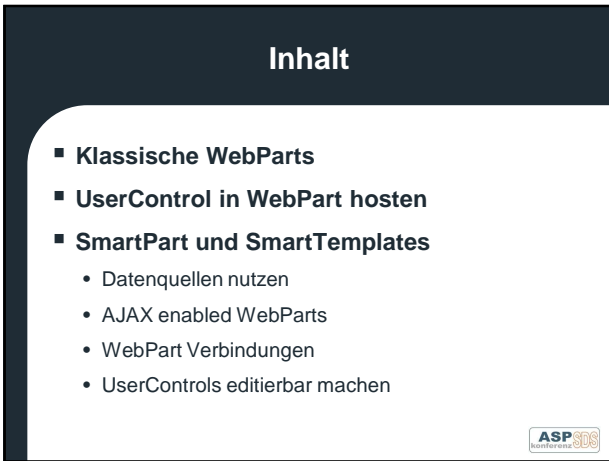




ASP.NET UserControl wird WebPart
Robert Vogel
robertv@ppedv.de

Sponsoren: SharePoint Camp, besser burghausen
Mediapartner: ASP.NET professional, VISUAL STUDIO
Veranstalter: ppedv

Notizen

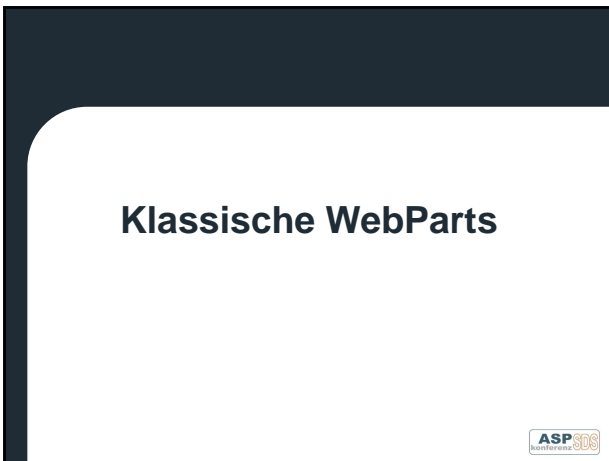


Inhalt

- Klassische WebParts
- UserControl in WebPart hosten
- SmartPart und SmartTemplates
 - Datenquellen nutzen
 - AJAX enabled WebParts
 - WebPart Verbindungen
 - UserControls editierbar machen

ASP konferenz SDS

Notizen



Klassische WebParts

ASP konferenz SDS

Notizen

Klassische WebParts

- Eigenes Item/Project-Template in VSeWSS
- Ist ein CustomControl
- Einfache Lösung um eigene Funktionalität in den SharePoint einzubringen
- Möglichkeit zum Datenaustausch zwischen WebParts
- Möglichkeiten der Konfigurierbarkeit



Notizen

Klassische WebParts

Vorteile:

- Item/Project-Template in VSeWSS
- Einfaches / automatisches Deployment


Nachteile:

- Nur Code-Ansicht, kein Designer
- Gestaltung der Oberfläche umständlich



Notizen

UserControl in WebPart hosten



Notizen

UserControl in WebPart hosten

Code im WebPart:

```
protected override void CreateChildControls()
{
    base.CreateChildControls();
    this.Controls.Clear();
    this.Controls.Add(
        (UserControl)this.Page.LoadControl(
            "path/to/usercontrol.ascx"));
}
```



Notizen

UserControl in WebPart hosten

Vorteile:

- Visueller Designer
- Deklaratives Einbinden und Konfigurieren von Datenquellen


Nachteile:

- Keine Item-/Project-Templates
- Konfigurierbarkeit und Datenaustausch schwierig



Notizen

SmartPart und Smart Templates



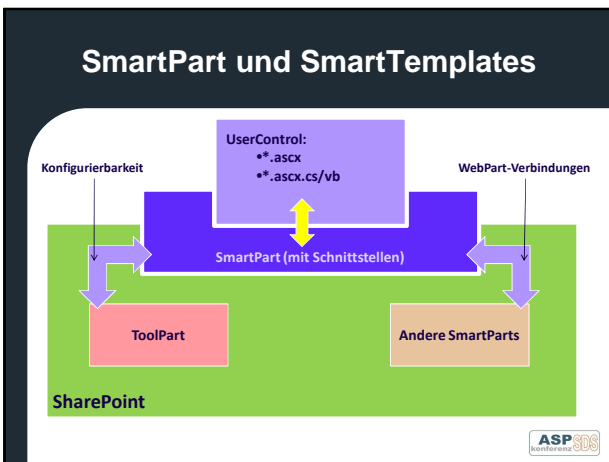
Notizen

SmartPart und SmartTemplates

- Community Projekt (smartpart.codeplex.com, smarttemplates.codeplex.com)
- Vorgefertigtes WebPart zum Hosten von UserControls
- Template zum Erstellen eines UserControls mit automatischer Konfiguration des SmartParts



Notizen



Notizen

SmartPart und SmartTemplates

Datenquellen nutzen

- Deklaratives Einbinden einer Datenquelle
- Gegebenenfalls anpassen der Web.Config oder Einbetten des ConnectionStrings in das UserControls selbst
- Zugriff auf SharePoint Objektmodell/Kontext → Einbinden der **Microsoft.SharePoint.dll**




Notizen

SmartPart und SmartTemplates

AJAX enabled WebParts

- MOSS für AJAX Steuerelemente vorbereiten
 - .Net Framework 3.5 (SP1)
 - Web.Config der entsprechenden Webanwendung anpassen
- Gegebenenfalls ScriptManager auf MasterPage bereitstellen



Notizen


SmartPart und SmartTemplates

AJAX enabled WebParts

- Gegebenenfalls SharePoint „From OnSubmit Wrapper“ deaktivieren

```
<script type='text/javascript'>  
  _spOriginalFormAction = document.forms[0].action;  
  _spSuppressFormOnSubmitWrapper = true;  
</script>
```

- Steuerelemente platzieren und nutzen



Notizen

SmartPart und SmartTemplates

UserControls editierbar machen

- Attribuierung von Properties
- ToolParts




Notizen

SmartPart und SmartTemplates

UserControls editierbar machen: Attributierung von Properties

```
[Personalizable(PersonalizationScope.Shared),  
WebBrowsable(true),  
WebDisplayName("Eigenschaft"),  
WebDescription("Diese Eigenschaft bearbeiten")]  
public String MyProperty { get; set; }
```




Notizen

SmartPart und SmartTemplates

UserControls editierbar machen: ToolParts

- Im Toopart:
 - Ableiten von `Microsoft.SharePoint.WebPartPages.Toolpart`
 - Überschreiben der Methoden
 - > ApplyChanges
 - > SyncChanges
 - > CreateChildControls



Notizen


SmartPart und SmartTemplates

UserControls editierbar machen: ToolParts

- Auf das SmartPart zugreifen:

```
SmartPart hostWebPart =  
    (SmartPart)this.ParentToolPane.SelectedWebPart;  
SmartPart hostWebPart = (SmartPart)this.WebPartToEdit;
```
- Eigenschaften auslesen und verändern

```
hostWebPart.SetCustomProperty("PropertyName", PropertyValue);  
hostWebPart.RefreshPropertyValues();  
hostWebPart.SaveWebPartProperties = true;  
hostWebPart.GetCustomProperty("PropertyName")
```




Notizen

SmartPart und SmartTemplates

UserControls editierbar machen: ToolParts

- Im UserControl:
 - Einbinden der **ReturnOfSmartPart.dll**
 - Implementieren von **SmartPart.IAdvancedUserControl**
 - > GetCustomToolParts
 - > SetContext



Notizen


SmartPart und SmartTemplates

UserControls editierbar machen: ToolParts

```
ToolPart[] IAdvancedUserControl.GetCustomToolParts(){
    List<ToolPart> toolParts = new List<ToolPart>();
    toolParts.Add(new MeinToolPart());
    return toolParts.ToArray();
}

void IAdvancedUserControl.SetContext(SmartPart webpart)
{
    this.HostWebPart = webpart;
}

this.HostWebPart = (SmartPart.SmartPart)this.Parent;
```



Notizen

SmartPart und SmartTemplates

WebPart Verbindungen

- Einbinden der **ReturnOfSmartPart.dll**
- **SmartPart.IConnectionConsumerControl**
- **SmartPart.IConnectionProviderControl**
- Im **Edit-Mode** der WebPart-Seite die Verbindung herstellen
- Auch via AJAX möglich:
SmartPart.AJAXConnectionData



Notizen

SmartPart und SmartTemplates

```
#region IConnectionProviderControl Members
public object GetProviderData()
{
    return tbEingabe.Text;
}

public string ProviderMenuLabel
{
    get { return "Sendet Text an"; }
}
#endregion
```



Notizen

SmartPart und SmartTemplates

```
#region IConnectionConsumerControl Members
public void SetConsumerData(object data)
{
    if (data.GetType() == typeof(string))
    {
        tbAusgabe.Text = data.ToString();
    }
}

public string ConsumerMenuLabel
{
    get { return "Empfängt Text von"; }
}
#endregion
```



Notizen

SmartPart und SmartTemplates

WebPart Verbindungen: AJAX

- Helferklasse
`SmartPart.AJAXConnectionData(object data, Control triggerControl, string triggerEvent)`
- Mehrere TriggerControls
`AJAXConnectionData.TriggerControls`



Notizen

SmartPart und SmartTemplates

```
#region IConnectionProviderControl Members
public object GetProviderData()
{
    return new SmartPart.AJAXConnectionData(
        this.tbEingabe.Text, this.btnOk, "Click");
}

public string ProviderMenuLabel
{
    get { return "Sendet Text an"; }
}
}
#endregion
```



Notizen

SmartPart und SmartTemplates

```
#region IConnectionConsumerControl Members
public void SetConsumerData(object data){
    if(data != null){
        if(data.GetType() == typeof(AJAXConnectionData)){
            AJAXConnectionData ajaxData =
                (AJAXConnectionData)data;
            ajaxData.RegisterTriggerControls(this.updtPanel);
            this.tbEingabe.Text = ajaxData.Data.ToString();
        }
    }
}

public string ConsumerMenuLabel{...}
}
#endregion
```



Notizen

FRAGEN ?




Notizen

Hat Ihnen mein Vortrag gefallen?



Ich freue mich auf Ihr Feedback!



Notizen

Wir sehen uns wieder:

-  **In 5 Tagen zum SharePoint Profi!**
20.-24. Juli 2009, München
www.SharePointCamp.de
-  **Schneller zum .NET 3.5 Developer**
21.-26. Sept. 2009, Burghausen
www.DevTrain.de/Camp
-  **07.-08. Okt. 2009, Bonn**
www.ADC09.de



Notizen



VIELEN DANK !

Robert Vogel

Notizen
