



PDF SHARE FORMS

Online, Offline, OnDemand

PDF forms and SharePoint are better together

Populate dropdown with metadata

Product: PDF Share Forms Enterprise for SharePoint 2013

Contents

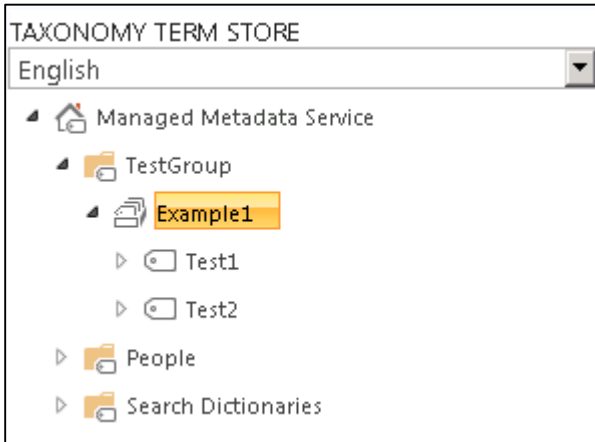
Managed metadata.....	2
Creating Managed Metadata column.....	4
Preparing template.....	6
Runtime.....	10



This guide will show how to connect a managed metadata column to a drop-down populated with terms from term store.

Managed metadata

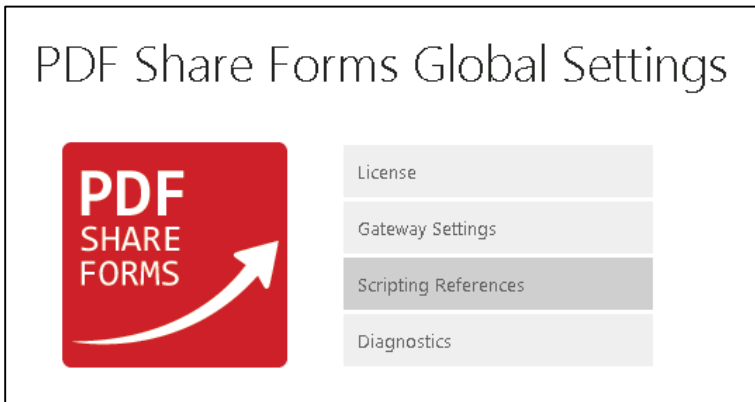
Step 1. Prepare metadata



There is one term store (Managed Metadata Service), one group (TestGroup) and one term set (Example1) with couple of terms.

Ensuring that proper assemblies are added

Step 1. Go to Central Administration → PDF Share Forms → Scripting Settings:





Step 2. Add Microsoft.SharePoint.Taxonomy assembly

Approved Assemblies

```
Microsoft.BusinessData, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c  
Microsoft.Office.Server, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c  
Microsoft.Office.Server.UserProfiles, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c  
Microsoft.SharePoint, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c  
System.Web.Extensions, Version=4.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35  
mscorlib, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089  
System.Data, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089  
PDFShareForms.Enterprise, Version=1.0.0.0, Culture=neutral, PublicKeyToken=ec1de91b62c9319e  
Microsoft.SharePoint.Taxonomy, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c
```

Step 3. Add Microsoft.SharePoint.Taxonomy namespace

Approved Namespaces

```
Microsoft.BusinessData.MetadataModel  
Microsoft.BusinessData.Runtime  
Microsoft.SharePoint  
Microsoft.SharePoint.Administration  
Microsoft.SharePoint.BusinessData.Infrastructure  
Microsoft.SharePoint.BusinessData.Runtime  
Microsoft.SharePoint.BusinessData.SharedService  
System.Xml.Linq  
System.Web.Script.Serialization  
System.Collections.Generic  
System.Data  
PDFShareForms.BCS  
Microsoft.SharePoint.Taxonomy
```

Creating Managed Metadata column

Step 1. Go to Share Point **Site Settings** → **Site Columns** → **Create** and choose Managed Metadata type column

Site Columns ▸ Create Column ⓘ

Name and Type

Type a name for this column, and select the type of information you want to store in the column.

Column name:

The type of information in this column is:

- Single line of text
- Multiple lines of text
- Choice (menu to choose from)
- Number (1, 1.0, 100)
- Currency (\$, ¥, €)
- Date and Time
- Lookup (information already on this site)
- Yes/No (check box)
- Person or Group
- Hyperlink or Picture
- Calculated (calculation based on other columns)
- Task Outcome
- Full HTML content with formatting and constraints for publishing
- Image with formatting and constraints for publishing
- Hyperlink with formatting and constraints for publishing
- Summary Links data
- Rich media data for publishing
- Managed Metadata

Step 2. Configure Term Set Setting

Term Set Settings

Enter one or more terms, separated by semicolons, and select Find to filter the options to only include those which contain the desired values.

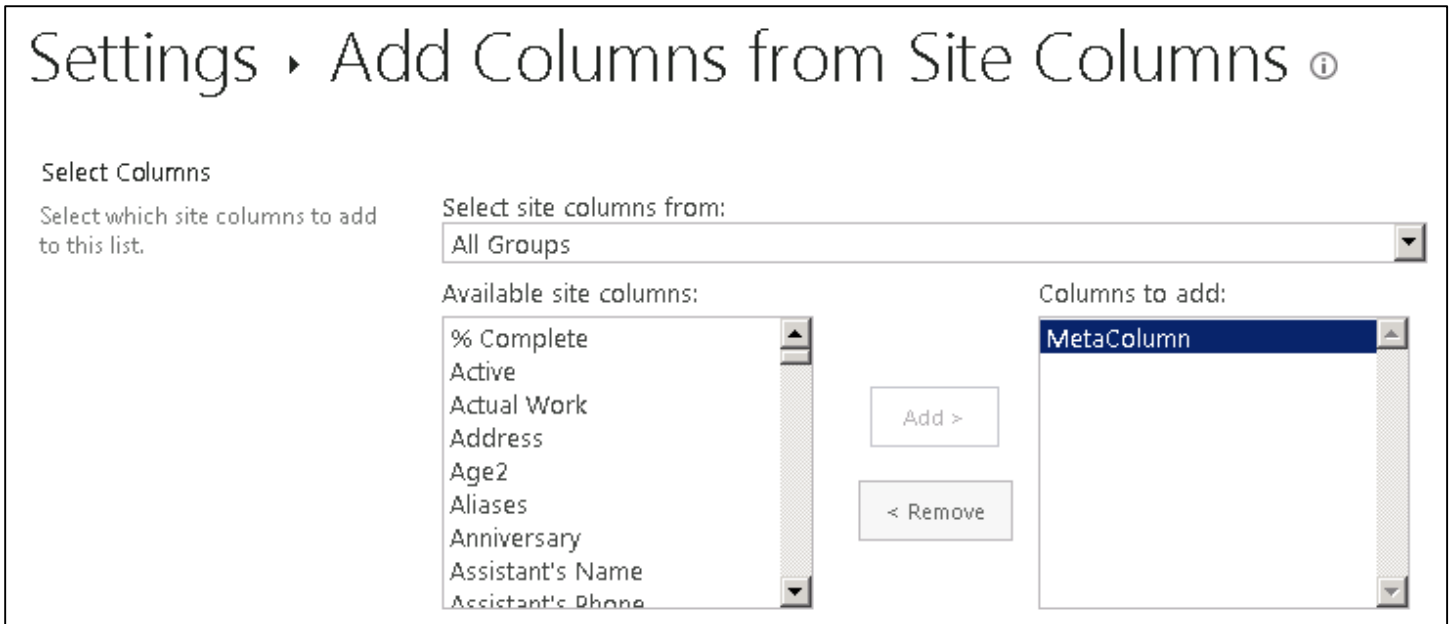
After finding the term set that contains the list of values to display options for this column, click on a term to select the first level of the hierarchy to show in the column. All levels below the term you select will be seen when users choose a value.

Use a managed term set:

Find term sets that include the following terms.

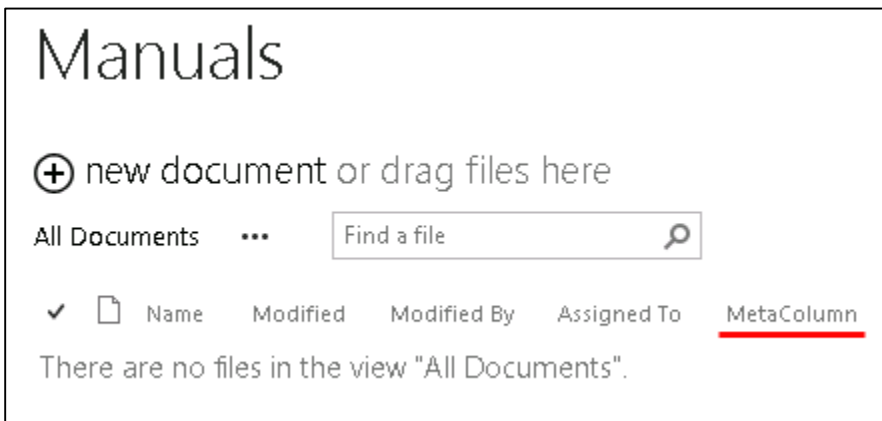
- ▾ Managed Metadata Service
 - ▾ People
 - ▾ Search Dictionaries
 - ▾ TestGroup
 - ▾ Example1
 - ▾ Test1
 - ▾ Test2

Step 3. Modify library view to show created column. Go to library where template will be deployed.
Go to **Library** → **Library Settings** → **Add from existing site columns**



The screenshot shows the 'Settings > Add Columns from Site Columns' interface. It features a 'Select Columns' section with instructions to 'Select which site columns to add to this list.' Below this is a 'Select site columns from:' dropdown menu set to 'All Groups'. There are two list boxes: 'Available site columns:' containing items like '% Complete', 'Active', 'Actual Work', 'Address', 'Age2', 'Aliases', 'Anniversary', 'Assistant's Name', and 'Assistant's Phone'; and 'Columns to add:' containing 'MetaColumn'. Between the list boxes are 'Add >' and '< Remove' buttons.

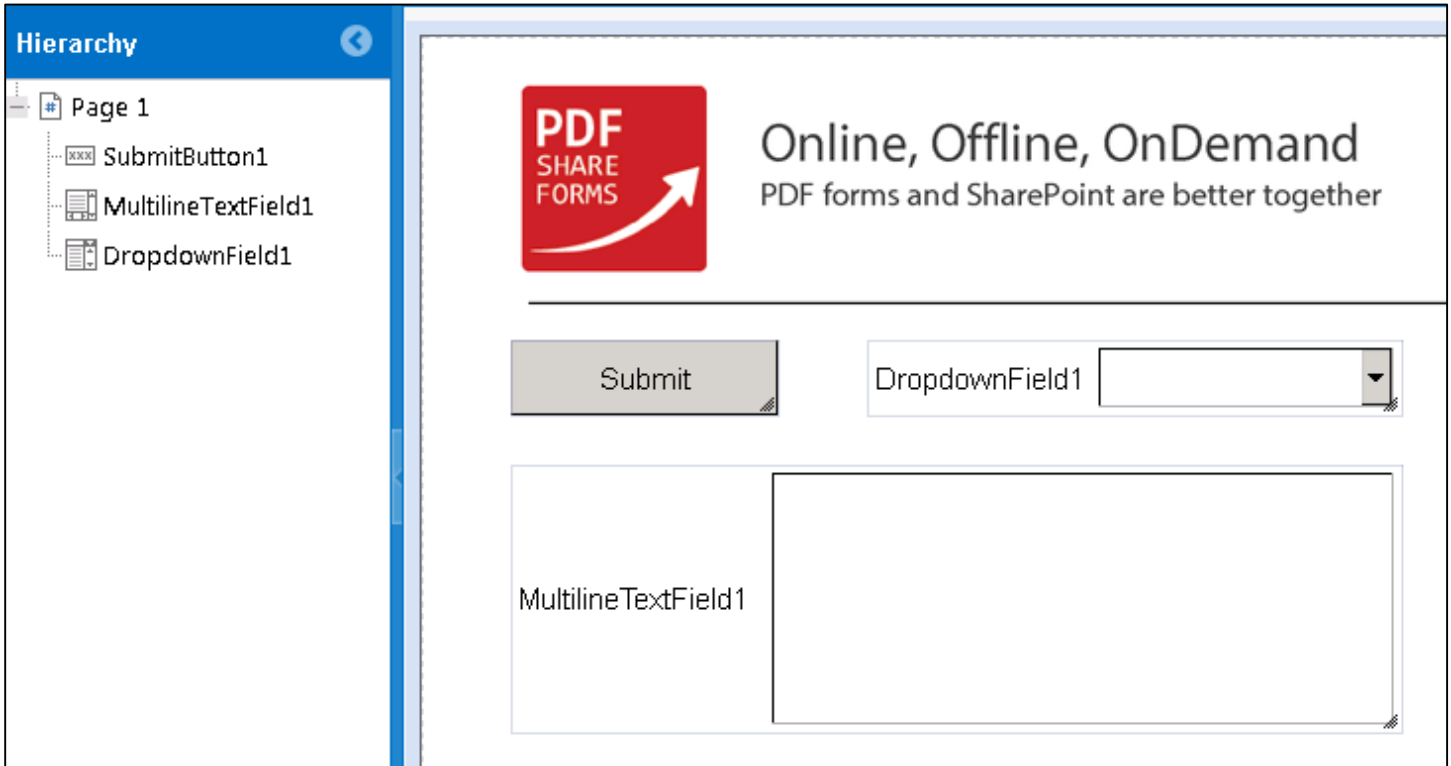
Step 4. Result



The screenshot shows the 'Manuals' library view. At the top, it says 'Manuals' and 'new document or drag files here'. Below that is a search bar with 'Find a file' and a magnifying glass icon. A table header is visible with columns: 'Name', 'Modified', 'Modified By', 'Assigned To', and 'MetaColumn' (which is underlined in red). Below the header, it states 'There are no files in the view "All Documents".'

Preparing template

Step 1. Add fields



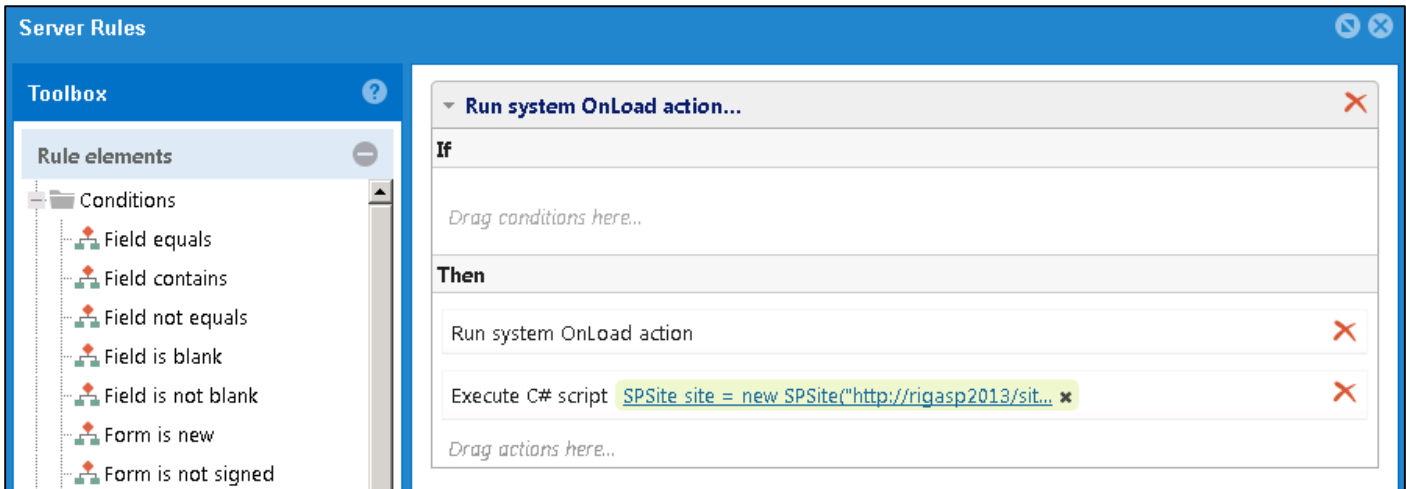
In this example is being used Submit button; DropdownField1 – to store values from Term Store and MultilineTextField – to store term label and ID.

NOTE: MultilineTextField1 should be set to hidden (select field → **Properties** → **Size & Position** → **Hidden**):



It is set to **Visible** only for demonstrating purposes.

Step 2. Add **Execute C# script** action under **Run system OnLoad action** in Form Load Rule.
Go to **Developer** → **Form Load**



Script:

```
//part that is marked with red must be changed for proper case
SPSite site = new SPSite("http://rigasp2013/sites/training005");
Dictionary<string,string> metDictionary = new Dictionary<string,string>();
TaxonomySession session = new TaxonomySession(site);
//Term Store name is marked with red
TermStore termstore = session.TermStores["Managed Metadata Service"];
//Term group is marked with red
Microsoft.SharePoint.Taxonomy.Group group = termstore.Groups["TestGroup"];
//Term Set is marked with red
TermSet termset = group.TermSets["Example1"];
TermCollection tc = termset.Terms;
foreach (Term t in tc)
{
    if (t.IsRoot)
    {
        metDictionary.Add(t.Name.ToString(),t.Id.ToString());
    }
}
JavaScriptSerializer serializer = new JavaScriptSerializer();
//write dictionary in to MultilineTextField1 field
data.resolveNode("MultilineTextField1").Value = serializer.Serialize(metDictionary);
```

This script connects to **Managed Metadata Service** and gets terms then store them in **MultilineTextField1**.

Step 3. For dynamic (xfa) forms. Add Client Script.
Go to **Developer → Global events → Form Ready**

Script:

```
var val = xfa.resolveNode("$data..DropdownField1").value;
eval("var yourdata=" + xfa.resolveNode("$data..MultilineTextField1").value + ";");
var dropdown = xfa.resolveNode("$form.Root.Default..DropdownField1")
for(var key in yourdata)
{
    dropdown.addItem(key, key);
}
xfa.resolveNode("$data..DropdownField1").value = val;
```

This script will take key values from MultilineTextField1 and populate dropdown.

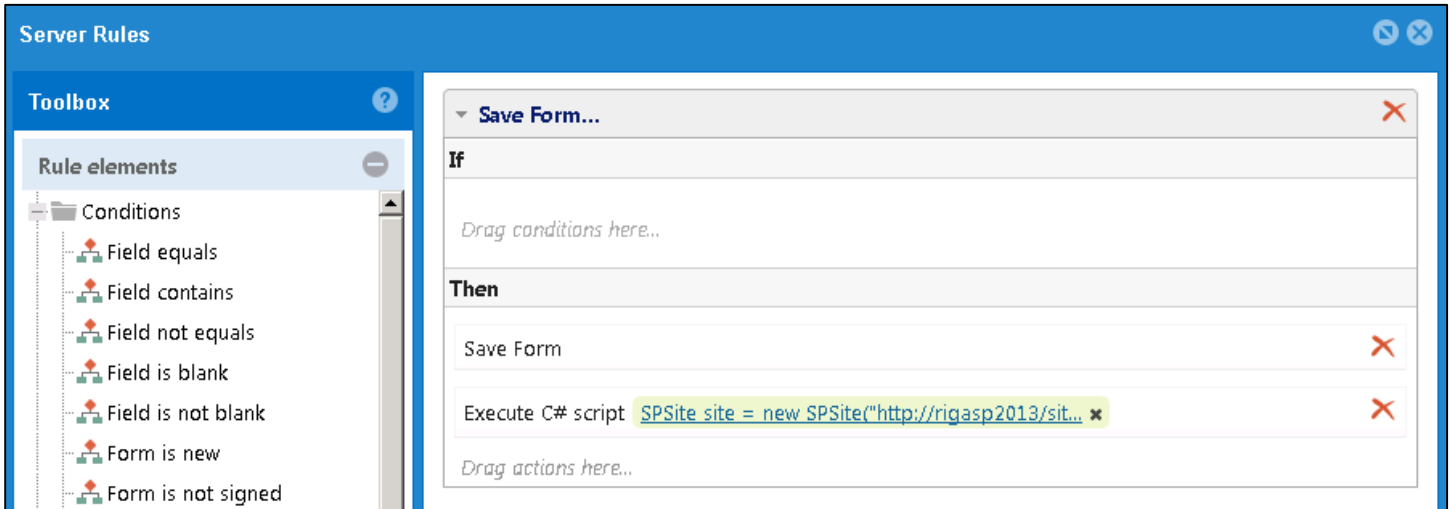
Step 3.* For flat (acro) forms. Add Client Script. Go to **Developer → Global JavaScript**

Script:

```
var val = this.getField("DropdownField1").value;
eval("var yourdata=" + this.getField("MultilineTextField1").value + ";");
var dropdown = this.getField("DropdownField1")
var i = 0;
for(var key in yourdata)
{
    dropdown.insertItemAt(key, key, i);
    i++;
}
this.getField("DropdownField1").value = val;
```

This script will take key values from MultilineTextField1 and populate dropdown.

Step 4. In **Form Submit Rule** add **Execute C# script** action under **Save Form** action. Go to **Developer** → **Form Submit**



Script:

```
//part that are marked with red should be changed for proper case
SPSite site = new SPSite("http://rigasp2013/sites/training005");
JavaScriptSerializer serializer = new JavaScriptSerializer();
Dictionary<string, string> values = serializer.Deserialize<Dictionary<string,
string>>(data.resolveNode("MultilineTextField1").Value);
SPList list = currentWeb.Lists["Manuals"]; //library's name
TaxonomyField taxonomyField = list.Fields["MetaColumn"] as TaxonomyField; //column where
metadata will be stored
TaxonomyFieldValue taxonomyFieldValue = new TaxonomyFieldValue(taxonomyField);
taxonomyFieldValue.TermGuid = values[data.resolveNode("DropdownField1").Value];
taxonomyFieldValue.Label = data.resolveNode("DropdownField1").Value;
SPListItem item = list.GetItemByUniqueId(form.DocumentID);
item["MetaColumn"] = taxonomyFieldValue;
item.Update();
```

This script fill in meta-column with value from dropdown.



Runtime

Step 1. Create new form and choose value in dropdown field

Submit DropDownField1 Test2

MultilineTextField1
{"Test1": "37c3105c-03d2-4d8b-8c7f-93094a976033", "Test2": "81abdc40-b00f-4c3a-a8ff-b4c2e1b26eae"}

Submit form.

Step 2. Result

+ new document or drag files here

All Documents Find a file

✓	Name	Modified	Modified By	Assigned To	MetaColumn
	Populate drop-down with metadata_1243 *	...	A few seconds ago	<input type="checkbox"/> Administrator	Test2