

Concepts for EPiServer Demo

Getting Started

The Concepts for EPiServer Demo is a set of page templates and EPiServer page types that showcases some of the features of Concepts for EPiServer. The following guide lists the demo requirements and the installation instructions in order to run the demo site.

Requirements

Ensure that you have the following installed before trying to install the demo.

1. Web3 Platform 1.3 (<http://www.networkedplanet.com/products/web3>)
2. EPiServer CMS 6 or later.
3. An EPiServer website with Concepts for EPiServer 1.0 installed.
4. Web3 Admin PowerShell command snap-in.

Installation

The demo package contains the schema file and topic map data file for the demo. These should be imported into the Web3 Platform before the module is installed. The files are located within the download package under '/Data'.

Import Web3 Schema and Topic Map using PowerShell

1. Open a new PowerShell prompt and load the Web3Admin snap-in
 - > Add-pssnapin Web3Admin
2. Import the schema into Web3
 - > New-Schema
 - > Web3BaseAddress: <address of Web3 site>
 - > FilePath: <path to 'organisation-schema.xml' file>
3. Open the Web3 server in a new browser and create a new topic map using the 'EPiServerWeb3Demo' schema.
4. In PowerShell, import the topic map data into the new topic map
 - > Import-TopicMap
 - > FilePath: <path to 'organisation-data.xtm' data file>
 - > BaseUri: <http://www.networkedplanet.com/epiweb3/types/>
 - > TopicMapAddress: <address of the topic map created in step 3>

Install Concepts for EPiServer Demo module

1. Open the EPiServer deployment centre
2. Select "Install a Module from a Compressed File"
3. Browse to and select the 'ConceptsForEPiServer-Demo-Install.epimodule' file.
4. The deployment centre will present the install options to 'Overwrite existing files' and 'Apply configuration changes'. Click Next.
5. Select an EPiServer site where the demo will be installed.
6. Then click "Install" to deploy the files.

Update the web.config

1. Open the web.config file for the EPiServer site where the demo module was installed.

2. Find the `<Web3Settings>` and change the following attributes;

Web3Service	The base URL where the Web3 service is installed. <i>e.g. http://localhost/web3</i>
TopicMapAddress	The full URL of the Topic Map where the demo data resides. <i>e.g. http://localhost/web3/topicmaps/cc05ce31-efa7-4dec-99c6-0ee443e32b35</i>

Import EPiServer Page Data

Sample pages and page types are included in the /Data folder named 'ConceptsForEPiServer-Demo.episerverdata' file. This can be imported into the EPiServer site using the Import Data function found in the site Admin Mode.

Run the Web3 Consistency Checker

Before the demo site can function, the consistency checker must be run to synchronise the EPiServer pages with the data in the topic map.

1. Open the demo site and navigate to the Admin interface
2. Find the scheduled job named 'Web3 Data Consistency Checker'
3. Click 'Start Manually'
4. Check that the job completes successfully.

The demo site is now installed.

Demo Overview

The demo site is designed to demonstrate how content in the Web3 Platform can be used within the EPiServer environment and vice versa. Topic content without the need for EPiServer pages can be displayed in an EPiServer site using page templates, as well as using EPiServer page types to map directly to topics in the topic map. The following guide will first briefly outline the topic map schema used, how to map page templates to topic types and how to link EPiServer pages to content within the Web3 Platform.

Topic Map Schema

The demo site uses a schema that contains the taxonomy of a fictional organisation. The schema has the following topic types that allow for certain associations:

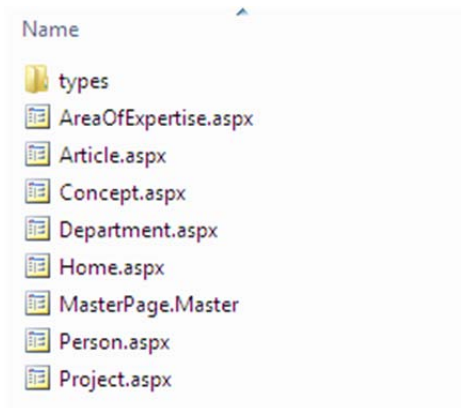
- Department
 - o Has members.
 - o Has sub-departments or is a sub-department for another department.
 - o Owns projects.
- Person
 - o Can be a member of a Department.
 - o Can have an expertise.
- Project
 - o Is owned by a Department.
 - o Can have a related Project.
- Area of Expertise

- Expertise can have a narrower or broader expertise.
- Can be related to another Expertise.

For a more detailed view of the schema, open the Web3 Platform and navigate to the schema named 'EPiServerWeb3Demo'.

Topic Type Mappings

Template pages have been installed at the root of the site under '/templates'. These template pages consist of ASPX files that can be used to render either topic data without the need for an EPiServer page type or for a configured EPiServer Page Type either representing a topic or simply fetching related topic data.



1 ASPX Template pages under the /templates directory

To configure the template pages, the topic subject identifier (SI) is mapped to the ASPX filename. These are added to the TopicTypeMappings section for the Web3Settings. For example, to map the template page name 'Person.aspx' to the SI 'http://www.networkedplanet.com/epiweb3/types/person' the following configuration would be added:

```
<Web3Settings>
<TopicTypeMappings>
  <add PageTemplate="Person"
    TopicTypeIdentifier="http://www.networkedplanet.com/epiweb3/types/person" />
  ...
</TopicTypeMappings>
```

2 Configuring the page template 'Person.aspx' to the SI 'http://www.networkedplanet.com/epiweb3/types/person'

Included EPiServer Page Types

Two page types have been created within the demo site. Each page type demonstrates the two basic concepts offered by Concepts for EPiServer.

Department Page Type

The department page type uses the PageTopicMappingProperty. This property allows a page created in EPiServer to represent a topic from the Web3 Platform.

Department

+ Add Property Settings

Move Up	Move Down	Name	Type	Tab	Unique value per language	Field name	Help Text
	↓	description	Long string (>255)	Content	No	description	
↑		DepartmentTopic	PageTopicMappingProperty	Content	No	Department Topic	

3 Department Page Type maps Department Topics to the page type

To map the property to the Department topic, the Subject Identifier has been selected as the Department topic, which has already been configured in the page topic mapping setting in web.config.

Edit Property

Common Settings Custom Settings

PageTopicMappingProperty

Settings

Subject Identifier: Department

Hierarchy Identifier: -- No Selection --

PageTopicMappingPropertyControl

This class does not have any custom settings.

Save Delete Cancel

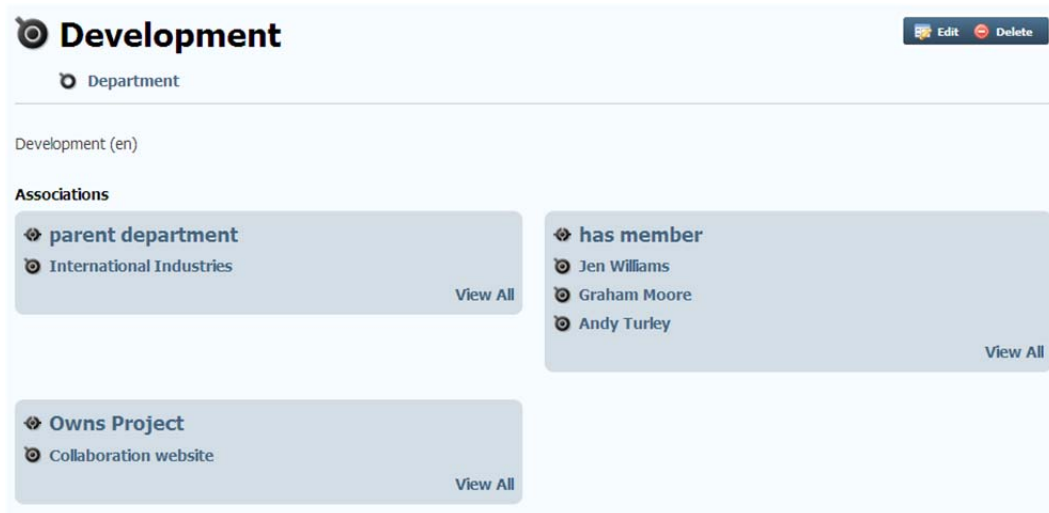
4 Setting the PageTopicMappingProperty to the Department subject identifier

In the demo site, pages for each department in the topic map have been created. Using the PageTopicMappingProperty these pages have been related to appropriate Department topics. For example the 'Development' department page, has been related to the 'Development' topic using the topic picker control.

The screenshot shows the EPiServer CMS interface for editing a page titled 'Development'. The page type is 'Department' and its status is 'Published version'. The 'Department Topic' field is set to 'Development'. A 'topic picker' dialog is open, showing a list of topics: Call Centre, Development, HR, and International Industries. The 'Development' topic is selected.

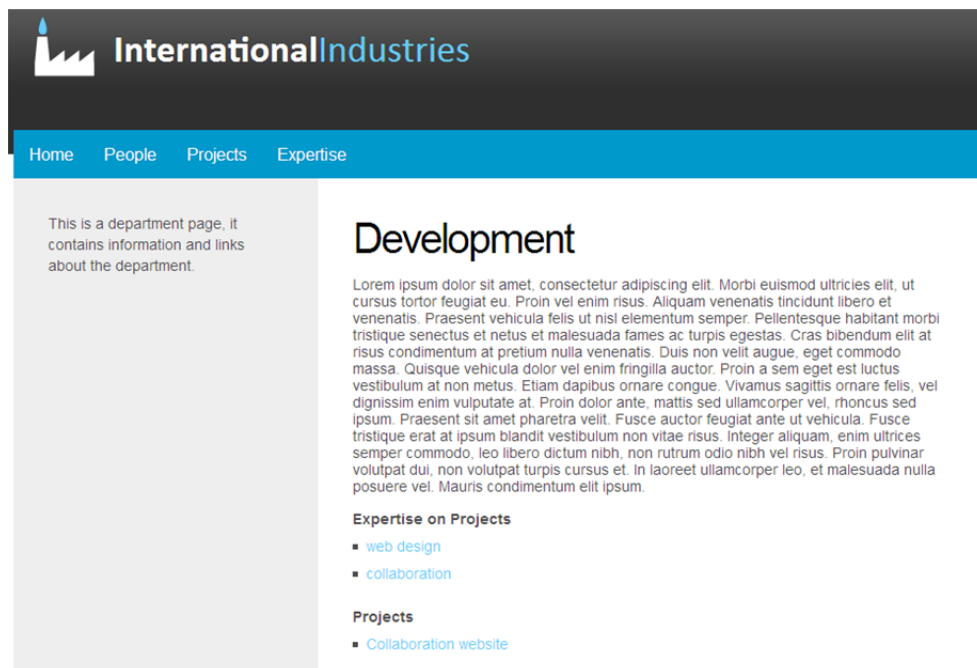
5 Mapping the Development EPiServer page to the Development topic using the topic picker

Since the page is mapped to the Development topic in the Web3 Platform, the same data available in for that topic is available to the Page using Concepts for EPiServer. For the Development topic, in the Web3 Platform we can see that it owns the Project 'Collaboration website'.



6 Viewing the Development topic in the Web3 Platform

In the EPiServer Development page, there is no specific reference to the 'Collaboration topic' within EPiServer, however since the page has been mapped to the Development topic, any associations that belong to the topic are available using the Concepts for EPiServer API.



7 Browsing to the EPiServer 'Development' page

Browsing to the 'Development' page in EPiServer, we can see that the page has under Projects heading, the 'Collaboration website' topic link displayed, despite the topic not having a related EPiServer page. The Department page is able to do this because it represents the actual 'Department' topic in the topic map.

The Department ASPX file, in the Page_Load method uses the extension method GetTopicForPage(), which returns the Topic object in the topic map, which in this case is the Department topic.

```
public partial class Department : TemplatePage
{
    private Topic _topic;

    protected void Page_Load(object sender, EventArgs e)
    {
        // try and get the topic as though it was a normal concept page
        _topic = this.GetTopic();
        // if no topic see if there is a mapped topic
        if (_topic == null)
        {
            _topic = this.GetTopicForPage();
        }
    }
}
```

8 Using the GetTopicForPage() extension method to get the Topic object for the page

Looking in the topic map in the Web3 Platform, the association for Department -> Project uses the role 'Owns Project'.



9 The Department Responsible for Project Association relating Department to Project

The Topic object has an extension method RelatedTopicLinks(). This method accepts either the full role SI or partial role SI if this has been configured in Web3Settings. In the case of the Department Responsible for Project association, the role of the Department in the association is 'ownsProject'.

```
public IEnumerable<Link> Projects
{
    get { return _topic.RelatedTopicLinks("ownsProject"); }
}
```

10 Using the RelatedTopicLinks method to fetch Links to Projects owned by Department.

As well as using the standard API methods for fetching related links and Pages, it is also possible to execute a SPARQL query to fetch related topics in a more complex manner. To fetch the 'Expertise on projects' topics on the Department page that appear under the heading, the following SPARQL query is used to fetch Expertise topics that are related to Projects owned by the Department:

```
string query = @"PREFIX tms: <http://www.networkedplanet.com/tmsparql/> ";
query += "PREFIX ont: <"+ Web3Configuration.Instance.SchemaPrefix + "> ";
query += "SELECT ?expertise ?label WHERE { <" + _topic.WebAddress;
```

```
query += "> ont:ownsProject ?project . ?project ont:relatedExpertise ?expertise";
query += " . ?expertise tms:label ?label }";
```

```
XDocument result = Web3DataHelper.ExecuteSparql(query);
```

11 SPARQL query to fetch related Expertise topics for Projects owned by the Department

Article Page Type

Article pages do not use the PageTopicMappingProperty but can still be related to topics in the topic map. Using the RelatedTopicProperty, the Article page type can be related to any topic in the Web3 Platform. Article uses two of these properties to relate the EPiServer page to Expertise and Person (author) topics.

Article

article

[Add Property](#) [Settings](#)

Move Up	Move Down	Name	Type	Tab	Unique value per language	Field name	Help Text
	↓	Body	Long string (>255)	Content	No	Body	
↑	↓	relatedExpertise	RelatedTopicProperty	Content	No	relatedExpertise	
↑		author	RelatedTopicProperty	Content	No	author	

12 Article Page Type maps Author and Expertise topics to the page type

As with the PageTopicMappingProperty the RelatedTopicProperty uses an SI for the topic type that it should be used to create links from. For example, the Author property uses the 'Person' topic type SI in order to be able to select Person type topics as an author.

Edit Property

[Common Settings](#) [Custom Settings](#)

RelatedTopicProperty

Settings

Subject Identifier

Hierarchy Identifier

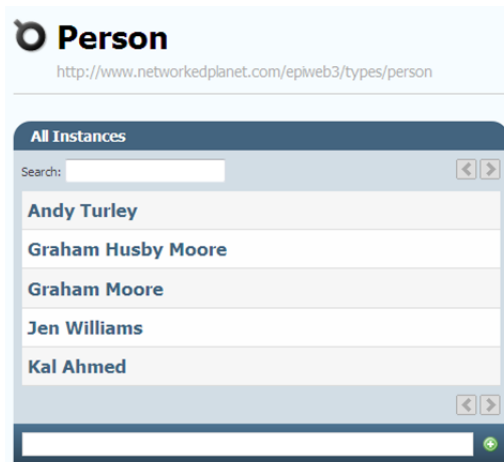
RelatedTopicPropertyControl

This class does not have any custom settings.

[Save](#) [Delete](#) [Cancel](#)

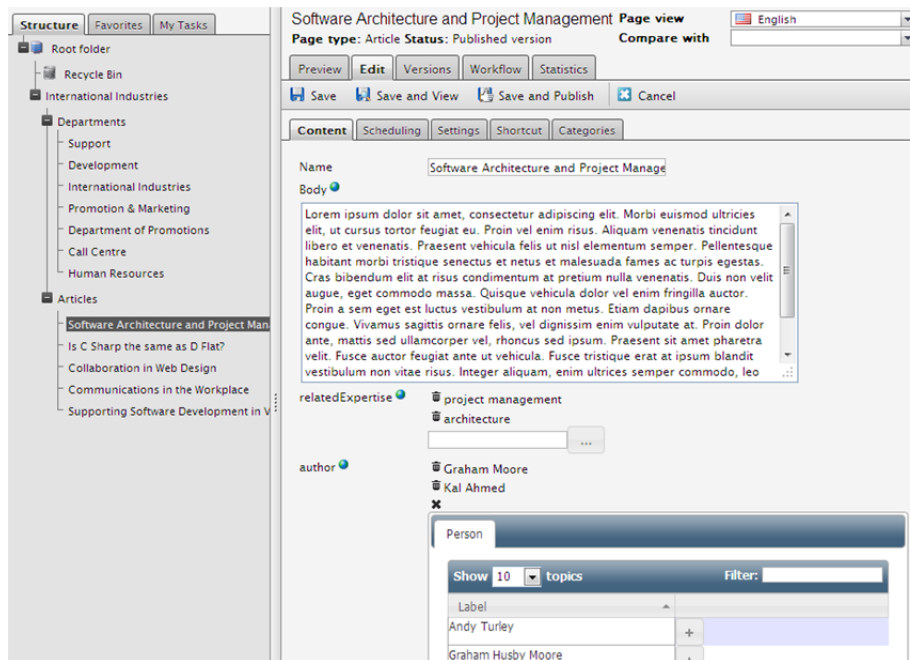
13 Setting the Author property to use Person SI topic type.

In the topic map in the Web3 Platform, the same Person topic types can be selected using the topic picker from Article page types.



14 Person topic types in Web3 Platform

The article page 'Software Architecture and Project Management' links to two Person topics 'Graham Moore' and 'Kal Ahmed' as related authors.



15 Mapping Authors and Related Expertise topics to an Article page

To fetch the related author links in the article page, the extension method RelatedTopicLinks() on PageData is used by supplying the property name 'author'. Since the Article page has no topic in the topic map it is unnecessary to fetch a topic for the current page.

```
public IEnumerable<Link> Authors
{
    get { return CurrentPage.RelatedTopicLinks("author"); }
}
```

16 Fetching related topic links by using the property name 'author'.

Concept Pages

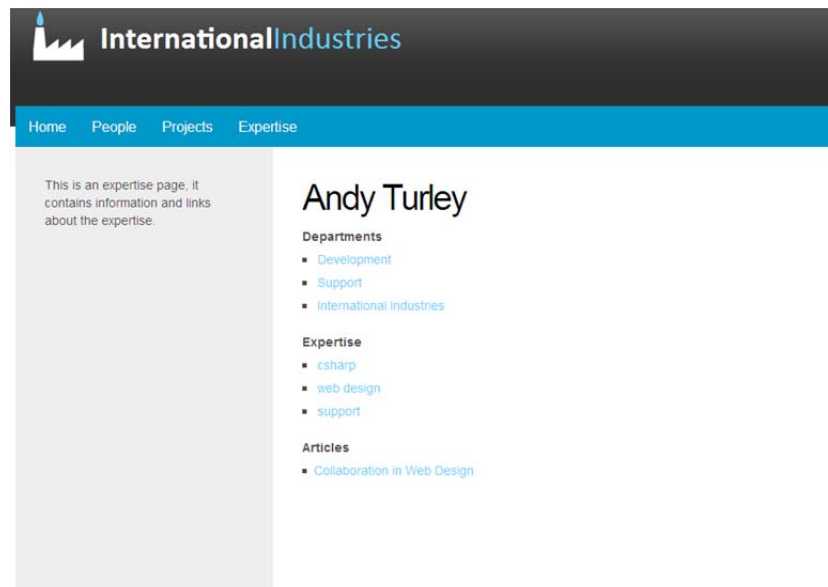
To render a page for a topic that doesn't have a page in EPiServer, template pages are used. As shown in the Topic Type Mappings section, template pages can be mapped to a topic type as an ASPX page to render content.

For example, the topic type Person has been mapped to the ASPX page Person.aspx. Any data from the topic in the Web3 Platform can be retrieved from the ASPX code behind.



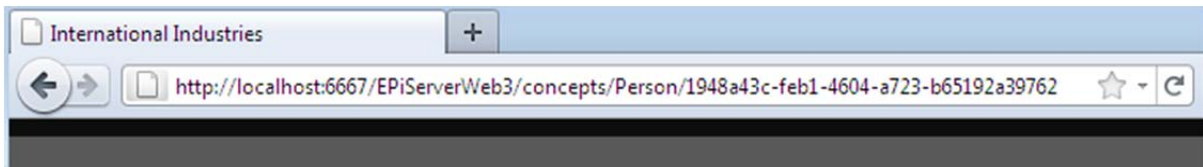
17 Person topic type in the Web3 Platform

From the Person topic type, the associations from 'member of Department' and 'Expert In' have been rendered in the Person.aspx page.



18 Person topic type page rendering template.

Concept page templates will be for a topic instance when they are loaded. This is determined by the topic GUID supplied in the URL. Concept pages use the URL format `/concepts/{template mapping}/{topic guid}` to determine the rendering page template to use. For example a Person type Concept page URL may look something like the following:



19 Concept page mapping URL format

Any links that are generated to other Concept pages for example using the Web3DataHelper method RelatedTopicLinks() will create links in this format to other Concept pages.

To retrieve the Topic object that the page template has been loaded for, the System.Web.UI.Page extension method GetTopic() will return the Topic for the current request.

```
private Topic _topic;

protected void Page_Load(object sender, EventArgs e)
{
    _topic = this.GetTopic();
    ...
}
```

20 Loading the Page Topic using the GetTopic() method.

With the Topic object retrieved, related links and pages can be retrieved, as well as the Topic label:

```
public string GetPersonLabel()
{
    // simple call to get the display label
    return _topic.GetDisplayLabel();
}
```

21 Using Topic GetDisplayLabel() to fetch the topic label.

To fetch related links, the RelatedTopicLinks() Topic extension method is used by passing in the partial SI of the desired role type.

```
public IEnumerable<Link> Expertise
{
    get { return _topic.RelatedTopicLinks("expertIn"); }
}
```

22 Fetching related Expertise topic links by the partial role SI 'expertIn'.

Since topics may have EPiServer pages related to them using the PageTopicMappingProperty in the EPiServer site, a topic that has a template page only can fetch EPiServer pages by using the property name that was used to associate it. For example, as was seen in the Article page type, this used an 'author' RelatedTopicProperty to associate Person topics. Using the Topic extension method RelatedPages(), the same 'author' property name can be used to fetch the related EPiServer pages.

```
public string GetArticles()
{
    PageDataCollection pages = _topic.RelatedPages("author");
}
```

23 Fetching related EPiServer Article pages using the RelatedPages() method on Topic.