

**Asset ID: 23057 /Demo Site - Code Examples/\_Partner Summit/CachedAssets/\_CachedAssetsBad Template ID: 22961**

Code Sample 1a – Original code to render XML – Template has two common mistakes. First it is reloading lookup assets throughout a loop, which can often load the same asset over and over again. Second the filter list is not using fieldnames, requiring the fields to be loaded in a secondary call.

```
<%@ Page Language="C#" Inherits="CrownPeak.Internal.Debug.OutputInit" %>
<%@ Import Namespace="CrownPeak.CMSAPI" %>
<%@ Import Namespace="CrownPeak.CMSAPI.Services" %>
<%@ Import Namespace="CrownPeak.CMSAPI.CustomLibrary" %>
<!--DO NOT MODIFY CODE ABOVE THIS LINE-->
<% if(context.IsPublishing == false) { %><textarea style="width: 100%; height: 900px;"><% } %>
<?xml version="1.0"?>
<pages>
<%
    Asset rootFolder = Asset.Load("/") + asset.AssetPath[0] + "/";
    if(rootFolder.IsLoaded)
    {
        //setup filter for filter list
        FilterParams fp = new FilterParams();
        fp.Add(AssetPropertyNames.TemplateLabel, Comparison.Equals, "Page Asset");

        //load filter list based on filter params
        List<Asset> pageList = rootFolder.GetFilterList(fp);

        //loop through filter list and process results - xml node
        foreach(Asset page in pageList)
        {
            Out.WriteLine("<item>");
            Out.WriteLine("<asset_id>" + page.Id.ToString() + "</asset_id>");
            Out.WriteLine("<page_label><![CDATA[" + page["page_label"] + "]]></page_label>");

            //load lookup asset and render output based on that asset
            Asset attach = Asset.Load(page["attach"]);
            if(attach.IsLoaded)
            {
                Out.WriteLine("<attach_title>" + attach["off_title"] + "</attach_title>");
            }

            Out.WriteLine("</item>");
        }
    }
%>
</pages>
<% if(context.IsPublishing == false) { %></textarea><% } %>
```

Asset ID: 23059 /Demo Site - Code Examples/\_ Partner Summit/CachedAssets/\_CachedAssetsGood Template ID: 22964

Code Sample 1b – Code is updated to preload the lookup assets and populate a dictionary object. It also uses fieldnames parameter whenever loading a filter list to improve performance.

```
<%@ Page Language="C#" Inherits="CrownPeak.Internal.Debug.OutputInit" %>
<%@ Import Namespace="CrownPeak.CMSAPI" %>
<%@ Import Namespace="CrownPeak.CMSAPI.Services" %>
<%@ Import Namespace="CrownPeak.CMSAPI.CustomLibrary" %>
<!--DO NOT MODIFY CODE ABOVE THIS LINE-->
<% if(context.IsPublishing == false) { %><textarea style="width: 100%; height: 900px;"><% } %>
<?xml version="1.0"?>
<pages>
<%
    //Dictionary will be created for all of the lookup assets - this will be used to render the appropriate values
    Dictionary<string, string> titleDict = new Dictionary<string, string>();

    //build a filter list of lookup assets - this will be used to populate the dictionary
    FilterParams titleParams = new FilterParams();
    titleParams.Add(AssetPropertyNames.TemplateLabel, Comparison.Equals, "Lookup Asset");
    //set field names - this improves performance of the results, returning used fieldnames
    titleParams.FieldNames = Util.MakeList("off_title");

    //populate a dictionary object with the lookup values
    Asset titleFolder = Asset.Load("/") + asset.AssetPath[0] + "_Partner Summit/Lookup Assets/");
    List<Asset> titleAssets = titleFolder.GetFilterList(titleParams);
    foreach (Asset title in titleAssets)
    {
        titleDict.Add(title.Id.ToString(), title["off_title"]);
    }

    Asset rootFolder = Asset.Load("/") + asset.AssetPath[0] + "/");
    if(rootFolder.IsLoaded)
    {
        //setup filter for filter list
        FilterParams fp = new FilterParams();
        fp.Add(AssetPropertyNames.TemplateLabel, Comparison.Equals, "Page Asset");
        //set field names - this improves performance of the results, returning used fieldnames
        fp.FieldNames = Util.MakeList("page_label", "attach_id");

        //load filter list based on filter params
        List<Asset> pageList = rootFolder.GetFilterList(fp);

        foreach(Asset page in pageList)
        {
```







```

        Out.WriteLine("<item>");
        Out.WriteLine("<asset_id>" + page.Id.ToString() + "</asset_id>");
        Out.WriteLine("<page_label><![CDATA[" + page["page_label"] + "]]></page_label>");
        //read from the lookup dictionary instead of loading the asset each time
        if(titleDict.ContainsKey(page["attach_id"]))
        {
            Out.WriteLine("<attach_title>" + titleDict[page["attach_id"]] + "</attach_title>");
        }
        Out.WriteLine("</item>");
    }
}

%>
</pages>
<% if(context.IsPublishing == false) { %></textarea><% } %>

```

Looking at the rendered stats, GetField, LoadDirect & Load are all considerably higher before the changes than after. Under size and load, this can result in very long running templates, include ones that often time out or backup publishing.

<div data-bbox="380 94 926 1161"> <p><b>Overview</b></p> <p><b>Information</b>   </p> <p><a href="#">Preview</a>   <a href="#">Edit</a>   <a href="#">Form</a>   <a href="#">Refresh</a></p> <p>Label : _CacheAssetsBad  Asset Id : 23057  Type : File  Last Modified By : Dae Kim  Last Modified On : 10/23/2014 2:43:06 AM</p> <p>Status : N/A  Status Modified By : Dae Kim  Status Modified On : 10/23/2014 2:43:06 AM</p> <p>Template : /System/Templates/_Partner  Summit/CacheAssetsBad  Template Id : 22961</p> <p><b>Performance Data</b>  Execution Time : 1.43659</p> <table border="1"> <thead> <tr> <th>Method</th> <th>Count</th> <th>Duration (sec)</th> </tr> </thead> <tbody> <tr> <td>GetFilterList</td> <td>1</td> <td>0.26978</td> </tr> <tr> <td>GetField</td> <td>126</td> <td>0.13151</td> </tr> <tr> <td>LoadDirect</td> <td>44</td> <td>0.05269</td> </tr> <tr> <td>Load</td> <td>22</td> <td>0.02538</td> </tr> <tr> <td>GetContentFields</td> <td>1</td> <td>0.00081</td> </tr> <tr> <td>Other Non-API</td> <td>0</td> <td>0.95642</td> </tr> </tbody> </table> <p><a href="#">View More Details</a></p> </div>	Method	Count	Duration (sec)	GetFilterList	1	0.26978	GetField	126	0.13151	LoadDirect	44	0.05269	Load	22	0.02538	GetContentFields	1	0.00081	Other Non-API	0	0.95642	<div data-bbox="1062 94 1608 1161"> <p><b>Overview</b></p> <p><b>Information</b>   </p> <p><a href="#">Preview</a>   <a href="#">Edit</a>   <a href="#">Form</a>   <a href="#">Refresh</a></p> <p>Label : _CacheAssetsGood  Asset Id : 23059  Type : File  Last Modified By : Dae Kim  Last Modified On : 10/23/2014 2:43:07 AM</p> <p>Status : N/A  Status Modified By : Dae Kim  Status Modified On : 10/23/2014 2:43:06 AM</p> <p>Template : /System/Templates/_Partner  Summit/CacheAssetsGood  Template Id : 22964</p> <p><b>Performance Data</b>  Execution Time : 1.53541</p> <table border="1"> <thead> <tr> <th>Method</th> <th>Count</th> <th>Duration (sec)</th> </tr> </thead> <tbody> <tr> <td>GetFilterList</td> <td>2</td> <td>0.81543</td> </tr> <tr> <td>LoadDirect</td> <td>3</td> <td>0.00672</td> </tr> <tr> <td>Load</td> <td>2</td> <td>0.00450</td> </tr> <tr> <td>GetField</td> <td>1</td> <td>0.00101</td> </tr> <tr> <td>GetContentFields</td> <td>1</td> <td>0.00084</td> </tr> <tr> <td>MakeList</td> <td>2</td> <td>0.00001</td> </tr> <tr> <td>Other Non-API</td> <td>0</td> <td>0.70690</td> </tr> </tbody> </table> <p><a href="#">View More Details</a></p> </div>	Method	Count	Duration (sec)	GetFilterList	2	0.81543	LoadDirect	3	0.00672	Load	2	0.00450	GetField	1	0.00101	GetContentFields	1	0.00084	MakeList	2	0.00001	Other Non-API	0	0.70690
Method	Count	Duration (sec)																																												
GetFilterList	1	0.26978																																												
GetField	126	0.13151																																												
LoadDirect	44	0.05269																																												
Load	22	0.02538																																												
GetContentFields	1	0.00081																																												
Other Non-API	0	0.95642																																												
Method	Count	Duration (sec)																																												
GetFilterList	2	0.81543																																												
LoadDirect	3	0.00672																																												
Load	2	0.00450																																												
GetField	1	0.00101																																												
GetContentFields	1	0.00084																																												
MakeList	2	0.00001																																												
Other Non-API	0	0.70690																																												
<p>Before Changes – Note high GetField, LoadDirect &amp; Load</p>	<p>After Changes – Note that GetField, LoadDirect &amp; Load are all considerably lower, improving performance for the template.</p>																																													

Code Example 2a – This example consists of a list panel, which links to external assets. The template loops through the panel to pull content from the assets that are linked. What is less obvious is that each field reference is an actual DB call to retrieve the field.

```
<%@ Page Language="C#" Inherits="CrownPeak.Internal.Debug.OutputInit" %>
<%@ Import Namespace="CrownPeak.CMSAPI" %>
<%@ Import Namespace="CrownPeak.CMSAPI.Services" %>
<%@ Import Namespace="CrownPeak.CMSAPI.CustomLibrary" %>
<!--DO NOT MODIFY CODE ABOVE THIS LINE-->
<%//This plugin uses OutputContext as its context class type%>
<%
    //loop through list panel in asset to get lookup assets
    List<PanelEntry> list = asset.GetPanels("lookup_asset");
    foreach(PanelEntry pe in list)
    {
        //load asset linked in the panel
        Asset lookupAsset = Asset.Load(pe["lookup_asset"]);
        if(lookupAsset.IsLoaded)
        {
            //render out fields
            Out.WriteLine("Office Title: " + lookupAsset["off_title"] + "<br />");
            Out.WriteLine("Office Name: " + lookupAsset["off_office_name1"] + "<br />");
            Out.WriteLine("Address Line 1: " + lookupAsset["off_address_line1"] + "<br />");
            Out.WriteLine("Address Line 2: " + lookupAsset["off_address_line2"] + "<br />");
            Out.WriteLine("City: " + lookupAsset["off_city"] + "<br />");
            Out.WriteLine("Region: " + lookupAsset["off_region"] + "<br />");
            Out.WriteLine("Country: " + lookupAsset["off_country"] + "<br />");
            Out.WriteLine("State Text: " + lookupAsset["off_state"] + "<br />");
            Out.WriteLine("Zip: " + lookupAsset["off_zip"] + "<br />");
            Out.WriteLine("Address Line 3: " + lookupAsset["off_address_line3"] + "<br />");
            Out.WriteLine("Address Line 4: " + lookupAsset["off_address_line4"] + "<br />");
            Out.WriteLine("Phone: " + lookupAsset["off_phone"] + "<br />");
            Out.WriteLine("Fax: " + lookupAsset["off_fax"] + "<br />");
            Out.WriteLine("X Coordinate: " + lookupAsset["x"] + "<br />");
            Out.WriteLine("Y Coordinate: " + lookupAsset["y"] + "<br />");

            Out.WriteLine("<br /><br /><br />");
        }
    }
%>
```

**Asset ID: 23063 /Demo Site - Code Examples/\_ Partner Summit/GetContent/GetContentGood Template ID: 22971**




Code Example 2b – This is the same code with a single difference. Once the external asset is loaded, the content is cached by using Asset.GetContent(). This call will load all of the content fields for the asset in one call instead of individually accessing the fields. The fields are then cached, streamlining processing time.

```
<%@ Page Language="C#" Inherits="CrownPeak.Internal.Debug.OutputInit" %>
<%@ Import Namespace="CrownPeak.CMSAPI" %>
<%@ Import Namespace="CrownPeak.CMSAPI.Services" %>
<%@ Import Namespace="CrownPeak.CMSAPI.CustomLibrary" %>
<!--DO NOT MODIFY CODE ABOVE THIS LINE-->
<%//This plugin uses OutputContext as its context class type%>
<%
    //loop through list panel in asset to get lookup assets
    List<PanelEntry> list = asset.GetPanels("lookup_asset");
    foreach(PanelEntry pe in list)
    {
        //load asset linked in the panel
        Asset lookupAsset = Asset.Load(pe["lookup_asset"]);
        if(lookupAsset.IsLoaded)
        {
            //load content fields - this will cache the fields in one call
            lookupAsset.GetContent();
            //render out fields
            Out.WriteLine("Office Title: " + lookupAsset["off_title"] + "<br />");
            Out.WriteLine("Office Name: " + lookupAsset["off_office_name1"] + "<br />");
            Out.WriteLine("Address Line 1: " + lookupAsset["off_address_line1"] + "<br />");
            Out.WriteLine("Address Line 2: " + lookupAsset["off_address_line2"] + "<br />");
            Out.WriteLine("City: " + lookupAsset["off_city"] + "<br />");
            Out.WriteLine("Region: " + lookupAsset["off_region"] + "<br />");
            Out.WriteLine("Country: " + lookupAsset["off_country"] + "<br />");
            Out.WriteLine("State Text: " + lookupAsset["off_state"] + "<br />");
            Out.WriteLine("Zip: " + lookupAsset["off_zip"] + "<br />");
            Out.WriteLine("Address Line 3: " + lookupAsset["off_address_line3"] + "<br />");
            Out.WriteLine("Address Line 4: " + lookupAsset["off_address_line4"] + "<br />");
            Out.WriteLine("Phone: " + lookupAsset["off_phone"] + "<br />");
            Out.WriteLine("Fax: " + lookupAsset["off_fax"] + "<br />");
            Out.WriteLine("X Coordinate: " + lookupAsset["x"] + "<br />");
            Out.WriteLine("Y Coordinate: " + lookupAsset["y"] + "<br />");

            Out.WriteLine("<br /><br /><br />");
        }
    }
%>
```

Looking at the rendered stats, the big savings is the GetField calls, which drops from 800 to 100, improving overall performance for rendering.

**Overview**

**Information**   

[Preview](#) | [Edit](#) | [Form](#) | [Refresh](#)

Label : GetContentBad  
 Asset Id : 23061  
 Type : File  
 Last Modified By : Dae Kim  
 Last Modified On : 10/23/2014 2:43:21 AM

Status : N/A  
 Status Modified By : Dae Kim  
 Status Modified On : 10/23/2014 2:43:07 AM

Template : /System/Templates/\_Partner  
 Summit/GetContentBad  
 Template Id : 22967




**Performance Data**  
 Execution Time : 5.94429

Method	Count	Duration (sec)
GetField	800	1.20298 ■
Load	50	0.08225
LoadDirect	2	0.00446
GetContentFields	1	0.00122
GetPanels	1	0.00013
Other Non-API	0	4.65325 ■■■■■

[View More Details](#)

Note the high GetField count. This can result in high processing time.

**Overview**

**Information**   

[Preview](#) | [Edit](#) | [Form](#) | [Refresh](#)

Label : GetContentGood  
 Asset Id : 23063  
 Type : File  
 Last Modified By : Dae Kim  
 Last Modified On : 10/23/2014 2:43:24 AM

Status : N/A  
 Status Modified By : Dae Kim  
 Status Modified On : 10/23/2014 2:43:10 AM

Template : /System/Templates/\_Partner  
 Summit/GetContentGood  
 Template Id : 22971

**Performance Data**  
 Execution Time : 1.99771

Method	Count	Duration (sec)
LoadDirect	102	0.19663 ■
GetField	100	0.17341 ■
Load	50	0.08349
GetContent	50	0.06711
GetContentFields	1	0.00522
GetPanels	1	0.00020
Other Non-API	0	1.47165 ■■■■■

[View More Details](#)

The GetField count is reduced considerably, improving performance