



ALLAN THRAEN | 16 years ago | PDF |

FUN WITH THE MAPPED PAGE PROVIDER



Here's two more small code samples I did in less than 30 min. based on the [Mapped Page Provider](#).

Filesystem Page Provider

This provider will expose folder on the file-system as pages in EPiServer CMS 5 R2.

```
public class FileProvider : MappedPageProvider
{
    private string rootFolder;

    public override void Initialize(string name, System.Collections.Specialized.NameValueCollection config)
    {
        base.Initialize(name, config);
        rootFolder = config["rootFolder"];
    }
}
```

It takes a "rootFolder" from web.config, and shows all files and folders below that. As a unique key it uses the path to the folder/file - and it simply uses the filename without extension as a PageName.

If a file is a .txt file, the contents will be put into the MainBody property - otherwise some basic file/directory info like creation time and last access time is shown there.

This version doesn't support modifying or changing files - but it would be relatively easy to implement. Web.config has these registration lines:

```
<pageProvider>
  <providers>
    <add name="File" type="EPiServer.Templates.PageProvider.FileProvider,EPiServer.Templates.Public"
    entryPoint="28" rootFolder="d:\temp"/>
  </providers>
</pageProvider>
```

And of course the NETWORK_SERVICE user has read access to that part of my harddisk.

EPiServer Profile Page Provider

Another idea I had was to use the EPiServerProfiles to create "User Profile pages". Now, that's also fairly easy:

```
public class ProfileProvider : MappedPageProvider
{
    protected override List<string> GetRootChildren()
    {
        List<string> l = new List<string>();
        foreach (MembershipUser u in Membership.GetAllUsers())
        {
            l.Add(u.UserName);
        }
        return l;
    }
}
```

This time, we only have one level to worry about - the root. I suppose I could have organized the profiles based on their roles - but anyway - this is a 5 min demo.

Unique key is the username. MainBody is set to the users email address - but could be any setting stored in the profile. Web.config registration is so plain and simple that there's no reason to show it here.

CodeArt ApS

Teknikerbyen 5, 2830 Virum, Denmark

Email: info@codeart.dk

Phone: +45 26 13 66 96

CVR: 39680688

