS UDDI
- SUN UDDI
- jUDDI
- UDDI Reference
- UDDI Books

**SOAP Tools**

- SOAP Security
- SOAP Message Builder
- SOAP Client
- Interop Tester
- Canonical XML
- News Reader
- SOAP Server
- Google Search

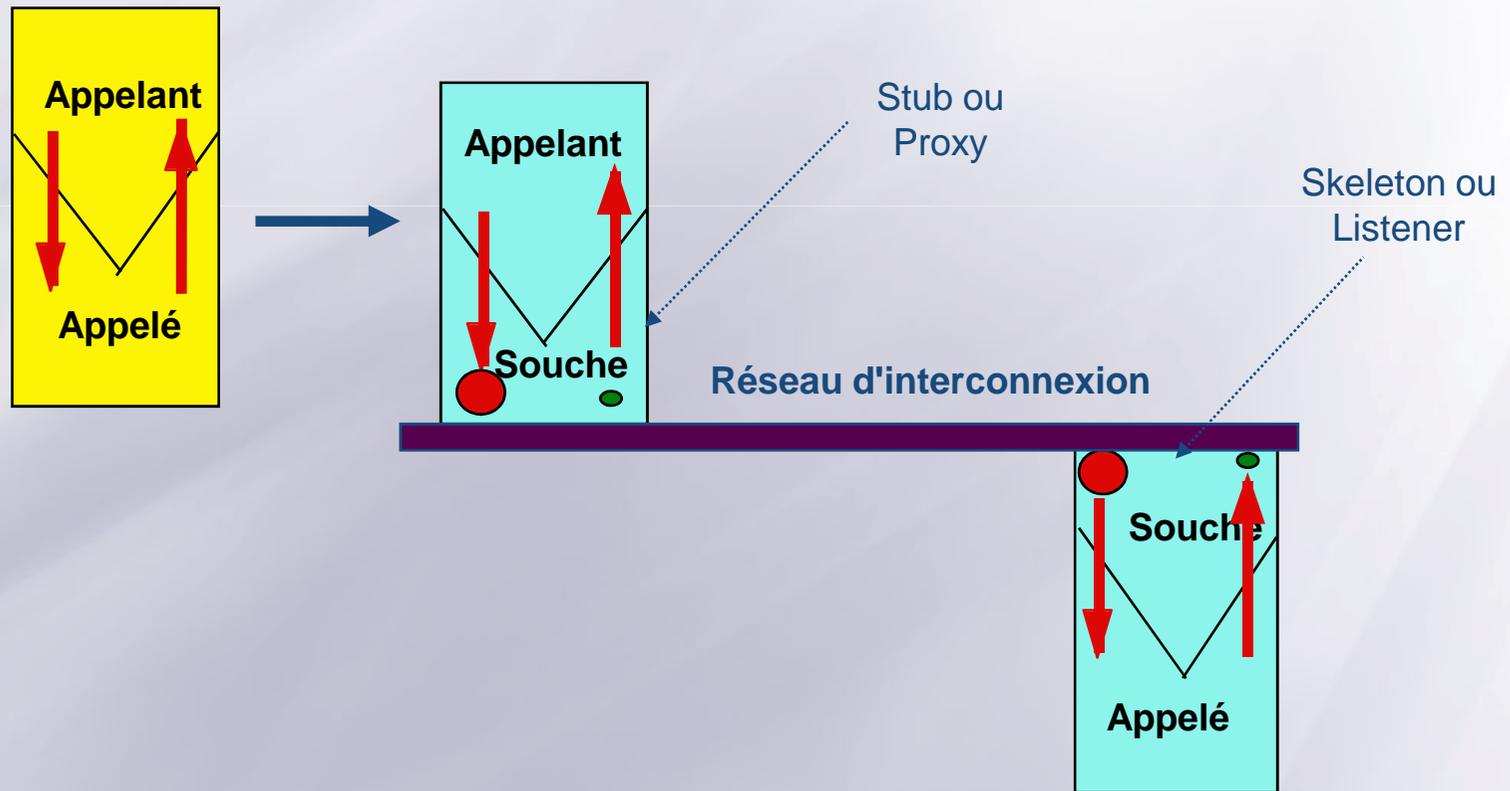
**News Feeds**

- UDDI News

# 3. SOAP

☰ Inspiré du RPC DCE adapté à XML

● Appel de procédure distante sur Internet



# Qu'est-ce-que SOAP ?

## ☰ Simple Object Access Protocol

- Ni simple, ni objet ...

## ☰ Invocations de fonctions en XML au-dessus de HTTP :

- Pour accéder services, objets et serveurs
- Indépendant de toute plate-forme

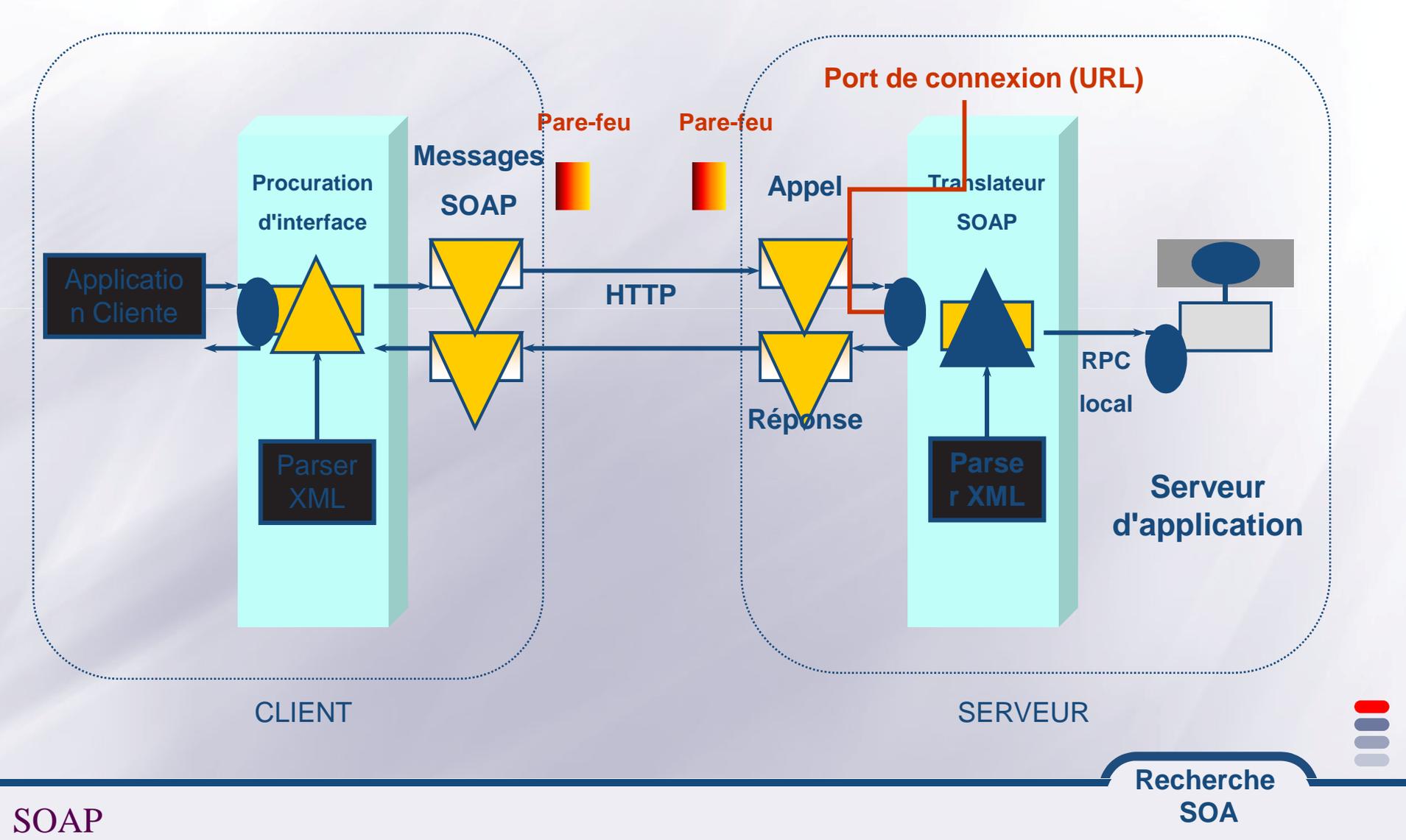
## ☰ Codage universel XML par opposition aux spécifiques :

- DCOM - DCE/NDR
- CORBA - IIOP/CDR
- JAVA - RMI/JRMP
- RDA - XDR

## ☰ Résolution des problèmes de session, sécurité, transaction ...

## ☰ Associé à WSDL pour publication d'interfaces

# Un échange type



# Éléments d'un message

## Envelope

- Élément pouvant contenir des déclarations d'espaces de noms ou des sous-éléments

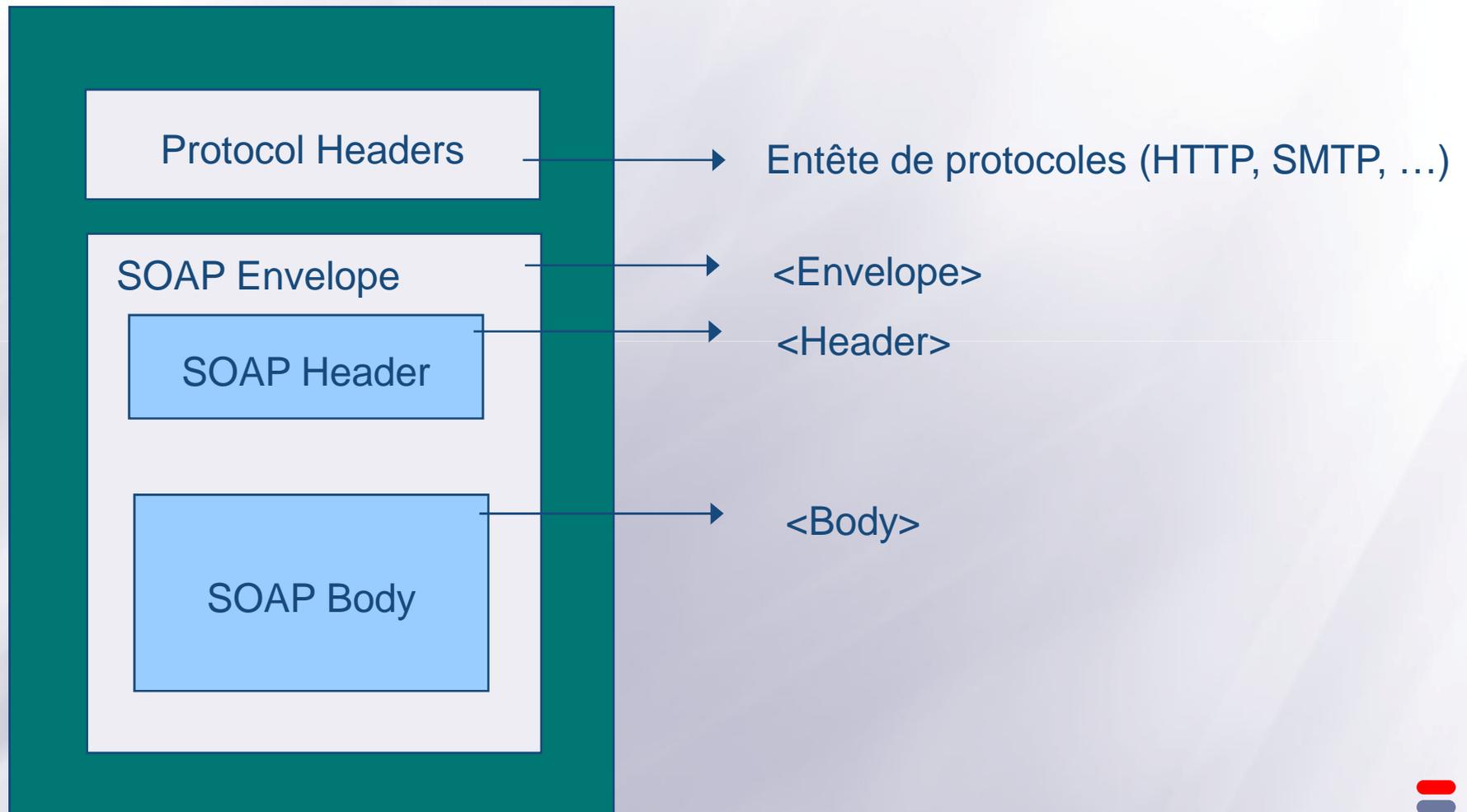
## Header

- Élément optionnel fils de Envelope
- Permet des extensions telles que authentification, session, etc.

## Body

- Élément obligatoire fils de Envelope
- Définit la méthode appelée, contient les paramètres
- Peut contenir un élément Fault en cas d'erreur

# Structure d'un message

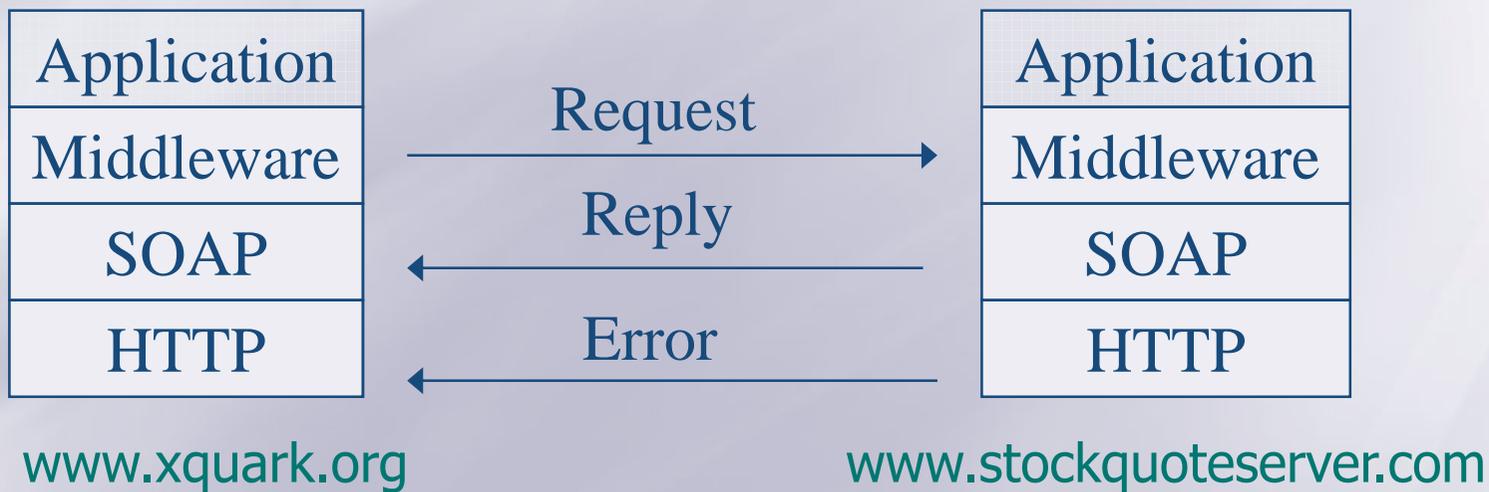


# Exemple

 [www.stockquoteserver.com](http://www.stockquoteserver.com)

 float GetLastTradePrice (Symbol)

 Le dialogue :



# La requête

**POST /StockQuote HTTP/1.1**

**Host: www.stockquoteserver.com**

Standard HTTP

**Content-Type: text/xml; charset="utf-8"**

**Content-Length: nnnn**

**SOAPAction: "Some-URI#GetLastTradePrice"**

```
<SOAP:Envelope
xmlns:SOAP="http://schemas.xmlsoap.org/soap">
  <SOAP:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP:Body>
</SOAP:Envelope>
```

# La réponse

**HTTP/1.1 200 OK**

**Content-Type: text/xml; charset="utf-8"**

Standard HTTP

**Content-Length: nnnn**

```
<SOAP:Envelope
xmlns:SOAP="http://schemas.xmlsoap.org/soap"/>
  <SOAP:Body>
    <m:GetLastTradePriceResponse
xmlns:m="Some-URI">
      <Price>34.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP:Body>
</SOAP:Envelope>
```

# Bilan SOAP

## ☰ Protocole d'invocation de services Web ☰ Questions ?

- Lisible et extensible
- Au-dessus de HTTP
- Description en WSDL
- Possibilité d'attachements MIME (PDF, GIF, BIN, ...)
- Pousser par Microsoft, IBM, BEA, ...

## ☰ CORBA ou DCOM « killer » ?

- Associé à nomination des objets
- Véritable échange de messages entre objets
- Performance ?



# Plan

1. Introduction aux architectures orientées services (SOA)
2. Travaux de recherche liés aux SOA
3. Sémantique dans les SOA
4. L'adaptation au contexte dans les SOA

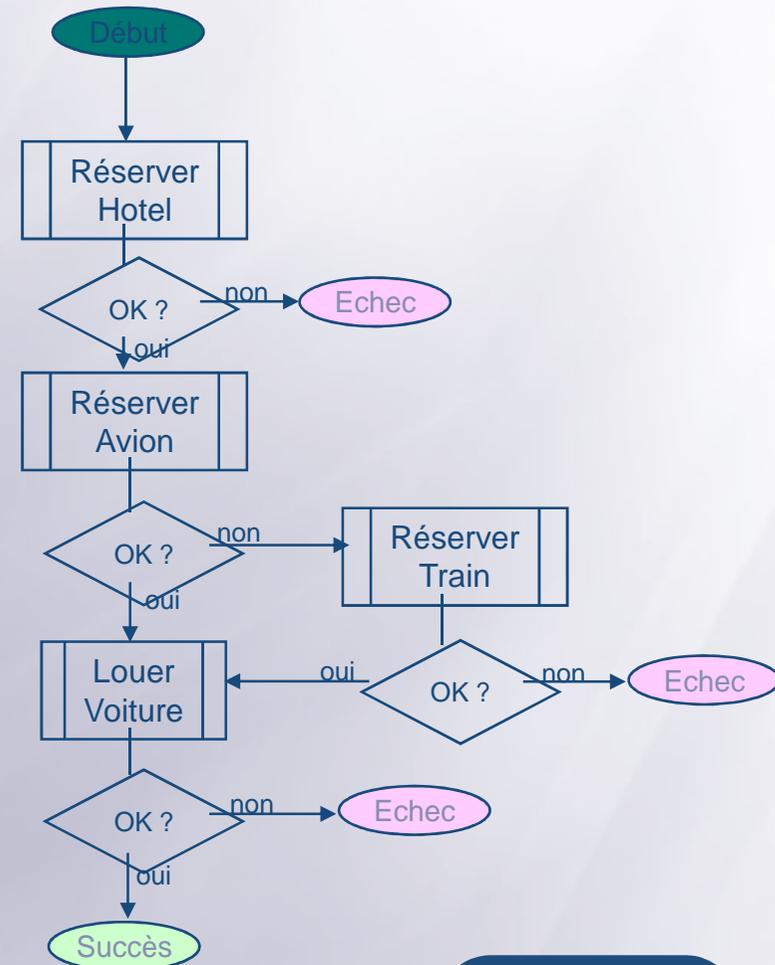


# Orchestration de Services Web

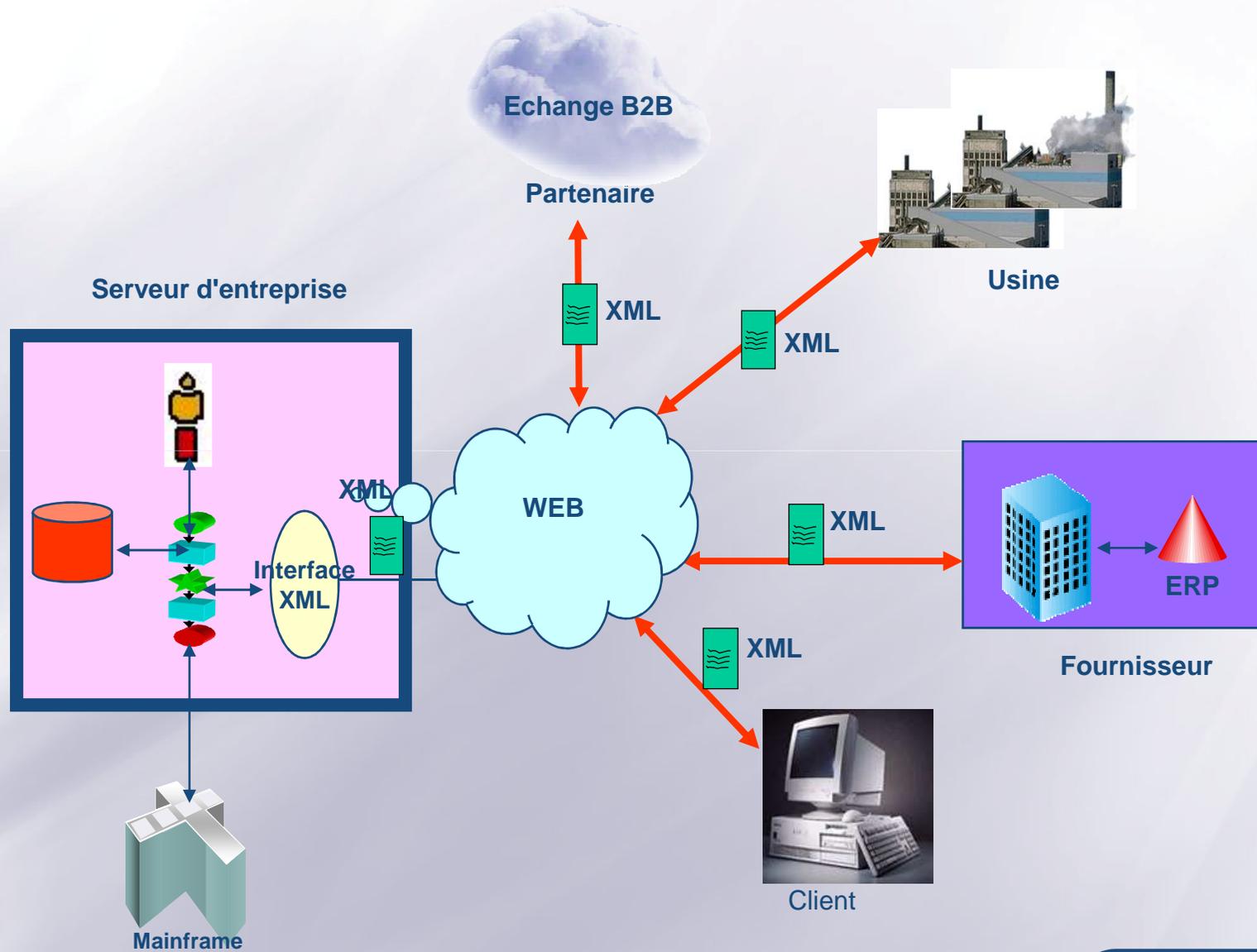
## Objectifs

- Modéliser des processus métiers (business process)
- Composer des services Web distribués
- Piloter l'exécution
  - Orchestration d'activités
  - Echanges XML
  - Gestion de transactions
- Business Process Management
  - Workflow
  - Transaction et Saga

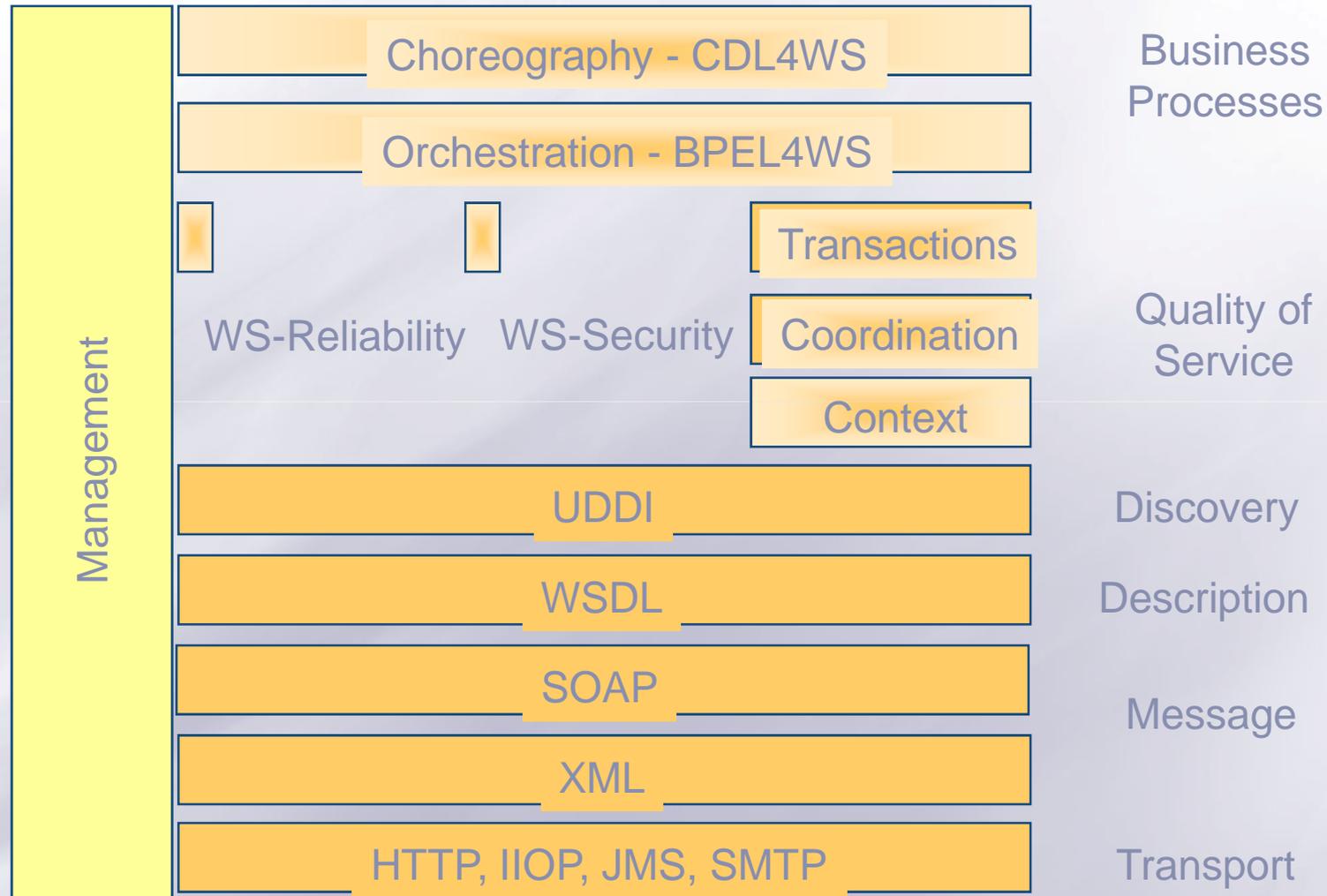
## Exemple : réservation



# Exemple : Pilotage Fabrication



# WEB Services et standards associés



# Composition de services

## Objectifs:

- Alliances entre business pour offrir des services intégrés à valeur ajoutée en combinant des services existants
- Réutilisation et extension de services existants
- Support pour la planification, la définition et l'implémentation de services composés
- Développement d'applications distribuées composées de services web

# Quelques définitions

## ☰ Processus métier (Business Process)

- Module fonctionnel réalisé par enchaînement d'activités business exécutées par des acteurs échangeant des messages et implémentant des objets et règles spécifiques à une entreprise.

## ☰ Orchestration d'activité (Activity Orchestration)

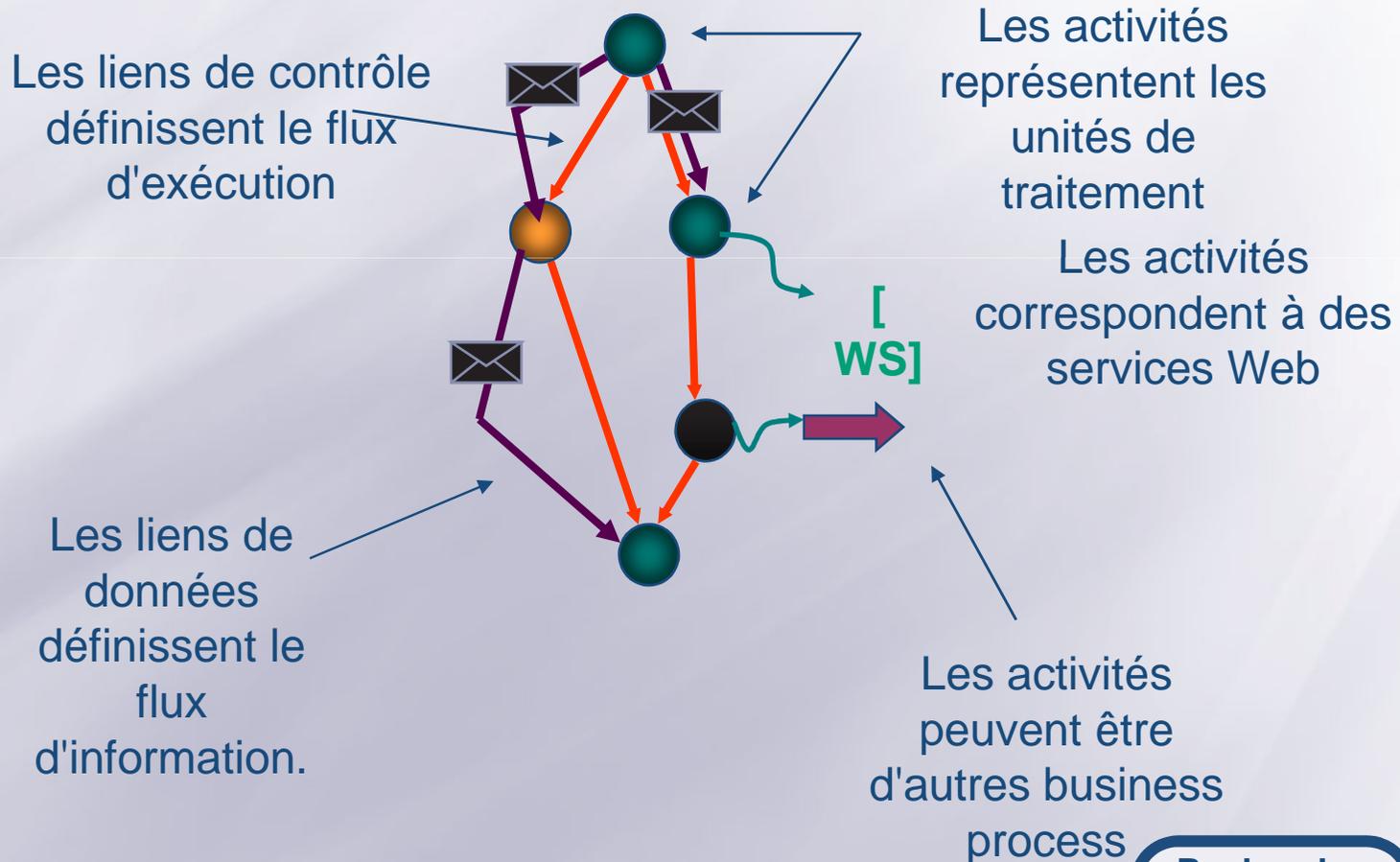
- Mécanisme d'invocation, de contrôle et de coordination des activités participant à la réalisation de processus métier.

## ☰ Composition de services (Services Composition)

- Techniques permettant d'assembler des services Web pour réaliser des processus métiers par des primitives de contrôles (boucles, tests, traitement d'exception, etc.) et d'échanges (envoi et réception de messages).

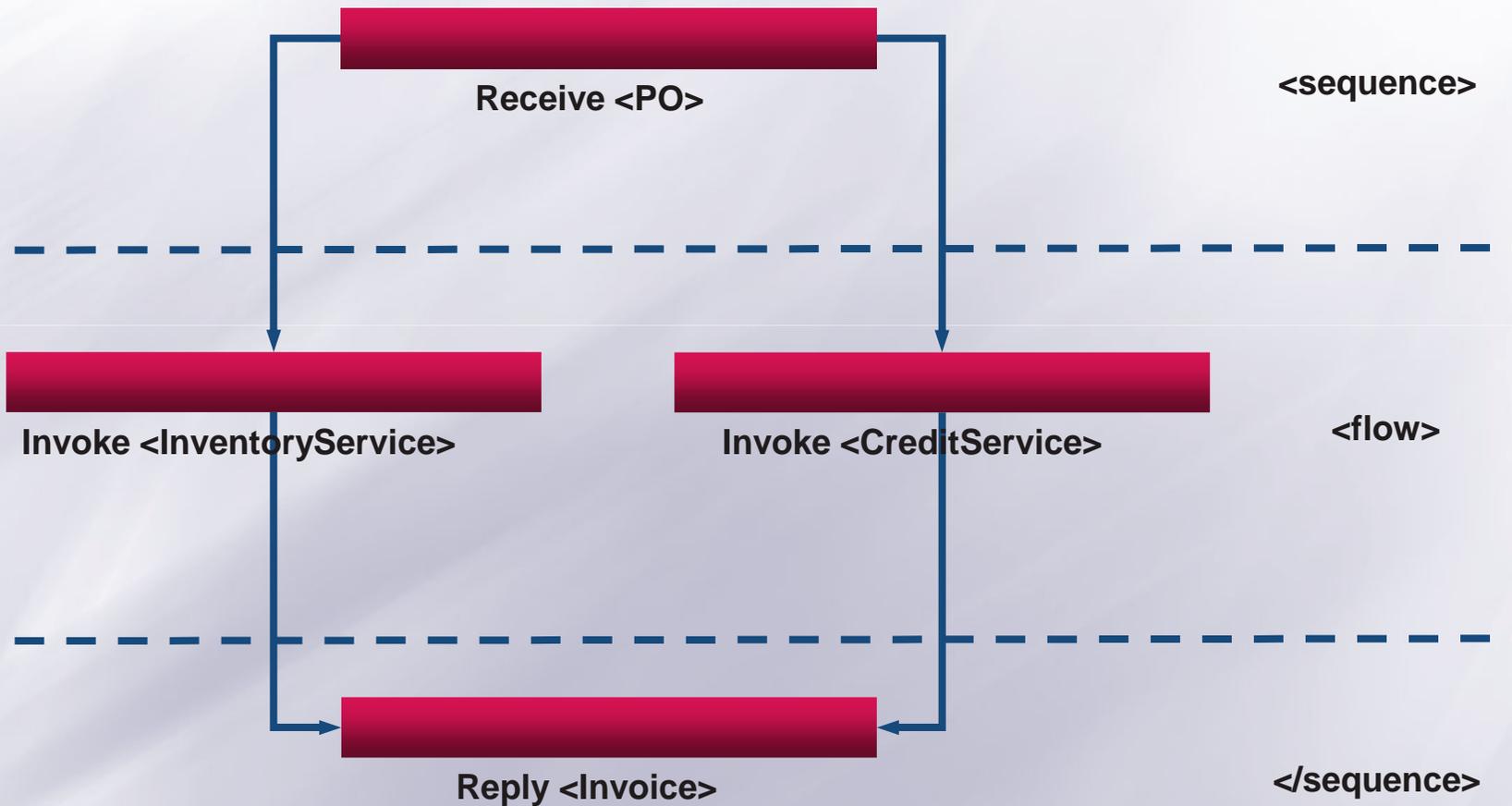
# Modélisation par Workflow

## Graphe acyclique d'activités modélisant un processus métier

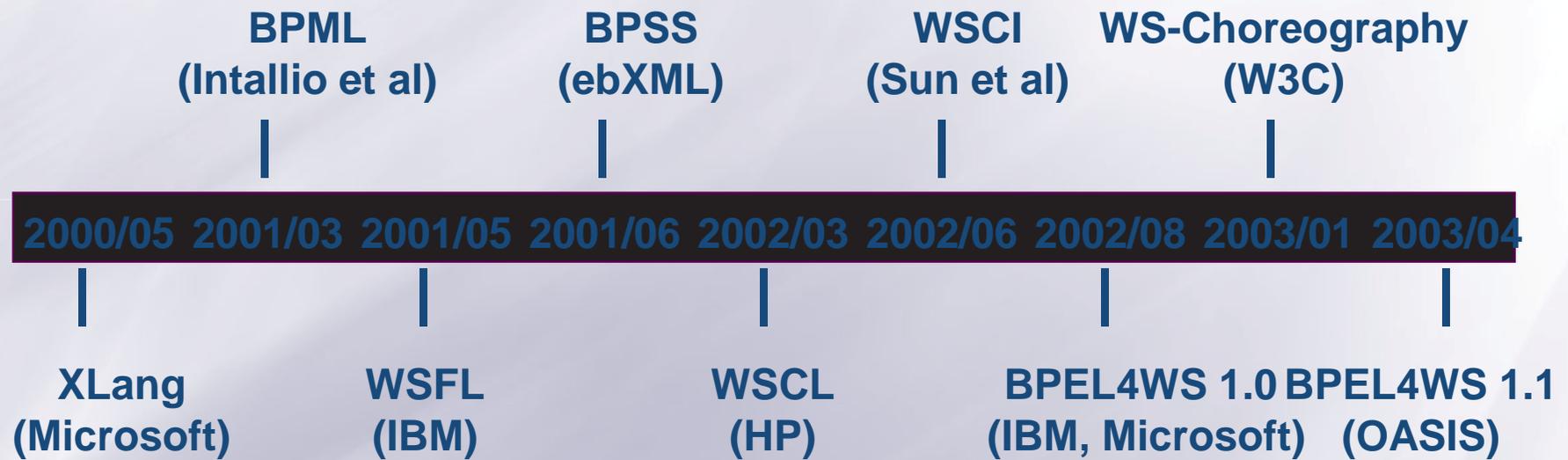


# Composition d'activités simples et complexes

## Description du flux d'activité en XML



# Vers un standard ?



# BPEL: Structure d'un Business Process

```
<process>
  <!-- Definition and roles of participants -->
  <partnerLinks> ... </partnerLinks>
  <!-- Données utilisées dans le process -->
  <variables> ... </variables>
  <!-- Propriétés permettant les conversations -->
  <correlationSets> ... </correlationSets>
  <!-- Gestion des exception -->
  <faultHandlers> ... </faultHandlers>
  <!-- Traitement des erreurs -->
  <compensationHandlers> ... </compensationHandlers>
  <!-- Gestion des compensations -->
  <eventHandlers> ... </eventHandlers>
  <!-- Workflow d'activités -->
  (activités)*
</process>
```



# BPEL: Exemple

```
<sequence>
  <receive partnerLink="customer" portType="Ins:purchaseOrderPT"
    operation="sendPurchaseOrder" variable="PO"
    createInstance="yes" />
  <flow>
    <invoke partnerLink="inventoryChecker" portType="Ins:inventoryPT"
      operation="checkINV" inputVariable="inventoryRequest"
      outputVariable="inventoryResponse" />
    <invoke partnerLink="creditChecker" portType="Ins:creditPT"
      operation="checkCRED" inputVariable="creditRequest"
      outputVariable="creditResponse" />
  </flow>
  ...
  <reply partnerLink="customer" portType="Ins:purchaseOrderPT"
    operation="sendPurchaseOrder" variable="invoice"/>
</sequence>
```



## Nécessité de fiabiliser:

- Les messages (WS-Reliability)
  - Garantie la livraison, les non dupliqués et l'ordre
  - S'ajoute comme un en-tête SOAP
  - Standard OASIS
- Les activités (WS-Transactions)
  - Courtes (Atomic Transactions)
  - Longues (Business Activities)

## Nécessité de sécuriser:

- Les échanges confidentiels (Standard WS-Security)
  - PKI, Signature et Cryptage

# Bilan Processus Métiers

## ☰ De multiples propositions

- un domaine chaud

## ☰ BPEL s'impose comme un standard de l'OASIS

## ☰ Vers trois standards pour la qualité de services

- WS-Reliable Messaging
- WS-Transactions
- WS-Security

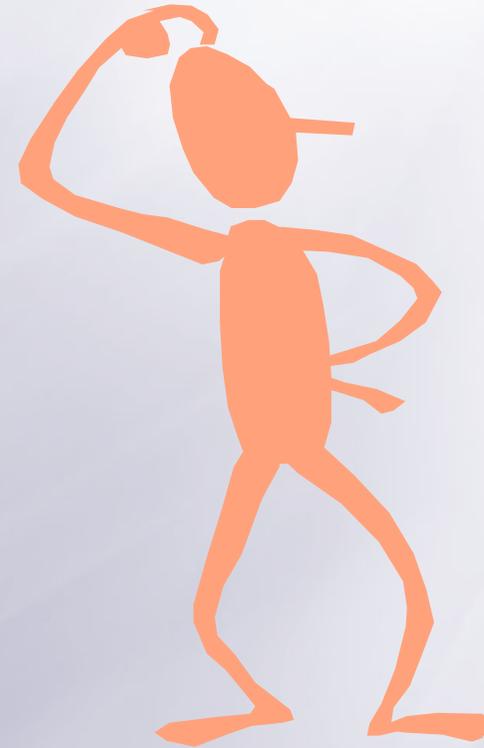
## ☰ Questions ?



# En résumé ...

- Invocation dynamique de services WEB décrits en WSDL
- SOAP est le protocole d'invocation (WP) sur HTTP
- Intégrité et typage des données (schémas XML)
- Possibilité de découvrir dynamiquement les services (UDDI)
- Lisibilité et sécurité renforcée (standard de cryptage)
- Indépendance des constructeurs (W3C)
- Intégration des workflows (BPEL)
- Ajout de contrôle qualité (transaction, sécurité)

■ Questions ?



# C'est beau tout ça mais il y a encore beaucoup de travail à faire

- On veut des services fiables...
- Comment mesurer cette fiabilité?
- Comment mesurer la qualité d'un service (simple et composite) d'une façon générale?
- Comment gérer la qualité de service?
- Qu'est ce qu'on doit faire si la QoS se dégrade?
- Comment établir des contrats de qualité de services
- Comment réaliser une composition automatique (dynamique) de services?
- Est-ce que la composition générée est pertinente?

- On n'a pas des vraies solutions à ces problèmes



- Manque de stratégies et de moyens qui permettent de guider la composition de services et la gestion de leurs qualités
- Manque de sémantique dans les stratégies et les modèles existants

## Quelques solutions proposées pour les problèmes évoqués

- Etude d'actions qui permettent d'améliorer la QoS
- Etude de moyens qui permettent de mesurer la QoS
- Stratégies d'adaptation aux changements de QoS
- Utilisation des ontologies dans les modèles de QoS
  - Injecter de la sémantique dans les SOA

# Plan

1. Introduction aux architectures orientées services (SOA)
2. Travaux de recherche liés aux SOA
3. **Sémantique dans les SOA**
4. L'adaptation au contexte dans les SOA



# Ontologie : représentation du sens

Etude de ce qui est  
terme emprunté à la philosophie

Pour nous : *formalisation d'une conceptualisation*



# Ontologie

Se mettre d'accord sur le *sens* des termes employés dans une organisation, une communauté, un métier

Faire en sorte que les personnes et les logiciels *se comprennent*

Utile pour des applications distribuées telles que le Web



# Pourquoi représenter le sens?

‘chambre’ :

Chambre d'hôtel ?

Chambre d'écho ?

Chambre des députés ?

Chambre d'enregistrement ?



# Ontologie

- Identifier, modéliser les concepts d'un domaine, pertinents pour une/des applications
- Se mettre d'accord, au sein d'une communauté, sur les *termes* employés pour se référer à ces concepts
- Identifier/modéliser les concepts et les relations conceptuelles
- formaliser la conceptualisation, et le vocabulaire correspondant
- Formalisation pour lever toute ambiguïté

# Connaissance

## ☰ Continuum : donnée, information, connaissance

● donnée : `hasState(fireDetector, off)`

● information : **SOS**

● connaissance : en cas d'alerte,  
déclencher les secours

☰ La connaissance permet de produire de nouvelles données, informations, connaissances : inférence



# Types de connaissances

## ☰ Connaissances de résolution de problème :

- conception, diagnostic, évaluation, planification
- tâches, inférences

## ☰ Connaissances du domaine

- e.g. électronique, mécanique, médecine, etc.
  - ontologie : *réutilisable*
  - modèles de domaine : fonctionnel, causal, structurel, exprimés dans le vocabulaire de l'ontologie.



# Structure d'ontologie

Une structure d'ontologie est un quintuplet  $O := \{C, \mathcal{R}, \mathcal{H}^C, \text{rel}, \mathcal{A}^O\}$

- $C$  et  $\mathcal{R}$ : ensembles disjoints des concepts et des relations
- $\mathcal{H}^C$  hiérarchie (taxonomie) de concepts :  $\mathcal{H}^C \subseteq C \times C$ ,  $\mathcal{H}^C(C_1, C_2)$  signifie que  $C_1$  est un sous-concept de  $C_2$  (relation orientée)
- $\text{rel}$ : relation  $\text{rel}: \mathcal{R} \rightarrow C \times C$  (définit des relations sémantiques non taxonomiques) avec 2 fonctions associées
  - $\text{dom}: \mathcal{R} \rightarrow C$  avec  $\text{dom}(\mathcal{R}) := \Pi 1(\text{rel}(\mathcal{R}))$ : source de la relation
  - $\text{range}: \mathcal{R} \rightarrow C$  avec  $\text{range}(\mathcal{R}) := \Pi 2(\text{rel}(\mathcal{R}))$ : destination de la relation
  - $\text{rel}(\mathcal{R}) = (C_1, C_2)$  s'écrit aussi  $\mathcal{R}(C_1, C_2)$
- $\mathcal{A}^O$ : ensemble d'axiomes, exprimés dans un langage logique adapté (logique de description, logique du 1er ordre)

Dans l'ontologie on trouve aussi les instances de concepts et des relations

# Modèles de domaine

## Modèle :

Concept : Engine

Concept : Vehicule

ObjectProperty: isPartOf(Engine, Vehicule)

DataTypeProperty: hasState(Engine, String)

DataTypeProperty: hasState(Vehicule, String)

## Instance :

instanceOf(Engine, "1290 45 333")

instanceOf(Vehicule, "WDM 203 1290 45 333")

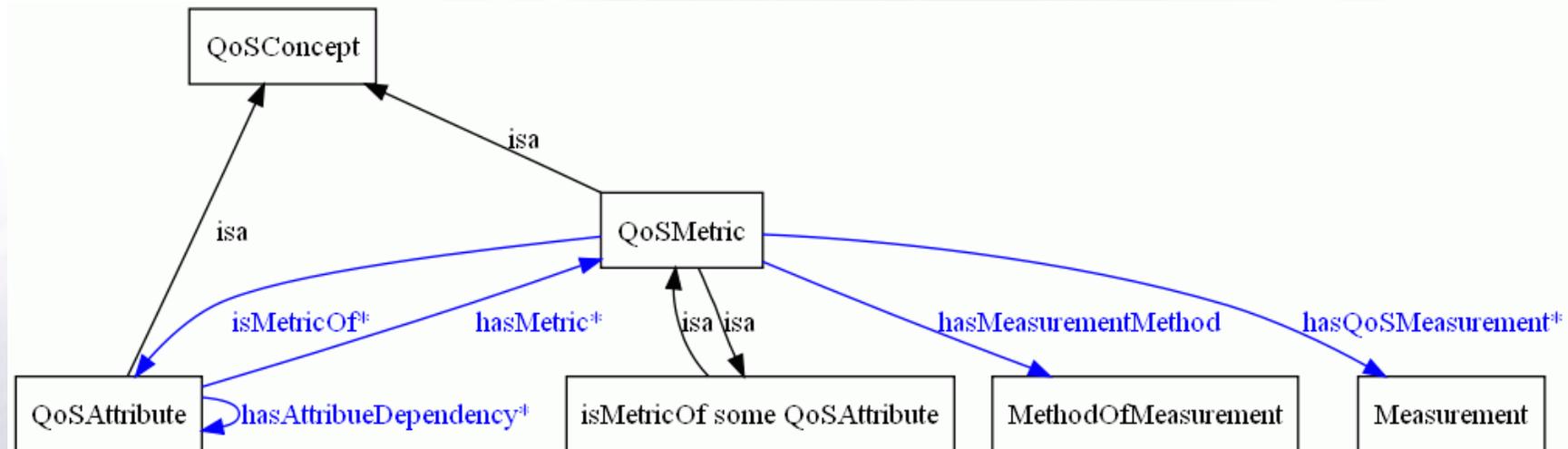
isPartOf("1290 45 333", "WDM 203 1290 45 333")

## Règles :

**hasState(?moteur, "BreakDown") ^ isPartOf(?moteur, ?voiture) →  
hasState(?voiture, "BreakDown")**



# Exemple d'une (partie d'une) ontologie de QoS



## QoSOnt: a QoS ontology for service-centric systems

Dobson, G.; Lock, R.; Sommerville, I.

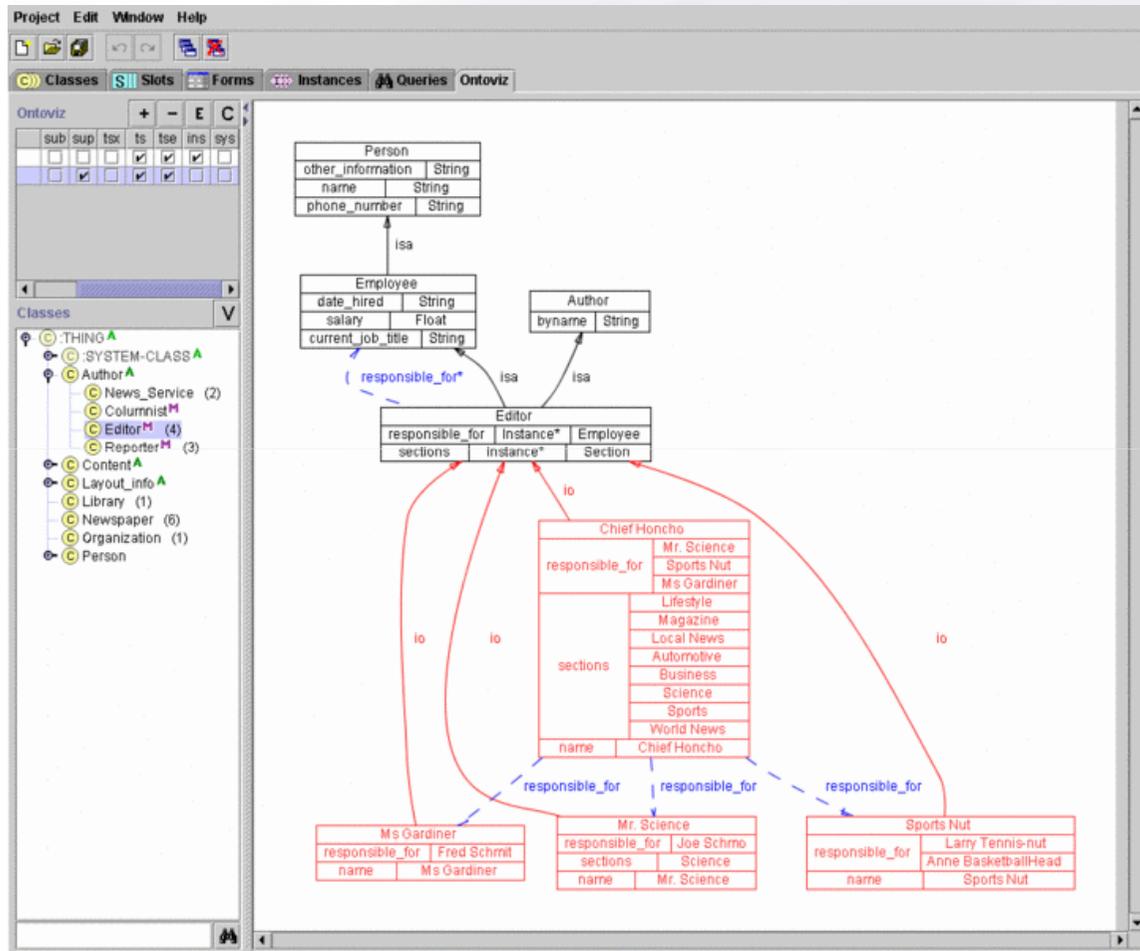
Software Engineering and Advanced Applications, 2005. 31st EUROMICRO  
Conference on

Volume , Issue , 30 Aug.-3 Sept. 2005 Page(s): 80 - 87

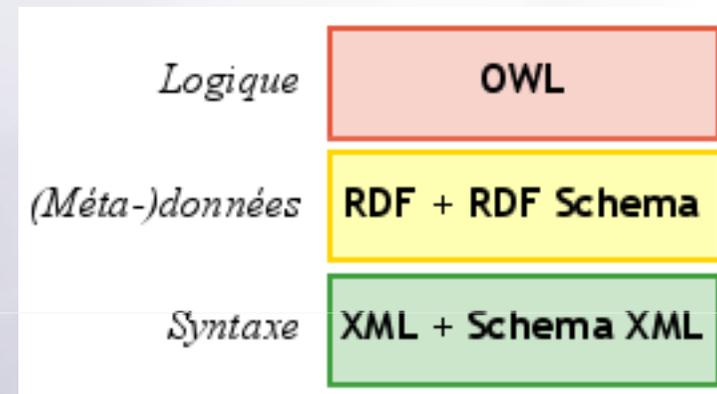
- Manque de sémantique dans la relation *hasAttributeDependency*
- il faut préciser quel type de dépendance

# Outils et langages d'ontologies

## Outils principal : Protégé



## Langages principaux



# L'ancienne vision du web



# Résolution des problèmes de WEB classique

## Serious Problems in

- information finding,
- information extracting,
- information representing,
- information interpreting and
- and information maintaining.

Static

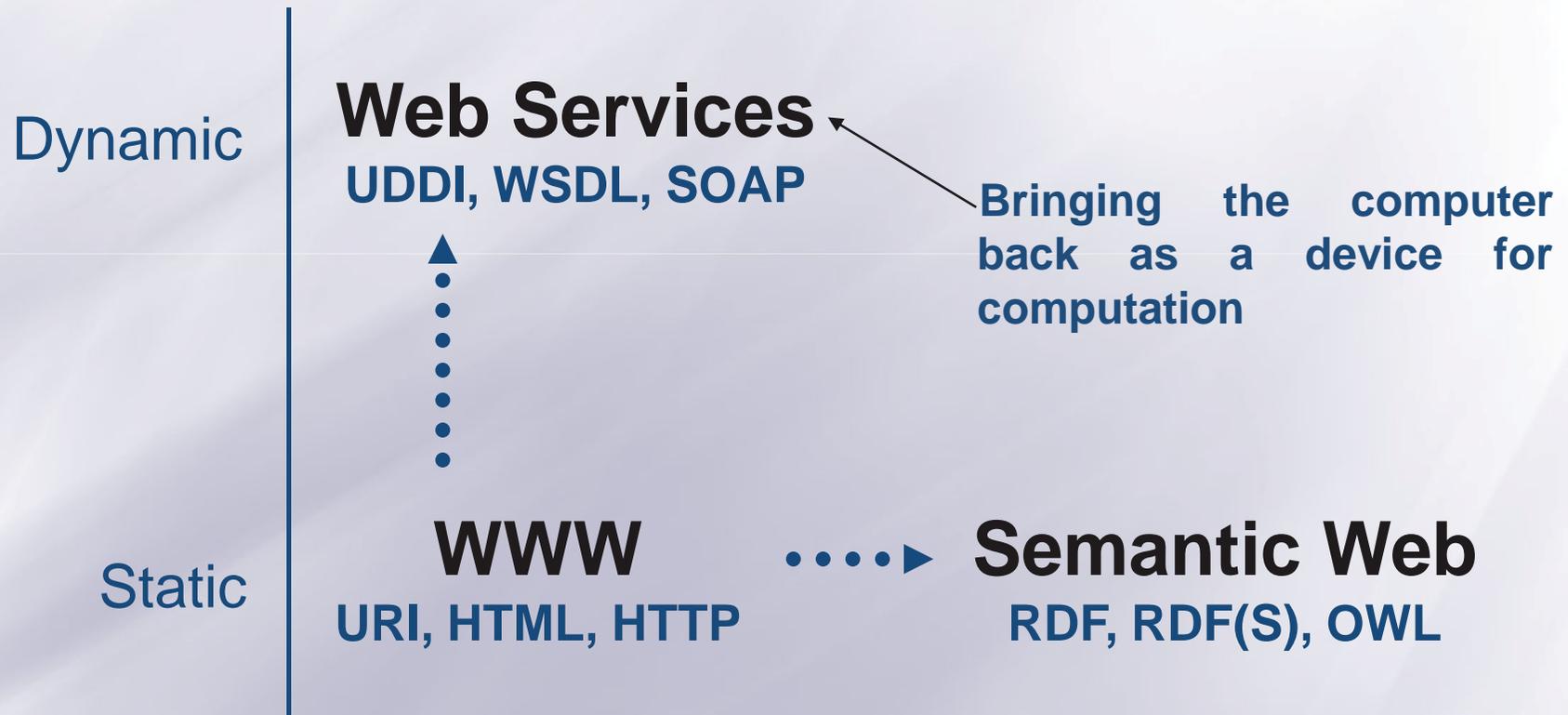
**WWW**  
URI, HTML, HTTP



**Semantic Web**  
RDF, RDF(S), OWL



# L'intégration du WEB dynamique



# La force des ontologies avec les WEB services

Bringing the web to its full potential

Dynamic

**Web Services**  
UDDI, WSDL, SOAP

**Semantic Web Services**

Static

**WWW**  
URI, HTML, HTTP

**Semantic Web**  
RDF, RDF(S), OWL



# Les ontologies des services web

	OWL-S	WSMO	<i>current Web Service technologies</i>
<b>Discovery</b> <i>detection of suitable WS</i>	Profile	Goals and Web Services (capability)	<i>UDDI API</i>
<b>Consumption &amp; Interaction</b> <i>How to consume &amp; aggregate</i>	Process Model	Service Interfaces (Choreography + Orchestration)	<i>BPEL4WS / WS-CDL</i>
<b>Invocation</b> <i>How to invoke</i>	Grounding+ WSDL/SOAP	Grounding (WSDL / SOAP, ontology-based)	<i>WSDL / SOAP</i>
<b>Mediation</b> <i>Heterogeneity handling</i>	-	Mediators	-



# Plan

1. Introduction aux architectures orientées services (SOA)
2. Travaux de recherche liés aux SOA
3. Sémantique dans les SOA
4. L'adaptation au contexte dans les SOA



# Context-awareness

## La créateur des systèmes pervasifs (M. Weiser)

- «The most profound technologies are those that disappear»

## Informatique context-aware

- Prise en compte de l'environnement dans lequel se trouve l'utilisateur
- Capture et accès automatique
- Délivrer l'information pertinente
  - Quand?
  - Où?
  - Comment?



# Définition du contexte

- **Créateur du domaine de context-awareness [Dey, 99]**
- **« Environmental information or context covers information that is part of an application's operating environment and that can be sensed by the application. This typically includes the location, identity, activity and state of people, groups and objects. »**



# Définition du contexte

## [Winograd 01]

 “Something is context because of the way it is used in interpretation, not due to its inherent properties. The voltage on the power lines is a context if there is some action by the user and/or computer whose interpretation is dependant on it, but otherwise is just part of the environment.”



# Définition du contexte

## Pertinence

- Le concepteur de l'application décide quelles informations contextuelles sont intéressantes
  - Dépend de l'application

## Le contexte

- est souvent constitué d'un ensemble d'informations implicites auxquelles l'application n'a pas accès
- rend les applications plus « smart »



# Contenu du contexte

## Classiquement, 4 axes

- Utilisateur
  - Profil et préférences, localisation...
- Terminaux et matériels
  - Taille de l'écran, résolution, couleurs, mémoire, API...
- Réseau
  - Bande passante, connectivité, Qos
- Méta-données de l'application
  - Codage, langue, versions

## De plus en plus souvent un 5<sup>ème</sup>

- Activité
  - Tâche en cours



# Modélisation du contexte

☰ Très souvent dépendante de l'application, encore peu de travaux « génériques » sur cette modélisation

☰ 3 approches principales

- Couples attribut/valeur
- Extension de CC/PP
- Modélisation par ontologies
  - Très en vogue – de très nombreux travaux en cours



# Modélisation du contexte par attribut/valeur

 **Contexte = paires (attribut, valeur)**

- **userName=tchaari**

- **userLocation = 8207**

 **Les paires sont indépendantes**

**+ Facilité d'implantation**

**- Cohérence de l'ensemble**

**- Sémantiquement pauvre**



# Modélisation du contexte avec CC/PP

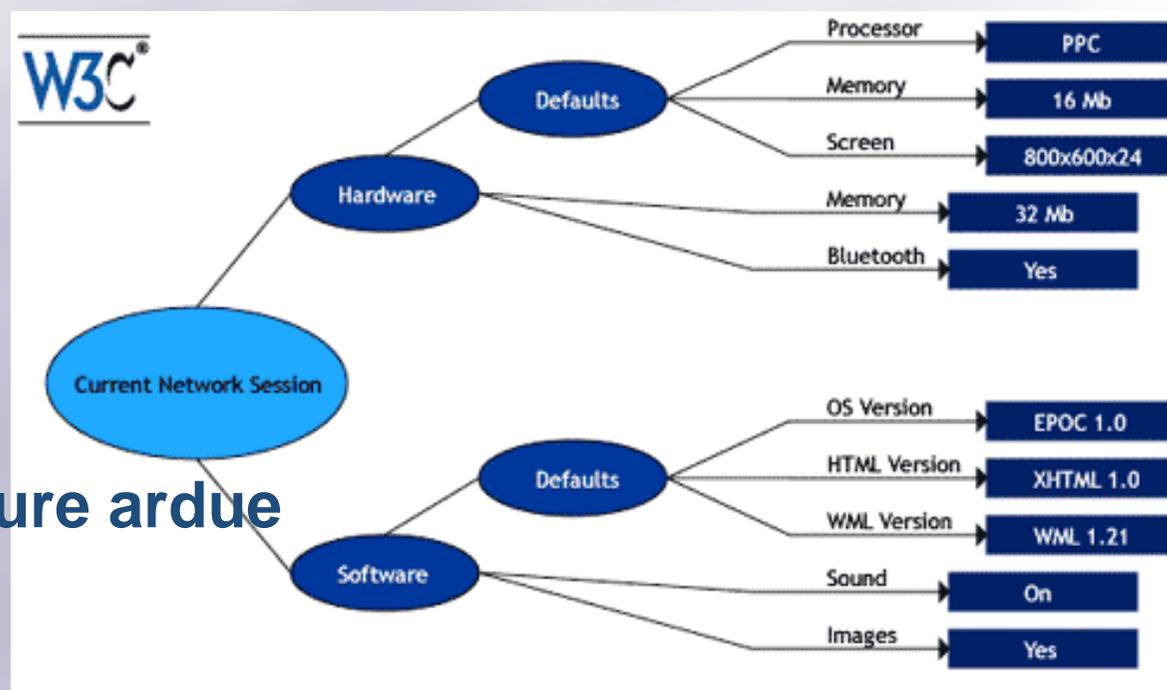
## Composite Capabilities / Preferences Profile (W3C)

- description des terminaux et des préférences utilisateur
- Document RDF pour décrire les attributs
- Modéliser le contexte = faire des extensions

+ standard

- extensions =>

complexité, lecture ardue



# Exemple de document XML CC/PP

```
<?xml version="1.0"?>  
<!-- Checked by SiRPAC 1.16, 18-Jan-2001 -->  
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
  xmlns:ccpp="http://www.w3.org/2000/07/04-ccpp#">  
  <rdf:Description rdf:about="HWDefault">  
    <rdf:type rdf:resource="HardwarePlatform" />  
    <display>320x200</display>  
    <memory>16Mb</memory>  
  </rdf:Description>  
</rdf:RDF>
```



# Modélisation du contexte par ontologies

- ☰ Une ontologie est un ensemble structuré de concepts. Les concepts sont organisés dans un graphe dont les relations peuvent être :
  - des relations sémantiques
  - des relations de composition et d'héritage (au sens objet)

- ☰ Intérêt des ontologies pour modéliser le contexte
  - Utilisables dans des environnements d'envergure
  - Sémantiquement riches
  - Inférences et gestion de conflicts



# Modélisation du contexte par ontologies

## Exemple CoOL [Strang & al. 03]

```
<instance xmlns=http://demo.heywow.com/schema/cool
xmlns:a=http://demo.heywow.com/schema/aspects
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <contextInformation>
    <entity system="urn:phonenumber">+49-179-1234567</entity>
    <characterizedBy>
      <aspect name="GaussKruegerCoordinate">
        <observedState
xsi:type="a:o2GaussKruegerType">367032533074</observedState>
        <units>10m</units>
      </aspect>
      <certaintyOfObserver>90</certaintyOfObserver>
    </characterizedBy>
  </contextInformation>
</instance>
```



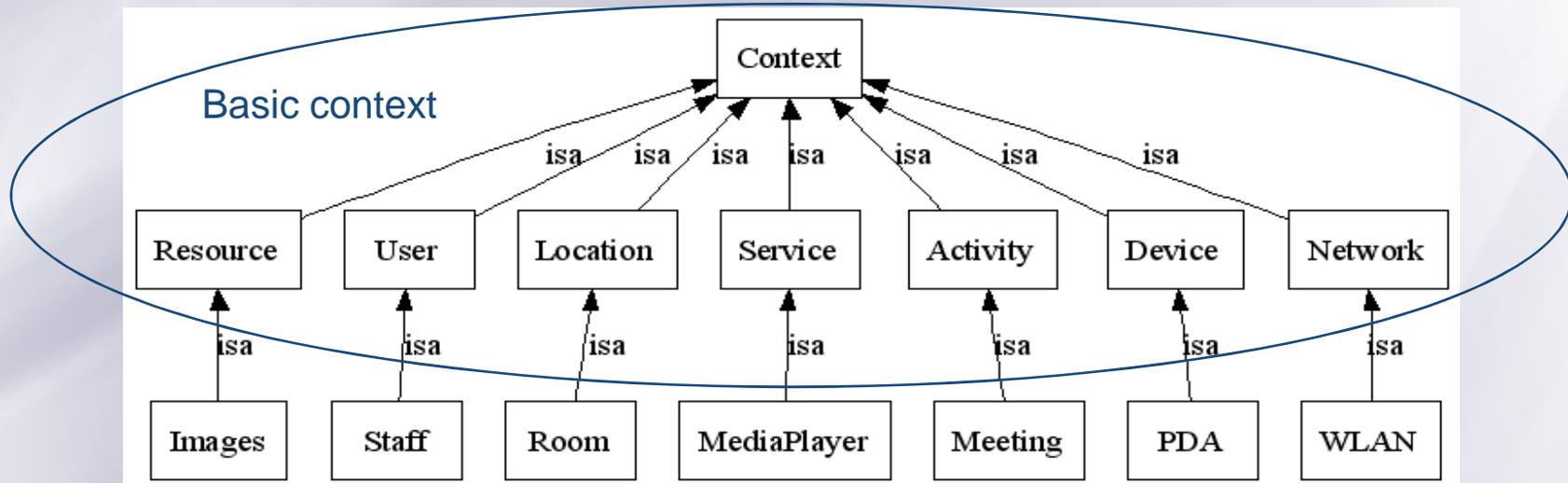
# Modélisation du contexte par ontologies

## Exemple Gcm [T. Chaari, D. Ejigu et al. 06]

### ☰ Descripteurs de contexte génériques / de base

- User context, device context, network context, activity context, service context, location context and resource context

### ☰ Descripteurs de contextes spécifiques à l'application



## ...Suite Gcm



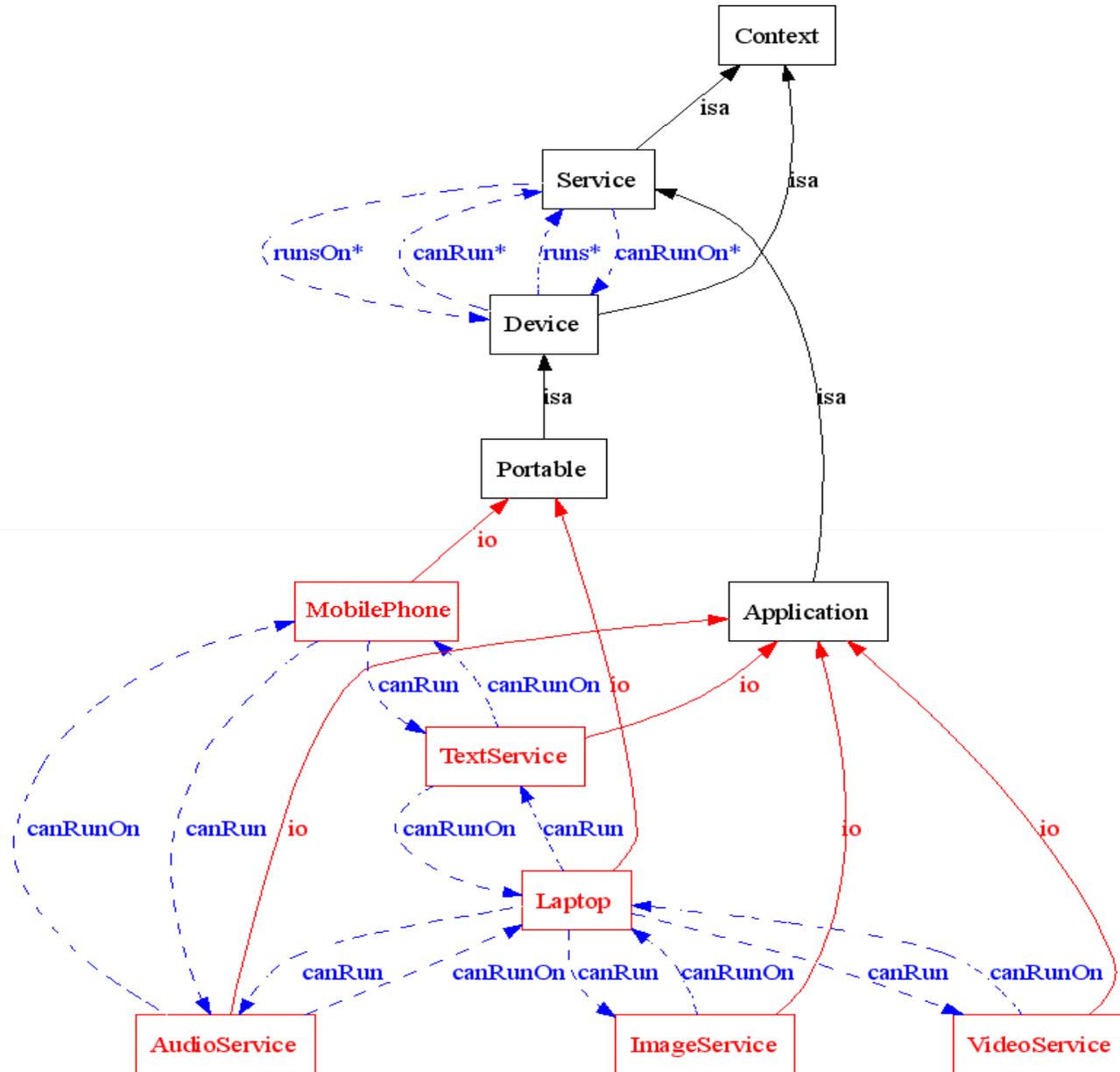
**Context element=(Subject, Predicate, Value, Time, Certainty)**

- **Subject = propriétaire de l'élément de contexte**
- **Predicate = propriété**
- **Value = valeur de l'élément**
- **Time = date du relevé de la valeur**
- **Certainty = confiance dans la valeur**

**Exemple : (VideoService, runsOn, PDA-01, 20061013, 0.9)**



# Exemple gcm



# Synthèse des modélisations de contextes

	Expressivité et richesse sémantique	Facilité d'implantation	Gestion des conflits
Paires Attribut / valeur	-	+	-
Extension de CC/PP	+	+	-
Ontologies	++	-	++

# Conception d'une application context-aware

- **Spécifications : Spécifier le contexte nécessaire**
- **Découverte : localiser les composants qui peuvent acquérir du contexte**
- **Séparation des problèmes : séparer l'acquisition du contexte de son utilisation**
- **Stockage : importance des historiques de contexte**
- **Interprétation : pour des informations de plus haut niveau**
- **Dissémination : Communications transparentes**
- **Disponibilité permanente du contexte**

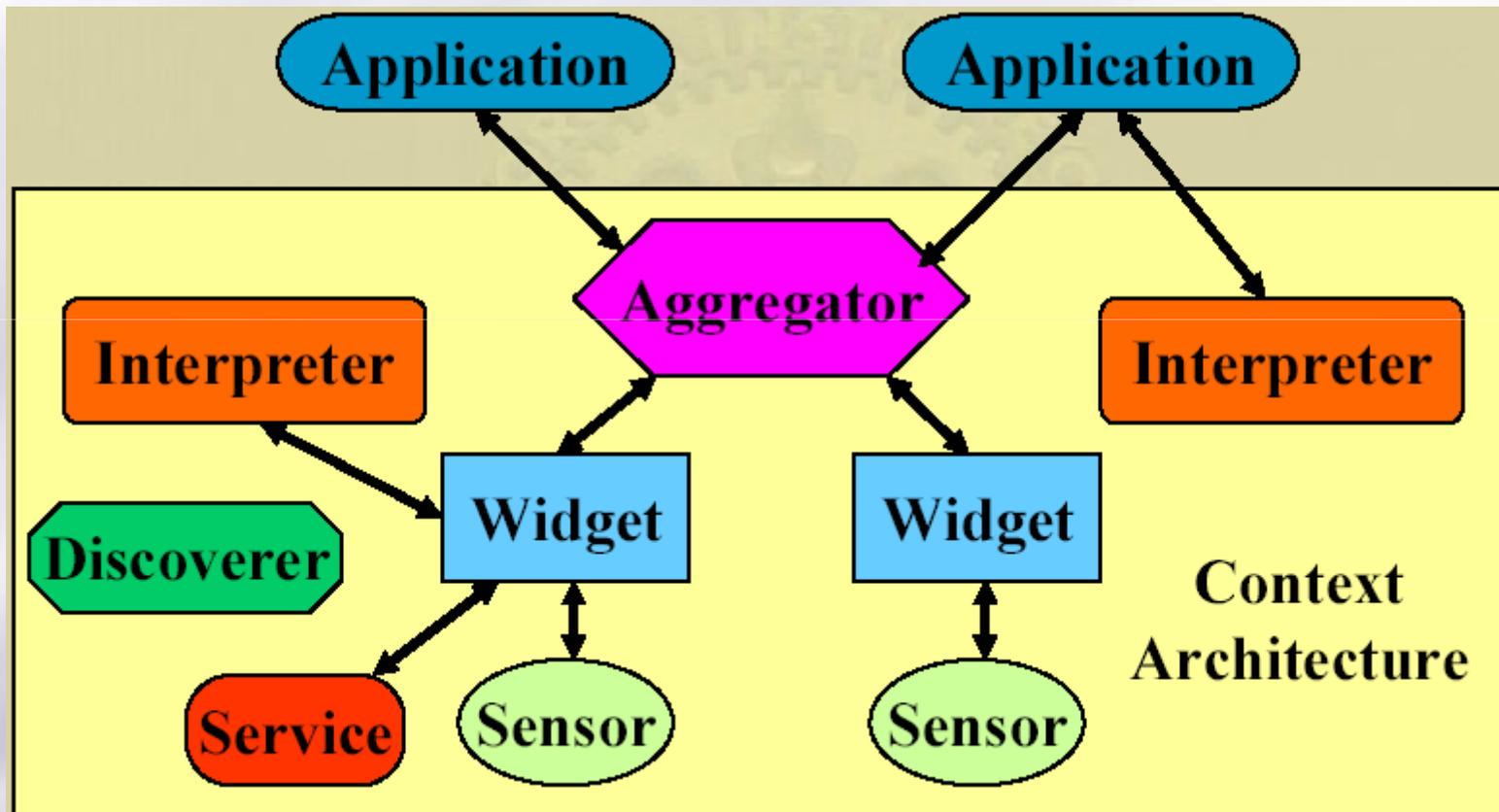


## Context Widgets

- Par analogie avec les widgets des interfaces graphiques
  - Encapsulation, abstraction
- But
  - Acquérir et abstraire les données des capteurs
  - Séparer les problématiques
  - Stockage



# Architecture de Dey



# Application context-aware

☰ « Context-aware applications sense context information and modify their behaviour accordingly without necessary explicit user intervention »

☰ Comment modifier le comportement de l'application?

- Quelles sont les actions à prendre?
- Les appliquer sur quels composants?
- Quand et comment les appliquer?



# Adaptation

## Principes génériques d'adaptation

- Statique
- Dynamique

## 3 domaines principaux

- Adaptation des interfaces utilisateurs
- Adaptation du contenu
- Adaptation des services



# Adaptation statique

- Préparer plusieurs versions d'une ressource avant son exploitation
  - En fonctionnement, adaptation = choix de la version correspondant au contexte
  - Beaucoup utilisée dans les débuts des applications multi-terminaux
    - Version existante pour terminaux standards
    - Nouvelles versions pour autres terminaux
- + simplicité et efficacité de fonctionnement
- pb d'échelle, lourdeur de prise en compte d'une nouvelle version

# Adaptation dynamique

☰ Transformations sur la ressource en cours de fonctionnement

☰ Exemples

- CSS : transformations à la volée de la forme d'un document XML=>HTML

- transformation de format de données

+ gain de temps pour prendre en compte une nouvelle version

- complexité de la réalisation

- temps de réponse



# Trois domaines principaux d'adaptation

 **Adaptation des interfaces utilisateurs**

 **Adaptation du contenu**

 **Adaptation des services**



# Adaptation des interfaces utilisateurs



# Adaptation des interfaces utilisateur

## Approches cognitives

- Modèles utilisateur/tâche/dialogue
- Généralement instrumentés par des plates-formes de génération automatique ou semi-automatique

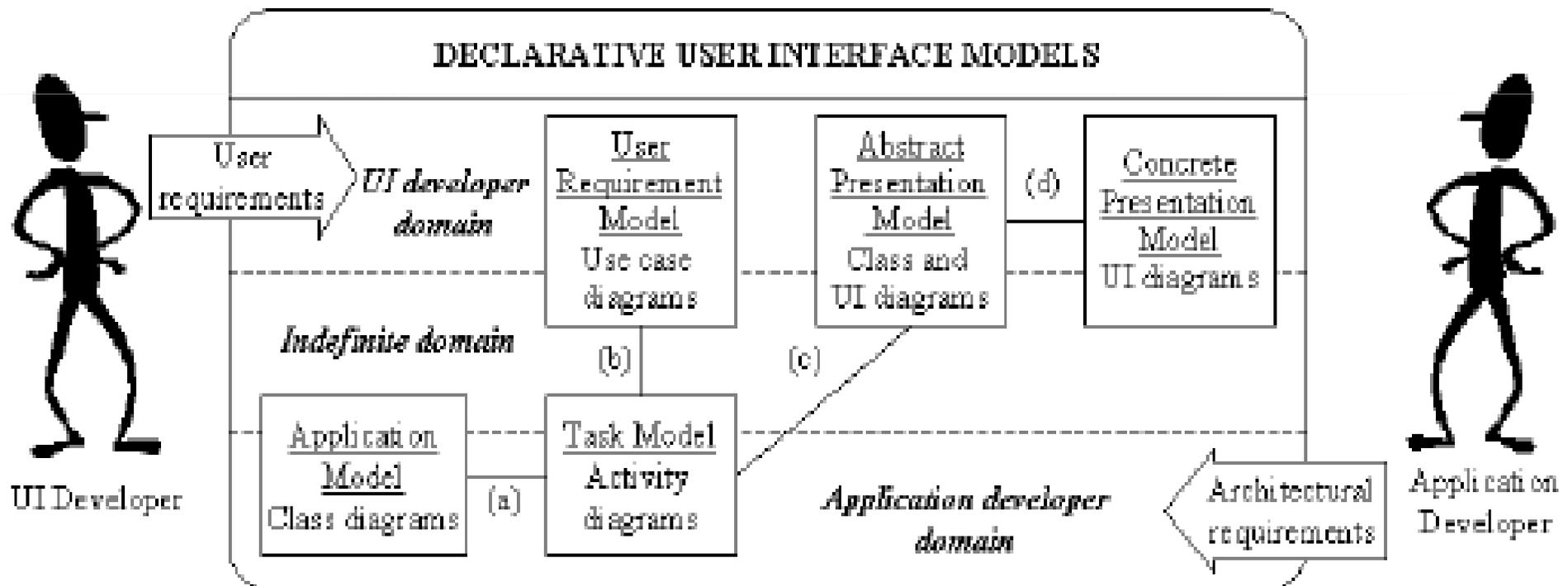
## Approches d'ingénierie

- Outils de conception et développement des IHM
- Question : passer d'une problématique de visualisation/mise à jour à une solution concrète (code)
- Généralement basé sur des modèles
  - À la XML : UIML, AUIML, SunML, XIML, Plastic ML, USIXML, XUL, XAML...
  - À la UML : UMLi



# Modèle UMLi

- [Pinheiro da Silva et Paton, 2000 ]
- Extension de UML: aspect interaction
- Diagrammes complexes et fastidieux



# Plates-formes de génération de code d'IU

- **Fondées sur plus ou moins de modèles, parmi:**
  - **Domaine** : description des services de l'application, modèle de BD...
  - **Tâches** : graphes d'activités
  - **Utilisateur** : préférences
  - **Dialogue** : partie interactive de l'interface
  - **Présentation** : partie visuelle de l'interface
- **Une génération plus ou moins automatique du code**
- **Des modèles inter-reliés, long, complexes et fastidieux... il est souvent plus rapide de coder soi-même l'interface!**

# Le projet SEFAGI

## [Chaari 2004]

### Cadre applicatif

- projet Système d'Information COMmunicant pour la santé
- Adaptation des interfaces utilisateurs au terminal

### Interface graphique du Projet SICOM

- Une partie générique
- Des parties spécifiques
  - dédiées à une pathologie
  - dédiées à un type d'utilisateurs
    - ==> Nombre de fenêtres différentes élevé
    - ==> Temps de développement important



# Objectifs

## Génération automatique des fenêtres spécifiques

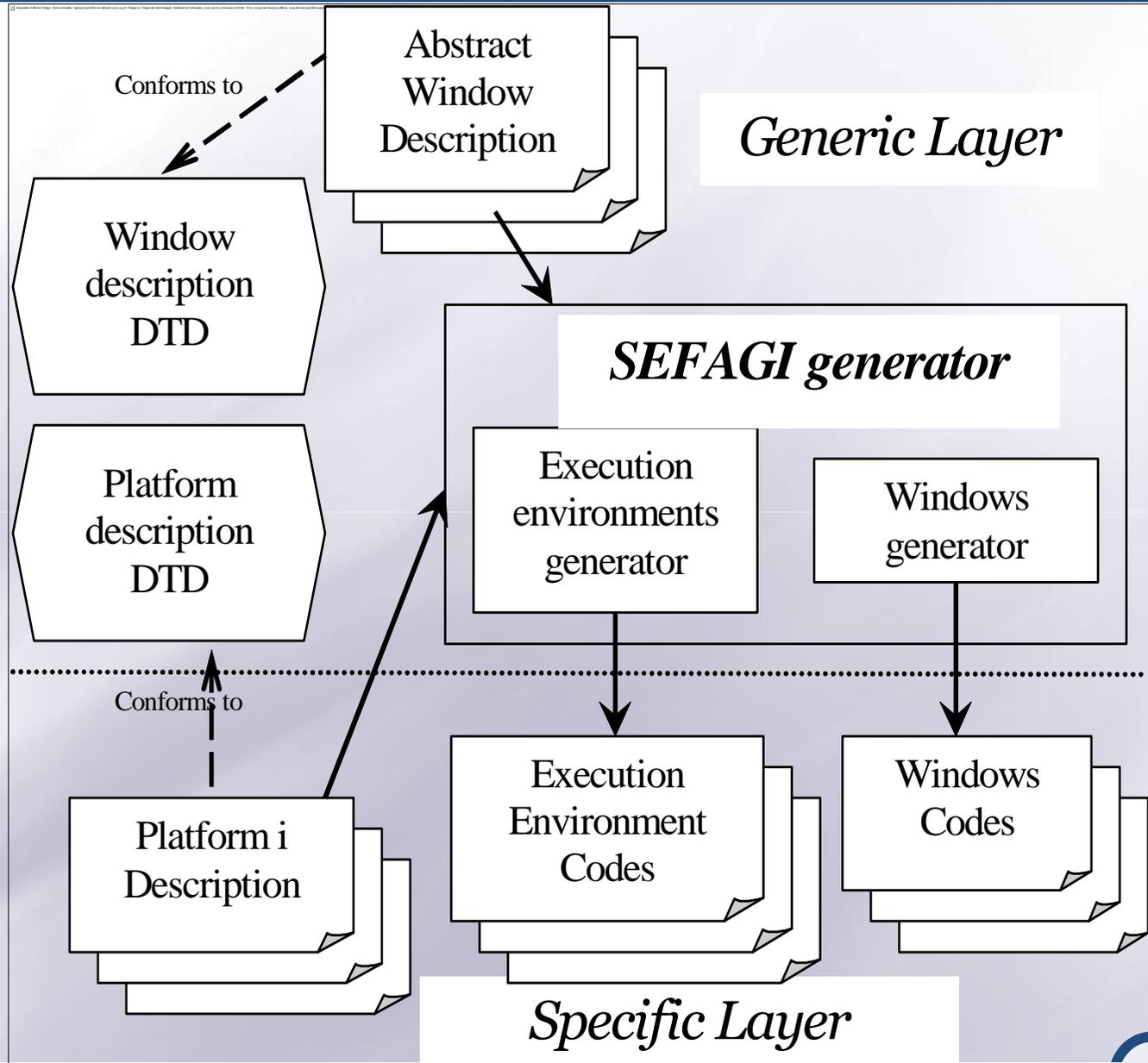
- pour des terminaux de types divers
  - PC (J2SE)
  - Terminaux très légers (MIDP- CLDC)
  - Terminaux légers (CDC)

## Génération automatique et complète des :

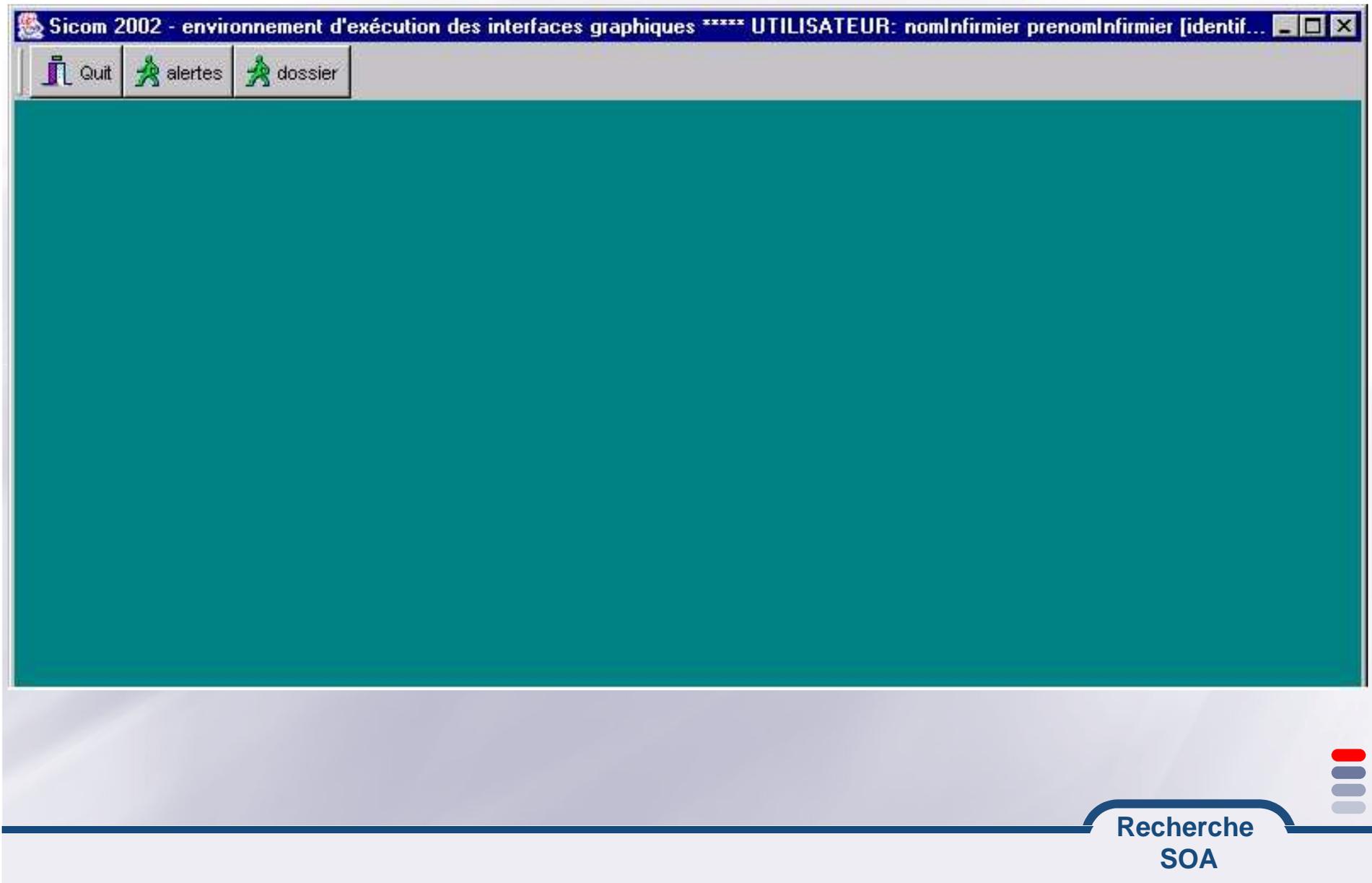
- Affichages et leurs inter-relations
- Interactions avec les traitements
- Et de l'environnement d'exécution sur les terminaux



# Architecture logique



# Environnement d 'exécution sur PC



# Description des fenêtres à générer

☰ fenêtr = liste de panneaux

☰ panneau =

- Une représentation graphique dépendant du type de terminal
- Des services associés (getData / setData)

☰ Types de panneaux

- 6 pour l'instant, extensible
- Chaque type de panneau correspond à une classe dans une bibliothèque de composants graphiques

Numéro Alerte	Numéro Dossier	Date	
15	3	2002-02-24 00:00:00	Anome
16	3	2002-02-24 00:00:00	Poids €
17	3	2002-02-24 00:00:00	Tensio
18	3	2002-02-24 00:00:00	Anome
25	3	2002-03-11 00:00:00	Tempé
26	3	2002-03-11 00:00:00	Poids €
27	3	2002-03-11 00:00:00	Tensio
28	3	2002-03-11 00:00:00	Tensio
3	1	2002-02-24 00:00:00	Anome
4	1	2002-02-24 00:00:00	Anome

# Panneaux tableau, texte, graphique

## Panneau de type tableau

- ☰ Saisie et consultation des données
- ☰ Une case = un widget de base (zone de saisie, liste énumérée, cases à cocher...)
- ☰ Contraintes chaque case

Températures	
general.dates	ge
2002-03-11 00:00:00	41
2002-03-12 00:00:00	39
2002-03-13 00:00:00	40
2002-03-14 00:00:00	41
2002-03-15 00:00:00	39
2002-03-16 00:00:00	38
2002-03-17 00:00:00	38
2002-03-18 00:00:00	37
2002-03-19 00:00:00	37

PC standard

Veillez choisir une valeur

2002-03-11 00:00:00

2002-03-12 00:00:00

2002-03-13 00:00:00

2002-03-14 00:00:00

2002-03-15 00:00:00

2002-03-16 00:00:00

2002-03-17 00:00:00

2002-03-18 00:00:00

2002-03-19 00:00:00

↓

**\*\* Valeurs \*\***

general.dates

general.temperatures

PC de poche

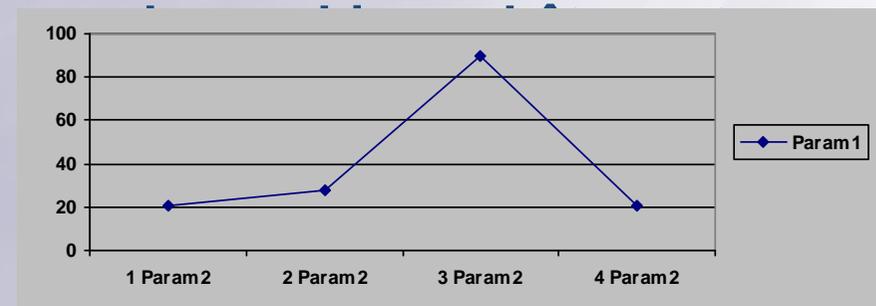
## Panneau de type texte

- Grande zone texte (commentaires, texte)

Texte Libre

## Panneau de type graphique

- Présentation de courbes 2D

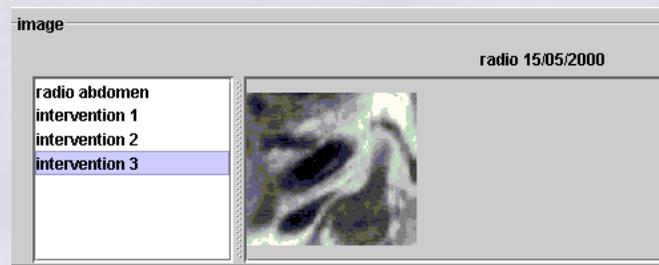


Recherche  
SOA

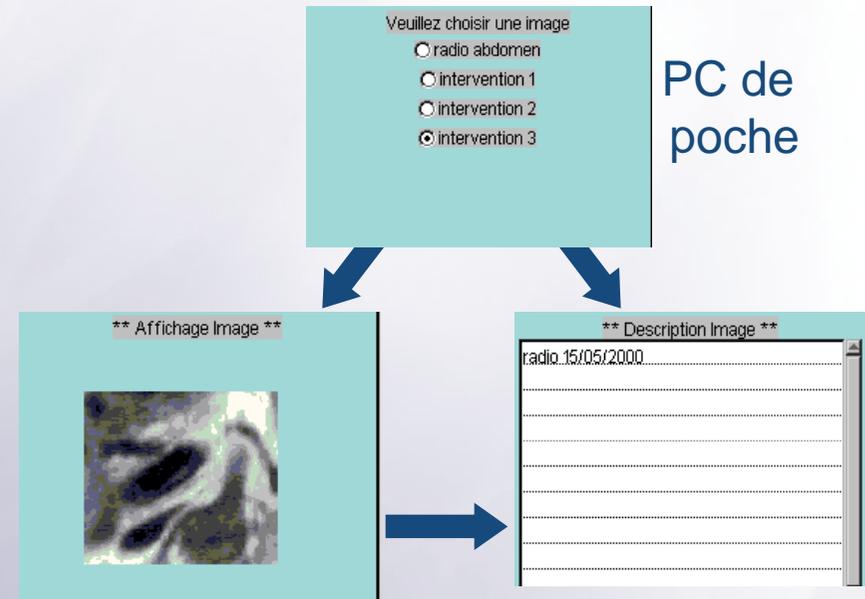
# Panneau de type image, vidéo et commande

## Panneaux image ou vidéo

- Liste des images/vidéos
- Image/vidéo sélectionnée
- Texte descriptif associé



PC  
standard



## Panneau commande

- Boutons pour lancer l'appel des services associés aux panneaux de la fenêtre



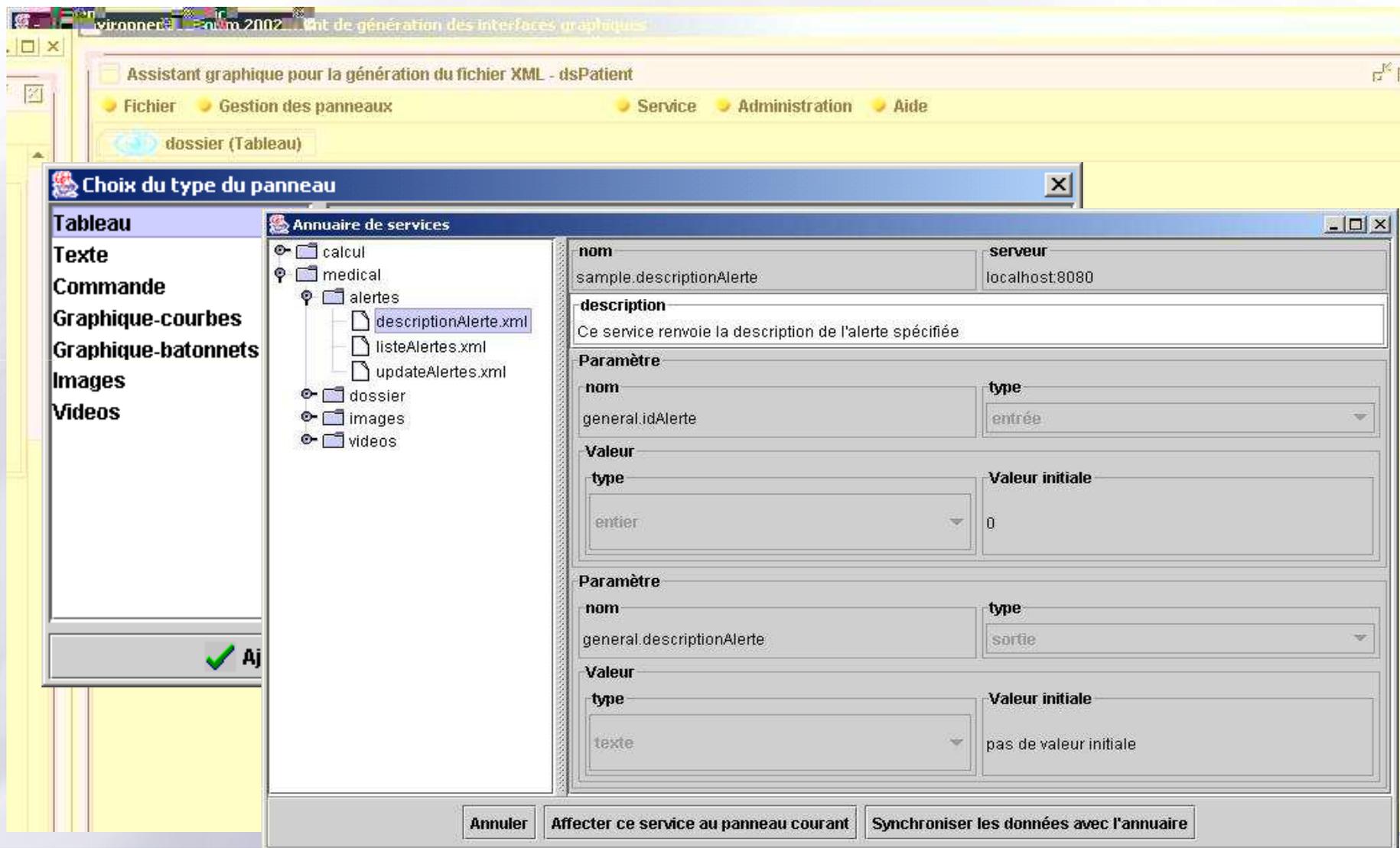
PC standard

Description  
Alertes  
Images  
Update

PC de  
poche

Recherche  
SOA

# Assistant de SEFAGI



Structure	Values
interface	
nom	nom de la fenêtre
titre	titre de la fenêtre
panneau	
id	identifiant unique du panneau
type	type du panneau (tableau, images,...)
servicepardefaut	le service par défaut du panneau
titre	titre du panneau
service	
id	identifiant unique du service (nom.méthode)
server	serveur hébergeant le service (serveur:port)
parametre	
type	type du parametre (entrée, sortie,entrée/sortie)
nom	nom du parametre
valeur	
type	type de la valeur
localisation	composant ou variable (lien XPointer)
reference_groupe	1 iere référence sur la localisation
reference_specifique	2 ieme référence sur la localisation
val_initiale	valeur par défaut du parametre
contraintes	
type	énumération, intervalle...
contrainte	1 ère contrainte
contrainte	2 ème contrainte...
affichage	
composant	
type	type du composant (texte, numérique, liste,...)
editable	oui / non
visible	oui / non
id	identifiant unique du composant
titre	titre du composant
action	
panneaucible	id d'un panneau cible
servicecible	id du service cible

**Structure XML de description abstraite des fenêtres (aucune référence aux plateformes cibles)**

# Environnement de génération

## ☰ Interface principale permettant de

- Choisir une description abstraite de fenêtre
- Générer pour un type de terminal donné le code source de la fenêtre correspondante et de son gestionnaire d'évènements
- Compiler le code source généré



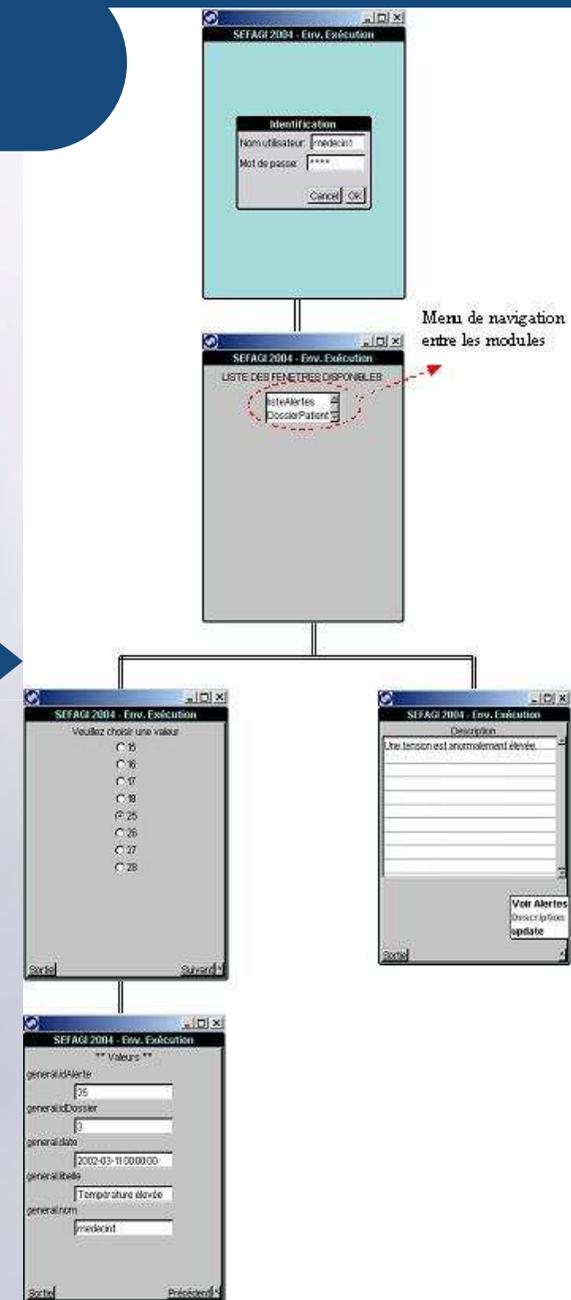
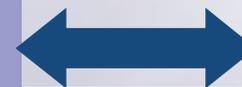
# Exemple complet sur PC et mobile

## Menu de navigation

The screenshot shows the SEFAGI 2004 PC application. At the top, a navigation menu contains icons for 'Quit', 'DossierPatient', and 'listeAlertes'. The 'listeAlertes' window is active, displaying a table of alerts with the following data:

general.idAlerte	general.idDossier	general.date	general.libelle	general.nom
15	3	2002-02-24 00:00:00	Anomalie Cathéter	medecin1
16	3	2002-02-24 00:00:00	Poids élevé	medecin1
17	3	2002-02-24 00:00:00	Tension élevée	medecin1
18	3	2002-02-24 00:00:00	Anomalie Cathéter	medecin1
25	3	2002-03-11 00:00:00	Température élevée	medecin1
26	3	2002-03-11 00:00:00	Poids élevé	medecin1
27	3	2002-03-11 00:00:00	Tension élevée	medecin1
28	3	2002-03-11 00:00:00	Tension élevée	medecin1

Below the table, there is a 'Description' section with the text 'Une tension est anormalement élevée.' and a 'Commandes' section with 'description' and 'update' buttons.



# Conclusion SEFAGI

## Simplicité

- Avoir des interfaces graphiques spécifiques sans toucher à la programmation
- Un fichier XML par fenêtre très court par rapport aux autres modèles existants

## Évolutivité

- Enrichissement de la liste des panneaux

## Portabilité

- Entièrement développée en JAVA



# Adaptation de contenu



# Adaptation du contenu

## ☰ Adaptation des données : modifier une donnée pour

- Qu'elle soit exploitable par le terminal cible
- Qu'elle soit conforme aux règles de protection des données
- ...

## ☰ Adaptation dynamique (à la demande) ou statique (plusieurs versions stockées)

## ☰ Localisation de l'adaptation

- sur le client : ne convient pas aux terminaux légers
- sur le serveur : si tient la charge
- Sur un proxy entre le client et le serveur
- Distribué



# Adaptation selon le type de la donnée source

## Source texte

- Conversion de format (html -> txt, doc -> pdf...)
- Résumé
- Traduction
- Compression/décompression
- Synthèse vocale

## Source image

- Conversion de format (jpeg -> png)
- Modification de résolution, nombre de couleurs...
- Compression / décompression (e.g. sémantique jpeg ou brute zip)



# Adaptation selon le type de la donnée source

## Source sonore

- Conversion de format
- Synthèse textuelle et reconnaissance vocale
- Compression / décompression (e.g. sémantique MP3 ou brute zip)

## Source vidéo

- Conversion de format (résolution, nb images/sec)
- Décomposition / recomposition spatiale (zoom...)
- Compression / décompression (e.g. sémantique MPEG4 ou brute zip)



# Opérateurs pour l'adaptation de contenu

## Codage

- transcodage de média (réduction nb couleurs)
- transformation de modalité (texte -> audio)
- compression (jpeg)

## Format (Wav->MP3)

## Structure (HTML ->WML)

## Remplacement (image par texte descriptif)

## Sélection (sélection d'images moins volumineuses)

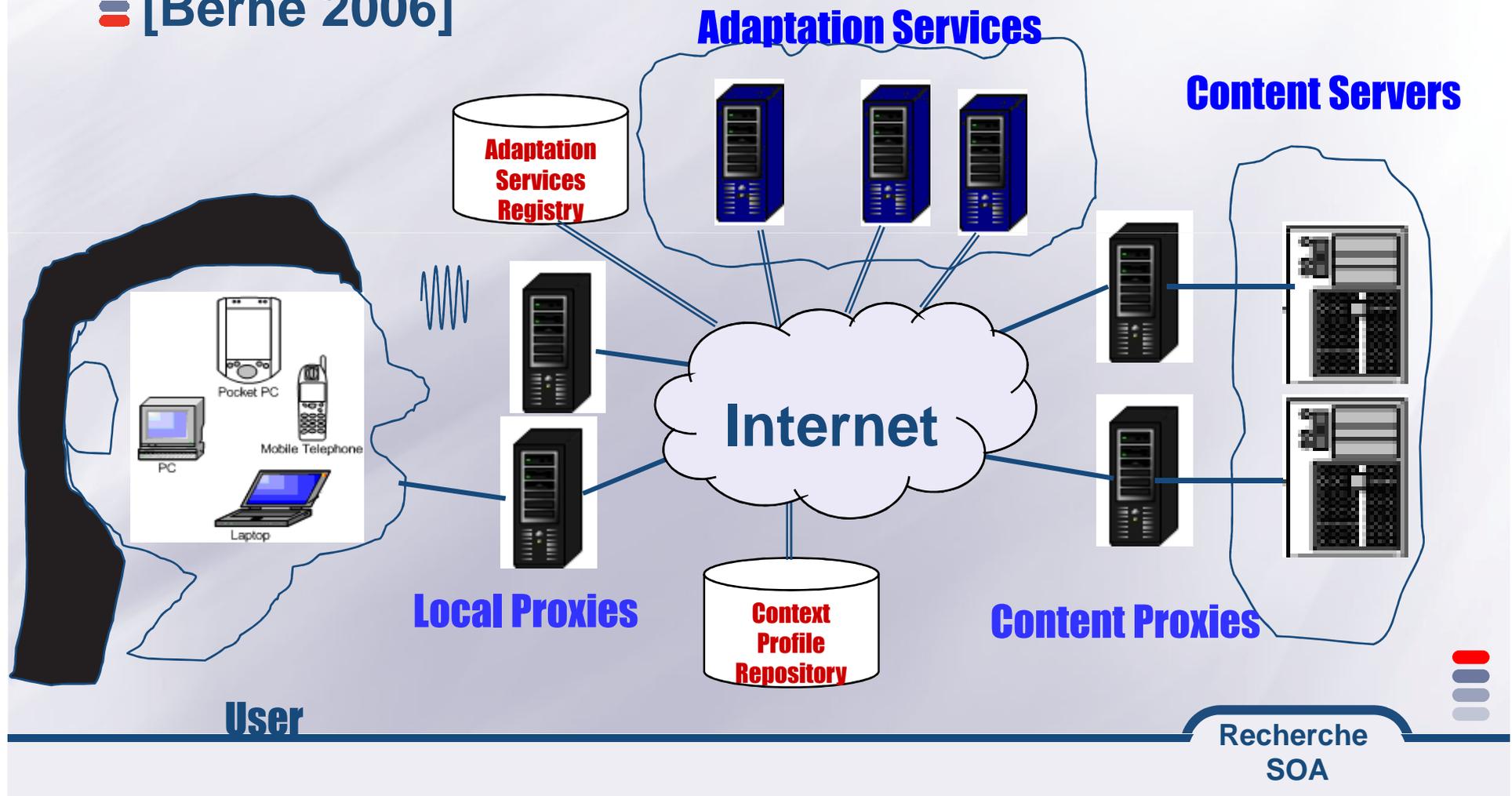
## Intégration (données multi-serveurs)



# Adaptation de contenu - DCAF

## Distributed Content Adaptation Framework

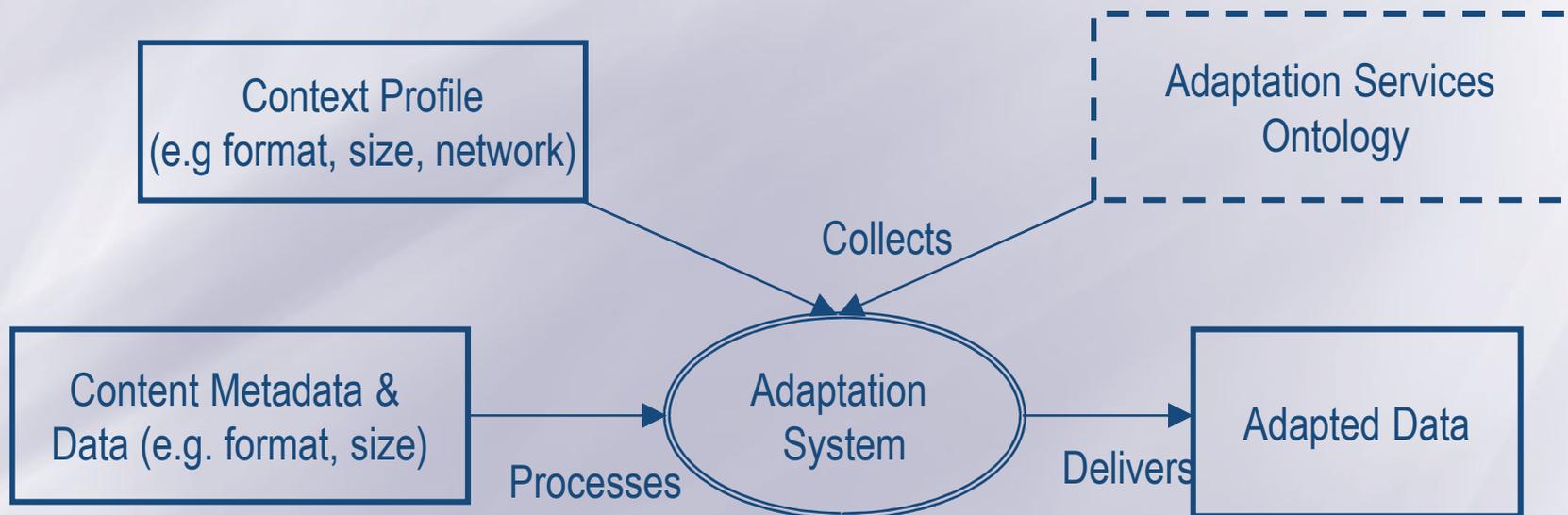
[Berhe 2006]



# Adaptation de contenu - DCAF

## ☰ Description des services d'adaptation

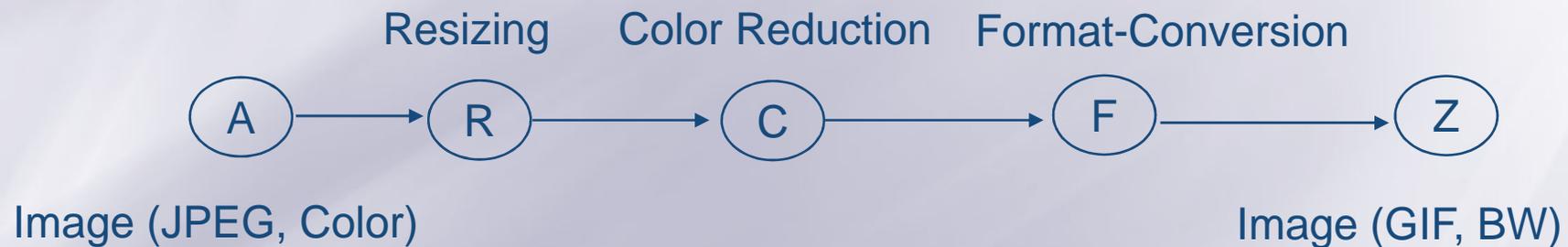
- Services d'adaptation = services Web
- WSDL insuffisant
  - Pas de coût, sémantique, temps d'exécution...
- Utilisation d'une ontologie



# Adaptation de contenu - DCAF

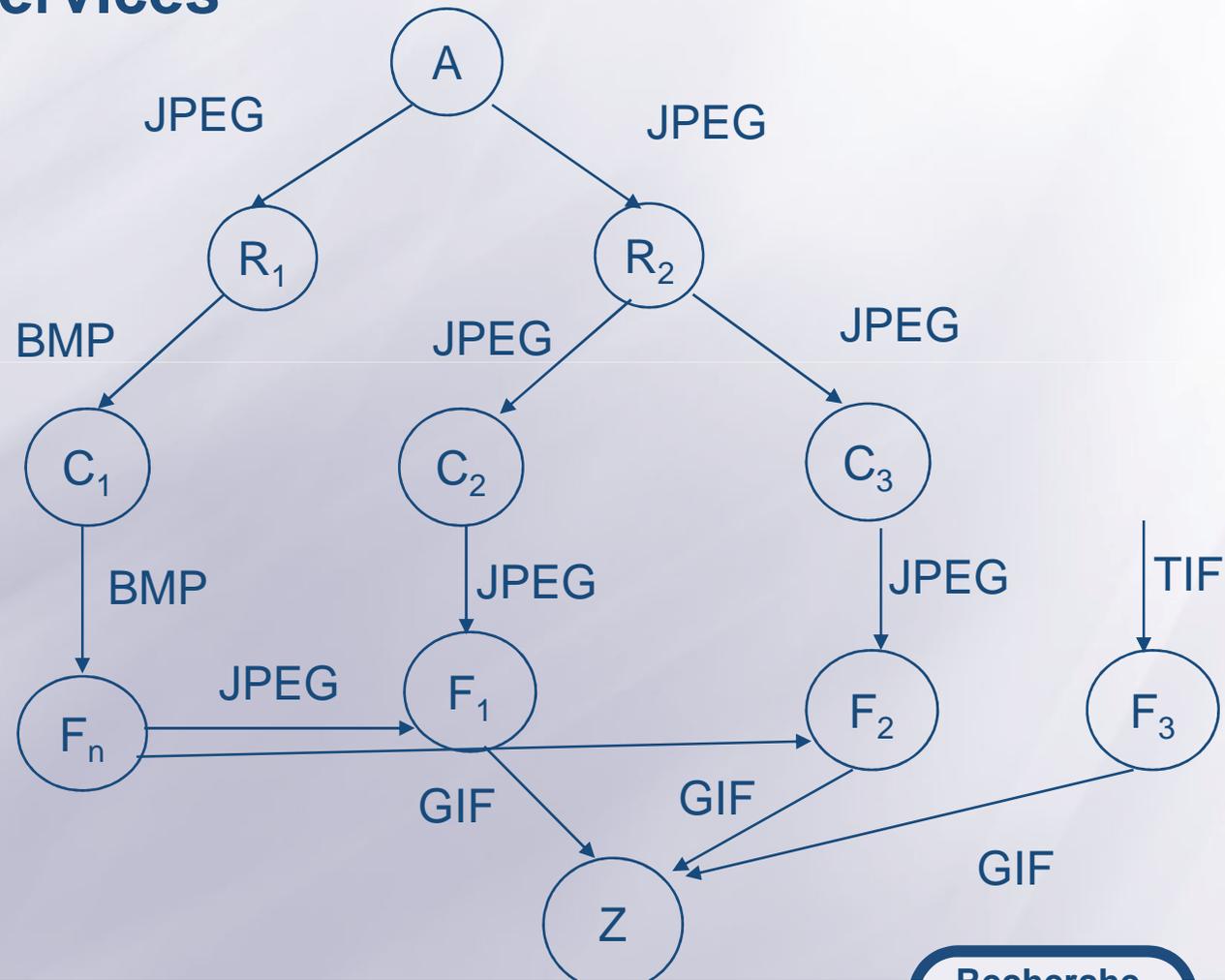
## AD : décision d'adaptation

- Calcule le type et le nombre d'adaptations nécessaires
- Résultat : transformation prescript

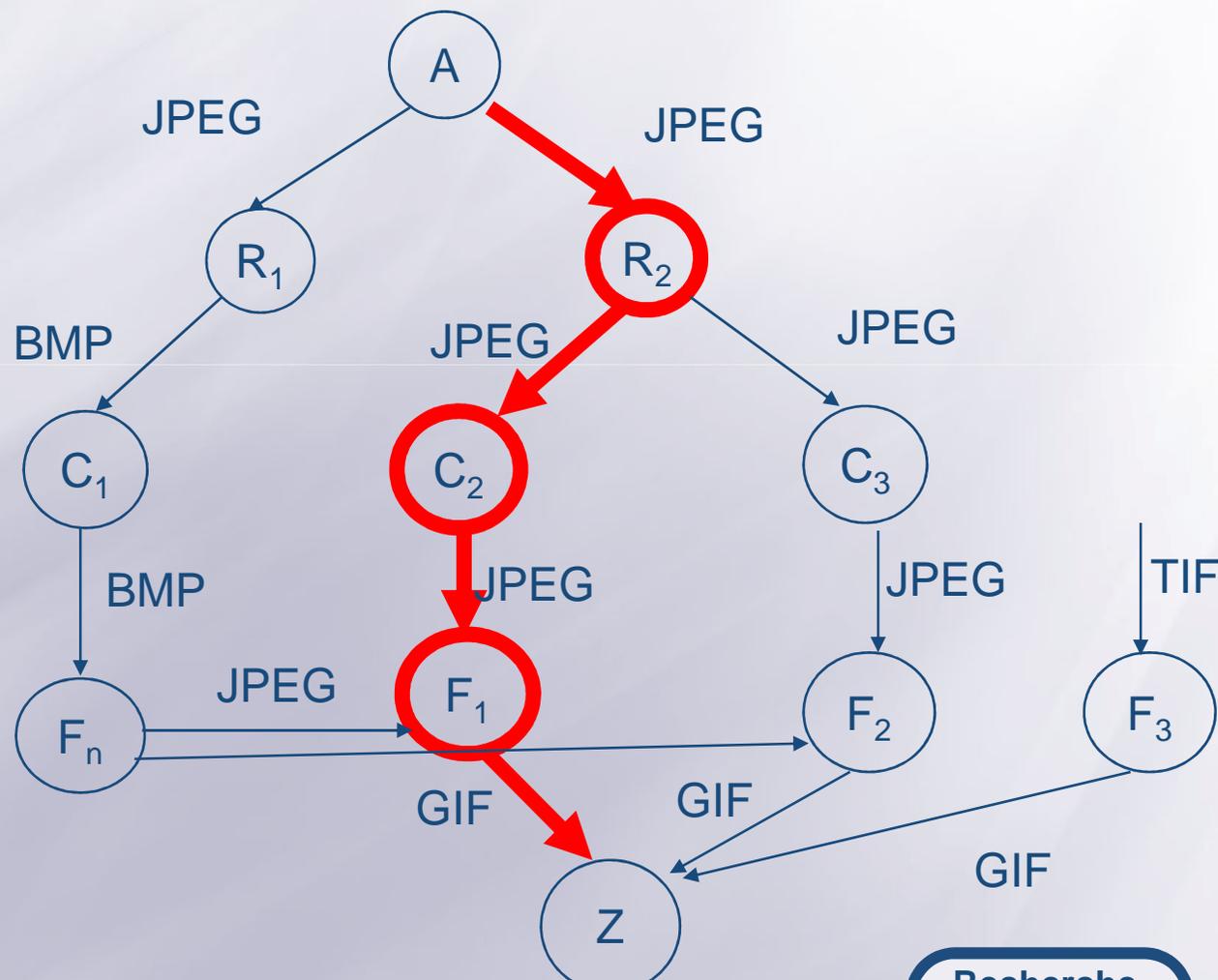


# Adaptation de contenu - DCAF

## Graphe de services



# Adaptation de contenu - DCAF



# Adaptation de services :

## Le projet SECAS



# SECAS: Simple Environment for Context-Aware Systems



**[Chaari 2006]**



**Encapsuler une application existante avec une couche d'adaptation**

- sans modification de l'application d'origine
  - Pour pouvoir intégrer toute application existante
- à l'aide de services web



**Adaptation en fonction du contexte**

1. des services
2. des contenus
3. de l'interface utilisateur



# SECAS: Objectifs

## Stratégie complète d'adaptation

- Source du besoin (adapter à quoi?)
- Objet de l'adaptation (adapter quoi?)
- Outils d'adaptation (adapter avec quoi?)
- Utilisation de ces outils pour assurer l'adaptation (adapter comment?)

## Plateforme pour l'adaptation d'applications au contexte



# SECAS

## Adapter à quoi?

### ☰ Notre définition du contexte

- L'ensemble des paramètres externes à l'application pouvant influencer sur son comportement en définissant de nouvelles vues sur ses données et ses services

### ☰ Conséquence

- Séparation des paramètres du contexte et des données de l'application
- Ajout de la sensibilité au contexte à des applications existantes

### ☰ Propriétés

- Les paramètres du contexte ne parviennent pas des sources de données de l'application
- Nouvelle instance = nouvelle *situation contextuelle*

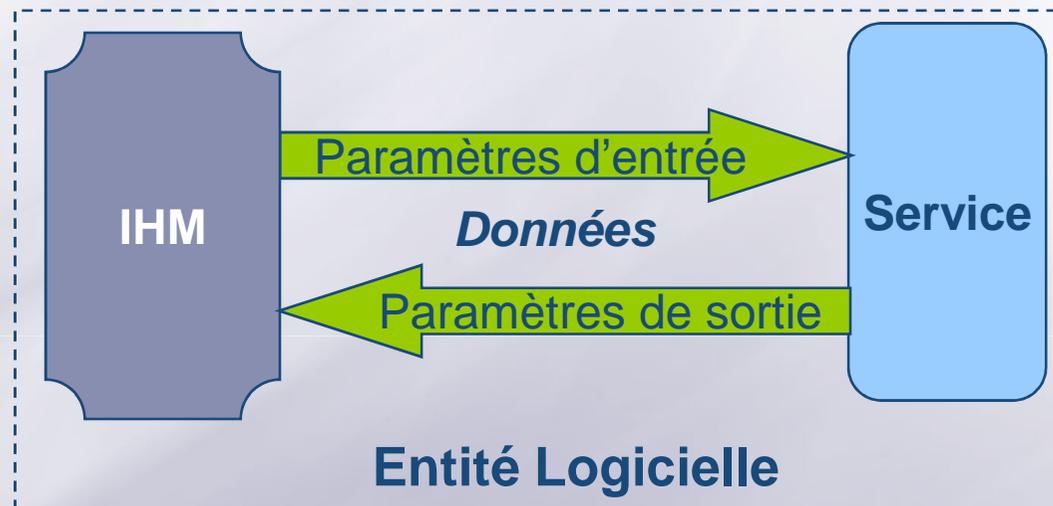
### ☰ Modélisation

- Terminal, Utilisateur, Localisation, Communication et Environnement
- Utilisation des ontologies (voir GCM)

# SECAS

## Adapter quoi?

### ☰ Vue externe d'une application : entité logicielle

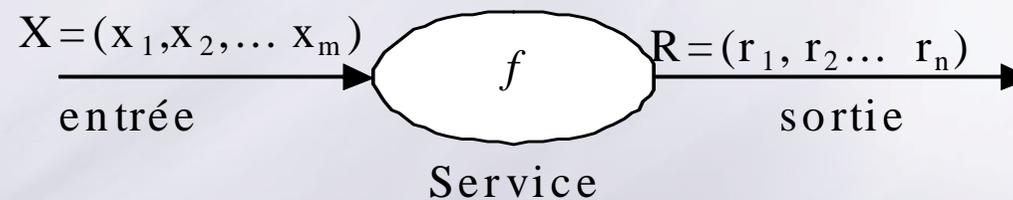


- Fonction (service)
- Présentation (IHM)
- Contenu (données)

# SECAS

## Modélisation d'un service

*Un processus applicatif autonome  $R = f(X)$*



Vecteur d'entrée du service *infosClients*

codePostal

69 100

Vecteur de sortie du service *infosClients*

id Client	nom Client	photo Client
-----------	------------	--------------

1	Dupont	1.jpg
2	Durant	2.jpg
k	Martin	k.jpg

### ☰ Modèle d'un service $f$

- $R = f(X)$ ,  $R = (r_1, r_2, \dots, r_n)$  et  $X = (x_1, x_2, \dots, x_m)$

### ☰ Dépendances de services $R_2 = g(X_1, X_2)$

- $X_1 = (e_1, e_2, \dots, e_l)$  des paramètres d'entrée spécifiques à  $g$
- $X_2 = (r_1, r_2, \dots, r_k)$   $k \leq n$  un sous ensemble des paramètres de sortie de  $f$
- Dans ce cas  $g$  dépend de  $f$

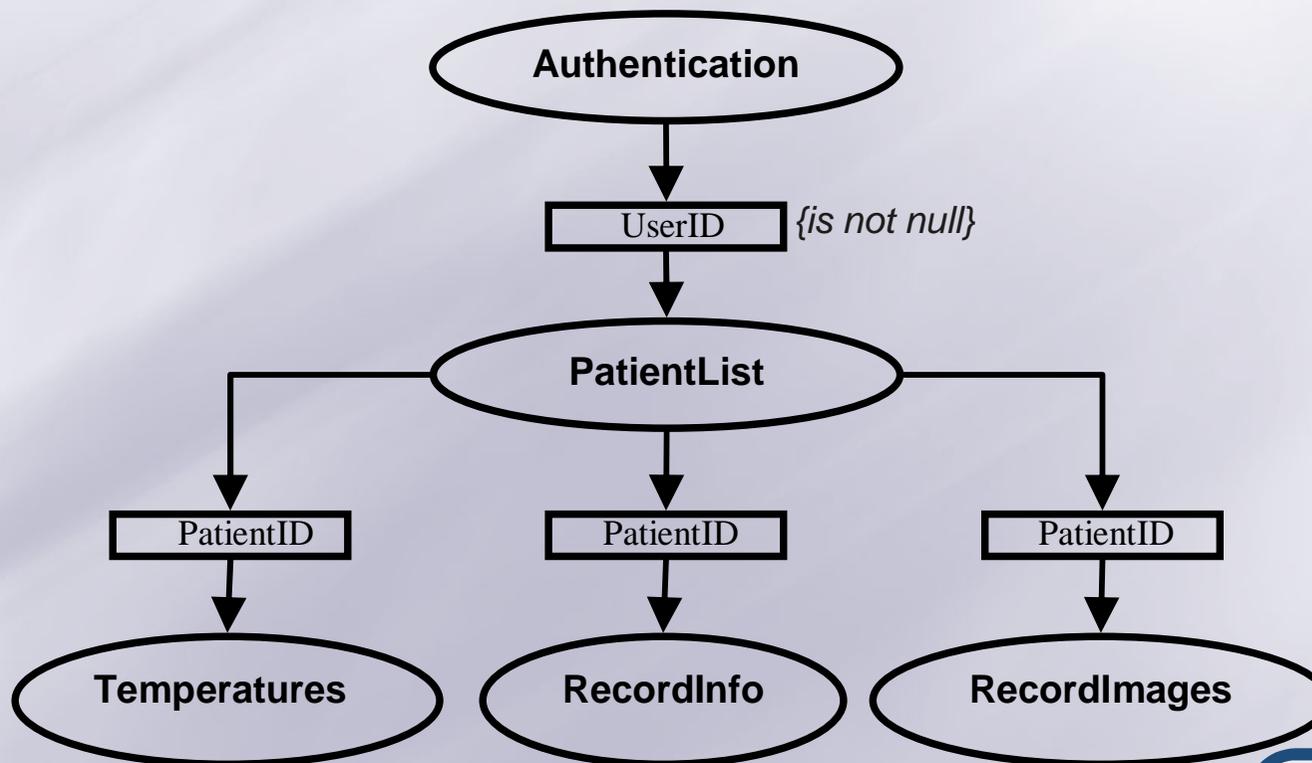
### ☰ Modèle fonctionnel de l'application = les services de l'application + toutes les dépendances entre eux

# SECAS

## Exemple de modèle fonctionnel

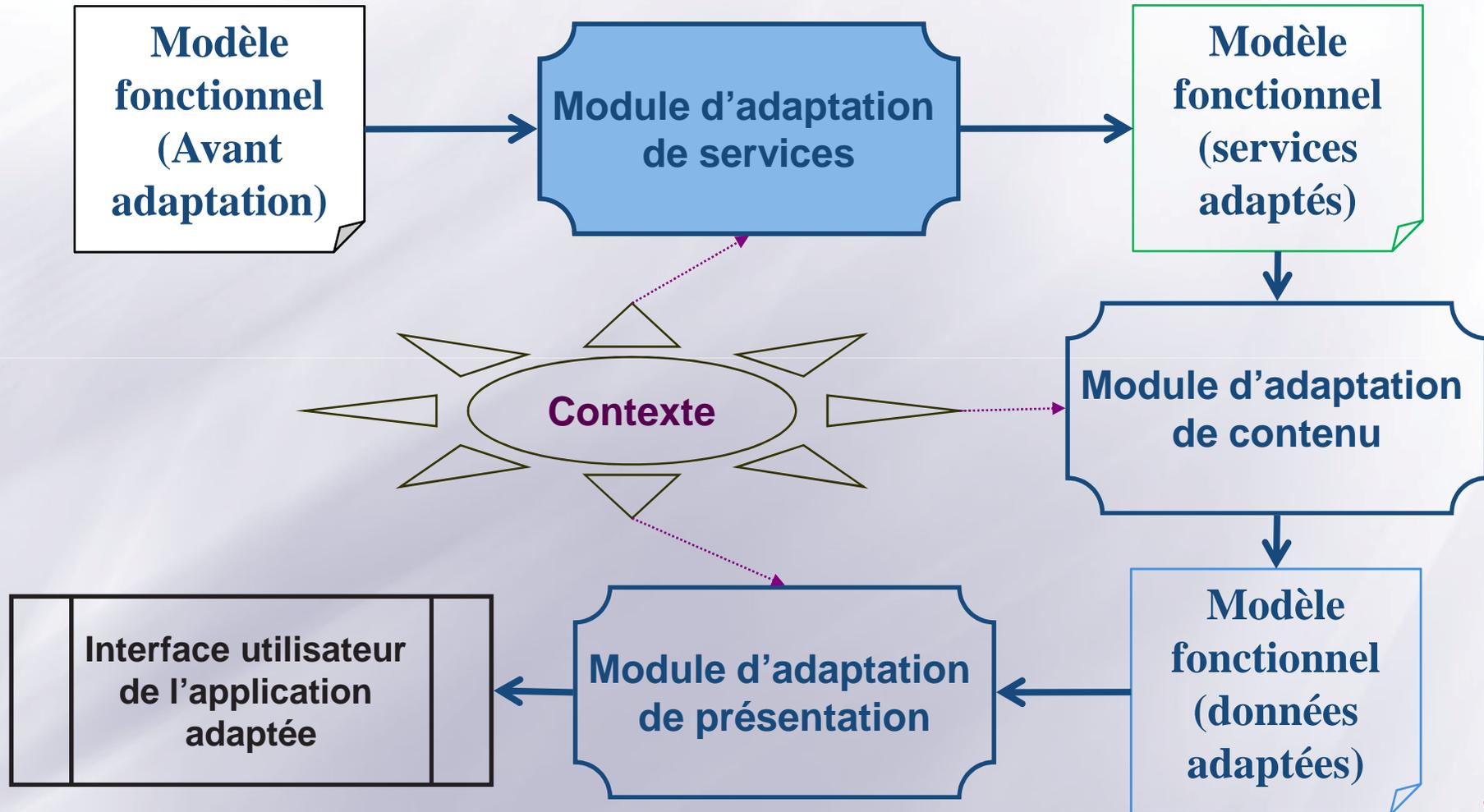
### ☰ Réseau de pétri (f0, F, T)

- F = l'ensemble des services de l'application (places)
- T = l'ensemble des dépendances entre les services (transitions)
- f0 = service initial de l'application



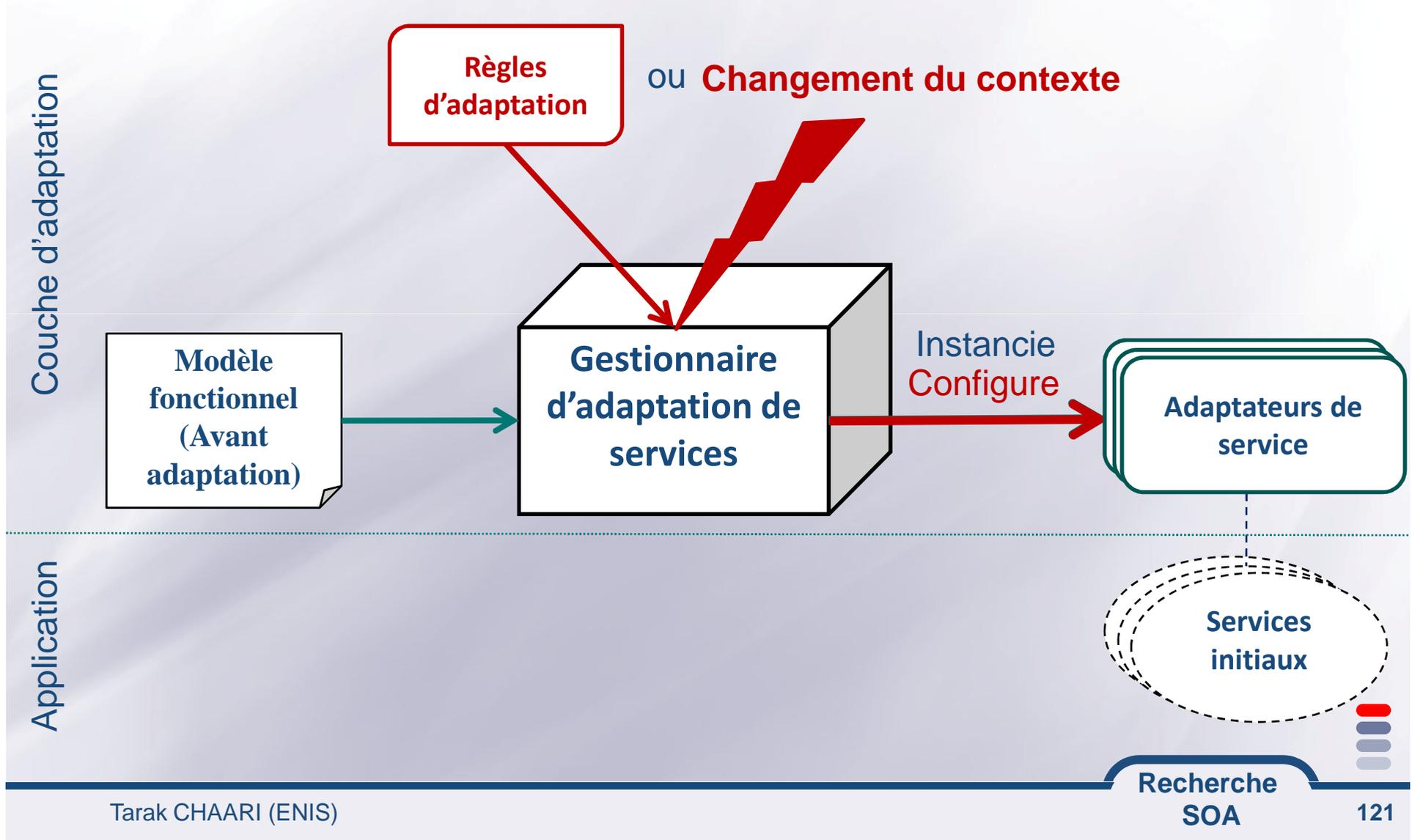
# SECAS

## Notre approche d'adaptation



# SECAS

## Module d'adaptation fonctionnelle (services)



# SECAS

## Règles d'adaptation fonctionnelle

### ☰ Paires (situation contextuelle, action d'adaptation)

- Situation contextuelle = Expression logique sur les paramètres du contexte et les éléments du modèle fonctionnel
- Action d'adaptation = Liste d'appels à des opérateurs d'adaptation

### ☰ Exemple

$$\{(\neg \text{context.terminal.acceptedDataTypes.acceptImages}) \wedge$$
$$(\forall f \in F \mid \exists i \mid f.R[i].type = \text{"secas:binaryImage"})$$
$$\rightarrow \text{lockService}(f) \}$$

### ☰ Transformations sur le modèle fonctionnel

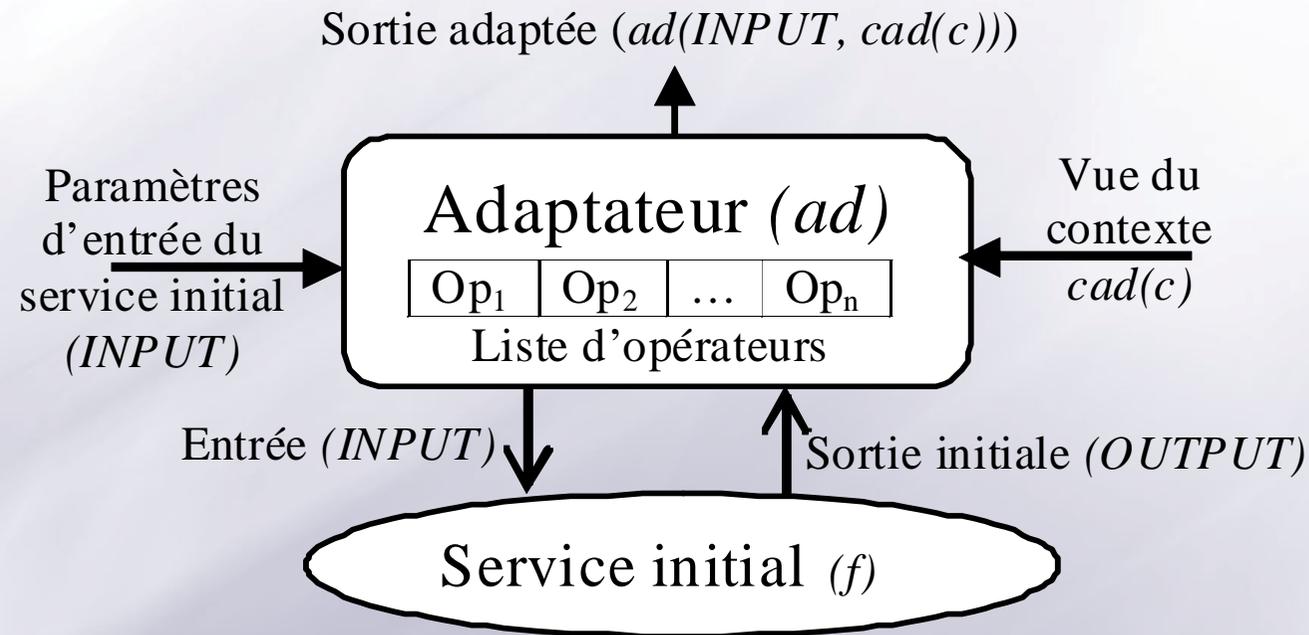
### ☰ **res = operator(parameters)**

res et parameters  $\subseteq$  modèle fonctionnel

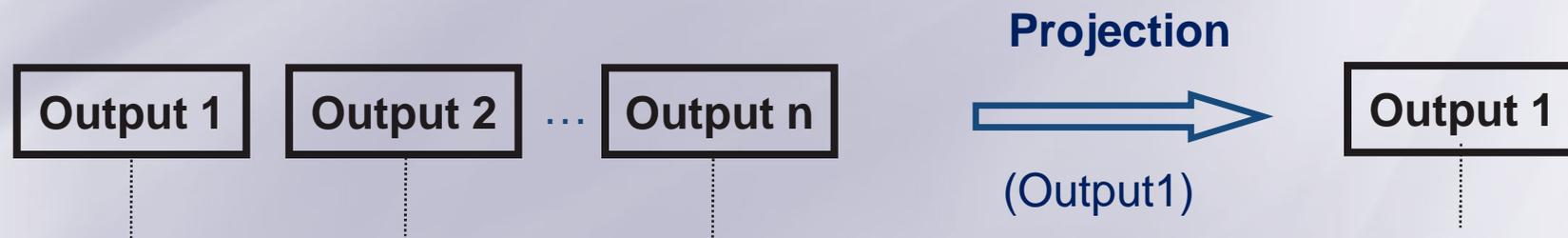
- **Opérateurs d'adaptation intra-service**
  - Vecteurs de sortie des services
  - Sans modifier la structure générale du modèle fonctionnel
- **Opérateurs d'adaptation inter-services**
  - Mise à jour des places ou des interconnexions
  - Modification de la structure du modèle fonctionnel

# SECAS

## opérateurs d'adaptation intra-service



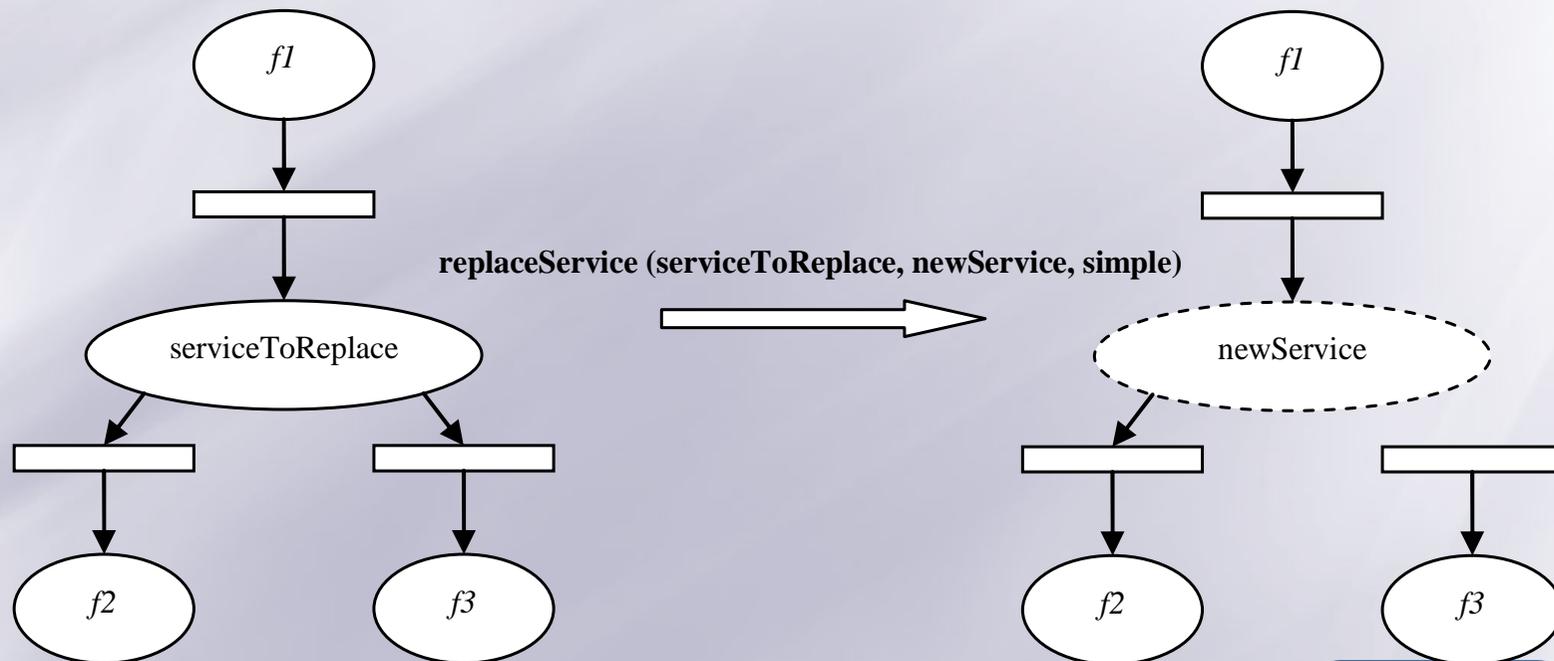
### Exemple d'un opérateur intra-service



# SECAS

## Opérateur d'adaptation inter-services

- Transformations sur le modèle fonctionnel
- Gérées par le gestionnaire d'adaptation de services
- Exemples: remplacement de services, insertion d'un service, verrouillage / déverrouillage d'accès

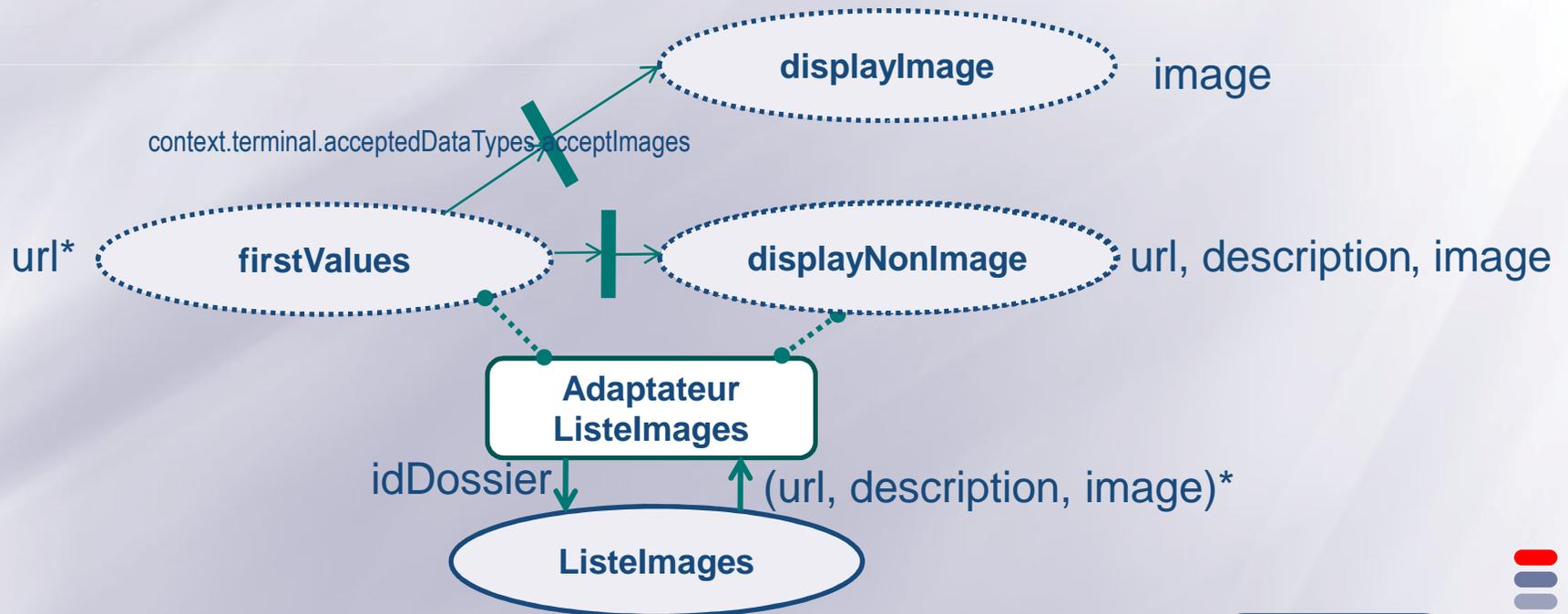


# SECAS

## Exemple d'adaptation fonctionnelle

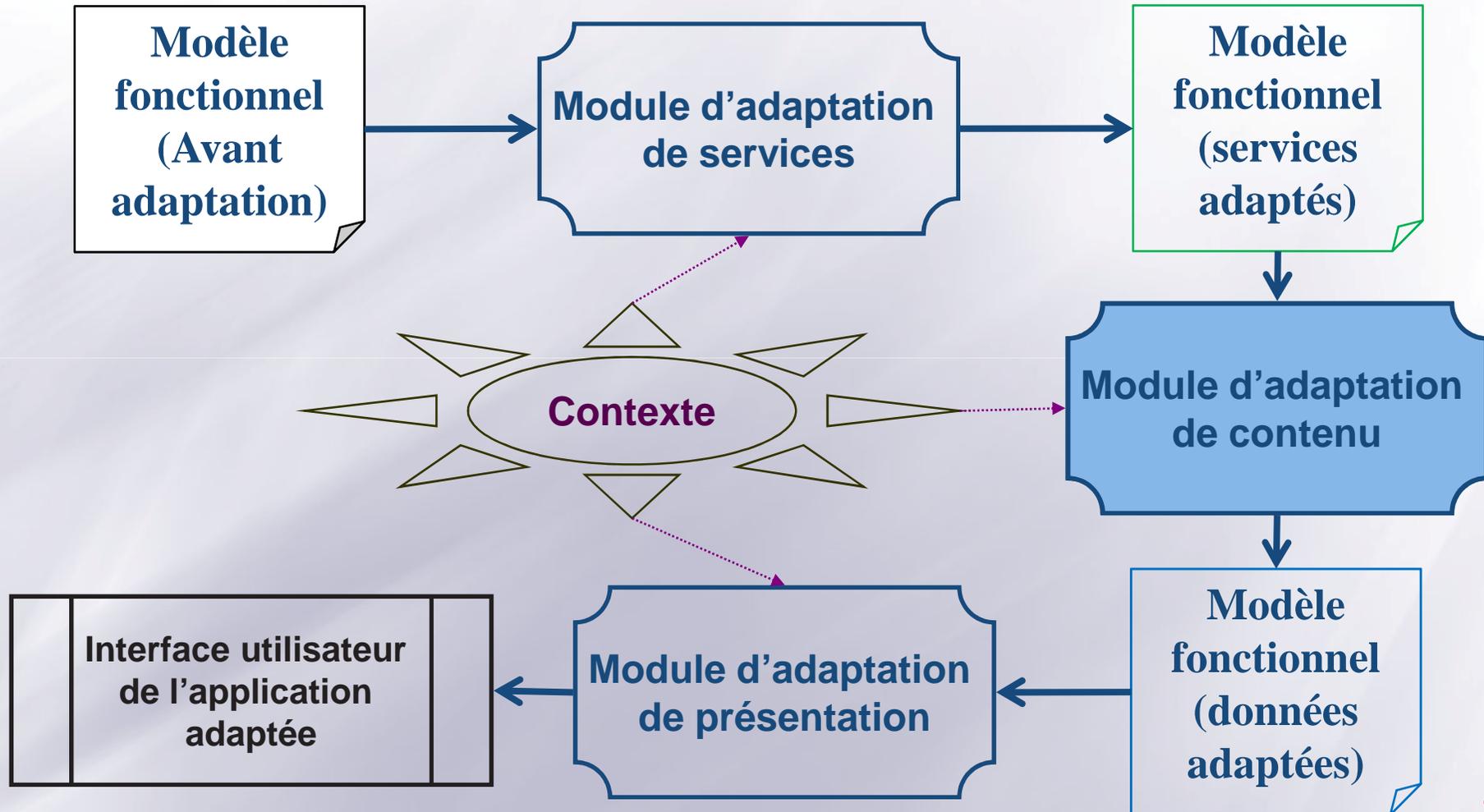
```

{((context.terminal.type="cldc") ^ (∑ f ∈ F | f.OUTPUT.length > 1)
 ^ ((context.terminal.hardwarePlatform.memory ≤ 10240) ^ (∑ f ∈ F | f.OUTPUT.length > 1) →
 ^ (∑ i ∈ I0 | i.OUTPUT.length | i.OUTPUT.type="secas.binaryImage") →
 firstValues(selectionOf(context.terminal.acceptedDataTypes,secas.selectedInstance)
 ^ (displayNonImage = projection(f, f.OUTPUT)))
 ^ (displayImage = projection(f, f.OUTPUT)))
 ^ (∑ f ∈ F | ∑ i ∈ I0 | selectionOf(f, f.OUTPUT.type="secas.selectedInstance")
 ^ (displayImage = projection(f, f.OUTPUT)))
 ^ insertServiceAfter(firstValues.selectedInstance)
 ^ insertAlternativeService(displayNonImage, displayImage)
 ^ replaceService(f,displayNonImage)
 ^ insertAlternativeService(displayNonImage, displayImage)
 }
  
```



# SECAS

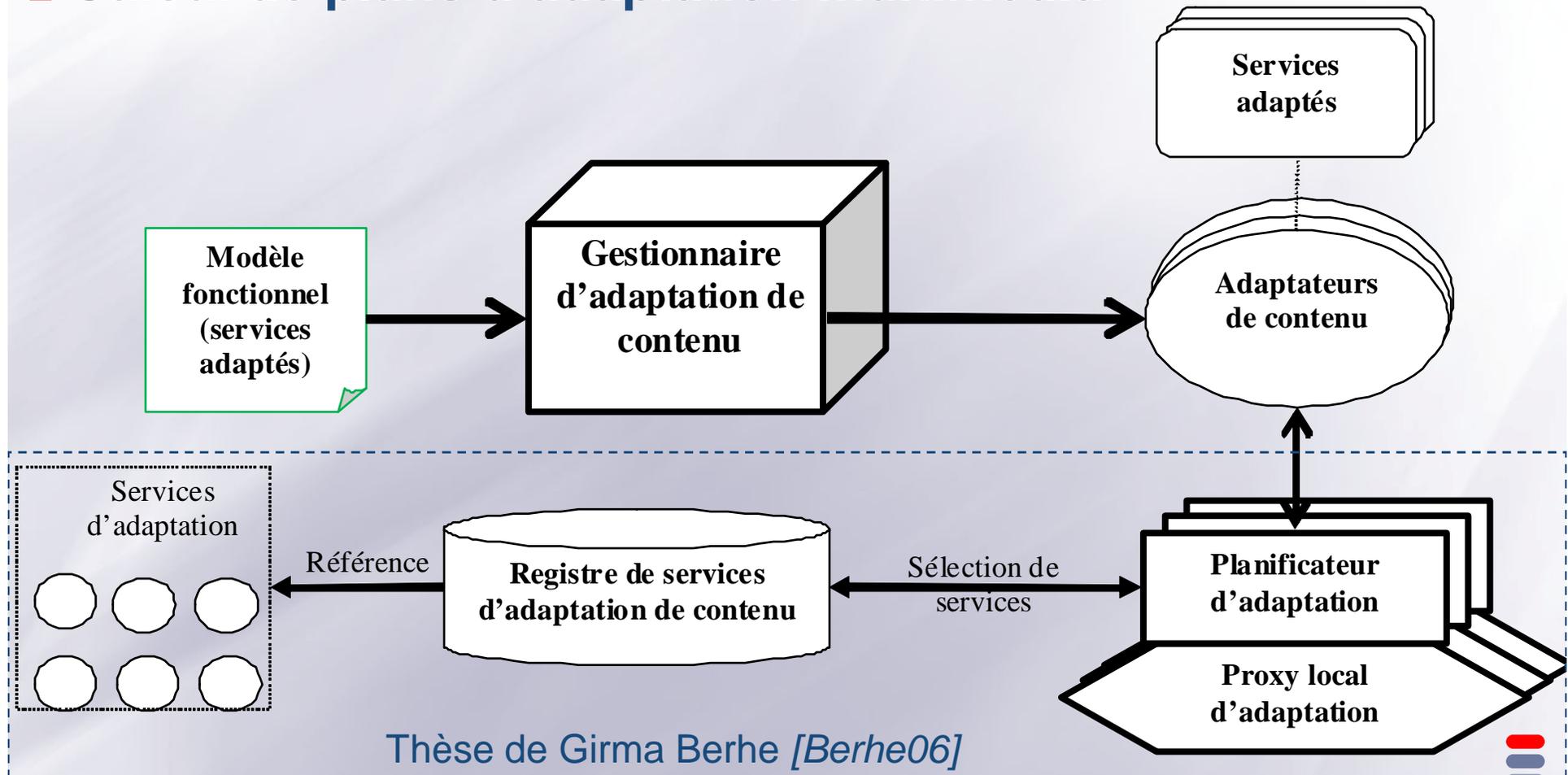
## Notre approche d'adaptation



# SECAS

## Module d'adaptation de contenu

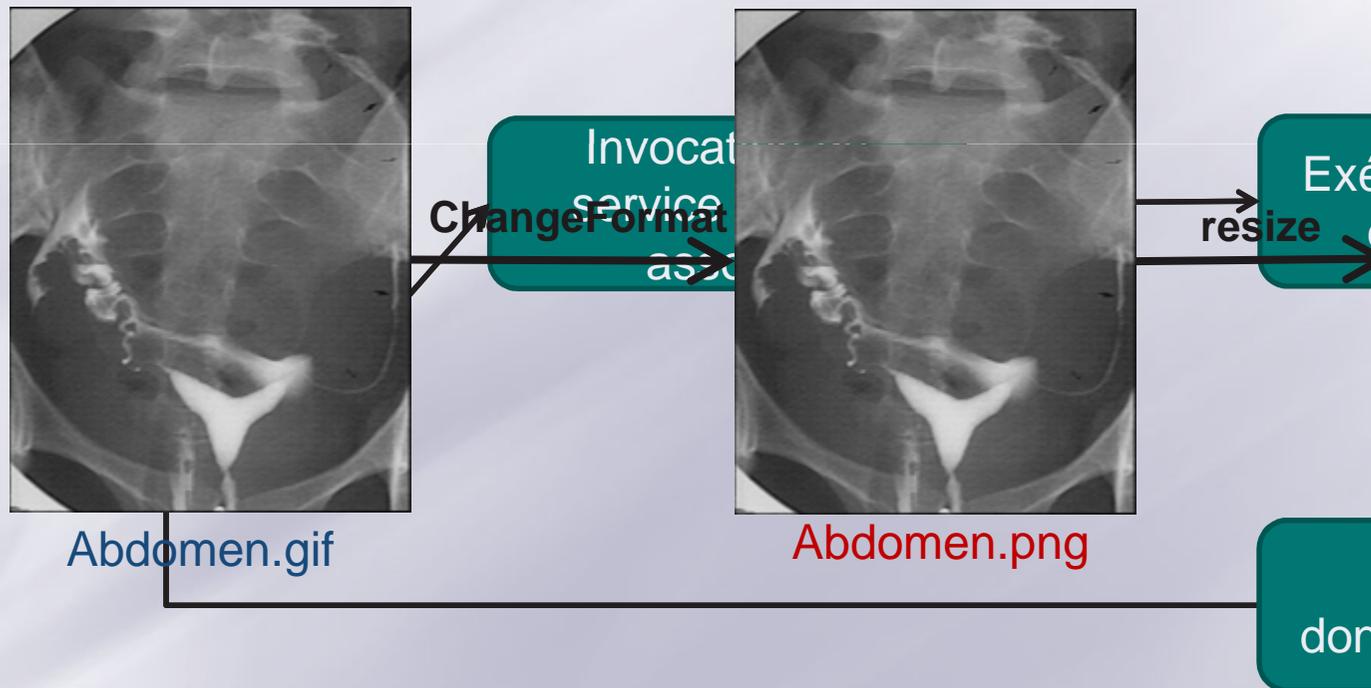
### Calcul de plans d'adaptation multimédia



# SECAS

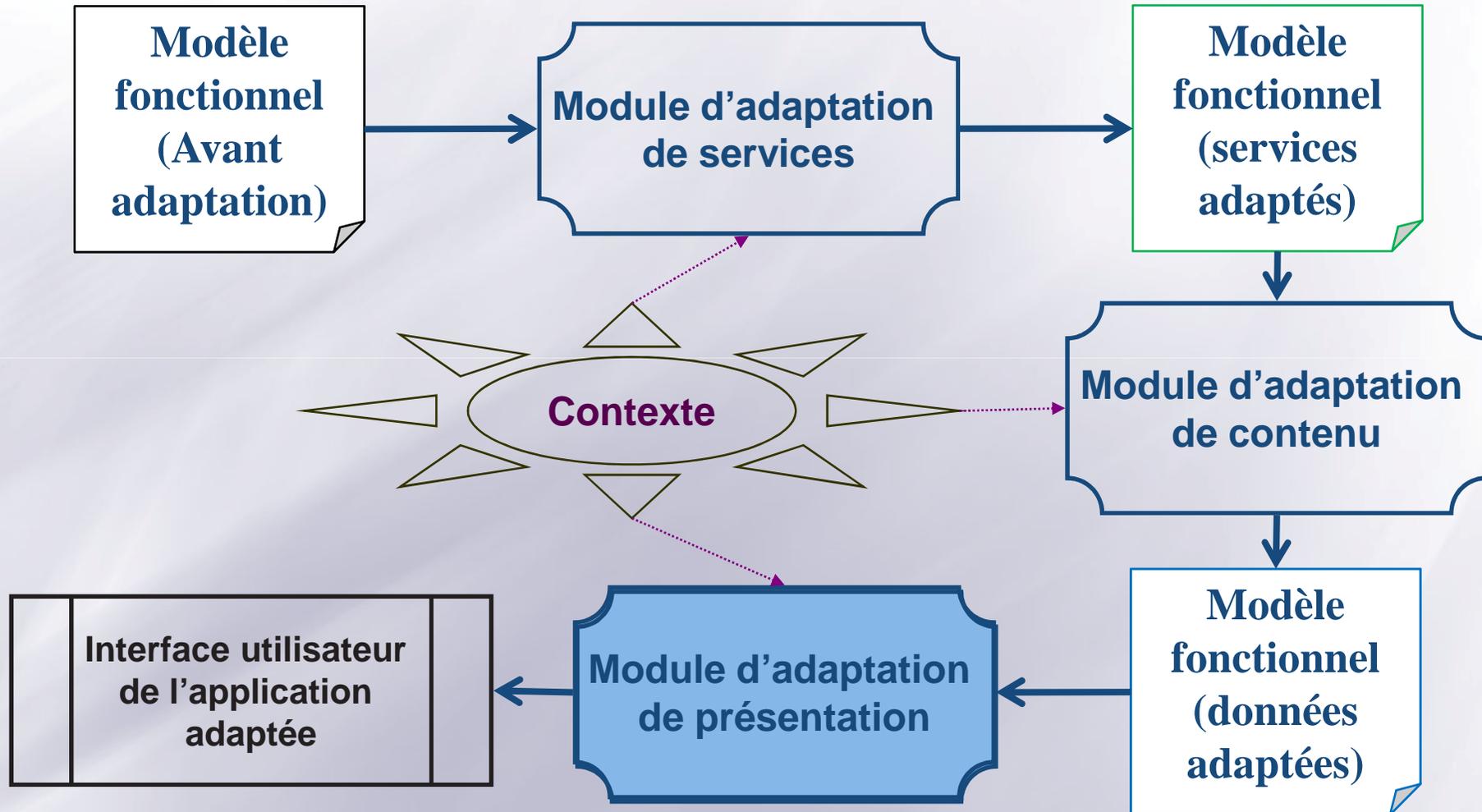
## Entité d'adaptation de données

### ☰ Fonctionnement de l'adaptateur de contenu



# SECAS

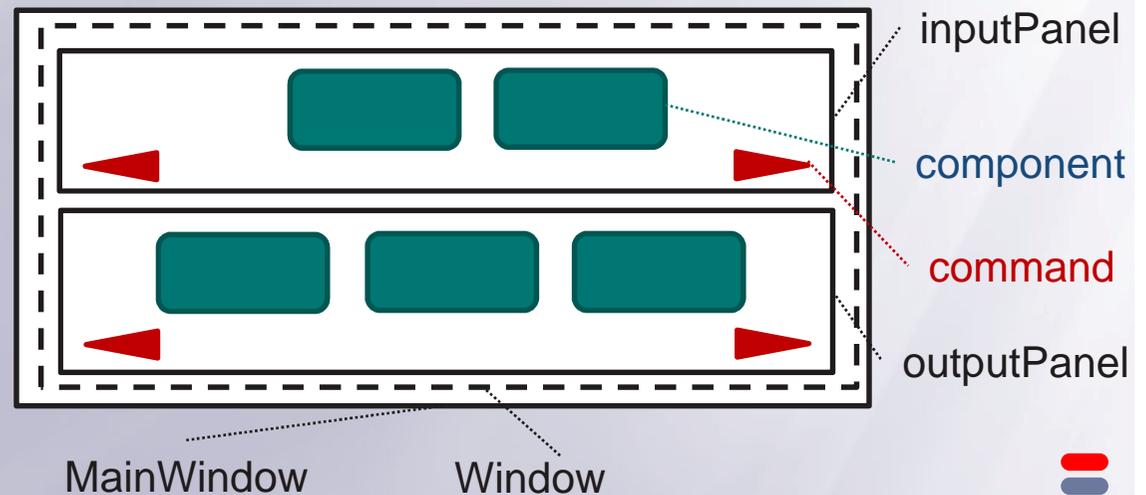
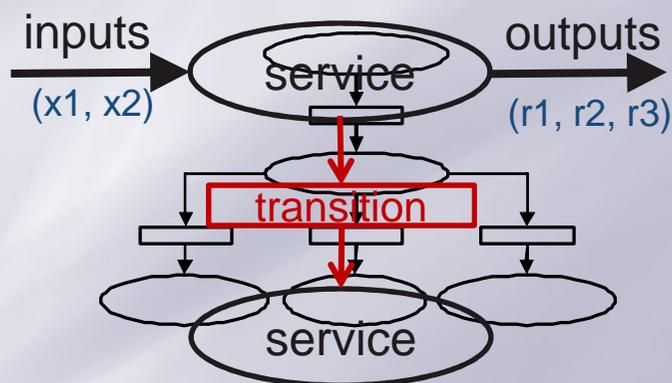
## Notre approche d'adaptation



# SECAS

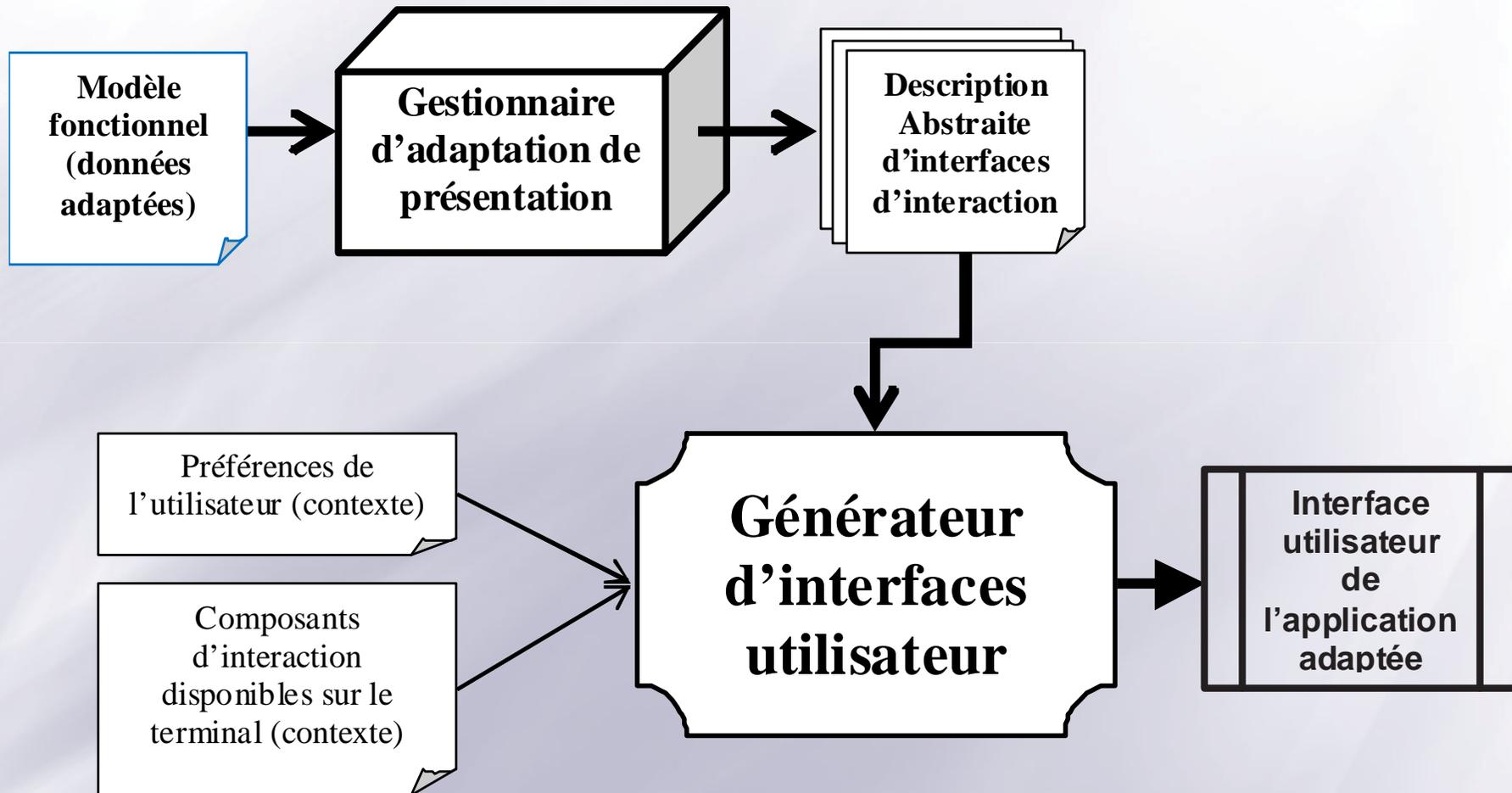
## Adaptation de présentation (IHM)

- Générer automatiquement une interface utilisateur adaptée au contexte d'utilisation de l'application
- Structure abstraite d'interaction avec une application



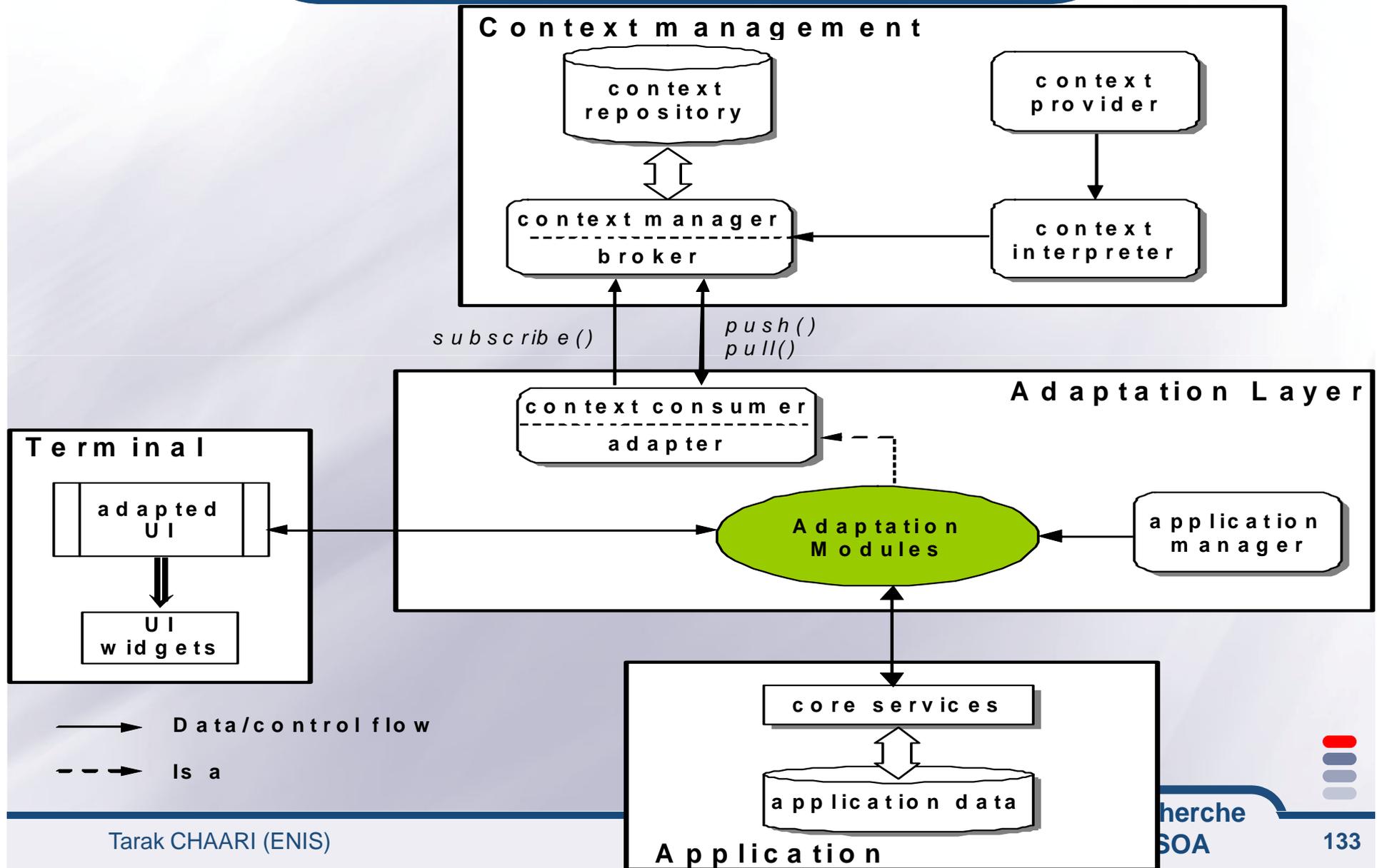
# SECAS

## Principe de l'adaptation de présentation



# Réalisations

## Architecture de la plateforme SECAS





# Réalisations SICOM avant adaptation

SECAS

File Configuration Help

**MENU**

- SERVICES
  - listeAlertes
  - listeDossierDialyse
  - listelimages
  - valeursTemperature
  - descriptionAlerte

EXECUTE

Last executed service: listelimages

**listeAlertes**

general.idAlerte	general.nom	general.idDossier	general.date	rank	general.libelle
15	nomInfirmier	3	2002-02-24 00:00:00	0	Anomalie Cathéter
16	medecin1	3	2002-02-24 00:00:00	1	Poids élevé
17	medecin1	3	2002-02-24 00:00:00	2	Tension élevée
18	medecin1	3	2002-02-24 00:00:00	3	Anomalie Cathéter
25	medecin1	3	2002-03-11 00:00:00	4	Température élevée
26	medecin1	3	2002-03-11 00:00:00	5	Poids élevé
27	nomInfirmier	3	2002-03-11 00:00:00	6	Tension élevée
28	nomInfirmier	3	2002-03-11 00:00:00	7	Tension élevée

**listeDossierDialyse**

general.dateCr...	general.modeD...	general.nom	general.prenom	general.typeDia...	general.idDoss...	rank	general.dateMo...
2002-01-01	DPCC	nomPatient2	prenomPatient2	DPA	3	0	2002-01-01

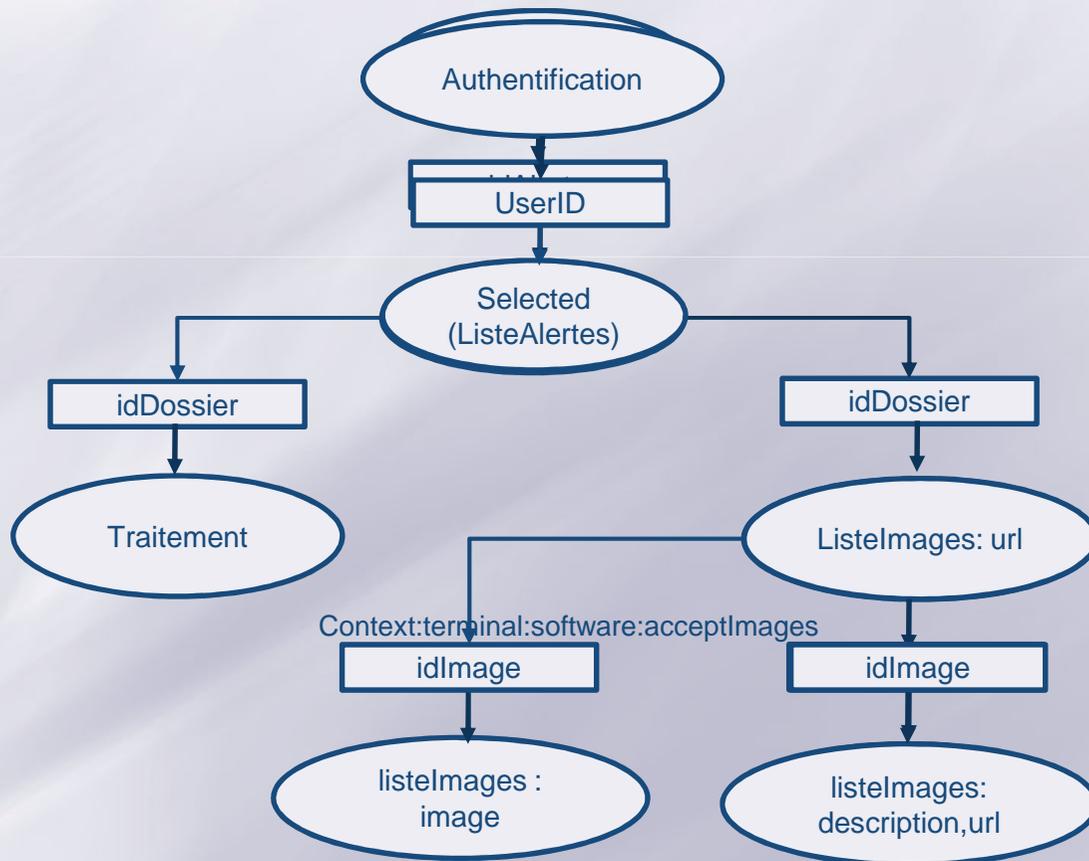
**listelimages**

general.urlImage	general.descriptionImage	general.nomImage	rank
http://localhost:8080/images/abd...	Radio sur l'abdomen du patient ...	radio abdomen	0
http://localhost:8080/images/inte...	analyse après lapremière interve...	intervention 1	1
http://localhost:8080/images/inte...	Analyse après la deuxième interv...	intervention 2	2

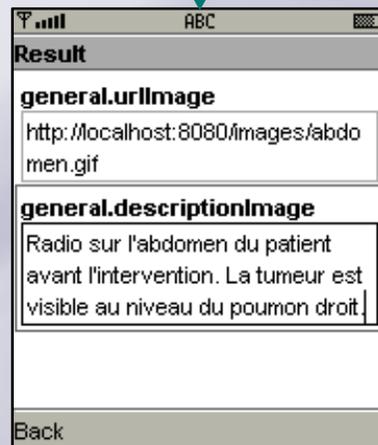
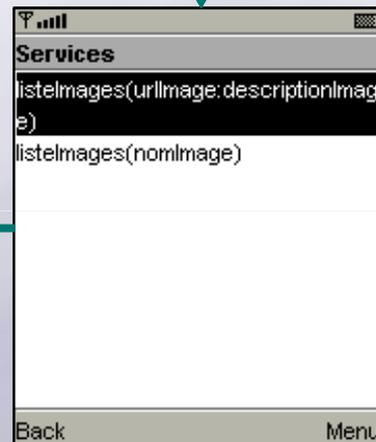
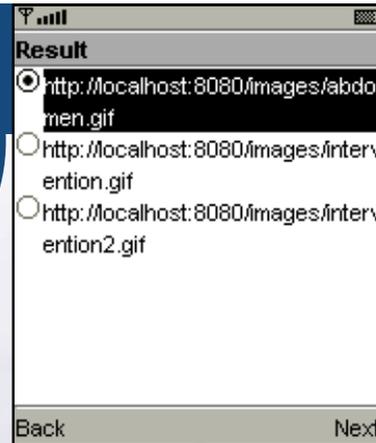


# Réalisations

## adaptation de l'application SICOM



# Réalisations SICOM adapté pour téléphones mobiles



# Conclusions

- ☰ **Stratégie complète et générique d'adaptation du comportement des applications au contexte**
  - Complète: services, données et interface utilisateur
  - Générique: applicable à tout type d'applications
  
- ☰ **Plateforme SECAS garantissant**
  - L'adaptation d'applications existantes à de nouveaux contextes
  - La création incrémentale d'applications sensibles au contexte
  
- ☰ **Facilité d'évolution de notre approche d'adaptation**
  - Intégration facile de nouveaux opérateurs d'adaptation
  - Intégration de modèles de contexte plus évolués

**Merci !**

# Questions ?



# Bibliographie SECAS

- **A Comprehensive Approach to Model and Use Context for Adapting Applications in Pervasive Environments**

*Int. Journal of Systems and Software, Elsevier, 2007. DOI: <http://dx.doi.org/10.1016/j.jss.2007.03.010>*

- **Adaptation in Context-Aware Pervasive Information Systems**

*International Journal of Pervasive Computing and Communications. vol3, 2006*

- **Modeling and Using Context in Adapting Applications to Pervasive Environments**

*IEEE International Conference on Pervasive Services, Lyon, France, June 2006*

- **A Generic Description Language for the Automatic Generation of Pervasive Medical User Interfaces: The SEFAGI Project**

*Health Pervasive Systems Workshop, edited by the IEEE, Lyon, France, June 2006*

- **Service-Oriented Context-Aware Application Design**

*First International Workshop on Managing Context Information in Mobile and Pervasive Environments (MCMP), May 9 2005, Ayia Napa, CYPRUS*

- **SEFAGI: Simple Environment For Adaptable Graphical Interfaces - Generating user interfaces for different kinds of terminals**

*7th International Conference on Enterprise Information Systems (ICEIS), 24-28 May 2005, Miami, USA.*

- **Adaptation des applications médicales à des contextes multiples**

*11 èmes Journées Francophones Informatique Médicale 12 et 13 mai 2005 à Lille*

- **Adaptation des applications au contexte en utilisant les services WEB: Le projet SECAS**

*Deuxièmes Journées Francophones: Mobilité et Ubiquité 2005 Mardi 31 mai - Vendredi 3 juin 2005, Grenoble, France.*

- **Génération et adaptation automatiques des interfaces utilisateurs pour des environnements multi-terminaux**

*Revue Ingénierie des systèmes d'Information, n° spécial systèmes d'information pervasifs, volume 9 - n°2/2004:11-38.*

