



Microsoft Partner
Learning

SharePoint

Designer

2013

IHRE KURSUNTERLAGE



Inhalt

Konfiguration der Anwendungseinstellungen für den SharePoint Designer.....	2
Eigene Liste mit Spaltengültigkeitsprüfung erstellen und als Vorlage exportieren	5
Verbinden zweier Listen mittels Nachschlage-Feld.....	8
Anzeigeformular für Elemente um Informationen aus verbundener Liste erweitern	9
WebPart-Page mit zwei verbundenen Listen	13
Benutzerdefiniertes Formular über eigenen Ribbon-Eintrag öffnen (Custom Action)	16
Benutzerdefinierten Inhaltstyp mit Dokumentvorlage erstellen	20
Externen Inhaltstyp anlegen (Business Data Connectivity Service).....	25
Linked Data Source für aggregierte Datenansicht.....	31
REST Webservice anbinden – Erstellung eines Wetter-WebParts	36
XML-Datei anbinden	42
Listenworkflow Urlaubsantrag mit Stages und AppStep	47
Erweiterung des Listenworkflow durch Implementierung eines AppStep.....	51
Websiteworkflow mit Loop, Webservice-Call und Dictionary Variable	52

Konfiguration der Anwendungseinstellungen für den SharePoint Designer

Der SharePoint Designer 2013 kann entsprechend regulatorischer Vorgaben bzw.

Unternehmensrichtlinien in der Zentraladministration in seinem Funktionsumfang eingeschränkt oder komplett deaktiviert werden.

- In der Zentraladministration findet sich unter dem Punkt „General Application Settings“ der Eintrag für den SharePoint Designer

The screenshot shows the SharePoint General Application Settings page. On the left, there is a navigation menu with items like Central Administration, Application Management, System Settings, Monitoring, Backup and Restore, Security, Upgrade and Migration, General Application Settings (which is selected), Apps, and Configuration Wizards. On the right, there are several service links: External Service Connections, InfoPath Forms Services, SharePoint Designer (which is highlighted with a red box), Search, Content Deployment, and SQL Server Reporting Services. The SharePoint Designer link leads to the configuration page shown below.

- Nach einem Klick auf „Configure SharePoint Designer settings“ gelangt man auf die Konfigurationsseite für den SharePoint Designer, die folgende fünf Optionen bietet:

The screenshot shows the SharePoint Designer Settings configuration page. It has a sidebar with the same navigation menu as the previous page. The main area contains five configuration sections: 1. Web Application: A dropdown menu set to "http://dc/". 2. Enable SharePoint Designer: A checked checkbox. 3. Enable Detaching Pages from the Site Definition: A checked checkbox. 4. Enable Customizing Master Pages and Layout Pages: A checked checkbox. 5. Enable Managing of the Web Site URL Structure: A checked checkbox. Each section has a brief description below it.

○ Web Application

Hier ist die betroffene Webanwendung auszuwählen. Dies ist nur interessant, wenn mehrere Webanwendungen angelegt wurden.

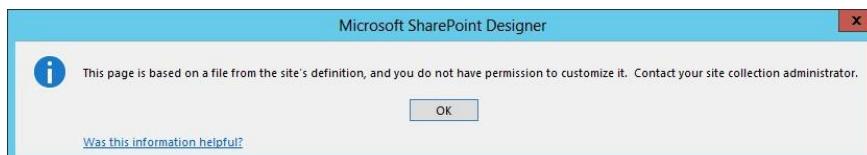
- Allow SharePoint Designer to be used in this Web Application

Hier kann die Nutzung des Designers komplett deaktiviert werden. Wenn dies der Fall ist, bekommen Benutzer, die versuchen, eine Website mit dem Designer zu öffnen, folgende Fehlermeldung angezeigt:

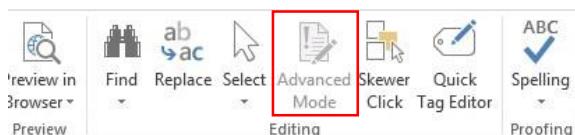


- Allow Site Collection Administrators to Detach Pages from the Site Template Hier kann das Trennen der Gestaltungsvorlage (standardmäßig die seattle.master) von der Website unterbunden werden. Möchte der Websitesammlungsadministrator bspw. einer Website eine benutzerdefinierte Systemgestaltungsvorlage zuweisen

(Kontextmenüeintrag „Set as Custom Master Page“), erscheint folgende Meldung:



Weiterhin wird durch das Deaktivieren dieser Checkbox auch der „Advanced Mode“ (Erweiterter Bearbeitungsmodus) abgeschaltet und ist im Ribbon nicht mehr anklickbar. Dann können die gelblich hinterlegten strukturellen Bereiche einer Page, die standardmäßig gesperrt sind, nicht mehr zur Bearbeitung freigegeben werden.

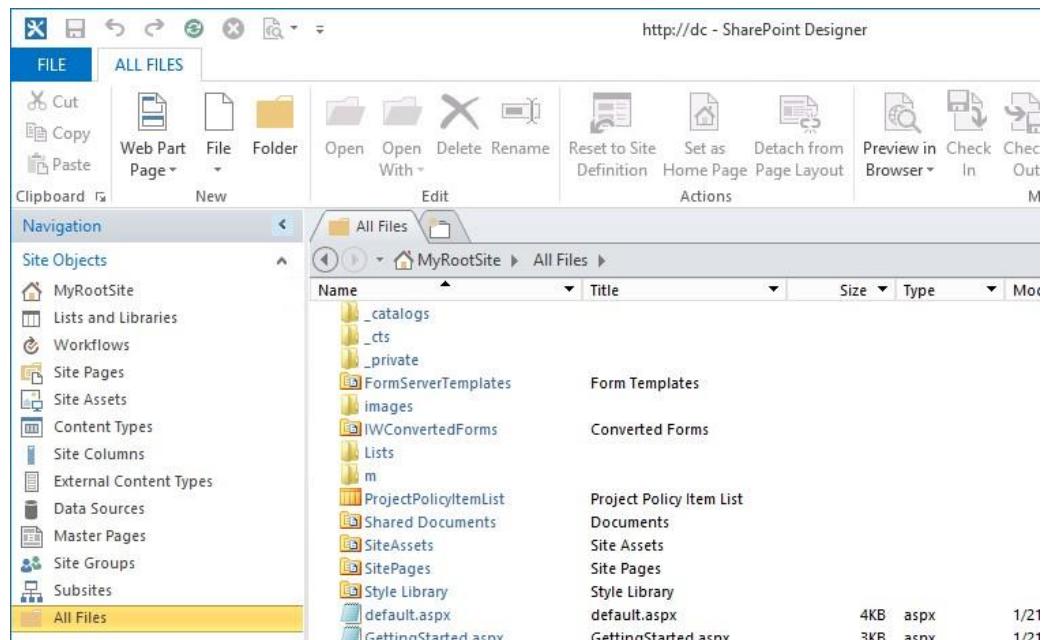


- Allow Site Collection Administrators to Customize Master Pages and Layout Pages Diese aktivierte Option erlaubt dem Websitesammlungsadministrator das Verändern von Gestaltungsvorlagen und Layoutseiten. Ist diese Option deaktiviert, fehlt im Navigationsbereich des Designers der Eintrag „Master Pages“.

Navigation	Navigation
<p>Site Objects</p> <ul style="list-style-type: none"> MyRootSite Lists and Libraries Workflows Site Pages Site Assets Content Types Site Columns External Content Types Data Sources Master Pages Site Groups Subsites All Files 	<p>Site Objects</p> <ul style="list-style-type: none"> MyRootSite Lists and Libraries Workflows Site Pages Site Assets Content Types Site Columns External Content Types Data Sources Site Groups Subsites All Files
bei aktivierter Option	bei deaktivierter Option

- Allow Site Collection Administrators to see the URL Structure of their Web Site

Ist diese Checkbox angehakt, finden Websitesammlungsadministratoren im Navigationsbereich des Designers den Eintrag „All Files“, welcher auf Dateisystemebene den Zugriff auf die Struktur der Websitesammlung und allen entsprechenden (Unter)Ordnern erlaubt. Ist diese Option deaktiviert, fehlt dieser Eintrag im Navigationsbereich.



□ Weiterhin können diese Einstellungen auch auf Websitesammlungsebene vorgenommen werden.

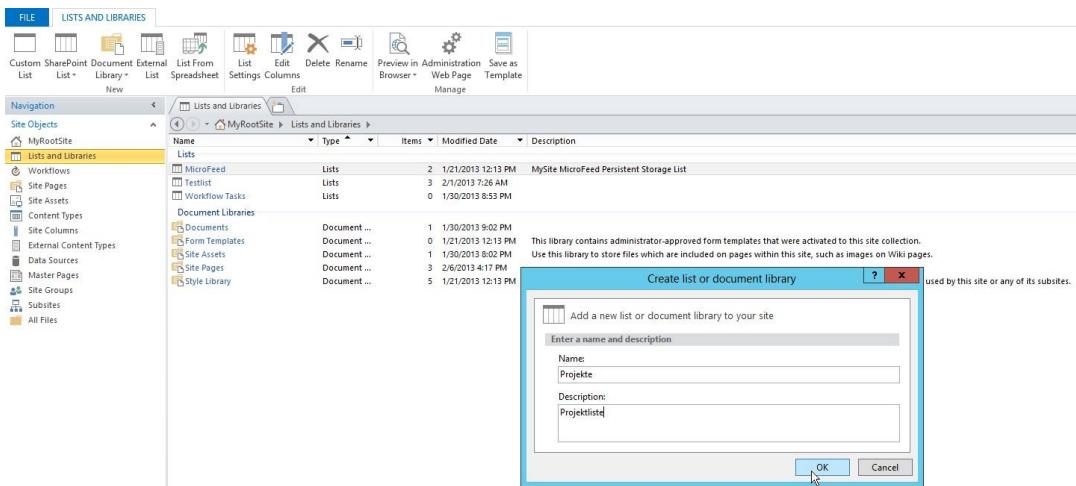
Dazu befindet sich ein entsprechender Eintrag in den „Site Collection Settings“ auf der entsprechenden Websitesammlung. Dabei ist zu beachten, dass die Einstellungen der Zentraladministration auf Webanwendungsebene denen der Websitesammlung übergeordnet sind. So kann ein Websitesammlungsadministrator bei restriktiven Einstellungen seitens der ZA keine Lockerung der Berechtigungen auf seiner Site Collection bewirken, wie die folgende Grafik zeigt:

Setting	Status	Reason
Allow Site Owners and Designers to use SharePoint Designer in this Site Collection	<input checked="" type="checkbox"/>	Enabled
Allow Site Owners and Designers to Detach Pages from the Site Definition	<input type="checkbox"/>	This option was disabled by your server administrator
Allow Site Owners and Designers to Customize Master Pages and Page Layouts	<input type="checkbox"/>	This option was disabled by your server administrator
Allow Site Owners and Designers to See the Hidden URL structure of their Web Site	<input type="checkbox"/>	This option was disabled by your server administrator

Eigene Liste mit Spaltengültigkeitsprüfung erstellen und als Vorlage exportieren

Mit wenigen Klicks lässt sich eine benutzerdefinierte SharePoint Liste (Custom List) samt Spalten anlegen. Die Spalten können auch mit Gültigkeitsprüfungen belegt werden, um valide Eingaben sicher zu stellen. Anschließend kann diese als Listenvorlage zur Verfügung gestellt werden.

- Zunächst die entsprechende Website im Designer öffnen. Im Navigationsmenü befindet sich der Eintrag „Lists and Libraries“. Dort im Ribbon auf den ersten Eintrag „Custom List“ klicken, um eine neue Benutzerdefinierte Liste zu erstellen. Im folgenden Dialogfeld dann den Namen und ggf. eine Beschreibung für die Liste eintragen.



- Nachdem die Liste erstellt wurde, taucht diese im Punkt „Lists and Libraries“ mit auf. Nach einem Klick auf die entsprechende Liste öffnet sich die Listenübersichtsseite (Bild links). Ein Klick auf „Edit List Columns“ im Bereich „Customization“ öffnet die Übersicht der Listenspalten. Dort kann über den Ribboneintrag „Add New Column“ eine Listenspalte erstellt werden (Bild rechts).

Anmerkung: es besteht auch die Möglichkeit, eine vorhandene Websitespalte an die Liste zu binden.

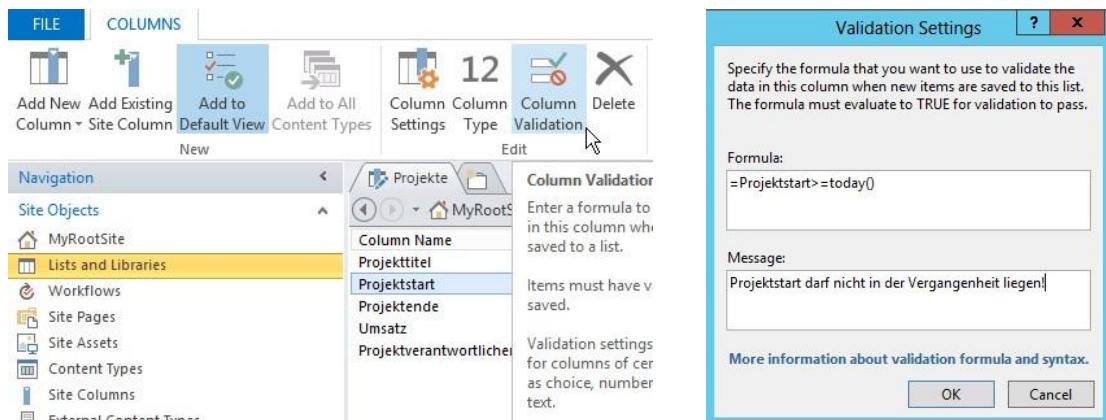
Unterschied: die Listenspalte wird im Kontext der Liste erstellt, die Websitespalte hingegen „global“ für die entsprechende Website zur Verfügung gestellt. Die Websitespalte ist somit für andere Listen wiederverwendbar, während die Listenspalte fest an eine bestimmte Liste gebunden ist.

- Es besteht die Möglichkeit, Spaltengültigkeitsprüfungen zu implementieren, um die Eingaben der User beim Ausfüllen des Formulars zu validieren und bei Verstoß gegen bestimmte Regeln eine

Fehlermeldung auszugeben. Dies kann die Qualität der in das Formular eingetragenen Daten steigern und Falscheingaben verhindern.

Dazu die zu prüfende Spalte markieren und im Ribbon auf „Column Validation“ klicken (Bild links).

Im Dialogfenster muss dann die Bedingung eingetragen werden, die es zu erfüllen gilt. Außerdem ist eine Meldung zu definieren, welche erscheint, falls gegen die Bedingung verstößen wird (Bild rechts).



- Beim Erstellen der Listenspalten sind Datentyp und Spaltenname anzugeben. Es kann auch festgelegt werden, ob es sich um ein Pflichtfeld handeln soll. Im Beispiel wurden einige Listenspalten erzeugt (siehe Bild). Es ist auch möglich, die Standard-Spalte „Title“ umzubenennen¹ (im Bsp. „Projekttitel“). Nach dem Hinzufügen bzw. Bearbeiten der Spalten müssen die Änderungen noch gespeichert werden (erkennbar durch den kleinen Stern im Reiter der Liste). Erst dann werden die Änderungen an den Server übertragen.

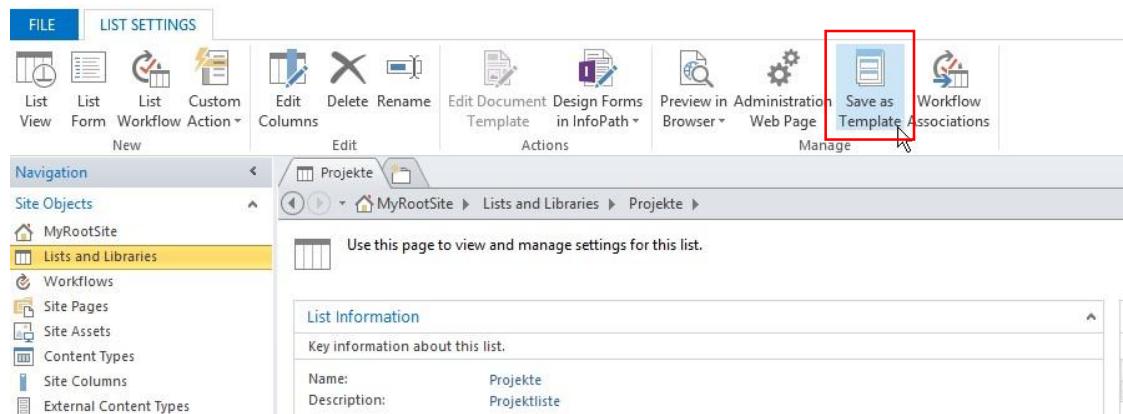
Column Name	Type	Description	Required
Titel	Single line of text		Yes
Projektstart	Date and Time		
Projektende	Date and Time		
Umsatz	Currency (\$, £, €)		
Projektverantwortlicher	Person or Group		

- Nachdem die Liste inkl. Der Spalten angelegt wurde, kann daraus eine Listenvorlage erzeugt werden. Eine Listenvorlage hat den Vorteil, dass Endanwender über die Weboberfläche dann

¹ Anmerkung: dabei handelt es sich um die Anzeigespalte („DisplayName“). Der interne Spaltenname bleibt weiterhin „Title“. Dieser ist für den Anwender jedoch nicht sichtbar.

mit einem Klick eine Liste nach dem entsprechenden Muster (im Bsp. Eine Projektliste mit projektbezogenen Spalten) erstellen können.

Dazu in der Listenübersichtsseite im Ribbon auf den Eintrag „Save as Template“ klicken, um auf die entsprechende Page im Webbrowser weitergeleitet zu werden.



- Im Browser müssen dann noch Dateiname, Titel der Vorlage sowie optional auch eine Beschreibung eingetragen werden. Die Option „Include Content“ kann bereits erstellte Inhalte (im Bsp. Projekte) in die Vorlage mit einfügen. Da die Liste gerade erst erstellt wurde, sind natürlich noch keine Inhalte in Form von Listenelementen vorhanden.

The screenshot shows the 'Save as Template' dialog. The 'File Name' field contains 'Projektliste'. The 'Template name:' field contains 'Projektliste'. The 'Template description:' field contains 'Listenvorlage zum Anlegen und Bearbeiten von Projekten'.

- Nach einem Klick auf OK wurde die Listenvorlage erfolgreich im *.stp-Format erstellt, was mit einer Meldung bestätigt wird. Die Listenvorlage befindet sich nun im Bereich „Web Designer Galleries“ → „List Templates“, erreichbar über die Websiteeinstellungen der entsprechenden Website.

Name	Edit	Modified	Title	Language	Product Version	Feature ID
Projektliste		2/7/2013 10:20 AM	Projektliste	English	4	{00BFEA71-DE22-43E

- Um eine neue Liste basierend auf dieser Listenvorlage zu erstellen, muss im Websitemenü auf „Add an App“ und anschließend auf die entsprechende Listenvorlage (im Bsp.

„Projektliste“) geklickt werden. Anschließend sind noch Listenname und ggf. eine Beschreibung anzugeben und die Liste wird erzeugt.

The screenshot shows the SharePoint 'Site Contents > Your Apps' page. At the top, there are navigation links: 'MyRootSite', 'How To Use This Library', 'Lpz', and 'EDIT LINKS'. Below the navigation is a search bar with the placeholder 'Find an app' and a magnifying glass icon. The main content area is divided into two sections: 'Noteworthy' and 'Apps you can add'. In the 'Noteworthy' section, three apps are listed: 'Document Library' (Popular built-in app), 'Custom List' (Popular built-in app), and 'Tasks' (Popular built-in app). Each item has a blue icon, the name, a brief description, and a 'App Details' link. In the 'Apps you can add' section, four apps are listed: 'Projektliste' (highlighted with a red border), 'Promoted Links', 'Report Library', and 'Survey'. Each item has a blue icon, the name, a brief description, and a 'App Details' link. The 'Projektliste' icon is highlighted with a red border.

Verbinden zweier Listen mittels Nachschlage-Feld

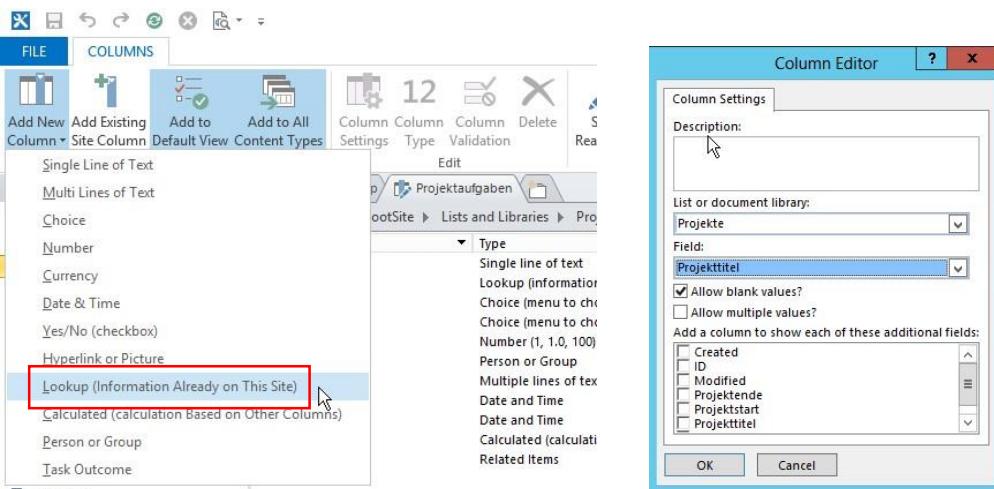
Genau wie in einem Datenbanksystem, in dem zwei Tabellen zueinander in Beziehung stehen können (z.B. 1:n-Beziehung), ist es in SharePoint möglich, Listen mit Hilfe sog. Nachschlage-Felder (Lookup) miteinander zu verbinden².

- Um eine 1:n-Beziehung zu erzeugen müssen „Master“-Liste (im Bsp. Projekte) und „Detail“-Liste (im Bsp. Projektaufgaben) vorhanden sein. Weiterhin muss eine Fremdschlüsselbeziehung hergestellt werden. Dazu erhält die Detailliste eine zusätzliche Spalte vom Typ Nachschlagefeld (Lookup), welche auf eine Spalte aus der Masterliste (im Bsp. Projekttitel) verweist, wie auf der linken Darstellung abgebildet. Im Dialogfenster (Bild rechts) sind Zielliste und –Spalte auszuwählen. Optional können noch sog. Picker-Felder ausgewählt werden, d.h. Spalten der Masterliste, welche zusätzlich in der Detailliste mit angezeigt werden. Mit der Option „Allow blank values?“ kann geregelt werden, ob es sich um

² Achtung: Standardmäßig ist in SharePoint die referentielle Integrität (Löscheschränkung / Löschweitergabe) der Listenbeziehung deaktiviert. Dies ermöglicht unabhängiges Löschen von Elementen in beiden Listen. Wird z.B. ein Projekt gelöscht, dem bereits Projektaufgaben zugewiesen sind, liegen nach dem Löschvorgang „verwaiste“ Projektaufgaben vor, die keine Projektreferenz mehr besitzen.

Die referentielle Integrität lässt sich im SharePoint Designer beim Anlegen der Lookup-Spalte leider nicht aktivieren. Dies kann im Web in den entsprechenden Spalteneinstellungen nachgeholt werden.

ein Pflichtfeld handelt. „Allow multiple values?“ ermöglicht die Realisierung einer m:n-Beziehung (ein Projekt kann mehreren Projektaufgaben zugeordnet sein und umgekehrt).



- Der Name der Nachschlagespalte muss nachträglich eingetragen werden (umbenennen von NewColumn1 in z.B. „Projekttitle“), da der Designer standardmäßig nicht die Möglichkeit bietet, diesen beim Erstellen direkt zu vergeben.
- Im Anschluss muss die Liste gespeichert werden, damit die neue Spalte hinzugefügt wird.

Anzeigeformular für Elemente um Informationen aus verbundener Liste erweitern

Der Mehrwert einer 1:n-Beziehung zwischen zwei Listen offenbart sich durch die Darstellung und Filterung beider Listen auf einer Page bzw. einem Formular. So lässt sich bspw. im „Anzeigen“Formular des aktuell ausgewählten Projekts darstellen, welche zugehörigen Projektaufgaben vorhanden sind. Dies kann wie folgt realisiert werden:

- Da die Informationen auf dem Ansicht-Formular der Masterliste (z.B. Projekte) dargestellt werden sollen, muss dieses angepasst werden. Dazu in der Listenübersichtsseite rechts im Bereich „Forms“ auf „DispForm.aspx“ klicken, um das Formular zu bearbeiten.

The screenshot shows the 'List Information' page for a list named 'Projekte'. It includes sections for 'List Information', 'Customization', 'Settings', 'Content Types', and 'Views'. The 'Views' section shows a table with one item: 'All Items' (Type: HTML, Default: Yes). The 'Forms' section shows a table with three items: 'DispForm.aspx' (Type: Display, Default: Yes), 'EditForm.aspx' (Type: Edit, Default: Yes), and 'NewForm.aspx' (Type: New, Default: Yes). The 'DispForm.aspx' row is highlighted with a red border.

- Es öffnet sich die Code-Ansicht der DispForm.aspx. Hier muss der Mauscursor direkt hinter den </WebPart>-Tag gesetzt werden, kurz vor Ende des editierbaren Inhaltsbereichs³.
- An dieser Stelle wird über das Ribbon → Insert → Data View eine Datenblattansicht auf die Detailliste (z.B. Projektaufgaben) eingefügt.

The screenshot shows the SharePoint ribbon with the 'Insert' tab selected. In the 'Data' section, the 'View' button is highlighted. Below the ribbon, the navigation pane shows 'Lists and Libraries' selected. A yellow box highlights the 'Projektaufgaben' list item under 'Projekte'. The code editor at the bottom contains the XML for the DispForm.aspx page, with a yellow box highlighting the area just before the closing </WebPart> tag.

- Im Anschluss erscheint im Ribbon der Bereich „List View Tools“, in dessen Reiter Options sich ein Eintrag „Add/Remove Columns“ befindet (Bild links). Über diesen kann festgelegt werden, welche Spalten der Detailtabelle mit angezeigt werden sollen (Bild rechts). Im Bsp. kann das

³ Der gelblich hinterlegte Codebereich ist im normalen Bearbeitungsmodus gesperrt und kann nicht verändert werden. Dies betrifft Code, der von der Page-Definition (Masterpage) geerbt wird, also die Seitenstruktur. Der Codebereich, der nicht gelb hinterlegt ist, ist der Inhaltsbereich. Dieser kann beliebig bearbeitet werden.

Projektaufgaben-Feld „Projekttitle“ entfernen werden, weil diese Information sonst doppelt auftauchen würde.

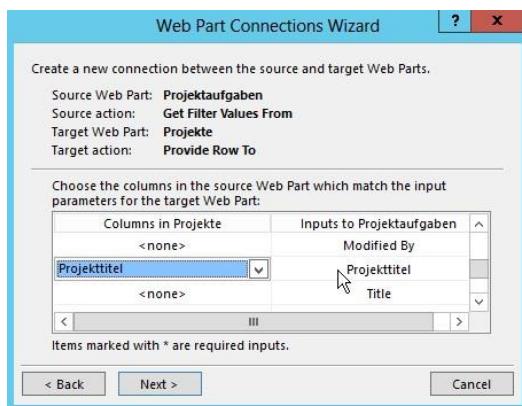
- Mit der Option „Paging“ kann die Anzahl der maximal anzuzeigenden Elemente festgelegt werden.
„Display sets of“

- Nun muss noch eine Webpart-Verbindung eingerichtet werden, die dafür sorgt, dass der Provider-Webpart (z.B. Projekte) seine Identität an den Consumer-Webpart (z.B. Projektaufgaben) sendet. Dazu einfach im grünen Ribbon-Bereich „List View Tools“ auf „Add Connection“ klicken.

- Im folgenden Assistenten wird die Webpart-Verbindung konfiguriert. Wenn die Verbindung vom Consumer-Webpart (z.B. Projektaufgaben) aus erzeugt wird, müssen folgende Einstellungen getroffen werden: „Source Webpart: Projektaufgaben“ → „Source Action: Get filter Values from“ → „Target Webpart: Provider-Webpart (z.B. Projekte)“ → „Target Action: Provide Row to“.

Anschließend muss noch die Spaltenzuweisung für die Fremdschlüsselbeziehung festgelegt werden. Hier muss die Spalte, über die die Listen miteinander verbunden werden können, ausgewählt werden. Dies ist im Bsp. jeweils die Spalte „Projekttitle“. Anschließend kann der Assistent mit einem

Klick auf „Next“ und „Finish“ geschlossen werden. Die folgende Grafik zeigt noch einmal alle Einstellungen des Assistenten im Überblick:



- Zum Schluss muss die DispForm.aspx noch einmal gespeichert werden. Im Anschluss kann sie gleich im Browser getestet werden. Dazu in der Weboberfläche die entsprechende Masterliste (z.B. Projekte) aufrufen, ein Element auswählen und auf „View Item“ klicken (Bild links). Im darauf erscheinenden Formular sollte nun unterhalb des entsprechenden MasterElements die zugehörigen Detail-Elemente angezeigt werden (Bild rechts).

Projekte ①

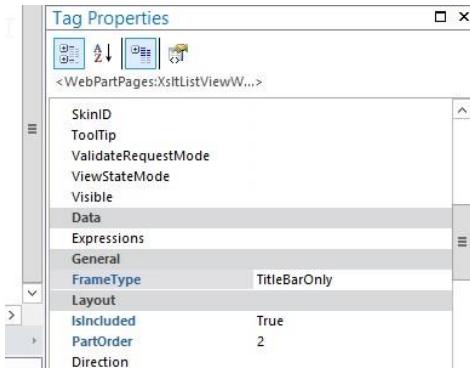
[+ new item or edit this list](#)

All Items	...	Find an item
✓ Projekttitle	Projektstart	Projektende
✓ Projekt A ✎	...	Umsatz
✓ Projekt B ✎	...	Projektverantwortlich
	NewDispForm	
	View Item	
	Edit Item	
	Compliance Details	
	Workflows	

Projekttitle	Projekt A
Projektstart	
Projektende	
Umsatz	
Projektverantwortlicher	
Created at 2/7/2013 2:36 PM by	<input type="checkbox"/> System Account
Last modified at 2/7/2013 2:36 PM by	<input type="checkbox"/> System Account
Close	

- Um das Ansichtsformular nun noch etwas übersichtlicher zu gestalten bietet es sich an, über den entsprechenden Web Parts noch die passende Überschrift anzeigen zu lassen. Dies geschieht am einfachsten durch das setzen des „FrameType“-Attributs auf „TitleBarOnly“ im Bereich Tag Properties⁴ des WebParts. Dadurch wird bewirkt, dass der WebPart-Titel immer angezeigt wird.

⁴ Der Bereich „Tag Properties“ muss einmalig eingeblendet werden, da er standardmäßig nicht mit angezeigt wird.
Ribbon → View → Task Panes → Tag Properties



- Nachdem alle Anpassungen vorgenommen wurden sollte das Anzeigeformular für die Masterliste (z.B. Projekte) nun etwa so aussehen:

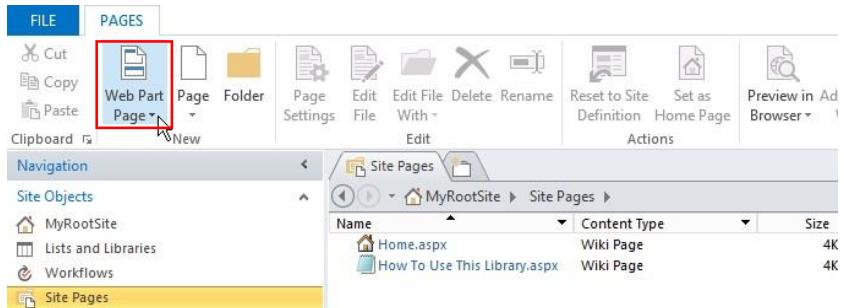
WebPart-Page mit zwei verbundenen Listen

Ein weiterer simpler Weg, Informationen von zwei Listen auf einer Seite darzustellen, ist die Verwendung einer WebPart-Page mit zwei verbundenen WebParts.

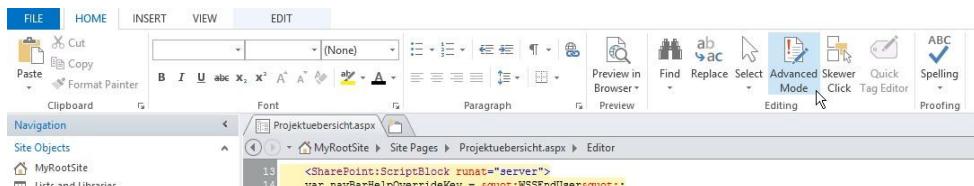
Im Beispiel soll eine einfache Projektübersichtsseite in Form einer WebPart Page mit zwei Listenansichts-WebParts Projekte und Projektaufgaben erstellt werden⁵.

- Als erstes ist eine leere WebPart-Page zu erstellen. Dies erfolgt in der Navigationsleiste im Punkt „Site Pages“. Im anschließend angepassten Ribbon ist der Bereich „New“ mit dem Eintrag „WebPart-Page“ finden. Nach einem Klick auf diesen Eintrag kann eine Vorlage für die WebPart-Page ausgewählt werden. Da im Beispiel nur zwei WebParts untereinander eingebunden werden sollen, reicht in dem Fall die Standard-Vorlage mit einem horizontal breitflächig dargestellten WebPart.

⁵ In SharePoint 2013 können keine Listen vom Typ „Aufgabe“ mehr als Detail-Liste (untergeordnet) für die Verbindung verwendet werden, da der Filter nicht mehr greift. Bei einer Custom List funktioniert die Filterung. ⁶ Vorsicht: Im erweiterten Bearbeitungsmodus sind alle Bereiche zur Bearbeitung freigegeben. Bei falscher Handhabung kann so schnell die Struktur beschädigt und somit ein fehlerfreies Parseen dieser Page seitens des Web Servers verhindert werden. Es empfiehlt sich daher, die gelb hinterlegten Code-Bereiche nicht zu ändern.



- Nachdem die WebPart-Page erzeugt wurde, kann der Dateiname gleich umbenannt werden (im Bsp. In „Projektuebersicht.aspx“). Dabei ist darauf zu achten, dass die Dateiendung „.aspx“ erhalten bleibt, da der SharePoint die Page sonst nicht interpretieren kann.
- Durch einen Klick auf die WebPart-Page erscheint die Übersichtsseite der Page, in der sich der Bereich „Customize“ mit dem Link „Edit file“ befindet. Mit diesem Link wird die Page in der Code-Ansicht im Standard-Bearbeitungsmodus geöffnet, was durch die gelb hinterlegten, gesperrten Codebereiche deutlich zu erkennen ist.
- Um Listenansichts-WebParts einzufügen, muss in den erweiterten Bearbeitungsmodus gewechselt werden⁶. Dies erfolgt im Ribbon → Home → Advanced Mode.



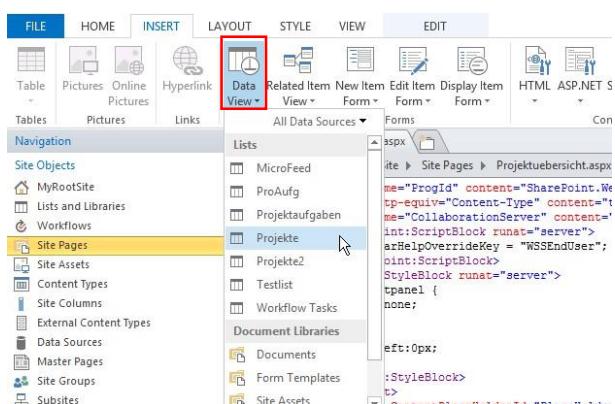
- Bevor die WebParts eingefügt werden können, muss der Cursor an die richtige Stelle platziert werden. Diese liegt zwischen den <ZoneTemplate>-Tags innerhalb des <table>-Tags am Ende der Page, wie folgendes Bild verdeutlicht:

```

39 " <table class="ms-core-tableNoSpace ms-webpartPage-root" width="100%">
40   <tr>
41     <td id="invisiblEmpty" name="_invisiblEmpty" valign="top" width="100%">
42       <WebPartPages:WebPartZone runat="server" Title="loc:FullPage" ID="FullPage" FrameType="TitleBarOnly"><ZoneTemplate></ZoneTemplate></WebPartZone>
43     </tr>
44   </table>
45 </asp:Content>
46

```

- Zum Einfügen des ersten WebParts im Ribbon → Insert → Data View → die entsprechende Liste (im Bsp. Projekte) auswählen.



- Analog der Vorgehensweise aus Kapitel... lassen sich mit „Add/Remove Columns“ die anzuzeigenden Spalten und mit „Paging“ die maximal anzuzeigenden Elemente festlegen.
- Direkt unter dem ersten WebPart soll nun das zweite eingefügt werden (im Bsp. Projektaufgaben). Damit es an der richtigen Stelle im Code platziert werden kann und die entsprechenden Ribbon-Einträge verfügbar gemacht werden, muss der Cursor zuvor an das Ende des ersten WebParts gesetzt werden, nach dem schließenden Tag:
`</WebPartPages:XsltListViewWebPart>`. Das Einfügen des zweiten WebParts erfolgt genau wie beim Ersten.
- Die letzten Schritte zum Erstellen der WebPart-Verbindung und zum Einfügen des WebPartTitels erfolgen analog der Vorgehensweise, die in Kapitel ... beschrieben wurde.
- Nach letztmaligem Speichern kann nun mittels Vorschaufunktion in der Schnellzugriffsleiste direkt in den Browser auf die WebPart-Page gewechselt werden.



- Nun sollte die WebPart-Page in etwa wie auf folgender Abbildung aussehen. Nach einem Klick auf das neue bilaterale Pfeilsymbol im Bereich Projekte werden im Beispiel in der unteren Ansicht nur die dem ausgewählten Projekt zugeordneten Projektaufgaben angezeigt:

MyRootSite How To Use This Library Lpz EDIT LINKS

Projektübersicht ⓘ



Projekte

[+ new item or edit this list](#)

✓	Select	Projekttitle	Projektstart	Projektende	Umsatz	Projektverantwortlicher
		Projekt A *	...			
		Projekt B *	...			

Projektaufgaben

[+ new item or edit this list](#)

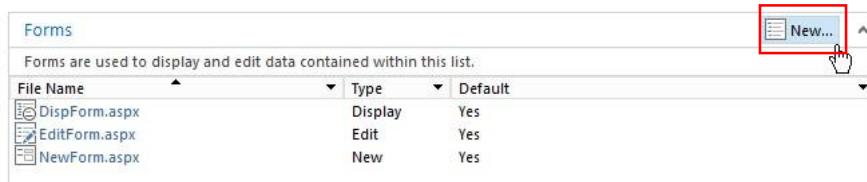
✓	Title	Zugewiesen an	Projekttitle
	Task1 *	...	Projekt A
	Anderes Task *	...	Projekt A

Benutzerdefiniertes Formular über eigenen Ribbon-Eintrag öffnen (Custom Action)

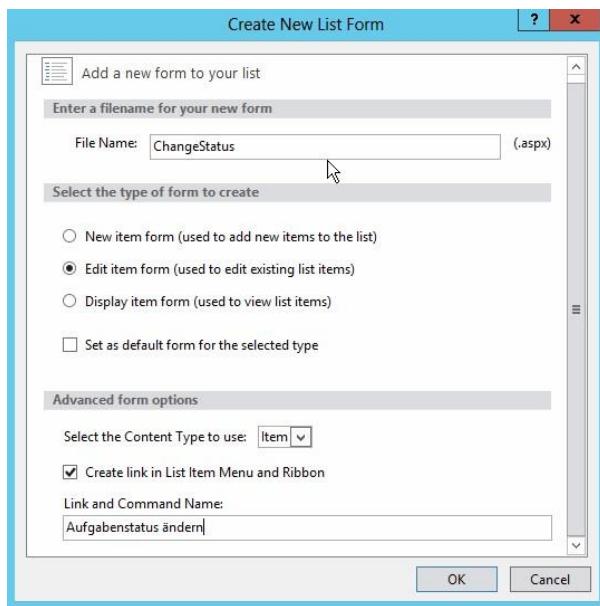
Mit dem Designer lassen sich benutzerdefinierte Formulare zum Anzeigen (*DispForm.aspx*), Anlegen (*NewForm.aspx*) und Bearbeiten (*EditForm.aspx*) erstellen. So kann z.B. erreicht werden, dass bestimmte Felder des Standard Formulars gesperrt oder entfernt werden.

Weiterhin kann dieses benutzerdefinierte Formular über einen zusätzlichen Ribbon- oder Elementmenüeintrag - ggf. abhängig von bestimmten Berechtigungsstufen – aufgerufen werden.

- Im ersten Schritt wird das neue Formular erstellt. Dies geschieht in der entsprechenden Listenübersichtsseite rechts im Bereich „Forms“. Im Beispiel soll ein Formular zum Ändern (*EditForm*) des Aufgabenstatus einer Projektaufgabenliste erzeugt werden.



- Im Dialogfeld ist Name und Typ des Formulars einzugeben. Ggf. kann es auch als Standardformular festgelegt werden. In dem Fall würde es immer bei einem Klick auf „Edit Item“ aufgerufen werden. Im unteren Bereich kann bereits eine erste benutzerdefinierte Aktion (Custom Action) automatisch erzeugt werden. Dabei handelt es sich um das Anlegen eines Eintrages im Element-Kontextmenü⁶.



- Nach einem Klick auf „OK“ wird die Codeansicht des neu erzeugten Formulars aufgerufen. Standardmäßig erzeugt er immer ein Standardformular, welches nun angepasst werden kann

⁶ Fälschlicherweise wird angekündigt, den Eintrag auch im Ribbon zu erzeugen, was jedoch nicht zutrifft. Der Eintrag wird lediglich im List Item Menü (Kontextmenü) erzeugt. D.h. für den Ribbon-Eintrag ist eine gesonderte Custom Action notwendig.

(z.B. verschiedene Spalten ausblenden oder mit Schreibschutz versehen). Im Bsp. Sollen alle Felder außer Aufgabenstatus schreibgeschützt dargestellt werden.

Dazu im Quelltext runter scrollen und im <table>-Bereich nach den zu ändernden Spalten Ausschau halten. In folgendem Beispiel wird das Feld „Title“ der Projektaufgabe schreibgeschützt gesetzt. Dies geschieht, indem im entsprechenden

<SharePoint:FormField>-Tag das „ControlMode“ property von „Edit“ auf „Display“ geändert wird.

```
<td width="190px" valign="top" class="ms-formlabel">
    <H3 class="ms-standardheader">
        <nobr>Title<span class="ms-formvalidation"> *</span>
        </nobr>
    </H3>
</td>
<td width="400px" valign="top" class="ms-formbody">
    <SharePoint:FormField runat="server" id="ff1[$Pos]" ControlMode="Display" FieldName="Title" __designer:bind="{ddwrt:DataBind 'true';Index('1')}">
        <SharePoint:FieldDescription runat="server" id="ff1description[$Pos]" FieldName="Title" ControlMode="Edit"/>
    </SharePoint:FormField>
    </td>
</tr>
```

- Gleicher muss entsprechend für alle gewünschten Spalten wiederholt werden.
- Wenn eine Spalte nicht sichtbar sein soll, muss die betroffene Table Row ausgeblendet⁷ oder gelöscht werden. Ersteres geschieht am einfachsten durch Einfügen eines „style“-Attributs mit dem Wert „display:none“. Im Beispiel soll die Spalte „Kategorie“ ausgeblendet werden.

```
<tr style="display:none">
    <td width="190px" valign="top" class="ms-formlabel">
        <H3 class="ms-standardheader">
            <nobr>Kategorie</nobr>
        </H3>
    </td>
    <td width="400px" valign="top" class="ms-formbody">
        <SharePoint:FormField runat="server" id="ff5[$Pos]" ControlMode="Edit" FieldName="Kategorie" __designer:bind="{ddwrt:DataBind 'true';Index('5')}">
            <SharePoint:FieldDescription runat="server" id="ff5description[$Pos]" FieldName="Kategorie" ControlMode="Edit"/>
        </SharePoint:FormField>
    </td>
</tr>
```

Im Beispiel sieht das Formular mit gesperrten Titel-, Zugewiesen an-, Projekttitel-, Date due-, Spalten und ausgeblendeter Kategorie-Spalte am Ende wie folgt aus:

The screenshot shows a SharePoint form with the following fields and their values:

Title *	Task1
Zugewiesen an	<input type="checkbox"/> Administrator
Projekttitel	Projekt A
Date Due	2/9/2013 12:00 AM
Status	<div style="border: 1px solid #ccc; padding: 2px;">Created In Progress Finished</div>

Below the form, a status bar shows:

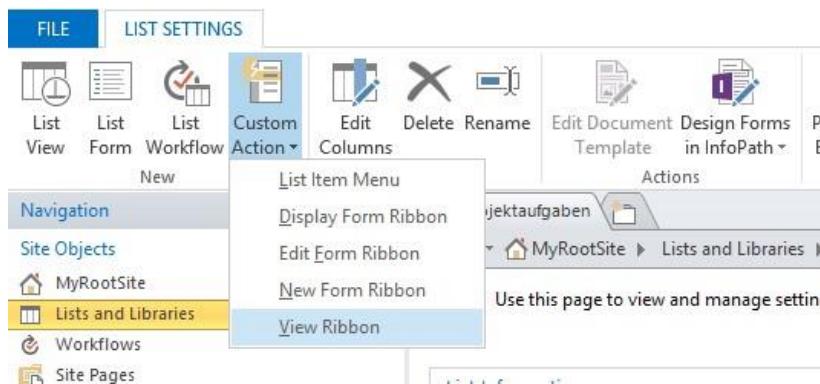
Created at 2/7/2013 4:47 PM by System Account
Last modified at 2/8/2013 4:46 PM by System Account

At the bottom right are "Save" and "Cancel" buttons.

- Dieses benutzerdefinierte Formular ist jetzt in der Liste über das Kontextmenü eines ausgewählten Listenelements aufrufbar. Damit das Formular aber auch über einen RibbonEintrag aufrufbar wird, muss eine benutzerdefinierte Aktion (Custom Action) erstellt werden.

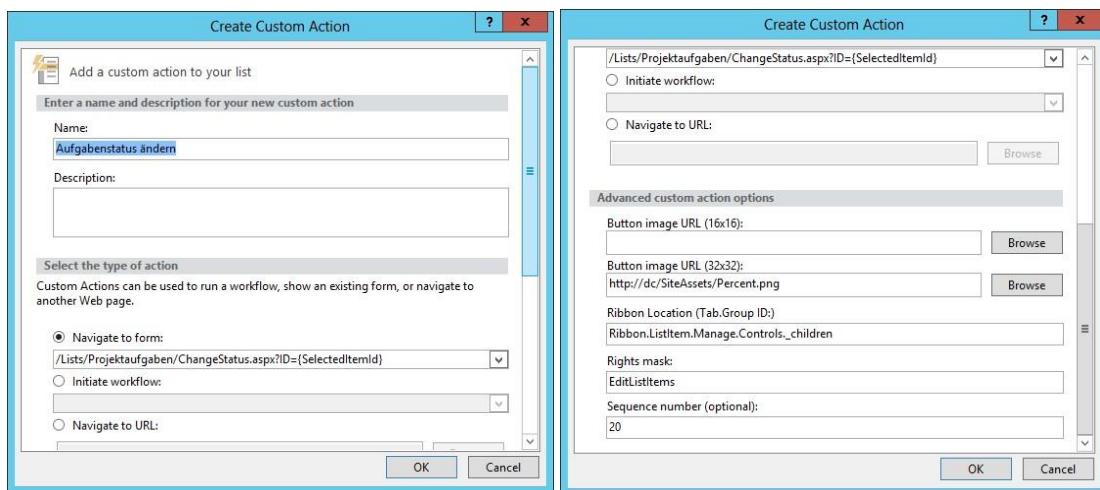
⁷ Anmerkung: Das <SharePoint:FormField> Steuerelement wird trotzdem serverseitig geparsst und somit an den Client ausgeliefert. Es wird durch den style-Tag lediglich ausgeblendet, taucht jedoch im Quellcode auf. Dies ist die weichere, aber ggf. auch unsicherere Variante. Wenn das Steuerelement komplett verbannt werden soll, muss die Table Row nicht ausgeblendet, sondern komplett gelöscht werden.

Dies erfolgt im Designer auf der Übersichtsseite der Liste über den Ribbon-Bereich „New“ → „Custom Action“ → „View Ribbon“.



- Im Dialogfenster sind folgende Einträge vorzunehmen:

Name	Name der Aktion / das Label des Ribbon-Eintrags
Navigate to form	Zielformular, das aufgerufen werden soll. Im Bsp. "ChangeStatus.aspx"
Button image URL (32x32)	Link zu einem Bild, das der Ribbon-Eintrag verwenden soll ⁸ . Es muss 32x32px groß sein, da keine Skalierung möglich ist.
Ribbon Location	Zielbereich / Stelle im Ribbon. Im Bsp. "Ribbon.ListItem.Manage.Controls._children" ⁹ (steht für das Menüband -> Reiter Element -> im Bereich Verwalten)
Rights mask	Berechtigungsstufe, die erforderlich, damit der angemeldete User diesen Ribbon-Eintrag benutzen kann. Im Bsp. „EditListItems“ (Listenelemente bearbeiten)
Sequence number	Position innerhalb des Ribbon-Bereichs (Location). 0 ist die erste Position, 10 die zweite, 20 die dritte etc.

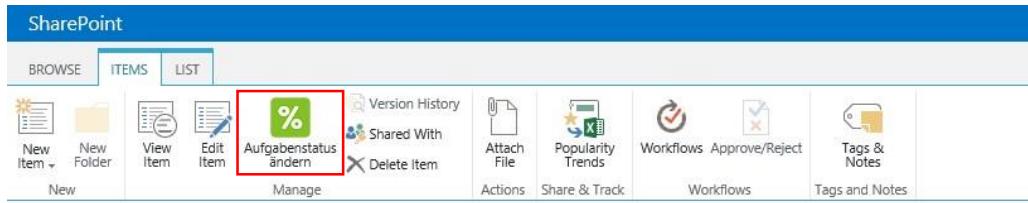


- Nachdem die Aktion erstellt wurde, ist der Ribbon-Eintrag verfügbar. Zum Testen muss die Listenansicht im Browser einfach aktualisiert werden (F5) und anschließend sollte der Eintrag

⁸ Z.B. Referenz auf ein Bild, welches in der Bibliothek SiteAssets (WebsiteObjekte) liegt.

⁹ Falls der Name des Zielbereichs nicht bekannt ist, kann dieser mit Hilfe der Hervorhebefunktion der IE Entwicklertools (F12 anschließend STRG+B) ermittelt werden, indem die gewünschte Stelle im Ribbon angeklickt wird. Im Quelltext erscheint dann der Bereich.

an der gewünschten Position im Menüband auftauchen.



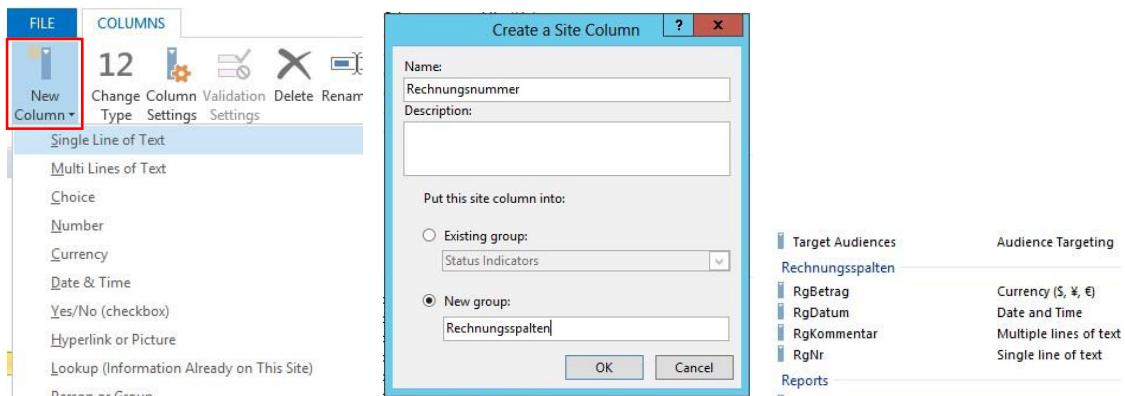
The screenshot shows a SharePoint interface with the 'LIST' tab selected in the ribbon. The ribbon toolbar includes various icons for actions like 'New Item', 'View Item', 'Edit Item', and 'Aufgabenstatus ändern'. A red box highlights the 'Aufgabenstatus ändern' icon. Below the ribbon, there's a navigation bar with links like 'Home', 'Documents', 'Site Pages', and 'Recent'. The main area displays a list of items with columns: Title, Zugewiesen an, Projekttitle, Date Due, Kategorie, and Status. One item is selected, showing 'Task1' in the Title column and 'Administrator' in the Zugewiesen an column.

Title	Zugewiesen an	Projekttitle	Date Due	Kategorie	Status
Task1	Administrator	Projekt A	2/9/2013 12:00 AM	IT	Created
NochnTask	...	Projekt B			
Anderes Task	...	Projekt A			

Benutzerdefinierten Inhaltstyp mit Dokumentvorlage erstellen

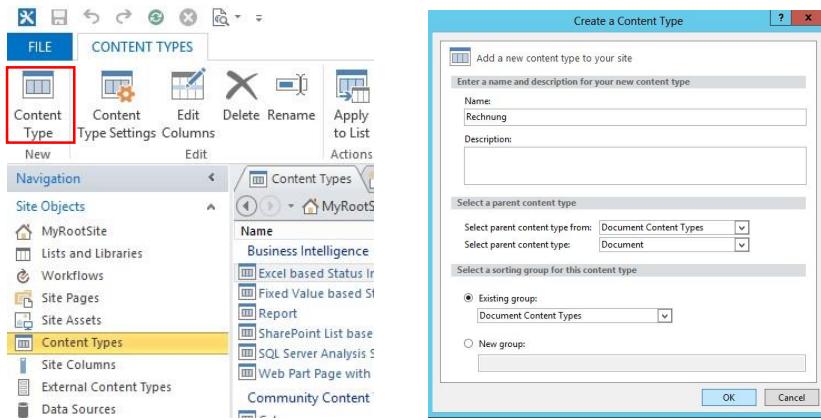
Mit wenigen Klicks lässt sich im SharePoint Designer ein eigener Inhaltstyp anlegen, der dann an beliebige Listen bzw. Bibliotheken gebunden werden kann. Neben Websitespalten kann ein Inhaltstyp auch eine benutzerdefinierte Dokumentvorlage enthalten, wie im Folgenden am Beispiel des Inhaltstyps „Rechnung“ gezeigt wird.

- Bevor der Inhaltstyp erstellt werden kann, müssen erst die entsprechenden Websitespalten angelegt werden. Dies geschieht im Navigationsbereich unter „Site Columns“ und dann im Ribbon mit einem Klick auf „New Column“ und anschließender Auswahl des Spaltentyps (Bild links). Im darauf folgenden Fenster sind die Spalteneigenschaften anzugeben. Es bietet sich an, logisch zusammengehörige Spalten in einer eigenen Spaltengruppe zusammen zu fassen. (Bild Mitte). Im Bsp. werden die Spalten Rechnungsnummer, -betrag, -datum und – Kommentar in der Gruppe „Rechnungsspalten“ zusammengefasst¹⁰ (Bild rechts).

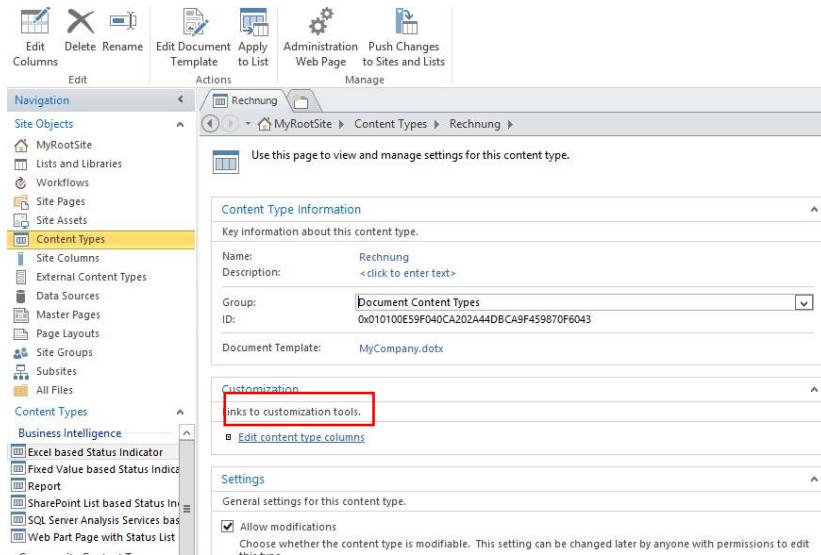


- Nachdem alle Spalten erstellt und die Änderungen gespeichert und somit an den Server übertragen wurden folgt im nächsten Schritt die Erstellung des Inhaltstyps selbst. Im Navigationsbereich unter „Content Types“ kann dies im Ribbon mit einem Klick auf „New Content Type“ durchgeführt werden (Bild links). Im Dialogfenster sind Name, übergeordneter Inhaltstyp und Zielgruppe des neu zu erstellenden Inhaltstyps anzugeben. Im Bsp. soll ein Dokumentinhaltstyp erstellt werden, weshalb bei übergeordneter Inhaltstyp „Document“ aus der Gruppe „Document Content Types“ ausgewählt wurde. Der neue Inhaltstyp „Rechnung“ soll in der Gruppe „Document Content Types“ angelegt werden (Bild rechts). Prinzipiell spielt die Gruppe jedoch keine Rolle und dient, genau wie die Gruppierung der Websitespalten, nur der besseren Übersicht für den Anwender.

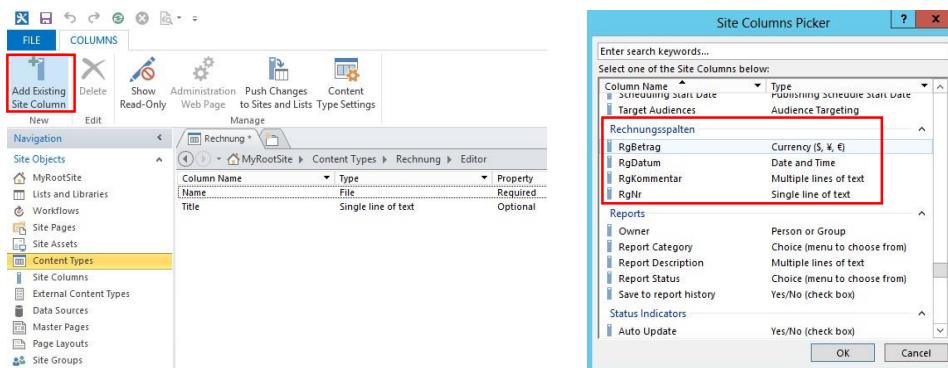
¹⁰ Nach dem Erstellen der Spaltengruppe mit der ersten Spalte muss einmal gespeichert werden, damit der Designer die Spaltengruppe an den Server übermitteln kann, andernfalls ist die Gruppe bei den später erstellten Spalten noch nicht verfügbar.



- Nachdem der Inhaltstyp erstellt wurde, können die Websitespalten zugeordnet werden. Hierfür ist der entsprechende Inhaltstyp zu öffnen und im Bereich „Customization“ auf „Edit content Type Columns“ zu klicken.



- Im Ribbon können nun mit einem Klick auf „Add Existing Site Columns“ (Bild links) die gewünschten Websitespalten nacheinander hinzugefügt werden (Bild rechts). Anschließend muss der Inhaltstyp erneut gespeichert werden, damit die Änderungen an den Server übertragen werden.



- Jetzt ist die Datenstruktur des Inhaltstyps fertig gestellt worden. Als nächstes folgt die Zuordnung einer benutzerdefinierten Vorlage. Dies kann wie im Bsp. eine Word-Vorlage (Dateityp *.dotx) sein. Um die Vorlage hochzuladen muss zunächst auf die

Administrationsseite des Inhaltstyps gewechselt werden, was mit einem Klick auf den

The screenshot shows the SharePoint 'CONTENT TYPE SETTINGS' page. The ribbon has 'FILE' selected. Below the ribbon are buttons for 'Edit', 'Delete', 'Rename', 'Edit Document Template', 'Apply to List', 'Administration Web Page' (which is highlighted with a red box), 'Push Changes to Sites and Lists', and 'Manage'. The left navigation pane shows 'Navigation' with 'Rechnung' selected, and 'Site Objects' with 'MyRootSite' and 'Lists and Libraries'. The main content area says 'Use this page to view and manage settings for this content type.'

Ribboneintrag „Administration Web“

Page“ geschieht.

- Im Beispiel soll eine Word-Dokumentvorlage mit folgenden Inhalten erstellt werden:
Überschrift (Rechnung Nr.) und Firmenlogo, zweispaltige Tabelle mit den Labels der Site Columns des Inhaltstyps in der linken und vorerst keinen Einträgen in der rechten Spalte sowie einem kurzen Dankes-Text unterhalb der Tabelle (Bild links). Die Datei ist anschließend als Word-Vorlage (dotx-Format)¹¹ lokal abzuspeichern (Bild rechts).

The screenshot shows a Microsoft Word document titled 'MyCompany'. It contains a logo, the text 'Rechnung Nr.', and a table with four rows labeled 'Rechnungsnummer', 'Rechnungsbetrag', 'Rechnungsdatum', and 'Rechnungskommentar'. At the bottom, it says 'Vielen Dank für Ihre Bestellung!'. To the right, a 'Save As' dialog box is open, showing the file name 'MyCompany' and the save type 'Word Template'.

- Auf der Verwaltungsseite des Inhaltstyps kann die Dokumentvorlage unter „Advanced Settings“ → „Upload New Document Template“ hochgeladen werden.

The screenshot shows the 'Site Content Type Advanced Settings' page. Under 'Document Template', there is a field to 'Enter the URL of an existing document template' and a 'Browse...' button. Below it, there is a field to 'Upload a new document template:' with a 'Browse...' button, which is highlighted with a red box. Other settings include 'Read Only', 'Should this content type be read only?', and 'Update all content types inheriting from this type?'.

- Nun kann der Inhaltstyp erstmals verwendet werden. Dazu muss zunächst eine Zielbibliothek ausgewählt bzw. eine neue erstellt – im Bsp. die neu erstellte Dokumentbibliothek „Rechnungen“ - und für diese anschließend die Verwaltung von Inhaltstypen aktiviert werden. Dies geschieht, indem

¹¹ Achtung: nachdem der Typ „Word Template“ ausgewählt wurde, wechselt das Zielverzeichnis automatisch nach „C:\Users\[username]\Documents\Custom Office Templates“.

in den Library Settings → Advanced Settings → der Punkt „Allow management of content types“ auf „Yes“ gesetzt wird.

The screenshot shows the 'Advanced Settings' page for a library. On the left, there's a navigation bar with links like 'Zurück zur Startseite', 'Blog', 'Urlaubsantrag', 'Projektübersicht', 'Lists', and 'Mitarbeiter'. The main area has a heading 'Content Types' with a sub-section about specifying content types for the library. A large red box highlights the 'Allow management of content types?' section, which contains two radio buttons: 'Yes' (selected) and 'No'.

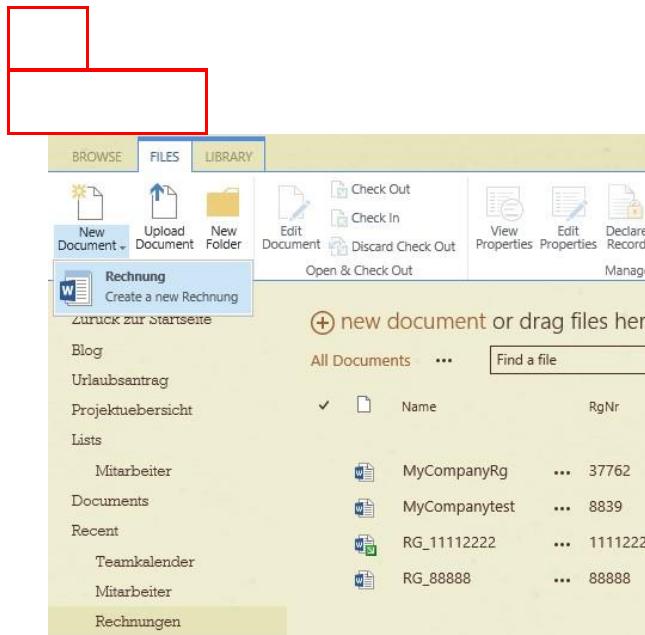
- Im Anschluss erscheint in den Library Settings der entsprechenden Bibliothek der Bereich „Content Types“. Hier ist der Punkt „Add from existing content types“ anzuklicken (Bild links). Anschließend ist der gewünschte Inhaltstyp auszuwählen, mit „Add“ zur Bibliothek hinzuzufügen (Bild rechts) und der Vorgang mit einem Klick auf „OK“ zu beenden.

The left screenshot shows the 'Content Types' settings for a library. It includes a note that the library allows multiple content types and lists 'Content Type' and 'Visible on New' options. A red box highlights the 'Add from existing site content types' link. The right screenshot shows a modal dialog titled 'Select site content types from: Document Content Types'. It lists several content types: 'basic:range', 'Dublin Core Columns', 'Form', 'Link to a Document', 'List View Style', 'Master Page', 'Master Page Preview', 'Picture', 'Rechnung' (which is highlighted with a red box), and 'Web Part Page'. Below the list is a 'Description:' field set to 'None'. To the right of the list are 'Add >' and 'Remove' buttons, with a red box highlighting the 'Add >' button.

- Der Inhaltstyp wurde nun an die Bibliothek gebunden. Die Bsp-Bibliothek besitzt jetzt zwei Inhaltstypen. Den Standardinhaltstyp „Document“ und den gerade hinzugefügten Inhaltstyp „Rechnung“.
Damit die Endanwender nur noch in der Lage sind, Dateien vom Typ „Rechnung“ hochzuladen – schließlich handelt es sich um eine Rechnungsbibliothek – kann nun noch der „alte“ Dokumentinhaltstyp ausgeblendet werden. Dazu muss in den Library Settings → Bereich „Content Types“ → auf den Link „Change new button order and default content type“ geklickt und der Inhaltstyp „Document“ durch Deaktivieren der „Visible“ Checkbox ausgeblendet werden. Nach einem Klick auf „OK“ ist der Rechnungsinhaltstyp dann als Standard festgelegt.

The screenshot shows the 'Change New Button Order' settings page. It features a 'Content Type Order' section with a note that non-visible content types won't appear on the new button. It includes a 'Visible' checkbox for each content type. The 'Rechnung' content type has its 'Visible' checkbox checked (highlighted with a red box), while the 'Document' content type's checkbox is unchecked. To the right, there's a table mapping 'Content Type' to 'Visible' status: 'Rechnung' is 'Visible' and 'Document' is 'Not Visible'.

- Nun kann ein erstes Dokument aus der Bibliothek heraus erstellt werden. Dies erfolgt im Ribbon unter Files → New Document → Rechnung. Anschließend öffnet sich Word und die Dokumentvorlage inkl. Dokumentinformationsbereich wird angezeigt¹².



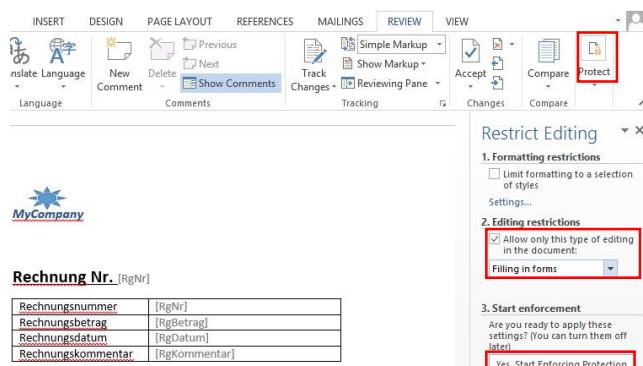
- Als nächstes gilt es, die bisher leere Dokumentvorlage mit Metadaten der Inhaltstypspalten zu füttern. Dies geschieht mit Hilfe der sog. Schnellbausteine, welche automatisch die Metadaten des Dokuments in die Vorlage schreiben. Dazu müssen über das Ribbon → Insert → Quick Parts → Document Property → die gewünschten Eigenschaften in die rechte Tabellenspalte sowie Überschrift eingefügt werden, wie folgende Bilder beschreiben.

The left screenshot shows the Word ribbon with the 'Quick Parts' tab selected. A red box highlights the 'Document Property' option under the 'Text' section. The right screenshot shows the Word document 'MyCompanyRg - Word' with a table containing a single row. The first column has the header 'Rechnung Nr.' and contains four empty cells. The second column has the header 'Rechnungsnummer' and contains four placeholder fields: '[RgNr]', '[RgBetrag]', '[RgDatum]', and '[RgKommentar]'. Above the table, the company logo 'MyCompany' is visible. The status bar at the bottom right shows 'Viele Dank für Ihre Bestellung!'.

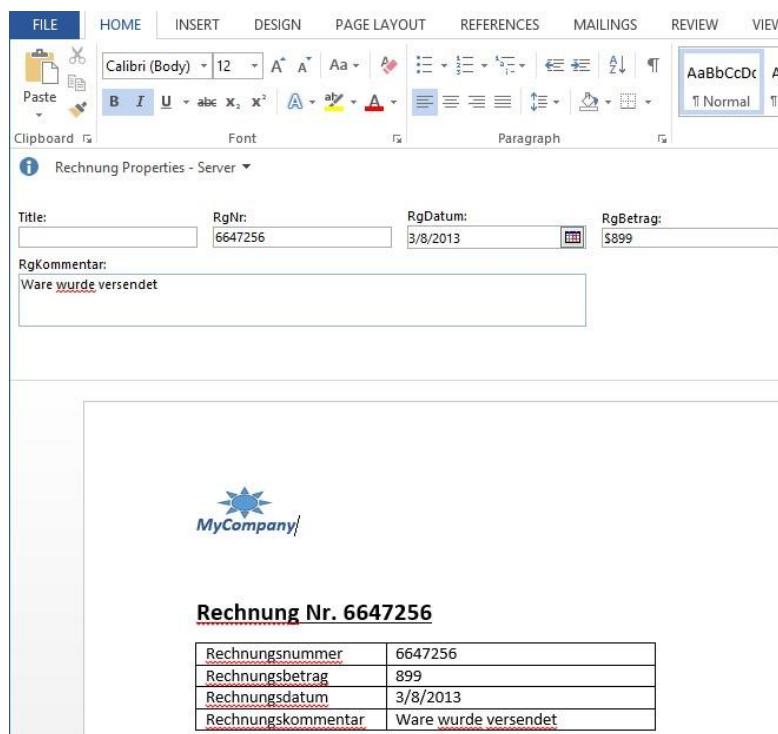
- Optional kann noch festgelegt werden, dass die Dokumentvorlage – abgesehen vom Ausfüllen der Daten im Dokumentinformationsbereich – zur Bearbeitung gesperrt werden soll. Dies erfolgt über das Ribbon → Review → Protect → im Bereich Restrict Editing → 2. Editing

¹² Sollte es beim Öffnen des Dokuments zu Problemen kommen muss die sog. „Geschützte Ansicht“ einmalig pro Client wie folgt deaktiviert werden: Ribbon File → Options → Trust Center → Trust Center Settings → Protected View → den Punkt „Enable Protected View for files located in potentially unsafe locations“ deaktivieren.

Restrictions → die Checkbox "Allow only this type of editing in the document" aktivieren → Auswahl "Filling in Forms" → 3. Start enforcement → Klick auf "Yes, Start Enforcing Protection" → mit Passwortschutz versehen.



- Als letzter Schritt ist die Dokumentvorlage zu aktualisieren. Dazu muss diese erneut lokal als „Word Template“ gespeichert (ggf. die alte überschreiben) und an den Inhaltstyp gebunden werden. Dies erfolgt wie bereits im 6./7. Punkt beschrieben.
- Der Inhaltstyp ist nun fertig implementiert und dank der Dokumentvorlage mit Schnellbausteinen können die erfassten Rechnungsdaten automatisch in die Word-Datei geschrieben werden. Dies kann getestet werden, indem erneut eine Rechnung aus der Bibliothek heraus erzeugt wird.



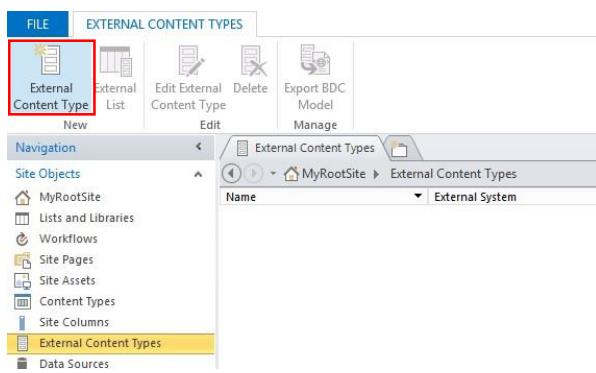
Externen Inhaltstyp anlegen (Business Data Connectivity Service)

Wie bereits der SharePoint Designer 2010 bietet auch die 2013er Version die Möglichkeit, eine Verbindung zu einer externen Anwendung (z.B. MS SQL Server) mittels eines sog. Externen Inhaltstyps in Form einer sog. Externen Liste zu erstellen.

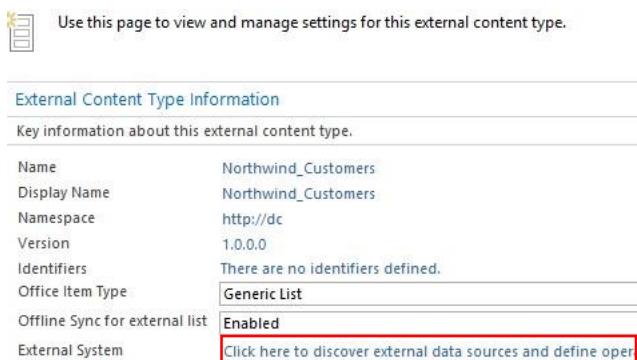
Die Verwendung eines externen Inhaltstyps offenbart Vorteile im Bereich der Usability. Für den Anwender ist es auf den ersten Blick nicht erkennbar, dass es sich bei der externen Liste nicht um eine SharePoint interne Datenquelle, sondern lediglich um eine Referenz auf eine externe Anwendung handelt. Das Look&Feel – also Aussehen und Funktionalitäten der Ansichten und Formulare – einer externen Liste entspricht weitestgehend dem einer nativen SharePoint Liste.

Im folgenden Beispiel soll eine Verbindung zur Tabelle „Customers“ der SQL Beispieldatenbank „Northwind“ erstellt werden.

- Nach einem Klick auf den Eintrag „External Content Types“ in der Navigationsleiste des SharePoint Designers ist im Ribbon → Bereich „New“ → der Eintrag „External Content Type“ auszuwählen, um mit der Erstellung eines neuen externen Inhaltstyps zu beginnen.



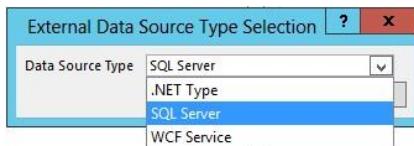
- Nachdem im Arbeitsbereich nun allgemeine Angaben zum Name, DisplayName, ggf. Office Elementtyp¹³ oder Offline-Synchronisierbarkeit gemacht wurden, kann der Assistent zur Erstellung der Verbindung mit dem externen System aufgerufen werden.



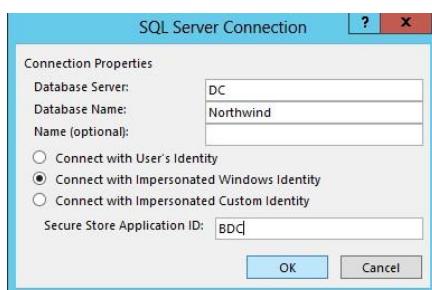
- Im nächsten Arbeitsbereich, dem Operation Designer führt ein Klick auf „Add Connection“ zu einem Auswahldialog, in dem die Verbindungsoptionen mit einer .NET-Verbindungsassembly, einer SQL

¹³ Nur von Interesse, wenn die Inhalte der externen Liste später mit Office-Produkten integriert werden sollen. Soll z.B. eine externe Liste vom Typ Kontakt soll mit Outlook synchronisiert werden, ist hier als Office Elementtyp „Contact“ auszuwählen.

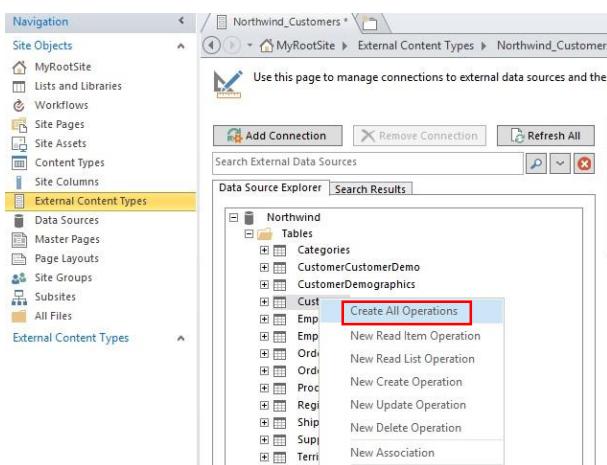
Datenbank oder einem WCF-Dienst zur Verfügung stehen. Da es sich im Beispiel um eine Tabelle einer SQL Datenbank handelt, ist der Eintrag „SQL Server“ zu wählen.



- Nun erscheint der Assistent zum Ermitteln des Connection Strings der Datenbank. Hier sind Datenbankserver, Datenbankname und Authentifizierungsmethode¹⁴ anzugeben. Im Bsp. Wurde die Authentifizierung mit Hilfe einer umgeleiteten Windows-Authentifizierung durch eine Secure Store Service Dienstanwendung realisiert. Dies setzt allerdings voraus, dass diese (im Bsp mit der Application ID „BDC“) zuvor in der Zentraladministration erstellt und konfiguriert wurde. Die Authentifizierungsmethode lässt sich im Nachhinein noch ändern.



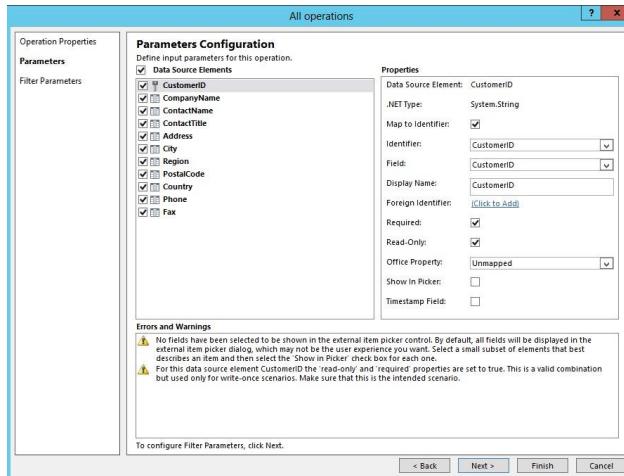
- Nachdem die Authentifizierung mit der DB erfolgt ist, kann die Zieltabelle ausgewählt werden. Mit einem Rechtsklick auf die entsprechende Tabelle (im Bsp. Customers) oder Ansicht werden die zu erstellenden Zugriffsmethoden angelegt. Um volle Schreibrechte zu erhalten ist hier „Create all Operations“ auszuwählen.



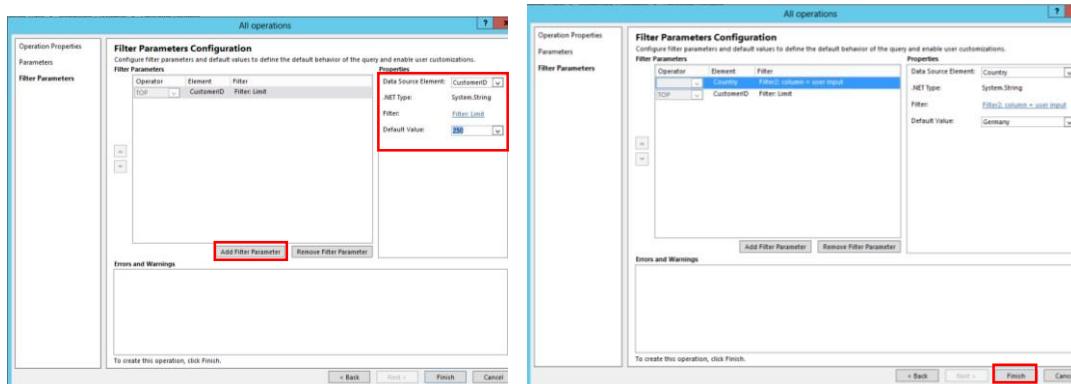
- Im anschließenden Assistenten führt ein Klick auf „Next“ zur Auswahl der Spalten, die in SP verfügbar gemacht werden sollen. Hierbei ist zu beachten, dass in den Spalteneigenschaften für die

¹⁴ Damit die erste Option „Connect with User's Identity“ funktioniert, benötigt der Account „NT Authority/IUSR“ einen Login für den DB Server und entsprechende Zugriffsrechte auf die Zieldatenbank. Aus Sicherheitsgründen ist dies jedoch nicht zu empfehlen.

SP-Zielspalten die Regeln der Datenbank zwar verschärft (z.B. das Feld ContactName wird für die externe Liste zum Pflichtfeld), jedoch nicht aufgeweicht (z.B. das Pflichtfeld CompanyName muss für die externe Liste kein Pflichtfeld mehr sein) werden können.



- Ein weiterer Klick auf „Next“ ermöglicht nun das Anlegen von Filterbedingungen (vgl. WHERE-Klausel in SQL), um die Auswahl der Abfrage einzuschränken. Dies ist optional, wird aber bei großen Datenmengen empfohlen. Eine Filterbedingung wird durch „Add Filter Parameter“ hinzugefügt und muss anschließend im rechten Bereich spezifiziert werden. Im Bsp. begrenzt ein „Limit“-Filter die Abfrage auf die 250 ersten Datensätze und es werden zusätzlich nur Kunden mit Sitz in Deutschland angezeigt (Bild links). Ein anschließender Klick auf „Finish“ verlässt den Assistenten und beendet die Konfiguration des Externen Inhaltstyps mit einer erfolgreichen Meldung (Bild rechts).



- Nun muss der Externe Inhaltstyp noch gespeichert werden. In diesem Moment sendet der SharePoint Designer die Informationen an die Business Data Connectivity Dienstanwendung.
- Bevor der gerade erstellte externe Inhaltstyp an eine externe Liste gebunden und in SharePoint genutzt werden kann, ist zunächst der Zugriff auf diesen zu berechtigen. Dazu in der Zentraladministration unter Application Management → Manage Service Applications →

auf „Business Data Connectivity Service“ klicken (Bild links). Auf der nächsten Seite ist bereits der neu angelegte externe Inhaltstyp zu sehen und muss ausgewählt und im Ribbon auf „Set Object Permissions“ geklickt werden.

- Im Dialog ist nun festzulegen, wer mit welchen Rechten auf den externen Inhaltstyp (nicht das externe Zielsystem, also die DB) zugreifen darf. Dabei sollte ein administrativer Account mit vollen Rechten eingetragen werden (Bild links) und User Groups, die „normalen“ Zugriff erhalten (Bild rechts).

- Nun kann im letzten Schritt die externe Liste erstellt werden. Dazu in der Navigationsleiste des Designers auf Lists/Libraries und anschließend im Ribbon auf „External List“ klicken, um im nächsten Schritt den zu verwendenden externen Inhaltstyp zu wählen (Bild links) und anschließend einen Namen für die externe Liste anzugeben.

- Der externe Inhaltstyp ist nun verwendbar und kann im Browser durch die eben angelegten externen Liste aufgerufen werden¹⁵. Im Bsp. ist dabei Lese- und Schreibzugriff möglich.

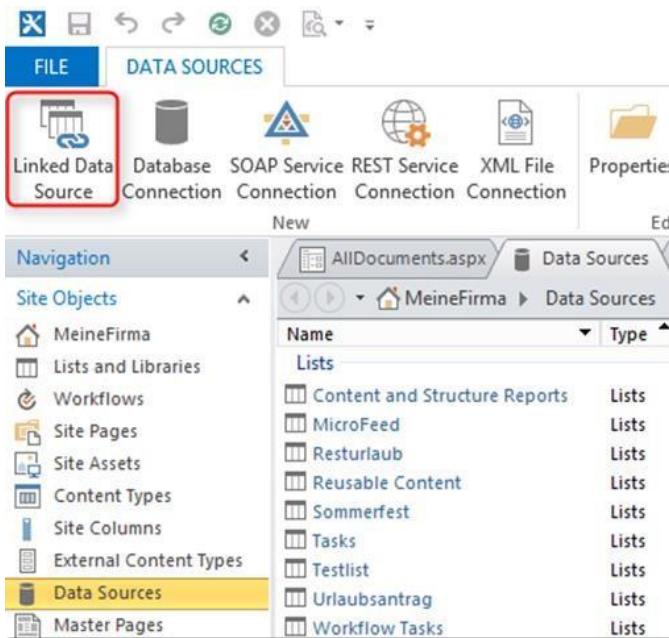
Northwind_Customers Read List											
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
Home	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Oberse Str. 57	Berlin		12209	Germany	030-0074321	030-0076545
Documents	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forsterstr. 57	Mannheim		68306	Germany	0621-08460	0621-08924
Recent	DRACD	Drachenblut Delikatessen	Sven Ottlieb	Order Administrator	Walserweg 21	Aachen		52066	Germany	0241-039123	0241-059428
Bewertungen	FRANK	Frankenversand	Peter Franken	Marketing Manager	Berliner Platz 43	München		80805	Germany	089-0877310	089-0877451
Rechnungen	KOENE	Königlich Essen	Philip Cramer	Sales Associate	Mauelstr. 90	Brandenburg		14776	Germany	0555-09876	
Site Contents	LEHMS	Lehmans Marktstand	Renate Messner	Sales Representative	Magazinweg 7	Frankfurt a.M.		60528	Germany	069-0245984	069-0245874
	MORGK	Morgenstem Gesundkost	Alexander Feuer	Marketing Assistant	Heerstr. 22	Leipzig		04179	Germany	0342-023176	
	OTTIK	Ottiles Käseladen	Henriette Pfalzheim	Owner	Mehrheimerstr. 369	Köln		50739	Germany	0221-0644327	0221-0765721
	QUICK	QUICK-Stop	Horst Kloss	Accounting Manager	Taucherstraße 10	Cunewalde		01307	Germany	0372-035188	
	TOMSP	Toms Spezialitäten	Karin Josephs	Marketing Manager	Luisenstr. 48	Münster		44087	Germany	0251-031259	0251-035695
	WANDK	Die Wandernde Kuh	Rita Müller	Sales Representative	Adenauerallee 900	Stuttgart		70563	Germany	0711-020361	0711-035428

¹⁵ Falls die Authentifizierungsmethode die Secure Store Service Dienstanwendung verwendet kann zuvor ein einmaliges Eintragen der Benutzerkennung erforderlich sein.

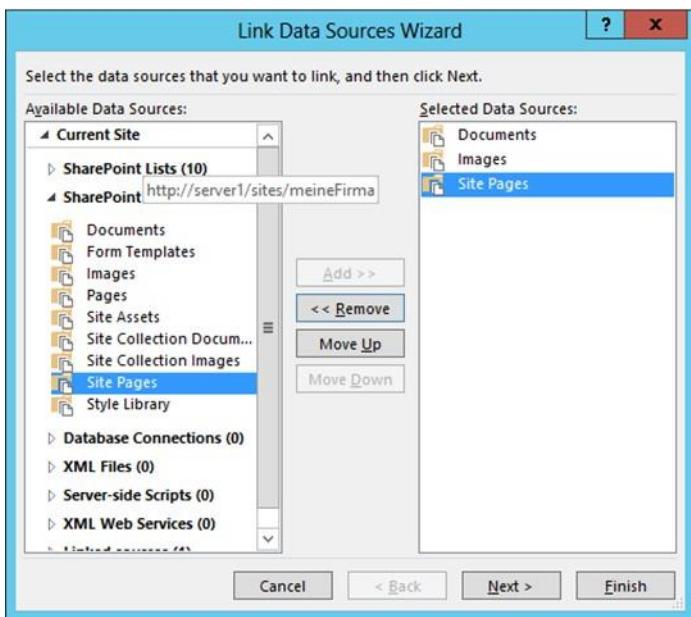
Linked Data Source für aggregierte Datenansicht

Damit sämtliche Elemente aus verschiedenen Listen in einer einzigen Datenansicht zusammen gefasst (aggregiert/merged) dargestellt werden können, wird die sog. verknüpfte Datenansicht (Linked Data Source) benötigt. Im folgenden Beispiel werden die Dokumente aus drei verschiedenen Bibliotheken in einer gemeinsamen Ansicht angezeigt.

Als erstes ist im Navigationsbereich im Punkt “Data Sources” eine neue Linked Data Source anzulegen und im Anschluss zu konfigurieren.

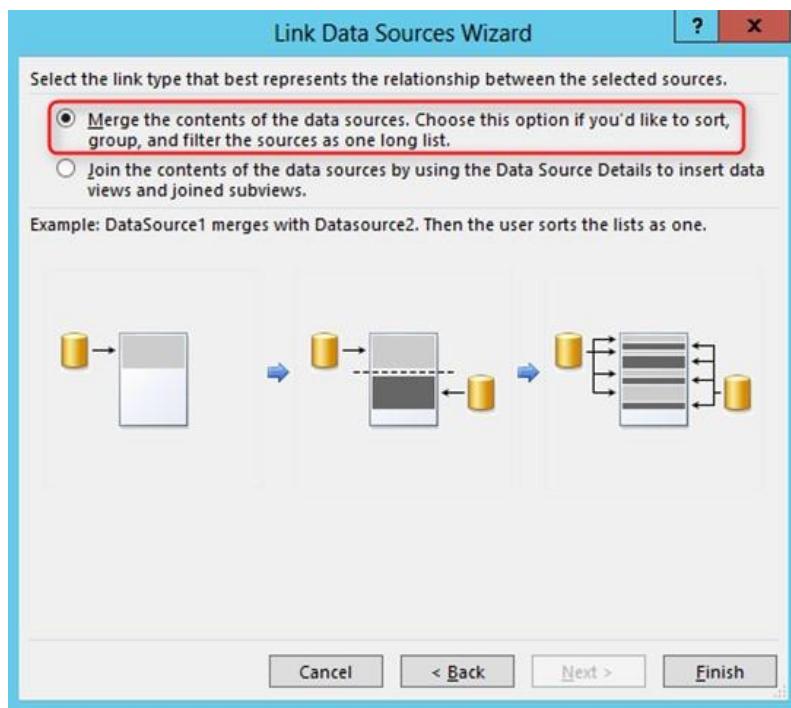


Hierfür müssen die zu verbindenden Datenquellen - im Bsp. drei verschiedene Bibliotheken - ausgewählt werden.

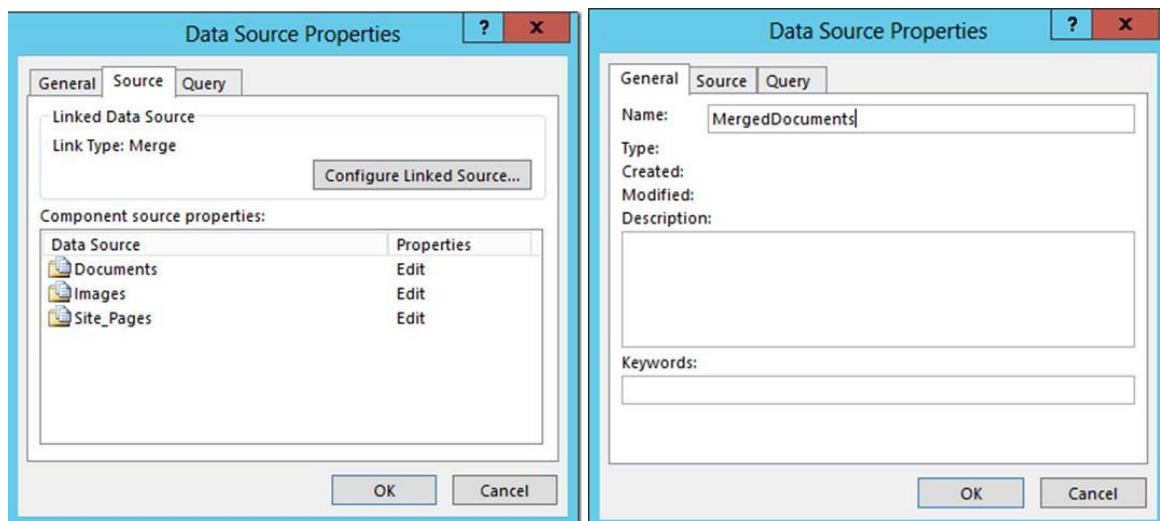


Im nächsten Fenster ist die Art der Verbindung zu wählen. Die erste Auswahlmöglichkeit “Merge” dient der Zusammenfassung mehrerer Inhalte in einer Ansicht mit nur einer Hierarchie-Ebene. Der zweite Punkt “Join” ermöglicht das Verknüpfen von Elementen mit Hilfe einer Unteransicht. So lässt sich bspw. eine 1:n-

Beziehung darstellen. In diesem Beispiel sollen die Elemente aggregiert bzw. zusammengefasst werden, darum ist "Merge" auszuwählen.



Nach dem Fertigstellen der Auswahl sind im nächsten Dialog im Bereich "Source" die zusammenzufassenden Datenquellen zu sehen. Im Reiter "General" ist anschließend, falls noch nicht geschehen, der Name der Datenansicht zu vergeben.



Die neu erstellte Datenquelle kann verwendet werden, indem sie z.B. auf einer neuen WebPart-Page im erweiterten Bearbeitungsmodus als Datenansicht hinzugefügt wird. Dazu muss der Cursor direkt zwischen die beiden unteren ZoneTemplate-Tags der WebPartPage platziert werden.

The screenshot shows the SharePoint ribbon interface with the "Advanced Mode" tab highlighted in red. A tooltip for "Edit Page Contents in Advanced Mode" is displayed, stating: "Edit this page with access to all of the page markup." It also notes: "In this mode, you can customize the page from the site definition." Below the ribbon, the SharePoint page source code is visible.

```

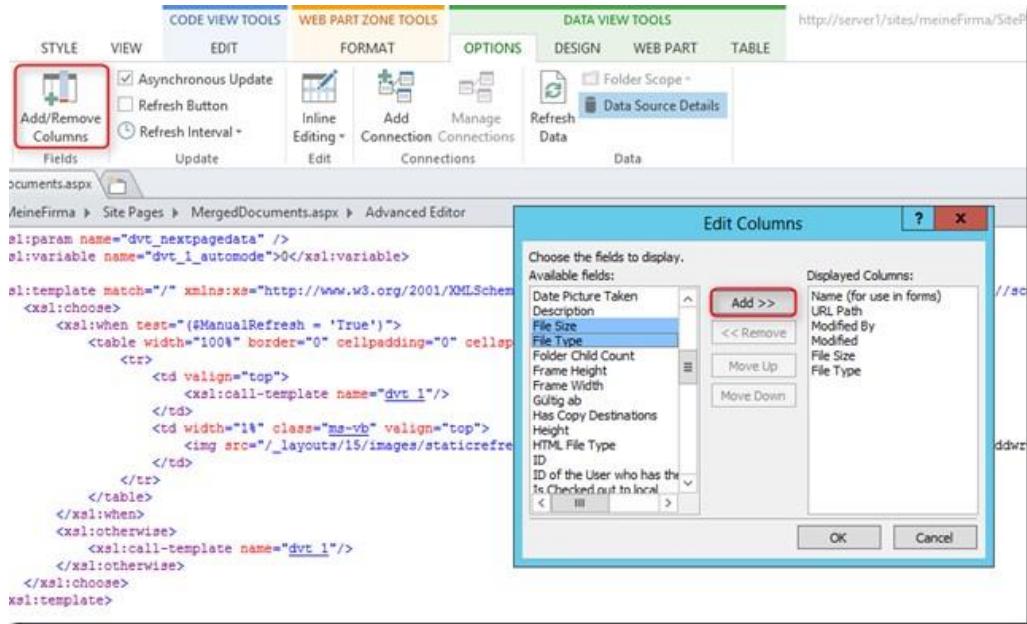
1" --%>
<url/default.master" Inherits="Microsoft.SharePoint.WebControls" Assembly="Microsoft.SharePoint.WebControls, Version=12.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c">
<asp:PageTitle runat="server">
<asp:PageName maxlength="40" runat="server"/>
</asp:PageName>
<asp:AdditionalPageHead runat="server">
<asp:SharePointPagePartDocument>
<asp:Text> charset=utf-8 />
<asp:PageTitle> SharePoint Team Web Site </asp:PageTitle>
<asp:Content>
<!--#include virtual="~/PAGEINFO.PHR" -->
<div id="TitleBar" runat="server" title="loc:TitleBar" id="TitleBar" AllowLayoutChange="false" AllowPersonalization="false" Style="display:none;">
<div>
<table class="ms-core-tableNoSpace ms-webpartPage-root" width="100%">
<tr>
<td id="invisiblEmpty" name="invisiblEmpty" valign="top" width="100%">
<asp:WebPartZone runat="server" Title="loc:FullPage" ID="FullPage" FrameType="TitleBarOnly"></asp:WebPartZone>
</td>
</tr>
</table>
</div>
<asp:ScriptBlock runat="server">if(typeof(MSOLayout_MakeInvisibleIfEmpty) == "function") {MSOLayout_MakeInvisibleIfEmpty();}</asp:ScriptBlock>
</asp:Content>

```

Nun kann die zusammengefasste Datenansicht über das Ribbon → Insert → Data View → Linked Sources hinzugefügt werden.

The screenshot shows the SharePoint ribbon with the "Data View Tools" tab selected. In the "Data View" section of the ribbon, the "Data View" button is highlighted. The "Data View" dropdown menu is open, showing options like "Related Item View", "New Item Form", "Edit Item Form", and "Display Item Form". Below the ribbon, the SharePoint navigation pane shows "Site Pages" selected. The "Data View" dropdown menu is overlaid on the page content area, which displays the page source code.

Die Datenquelle wurde nun hinzugefügt und kann bearbeitet werden. So können bspw. die standardmäßig angezeigten Spalten (Name, Url Path, Modified by, Modified) angepasst und durch weitere Felder ergänzt werden.

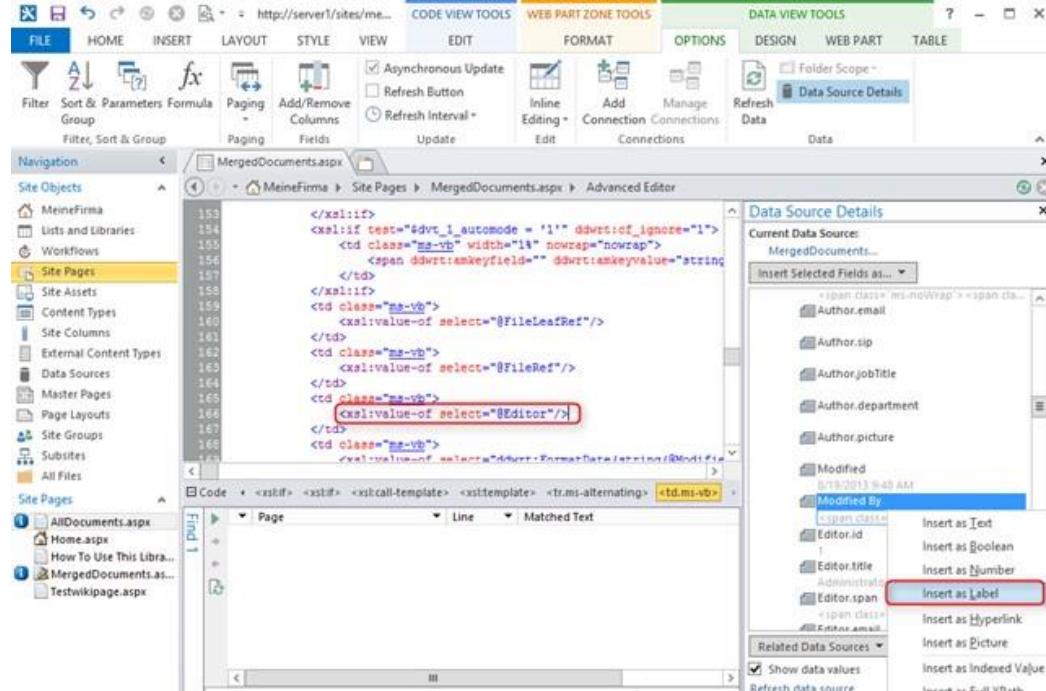


Wenn die Page zu diesem Zeitpunkt gespeichert und in der Vorschau betrachtet wird, fällt auf, dass der Wert "Modified by" nicht korrekt interpretiert sondern als plain html dargestellt wird.

Um dies zu ändern, muss die entsprechende Spalte erneut als "Label" eingefügt, und somit der bisherige Wert

`<xsl:value-of select="@Editor"/>`

ersetzt werden. Dies erfolgt im Data Source Explorer durch einen Rechtsklick auf die entsprechende Spalte – „Insert as Label“.



Da häufig noch der Hyperlink zum Dokument angezeigt werden soll, soll im Bsp. der Feldwert "Url Path" als Hyperlink zum Dokument formatiert werden. Dazu ist der Standardwert der XSL-Darstellung

`<xsl:value-of select="@FileRef"/>`

in einen Hyperlink umzuwandeln

<xsl:value-of select="@FileRef"/>

Nun kann die Page gespeichert und erneut in der Vorschau betrachtet werden.

The screenshot shows a SharePoint interface with the title 'MergedDocuments'. The left navigation bar includes 'Home', 'Coburg', 'server', 'impressum', 'printer', 'Wiki', 'EDIT LINKS', and a 'Liste' section with 'Kunden' and 'Home' selected. The main content area displays a table of files:

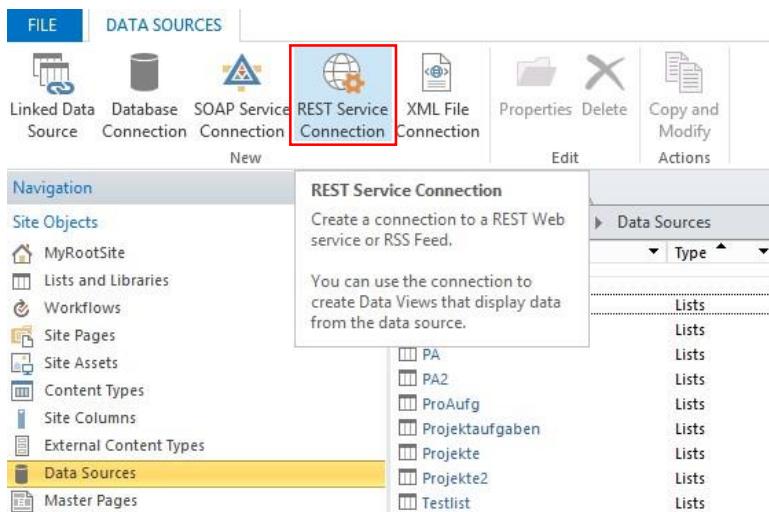
Name (for use in forms)	URL Path	Modified By	Modified	File Size	File Type
Mitarbeiterbonus1.docx	/sites/meineFirma/Shared Documents/Mitarbeiterbonus1.docx	Administrator	8/19/2013 9:48 AM	29924	docx
Mappe1.xlsx	/sites/meineFirma/Shared Documents/Mappe1.xlsx	Administrator	8/23/2013 5:30 PM	21808	xlsx
RechnungsVorlage.dotx	/sites/meineFirma/Shared Documents/RechnungsVorlage.dotx	Administrator	8/23/2013 5:33 PM	53786	dotx
Untitled_wf_binding_error.png	/sites/meineFirma/PublishingImages/Untitled_wf_binding_error.png	Administrator	8/23/2013 5:32 PM	51018	png
wind.png	/sites/meineFirma/PublishingImages/wind.png	Administrator	8/23/2013 5:32 PM	1713	png
firefox.png	/sites/meineFirma/PublishingImages/firefox.png	Administrator	8/23/2013 5:32 PM	8110	png
ie10.png	/sites/meineFirma/PublishingImages/ie10.png	Administrator	8/23/2013 5:32 PM	1940	png
apple.jpg	/sites/meineFirma/PublishingImages/apple.jpg	Administrator	8/23/2013 5:32 PM	1401	jpg
Home.aspx	/sites/meineFirma/SitePages/Home.aspx	Administrator	8/19/2013 10:20 AM	3356	aspx
How To Use This Library.aspx	/sites/meineFirma/SitePages/How To Use This Library.aspx	System Account	8/13/2012 9:49 AM	3356	aspx

At the bottom right of the table, there is a page number '1 > 10'.

REST Webservice anbinden – Erstellung eines Wetter-WebParts

Der Designer kann verschiedene Datenquellen anbinden und verarbeiten, darunter auch REST¹⁶ Webservices wie z.B. RSS-Feeds. Im folgenden Beispiel soll das Wetter über einen Webservice abgerufen und anschließend in Form eines WebParts dem SharePoint zur Verfügung gestellt werden.

- Als erstes muss die WebService-Datenverbindung erstellt werden. Im Beispiel soll das die Yahoo Weather RSS Feed¹⁷ angesprochen werden. Im Navigationsbereich muss dazu der Bereich Data Sources (Datenquellen) geöffnet werden. Anschließend erscheinen im Menüband die verfügbaren Möglichkeiten, Datenquellen anzubinden, darunter auch der auszuwählende Eintrag „REST Service Connection“.

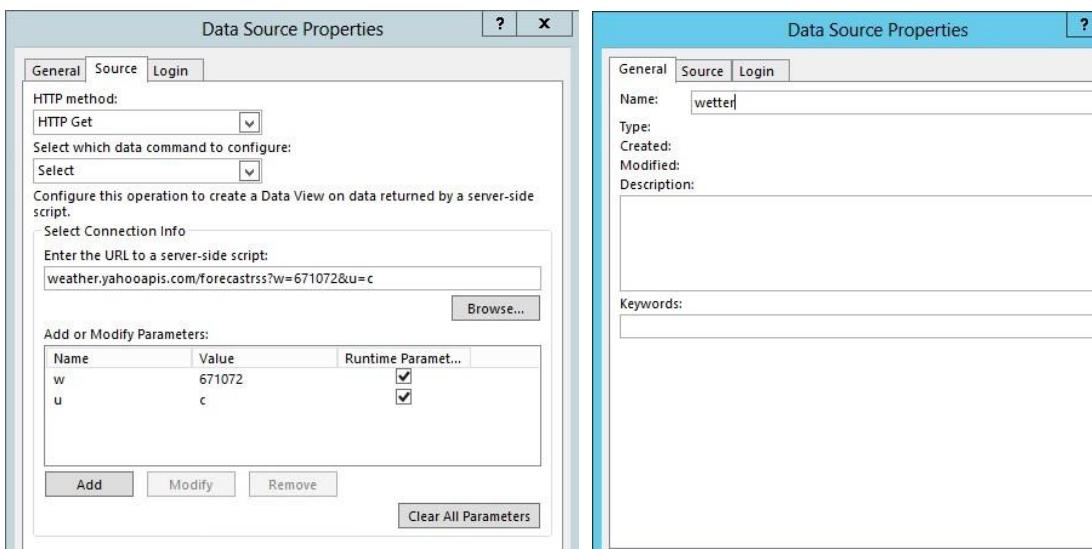


- Im darauf folgenden Dialogfenster sind Methode (GET / POST), Operation, Ziel URL und Übergabeparameter anzugeben (Bild links). Im Beispiel handelt es sich um eine GET-Abfrage. Die URL adressiert den Webservice und übergibt die Parameter an den Server. Im Bsp. Werden die Parameter w=GMXX0072¹⁸ für den Ort Leipzig und u=c für die Maßeinheit Grad Celsius übergeben. Im Reiter General kann noch der Name angepasst werden (Bild rechts).

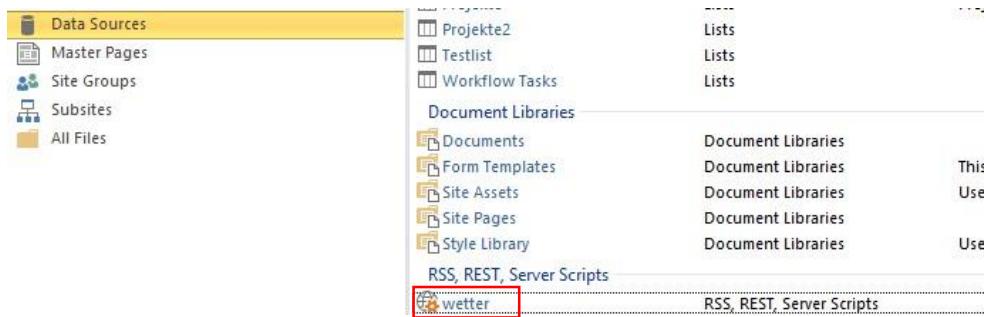
¹⁶ Representational State Transfer → Zustandsloser Webservice der über die URL Parameter übergeben kann.

¹⁷ Weitere Infos unter: <http://developer.yahoo.com/weather/>

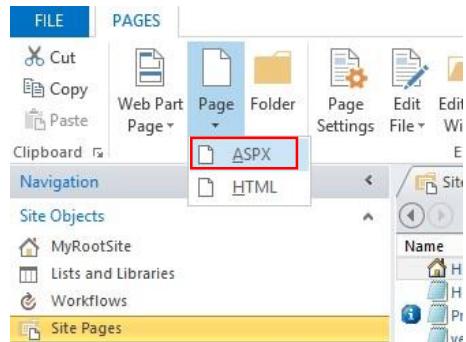
¹⁸ Die WOEID (Where On Earth Identifier) kann unter <http://woeid.rosselliot.co.nz/> ermittelt werden.



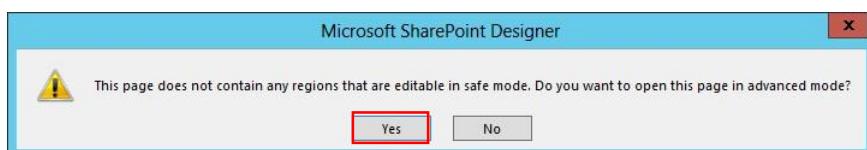
- Wenn die Webserviceverbindung erstellt wurde, taucht diese im Bereich DataSources mit auf und kann dort auch ggf. später bearbeitet werden.



- Anschließend kann die Verbindung in einer aspx-Page hinzugefügt und verwendet werden. Zunächst ist eine solche aspx-Page zu erstellen, im Navigationsbereich unter SitePages → Ribbon → New → Page → ASPX.



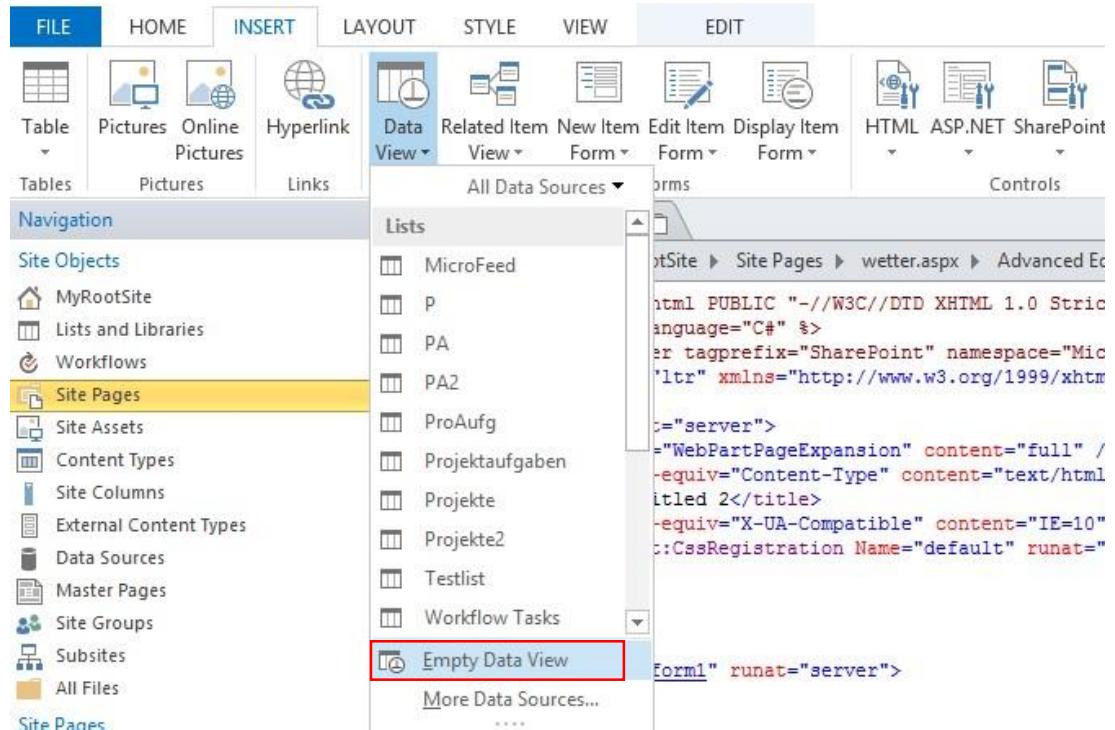
- In der Übersichtsseite der aspx-Page → Bereich Customization → führt der Link „Edit file“ dann in den Bearbeitungsmodus der neu angelegten Page. Anschließend muss noch in den erweiterten Bearbeitungsmodus gewechselt werden, indem die folgende Meldung mit YES bestätigt wird.



□

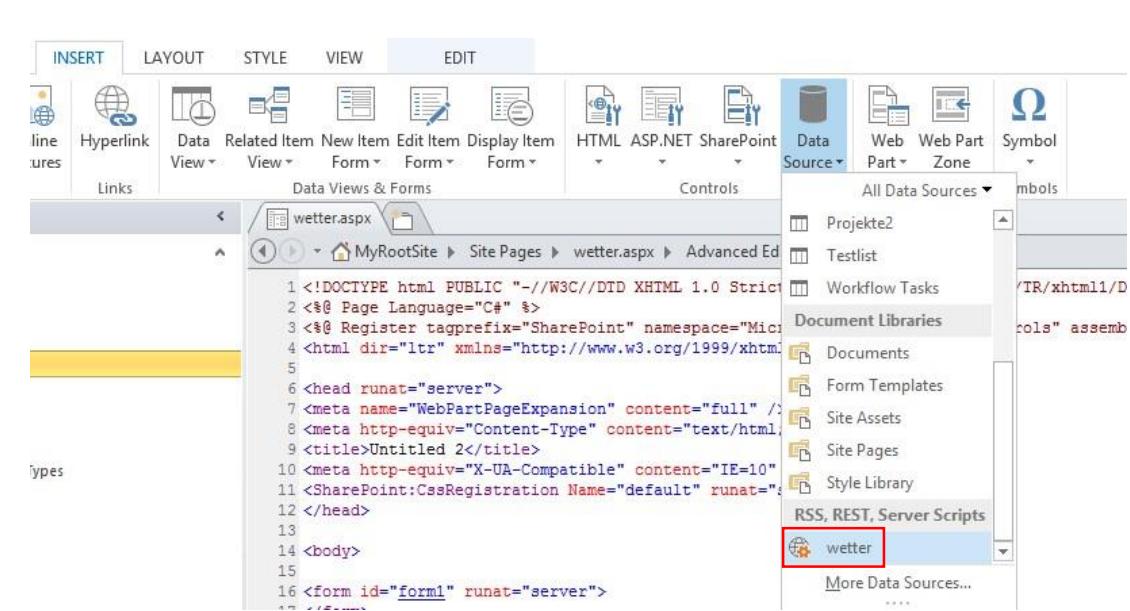
Als erstes wird der Page eine leere Datenansicht hinzugefügt, welcher im nächsten Schritt dann die Datenquelle zugewiesen wird.

Im Ribbon → Insert → Data View → auf „Empty Data View“ klicken, um die Ansicht einzufügen.



- Nun muss dieser leeren Datenansicht noch eine Datenverbindung zum Webservice hinzugefügt werden.

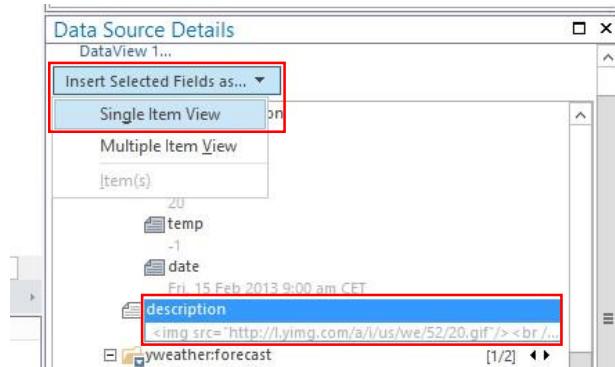
Dazu den Cursor im Code innerhalb der <DataSources> Tags positionieren. Danach im Ribbon → Insert → Data Source → auf die Datenquelle des Webservice (im Bsp. „Wetter“) klicken, um diese hinzuzufügen.



Nun befindet sich eine leere Datenansicht mit gebundener Datenquelle auf der aspx-Page.

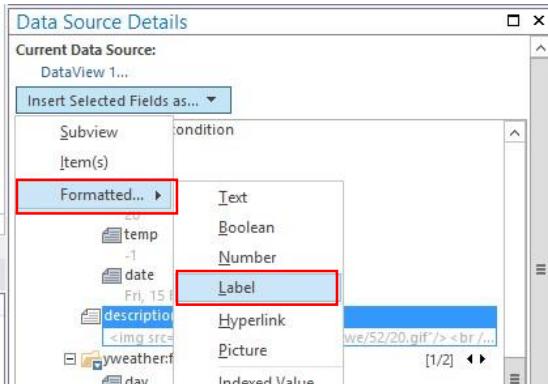
Jetzt muss noch das anzugebende Feld eingefügt werden (im Bsp. „description“).

Dazu im Bereich „Data Source Details“ das gewünschte Feld anklicken und den Button „Insert Selected Fields as...“ → „Single Item View“ anklicken.



- Nun wird automatisch ein XSL-Stylesheet eingefügt, der das Feld inkl. html-Gerüst auf der Page einfügt. Allerdings wird es als Text und nicht als html-Statement interpretiert.

```
<br /><b>Current Conditions:</b><br /> Mist, -1 C<BR /> <BR /><b>Forecast:</b><br /> Fri - Partly Cloudy. High: 2 Low: -1<br /> <br /> <a href="http://us.rd.yahoo.com/dailynews/rss/weather/Leipzig__DE/*http://weather.yahoo.com/forecast/GMXX0072_c.html">Full Forecast at Yahoo! Weather</a><br /><br /> (provided by <a href="http://www.weather.com">The Weather Channel</a>)<br />
```
- Damit der Code richtig interpretiert werden kann, muss er in Form eines Label-Steuerelements dargestellt werden. Dies erfolgt, indem mit dem Cursor auf den Code der bisherigen Darstellung `<xsl:value-of select="..//description" />` geklickt und im Bereich „Data Source Details“ erneut auf das einzufügende Feld klicken, diesmal steht jedoch die Auswahl „Insert Selected Fields as...“ → „Formatted...“ → „Label“ zur Verfügung.



- Nun sieht die Page in der Browervorschau wie folgt aus:



Current Conditions:
Mist, 1 C

description:	Forecast:
	Fri - Partly Cloudy. High: 2 Low: -2 Sat - Mostly Cloudy. High: 2 Low: -1
Full Forecast at Yahoo! Weather	
(provided by The Weather Channel)	

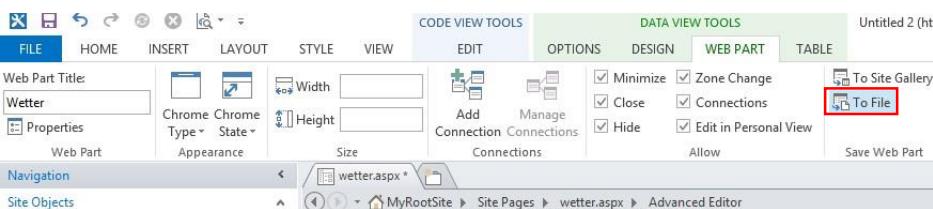
Um die Linke Spalte mit dem fett gedruckten Text „description“ zu entfernen, muss der in folgender Grafik markierte Codeblock entfernt werden. Dieser <td>-Tag steht für die linke Spalte (wo der fett gedruckte Schriftzug angezeigt wird).

```

112      <td>
113          <table border="0" cellspacing="0" width="100%">
114              <tr>
115                  <td width="25%" class="ms-vb">
116                      <b><description:</b>
117                  </td>
118                  <td width="75%" class="ms-vb">
119                      <asp:Label runat="server" id="ff1($Pos)" text="{description}" _>
120                  </td>
121              </tr>
122      </td>

```

- Zu guter letzt kann die Wetterinfo separat als WebPart in der WebPart-Bibliothek auf dem SharePoint gespeichert oder auch als *.webpart-Datei exportiert werden. Zweiter Weg bietet sich an, wenn das WebPart als Datei direkt portabel zur Verfügung gestellt werden soll. Der Export erfolgt über das Ribbon → Data View Tools → Web Part → To File. Zuvor kann noch der Name des WebParts (im Bsp. „Wetter“) und weitere Optionen angepasst werden.



- Die fertige .webpart-Datei kann nun von der SharePoint Weboberfläche auf hochgeladen und anschließend verwendet werden.

□

Dies erfolgt indem über die SiteSettings → im Bereich „Web Designer Galleries“ → auf „Web parts“ geklickt (Bild links) und dort im Ribbon mittels „Upload Document“ die WebPart-Datei hochgeladen (Bild rechts). Anschließend sind noch Name und ggf. Titel und Gruppe des WebParts anzugeben.

MyRootSite How To Use This Library Lpz EDIT LINKS

Site Settings

The screenshot shows the SharePoint Site Settings ribbon. On the left, under 'Web Designer Galleries', the 'Web parts' link is highlighted with a red box. On the right, the ribbon tabs are BROWSE, FILES (selected), and LIBRARY. Under the FILES tab, the 'Upload Document' button is highlighted with a red box.

Users and Permissions
People and groups
Site permissions
Site collection administrators
Site app permissions

Look and Feel
Title, description, and lo
Quick launch
Top link bar
Tree view
Change the look

Web Designer Galleries
Site columns
Site content types
Web parts
List templates
Master pages
Solutions
Themes
Composed looks

Site Actions
Manage site features
Save site as template
Enable search configura
Reset to site definition
Delete this site

SharePoint

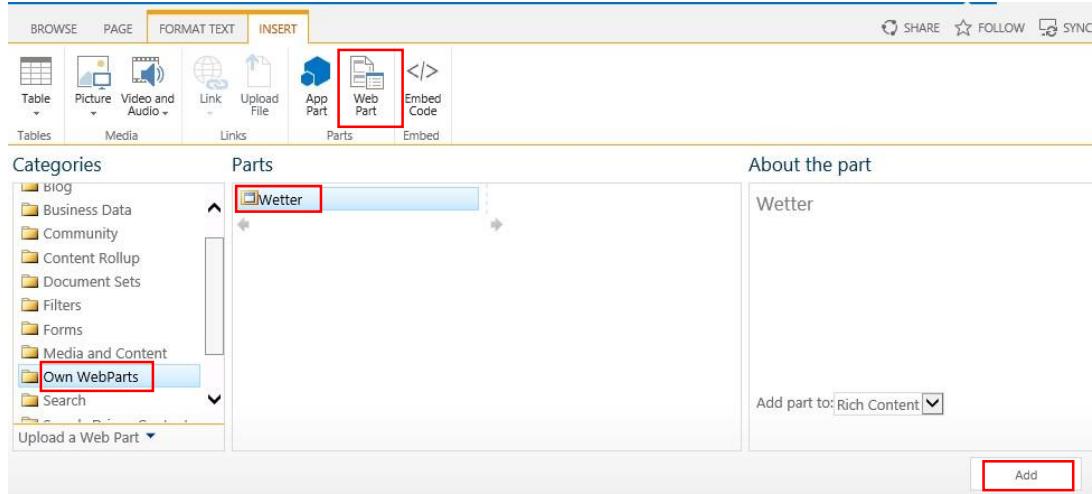
BROWSE FILES LIBRARY

New Document Upload Document New Folder

Check Out Check In Edit Document Discard Check Out Open & Check Out

- Schlussendlich kann das WebPart im „Edit Page“-Modus einer beliebigen WebsiteSeite hinzugefügt werden. Dazu die Page (im Bsp. Auf der Homepage.aspx) bearbeiten, im Ribbon → Insert → Web Part → Kategorie des WebParts → das WebPart (im Bsp. „Wetter“) mit

einem Klick auf „Add“ einfügen und anschließend den Seitenbearbeitungsmodus beenden.



- Das fertig eingebundene WebPart sollte nun wie auf der folgenden Abbildung angezeigt werden. Es kann entsprechend auf jeder beliebigen Seite eingefügt werden.

MyRootSite How To Use This Library Lpz EDIT LINKS

MyRootSite

Get started with your site REMOVE THIS

HAZE

Current Conditions:
Haze, 2 C

Forecast:
Fri - Partly Cloudy. High: 2 Low: -2
Sat - Mostly Cloudy. High: 2 Low: -1

[Full Forecast at Yahoo! Weather](#)

(provided by The Weather Channel)

Newsfeed

Start a conversation

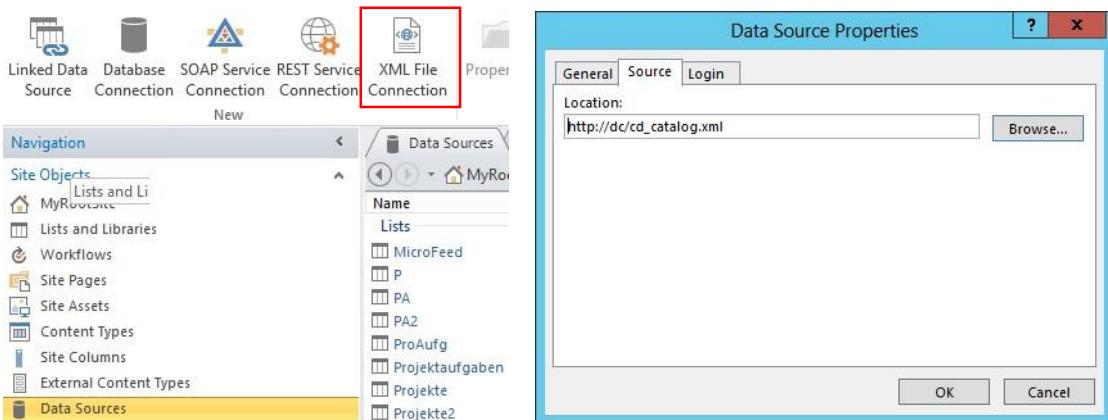
It's pretty quiet here. [Invite](#) more people to the site, or [start](#) a conversation.

XML-Datei anbinden

Mit dem SharePoint Designer ist es möglich, verschiedene Datenquellen anzubinden. So auch XML, den quasi-Standard unter den Datenaustauschformaten. In dieser Anleitung soll über eine neu erstellte WebsiteSeite (WebPart-Page) lesend und schreibend auf eine vorhandene XML-Datei (im

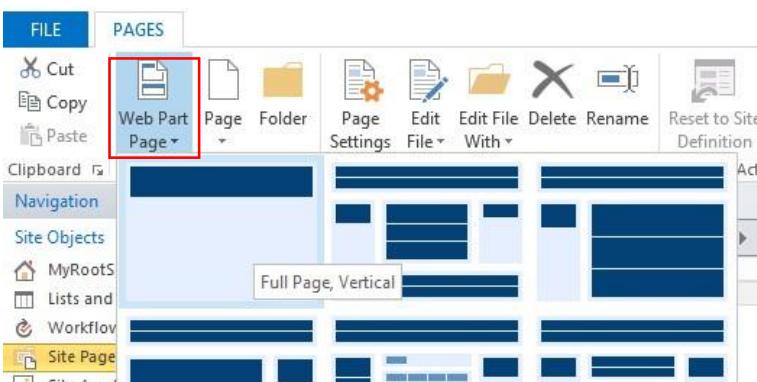
Bsp. CD-Katalog.xml) zugegriffen werden. Dazu muss zunächst die Datenquelle definiert und diese anschließend als Datenansicht in die gewünschte Page eingebunden werden.

- Als erstes muss die XML-Datei im Designer als Datenquelle hinzugefügt werden.
- Dies erfolgt im Navigationsbericht des SharePoint Designers → Data Sources → Ribbon → Neu → XML File Connection (Bild links). Anschließend die Datei auswählen und nach SharePoint importieren¹⁹ lassen (Bild rechts).



Tipp: Alternativ kann die Datei auch direkt in die Bibliothek WebsiteObjekte (Site Assets) gelegt werden. In diesem Fall würde der Designer die XML-Datenquelle automatisch finden und hinzufügen.

- Im zweiten Schritt ist eine Neue WebPart-Page zu erstellen.
Im Navigationsbereich unter Site Pages → Ribbon → New → Web Part Page die Seite anlegen und anschließend umbenennen.



¹⁹ Sollte es sich um eine lokale Datei handeln, weißt der Designer darauf hin, dass die Datei in die Website importiert werden muss. Wird dies mit OK bestätigt, erscheint die Datei anschließend im root-Verzeichnis der Website (im Navigationsbereich des Designers unter „All Files“ zu finden).

- Als nächstes ist eine leere Datenansicht in die WebPart-Page einzufügen. Dazu den Cursor in den <ZoneTemplate>-Tag innerhalb des <table>-Tags am Ende der Page platzieren und in den Advanced Mode²⁰ (erweiterten Bearbeitungsmodus) wechseln.

```

39  ``<table class="ms-core-tableNoSpace ms-webpartPage-root" width="100%">
40      <tr>
41          <td id="_invisibleIfEmpty" name="_invisibleIfEmpty" valign="top" width="100%">
42              <WebPartPages:WebPartZone runat="server" Title="loc:FullPage" ID="FullPage" FrameType="TitleBarOnly"><ZoneTemplate></ZoneTemplate></Web
43          </tr>
44      </table>
45  </asp:Content>
46

```

- Nun im Ribbon → Insert → Data View → die XML-Datenverbindung (im Bsp. Catalog) auswählen, um diese hinzuzufügen (Bild links). Nun lässt sich bereits lesend auf die XMLDatenquelle zugreifen (Bild rechts).

The screenshot shows the SharePoint ribbon with the 'Data View' button highlighted in the 'Insert' tab. Below the ribbon, the 'Navigation' pane is open, showing various site objects like Site Pages, Site Assets, and Content Types. On the right side, a preview of an XML file named 'cd_catalog.xml' is displayed, showing a list of songs with columns for TITLE and ARTIST. A red box highlights the 'cd_catalog.xml' item in the navigation pane.

TITLE	ARTIST
Empire Burlesque	Bob Dylan
Hide your heart	Bonnie Tyler
Greatest Hits	Dolly Parton
Still got the blues	Gary Moore
Eros	Eros Ramazzotti
One night only	Bee Gees
Sylvias Mother	Dr.Hook
Maggie May	Rod Stewart
Romanza	Andrea Bocelli
When a man loves a woman	Percy Sledge

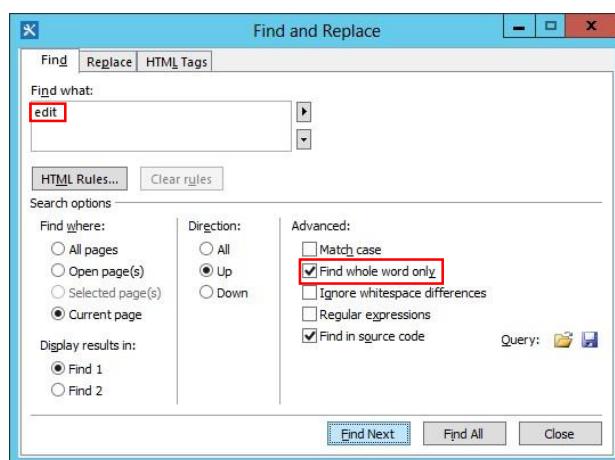
- Damit die Datenquelle auch bearbeitet werden kann, muss diese Funktionalität noch aktiviert werden, dies geschieht über die Inlinebearbeitung: Tabelle Markieren → Ribbon → Tools für die Datenanzeige → Bearbeiten → Inlinebearbeitung → „Links ,Element bearbeiten‘ anzeigen“ und „Links ,Element einfügen‘ anzeigen“ aktivieren (ggf. auch „Links ,Element löschen‘ anzeigen“).

The screenshot shows the SharePoint ribbon with the 'FORMAT' tab selected. The 'Inline Editing' dropdown is open, showing options: 'Show Edit Item Links', 'Show Insert Item Link', and 'Show Delete Item Links'. A red box highlights the 'Inline Editing' button. Below the ribbon, the code view tools are visible, showing the XML code for the page. A red box highlights the 'xml.aspx' file in the code editor.

²⁰ Ribbon → Home → Bereich Editing → „Advanced Mode“

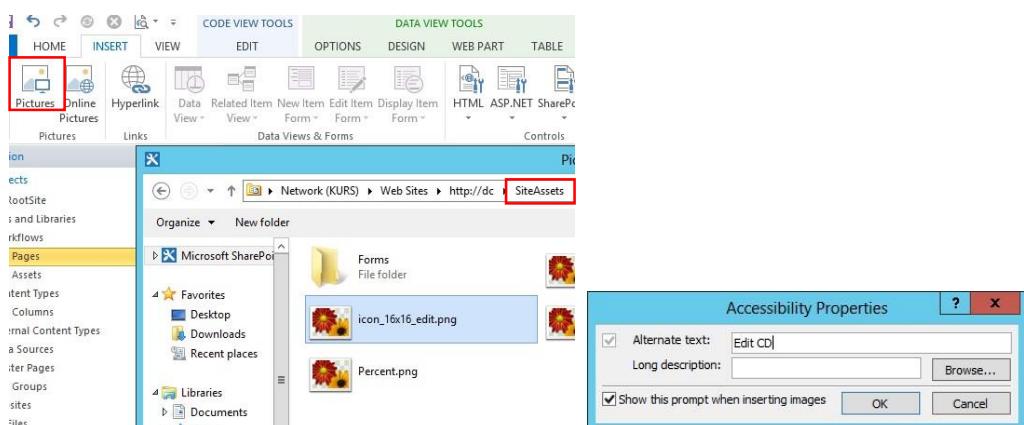
Vorsicht: Die Option „Links „Element löschen“ anzeigen“ ermöglicht es, Datensätze ohne Nachfrage endgültig aus der XML-Datei zu löschen!

- Ggf. noch die Links „Bearbeiten“, „Löschen“ und „Einfügen“ durch Icons ersetzen. Diese sollten vorher in die Bibliothek WebsiteObjekte verschoben und im Anschluss, wie im Folgenden für das Beispiel der „Bearbeiten“-Funktion beschrieben, eingebunden werden:
 - Zunächst muss die Stelle, an der das Icon eingefügt werden soll, lokalisiert werden. Dies geschieht mittels der Suche im Quellcode (STRG + F) nach dem Stichwort „edit“ und der Option „Find whole word only“.



- Das Suchergebnis, bei dem das markierte Wort „edit“ innerhalb eines <a href>-Tags steht, ist bereits die Stelle, an der das Icon eingefügt werden soll. Nun muss der Cursor noch am Anfang der Zeile, also vor dem <a href>-Tag, platziert werden.

Nun kann die Grafik eingefügt werden. Im Ribbon → Insert → Pictures anklicken, zum Ablageverzeichnis des Bildes (SiteAssets) navigieren und das Bild öffnen (Bild links). Im Anschluss noch einen Alternativtext für das Bild eintragen (optional, Bild rechts).



- Es wurde folgendes Codefragment hinzugefügt: Dieses ist im folgenden Schritt

zu markieren, auszuschneiden und damit das in der Suche gefundene Wort „Edit“ zu überschreiben²¹. Das Codefragment ist also in den Hyperlink-Tag (<a href>) zu verschieben. Der Code für die Bearbeiten-Funktion sollte nun so aussehen:

```
<a href="javascript:
{ddwrt:GenFireServerEvent(concat('__cancel;dvt_1_form_editkey={',$Key
Value,'}')})"></a>
```

- Optional: Die Edit- und Delete-Symbole werden nun untereinander angezeigt, statt nebeneinander angezeigt. Dies lässt sich beheben, indem die aufeinanderfolgenden <tr>-Tags (in folgender Grafik blau markierter Bereich) entfernt werden:

```
<xsl:otherwise><tr>
<td class="ms-vb" width="1%" nowrap="nowrap"><a href="1&
</td></tr><tr><td class="ms-vb" width="1%" nowrap="nowr
</xsl:choose>
..
```

- Damit Elemente nicht direkt nach anklicken der „Löschen“-Schaltfläche entfernt werden, kann die Lösch-Funktion in ein Javascript-Dialog eingebunden werden, welcher vor dem Löschen bestätigt werden muss²². Dazu den im Folgenden Beispiel den gelb hinterlegten Code in den <a href>-Tag der Löschen-Schaltfläche einfügen.

```
<a href="javascript: if(confirm('Möchten Sie das Element wirklich
löschen?')){ddwrt:GenFireServerEvent(concat('__cancel;__delete={'+$Ke
yField, '='+$KeyValue, '}';__commit'))}">
```

- Im Anschluss kann die Datei gespeichert und in der Browervorschau betrachtet werden.

Die aspx-Seite sollte nun in etwa so aussehen:

TITLE	ARTIST	COUNTRY	COMPANY	PRICE	YEAR
Greatest Hits	Dolly Parton	USA	RCA	9.90	1982
Still got the blues	Gary Moore	UK	Virgin records	10.20	1990
Eros	Eros Ramazzotti	EU	BMG	9.90	1997
One night only	Bee Gees	UK	Polydor	10.90	1998
Sylvias Mother	Dr.Hook	UK	CBS	8.10	1973
Maggie May	Rod Stewart	UK	Pickwick	8.50	1990
Romanza	Andrea Bocelli	EU	Polydor	10.80	1996
When a man loves a woman	Percy Sledge	USA	Atlantic	8.70	1987
Black angel	Savage Rose	EU	Mega	10.90	1995
1999 Grammy Nominees	Many	USA	Grammy	10.20	1999
Neues Element hinzufügen					

1 - 10 ▶

²¹ Achtung: Dieser Umweg ist leider notwendig, da der Designer andernfalls den Code der einzufügenden Grafik bei direkter, korrekter Cursorplatzierung (markiertes Wort „Edit“) falsch positioniert.

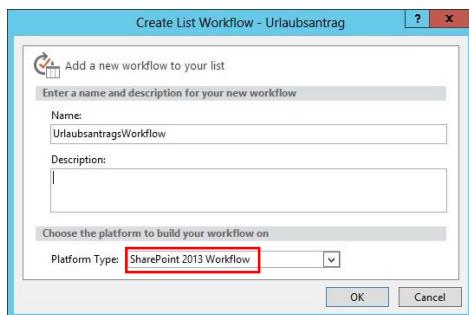
²² Vorsicht: Wenn das Element gelöscht wurde, ist es nicht wiederherstellbar, da es sich ja nicht um ein SharePoint-Objekt, sondern lediglich um einen Texteintrag in einer XML-Datei handelt.

Listenworkflow Urlaubsantrag mit Stages und AppStep

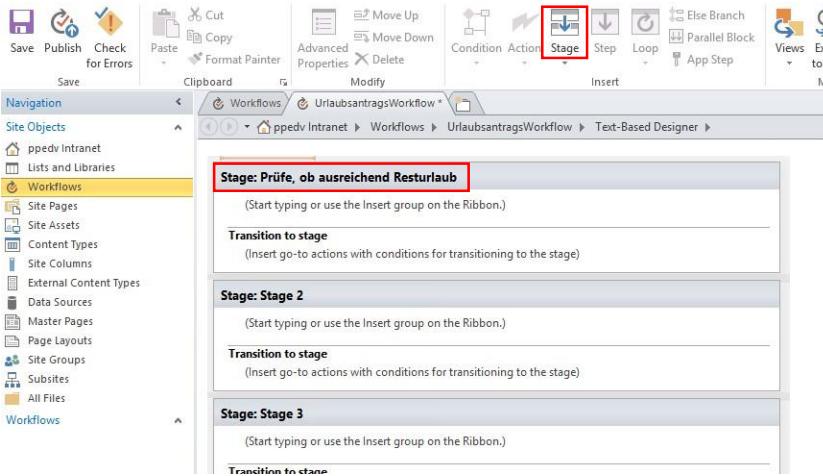
Eine nützliche Neuerung in der neuen SharePoint 2013 Workflow Engine sind die Stufen (Stages), welche als Container für mehrere Workflowaktionen, im Gegensatz zu den bereits bekannten Schritten (Steps) jedoch einen „Go to“-Ausgang (Sprungmöglichkeit in eine andere Stufe) besitzen. Eine weitere Neuerung der 2013er Workflow Engine ist der AppStep, welcher im Wesentlichen dem bereits aus 2010 bekannten Identitätswechselschritt entspricht. Der AppStep bewirkt, dass der Workflow den Benutzerkontext verlässt und somit auf Bereiche zugreifen kann, auf die der Nutzer (Workflow-Initiator) normalerweise keinen oder nur beschränkten Zugriff hat.

Im folgenden Beispiel wird ein oft verwendetes Szenario betrachtet - der Urlaubsantragsprozess. Im zweiten Teil der Übung soll der Workflow so angepasst werden, dass er auch für Nutzer, die keinen direkten Zugriff auf die Resturlaubstage-Liste haben, korrekt durchläuft. Diesen Zugriff übernimmt der Workflow durch Verwendung des AppSteps (zweiter Teil).

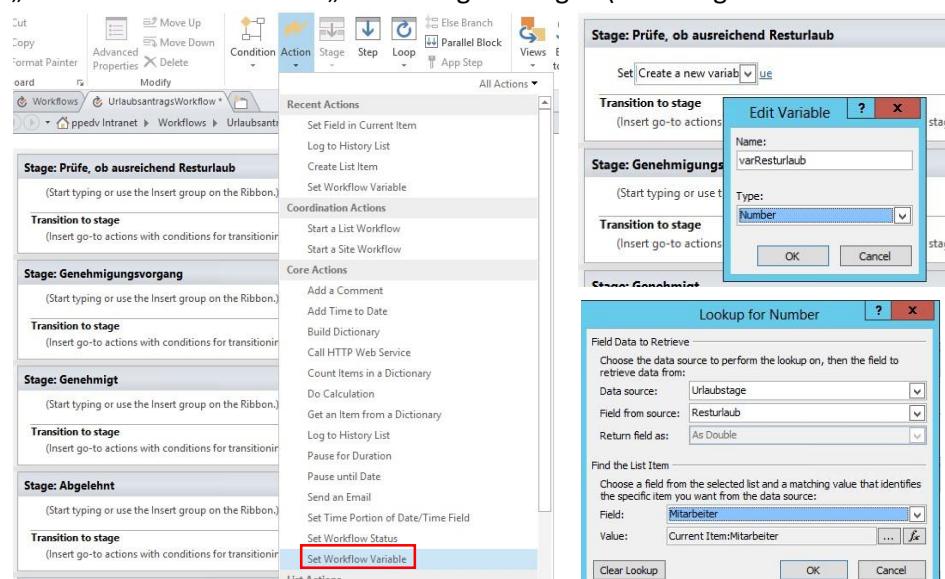
- Voraussetzung:
 - Liste „Urlaubstage“ (Titel, Mitarbeiter, Resturlaub) existiert und enthält Einträge.
 - Liste „Urlaubsantrag“ (Titel, Mitarbeiter, BeantragteUrlaubstage) existiert.
- Zunächst wird ein neuer Listenworkflow für die Liste „Urlaubsantrag“ erstellt. Als Plattform wird die neue SharePoint 2013 Workflow Engine gewählt.



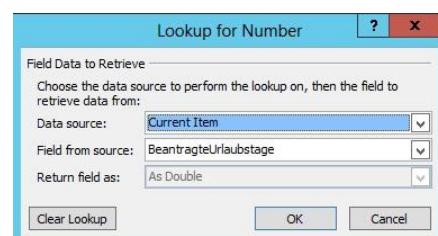
- Anschließend ist die Logik des Workflows zu implementieren. Diese lässt sich in fünf Stufen (Stages), welche einem Workflow-Status gleichzusetzen sind, untergliedern. Über den Ribboneintrag „Add Stage“ können diese hinzugefügt und wie folgt umbenannt werden:
 - Prüfe, ob ausreichend Resturlaub
 - Genehmigungsvorgang
 - Genehmigt
 - Abgelehnt
 - Nicht genügend Resturlaub



- Im ersten Stage („Prüfe, ob ausreichend Resturlaub“) sind folgende Aktionen einzufügen:
 - Action → Set Workflow Variable → „varResturlaub“ (Number) mit dem Feld „Resturlaub“ aus der Liste „Urlaubstage“ belegen (Matching über den Mitarbeiter).

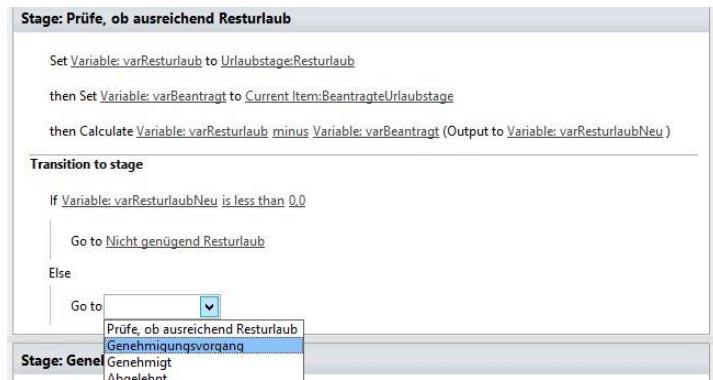


- Action → Set Workflow Variable → „varBeantragt“ (Number) mit dem Feld „BeantragteUrlaubstage“ aus dem aktuellen Element belegen.

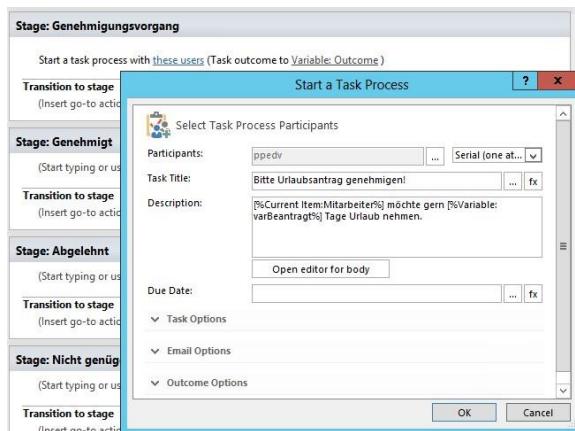


- Action → Do a Calculation → varResturlaub Minus varBeantragt → output in neue Variable „varResturlaubNeu“ ○ Am Ende des Stages soll geprüft werden, ob nach der Berechnung noch Resturlaub zur Verfügung steht (Condition → If any value equals value → Value 1: „varResturlaubNeu“ → Operator: „is less than“ → Value 2: „0“).

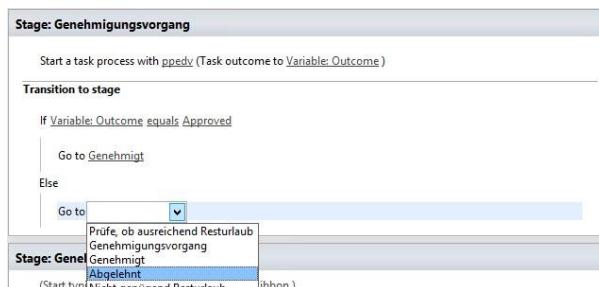
Falls nicht, soll in Stage „Nicht genügend Resturlaub“ gewechselt werden (Action → Goto Stage). Wenn hingegen noch Urlaubstage übrig sind, soll in den Stage „Genehmigungsvorgang“ gewechselt werden (Action → Goto Stage).



- Im zweiten Stage („Genehmigungsvorgang“) sind folgende Aktionen einzufügen:
 - Actions → Task Actions → Start a task process. Hiermit wird ein Genehmigungsvorgang für einen (ggf. auch mehrere) Benutzer gestartet. Innerhalb des Textkörpers können auch Referenzen auf die Attribute eingefügt werden.

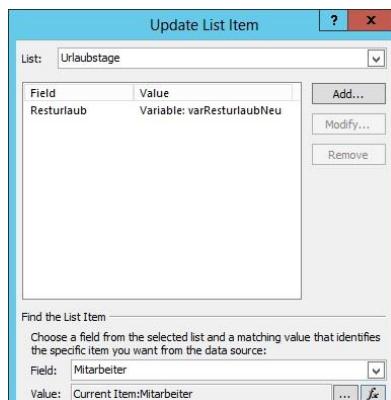


- Im „Transition to stage“-Bereich ist eine Bedingung einzufügen, die den Genehmigungsstatus überprüft.
Wenn genehmigt (Condition → If any value equals value → Value 1: Workflowvariable „Outcome“ → Operator: „is equal to“ → Approved), soll in den Stage „Genehmigt“, andernfalls in Stage „Abgelehnt“ gewechselt werden.

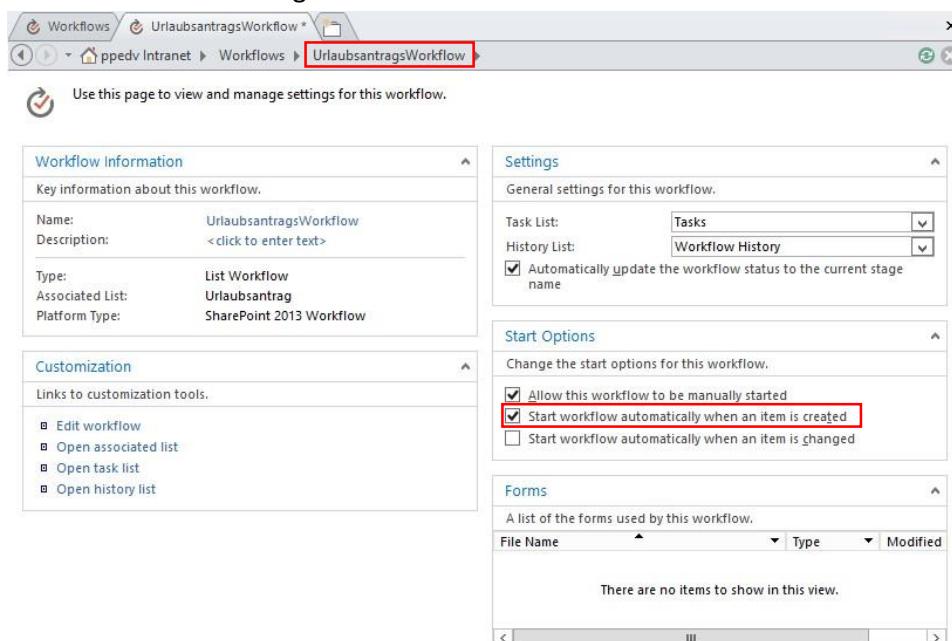


- Im dritten Stage (“Genehmigt”) sollen folgende Aktionen durchgeführt werden:

- Der Resturlaub muss in der Liste Urlaubstage aktualisiert werden (Actions → List Actions → Update list item → Liste Urlaubstage → Feld Resturlaub auf „varResturlaubNeu“ setzen → Matching über den Mitarbeiter.



- Benachrichtigung an den Ersteller des Urlaubsantrages (Actions → Send an Email).
 - Im „Transition to stage“-Bereich wird das Workflow-Ende als Ziel ausgewählt.
- Im vierten Stage („Abgelehnt“) soll eine Email an den Ersteller des Urlaubsantrages gesendet werden mit der Information, dass dieser abgelehnt wurde (Actions → Send an Email) und anschließend im „Transition to stage“-Bereich zum Workflow-Ende verwiesen werden.
- Im fünften Stage („Nicht genügend Resturlaub“) sind analog zum Vierten eine Email an den Antragsteller sowie Übergang zum Workflowende zu implementieren.
- Nachdem die Workflowlogik implementiert wurde, ist im Anschluss auf die Workflowübersichtsseite zu wechseln und dort noch den automatischen Startvorgang des Workflows bei Erstellung eines Listenelements zu aktivieren.



- Zuletzt muss der Workflow gespeichert und veröffentlicht werden.

Anschließend ist der Workflow direkt an die Zielliste gebunden und kann gestartet werden, indem ein neues Listenelement (also ein neuer Urlaubsantrag) erstellt wird. In der Listenansicht taucht eine neue Spalte auf, welche die aktuelle Stufe anzeigt, in der sich der Workflow im Moment befindet. Durch Klicken auf dem Hyperlink der Stufe öffnet sich die Workflowdetailansicht, wo Aufgaben, Fehlermeldungen und die Einträge der Workflowverlaufsliste einsehbar sind.

Erweiterung des Listenworkflow durch Implementierung eines AppStep

Der bisherige Workflow lässt sich solange problemlos ausführen, wie Workflow-Initiator Schreibrechte auf beide Listen besitzt. Wenn jedoch ein Nutzer den Workflow startet, der über keine Schreibrechte auf die Urlaubstage-Liste verfügt, kommt es beim Aktualisieren der Resturlaubstage zu einem Authentifizierungsfehler und somit zum Abbruch des Workflows.

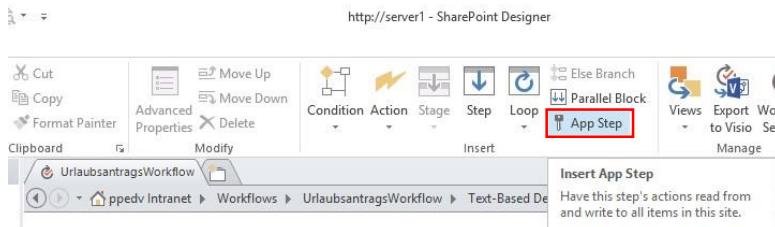
The screenshot shows a SharePoint page titled 'Workflow Status: UrlaubsantragsWorkflow'. It displays workflow information for an item initiated by 'max muster.' on 5/31/2013 at 11:56 AM. The internal status is shown as 'Canceled' (highlighted with a red box). A tooltip window is overlaid on this status, containing a large amount of JSON-formatted error data. The error details include RequestorId, System.ApplicationException, and various HTTP headers and server details, indicating a 401 Unauthorized error.

Dies lässt sich durch die Verwendung eines AppSteps vermeiden. Hierzu wird nun ein vereinfachter Workflow verwendet, um den Fokus auf den AppStep zu lenken. Die folgende Vorgehensweise lässt sich natürlich auch auf andere Workflows anwenden.

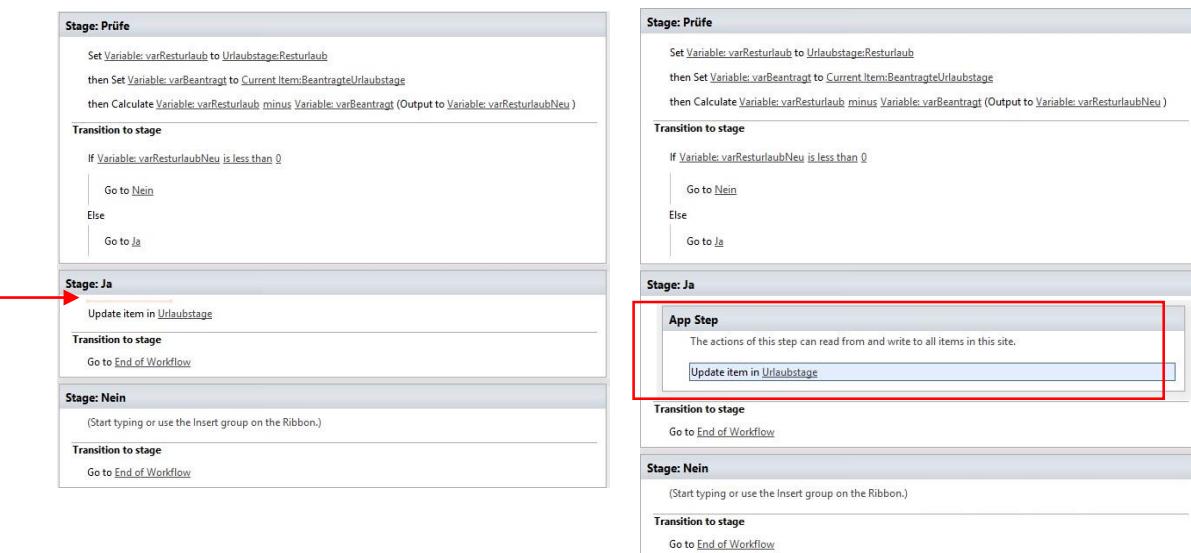
- Bevor der AppStep verwendet werden kann, muss erst das Website-Feature „Workflows can use app permissions“ in den Websiteeinstellungen aktiviert werden.

The screenshot shows the 'Site Settings' page under 'General Site Settings'. It lists several features with their descriptions and activation/deactivation buttons. The fourth feature, 'Workflows can use app permissions', has its 'Activate' button highlighted with a red box.

- Danach den Designer am besten neu starten, mit der Website verbinden und den Workflow öffnen. Nun sollte im Ribbon-Bereich „Insert“ der Eintrag „AppStep“ verfügbar sein.



- Zum Einfügen den Cursor an die gewünschte Stelle platzieren und auf den „AppStep“-Eintrag im Ribbon klicken und anschließend die gewünschte(n) Aktion(en) in den AppStep zu verschieben. Im Beispiel ist dies die Aktion „Update List Item“ (Aktuelle Resturlaubstage in Liste zurückschreiben). Folgende Grafik zeigt den Beispielworkflow vor (links) und nach (rechts) dem Einfügen des AppSteps.



- Am Ende den Workflow wieder speichern, veröffentlichen und anschließend als Nutzer ohne Listenschreibrechte (im Beispiel „max muster.“) testen. Es sollte nun kein Fehler mehr erscheinen, die Listenaktualisierung durchgeführt der Workflow ohne Probleme durchlaufen werden.

SharePoint

ppedy Intranet Berlin

Workflow Status: UrlaubsantragsWorkflow

Workflow Information	
Initiator:	max muster.
Started:	5/31/2013 12:08 PM
Last run:	5/31/2013 12:08 PM
Item:	3. versuch von max
Internal Status:	Completed
Status:	Ja

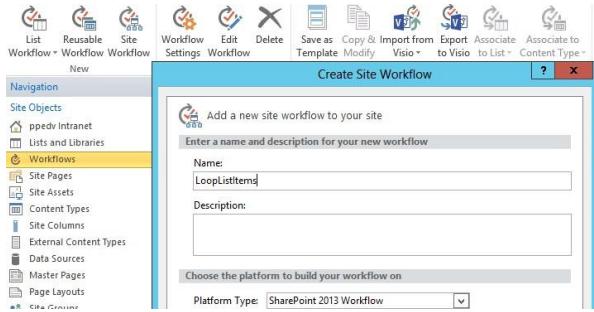
Websiteworkflow mit Loop, Webservice-Call und Dictionary Variable

Die neue SharePoint 2013 Workflow Engine bringt von Haus aus einige neue Aktionen mit. Dazu gehört neben der Implementierungsmöglichkeit für Schleifen der Aufruf eines REST Webservice.

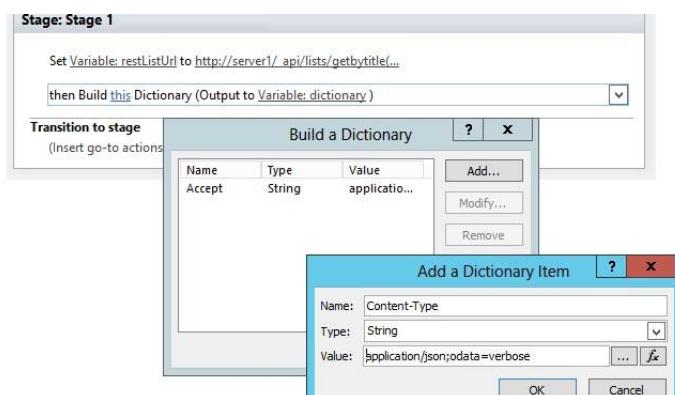
Weiterhin existiert mit dem neuen VariablenTyp „Dictionary“ nun die Möglichkeit, zusammengesetzte Variablen (unterschiedlichen Datentyps) – z.B. Listenelemente mit verschiedenen Attributen - in einem Container zu speichern.

Im Folgenden soll jedes Element einer Liste („Mitarbeiter“) mittels Schleife ausgelesen werden. Hierbei verwendet man typischerweise die sog. "for-each"-Schleife. Da aber im Designer nur zählergesteuerte (n -maliger Durchlauf) sowie kopfgesteuerte (wiederholter Durchlauf solange Bedingung erfüllt) unterstützt werden, müssen for-each-Schleifen mit Hilfe der Dictionary-Variable nachgebildet werden.

- Zunächst ist ein Website Workflow vom Typ „SharePoint 2013 Workflow“ zu erstellen



- Als erstes muss eine Workflowvariable für die URL zum Webservice-Aufruf der Zielliste erzeugt werden via Action → Set Workflow Variable. Diese beinhaltet die neue REST-API, welche im Beispiel wie folgt lautet: [http://server1/_api/lists/getbytitle\('Mitarbeiter'\)/items](http://server1/_api/lists/getbytitle('Mitarbeiter')/items)
 - Danach wird eine Aktion „Build Dictionary“ eingefügt. Dieser sind die zwei String-Variablen „Accept“ sowie „Content-Type“ je mit dem Wert „application/json;odata=verbose“ hinzuzufügen²³.

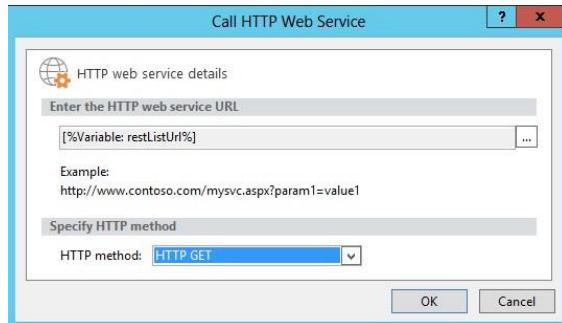


Die Dictionary-Variable kann nun im Ribbon → „Local Variables“ entsprechend umbenannt werden in „requestHeaders“.

- In der nächsten Aktion „Call http Web Service“ wird der Webservice-Aufruf durchgeführt. Dieser umfasst mehrere Schritte:

²³ Diese zusammengesetzte Variable (Dictionary) dient dazu, dem SharePoint beim Aufruf des Webservice zwei HTTP-Header Attribute zu übergeben, welche bewirken, dass die Rückgabe im JSON- und nicht wie sonst üblich im XML-Format erfolgen soll.

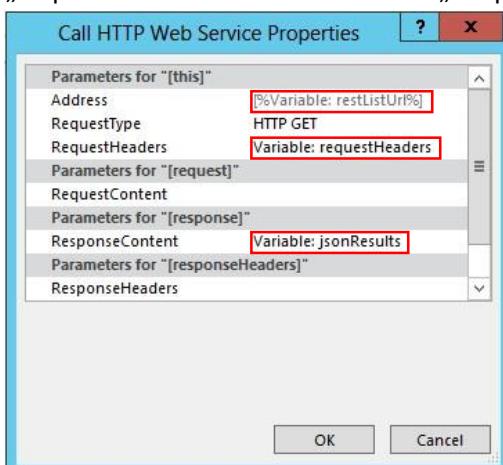
- Der erste Parameter „this“ ist mit der zuvor angelegten Variable „restListUrl“ sowie der Methode „HTTP GET“ zu belegen.



- Nun wird noch eine Dictionary-Variable „jsonResults“ für die Rückgabewerte erzeugt (Bild rechts). Dies geschieht durch Klick auf den „response“ Hyperlink (Bild links).



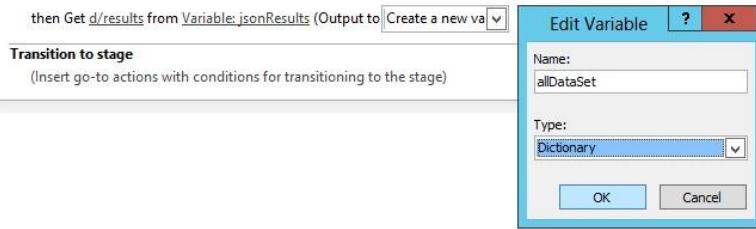
Anschließend ist der Webservice mit der rechten Maustaste anzuklicken und unter „Properties“ für die Eigenschaft „RequestHeaders“ auf die zuvor gesetzte Variable „requestHeaders“ zu setzen und für „ResponseContent“ die Variable „jsonResults“.



- Nun folgt die nächste Aktion „Get Item from Dictionary“, welche für die Extraktion der zurückgegebenen Daten verantwortlich ist. Der Parameter „item by name or path“ ist mit dem Wert „d/results“ und als Dictionary die Variable „jsonResults“ zu belegen²⁴. Das

²⁴ Da JSON genau wie XML baumstrukturartig geschachtelt ist, muss hier der Knotenpunkt, welcher ausgelesen werden soll, angesprochen werden. Um diese Struktur auslesen zu können empfiehlt sich das Tool JSON Viewer.

Ergebnis muss wieder als Dictionary abgespeichert werden, weshalb eine neue DictionaryVariable namens „allDataSet“ erstellt wird.



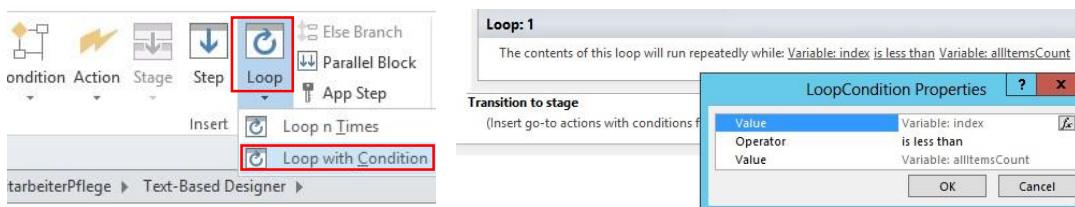
- Als nächste Aktion folgt „Count items in a Dictionary“, damit ermittelt werden kann, wieviele Listenelemente überhaupt vorhanden sind. Dieser Wert ist wichtig zur Realisierung des foreach-Schleifenkonstrukts im nächsten Schritt. Als Parameter ist hier die Variable „allDataSet“ anzugeben, welche einen Rückgabewert vom VariablenTyp Integer (im Bsp. „allItemCount“) erzeugt. Dieser wird im Folgenden als Schleifenzähler verwendet.

`then Count Items in Variable: allDataSet (Output to Variable: allItemCount)`

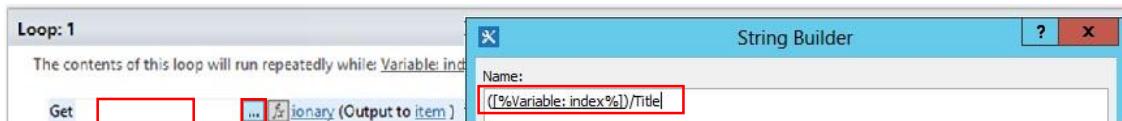
- Nun wird ein weiterer Stage eingefügt, im welchem die Schleife ausgeführt wird. Zuvor muss jedoch noch via Ribbon → „Local Variables“ eine Schleifenzählervariable („index“ vom Typ Integer) erzeugt werden, welche pro Schleifendurchlauf inkrementell erhöht wird. Diese ist direkt im Anschluss mit der Aktion „Set workflow variable“ auf 0 zu setzen.



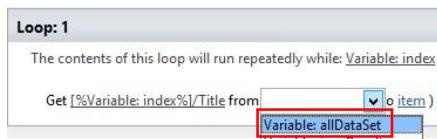
- Zum Einfügen der Schleife muss im Ribbon ein Loop → „Loop with Condition“ eingefügt werden (Bild links). Die Schleifenbedingung besteht aus drei Parametern. Der erste Wert wird mit der Schleifenzählervariable „index“ belegt. Der Operator ist mit „is less than“ und der zweite Wert mit der Gesamtzählervariable „allItemCount“ zu belegen (Bild rechts).



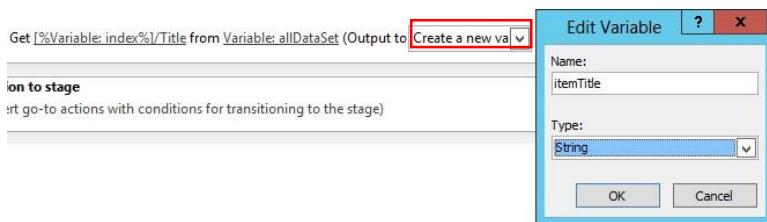
- Die erste Aktion innerhalb des Loops ist eine „Get an Item from Dictionary“-Aktion. Hier muss im ersten Parameter die Indexvariable gefolgt vom statischen Text „/Title“, womit der Elementtitel ausgelesen wird. Diese Verkettung kann über den String Builder (Aufruf über Button „...“) eingetragen werden, indem zuerst der Lookup auf die Variable gesetzt, diese dann in runde Klammern gesetzt und dahinter der Text „/Title“ eingetragen wird.



Als zweiter Parameter ist die Variable „allDataSet“ als Quelle anzugeben, da dieses Dictionary bereits in Stage 1 alle Items ausgelesen hat.

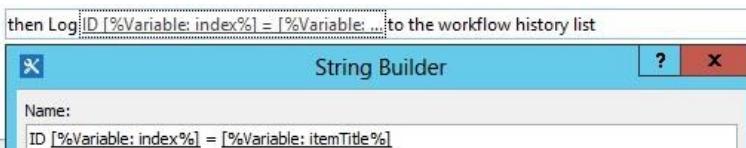


Der dritte Parameter, der standardmäßig mit „item“ belegt ist, wird per Klick auf „Create new Variable“ angelegt. Dies ist eine Variable vom Typ String, die den Rückgabewert speichert. Im Bsp. heißt sie „itemTitle“.



- Um zu testen, ob der aktuelle Elementtitel auch wirklich ausgelesen wurde, kann dieser mit einer weiteren Aktion namens „Log to History List“ für die Workflowverlaufsliste protokolliert werden. Zum Verketten des Eintrages wird auch hier wieder der String Builder verwendet.

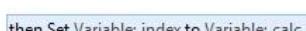
Verkettet werden Index, statischer Text und Elementtitel.



- Als dritte Schleifenaktion muss der Indexwert um 1 erhöht werden. Dies geschieht mit der Aktion „Do a Calculation“. Berechnet wird: „Index“ „plus“ „1“. Der Rückgabewert wird in der automatisch erzeugten Variable „calc“ gespeichert.



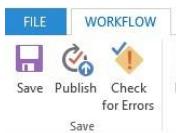
- Die letzte Aktion innerhalb der Schleife stellt das Aktualisieren des Indexwertes dar. Mit der Aktion „Set Workflow Variable“ wird „index“ auf „calc“ gesetzt und besitzt somit den aktuellen Wert.



- Zum Schluss muss noch darauf geachtet werden, dass die Stage-Transitions (Stufenübergänge) ordentlich implementiert sind. Am Ende von Stage 1 muss daher im Footer-Bereich „Transition to Stage“ eine „Go to a Stage“-Aktion eingefügt werden (Bild links). Der Stufenübergang von Stage 2 mündet im Workflowende (Bild rechts).



- Als letztes muss der Workflow dann gespeichert und veröffentlicht werden, bevor er verwendet werden kann. Dies geschieht über die Ribbon-Einträge „Save“ bzw. „Publish“.



- Nach dem Veröffentlichen kann der Workflow gestartet werden. Website Workflows befinden sich über die Weboberfläche in der aktuelle Navigation → Site Contents → Site Workflows zu erreichen.

Zum Starten des Workflows muss dieser einfach angeklickt werden. Im Anschluss reicht ein kurzes Aktualisieren des Browsers (F5) und weiter unten erscheint der Workflow Status (Bild links). Mit einem Klick auf den „Internal Status“ gelangt man in die Übersichtsseite des Workflows, wo auch Details zum Workflowverlauf bzw. Log-Einträge der History List

Date Occurred	Event Type	User ID	Description
6/4/2013 11:54 AM	Comment	ppedv	Liste beinhaltet: 5 Elemente
6/4/2013 11:54 AM	Comment	ppedv	ID 0 = Muster
6/4/2013 11:54 AM	Comment	ppedv	ID 1 = Müller
6/4/2013 11:54 AM	Comment	ppedv	ID 2 = Singer
6/4/2013 11:54 AM	Comment	ppedv	ID 3 = Schulze
6/4/2013 11:54 AM	Comment	ppedv	ID 4 = Helbig

aufgelistet sind (Bild rechts).