



LIFERAY®

Italian Partner Ecosystem

Dynamic Data Module

Give a name to your information

Mariuzzo Mauro

Liferay Architect

LIFERAY BOOT-CAMP REloaded

Come sfruttare al meglio le API del CMS di Liferay Portal 6.2

MILANO

12 GIUGNO 2014



Agenda

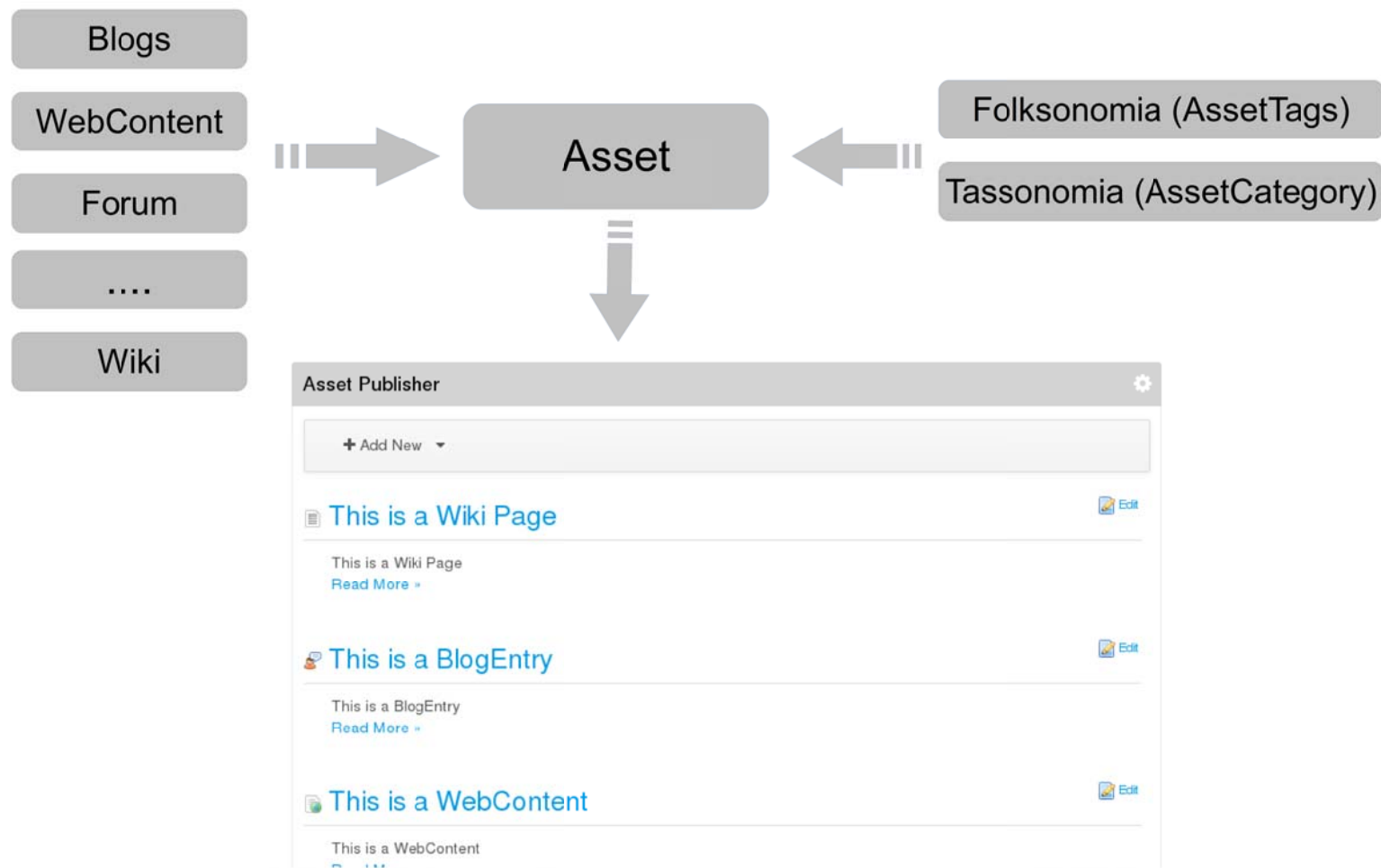
- Un po' di ripasso
- Dynamic Data Module – Strutture Dati
- Dynamic Data Module – Un po' di codice

Un po' di ripasso



Asset Framework

E' un meccanismo della componente CMS che consente la pubblicazione di informazioni eterogenee



Asset Framework

Permette di gestire nuove tipologie di informazioni realizzate come plugin (hot-deployable):

- Calendar
- KnowledgeBase
- ...



Web Content

Una tipologia predominante di informazione CMS è il WebContent

- Inizialmente chiamato JournalArticle (l'entità si chiama ancora così)
- Rappresenta un contenuto html

New Web Content Article

Structure: Default

Template: None

Default Language: English (United States)

Title (Required)

Content

Styles

Size

Content

[Abstract](#)

[Categorization](#)

[Schedule](#)

[Display Page](#)

[Related Assets](#)

[Permissions](#)

[Custom Fields](#)

Mischiare contenuto e stile non è mai una buona cosa. Per questo nascono

Structure and Template

Una Structure (o Struttura) descrive una informazione attraverso degli elementi base.

Ad esempio una “Scheda Quadro” è composta da:

- Artista → Testo
- Anno di Realizzazione → Data
- Luogo di Realizzazione → Testo
- Committente → Testo
-

Il Template (o Modello), tramite un mix di html e Velocity / FreeMarker, permette di descrivere come gli elementi della struttura si presenteranno all'utente.

Una Structure può avere più Template che la rappresentano

New Scheda Quadro

Structure: Scheda Quadro  (Use Default) Template: None

Default Language:  **English (United States)** [Change](#) 

Title (Required)

Autore

Realizzato il

06/07/2014

Realizzato a

Content

[Abstract](#)

[Categorization](#)

[Schedule](#)

[Display Page](#)

[Related Assets](#)

[Permissions](#)

Asset Publisher

Colazione sull'erba (Monet)

Quadro realizzato da Claude Monet su commissione di Sconosciuto.

Realizzato il	07 Jun 1866 - 00:00:00
Realizzato a	Parigi

Dynamic Data List e Dynamic Data Module

Liferay 6.1 introduce le Dynamic Data List: un componente che consente di descrivere una nuova entità, non direttamente riconducibile ad una tabella di database.

Semplificando:

- Tramite il Dynamic Data Module (DDM) descrivo un elemento logico composto da elementi semplici (Stringa, Data, Boolean, Documento nella DML, ...)
- Un elemento logico può estenderne un altro
- Una Dynamic Data List (DDL) dichiara una entità virtuale dove la struttura del record è
 - Uno o più elementi logici
 - Uno o più elementi semplici

Tutto questo significa:

- Un tool visuale per consentire all'utente di realizzare i DDM (tanto javascript e Ajax)
- Una serie di componenti per rappresentare i DDM (tanto Freemarker)
- Ovviamente i portlet DDM e DDL

Dynamic Data Module

Liferay 6.2 unifica i tool di gestione delle informazioni dinamiche / virtuali:

- Le componenti “Metadata Sets” e “Document Type” della Document and Media Library
- Le “Data Definitions” delle Dynamic Data List
- Le Structure del Journal

Tutte utilizzano il framework Dynamic Data Module

- Il suo tool di gestione visuale (basato su componenti AlloyUI)
- Il suo portlet
- La sua struttura dati

Dynamic Data Module

Struttura Dati



DDMStructure

- Contiene la descrizione dell'elemento logico, del Module, della Struttura (ex JournalStructure)
- La descrizione è un xml salvato all'interno del campo xsd
- Il campo ClassNameId identifica l'ambito di utilizzo (DML, DDL, WebContent, ...)

DDMTemplate

- Contiene i “Form Template” e “Display Template” delle “Data Definitions” (DML)
- Contiene gli ADT
- Contiene il Template del WebContent (ex JournalTemplate)

Dynamic Data Module e Dynamic Data List

DDMStructure

Id=12255
classNameId=10098 (DDLRecordSet)
structureKey=12254
xsd=<la struttura>

DDLRecord

Id=12259
ddmStorageId=12260
RecordSetId=12257
version=x.y

DDLRecordSet

Id=12257
ddmStructureId=12255
recordSetKey=12256

DDLRecordVersion

Id=12262
ddmStorageId=12260
RecordSetId=12257
RecordId=12259
version=x.y

DDMContent

Id=12260
xml=<i valori>

Dynamic Data Module e Document Type (DML)

DDMStructure

Id=12265
classNameId=10091 (DLFileEntryMetadata)
structureKey=12264
xsd=<la struttura>

DLFileEntryType

Id=12267
fileEntryTypeKey=12266

DLFileentryTypes_DDMStructures

structureId=12265
fileEntryTypeId=12267

DDMStorageLink

Id=12273
classNameId=10100 (DDMContent)
classPK=12272
structureId=12265

DDMContent

Id=12272
xml=<i valori>

Dynamic Data Module e Structure & Template

DDMStructure

Id=12288

classNameId=10109 (JournalArticle)

structureKey=12287

xsd=<la struttura>

DDMTemplate

Id=12290

classNameId=10102 (DDMStructure)

classPK=12288

tTemplateKey=12289

JournalArticle

Id=12293

StructureId=12287

templateId=12289

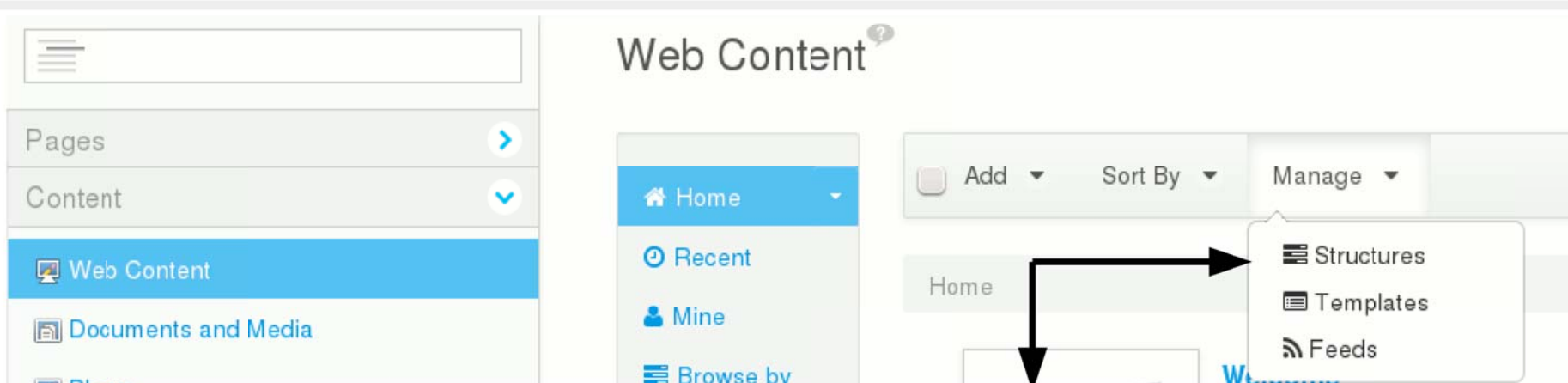
content=<i valori>

Dynamic Data Module

Un po' di codice



Manage Structures



The screenshot shows the Liferay Web Content Manager interface. On the left is a sidebar with navigation options: Pages, Content, Web Content (selected), Documents and Media, and Blogs. The main area is titled 'Web Content' and contains a 'Home' button, a 'Recent' list, a 'Mine' list, and a 'Browse by' section. A 'Manage' dropdown menu is open, showing options: Structures (selected), Templates, and Feeds. An arrow points from the 'Home' button to the 'Structures' option in the dropdown menu.

```
function <portlet:namespace />openStructuresView() {
  Liferay.Util.openDDMPortlet(
  {
    basePortletURL: '<%= PortletURLFactoryUtil.create(request, PortletKeys.DYNAMIC_DATA_MAPPING, theme...
    dialog: {
      destroyOnHide: true,
      on: {
        visibleChange: function(event) {
          if (!event.newVal) {
            Liferay.Portlet.refresh('#p_p_id_' + <%= portletDisplay.getId() %> + '_' );
          }
        }
      }
    },
    refererPortletName: '<%= PortletKeys.JOURNAL %>',
    refererWebDAVToken: '<%= portlet.getWebDAVStorageToken() %>',
    showGlobalScope: false,
    showManageTemplates: true,
    title: '<%= UnicodeLanguageUtil.get(pageContext, "structures") %>'
  }
  );
}
```

Manage Structures

Dal portlet Journal attivo il portlet Dynamic Data Mapping

Structures

+ Add

Keywords

Search

Delete

	ID	Name	Description	Modified Date	
	12240	Scheda Quadro		16 Hours Ago	▼ Actions
	12288	Test Mauro		15 Hours Ago	▼ Actions

Manage Structures

Add oppure Edit mi portano in edit_structure.jsp

← New Structure

There are content references to this structure. You may lose data if a field name is renamed or removed.

Name (Required)



Details

Description



Parent Structure

View

Source

Default Language:



English (United States)

Change



Add Translation

Fields

Settings



Boolean



Date



Decimal



Document...



HTML



Image



Integer



Link to Page

Manage Structures

Quello che ci interessa oggi è questa parte

Fields

Settings



Boolean



Date



Decimal



Document...



HTML



Image



Integer



Link to Page



Number



Radio



Select



Text



Text Box

Fields

Settings

Property Name	Value
Type	ddm-number
Field Label	Number
Show Label	Yes
Required	No
Name	Number1663
Predefined Value	
Tip	
Indexable	Yes
Repeatable	No
Width	Small

< >

Close

Number

Dynamic Data Module – edit_structure.jsp

edit_structure.jsp -> form_builder.jspf -> custom_fields.jspf

```
<au:script>
  AUI.add(
    'liferay-portlet-dynamic-data-mapping-custom-fields',
    function(A) {
      var FormBuilderTextField = A.FormBuilderTextField;
      var FormBuilderTypes = A.FormBuilder.types;
      ....
      var DDMNumberField = A.Component.create(
        {
          ATTRS: {
            dataType: {
              value: 'number'
            },

            fieldNamespace: {
              value: 'ddm'
            }
          },

          EXTENDS: A.FormBuilderTextField,

          NAME: 'ddm-number'
        }
      );
      ....
    }
  );
  ....
}
```

Dynamic Data Module – edit_structure.jsp

edit_structure.jsp -> form_builder.jspf -> custom_fields.jspf

```
....
    FormBuilderTypes['ddm-date'] = DDMDateField;
    FormBuilderTypes['ddm-decimal'] = DDMDecimalField;
    FormBuilderTypes['ddm-documentlibrary'] = DDMDocumentLibraryField;
    FormBuilderTypes['ddm-integer'] = DDMIntegerField;
    FormBuilderTypes['ddm-link-to-page'] = DDMLinkToPageField;
    FormBuilderTypes['ddm-number'] = DDMNumberField;
    FormBuilderTypes['ddm-paragraph'] = DDMParagraphField;
    FormBuilderTypes['ddm-separator'] = DDMSeparatorField;
    FormBuilderTypes['ddm-text-html'] = DDMHTMLTextField;
    FormBuilderTypes['wcm-image'] = WCMImageField;
    },
    {
    {
        requires: ['liferay-portlet-dynamic-data-mapping']
    }
    );
</aui:script>
```

I FieldType standard sono definiti in AlloyUI

- alui-form-builder-field-base
- alui-form-builder-field-button
- alui-form-builder-field-checkbox
- alui-form-builder-field-fieldset
- alui-form-builder-field-file-upload
- alui-form-builder-field-multiple-choice
- alui-form-builder-field-radio
- alui-form-builder-field-select
- alui-form-builder-field-text
- alui-form-builder-field-textarea

Dynamic Data Module – edit_structure.jsp

Ognuno di loro

- Estende il tipo base (A.FormBuilderField)
- dichiara le proprietà base (Label, Name, required, etc.) nel metodo getPropertyModel

```
getPropertyModel: function() {  
    var instance = this, strings = instance.getStrings();  
  
    var model = A.FormBuilderTextField.superclass.getPropertyModel.apply(instance, arguments);  
  
    model.push({  
        attributeName: WIDTH,  
        editor: new A.RadioCellEditor({  
            options: {  
                small: strings[SMALL],  
                medium: strings[MEDIUM],  
                large: strings[LARGE]  
            }  
        }  
    }),  
    formatter: function(o) {  
        return strings[o.data.value];  
    },  
    name: strings[WIDTH]  
});  
  
    return model;  
},
```

Dynamic Data Module – Extend DDMIntegerField

Supponiamo di voler estendere l'oggetto DDMIntegerField.

Supponiamo di voler avere un nuovo attributo testuale chiamato “bootcamp”

Poiché il FormBuilder è costruito nel file “custom_fields.jspf” procediamo con un semplice hook

Dynamic Data Module – Extend DDMIntegerField

```
var DDMIntegerField = A.Component.create(  
  {  
    ATTRS: {  
      dataType: {  
        value: 'integer'  
      },  
  
      fieldNamespace: {  
        value: 'ddm'  
      },  
  
      bootcamp: {  
        value: ''  
      },  
    },  
  
    EXTENDS: A.FormBuilderTextField,  
  
    NAME: 'ddm-integer',
```

Valore di default del nuovo attributo

Dynamic Data Module – Extend DDMIntegerField

```
prototype: {  
  getPropertyModel: function() {  
    var instance = this;  
  
    var model = DDMIntegerField.superclass.getPropertyModel.apply(instance, arguments);  
  
    model.push({  
      attributeName: 'bootcamp',  
      editor: new A.TextCellEditor(),  
      name: '<%= UnicodeLanguageUtil.get(pageContext, "bootcamp") %>'  
    });  
  
    return model;  
  }  
}  
};
```

Informo Engine che il nuovo
attributo è una semplice
stringa da gestire con una
casella di testo

Dynamic Data Module – Extend DDMIntegerField

Compilo ed installo hook. Creo una nuova struttura

The screenshot shows the 'Fields' tab of a configuration window. On the left is a table with field properties. On the right is a preview of the field's appearance, showing the label 'Integer' and an empty input box. Below the table is a 'Close' button.

Property Name	Value
Type	ddm-integer
Field Label	Integer
Show Label	Yes
Required	No
Name	Integer1905
Predefined Value	
Tip	
Indexable	Yes
Repeatable	No
Width	Small
bootcamp	

Close

Integer

Dynamic Data Module – Extend DDMIntegerField

Compilo gli altri campi. Faccio Save...

Structures

Your request failed to complete.

Please enter a valid XML schema definition.

[←](#) New Structure

There are content references to this structure. You may lose data if a field name is changed.



Dynamic Data Module – Extend DDMIntegerField

Questo perché la struttura è un xml che viene validato sfruttando "liferay-ddm-structure_6_2_0.xsd" ed il campo "bootcamp" non è censito.

Dove è indicato il file xsd da usare? Come faccio a sostituirlo?

Lo schema è dichiarato

- nel file portal-impl/src/META-INF/util-spring.xml

```
<bean id="com.liferay.portlet.dynamicdatamapping.util.DDMXML"
      class="com.liferay.portlet.dynamicdatamapping.util.DDMXMLImpl">
  <property name="XMLSchema">
    <bean class="com.liferay.portal.xml.XMLSchemaImpl">
      <property name="schemaLanguage" value="http://www.w3.org/2001/XMLSchema" />
      <property name="systemId"
        value="http://www.liferay.com/dtd/liferay-ddm-structure_6_2_0.xsd" />
    </bean>
  </property>
</bean>
```

- nel file portal-impl/src/com/liferay/portal/util/EntityResolver.java, dove l'omonima classe è usata da
 - com/liferay/portal/xml/XMLSchemaImpl.java, con un new EntityResolver()
 - com/liferay/portal/xml/SAXReaderImpl.java, con un new EntityResolver()

Dynamic Data Module – edit_structure.jsp

Devo creare uno schema che estenda “liferay-ddm-structure_6_2_0.xsd”.

Devo modificare la creazione del bean

“com.liferay.portlet.dynamicdatamapping.util.DDMXML”

Dynamic Data Module – edit_structure.jsp

Devo creare uno schema che estenda “liferay-ddm-structure_6_2_0.xsd”.

Devo modificare la creazione del bean

“com.liferay.portlet.dynamicdatamapping.util.DDMXML”

E lo posso fare solo in

EXT

Devo creare uno schema che estenda “liferay-ddm-structure_6_2_0.xsd”.

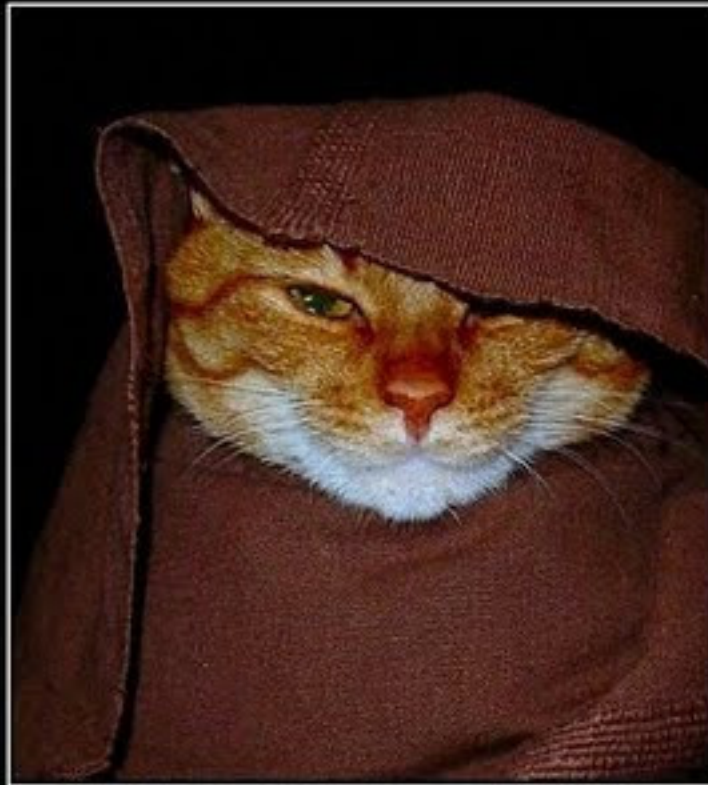
Devo modificare la creazione del bean

“com.liferay.portlet.dynamicdatamapping.util.DDMXML”

E lo posso fare solo in

EXT





USE THE FORCE, MAURO!
USE EXTUPDATER....

Dynamic Data Module – edit_structure.jsp

Creo “liferay-ddm-structure_6_2_0-ext.xsd” come copia di “liferay-ddm-structure_6_2_0-ext.xsd” in:

- docroot/WEB-INF/definitions/liferay-ddm-structure_6_2_0-ext.xsd
- docroot/WEB-INF/ext-web/dtd/liferay-ddm-structure_6_2_0-ext.xsd
- docroot/WEB-INF/ext-impl/src/com/liferay/portal/definitions/liferay-ddm-structure_6_2_0-ext.xsd

```
<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xs:attributeGroup name="dynamic-element-attribute-group">
    <xs:attribute name="dataType" type="data-type" />
    <xs:attribute name="fieldNamespace" type="xs:string" />
    <xs:attribute name="indexType" type="index-type" />
    <xs:attribute name="multiple" type="xs:boolean" />
    <xs:attribute name="name" type="name" use="required" />
    <xs:attribute name="readOnly" type="xs:boolean" />
    <xs:attribute name="repeatable" type="xs:boolean" />
    <xs:attribute name="required" type="xs:boolean" />
    <xs:attribute name="showLabel" type="xs:boolean" />
    <xs:attribute name="type" type="type" use="required" />
    <xs:attribute name="value" type="xs:string" />
    <xs:attribute name="width" type="xs:string" />
    <xs:attribute name="bootcamp" type="xs:string" />
  </xs:attributeGroup>
  <xs:element name="dynamic-element">
```

Dynamic Data Module – edit_structure.jsp

Creo “./docroot/WEB-INF/ext-impl/src/META-INF/ext-spring.xml”

```
<?xml version="1.0"?>

<beans
  default-destroy-method="destroy"
  default-init-method="afterPropertiesSet"
  xmlns="http://www.springframework.org/schema/beans"
  xmlns:aop="http://www.springframework.org/schema/aop"
  xmlns:util="http://www.springframework.org/schema/util"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.springframework.org/schema/aop ....
>

<bean id="com.liferay.portlet.dynamicdatamapping.util.DDMXML"
  class="com.liferay.portlet.dynamicdatamapping.util.DDMXMLImpl">
  <property name="XMLSchema">
    <bean class="com.liferay.portal.xml.XMLSchemaImpl">
      <property name="schemaLanguage" value="http://www.w3.org/2001/XMLSchema" />
      <property name="systemId"
        value="http://www.liferay.com/dtd/liferay-ddm-structure_6_2_0-ext.xsd" />
    </bean>
  </property>
</bean>

</beans>
```

Dynamic Data Module – edit_structure.jsp

```
# ant clean war  
# <deploy>
```

Provo a salvare una nuova struttura

Dynamic Data Module – edit_structure.jsp

```
# ant clean war  
# <deploy>
```

Provo a salvare una nuova struttura

Structures

Your request completed successfully.

+ Add

Delete

☐	ID	Name	Description
☐	12240	Scheda Quadro	
☐	12288	Test Mauro	
☐	12502	Test Bootcamp	



Property Name	Value
Type	ddm-integer
Field Label	Integer
Show Label	Yes
Required	No
Name	Integer1643
Predefined Value	
Tip	
Indexable	Yes
Repeatable	No
Width	Small
bootcamp	Eureka

Dynamic Data Module – Usare nuovo attributo

Abbiamo una DDMIntegerField con un nuovo attributo.

E adesso? A cosa ci serve?

- 1) Per gestire un comportamento nella fase di inserimento del valore intero; ovvero nella maschera di inserimento di un WebContent
- 2) Per gestire un comportamento nella fase di presentazione del valore all'utente finale

Dynamic Data Module – Usare nuovo attributo

Nella fase di inserimento di un WebContent viene usata
“portal-web/docroot/html/portlet/journal/article/content.jsp”

```
<c:otherwise>

    <%
    Fields ddmFields = null;

    if ((article != null) && Validator.isNotNull(article.getStructureId()) && Validator.isNotNull(content)) {
        ddmFields = JournalConverterUtil.getDDMFields(ddmStructure, content);
    }

    String requestedLanguageId = defaultLanguageId;

    if (Validator.isNotNull(toLanguageId)) {
        requestedLanguageId = toLanguageId;
    }
    %>

    <liferay-ddm:html
        classNameId="<%= PortalUtil.getClassNameId(DDMStructure.class) %>"
        classPK="<%= ddmStructure.getStructureId() %>"
        fields="<%= ddmFields %>"
        repeatable="<%= Validator.isNull(toLanguageId) %>"
        requestedLocale="<%= LocaleUtil.fromLanguageId(requestedLanguageId) %>"
    />

</c:otherwise>
```

Dynamic Data Module – Usare nuovo attributo

La taglib “liferay-ddm:html” usa DDMXSDUtil

```
<%@ include file="/html/taglib/ddm/html/init.jsp" %>

<div class="lfr-ddm-container" id="<%= randomNamespace %>">
    <c:if test="<%= Validator.isNotNull(xsd) %>">
        <%= DDMXSDUtil.getHTML(pageContext, xsd, fields, portletResponse.getNamespace(),
                                fieldsNamespace, mode, readOnly, requestedLocale) %>
    </c:if>
</div>
```

DDMXSDImpl utilizza scripts FreeMarket per generare l'html

portal-impl > src > com > liferay > portlet > dynamicdatamapping > dependencies > ddm >

Nome	Dimensione	Data di modifica
date.ftl	1,2 kB	19/03/2014
decimal.ftl	417 byte	19/03/2014
documentlibrary.ftl	3,1 kB	19/03/2014
integer.ftl	417 byte	19/03/2014
link-to-page.ftl	2,4 kB	19/03/2014
number.ftl	417 byte	19/03/2014
paragraph.ftl	158 byte	19/03/2014
separator.ftl	160 byte	19/03/2014
text-html.ftl	979 byte	19/03/2014

Dynamic Data Module – Usare nuovo attributo

Questo è “integer.ftl”

```
<#include "../init.ftl">

<@ui["field-wrapper"] data=data helpMessage=escape(fieldStructure.tip)>
  <@ui.input cssClass=cssClass dir=requestedLanguageDir helpMessage=escape(fieldStructure.tip)
    label=escape(label) name=namespacedFieldName type="text" value=fieldValue>
    <@ui.validator name="digits" />

    <#if required>
      <@ui.validator name="required" />
    </#if>
  </@ui.input>

  ${fieldStructure.children}
</@>
```

Dynamic Data Module – Usare nuovo attributo

Il flusso di costruzione è:

- Entro con getHTML(...)
- Per ogni elemento chiamo getHTMLField(...)
- All'interno chiamo ProcessFTL(...) per caricare lo script FreeMarker

....

```
String type = element.attributeValue("type");

String templateName = StringUtil.replaceFirst(
    type, fieldNamespace.concat(StringPool.DASH), StringPool.BLANK);

StringBundler resourcePath = new StringBundler(5);

resourcePath.append(_TPL_PATH);
resourcePath.append(StringUtil.toLowerCase(fieldNamespace));
resourcePath.append(CharPool.SLASH);
resourcePath.append(templateName);
resourcePath.append(_TPL_EXT);

String resource = resourcePath.toString();

URL url = getResource(resource);
```

<com/liferay/portlet/dynamicdatamapping/dependencies/ddm/integer.ftl>

Dynamic Data Module – Usare nuovo attributo

Devo fare altri interventi nel mio EXT:

- Copio “init.ftl” e “ddm/integer.ftl” da
“com/liferay/portlet/dynamicdatamapping/dependencies” a
“com/**ext**/portlet/dynamicdatamapping/dependencies”
- Modifico integer.ftl

```
<#include "../init.ftl">

<@ui["field-wrapper"] data=data helpMessage=escape(fieldStructure.tip)>
  <div>Bootcamp = ${fieldStructure.bootcamp}</div>
  <@ui.input cssClass=cssClass dir=requestedLanguageDir
    helpMessage=escape(fieldStructure.tip) label=escape(label)
    name=namespacedFieldName type="text" value=fieldValue>
    <@ui.validator name="digits" />

    <#if required>
      <@ui.validator name="required" />
    </#if>
  </@ui.input>

  ${fieldStructure.children}
</@>
```

Dynamic Data Module – Usare nuovo attributo

Creo la classe CustomDDMXSDDImpl come estensione di DDMXSDDImpl per far usare il template in ext per il campo integer

```
public class CustomDDMXSDDImpl extends DDMXSDDImpl {

    @Override
    protected URL getResource(String name) {
        if (StringUtil.endsWith(name, "integer.ftl")) {
            URL url = super.getResource(_EXT_TPL_PATH + "integer.ftl");
            if (url != null) {
                return url;
            }
        }

        return super.getResource(name);
    }

    private static final String _EXT_TPL_PATH =
        "com/ext/portlet/dynamicdatamapping/dependencies/ddm/";
}
```

Dynamic Data Module – Usare nuovo attributo

Aggiungo in ext-spring.xml la direttiva per caricare la mia CustomDDMXSImpl

```
<?xml version="1.0"?>

<beans
    default-destroy-method="destroy" ....
>

    <bean id="com.liferay.portlet.dynamicdatamapping.util.DDMXML"
        class="com.liferay.portlet.dynamicdatamapping.util.DDMXMLImpl">
        <property name="XMLSchema">
            <bean class="com.liferay.portal.xml.XMLSchemaImpl">
                <property name="schemaLanguage" value="http://www.w3.org/2001/XMLSchema" />
                <property name="systemId"
                    value="http://www.liferay.com/dtd/liferay-ddm-structure_6_2_0-ext.xsd" />
            </bean>
        </property>
    </bean>

    <bean id="com.liferay.portlet.dynamicdatamapping.util.DDMXSUtil"
        class="com.liferay.portlet.dynamicdatamapping.util.DDMXSUtil">
        <property name="DDMXSD">
            <bean class="com.ext.portlet.dynamicdatamapping.util.CustomDDMXSImpl" />
        </property>
    </bean>

</beans>
```

Dynamic Data Module – edit_structure.jsp

```
# ant clean war  
# <deploy>
```

Provo a creare un nuovo WebContent con la struttura “Test Bootcamp”

← New Web Content

Structure:

Test Bootcamp  (Use Default)

Template: Test Bootcamp 

Default Language:  English (United States) [Change](#) 

Title (Required)

Bootcamp = Eureka
Integer





LIFERAY BOOT-CAMP REloaded

Come sfruttare al meglio le API del CMS di Liferay Portal 6.2

MILANO

12 GIUGNO 2014