



Framework Corporatiu J2EE

Servei de Llistats

Versió 1.4

Barcelona, 13 / abril / 2007



Històric de modificacions

Data	Autor	Comentaris	Versió
09/12/2005	Atos Origin, sae openTrends	Versió inicial del document	1.0
13/02/2007	Atos Origin, SAE	Versió 1.2 d'OpenFrame	1.2
16/03/2007	Atos Origin, SAE	Versió 1.3 d'OpenFrame	1.3
13/04/2007	Atos Origin, SAE	Versió 1.4 d'OpenFrame	1.4

Llegenda de Marcadors



Índex

1.	INTRODUCCIÓ	4
1.1.	PROPÒSIT	4
1.2.	CONTEXT I ESCENARIS D'ÚS	4
1.3.	VERSIONS I DEPENDÈNCIES	4
1.3.1.	<i>Versions</i>	5
1.3.2.	<i>Dependències Bàsiques</i>	5
1.4.	A QUI VA DIRIGIT	5
1.5.	DOCUMENTS I FONTS DE REFERÈNCIA	5
1.6.	GLOSSARI	5
2.	DESCRIPCIÓ DETALLADA	7
2.1.	ARQUITECTURA I COMPONENTS	7
2.1.1.	<i>Interfícies i components genèrics</i>	7
2.1.2.	<i>Implementació de les interfícies</i>	8
2.1.3.	<i>Llibreria de tags</i>	9
2.2.	INSTAL·LACIÓ I CONFIGURACIÓ	12
2.2.1.	<i>Instal·lació</i>	12
2.2.2.	<i>Configuració</i>	12
2.3.	UTILITZACIÓ DEL SERVEI	19
2.4.	EINES DE SUPORT	20
2.5.	INTEGRACIÓ AMB ALTRES SERVEIS	20
2.5.1.	<i>Servei multi idioma</i>	20
2.5.2.	<i>Servei de persistència</i>	20
2.6.	PREGUNTES FREQUÈNTS	20
3.	EXEMPLES	21
4.	ANNEXOS	23

1. Introducció

1.1. Propòsit

El servei de llistats permet la presentació de llistats HTML en el que es permet a l'usuari:

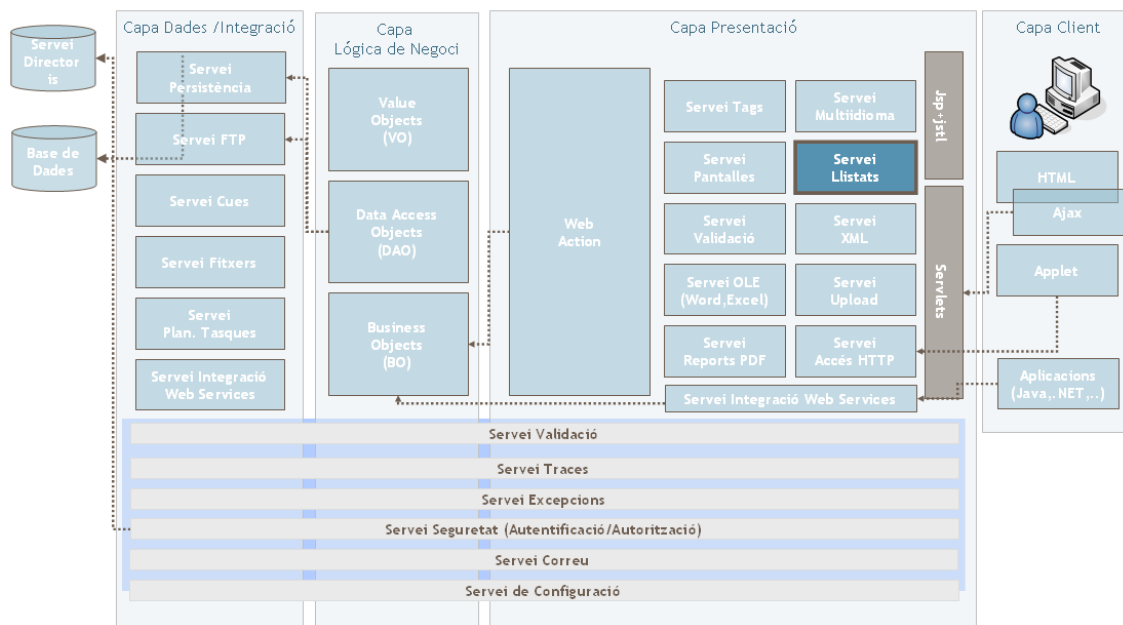
- Ordenar per columna de manera ascendent o descendent.
- Paginació dels resultats i navegació per pàgines mitjançant “avanç” i “retrocés” (primera, última, següent, anterior), i per accés directe a un número de pàgina.
- Presentar un número determinat de resultats per pàgina.
- Acotar el número màxim de resultats a mostrar.

A més, proporciona ajut a l'usuari, presentant:

- Rollover de la fila al passar el ratolí per sobre.
- Desactivació dels botons de navegació depenent de la pàgina en la que s'estigui ubicat.
- Indicació de la columna per la que s'està ordenant mitjançant icones, mostrant si és ascendent o descendent.
- Look & feel configurable.

1.2. Context i Escenaris d'Ús

El servei de llistats es troba dins dels serveis de Propòsit General de openFrame.



1.3. Versions i Dependències

En el present apartat es mostren quines són les versions i dependències necessàries per fer ús del Servei.



1.3.1. Versions

No s'han produït canvis respecte la versió 1.3.

1.3.2. Dependències Bàsiques

Nom	Tipus	Versió	Descripció
Valuelist	jar	0.1.7_20051027	http://valuelist.sourceforge.net/
Log4j	jar	1.2.12	
Commons-logging	jar	1.0.4	http://jakarta.apache.org/commons
Spring	jar	1.2.5	http://www.springframework.org
Hibernate	jar	3.0	http://www.hibernate.org/
Jakarta POI	jar	3.0-alpha1-20050704	http://jakarta.apache.org/poi/
Itext	jar	1.3.1	http://www.lowagie.com/iText/
OpenFrame-services-i18n	jar	1.0-SNAPSHOT	

1.4. A qui va dirigit

Aquest document va dirigit als següents perfils:

- Programador. Per conèixer l'ús del servei
- Arquitecte. Per conèixer quins són els components i la configuració del servei
- Administrador. Per conèixer com configurar el servei en cadascun dels entorns en cas de necessitat

1.5. Documents i Fonts de Referència

- [1] Value List <http://valuelist.sourceforge.net/overview.html>
- [2] Hibernate <http://www.hibernate.org>
- [3] Jakarta POI <http://jakarta.apache.org/poi/>
- [4] iText <http://www.lowagie.com/iText/>

1.6. Glossari

Log4j

Un dels framework de traces més extès. Es basa en l'ús de Appenders, Categories i Layouts. Veure l'annex per a més informació.

Value List



Llibreria de tags que ens permet la definició, execució i presentació de queries contra una base de dades

Jakarta POI

Llibreria que ens permet la generació de documents en format (principalment) Microsoft Word i Microsoft Excel

iText

Llibreria que ens permet la generació de fitxers PDF 'al vol', és a dir, sense cap plantilla predefinida sinó mitjançant la programació

Hibernate

Llibreria de persistència de gran rendiment, que permet desenvolupar les classes persistents amb un idioma orientat a objectes.

2. Descripció Detallada

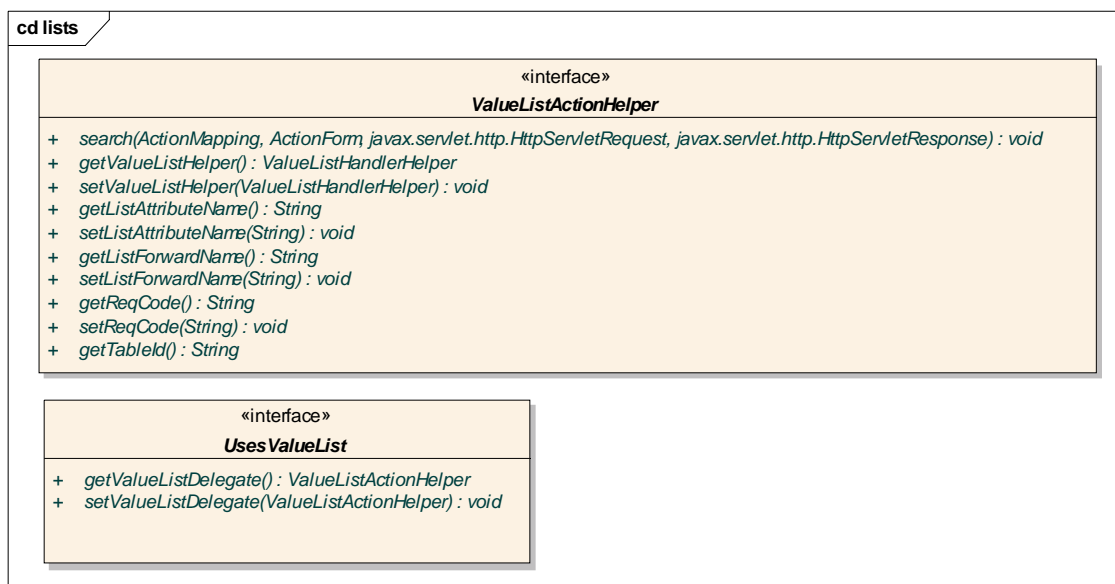
2.1. Arquitectura i Components

Existeixen tres tipus de components. Podem classificar-los en:

- Interfícies i components genèrics.
- Implementació de les interfícies.
- Llibreria de tags per a la presentació de resultats.

2.1.1. Interfícies i components genèrics

El servei de llistats defineix les següents interfícies:



Component	Package	Descripció
UsesValueList	net.opentrends.openframe.services.web.lists	Interfície bàsica que permet identificar una 'Action' d'Struts com 'de cerca'
ValueListActionHelper	net.opentrends.openframe.services.web.lists	<p>Ofereix entre d'altres (veure la documentació disponible al Javadoc per a més referència):</p> <ul style="list-style-type: none"> • public void search(ActionMapping, ActionForm, HttpServletRequest, HttpServletResponse): executa la búsqueda



2.1.2. Implementació de les interfícies

El servei de llistats defineix les següents implementacions:



Component	Package	Descripció
SimpleValueListActionHelper	net.opentrends.openframe.services.web.lists.impl	Implementa la búsqueda amb la llibreria de persistència Hibernate i el framework Struts.



Component	Package	Descripció
PdfDisplayProvider	net.opentrends.openframe.services.web.vlh.tag.support.	Permet l'exportació dels resultats de la búsqueda en format PDF
ExcelDisplayProvider	net.opentrends.openframe.services.web.vlh.tag.support.	Permet l'exportació dels resultats de la búsqueda en format Excel

2.1.3. Llibreria de tags

Els tags que s'utilitzen són els propis de la llibreria Value List. Per a més informació consultar la internet la documentació disponible

<vlh:root/>

Aquest tag és obligatori com a element pare per la resta de tags

Atribut	Descripció
value	Nom de la variable on es troben les dades (típicament "list").

cd support	<div><div>PdfDisplayProvider<i>DisplayProvider</i><ul style="list-style-type: none">+ doesIncludeBodyContent(): boolean+ getMimeType(): String+ getHeaderRowPreProcess(): String+ getHeaderRowPostProcess(): String+ getHeaderCellPreProcess(ColumnInfo, ValueListInfo): String+ getHeaderCellPostProcess(): String+ getHeaderLabel(ColumnInfo, TableInfo, ValueListInfo, Map): String+ getRowPreProcess(Attributes): String+ getRowPostProcess(): String+ getCellPreProcess(Attributes): String+ getCellPostProcess(): String+ emphase(String, String, String): String# generateHeaders(): void- getCell(String): Cell+ getInstance(HttpServletRequest): PdfDisplayProvider- getTableSize(): int+ buildDocument(Document): void+ getI18nService(): I18nService+ setI18nService(I18nService): void+ getLogService(): LoggingService+ setLogService(LoggingService): void+ getSkipColumns(): String[]+ setSkipColumns(String[]): void</div><div>ExcelDisplayProvider<i>DisplayProvider</i><ul style="list-style-type: none">+ doesIncludeBodyContent(): boolean+ getInstance(HttpServletRequest): ExcelDisplayProvider+ buildWorkbook(HSSFWorkbook): void+ getMimeType(): String+ getHeaderRowPreProcess(): String+ getHeaderRowPostProcess(): String+ getHeaderCellPreProcess(ColumnInfo, ValueListInfo): String+ getHeaderCellPostProcess(): String+ getHeaderLabel(ColumnInfo, TableInfo, ValueListInfo, Map): String+ getRowPreProcess(Attributes): String+ getRowPostProcess(): String+ getCellPreProcess(Attributes): String+ getCellPostProcess(): String+ emphase(String, String, String): String# generateContent(HSSFWorkbook): void# generateHeaders(HSSFWorkbook): void+ getI18nService(): I18nService+ setI18nService(I18nService): void+ getLogService(): LoggingService+ setLogService(LoggingService): void+ getSkipColumns(): String[]+ setSkipColumns(String[]): void</div></div>
------------	---



url	L'URL base per realitzar els posts a un action. Tots els links generats a la taula de resultats es muntaran a partir d'aquest atribut.
includeParameters (opcional)	Atribut en el que inclourem sempre el "reqCode" i la resta de paràmetres pels que ha pogut haver estat filtrat en el <i>Value List</i> . Els paràmetres es separen mitjançant el caràcter pipe " ".
configName	Nom del bean de configuració (típicament vlConfig).

<vlh:retrieve/>

Aquest tag és opcional ja que la informació ve implícita segons l'Action executat. S'utilitza si per una vista en concret es vol utilitzar un altre Value List del definit per defecte.

Atribut	Descripció
name	Nom de del adaptador que retorna les dades.

<vlh:paging/>

Tag que genera el panell per controlar la paginació dels resultats i navegació per pàgines mitjançant "avanç" i "retrocés" (primera, última, següent, anterior), i per accés directe a un número de pàgina.

Important

Aquest tag requereix per a que funcioni correctament, que en el fitxer referenciat com a "messageResource" s'especifiquin les següents "keys":

```
paging.text.totalRow={0} Total
paging.text.pageFromTotal= <b>{0}</b> of {1} page(s)
paging.first(off)=
paging.first(on)=
paging.previous(off)=
paging.previous(on)=
paging.forward(off)=
paging.forward(on)=
paging.last(off)=
paging.last(on)=
paging.delim=
paging.focus(on)=
paging.focus(off)=
paging.focus(disabled)=
paging.focus(error)=
```

<vlh:row/>

La funció d'aquest tag és la d'iteració dels resultats.

Atribut	Descripció
bean	Nom de la clau del PageContext on es troben les dades (típicament "job").
display (opcional)	Nom de del adaptador que retorna les dades (en el cas de que s'utilitzi l'adaptador per defecte de l'action tindrà el valor "<%=(String)request.getAttribute("subReqCode")%>").

<vlh:attribute/>

Aquest tag afegeix al tag pare atributs HTML.



Atribut	Descripció
name	Nom de l'atribut HTML
value	Valor de l'atribut HTML

<vlh:column/>

Aquest tag renderitza una columna per cada registre iterat.

Atribut	Descripció
property	Nom de la propietat a mostrar. Les associacions amb altres taules es poden definir de manera implícita de la manera: "property="productid.id""
sortable (opcional)	Defineix si el resultat està ordenat per aquest camp i amb quin ordre ("asc": ascendent, "desc": descendent). En el cas de no incloure aquest atribut no es podrà ordenar els resultats per aquesta columna.
titleKey	Nom de la columna definit en el "message resources".

<vlh:controls/>

Aquest tag renderitza una columna per incloure controls sobre cada registre (eliminar, modificar, etc.)

Atribut	Descripció
titlekey	Nom de la columna definit en el "message resources".

<vlh:action/>

Aquest tag defineix una URL per realitzar addicionals post per diferents accions segons el que es vulgui realitzar.

Atribut	Descripció
url (opcional)	Si no s'inclou s'utilitza l'URL del tag "root".

Nota

Es pot definir una URL que retorni la vista del resultat en format PDF.

Exemple:

```
<vlh:action url="categories.do?">  
  <vlh:addParam name="reqCode" value="searchExportPDF" temp="false"/>  
  <bean:message key="valueListInfo.toPDF"/>  
</vlh:action>
```

<vlh:addParam/>

Aquest tag afegeix paràmetres al tag pare (típicament "root" o "action").

Atribut	Descripció
name	Nom del paràmetre a afegir a l'URL.
property (opcional)	Valor dinàmic obtingut d'una propietat del registre.
value (optional)	Valor estàtic pel paràmetre.



temp (true/false)	Afegeix el prefix "ACTION_TEMP_PARAM" al nom del paràmetre.
-------------------	---

2.2. Instal·lació i Configuració

2.2.1. Instal·lació

La instal·lació del servei requereix de la utilització de la llibreria 'openFrame-services-web-lists' i les dependències indicades a l'apartat 'Introducció-Versions i Dependències'.

2.2.2. Configuració

La configuració del servei implica:

- 1) Definir el servei i injectar-li les seves dependències
- 2) Enllaç d'una acció amb el seu "Value List"

Definició del servei i de les seves dependències

```
<bean  
id="valueListActionHelper"  
...>
```

Fitxer de configuració: *openFrame-services-web-lists.xml*

Ubicació proposada: *<PROJECT_ROOT>/src/main/resources/spring*

La definició del servei requereix configurar un bean "pare" amb un identificador i els següents atributs:

Atributs:

Atribut	Requerit	Descripció
class	Sí	Implementació concreta del servei a utilitzar Opcions: <ul style="list-style-type: none">• net.opentrends.openframe.services.web.lists.impl.SimpleValueListActionHelper.

També es necessari configurar les següents propietats:

Propietat	Requerit	Descripció
valueListHelper	Sí	Classe d'ajuda per a les búsques
maxExportRows	No	Número màxim de files a exportar
logService	Sí	Referència al servei de traces

Exemple:

```
...
<bean name="valueListActionHelper"
class="net.opentrends.openframe.services.web.lists.impl.SimpleValueListActionHelpe
r">
    <property name="valueListHelper">
        <ref bean="valueListHelper" />
    </property>
    <property name="maxExportRows" value="2"></property>
    <property name="logService" ref="loggingService"/>
</bean>
...
```

Fitxer de configuració: openFrame-services-web-lists.xml

```
<bean id="valueListHelper"
...>
```

Ubicació proposada: <PROJECT_ROOT>/src/main/resources/spring

La definició del servei requereix configurar un bean “simplificador” per a executar les búsquedes:

Atributs:

Atribut	Requerit	Descripció
class	Sí	Classe d'ajuda a utilitzar Opcions: <ul style="list-style-type: none"> net.mlw.vlh.web.mvc.ValueListHandlerHelper

També es necessari configurar les següents propietats:

Propietat	Requerit	Descripció
valueListHandler	Sí	Delegat de les búsquedes

Exemple:

```
...
<bean id="valueListHelper"
class="net.mlw.vlh.web.mvc.ValueListHandlerHelper">
```



```
<property name="valueListHandler">
    <ref bean="valueListHandler" />
</property>
</bean>
...
```

```
<bean id="valueListHandler"
    ...>
```

Fitxer de configuració: openFrame-services-web-lists.xml

Ubicació proposada: <PROJECT_ROOT>/src/main/resources/spring

La definició del servei requereix configurar un bean “delegat” per a localitzar les búsquedes:

Atributs:

Atribut	Requerit	Descripció
class	Sí	Classe delegada a utilitzar Opcions: <ul style="list-style-type: none">net.mlw.vlh.DefaultValueListHandlerImpl

També es necessari configurar les següents propietats:

Propietat	Requerit	Descripció
config.adapters	Sí	Mapa que conté els identificadors de cada una de les búsquedes a utilitzar i els beans que es cridaran per a fer les búsquedes.

Exemple:

```
...
<bean name="valueListHandler"
    class="net.mlw.vlh.DefaultValueListHandlerImpl">
    <property name="config.adapters">
        <map>
            <entry key="productList">
                <bean parent="baseHibernateAdapter">
                    <property name="hql">
                        FROM
                        net.opentrends.openframe.samples.jpstore.model.Product
                        AS vo WHERE l=1
                        /~name: AND vo.name LIKE {name} ~/
                        /~category: AND vo.category.id LIKE {category} ~/
                        /~sortColumn: ORDER BY vo.[sortColumn] [sortDirection]~/
                    </value>
                </bean>
            </entry>
        </map>
    </property>
</bean>
```




```
        <property name="removeEmptyStrings" value="true"/>
    </bean>
    ...
```

```
<bean id="vlConfig"
    ...>
```

Fitxer de configuració: openFrame-services-web-lists.xml

Ubicació proposada: <PROJECT_ROOT>/src/main/resources/spring

La definició del servei requereix configurar un bean “configurador” per a la presentació de resultats:

Atributs:

Atribut	Requerit	Descripció
class	Sí	Classe configuradora a utilitzar Opcions: <ul style="list-style-type: none">net.mlw.vlh.web.ValueListConfigBean

També es necessari configurar les següents propietats:

Propietat	Requerit	Descripció
messageSource	Si	Referència al servei multi idioma
displayProviders	Si	Proveïdors dels diferents tipus de llistats. Opcions: <ul style="list-style-type: none">net.opentrends.openframe.services.web.vlh.tag.support.PdfDisplayProvidernet.opentrends.openframe.services.web.vlh.tag.support.ExcelDisplayProvider

Exemple:

```
...
<bean name="vlConfig" class="net.mlw.vlh.web.ValueListConfigBean"
    singleton="false">
    <property name="messageSource" ref="messageSource" />
    <property name="displayProviders">
        <map>
            <entry key="ExportPDF">
                <bean
class="net.opentrends.openframe.services.web.vlh.tag.support.PdfDisplayProvider"
singleton="false">
```




```
ref="loggingService"/>
ref="i18nService"/>

<value>jsp.categories.categoryList.actions</value>
<value>jsp.products.productList.actions</value>
</list>
</property>
</bean>
</entry>
</map>
</property>
</bean>
...
```

Eliminar columnas d'un llistat en exportar

Eliminar columnas por título de columna:

Dins del fitxer openFrame-services-web-lists.xml hi trobem el bean de configuració del valuelist ([ValueListConfigBean](#)). Aquest bean conté els displayProviders ExportPdf i ExportExcel. Aquests beans tenen una propietat que es diu skipColumns. Aquí hauré d'introduir el titleKey de la columna que volem ignorar a l'exportar (tant a pdf com a excel):

```
<entry key="ExportPDF">
  <bean
class="net.opentrends.openframe.services.web.vlh.tag.support.PdfDisplayProvider"
singleton="false">
    <property name="logService" ref="loggingService"/>
    <property name="i18nService" ref="i18nService"/>
    <property name="skipColumns">
      <list>
        <value>jsp.categories.categoryList.actions</value>
        <value>jsp.products.productList.actions</value>
      </list>
    </property>
  </bean>
</entry>
<entry key="ExportExcel">
  <bean
class="net.opentrends.openframe.services.web.vlh.tag.support.ExcelDisplayProvider"
singleton="false">
    <property name="logService" ref="loggingService"/>
    <property name="i18nService" ref="i18nService"/>
    <property name="skipColumns">
      <list>
        <value>jsp.categories.categoryList.actions</value>
        <value>jsp.products.productList.actions</value>
      </list>
    </property>
  </bean>
</entry>
```



Eliminar columnes per index de columna:

Per eliminar columnes pel seu index, haurem de configurar-ho en la Action que fa l'export:

```
public ActionForward searchExportPDF(ActionMapping mapping, ActionForm form,
    javax.servlet.http.HttpServletRequest request,
    javax.servlet.http.HttpServletResponse response) throws Exception {

    List skipColumnNumbers = new ArrayList();
    skipColumnNumbers.add("1");
    skipColumnNumbers.add("3");

    request.setAttribute(PdfDisplayProvider.SKIP_COLUMN_NUMBERS, skipColumnNumbers);

    request.setAttribute("displayProvider", "ExportPDF");

    return this.search(mapping, form, request, response);
}

public ActionForward searchExportExcel(ActionMapping mapping, ActionForm form,
    javax.servlet.http.HttpServletRequest request,
    javax.servlet.http.HttpServletResponse response) throws Exception {

    List skipColumnNumbers = new ArrayList();
    skipColumnNumbers.add("1");
    skipColumnNumbers.add("3");

    request.setAttribute(ExcelDisplayProvider.SKIP_COLUMN_NUMBERS, skipColumnNumbers);

    request.setAttribute("displayProvider", "ExportExcel");

    return this.search(mapping, form, request, response);
}
```

En l'exemple anterior, a l'exportar tant a pdf com a excel, estariem ignorant les columnes primera i tercera.

Enllaç d'una acció amb el seu “Value List”

```
<property name="valueListActionHelper">
    <bean parent="valueListActionHelper">
        ...>
    </bean>
</property>
```

Fitxer de configuració: action-servlet-XXX.xml

Ubicació proposada: <PROJECT_ROOT>/src/main/resources/spring



Cada acció que implementi una búsqueda necessita la definició d'un bean que hereti del bean "pare" del servei i que tingui els següents atributs:

Atributs:

Atribut	Requerit	Descripció
listName	Sí	Nom de la consulta que es llençarà
tableId	Sí	Identificador únic de la taula que es consulta, per a guardar en sessió els filtres de les búsquedes

Exemple:

```
...  
<property name="valueListActionHelper">  
  <bean parent="valueListActionHelper">  
    <property name="listName">  
      <value>categoriesList</value>  
    </property>  
    <property name="tableId" value="CATEGORY"/>  
  </bean>  
</property>  
...
```

2.3. Utilització del servei

Per a poder utilitzar el servei, s'han de seguir els següents passos:

1. Implementar la interfície 'UsesValueList':

```
public class CategoryAction extends DispatchActionSupport implements  
    UsesValueList {  
    ...  
}
```

2. Definir un atribut d'instància que implementi 'ValueListActionHelper'

```
private ValueListActionHelper valueListActionHelper;
```

3. Definir els 'setters' i 'getters' corresponents



```
public ValueListActionHelper getValueListActionHelper() {  
    return valueListActionHelper;  
}  
  
public void setValueListActionHelper(ValueListActionHelper  
valueListActionHelper) {  
    this.valueListActionHelper = valueListActionHelper;  
}
```

4. Implementar els mètodes de búsqueda corresponents

```
public ActionForward search(ActionMapping mapping,  
    ActionForm form,  
    javax.servlet.http.HttpServletRequest request,  
    javax.servlet.http.HttpServletResponse response)  
throws Exception {  
    valueListActionHelper.search(mapping, form, request, response);  
    return mapping.findForward("list");  
}
```

5. Crear els JSP's amb els tags necessaris per a presentar els resultats

2.4. Eines de suport

2.5. Integració amb altres serveis

2.5.1. Servei multi idioma

El servei de llistats s'integra amb el servei de multi idioma a partir de la propietat "messageSource" del bean "configurador" (típicament "vlConfig"). És necessari per això definir la configuració que s'utilitzarà (atribut "configName") en cadascun dels JSP de presentació de resultats

2.5.2. Servei de persistència

El servei de llistats s'integra amb el servei de persistència a partir de la propietat "sessionFactory" del bean "delegat" (típicament "baseHibernateAdapter") d'executar les queries.

2.6. Preguntes Freqüents

Per què el servei de llistats no em mostra els títols de les columnes?

Per a mostrar els "titleKey" corresponents en el servei de llistats és necessari configurar al bean "configurador" (típicament vlConfig) la referència al servei multi idioma (propietat "messageSource") i afegir als JSP el atribut "configName" al tag <vlh:root/>



3. Exemples

Com exemple d'utilització del servei de llistats s'inclou el resultat d'un llistat i el codi font de la vista i la seva *Action*.

```
public class CategoryAction extends DispatchActionSupport implements
    UsesValueList {
    private ValueListActionHelper valueListActionHelper;

    ...

    public ValueListActionHelper getValueListActionHelper() {
        return valueListActionHelper;
    }

    public void setValueListActionHelper(ValueListActionHelper
valueListActionHelper) {
        this.valueListActionHelper = valueListActionHelper;
    }
    ...
}
```

Codi 1 Exemple d'utilització del servei associat a una 'Action'

```
<% @ include file="layouts/fwkJtagLibs.jsp"%>

<fwk:html>

<!-- Search criteria -->
<table cellpadding="0" cellspacing="0" border="0">
  <tr>
    <td width="100%" align="center">
      <fwk:form action="/categories.do" reqCode="search" style="height: 50px; width:100%;" >
        <fwk:text property="id" key="category.id" mode="E,E,E" size="4" maxLength="4" styleClass="LABEL" />
        <fwk:text property="descn" key="category.descn" mode="E,E,E" size="40" maxLength="40" styleClass="LABEL" />
      </fwk:form>
      <layout:image styleClass="submit(reqCode:'search',message:'submit.button.search')" src="images/ast.gif"/>
    </td>
  </tr>
</table>

<!-- Result list -->
<logic:present name="list" scope="request">
  <vlh:root value="list" url="categories.do?" includeParameters="reqCode|id|descn" configName="vlConfig">
    <table width="450">
      <!-- pagination section -->
      <tr>
        <td align="left" nowrap="true">
          <c:out value="{list.valueListInfo.totalNumberOfEntries}" />
          <bean:message key="valueListInfo.totalNumberOfEntries" />
          (<c:out value="{list.valueListInfo.pagingPage}" />
          <bean:message key="valueListInfo.totalNumberOfEntries.of" />
          <c:out value="{list.valueListInfo.totalNumberOfPages}" />)
        </td>
        <td align="right">
```



```
<vlh:paging/>
</td>
</tr>
<!-- end of Pagination -->

<tr>
<td colspan="2">
<table width="450" class="classicLook" cellspacing="0"
  cellpadding="0">
<vlh:row bean="job"
  display="<%= (String)request.getAttribute("subReqCode")%>">
<vlh:attribute name="onmouseover">
  this.className='selected';

</vlh:attribute>
<vlh:attribute name="onmouseout">
  this.className='zebra0';

</vlh:attribute>
<vlh:column titleKey="category.id" property="id"
  sortable="desc" />
<vlh:column titleKey="category.name" property="name"
  sortable="desc" />
<vlh:controls titleKey="category.control">
<vlh:action url="products.do?">
  <vlh:addParam name="reqCode" value="search"
    temp="false" />
    <bean:message key="products.edit"/>
    <vlh:addParam name="category" property="id"
      temp="false" />

</vlh:action>
<vlh:action url="categories.do?">
  <vlh:addParam name="reqCode" value="edit"
    temp="false"/>
    <bean:message key="category.edit"/>
    <vlh:addParam name="id" property="id" temp="false" />

</vlh:action>
</vlh:controls>
</vlh:row>
</table>
</td>
</tr>
</table>

<vlh:action url="categories.do?">
<vlh:addParam name="reqCode" value="searchExportPDF"
  temp="false"/>
  <bean:message key="valueListInfo.toPDF"/>
</vlh:action>
</vlh:root>
</logic:present>

</fwk:html>
```

Codi 2 Exemple d'utilització del servei des d'una vista JSP

Nota

És important d'adonar-se'n com s'ha accedit a diferents propietats de la llistat (com el nombre de files, etc) a la "Pagination section".

Exemple: `<c:out value="${list.valueListInfo.totalNumberOfEntries}" />`



4. Annexos