

## This document explains How to create structured content in opencms.

In order to create your own structured content editor, the first step is to create an opencms module as a container for it.

As for now we can use default opencms module named  
“org.opencms.frontend.templateone.modules”

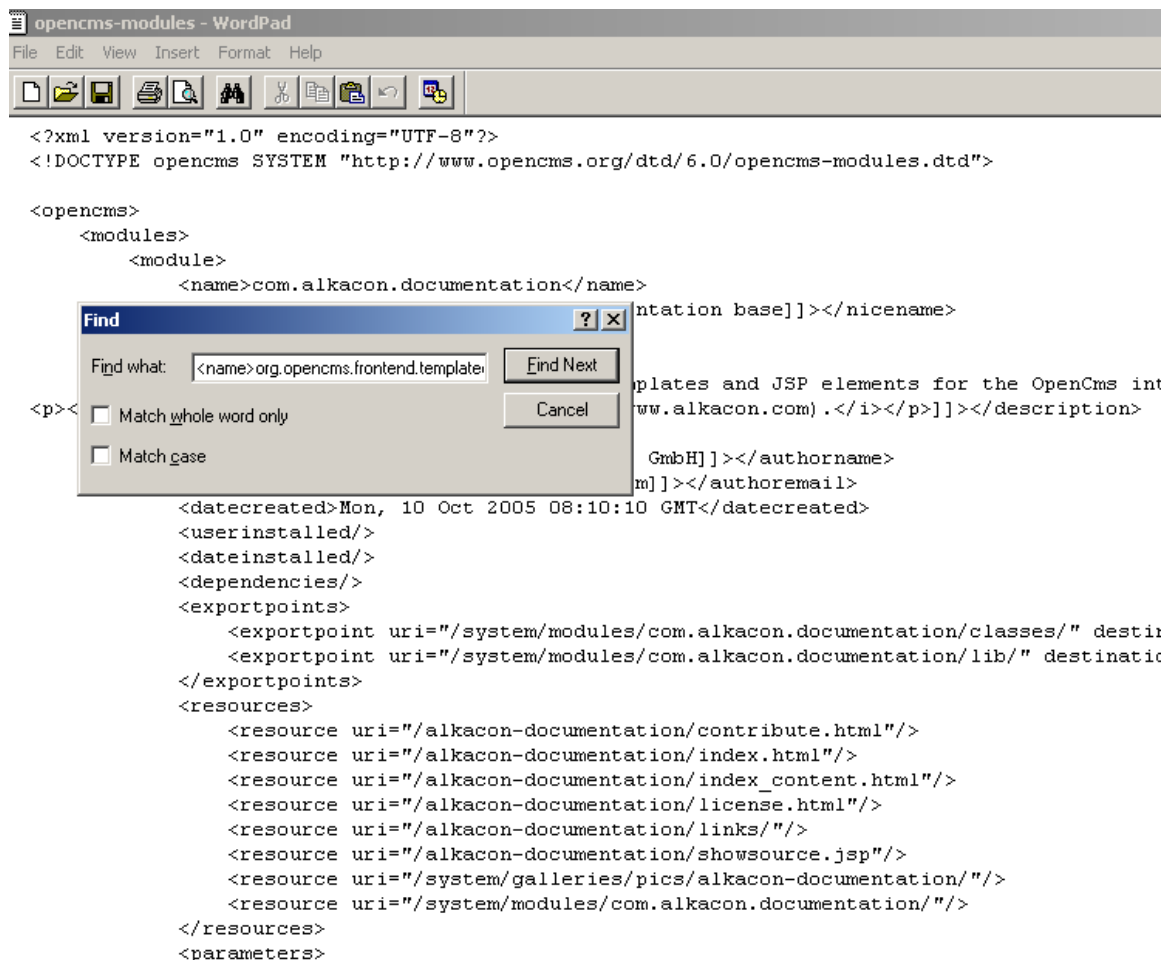
### Step 1:

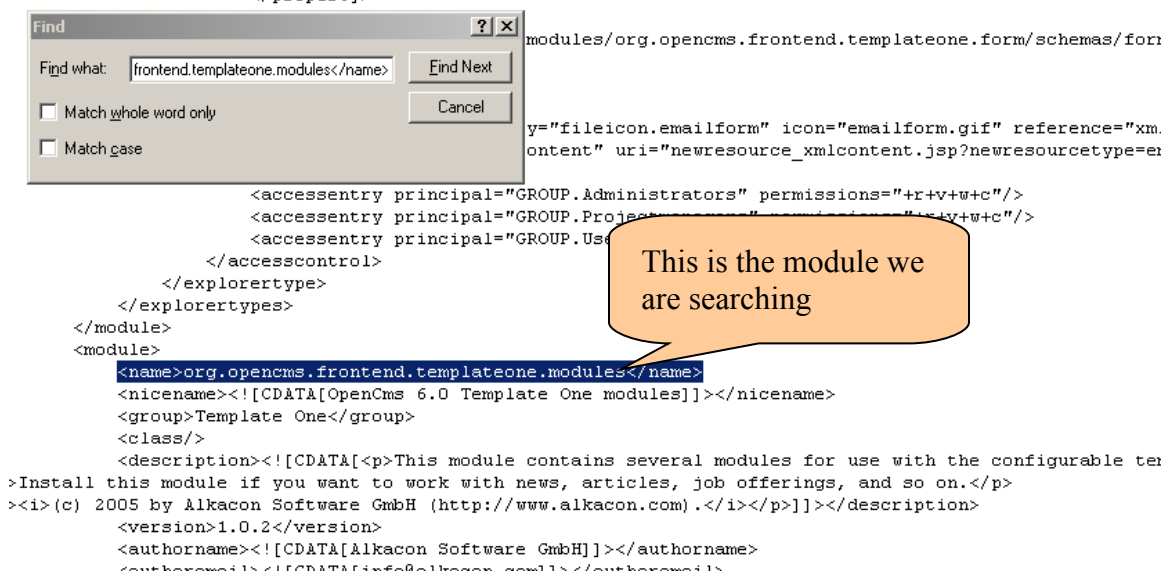
**Open the opencms-modules.xml file which is in the below path:**

\\ProgramFiles\\IBM\\WebSphere\\AppServer\\profiles\\default\\installedApps\\krishnanNode01Cell\\opencms\_war.ear\\opencms.war\\WEB-INF\\config\\

### Step 2:

**Go to the module “org.opencms.frontend.templateone.modules” by using the keyword “<name>org.opencms.frontend.templateone.modules</name>” to search.**





### Step 3:

Copy and paste the “type” which is shown in the below screen shot.

```
<type class="org.opencms.file.types.CmsResourceTypeXmlContent" name="news" id="30">
  <properties>
    <property>
      <name>content-conversion</name>
      <value type="shared"><![CDATA[cleanup;xhtml]]></value>
    </property>
    <property>
      <name>template-elements</name>
      <value type="shared"><![CDATA[/system/modules/org.opencms.frontend.templateone.modules
    </property>
  </properties>
  <param name="schema">/system/modules/org.opencms.frontend.templateone.modules/schemas/news.xsd
</type>
```

```
<type class="org.opencms.file.types.CmsResourceTypeXmlContent" name="news" id="30">
  <properties>
    <property>
      <name>content-conversion</name>
      <value type="shared"><![CDATA[cleanup;xhtml]]></value>
    </property>
    <property>
      <name>template-elements</name>
      <value type="shared"><![CDATA[/system/modules/org.opencms.frontend.templateone.modules
    </property>
  </properties>
  <param name="schema">/system/modules/org.opencms.frontend.templateone.modules/schemas/news.xsd
</type>
```

```
<type class="org.opencms.file.types.CmsResourceTypeXmlContent" name="faq" id="35">
  <properties>
```

This is the pasted one

#### Step 4:

Now we have to edit the pasted “type” as per our requirement.

```
<param name="schema">/system/modules/org.opencms.frontend.templateone.modules/schemas/news.xsd</param>
</type>

<type class="org.opencms.file.types.CmsResourceTypeXmlContent" name="test_article" id="33">
  <param name="schema">/system/modules/org.opencms.frontend.templateone.modules/schemas/sample.xsd</param>
</type>

<type class="org.opencms.file.types.CmsResourceTypeXmlContent" name="faq" id="35">
  <properties>
```

#### Step 5:

Now Copy and paste the “explorertype” as like in the screen shot given below

```
</resourcetypes>
<explorertypes>
  <explorertype name="news" key="fileicon.news" icon="article.gif" reference="xmlcontent">
    <newresource page="structurecontent" uri="newresource_xmlcontent.jsp?newresourcetype=news" o
    <accesscontrol>
      <accessentry principal="GROUP.Administrators" permissions="+r+v+w+c"/>
      <accessentry principal="GROUP.Projectmanagers" permissions="+r+v+w+c"/>
      <accessentry principal="GROUP.Users" permissions="+r+v+w+c"/>
    </accesscontrol>
  </explorertype>

  <explorertype name="news" key="fileicon.news" icon="article.gif" reference="
  <newresource page="structurecontent" uri="newresource_xmlcontent.jsp?new
  <accesscontrol>
    <accessentry principal="GROUP.Administrators" permissions="+r+v+w+c"/>
    <accessentry principal="GROUP.Projectmanagers" permissions="+r+v+w+c"/>
    <accessentry principal="GROUP.Users" permissions="+r+v+w+c"/>
  </accesscontrol>
</explorertype>
```

This is the pasted one

#### Step 6:

Now we have to edit the pasted “explorertype” as per our requirement.

```

        </accesscontrol>
    </explorertype>

    <explorertype name="test_article" key="fileicon.test_article" icon="article.gif" reference="xmlcontent">
        <newresource page="structurecontent" uri="newresource_xmlcontent.jsp?newresourcetype=test_article" order="99">
            <accesscontrol>
                <accessentry principal="GROUP.Administrators" permissions="+r+v+w+c"/>
                <accessentry principal="GROUP.Projectmanagers" permissions="+r+v+w+c"/>
                <accessentry principal="GROUP.Users" permissions="+r+v+w+c"/>
            </accesscontrol>
        </explorertype>

    <explorertype name="faq" key="fileicon.faq" icon="article.gif" reference="xmlcontent">
        <newresource page="structurecontent" uri="newresource_xmlcontent.jsp?newresourcetype=faq" order="51"/>
    </explorertype>

```

The name which is marked above should match the name that we had given in the step 4.

Now save the changes and close the file.

## Step 7:

**Go to the link given below in OpenCms Admin Interface**

</system/modules/org.opencms.frontend.templateone.modules/>

## Step 8:

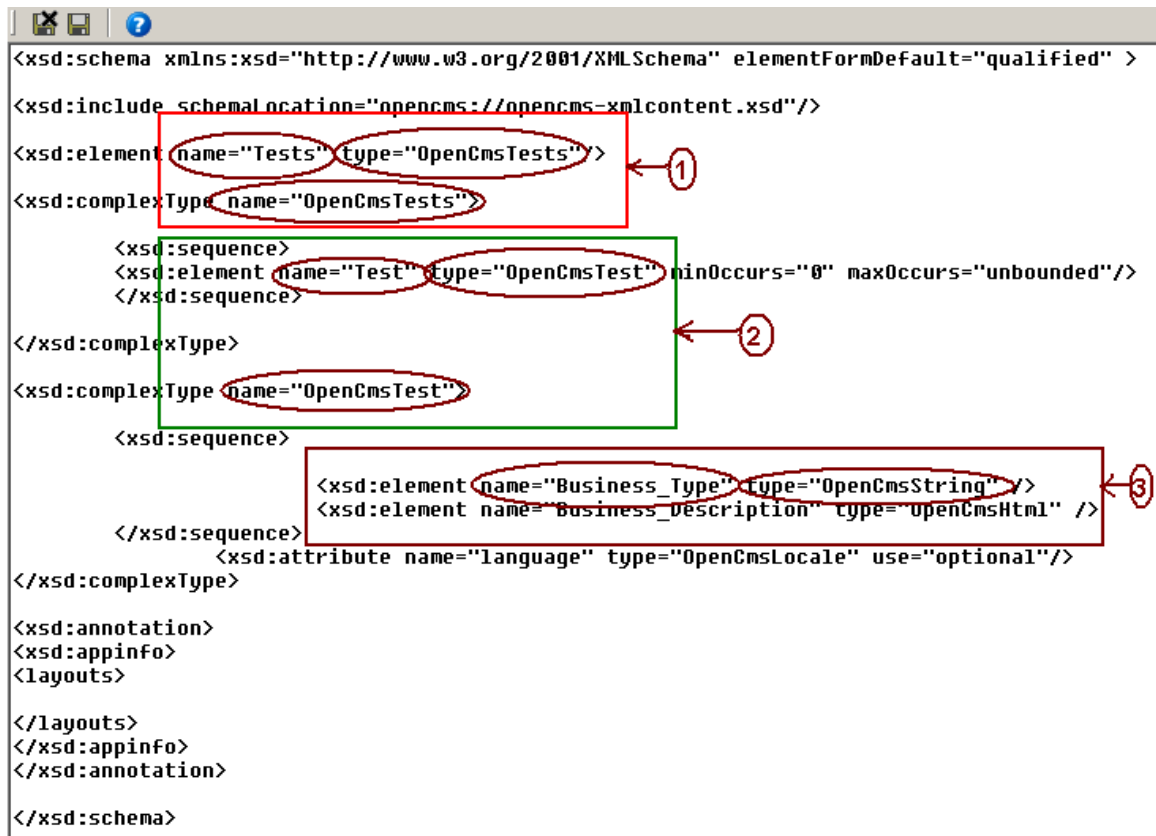
**Create your new “xsd” file inside the schemas folder.**

Level Up	Location	/system/modules/org.opencms.frontend.templateone.modules/schemas/				
		Name	Title	Type	Size	
		address.xsd	Nested XSD for contact address	plain	1587	1
		attachment.xsd	Nested XSD for attachments	plain	986	1
		event.xsd	XSD for events	plain	3274	1
		event_additionaldates.xsd	Nested XSD for additional event dates	plain	937	1
		event_address.xsd	Nested XSD for event address	plain	1524	1
		event_charges.xsd	Nested XSD for event charges	plain	1203	1
		event_dates.xsd	Nested XSD for event dates	plain	1394	1
		event_extended.xsd	Nested XSD for extended events	plain	1817	1
		event_organization.xsd	Nested XSD for event organization	plain	1665	1
		externallink.xsd	Nested XSD for external links	plain	1084	1
		faq.xsd	XSD for FAQ entries	plain	2537	1
		faqparagraph.xsd	Nested XSD for FAQ	plain	1429	1
		image.xsd	Nested XSD for images	plain	955	1
		internallink.xsd	Nested XSD for internal links	plain	995	1
		job.xsd	XSD for job entries	plain	3081	1
		job_extended.xsd	Nested XSD for extended jobs	plain	1446	1
		news.xsd	XSD for news articles	plain	3427	1
		sample.xsd		plain	0	7

**New xsd**

## Step 9:

Copy and paste any one of the sample xsd code and edit our changes.



In the Box 1 (red) you would edit the element name and type and also the complexType name.

Note:

The element name must be ended with the letter “s”.

In the element type you should add the text OpenCms in front of the element name as shown in the above Box 1.

For the ComplexType name also you should add the text OpenCms in front of the element name as shown in the above Box 1.

In the Box 2 (green) would edit the element name and type and also the complexType name.

Note:

Here the element name must not be ended with the letter “s”.

In the element type you should add the text OpenCms in front of the element name as shown in the above Box 2.

For the ComplexType name also you should add the text OpenCms in front of the element name as shown in the above Box 2.

In the Box 3 (brown) the element name “Business\_Type” which is marked in the above fig would display in the structured content file. You can change the label of the element in the workplace.properties file.

The type “OpenCmsString” is used to create the Text Field. Refer OpenCms Documentation for the various fields.

Note:

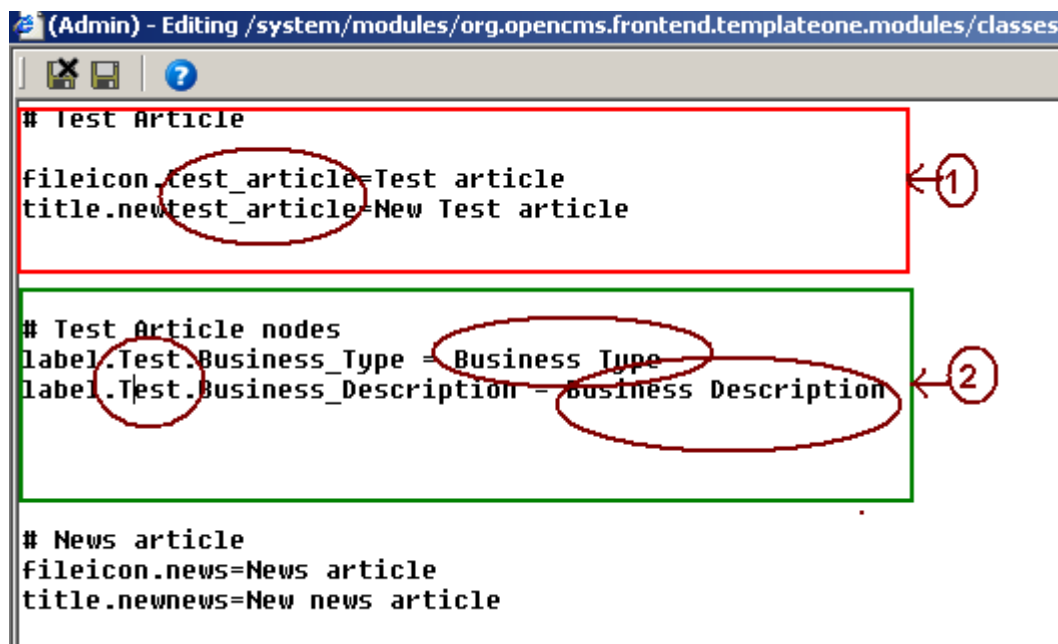
If you are not give any label for the element the default element name in the xsd file will display in the Structured Content file.

Finally save the changes and close the xsd file.

#### Step 10:

**Open the workplace.properties file the below path**

/system/modules/org.opencms.frontend.templateone.modules/classes/org/opencms/frontend/templateone/modules/



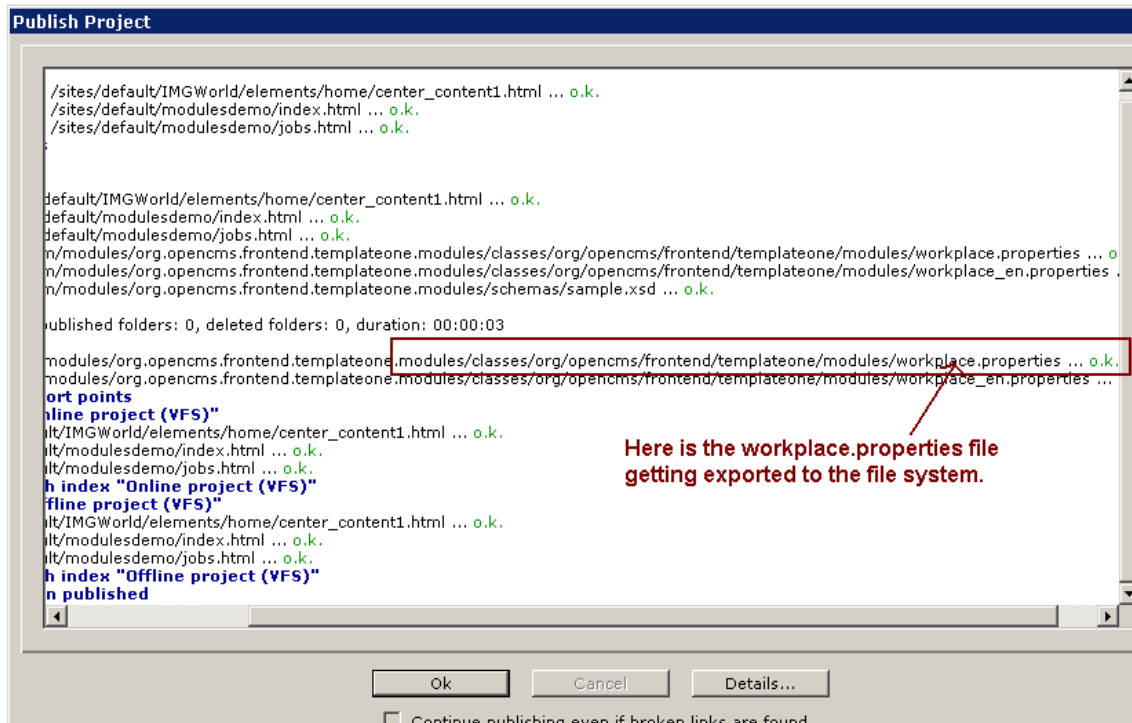
The “test\_article” which is marked in the box 1 (red) is the type name given in the **opencms-modules.xml** file.

The “Test” which is marked in the box 2(green) is the element name given in the **samples.xsd** file.

Now save the changes and close the file.

## Step 11.

### Publish the Project

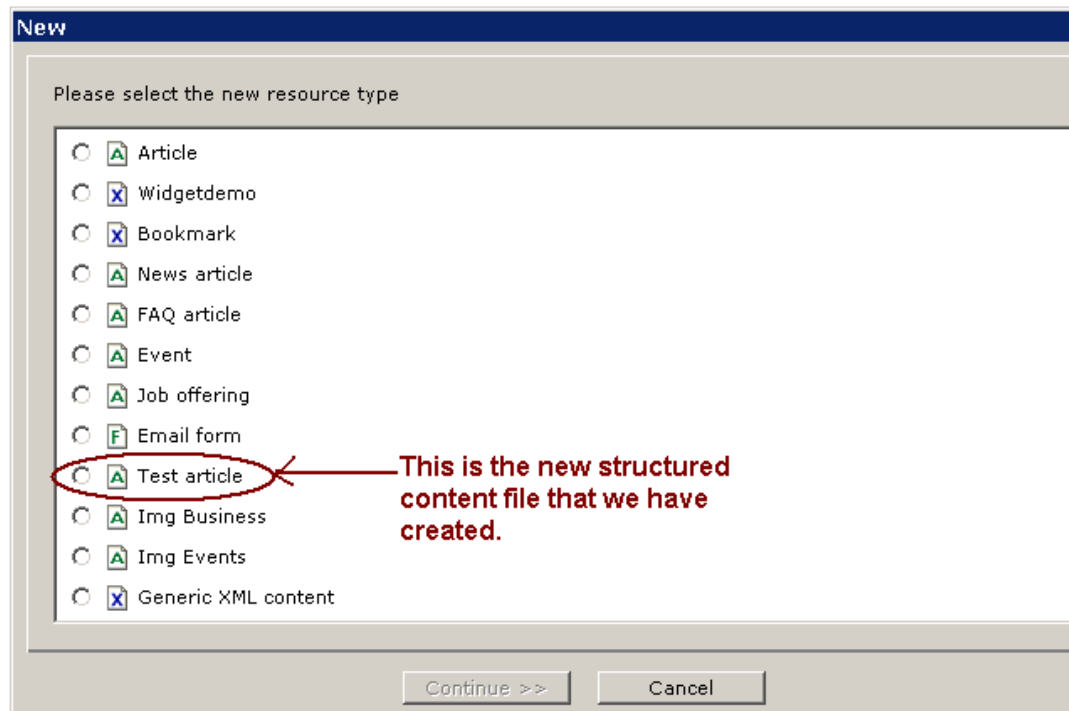
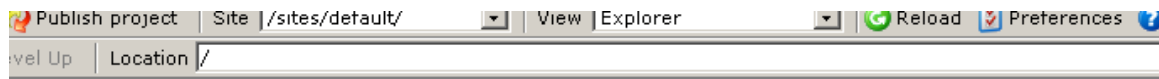


## Step 12:

After publishing the project you need to reload the OpenCMS instance in the server or restart the server.

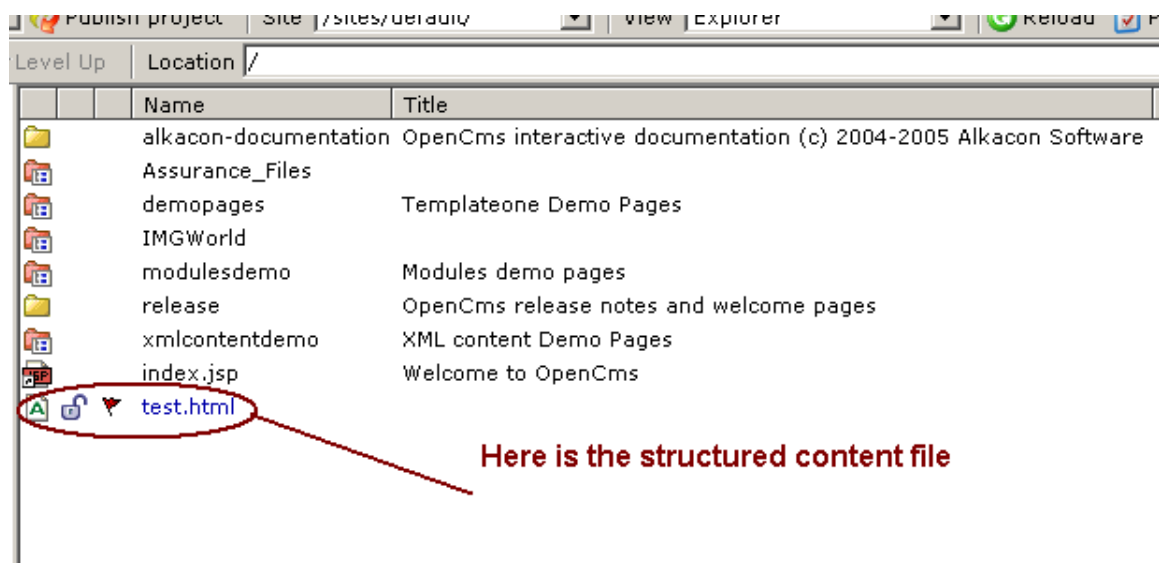
## Step 13:

Open the OpenCms Admin Interface now you can able to see your new structured content file.



#### Step 14:

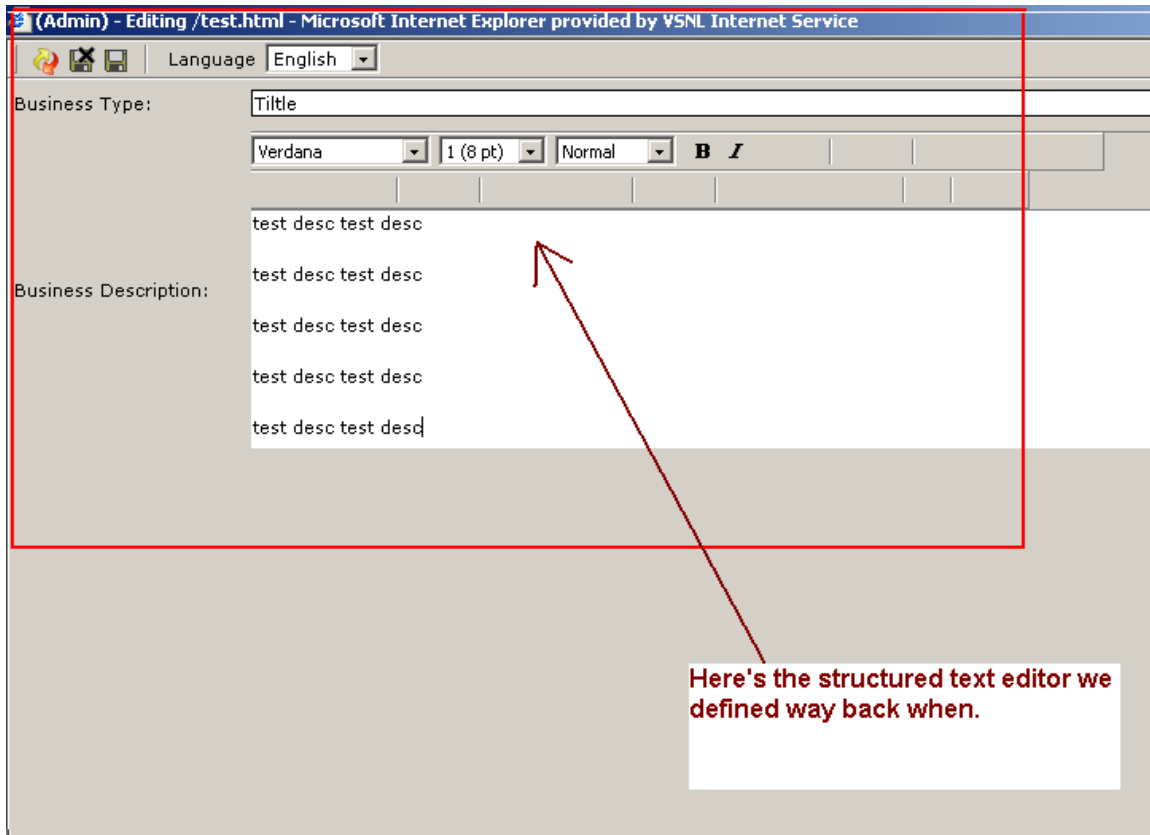
Now you can create a new xml file by using the newly created structured content file.





## Step 15:

Now you can edit your structured content file.



Enter the values in the field and save your changes.