

esup-commons : un framework de développement pour le projet ESUP-Portail

Pascal Aubry
sur-connaissant
IFSIC / Université de Rennes 1



<http://sourcesup.cru.fr/esup-commons>

Notes

- Parler de l'ouverture des applications à travers des WS

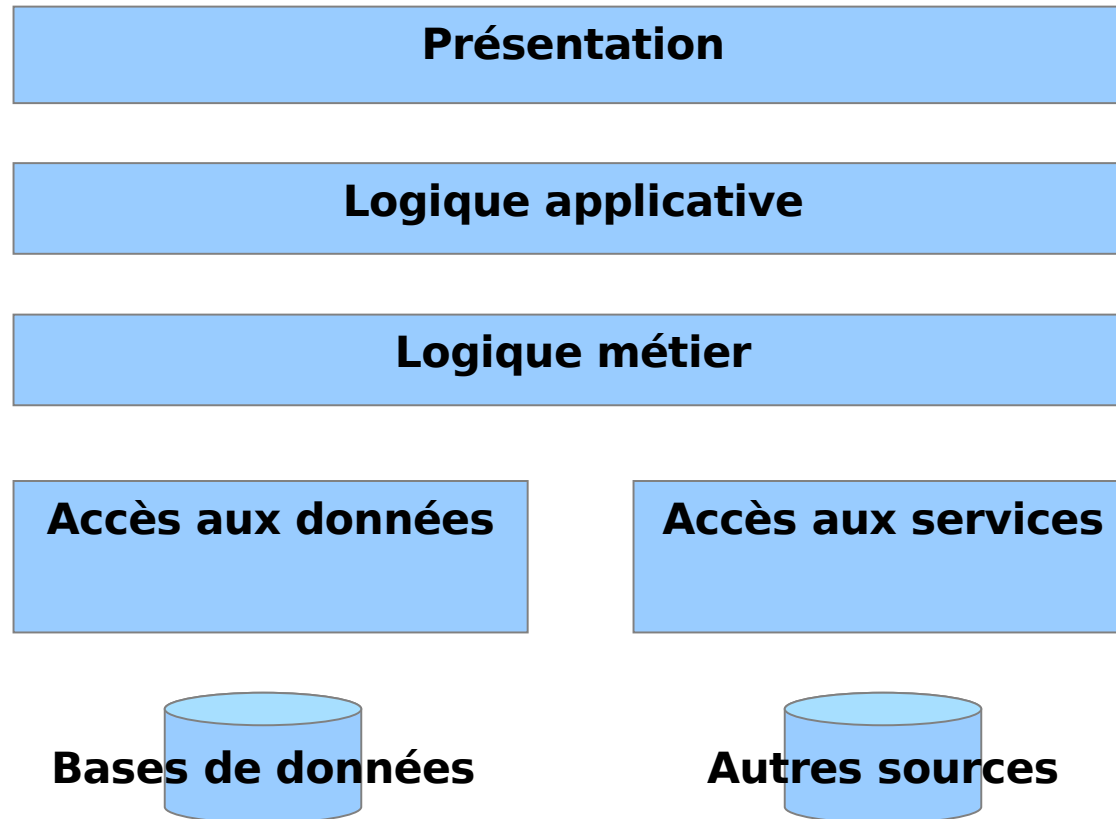
Copyright

- Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
- <http://www.gnu.org/licenses/licenses.html#FDL>

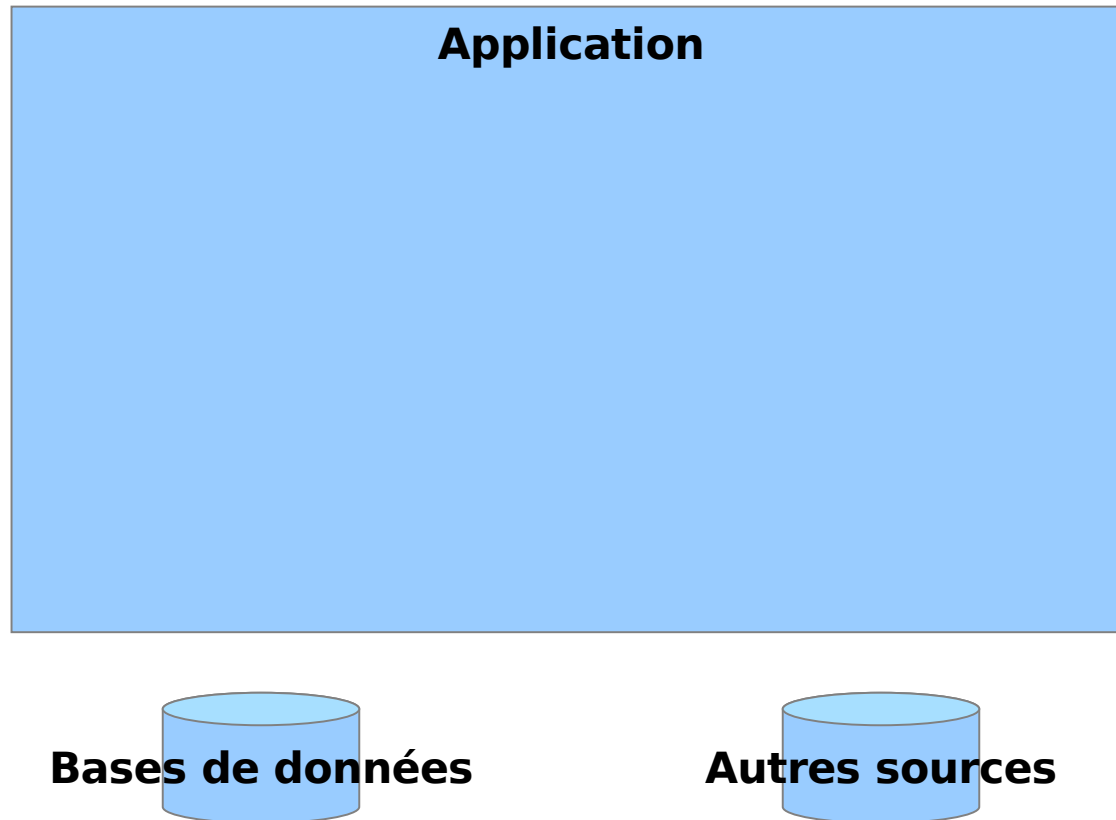
Once upon a time...

- Utilise dès que possible des outils de haut niveau !
- Sépare bien tes couches !
- Abstrait tes objets !
- Relis ton code deux semaines après !

L'idéal standard



Dans la pratique ;-)



Séquence nostalgie

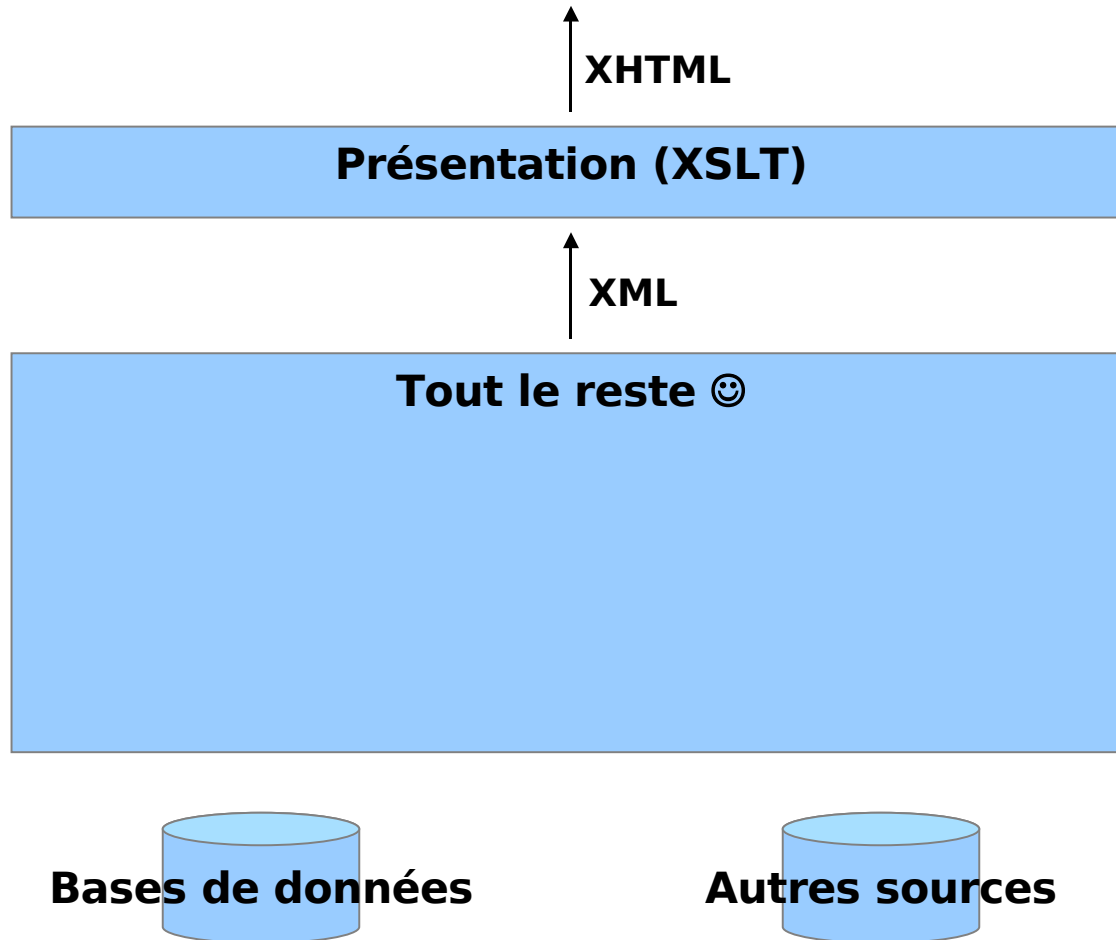
```
// logique applicative
if (!empty($_POST["userId"])) {

    // accès aux données
    $c = DB::Connect("mysql://user:passwd@host/db" );
    $res = $conn->Query(
        "SELECT * FROM user WHERE id = '" + userId + "'");
    $row = $res->FetchRow(DB_FETCHMODE_ASSOC);

    // logique métier
    $displayName = $row["display_name"];

    // présentation
    echo "Hello " . userName . "!";
}
```

En s'appuyant sur l'API uPortal



Les problèmes de cette approche

- Maintenance de XSLT
- L'adhérence de la présentation reste très importante
 - On ne remonte pas les objets métier
 - On produit du XML lié à la présentation
- Tout est mélangé, il faut être sur-connaissant

Le grand test de l'été 2007

- Écrit une requête SQL ?
- Écrit une classe Java ?
- Écrit une requête LDAP ?
- Utilisé un cache ?
- Écrit un fichier XML, XSL ?
- Été voir un cookie de près ?
- Inspecté un fichier WSDL ?
- Modifié une CSS ?
- Affiché la source d'une page HTML ?
- Envoyé un mail en commençant par HELO ?
- Tapé une URL « à la main » ?
- Appuyé sur le bouton « Démarrer » ?

Résultats du test

- Si vous avez répondu plus d'une fois oui,
vous êtes sur-connaissant !
- Sinon, vous vous êtes sans doute trompé de
salle,
c'est pour ça que vous ne pigez rien depuis une
heure
Vous êtes à Paris 5 aux ESUP-Days ;-)

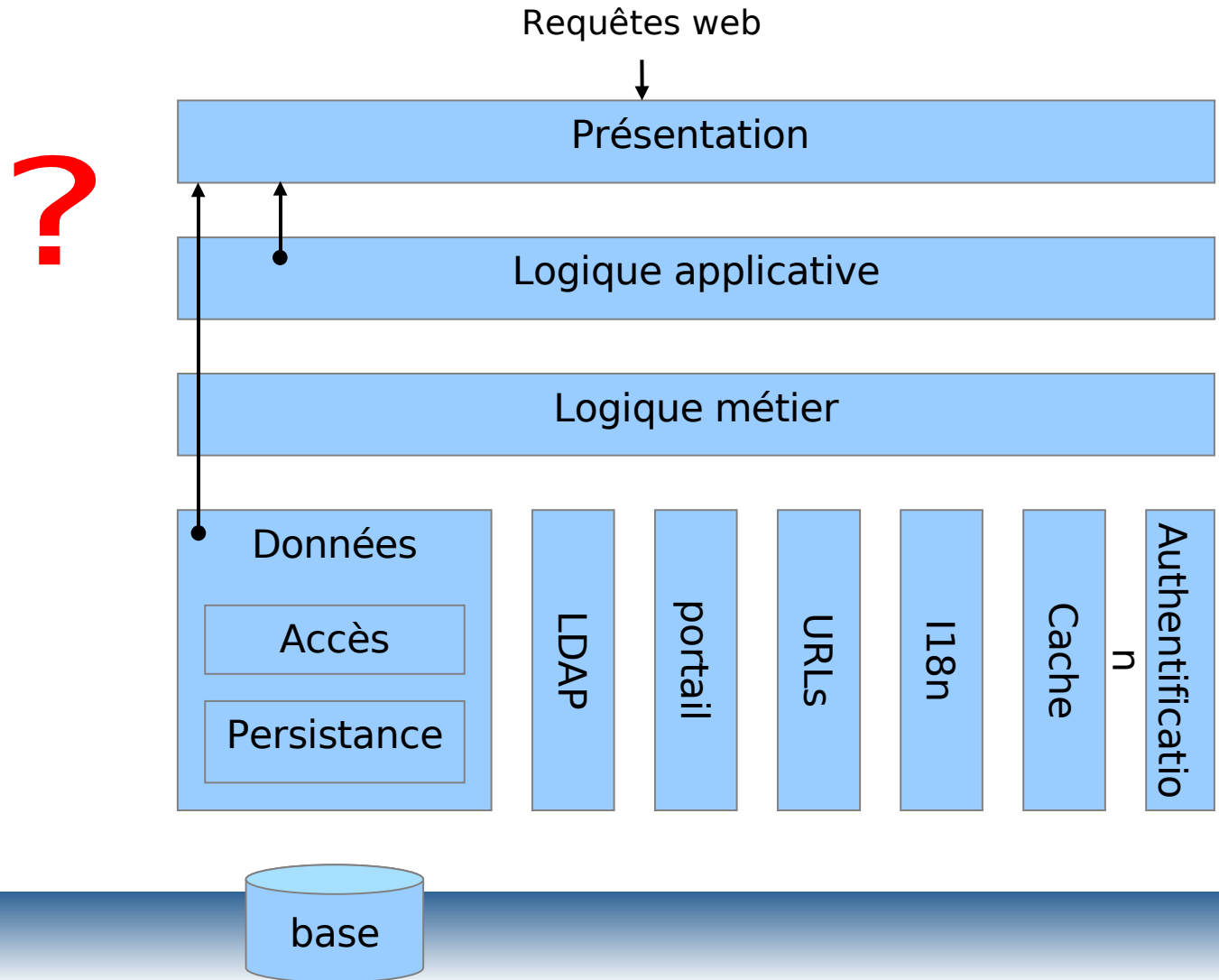
Être ou ne pas être sur-connaissant

- Le problème n'est pas là...
 - Les applications font sensiblement la même chose qu'il y a quelques années, et pourtant nous y arrivions bien...
 - Nous étions sur-connaissants !
- Qu'est ce qui a changé ?
 - Hier : PHP + HTML + SQL + LDAP
 - Aujourd'hui : Java + XML + XSLT + Hibernate/Ibatis + WS + LDAP + i18n + CAS + JSR-168 + Portail + Spring + JSF + ...
- Développeur : un métier à part entière

Alors, faut-il vraiment aller vers ces technos ?

- Simplifier la maintenance
 - Outils de plus haut niveau
 - Séparation des couches
- Faciliter l'adaptation aux configurations locales
 - Injection des données
 - Utilisation d'interfaces
- Permettre l'évolution des fonctionnalités
- Augmenter la productivité
 - Une fois la première marche franchie

L'architecture de esup-commons



Portlet/servlet : même combat

- Nous ne développons pas des portlets ou des servlets, nous développons des applications
- Il est très intéressant de pouvoir faire tourner une application tantôt en servlet, tantôt en portlet
 - Environnement de Développement plus simple en servlet
 - Permet de séparer les problèmes inhérents aux portlets
 - Publication, déploiement, interaction avec le portail
 - Diffusion en quick-start
- Problème avec Spring : deux MVC distincts

Le MVC Spring (portlet)

- import org.springframework.web.**servlet**.mvc.Controller;
import org.springframework.web.**servlet**.ModelAndView;

```
public class ServletController implements Controller {  
    public ModelAndView handleRequest(  
        HttpServletRequest request,  
        HttpServletResponse response)  
    throws ServletException, IOException {  
        logger.info("returning hello view");  
        return new ModelAndView("hello.jsp");  
    }  
}
```


Le MVC Spring (servlet)

- import org.springframework.web.**portlet**.mvc.Controller;
import org.springframework.web.**portlet**.ModelAndView;

```
public class PortletController implements Controller {  
    public ModelAndView handleRequest(  
        PortletRequest request,  
        PortletResponse response)  
        throws ServletException, IOException {  
        logger.info("returning hello view");  
        return new ModelAndView("hello.jsp");  
    }  
}
```

Le MVC JSF

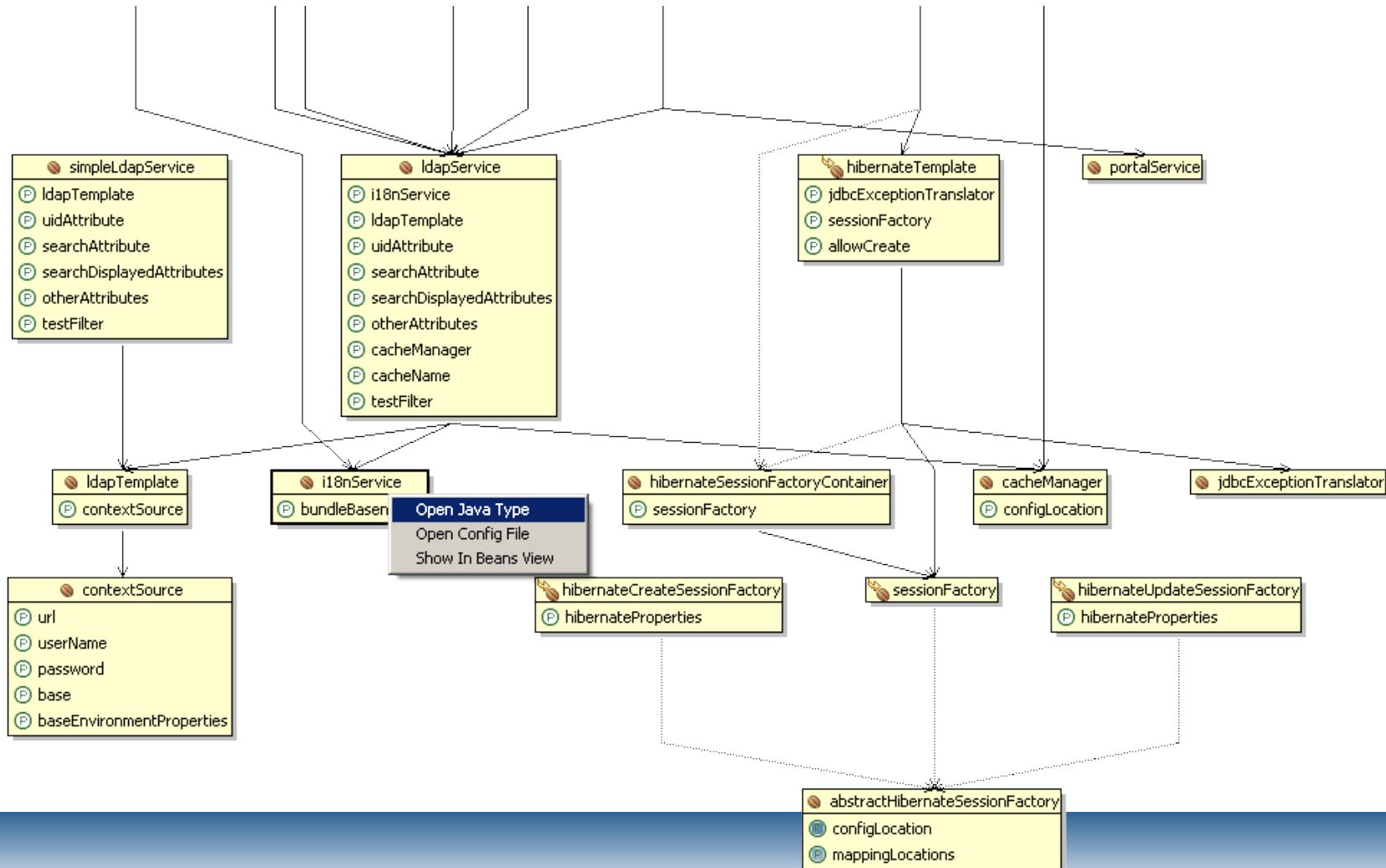
- ```
public class JsfController {
 public String callback() {
 logger.info("returning hello view");
 return "hello";
 }
}
```
- navigation-rules.xml
  - ```
<navigation-rule>  
    <from-view-id>anyView.jsp</from-view-id>  
    <navigation-case>  
        <from-outcome>hello</from-outcome>  
        <to-view-id>hello.jsp</to-view-id>  
    </navigation-case>  
</navigation-rule>
```

Alors, Spring ou JSF ?

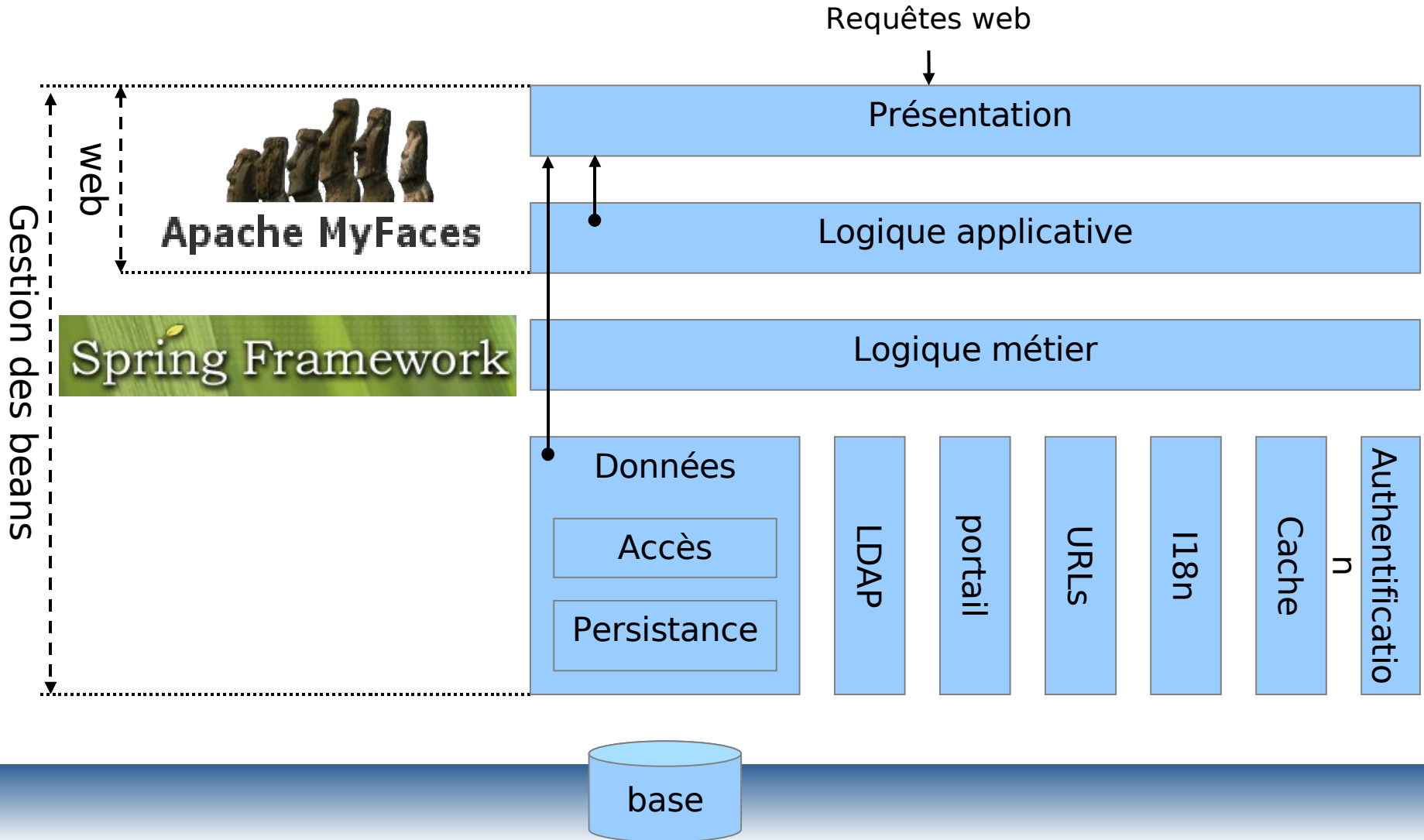
- Les deux !
- Spring pour la gestion des beans
 - De loin le meilleur (JSF le fait également)
 - Outils de développement (Spring IDE)
- JSF pour la logique applicative et la présentation
 - MVC : même code portlet/servlet
 - Présentation : taglib MyFaces + Tomahawk

Spring IDE

- Ⓟ versionupdate
- Ⓟ copyright
- Ⓟ vendor
- Ⓟ latestVersionBaseUrl



L'architecture de esup-commons



Déploiement simplifié (build.properties)

- Build.properties
 - Portlet
 - `deploy.type=portlet`
`deploy.home=C:/uPortal/portlets`
 - Servlet
 - `deploy.type=servlet`
`deploy.home=C:/Tomcat/webapps`
 - Quick-start
 - `quick-start=true`
- Adapter WEB-INF/web.xml (exemples fournis)

Les données : Ibatis ou Hibernate ?

- Peu importe !

- `/**`

- `* The DAO service interface.`

- `*/`

```
public interface DaoService {
```

```
    /**
```

```
        * @param id
```

```
        * @return the User that corresponds to an id.
```

```
        */
```

```
    User getUser(String id);
```

```
}
```

ibatis

- ```
<bean id="daoService"
 class="org.esupportail.example.dao.IbatisDaoServiceImpl" >
 <property
 name="sqlMapClientTemplate"
 ref="sqlMapClientTemplate" />
</bean>
```
- ```
/**
 * See /properties/dao/dao-example.xml
 */
public class HibernateDaoServiceImpl
implements DaoService {
    /**
     * @see DaoService#getUsers()
     */
    public List<User> getUsers() {
        return sqlMapClientTemplate.queryForList("getUsers",
null);
    }
}
```

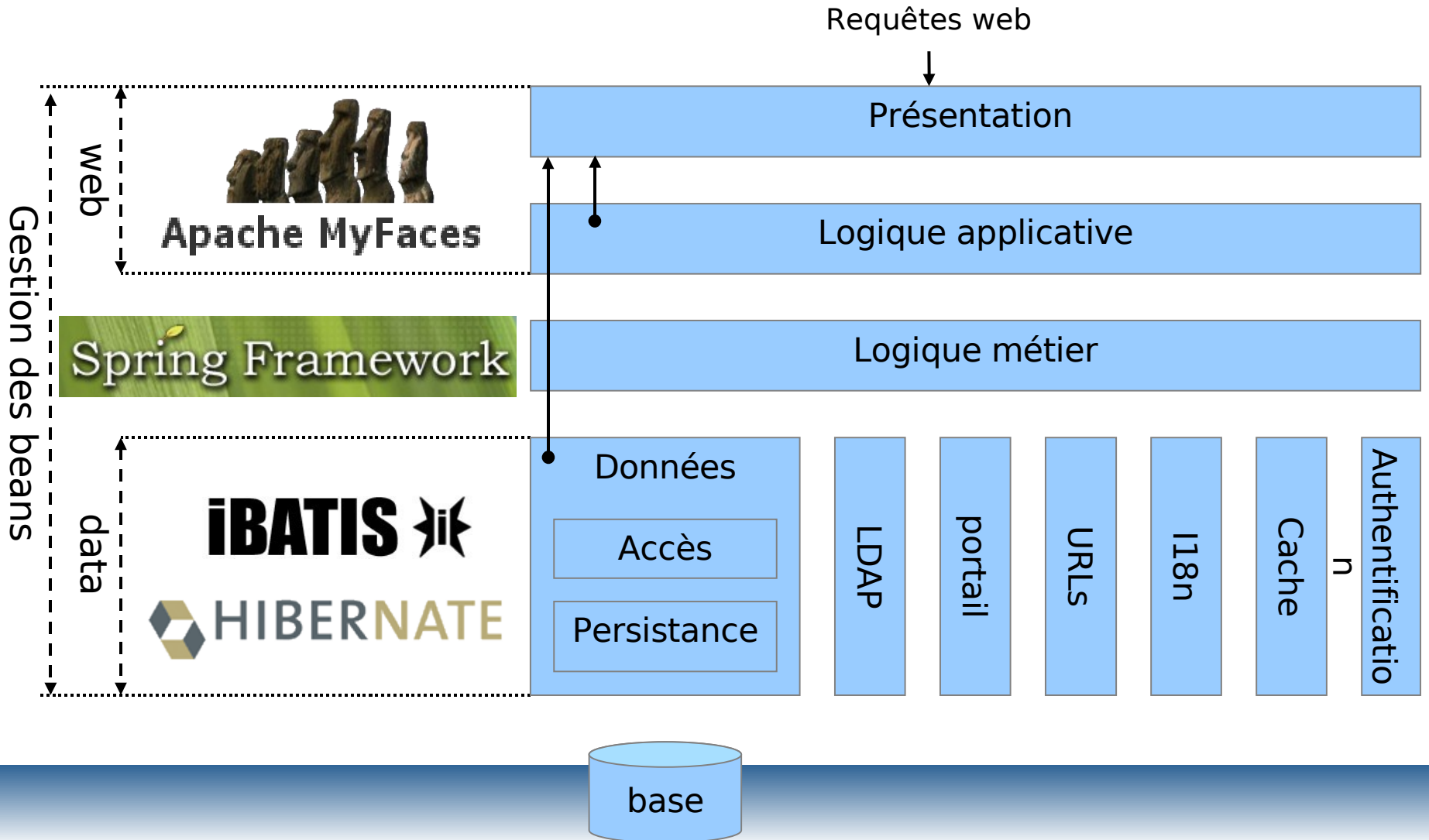

Hibernate

- ```
<bean id="daoService"
class="org.esupportail.example.dao.HibernateDaoServiceImpl" >
 <property
 name="hibernateTemplate"
 ref="hibernateTemplate" />
</bean>
```

- ```
/**
 * See /properties/dao/dao-example.xml
 */
```

```
public class HibernateDaoServiceImpl
implements DaoService {
  /**
   * @see DaoService#getUsers()
   */
  public List<User> getUsers() {
    return getHibernateTemplate().loadAll(User.class);
  }
}
```

L'architecture de esup-commons



Les « plus » de esup-commons

- Gestion des versions
- Gestion des exceptions
- Internationalisation native
- Envoi de courriers électroniques
- Commandes en ligne
- Gestion de l'authentification
- Accès au portail
- Accès à LDAP
- Pagination des données
- Taglib dynamique
- Gestion des URLs directes (*deep linking*)
- Gestion de cache

Intéressant avec Hibernate

- Gestion automatique de la structure de la base
 - Ant init : création
 - Ant upgrade : mise à jour
- Gestion des versions
 - Assurer que les versions de l'application et de la base de données sont les même, ou au moins compatibles
 - Essentiel en environnement cluster

Gestion des exceptions

HTTP Status 500 -

type Exception report

message

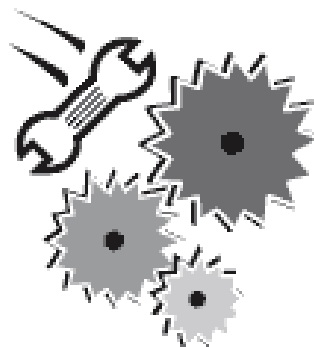
description The server encountered an internal error () that prevented it from fulfilling this request.

exception

```
org.hibernate.TransactionException: JDBC rollback failed
  org.hibernate.transaction.JDBCTransaction.rollback(JDBCTransaction.java:170)
  org.esupportail.commons.dao.HibernateThreadData.closeSession(HibernateThreadData.java:81)
  org.esupportail.commons.dao.HibernateUtils.close(HibernateUtils.java:118)
  org.esupportail.commons.web.servlet.FacesServlet.service(FacesServlet.java:322)
  javax.servlet.http.HttpServlet.service(HttpServlet.java:802)
  org.apache.myfaces.webapp.filter.ExtensionsFilter.doFilter(ExtensionsFilter.java:97)
  org.apache.myfaces.webapp.filter.ExtensionsFilter.doFilter(ExtensionsFilter.java:144)
```

note The full stack trace of the root cause is available in the [Apache Tomcat/5.5.17 logs](#).

Apache Tomcat/5.5.17



Error:

This channel failed to render



Refresh the Channel



Reboot the Channel

Gestion des exceptions

Exception report

Restart the application

The following exception was thrown:

Application	esup-repository
Version	0.11.0
Server	148.60.10.37 (caillou.ifsic.univ-rennes1.fr)
Date	jeu. 25-01-2007 12:17:43
User	unknown
Portal	unknown
Client	148.60.10.131 (paubry.ifsic.univ-rennes1.fr)
Query string	unknown
User agent	Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.8.0.9) Gecko/20061206 Firefox/1.5.0.9

Exception details

Name	NoSuchMethodException
Message	org.esupportail.repository.web.controllers.AboutController.throwException()
Short stack trace	<pre> javax.faces.FacesException: Error calling action method of component with id _idJsp25:_idJsp26 caused by: javax.faces.el.EvaluationException: Exception while invoking expression #{aboutControlle caused by: java.lang.NoSuchMethodException: org.esupportail.repository.web.controllers.AboutContro javax.faces.FacesException: Error calling action method of component with id _idJsp25:_idJsp26 org.apache.myfaces.application.ActionListenerImpl.processAction(ActionListenerImpl.java:69) javax.faces.component.UICommand.broadcast(UICommand.java:106) javax.faces.component.UIViewRoot._broadcastForPhase(UIViewRoot.java:94) javax.faces.component.UIViewRoot.processDecodes(UIViewRoot.java:136) org.apache.myfaces.lifecycle.LifecycleImpl.applyRequestValues(LifecycleImpl.java:218) </pre>

Request parameters

_idJsp25:_idJsp26	= [Test exception handling]
_idJsp25:_idcl	= []
_idJsp25_SUBMIT	= [1]
jsf_sequence	= [2]

System properties

awt.toolkit	= [sun.awt.windows.WToolkit]
catalina.base	= [C:\devel\esup-repository/apache-tomcat-5.5.17]
catalina.home	= [C:\devel\esup-repository/apache-tomcat-5.5.17]
catalina.useNaming	= [true]

Cookies

JSESSIONID	= [84A501ACD249D482287B82F926D3622C]
repository-user	= [pascal.aubry@univ-rennes1.fr:faed3d904bd017b0101a7a795bfaaef]

Hibernate properties

hibernate.bytecode.use_reflection_optimizer	= [false]
hibernate.version	= [3.2 cr2]

Request headers

accept-charset	= ISO-8859-1,utf-8;q=0.7,*;q=0.7
accept-encoding	= gzip,deflate
accept-language	= en-gb,en;q=0.5
accept	= text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,te
connection	= keep-alive

Session attributes

aboutController	= [AboutController#2721032]
administratorsController	= [AdministratorsController#31231651[em userToDelete=null, paginator=FixedQueryHibernatePaginator#1838 currentPage=0]]]
deepLinkingRedirector	= [org.esupportail.repository.web.deepLinking.DeepLinkingRedirector]

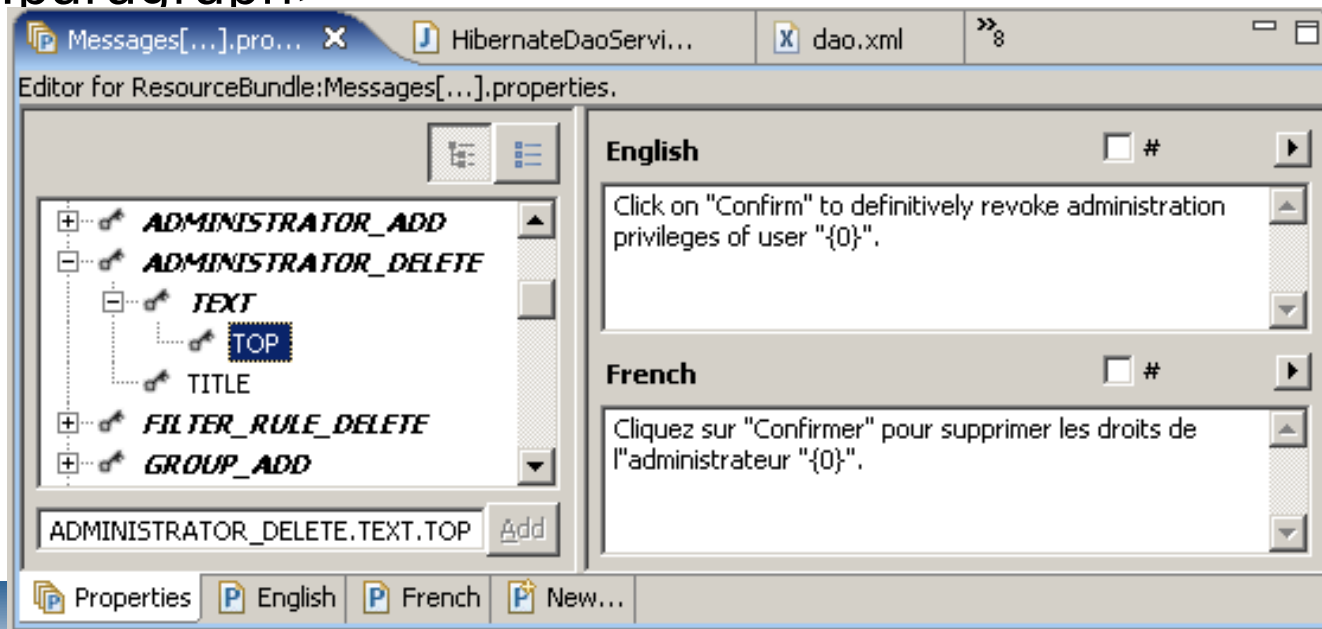
- Envoi possible par email

Internationalisation native

- Utilisation de bundles (RBE)

- `<e:paragraph`

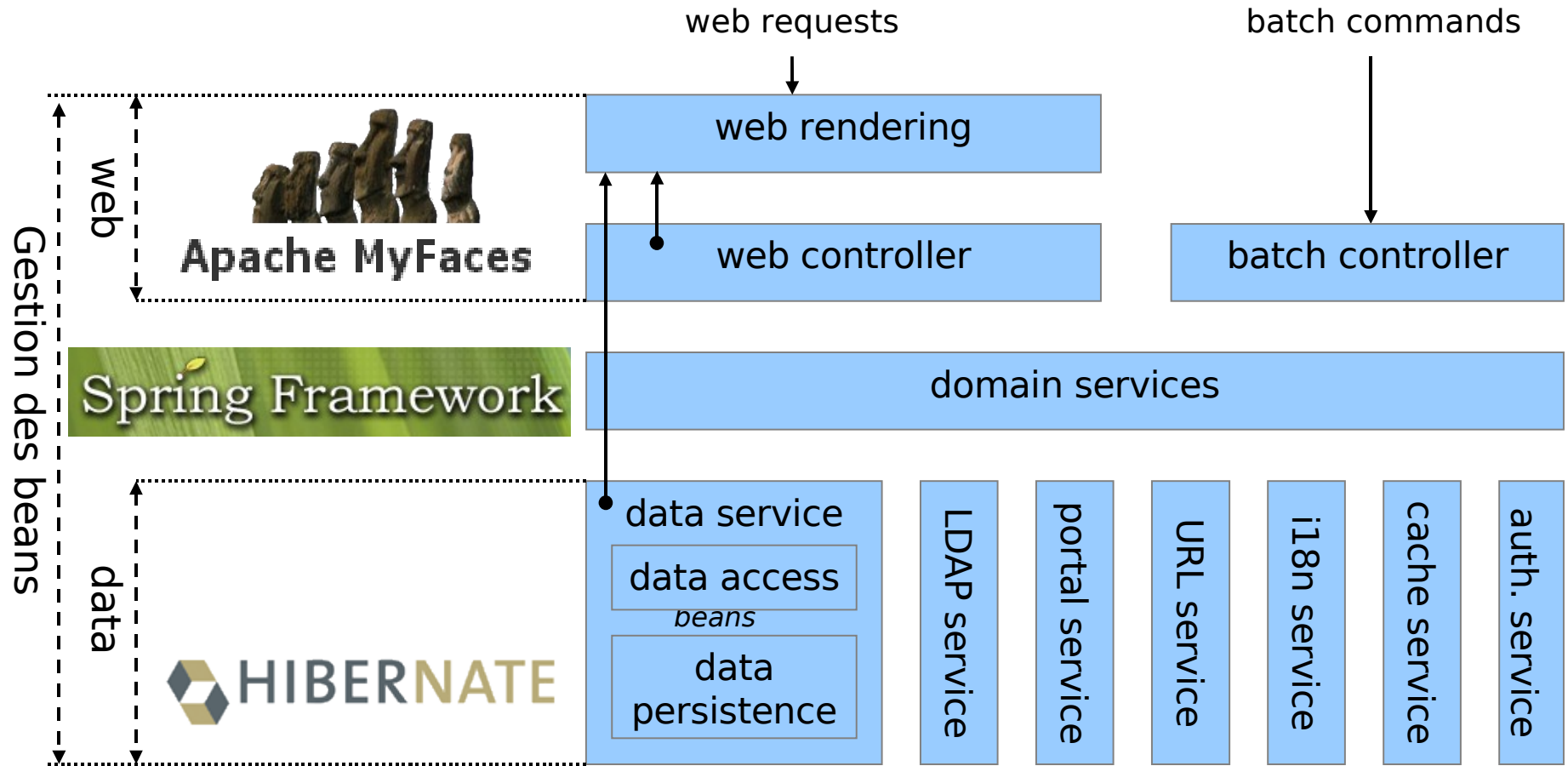
```
value="# {msgs['ADMINISTRATOR_DELETE.TEXT.TOP']}">  
  <f:param value="# {controller.user.displayName}" />  
</e:paragraph>
```



Envoi de courriers électroniques

- Pour envoyer n'importe quoi...
 - Rapports d'exception, alertes, informations de connexion
- Possibilité d'envoi en asynchrone
 - Pour éviter les timeouts
- Envoi plain/text et text/html
- Possibilité d'interception en période de test

Commandes en ligne



- Ouverture de l'application grâce à des web services



- Gestion des exceptions et de l'accès aux données

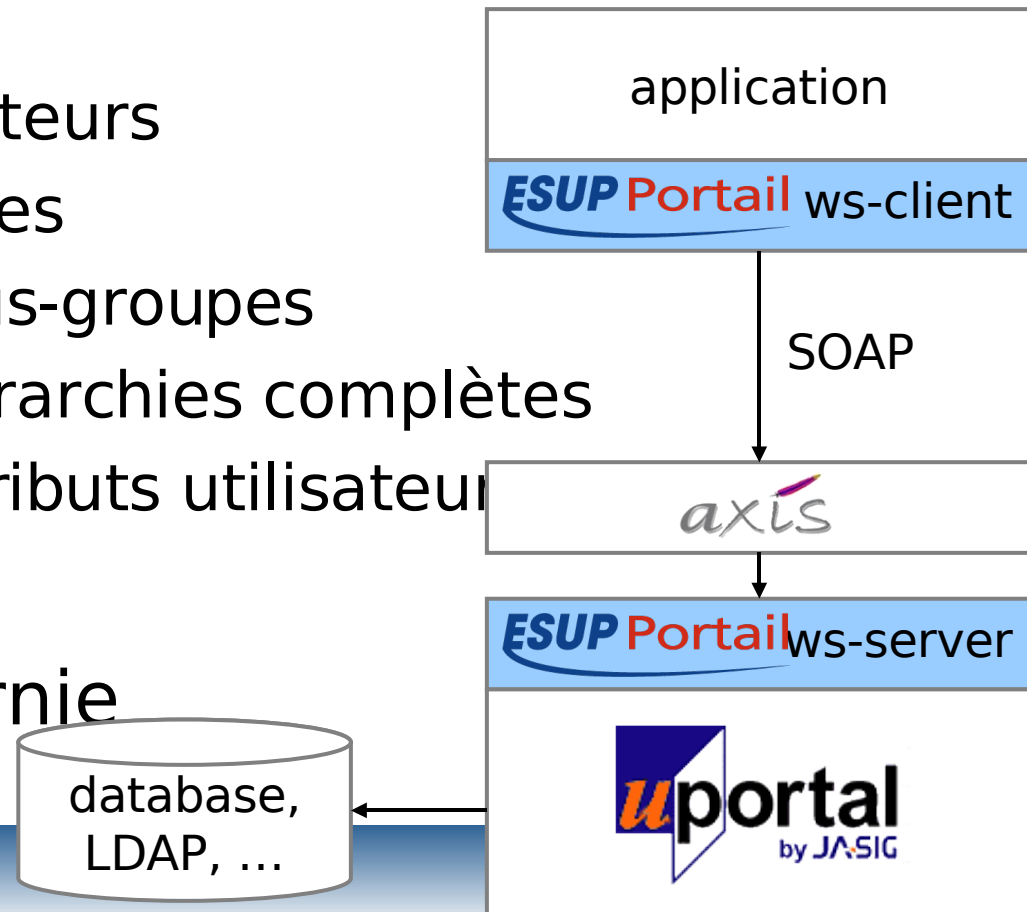
Gestion de l'authentification

- Externalisation vers un service d'authentification
- Implémentations pré-définies
 - Pour un filtre CAS
 - Pour les portlets
 - Pour une authentification manuelle (applicative)
- Classes pour développer « offline »

Accès au portail

- esup-portal-ws
 - Recherche des utilisateurs
 - Recherche des groupes
 - Récupération des sous-groupes
 - Récupération des hiérarchies complètes
 - Récupération des attributs utilisateur

- Implémentation fournie pour uPortal 2.5



Accès à LDAP

- Pour récupérer les attributs des utilisateurs
 - Pas seulement l'utilisateur connecté
- Cache des résultats
- Pour vérifier et appliquer des filtres LDAP
- Pour rechercher une personne dans l'annuaire
 - Beaucoup plus simple que le canal dédié
- Basé sur LdapTemplate

Statistiques LDAP

Nombre de requêtes : 14
Requêtes cachées : 9/14 (64%)
Opérations LDAP : 5/14 (35%)
Opérations réussies : 5/5 (100%)
Erreurs de connexion : 0/5 (0%)
Erreurs de filtre : 0/5 (0%)
Autres erreurs : 0/5 (0%)

Recherche LDAP

Recherche un utilisateur dans l'annuaire LDAP

Rechercher la chaîne :

esup-print v0.5.0

Resultat(s) de la recherche LDAP

Veuillez choisir parmi les utilisateurs listés sur cette page en cliquant sur la personne voulue :

Id	Attributs
26003775	cn= Bourges Caroline uriComposante= 930 uriTypeEntree= etu displayName= Caroline Bourges
27001051	cn= Bourges Pierre uriComposante= 941 uriTypeEntree= etu displayName= Pierre Bourges
27005655	cn= Bourges Gwendal uriComposante= 931 uriTypeEntree= etu displayName= Gwendal Bourges
27007822	cn= Bourges Marie uriComposante= 930 uriTypeEntree= etu displayName= Marie Bourges
bourges	cn= Bourges Raymond uriComposante= 957 uriTypeEntree= pers displayName= Raymond Bourges
kbourges	cn= Bourges Karen uriComposante= 902 uriTypeEntree= pers displayName= Karen Bourges

esup-print v0.5.0

Pagination

- Ergonomie
 - Pour présenter de longues listes de résultats
- Performance
 - Pour ne pas charger tous les résultats d'une requête alors que l'on n'en affiche que certains

Logs

Logs [21 - 25] of 122 Logs per page:

Date	User	Class	Operation
01-12-2006	paubry (Pascal Aubry)	Storage	storage update: Storage#21998437[id=[3], label=[CRI WebDAV], xlabel=[Serveur WebDAV du CRI], adaptor=[org.esupportail.diskquota.adaptors.webdav.Webdav], properties=[null]]
01-12-2006	paubry (Pascal Aubry)	Storage	storage update: Storage#7152460[id=[3], label=[CRI WebDAV], xlabel=[Serveur WebDAV du CRI], adaptor=[org.esupportail.diskquota.adaptors.webdav.Webdav], properties=[null]]
01-12-2006	paubry (Pascal Aubry)	Storage	storage update: Storage#27306986[id=[3], label=[CRI WebDAV], xlabel=[Serveur WebDAV du CRI], adaptor=[org.esupportail.diskquota.adaptors.webdav.Webdav], properties=[null]]
01-12-2006	paubry (Pascal Aubry)	Storage	storage creation: Storage#16757505[id=[3], label=[CRI WebDAV], xlabel=[Serveur WebDAV du CRI], adaptor=[null], properties=[null]]
30-11-2006	paubry (Pascal Aubry)	FilterRule	filter rule creation: FilterRule#2161302[id=[27], order=[6], filter=[null], value=[0], unit=[M], enabled=[false], comment=[null], date=[2006-11-30 17:05:08.967], user= User#21041412[id=[paubry], displayName=[Pascal Aubry], admin=[true], language=[en]], storage=[Storage#1161839[id=[1], label=[IFSIC], xlabel=[filer IFSIC], adaptor=[fr.univrennes1.ifsic.diskquota.adaptors.networkAppliance.NetworkAppliance], properties=[server=thot.ifsic.univ-rennes1.fr file=t:/quotas file.header=t:/quotas.header file.footer=t:/quotas.footer attr.homedir=homeDirectory attr.employeeType=uriTypeEntree attr.employeeType.staff=pers volume.staff=/vol/vol0 volume.other=/vol/vol0]]]]

Un taglib JSF

- **<%@ taglib
uri="http://commons.esup-portail.org"
prefix="e"%>
<e:page
stringsVar="msg"
menuItem="welcome"
locale="# {bean.locale}"
authorized="# {bean.allowed}">
...
</e:page>**
- **Les tags sont configurés dynamiquement par un bean (TagsConfigurator)**

Gestion des URLs directes (*deep linking*)

- Génération des URLs directes, en portlet et servlet
 - `https://sso-cas.univ-rennes1.fr/login?service=http%3A%2F%2Flocalhost%3A8080%2FLogin%3FuP_fname%3Desup-diskquota%26uP_args%3DcGFnZT13ZWxjb21l`
- Prise en charge de ces URLs pour des liens directs vers certaines pages de l'application

Gestion des caches

- Disponibilité native de caches
- Gains en performance
 - Requêtes LDAP
 - Requêtes au portail
 - Envoi des rapports d'exception
- Basé sur EhCache

Comment démarrer ?

- **Prise en main**
 - Esup-example : une application d'exemple, pour voir comment on configure une application bâtie sur esup-commons
- **Démarrage d'un développement**
 - Esup-blank : une application blanche qui fournit une application blanche, canevas d'un nouveau développement
 - Ant rename-application
- **Accompagnement**
 - Liste commons-devel@esup-portail.org
 - Formation fin 1er trimestre 2007

Utilisation actuelle

- **Achevé**
 - Esup-diskquota (gestionnaire de quotas disque)
- **En cours**
 - Esup-print (gestionnaire du contrôle d'accès aux imprimantes et des crédits d'impression)
 - Esup-lecture (lecture des annonces)
- **A venir**
 - Esup-helpdesk (suivi des demandes utilisateurs)
 - ...

esup-commons : la méthodologie

- Documentation pas-à-pas de la procédure d'installation
 - Installation et configuration d'Eclipse
 - Callisto, Checkstyle, RBE, Spring IDE...
 - Utilisation de SubVersion (sourcesup)
- Récupération des fichiers de configuration des versions antérieures pour faciliter les mises à jour
 - Ant recover-config

Bénéfices attendus de esup-commons

- Uniformiser les installations d'applications
 - Mêmes fichiers de configuration
- Faciliter les contributions
 - Utilisation de standards uniquement
- Gagner en productivité
 - Framework prêt à utiliser
- Faciliter la mobilité des développeurs
 - Facile de passer d'un projet à un autre
- Fiabiliser les applications
 - Outils de haut niveau

Mais au fait, vous vous dites sûrement...

- Combien de temps faut-il pour se mettre à une usine comme ça ?
 - Pas tant que l'on croit
- N'oublions pas qu'avec d'autres frameworks, ou sans framework du tout, nous étions tout aussi sur-connaissants...
 - Sinon plus
- Ce n'est pas la solution qui est compliquée, c'est le problème
 - *« Ne me dites pas que ce problème est difficile. S'il n'était pas difficile, ce ne serait pas un problème »
Ferdinand Foch*

Vous trouvez tout cela bien compliqué...

- Ce n'est pas la solution qui est compliquée c'est le problème
- « *Ne me dites pas que ce problème est difficile. S'il n'était pas difficile, ce ne serait pas un problème* » *Ferdinand Foch*

Résumé

- - Spring
 - + JSF
 - + Ibatis/Hibernate
 - + services
 - + méthodologie

 - = esup-commons

- <http://sourcesup.cru.fr/esup-commons>