

SAP iView Integration with Microsoft Office SharePoint 2007

Authors

Altaf Gilani, Consultant, MCS, Microsoft Corporation,
agilani@microsoft.com

Juergen Daiberl, Snr. Technical Evangelist, D&PE, Microsoft Corporation,
jdaiberl@microsoft.com

Summary

This paper describes how to integrate iViews from a SAP NetWeaver Portal in Microsoft® Office SharePoint® Server 2007. It is written for developers, technical consultants, and solution architects. This paper introduces the technologies and architecture used, and provides a detailed technical walkthrough of the implementation process. This paper also contains links to Microsoft® Office SharePoint® Server 2007 advanced features and technical topics including Single Sign-On (SSO).

Copyright

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred.

© 2007 Microsoft Corporation. All rights reserved.

Microsoft, Windows, Windows Server, Microsoft Office, Microsoft Office SharePoint Server 2007, SharePoint, Visual Studio, Visual Studio 2005 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Applies To

- Microsoft® Office SharePoint® Server 2007
- SAP NetWeaver
- SAP NetWeaver Portal
- SAP Business Suite (mySAP ERP)

Keywords

SAP NetWeaver, SharePoint 2007, Visual Studio 2005, SAP iView, iView Web Part

Contact

This document is provided to you by Microsoft Corporation. Please check the SAP interoperability area at www.microsoft.com/sap and the .NET interoperability area in the SAP Developer Network at <http://sdn.sap.com> for any updates or further information.

Contents

Introduction	1
Problem Description	1
Object Tag Page Viewer Web Part	1
Development	2
Deployment and Configuration	4
Deployment of Object Tag Page Viewer Web Part.....	4
Configuring the Object Tag Page Viewer web part.....	5
Security Configuration	7
Single Sign-On	8
Source Code	8
ObjectTagPageViewer.cs	8
ObjectTagPageViewer.webpart	10
ObjectTagPageViewer.xml	10
References	11
Acknowledgement	11

Introduction

There are scenarios where an end user would require logging in to SAP GUI or SAP Portal to perform some of their commonly performed operations. It would be convenient if these operations were performed from a portal like Microsoft Office SharePoint Server Portal (MOSS) which an end user might be using more frequently. In such scenarios the best point of integration is to display SAP iViews or Web Dynpro iViews in MOSS. MOSS out of the box provides two web parts that can be used for displaying pages from portals/web sites outside MOSS. These are Page Viewer Web Part and iView Web Part. When configuring SAP iView or Web Dynpro iViews, these web parts can go in to an infinite refresh loop with some iViews. In order to overcome this issue, there is a need to develop a custom web part that would display the SAP iView. The Object Tag Page Viewer Web Part is an approach that allows MOSS portals to display these iViews.

Problem Description

MOSS out of the box provides two web parts that can be leveraged for displaying SAP iViews. These are:

1. Page Viewer Web Part

The Page Viewer Web Part allows displaying a web page in MOSS. To display an iView the complete URL of the iView should be specified. Page Viewer Web Part doesn't provide integration with Single Sign-on (SSO), hence authentication of the iView should be handled outside of the web part. For SSO the SPNego Login Module from SAP could be used, for more information about this login module see <http://download.microsoft.com/download/c/6/c/c6c42b9f-66f4-47b3-99be-8e5afa1ddc9a/SSO%20with%20MS%20and%20SAP.pdf>.

2. iView Web Part

The iView Web Part is specifically designed for SAP iViews. There is a two steps process to display an iView:

- a. Configure SharePoint SSO for SAP Logon.
- b. Specify SSO and iView URL in the iView Web part.

Both, the iView Web Part and the Page Viewer Web Parts use an IFrame to display the web page/iView. When displaying SAP iViews, both Page Viewer and iView Web part can go into an infinite loop. The behavior of infinite loop can be seen in the screenshot of an IFrame in a plain HTML page.

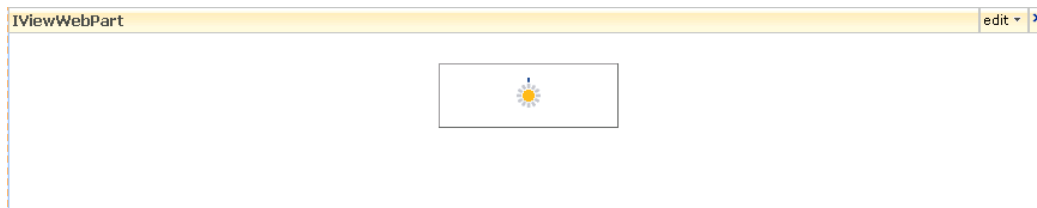


Figure: Infinite Loop in an iView Web part

Object Tag Page Viewer Web Part

A HTML Object Tag can be used as an alternative to an IFrame for displaying Web Pages. A SAP iView can be successfully displayed in such an object tag. This whitepaper shows the development of such a custom Web Part using an Object Tag for displaying SAP iViews in SharePoint.

SAP iView Integration with Microsoft Office SharePoint 2007

Here is an example of using an Object tag to load a web page:

```
<object id="foo" name="foo" type="text/html" data=http://www.live.com"/>
```

data: Defines a URL that refers to the object data.

type: Defines the MIME type of data specified in the data attribute.

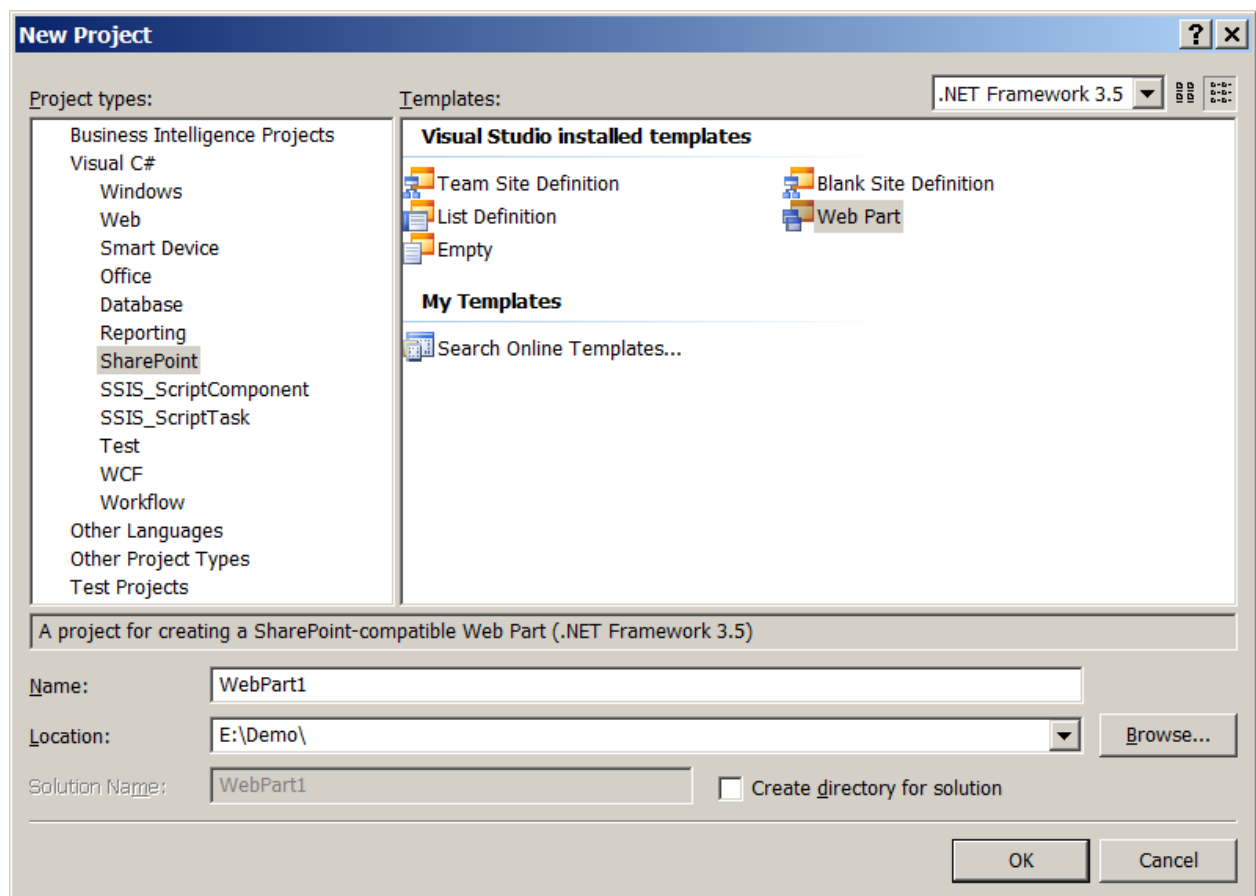
Development

This chapter guides you through the development of the Object Tag Page Viewer Web Part. It's a step-by-step guide developing the Web Part in Visual Studio 2008. It's necessary that you have the Visual Studio Extensions 2008 for Windows SharePoint Services version 1.2 installed. You can find the Visual Studio Extensions 2008 for Windows SharePoint Services at

<http://www.microsoft.com/downloads/details.aspx?familyid=7BF65B28-06E2-4E87-9BAD-086E32185E68&displaylang=en>.

Follow the steps to develop the web part:

1. Create a new Project in Visual Studio 2008 (Project Type -> SharePoint, Template -> Web Part)



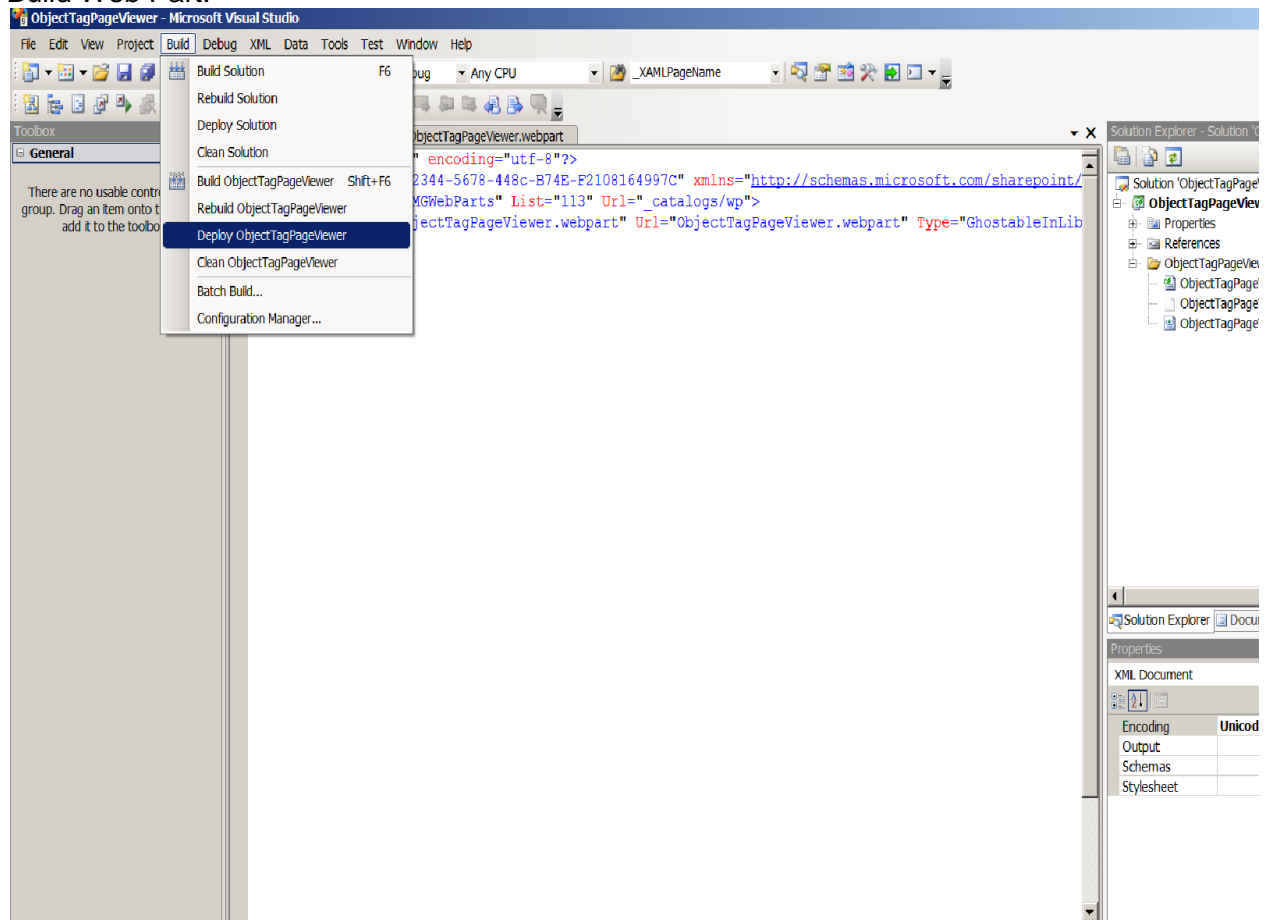
2. Override the Render method with the following code:

```
protected override void Render(HtmlTextWriter output)
{
    if (String.Compare(pageUrl, "http://") == 0 ||
```

SAP iView Integration with Microsoft Office SharePoint 2007

```
String.IsNullOrEmpty(pageUrl))
{
    output.WriteBreak();
    output.WriteLine(@"To link to content,
        <A ID='ObjectTagPageViewer_' + base.ID + ""
        HREF=javascript:MSOT1Pn_ShowToolPane2Wrapper('Edit','129','"
        + base.ID + "");>open the tool pane</A> and then type a URL
        in the <b> Page Url </b> text box under <b> Custom Property.
        </b>");
    output.WriteBreak();
}
else
{
    if (Uri.IsWellFormedUriString(pageUrl, UriKind.Absolute))
    {
        string objectTagID = "outputObject_" + base.UniqueID;
        output.AddAttribute(HtmlTextWriterAttribute.Id,
            objectTagID, false);
        output.AddAttribute(HtmlTextWriterAttribute.Name,
            objectTagID, false);
        output.AddAttribute(HtmlTextWriterAttribute.Width,
            "100%", false);
        output.AddAttribute(HtmlTextWriterAttribute.Height,
            "100%", false);
        output.AddAttribute(HtmlTextWriterAttribute.Type,
            "text/html", false);
        output.AddAttribute("data", pageUrl, false);
        output.RenderBeginTag(HtmlTextWriterTag.Object);
        output.RenderEndTag();
    }
    else
    {
        output.WriteBreak();
        output.WriteLine("Please enter a valid Absolute <b> Page Url
            </b>");
        output.WriteBreak();
    }
}
}
```

3. Build Web Part.



Deployment and Configuration

Deployment of Object Tag Page Viewer Web Part

On deploying the Web Part project with Visual Studio Extensions 2008 for Windows SharePoint Services version 1.2, it generates the following files in the corresponding Debug or Release folders:

1. Setup.bat
2. <WebPartName>.wsp, in this example it is ObjectTagPageViewer.wsp.

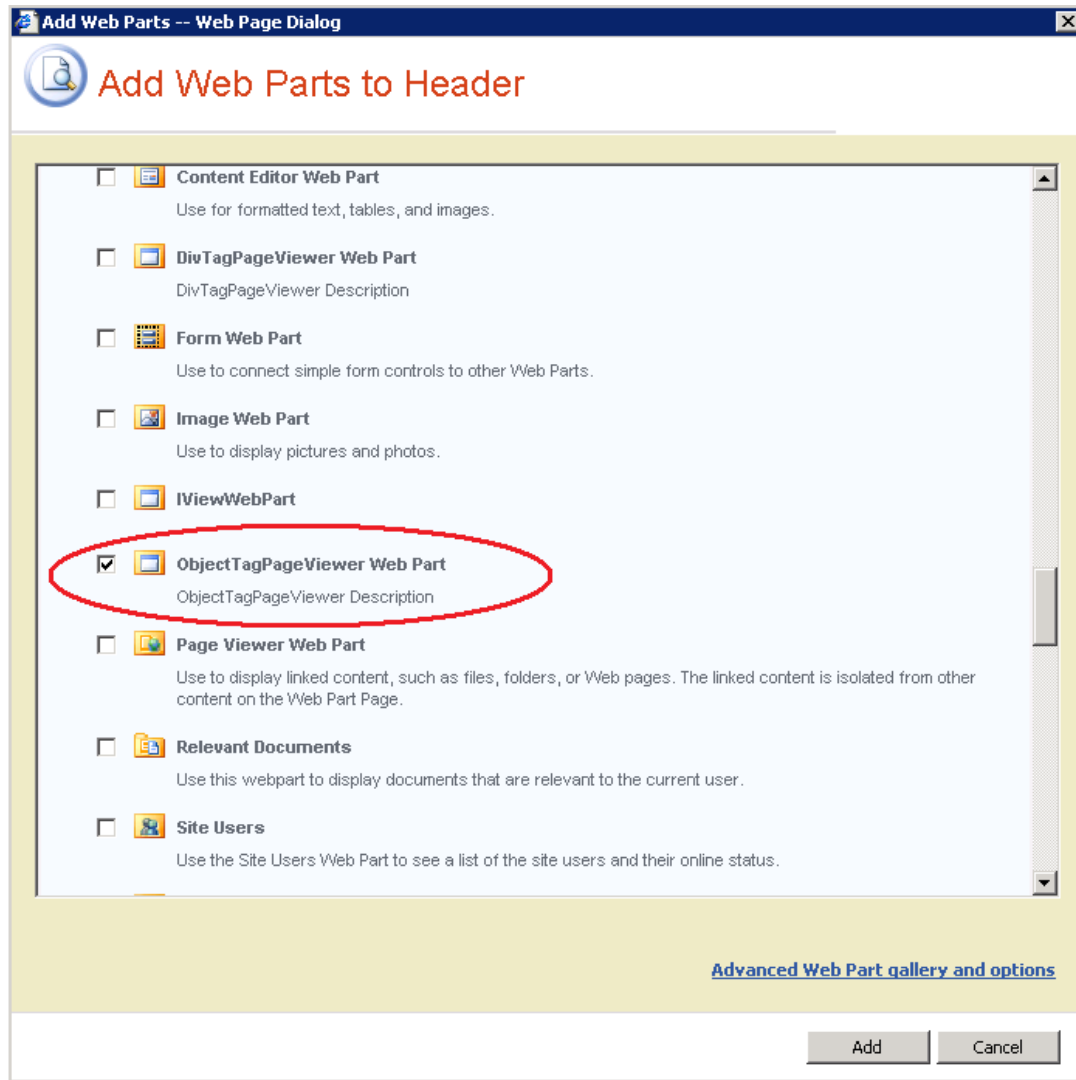
Following are the steps to install the web part on to a SharePoint Server:

1. Copy the two files on to the Server. For e.g. C:\Release.
2. Open a command prompt with administrative privileges.
3. Run Setup.bat from the folder create in step 1.
For e.g. setup /install /siteurl <site url for SharePoint> /weburl <web url for SharePoint>

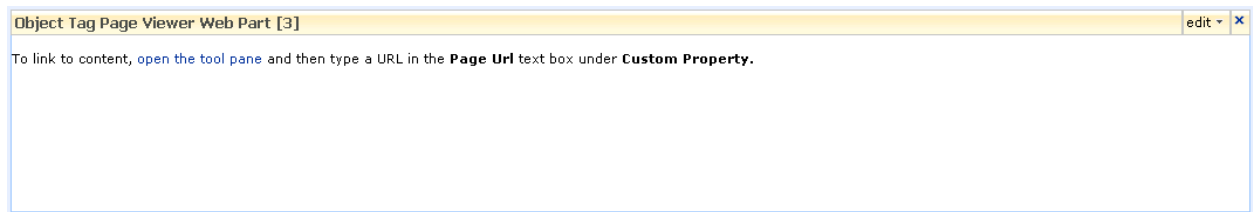
Configuring the Object Tag Page Viewer web part

The following are the steps to configure the Object Tag Page Viewer Web Part in SharePoint:

1. Edit a SharePoint page and click Add a Web Part. As shown in the figure below, select the Object Tag Page Viewer Web Part from the Add Web Parts dialog box.



2. Click on “open the tool page” or click Edit and select “Modify SharePoint Web Part”.



SAP iView Integration with Microsoft Office SharePoint 2007

3. Set the Page URL in the Custom Property area to the URL of the SAP iView. Note that the URL should be the absolute URL. The URL of an iView can be retrieved from the SAP NetWeaver Portal. Simply open an iView inside the SAP Portal in preview mode and you can copy the URL from the address bar of your browser.

Object Tag Page Viewer Web x

Appearance

Title
Object Tag Page Viewer Web

Height
Should the Web Part have a fixed height?
 Yes 550 Pixels
 No. Adjust height to fit zone.

Width
Should the Web Part have a fixed width?
 Yes
 No. Adjust width to fit zone.

Chrome State
 Minimized
 Normal

Chrome Type
Default

Layout
 Advanced

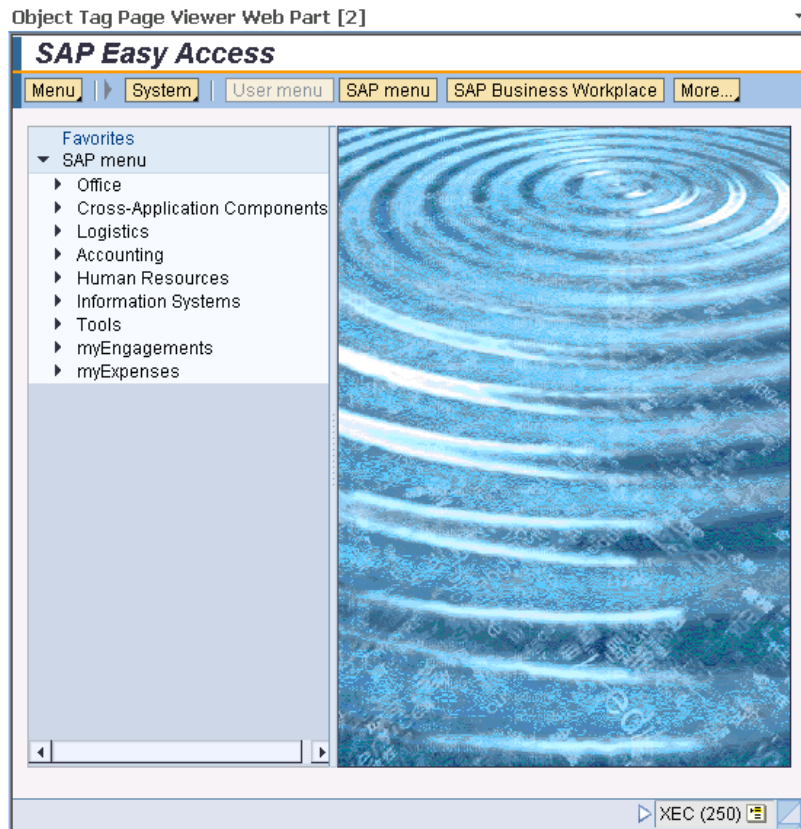
Custom Property

Page Url
http://

OK Cancel Apply

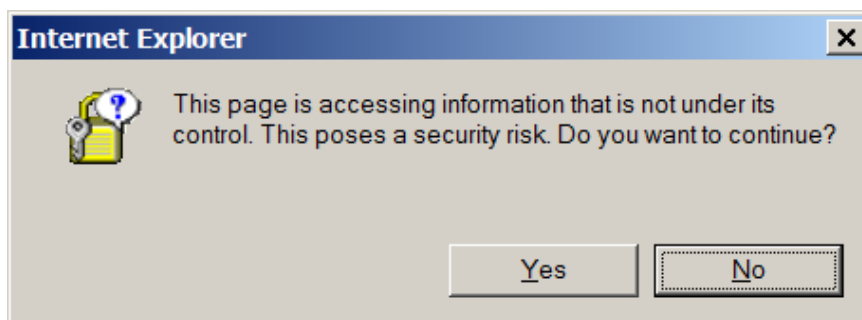
SAP iView Integration with Microsoft Office SharePoint 2007

As a result the Object Tag Page Viewer Web Part will display the iView specified in the URL.



Security Configuration

Once Object Tag Page Viewer Web Part is configured, on displaying the Web Part in Internet Explorer it would give the following security warning to the user:



The following actions would avoid the security warning:

1. Make sure the domain name of the SharePoint site and the SAP Portal URL are the same, for example <http://sharepoint.site.domain.com> and <http://sapportal.site.domain.com>

2. Add the SharePoint URL to the trusted site in Internet Explorer (Tools -> Internet Options -> Choose Security Tab -> Click on Trusted Sites -> Click on the Sites button and add the URL of the SharePoint site).

Single Sign-On

There are a variety of single sign-on options as it relates to HTTP traffic. Internet Explorer will automatically transmit windows/network credentials to intranet sites so that your Active Directory sign-on will also authenticate you to web sites. SAP can leverage these credentials via a variety of mechanisms. It can also leverage X.509 certificate authentication if your company has a PKI in place.

The other alternative is the use of the SAP Logon Tickets. The SAP ABAP stack cannot issue SAP Logon Tickets—it is only capable of receiving them, for issuing the logon tickets the SAP Java stack is necessary. You will find more details about single sign-on in a Microsoft – SAP landscape at

<http://download.microsoft.com/download/c/6/c/c6c42b9f-66f4-47b3-99be-8e5afa1ddc9a/SSO%20with%20MS%20and%20SAP.pdf>.

Source Code

ObjectTagPageViewer.cs

```
using System;
using System.Runtime.InteropServices;
using System.Web.UI;
using System.Web.UI.WebControls.WebParts;
using System.ComponentModel;

using Microsoft.SharePoint.WebPartPages;

namespace Microsoft
{
    public class ObjectTagPageViewer :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        const int heightWidth = 150;
        #region Properties

        /// <summary>
        /// The URL to display in the Object Tag
        /// </summary>
        private string pageUrl = "http://";
        /// <summary>
        /// The property that exposes the pageUrl variable
        /// </summary>
        [Personalizable(),
        WebBrowsable(true),
        Browsable(true),
        Category("Custom Property"),
        DefaultValue("http://"),
        WebPartStorage(Storage.Personal),
```

SAP iView Integration with Microsoft Office SharePoint 2007

```
FriendlyName("Page Url"),
Description("Default page URL such as http://www.live.com")]
public string PageUrl
{
    get { return pageUrl; }
    set { pageUrl = value; }
}
#endregion

protected override void OnPreRender(EventArgs e)
{
    if (base.Height.IsEmpty)
    {
        base.Height = heightWidth;
    }

    base.OnPreRender(e);
}

protected override void Render(HtmlTextWriter output)
{
    if (String.Compare(pageUrl, "http://") == 0 ||
String.IsNullOrEmpty(pageUrl))
    {
        output.WriteBreak();
        output.WriteLine(@"To link to content, <A ID=
'ObjectTagPageViewer_' + base.ID + "'
HREF=javascript:MSOTlPn_ShowToolPane2Wrapper('Edit','1
29','" + base.ID + "')>open the tool pane</A> and
then type a URL in the <b> Page Url </b> text box
under <b> Custom Property. </b>");
        output.WriteBreak();
    }
    else
    {
        if (Uri.IsWellFormedUriString(pageUrl,
UriKind.Absolute))
        {
            string objectTagID = "outputObject_" +
                base.UniqueID;
            output.AddAttribute(HtmlTextWriterAttribute.Id,
                objectTagID, false);
            output.AddAttribute(HtmlTextWriterAttribute.Name,
                objectTagID, false);
            output.AddAttribute(HtmlTextWriterAttribute.Width,
                "100%", false);
            output.AddAttribute(HtmlTextWriterAttribute.Height,
                "100%", false);
            output.AddAttribute(HtmlTextWriterAttribute.Type,
                "text/html", false);
            output.AddAttribute("data", pageUrl, false);
            output.RenderBeginTag(HtmlTextWriterTag.Object);
            output.RenderEndTag();
        }
        else
        {

```

SAP iView Integration with Microsoft Office SharePoint 2007

```
        output.WriteLine("Please enter a valid Absolute  
            <b> Page Url </b>");  
        output.WriteLine();  
    }  
}  
}
```

ObjectTagPageViewer.webpart

```
<?xml version="1.0" encoding="utf-8"?>  
<webParts>  
  <webPart xmlns="http://schemas.microsoft.com/WebPart/v3">  
    <metaData>  
      <!--  
        The following Guid is used as a reference to the web part class,  
        and it will be automatically replaced with actual type name at  
        deployment time.  
      -->  
      <type name="89012344-5678-448c-B74E-F2108164997C" />  
      <importErrorMessage>  
        Cannot import ObjectTagPageViewer Web Part.  
      </importErrorMessage>  
    </metaData>  
    <data>  
      <properties>  
        <property name="Title" type="string">  
          Object Tag Page Viewer Web Part  
        </property>  
        <property name="Description" type="string">  
          Object Tag Page Viewer Web Part displays a web page  
          like the Page Viewer Web Part. The difference being,  
          it uses an Object tag instead of IFrame. Use this Web  
          Part when you cannot render a page in an IFrame. For  
          example some the SAP iView cannot be rendered in a  
          IFrame, this web part is useful in such scenarios.  
        </property>  
      </properties>  
    </data>  
  </webPart>  
</webParts>
```

ObjectTagPageViewer.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<Elements Id="89012344-5678-448c-B74E-F2108164997C"  
  xmlns="http://schemas.microsoft.com/sharepoint/" >  
  <Module Name="MSWebParts"  
    List="113"  
    Url="_catalogs/wp">  
    <File Path="ObjectTagPageViewer.webpart"  
      Url="ObjectTagPageViewer.webpart"  
      Type="GhostableInLibrary" />  
    </File>  
  </Module>  
</Elements>
```

</Module>
</Elements>

References

1. http://www.w3schools.com/tags/tag_object.asp
2. <http://www.microsoft.com/downloads/details.aspx?familyid=7BF65B28-06E2-4E87-9BAD-086E32185E68&displaylang=en>

Acknowledgement

Thanks to Michael Sellman, Ramesh Raman Nair, James Hornstein and Steve Case.