
Benutzerhandbuch für SharePoint Administratoren

SharePoint Connector Guidelines



Die officeatwork AG hat dieses Handbuch mit der grösstmöglichen Sorgfalt erstellt, damit die darin enthaltenen Informationen verständlich, korrekt und zuverlässig sind. Dennoch haftet die officeatwork AG keinesfalls für mit diesem Handbuch in irgendeinem Zusammenhang stehende Angelegenheiten, einschliesslich und ohne jegliche Beschränkung seiner handelsüblichen Qualität und seiner Verfügbarkeit für besondere Zwecke. officeatwork AG wird die in diesem Handbuch beschriebene Software von Zeit zu Zeit überarbeiten und behält sich das Recht vor, dies ohne vorherige Benachrichtigung des Käufers zu tun. officeatwork AG ist keinesfalls für indirekte, besondere oder zufällige Folgeschäden haftbar, die aus dem Kauf oder dem Gebrauch dieses Handbuches oder der darin enthaltenen Informationen entstehen. Dieser Garantiausschluss hat keinen Einfluss auf die gesetzlich festgelegten Rechte des Verbrauchers.

Copyright© 1992–2020 officeatwork AG, Schweiz.
All rights reserved.

officeatwork® is a registered trademark of officeatwork AG.

Microsoft® Word, Microsoft® Office, Windows®, Windows 95™, Windows 98™, Windows NT®, Windows XP®, Windows Vista, Windows 7, Windows 8, Windows 10 and MS-DOS™ are trademarks of the Microsoft Corporation.

Andere Namen von Firmen, Produkten oder Dienstleistungen können Marken oder eingetragene Marken der jeweiligen Inhaber sein.

Inhaltsverzeichnis

Über diesen Guide	5
Für wen ist der Guide bestimmt.....	5
Was deckt dieser Guide ab.....	5
Das sollten Sie bereits wissen.....	5
Typografische Konventionen.....	5
Überblick	6
Nutzen.....	6
Systeme.....	6
osct Datei erstellen	7
Vorgehen	7
Funktionen	7
Syntax.....	7
Funktionsumfang.....	8
Integration	19
osct-Datei in Dokumenten-Bibliothek hochladen.....	19
Site Content Type erstellen.....	19
Definition der Content Typen Gruppen.....	20
Content Types aktivieren	21
Content Type in der Dokumenten Bibliothek aktivieren.....	21
Content Type als Kontextmenü im Webpart aktivieren.....	23
Appendix	25
OSCT-Beispiele.....	25
officeatwork Client Wizard starten.....	26
Verwendung von Kontakten aus einer SharePoint-Liste	27
Hilfe	29
Index	30

Über diesen Guide

Für wen ist der Guide bestimmt

Dieses Buch ist für SharePoint Administratoren bestimmt, welche «officeatwork shortcut file templates» (osct) erstellen und in SharePoint integrieren.

Was deckt dieser Guide ab

Diese Anleitung zeigt Ihnen Schritt für Schritt den Aufbau eines kundenspezifischen «officeatwork shortcut file template» (osct) sowie die Integration in SharePoint.

Wichtig: Die in dieser Anleitung abgebildeten Listen, Inhalte und Masken können sich von der spezifischen Lösung unterscheiden.

Das sollten Sie bereits wissen

Voraussetzung für das Verständnis dieser Anleitung sind Kenntnisse über die Produkte officeatwork und Microsoft SharePoint.

Typografische Konventionen

Bevor Sie zu lesen beginnen, lernen Sie die benutzten typografischen Konventionen kennen.

Die nachfolgenden Darstellungsarten heben Textabschnitte mit jeweils spezieller Bedeutung hervor.

Formatierungskonvention	Art der Information
Dreieckszeichen ➤	Schritt-für-Schritt-Prozedur. Sie können diesen Instruktionsschritten folgen, um eine spezielle Aufgabe zu lösen.
Fette Schrift	Objekte, die Sie auswählen müssen, wie zum Beispiel Menüs, Schaltflächen oder Begriffe in einer Liste.
GROSSBUCHSTABEN	Tastenbezeichnungen auf der Tastatur. Zum Beispiel SHIFT, CTRL oder ALT.
TASTE+TASTE	Tastatur-Kombinationen, die gleichzeitig gedrückt werden müssen, werden mit einem + gekennzeichnet. Beispiele: CTRL+P oder ALT+F4.
<u>Fett + unterstrichen</u>	Hinweise.

KAPITEL 1

Überblick

Nutzen

Mittels «officeatwork shortcut file templates» (osct) lassen sich Dokumente auf Basis von officeatwork Vorlagen erstellen. Zusätzlich können in der osct-Datei, mittels definierter Syntax, SharePoint Daten integriert werden.

Systeme

Um osct-Dateien zu integrieren werden folgende Systeme vorausgesetzt:

- Microsoft SharePoint 2010 oder neuer
- Microsoft Office 2007 SP2 oder neuer
- officeatwork ClientSuite 4.5 SP2 oder neuer

In SharePoint werden die osct-Dateien (XML-Format) hinterlegt, welche ein erweitertes Format der bereits bekannten osc-Dateien von officeatwork sind. Dadurch können die bereits mit officeatwork erstellten und zentral gewarteten Vorlagen auch aus SharePoint heraus verwendet werden.

KAPITEL 2

osct Datei erstellen

Vorgehen

Um eine osct-Datei zu erstellen, wird eine Textdatei mit der Dateiendung *.osct (officeatwork shortcut file template) verwendet. Der Inhalt dieser Datei muss im XML Format definiert sein und der XML Schema Struktur entsprechen. Das folgende Beispiel zeigt eine einfache Implementation einer solchen osct-Datei:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Parameters>
  <CreateDocument>
    <TemplateID>letter</TemplateID>
  </CreateDocument>
</Parameters>
```

Diese Anweisung erzeugt ein Dokument auf Basis des Master-Templates «Brief».

Hinweis: Detailliertere Syntaxbeschreibungen werden im nächsten Kapitel behandelt.

Funktionen

In diesem Dokumentabschnitt wird der allgemeine Syntax und die verfügbaren Funktionen detailliert beschrieben.

Syntax

Für das Erstellen Ihrer .osct Dateien stehen Ihnen Funktionen mit folgendem Syntaxaufbau zur Verfügung.

Voraussetzung für die einwandfreie Ausführung der Funktionen:

- Alle Funktionen sind in zwei führende {{ und schliessende geschweifte Klammern }} einzuschliessen
- Die Parameter sind innerhalb von runden ()-Klammern auszuführen
- Die einzelnen Parameter sind in Singelhochkomma ' einzuschliessen
- Die einzelnen Parameter sind mit Semikolon zu unterteilen

Beispiel

```
{{Funktion('Parameter1'; 'Parameter2')}}}
```

Hinweis: Sämtliche nachfolgende Funktionen (wo nicht anders beschrieben) geben einen String als Rückgabewert zurück.

Funktionsumfang

DocLibraryDescription()

Beschreibung

Gibt die Beschreibung der aktuellen Dokumenten Bibliothek zurück.

Syntax

```
{{DocLibraryDescription()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{DocLibraryDescription()}}
```

```
Rückgabe: http://oursharepoint.com/sites/legal/shared documents
```

DocLibraryRelativeUrl()

Beschreibung

Gibt die relative URL der aktuellen Dokumenten Bibliothek zum Web zurück.

Syntax

```
{{DocLibraryRelativeUrl()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{DocLibraryRelativeUrl()}}
```

```
Rückgabe: shared documents
```


DocLibraryTitle()

Beschreibung

Gibt den Titel der aktuellen Dokumenten Bibliothek zurück.

Syntax

```
{{DocLibraryTitle()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{DocLibraryTitle()}}
```

```
Rückgabe: Shared Documents Library
```

DocLibraryUrl()

Beschreibung

Gibt die absolute URL der aktuellen Dokumenten Bibliothek zurück.

Syntax

```
{{DocLibraryUrl()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{DocLibraryUrl()}}
```

```
Rückgabe: http://oursharepoint.com/sites/legal/shared%20documents
```

DocumentFileFullPath()

Beschreibung

Gibt den absoluten Speicherpfad inklusive Dateinamen des aktuellen Dokuments zurück.

Syntax

```
{{DocumentFileFullPath()}}
```

Parameter

Keine

Beispiel 1

Funktion: `{{DocumentFileFullPath()}}`

Rückgabe: `http://oursharepoint.com/sites/legal/shared documents/
Brief an Hans Muster`

Beispiel 2

Funktion: `{{DocumentFileFullPath()}}`

Rückgabe: `http://oursharepoint.com/sites/legal/shared documents/
Folder1/Folder2/Brief an Hans Muster`

DocumentFileName()

Beschreibung

Gibt den Dateinamen des aktuellen Dokuments zurück.

Syntax

```
{{DocumentFileName()}}
```

Parameter

Keine

Beispiel

Funktion: `{{DocumentFileName()}}`

Rückgabe: `Brief an Hans Muster`

If()

Beschreibung

Je nach Wahrheitswert der Condition gibt die Funktion den Wert von [Value when true] oder den Wert aus [Value when false] zurück.

Syntax

```
{{If('Condition'; 'Value when true'; ['Value when false'])}}
```

Parameter

Condition	Bedingung welche es zu prüfen gilt. Aufgebaut aus einem Operator und zwei Zahlen. Verfügbare Operatoren: [<code><=></code> <code>==</code> <code>!=</code> <code><></code>]
Value when true	Wenn das Resultat der Condition wahr ist, wird dieser Teil ausgeführt.
Value when false	Optionaler Parameter, wenn das Resultat der Condition falsch ist, wird dieser Teil ausgeführt.

Beispiel 1

```
Funktion: {{If('7!=6'; 'Stimmt! 7 ist ungleich 6!')}}
Rückgabe: Stimmt! 7 ist ungleich 6!
```

Beispiel 2

```
Funktion: {{If('7!=6'; 'Stimmt! 7 ist ungleich 6!'; 'Falsch, 7 ist gleich 6!')}}
Rückgabe: Stimmt! 7 ist ungleich 6!
```

Beispiel 3

```
Funktion: {{If('7==6'; 'Stimmt! 7 ist gleich 6!'; 'Falsch, 7 ist ungleich 6!')}}
Rückgabe: Falsch, 7 ist ungleich 6!
```

Beispiel 4

```
Funktion: {{If('{{ListItemCount('{{WebUrl()}}/Lists/Contacts)}}>=1'; 'Es hat Kontakte in der Liste'; 'Es hat keine Kontakte in der Liste')}}
Rückgabe: 'Es hat Kontakte in der Liste'
```

ListItemCount()

Beschreibung

Zählt die Anzahl der Elemente der angegebenen Liste.

Syntax

```
{{ListItemCount('Url')}}}
```

Parameter

Url	Absoluter Pfad zu einer Liste.
-----	--------------------------------

Beispiel 1

```
Funktion: {{ListItemCount('http://oursharepoint.com/legal/lists/contacts')}}
Rückgabe: 7 (Wenn sich sieben Kontakte in der Liste befinden)
```

Beispiel 2

```
Funktion: {{ListItemCount('{{WebUrl()}}/lists/contacts')}}
Rückgabe: 7 (Wenn sich sieben Kontakte in der Liste befinden)
```

ListItemField()

Beschreibung

Gibt den Wert des Feldes zum aktuellen Datensatz des referenzierten ListItemLoop() zurück.

Syntax

```
{{ListItemField('LoopID'; 'FieldName')}}}
```

Parameter

LoopID	Eindeutige Bezeichnung des referenzierten Loops
FieldName	Feldname des auszuwertenden Feldes

Hinweis: Die Funktion ListItemField() ist nur innerhalb eines ListItemLoop() anwendbar.

Beispiel 1

```
Funktion: {{ListItemField('TaskLoop'; 'Title')}}
Rückgabe: Planung erstellen
```

Beispiel 2

```
Funktion: {{ListItemLoop('TaskLoop'; '{{WebUrl()}}/lists/tasks';
'<Content ID = "Task">
  <Value Name="Title" Value = "{{ListItemField('TaskLoop';
'Title')}}"/>'
</Content>)}}
Rückgabe: <Content ID = "Task">
  <Value Name="Title" Value = "Planung erstellen"
</Content>
<Content ID = "Task">
  <Value Name="Title" Value = "GL-Präsentation"
</Content>
<Content ID = "Task">
  <Value Name="Title" Value = "Entwicklung"
</Content>
...
```

Beispiel 3

```
Funktion: {{ListItemLoop('TaskLoop'; '{{WebUrl()}}/lists/tasks';
'<Content ID = "Task">
  <Value Name="Title" Value = "{{ListItemField('TaskLoop';
'Title')}}"/>'
  <Value Name="Status" Value = "{{ListItemField('TaskLoop';
'Status')}}"/>'
</Content>)}}
Rückgabe: <Content ID = "Task">
  <Value Name="Title" Value = "Planung erstellen"
  <Value Name="Status" Value = "Completed"
</Content>
<Content ID = "Task">
  <Value Name="Title" Value = "GL-Präsentation"
  <Value Name="Status" Value = "In Progress"
</Content>
<Content ID = "Task">
  <Value Name="Title" Value = "Entwicklung"
  <Value Name="Status" Value = "Not started"
</Content>
...
```

ListItemLoop()

Beschreibung

Schleife über alle Einträge einer beliebigen SharePoint-Liste (Kontakte, Aufgaben, usw.) mit möglicher Datenrückgabe im Feld Content.

Syntax

```
{{ListItemLoop('LoopID'; 'Url'; 'Content')}}
```

Parameter

LoopID	Eindeutige Bezeichnung zur Referenzierung der zu verarbeitenden Liste
Url	Absoluter Pfad (URL) zu einer SharePoint-Liste
Content	Für den Output zu generierenden Content

Beispiel 1

```
Funktion: {{ListItemLoop('TaskLoop';
'http://oursharepoint.com/sites/legal/lists/tasks';
'{{ListItemField('TaskLoop'; 'Title')}}')}}
Rückgabe: Planung erstellen
          GL-Präsentation
          Entwicklung
          ...
```

Beispiel 2

```
Funktion: {{ListItemLoop('TaskLoop'; 'http://{{WebUrl()}}/lists/tasks';
'<Content ID = "Task">
  <Value Name="Title" Value = "{{ListItemField('TaskLoop';
  'Title')}}"/>'
</Content>')}}
Rückgabe: <Content ID = "Task">
          <Value Name="Title" Value = "Planung erstellen"
          </Content>
          <Content ID = "Task">
          <Value Name="Title" Value = "GL-Präsentation"
          </Content>
          <Content ID = "Task">
          <Value Name="Title" Value = "Entwicklung"
          </Content>
          ...
```

SiteDescription()

Beschreibung

Gibt die Beschreibung des RootWebs der aktuellen Site zurück.

Syntax

```
{{SiteDescription()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{SiteDescription()}}
Rückgabe: All Informations about Legal and Compliance
```

SiteTitle()

Beschreibung

Gibt den Titel des RootWebs der aktuellen Site zurück.

Syntax

```
{{SiteTitle()}}
```

Parameter

Keine

Beispiel

Funktion: `{{SiteTitle()}}`

Rückgabe: Legal and Compliance

SiteUrl()

Beschreibung

Gibt die absolute URL der aktuellen Site zurück.

Syntax

```
{{SiteUrl()}}
```

Parameter

Keine

Hinweis: Die Funktion SiteUrl() ist nur ausserhalb eines ListItemLoop() anwendbar.

Beispiel

Funktion: `{{SiteUrl()}}`

Rückgabe: `http://oursharepoint.com/sites/legal`

UserDisplayName()

Beschreibung

Gibt den Anzeigenamen des aktuellen SharePoint-Benutzers zurück.

Syntax

```
{{UserDisplayName()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{UserDisplayName()}}  
Rückgabe: Hans Muster
```

UserEmail()

Beschreibung

Gibt die Email-Adresse des aktuellen SharePoint-Benutzers zurück.

Syntax

```
{{UserEmail()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{UserEmail()}}  
Rückgabe: hans.muster@ourcompany.com
```

UserLoginName()

Beschreibung

Gibt den Loginnamen des aktuellen SharePoint-Benutzers zurück.

Syntax

```
{{UserLoginName()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{UserLoginName()}}  
Rückgabe: ourdomain\hans.muster
```

WebDescription()

Beschreibung

Gibt die Beschreibung des aktuellen Webs zurück.

Syntax

```
{{WebDescription()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{WebDescription()}}
```

```
Rückgabe: All Informations about Compliance
```

WebParentwebUrl()

Beschreibung

Gibt die absolute URL des Parentwebs des aktuellen Webs zurück.

Syntax

```
{{WebParentwebUrl()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{WebParentwebUrl()}}
```

```
Rückgabe: http://oursharepoint.com/sites/legal
```

WebRelativeUrl()

Beschreibung

Gibt die relative URL des aktuellen Webs zur Site zurück.

Syntax

```
{{WebRelativeUrl()}}
```


Parameter

Keine

Beispiel

```
Funktion: {{WebRelativeUrl()}}
```

```
Rückgabe: http://oursharepoint.com/sites/legal/global => global
```

WebTitle()**Beschreibung**

Gibt den Titel des aktuellen Webs zurück.

Syntax

```
{{WebTitle()}}
```

Parameter

Keine

Beispiel

```
Funktion: {{WebTitle()}}
```

```
Rückgabe: Compliance
```

WebUrl()**Beschreibung**

Gibt die absolute URL des aktuellen Webs zurück.

Syntax

```
{{WebUrl()}}
```

Parameter

Keine

Hinweis: Die Funktion WebUrl() ist nur ausserhalb eines ListItemLoop() anwendbar.

Beispiel

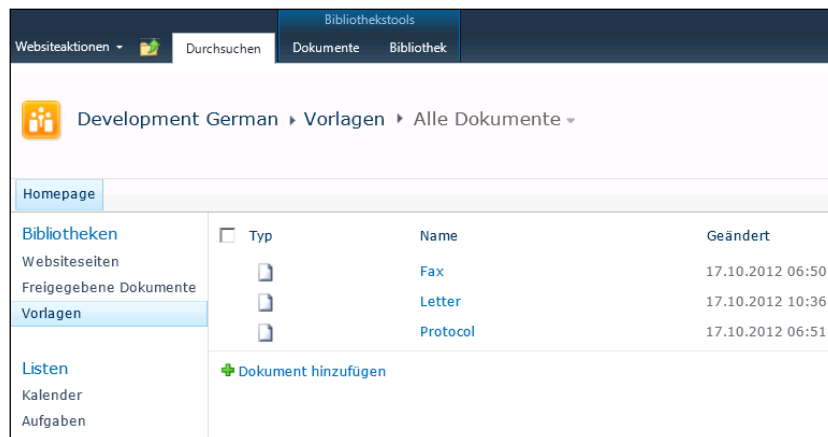
```
Funktion: {{WebUrl()}}
```

```
Rückgabe: http://oursharepoint.com/sites/legal/web
```


Integration

osct-Datei in Dokumenten-Bibliothek hochladen

- ✓ Laden Sie die von Ihnen erstellten osct-Dateien in eine bestehende Dokumenten Bibliothek (auf die die späteren Benutzer mindestens lesenden Zugriff haben) oder erstellen Sie diese Dokumenten Bibliothek neu.



Es wurde auf der Ebene der Root Site eine Dokumenten Bibliothek mit dem Namen „Vorlagen“ erstellt auf welche alle Benutzer lesenden Zugriff haben. In dieser Dokumenten Bibliothek wurden drei osct-Dateien hinterlegt.

Site Content Type erstellen

- ✓ Erstellen Sie für die osct-Dateien in den „Site Content Types“ (WebsiteInhaltstypen) eigene Content Types (Inhaltstypen).
- ✓ Geben Sie diesen einen Namen (Beispiel: Brief, Fax, Protokoll ...) und fügen Sie sie einer bestehenden Gruppe hinzu oder erstellen Sie eine neue Gruppe (Beispiel: „#Global:Global Templates“). Der Gruppenname wird zur Strukturierung des Vorlagen-Menüs verwendet.
- ✓ Als übergeordneten Inhaltstypen „officeatwork document base“ aus der Gruppe „officeatwork“ aus.

Development German > Websiteeinstellungen > Websiteinhaltsstypen > Neuer Websiteinhaltsstyp

Mithilfe dieser Seite können Sie einen neuen Websiteinhaltsstyp erstellen. Einstellungen für diesen Inhaltsstyp werden anfangs aus dem übergeordneten Inhaltsstyp kopiert, und durch künftige Aktualisierungen des übergeordneten Inhaltsstyps werden die Einstellungen für diesen Inhaltsstyp möglicherweise überschrieben.

Name und Beschreibung
Geben Sie einen Namen und eine Beschreibung für diesen Inhaltsstyp ein. Die Beschreibung wird auf der neuen Schaltfläche angezeigt.

Name:
Beschreibung:

Übergeordneter Inhaltsstyp:
Übergeordneter Inhaltsstyp auswählen aus:
Übergeordneter Inhaltsstyp:
Beschreibung: Base content type for all officeatwork document content types

Gruppe
Geben Sie eine Websiteinhaltsstyp-Gruppe an. Durch das Einstellen der Inhaltsstypen in Gruppen können Benutzer diese schneller finden.

Legen Sie diesen Websiteinhaltsstyp hier ab:
 Vorhandene Gruppe:
 Neue Gruppe:

OK Abbrechen

Erstellung des neuen Content Types «Protocol» in der Gruppe «#Global:Globale Vorlagen»

- ✓ Nun wird der Content Type noch mit einer osct-Datei verbunden. Dies können Sie in den „Erweiterten Einstellungen“ des jeweiligen Content Types vornehmen.

Development German > Websiteeinstellungen > Websiteinhaltsstypen > Protokoll > Erweiterte Einstellungen

Mithilfe dieser Seite können Sie erweiterte Einstellungen für diesen Inhaltsstyp ändern.

Dokumentvorlage
Geben Sie die Dokumentvorlage für diesen Inhaltsstyp an.

Geben Sie die URL einer vorhandenen Dokumentvorlage ein: (Vorlage bearbeiten)
 Neue Dokumentvorlage hochladen:

Schreibgeschützt
Geben Sie an, ob der Inhaltsstyp geändert werden kann. Diese Einstellung kann zu einem späteren Zeitpunkt von einer Person geändert werden, die über Berechtigungen zum Bearbeiten dieses Typs verfügt.

Soll dieser Inhaltsstyp schreibgeschützt sein?
 Ja
 Nein

Websites und Listen aktualisieren
Geben Sie an, ob alle untergeordneten Website- und Listeninhaltsstypen, die diesen Typ verwenden, mit den Einstellungen auf dieser Seite aktualisiert werden sollen. Dieser Vorgang kann sehr lange dauern, und alle an den untergeordneten Website- und Listeninhaltsstypen vorgenommenen Anpassungen gehen verloren.

Sollen alle Inhaltsstypen, die Einstellungen von diesem Typ erben, aktualisiert werden?
 Ja
 Nein

OK Abbrechen

Für den Content Type «Protokoll» wird die Verbindung zur osct-Datei «Protocol.osct» in der Bibliothek «Vorlagen» hergestellt.

Definition der Content Typen Gruppen

Die Gruppenbezeichnung der Content Typen hat zwei Aufgaben. Sie soll Ihnen die logische Gruppierung der Content Typen erleichtern und zweitens die Menü-Struktur im Ribbon-Menü „New document“ definieren.

Jeder Gruppen-Eintrag besteht auf folgenden Teilen:

Syntax

#Logische Gruppierung:Hauptmenü: [Untermenü]

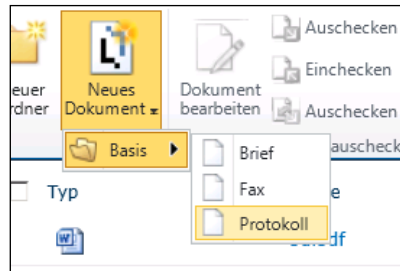
Parameter

Die Gruppen-Definition besteht aus folgenden Parametern:

- # Eindeutiges Startzeichen für officeatwork Content Types
- Logische Gruppierung Die logische Gruppierung ist eine rein logische Gruppierung um dem Anwender das Auffinden von Content Typen zur Einbindung in die entsprechenden Dokumenten

- Bibliothek zu erleichtern.
Dieser Begriff wird im Menü nicht ersichtlich.
- : Trennzeichen zwischen Logischer Gruppierung und Menü-Bezeichnungen
- Hauptmenü Die Menübezeichnung ist die effektive Menü-Bezeichnung in welcher der Content Type im officeatwork Ribbon-Menü aufzufinden ist.
- Untermenü Es können Untergruppen beliebiger Tiefe erstellt werden, indem einfach ein weiterer Doppelpunkt verwendet wird.

Beispiel



Menüstruktur für die Gruppenbezeichnung: #Global:Basis

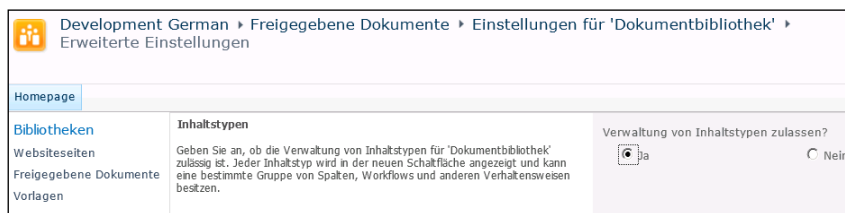
Content Types aktivieren

Die officeatwork Inhaltstypen (Content Types) können auf zwei Arten dem Benutzer verfügbar gemacht werden. Einerseits individuell in den Dokumenten Bibliotheken und andererseits über das Kontextmenü in einem Dokumenten Bibliothek-Webpart.

Nachfolgend werden die beiden Integrations-Varianten beschrieben.

Content Type in der Dokumenten Bibliothek aktivieren

- ✓ Damit Sie die neuen Content Types in einer Dokumenten Bibliothek verfügbar machen können, stellen Sie sicher, dass in den Bibliotheks-Einstellungen unter „Erweiterte Einstellungen“ die Option „Verwaltung von Inhaltstypen zulassen?“ auf „Ja“ gesetzt ist.



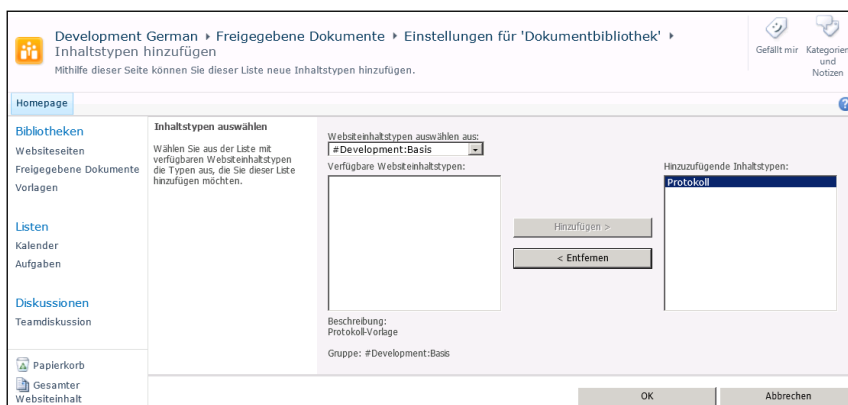
Aktivierung der Verwaltung von Content Types in den erweiterten Einstellungen der Dokumentenbibliothek

- ✓ Fügen Sie die erstellten Inhaltstypen der Dokumentbibliothek hinzu, in dem Sie in den Bibliotheks-Einstellungen auf „Aus vorhandenen Websiteinhaltenstypen hinzufügen“ klicken.



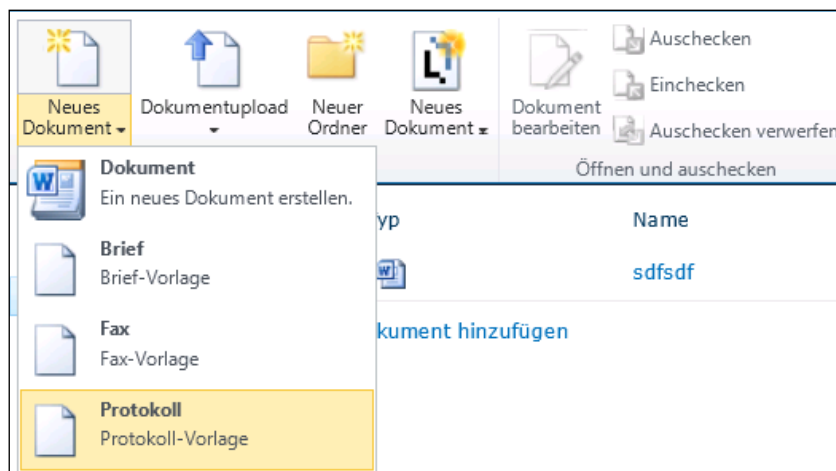
Funktion «Aus vorhandenen Websiteinhalten hinzufügen» aufrufen

- Wählen Sie die entsprechenden Content Types mit „Hinzufügen >“ aus und klicken Sie anschliessend auf „OK“.



Content Types der Dokumenten Bibliothek hinzufügen

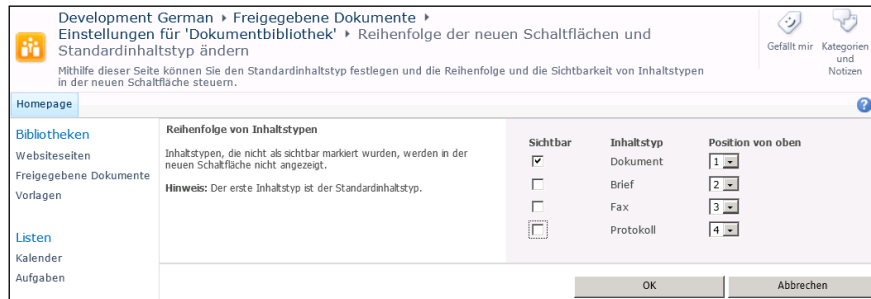
Nun wird Ihnen das officeatwork Ribbon-Menü „New document“ mit den entsprechenden Vorlagen angezeigt. Die osct-Dateien sind auch im Standard SharePoint Ribbon-Menü „Neues Dokument“ ersichtlich.



Im Standard SharePoint Menü «Neues Dokument» sind die osct-Dateien ersichtlich

Entfernen Sie diese Einträge folgendermassen:

- Deaktivieren Sie in den Dokumenten Bibliothek-Einstellungen für diese Content Types die Checkbox „Sichtbar“ und klicken Sie anschliessend auf OK“.

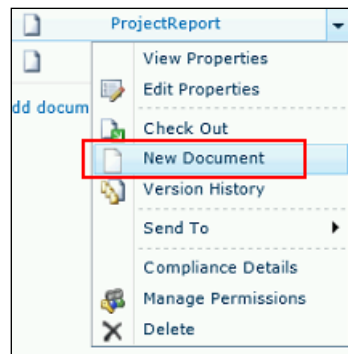


Entfernen der osct-Dateien aus dem Standard SharePoint Menü

Hinweis: Wenn die osct-Dateien nicht entfernt werden, so sind sie weiterhin ersichtlich jedoch nicht daraus ausführbar!

Content Type als Kontextmenü im Webpart aktivieren

In diesem Kapitel wird beschrieben, wie das Kontextmenü für officeatwork Inhalts Typen (Content Types) aktiviert werden kann.



Kontextmenü um ein neues Dokument zu erstellen

Voraussetzung ist, dass wie in den vorangehenden Kapiteln beschrieben, die OSCT-Dateien mit den Content Types verlinkt wurden.

In der Dokumenten Bibliothek mit den OSCT-Dateien sind nun die Content Types hinzuzufügen.

Nun sind die entsprechenden Content Types auf den einzelnen Files auf welchen der Kontextmenü-Eintrag verfügbar sein soll zuzuweisen.

- ✓ Öffnen Sie die Dokumenten Bibliothek mit den OSCT-Dateien im SharePoint.
- ✓ Wählen Sie im Kontextmenü des gewünschten Elements auf „Eigenschaften bearbeiten“.
- ✓ Wählen Sie in der Combobox „Inhaltstypen“ den entsprechenden Content Type aus.
- ✓ Klicken Sie auf die Schaltfläche „OK“

Das Kontextmenü für dieses Element ist nun um den Eintrag „Neues Dokument“ erweitert.

Mittels Dokumenten Bibliothek-Webpart und entsprechenden Ansichten (Views) können die Vorlagen nun in der gesamten SiteCollection verwendet werden.

Appendix

OSCT-Beispiele

officeatwork Client Wizard starten

Im folgenden OSCT-Beispiel ist ein Sprachcode (Language), aber keine TemplateID hinterlegt. Somit wird dem Benutzer der Wizard angezeigt und die Auswahl der Vorlage überlassen.

```
<Parameters>
  <CreateDocument>
    <Language>2055</Language>
    <TemplateID></TemplateID>
  </CreateDocument>
</Parameters>

<!--+++++----->
<!--Copyright officeatwork AG, Switzerland-->
<!--+++++----->
```

Verwendung von Kontakten aus einer SharePoint-Liste

Im folgenden OSCT-Beispiel werden mit der Funktion `{{ListItemLoop()}}` alle Kontakte aus der SharePoint-Liste „Contacts“ ausgelesen. Mit der Funktion `{{ListItemField()}}` werden die einzelnen Werte der Kontakte ausgelesen und in die MasterProperties „Recipient“ befüllt. Somit stehen diese dem Benutzer im officeatwork Client Wizard als Empfänger zur Auswahl.

```
<Parameters>
  <CreateDocument>
    <Language>2055</Language>
    <TemplateID>Letter</TemplateID>
    <ShowDocumentWizard>-1</ShowDocumentWizard>
    <Bookmarks>
      <Bookmark Name="Subject" Value="Send letter to contact of SharePoint contact list"/>
    </Bookmarks>
    <MasterProperties>
      {{ListItemLoop('ContactsListLoop'; '{WebUrl()}/Lists/Contacts/';
        '<MasterProperty IDName="Recipient">
          <Field Name="IDName" Value="{{ListItemField('ContactsListLoop'; 'Title')}}
            {{ListItemField('ContactsListLoop'; 'FirstName')}}; {{ListItemField('ContactsListLoop'; 'Company')}}"/>
          <Field Name="CompleteAddress" Value="{{ListItemField('ContactsListLoop'; 'Company')}}
            {{ListItemField('ContactsListLoop'; 'Title')}} {{ListItemField('ContactsListLoop'; 'FirstName')}}
            {{ListItemField('ContactsListLoop'; 'WorkAddress')}}
            {{ListItemField('ContactsListLoop'; 'WorkZip')}} {{ListItemField('ContactsListLoop'; 'WorkCity')}}
            {{ListItemField('ContactsListLoop'; 'WorkCountry')}}"/>
          <Field Name="Company" Value="{{ListItemField('ContactsListLoop'; 'Company')}}"/>
          <Field Name="FullName" Value="{{ListItemField('ContactsListLoop'; 'Title')}}
            {{ListItemField('ContactsListLoop'; 'FirstName')}}"/>
          <Field Name="AddressStreet" Value="{{ListItemField('ContactsListLoop'; 'WorkAddress')}}"/>
          <Field Name="AddressZIP" Value="{{ListItemField('ContactsListLoop'; 'WorkZip')}}"/>
          <Field Name="AdressCity" Value="{{ListItemField('ContactsListLoop'; 'WorkCity')}}"/>
          <Field Name="Telephone" Value="{{ListItemField('ContactsListLoop'; 'WorkPhone')}}"/>
          <Field Name="EMail" Value="{{ListItemField('ContactsListLoop'; 'Email')}}"/>
        </MasterProperty>'
      )}}
    </MasterProperties>
    <SaveAsLocation>{{WebUrl()}/Documents/{{DocumentFileName()}}.docx</SaveAsLocation>
  </CreateDocument>
</Parameters>

<!--+++++-->
<!--Copyright officeatwork AG, Switzerland-->
<!--+++++-->
```


KAPITEL 4

Hilfe

Die officeatwork Connect Plattform (connect.officeatwork.com) ist die erste Adresse um Alles rund um officeatwork zu erfahren. Auf der officeatwork Connect Plattform finden sie ein breites Angebot an Informationen:

- Knowledge-Base
- Q&A
- Download Center
- Installer
- Handbücher
- Videos
- Forum
- Glossar
- etc.

Für den Zugriff auf officeatwork Connect registrieren Sie Ihre Microsoft-Account unter www.officeatwork.com
→ [Connect](#)

Alle Supportoptionen und Ressourcen finden sie auf der Webseite www.officeatwork.com → [Support](#)

Weitere Services von officeatwork rund um Schulung und Beratung finden Sie auf der Webseite www.officeatwork.com → [Services](#)

Index

—F—

Funktion

- DocLibraryDescription(), 8
- DocLibraryRelativeUrl(), 8
- DocLibraryTitle(), 9
- DocLibraryUrl(), 9
- DocumentFileFullPath(), 9
- DocumentFileName(), 10
- If(), 10
- ListItemCount(), 11
- ListItemField(), 11
- ListItemLoop(), 12
- SiteDescription(), 13
- SiteTitle(), 14
- SiteUrl(), 14
- UserDisplayName(), 14
- UserEmail(), 15
- UserLoginName(), 15
- WebDescription(), 16
- WebParentwebUrl(), 16
- WebRelativeUrl(), 16
- WebTitle(), 17
- WebUrl(), 17

—I—

Integration, 19

- Content Types aktivieren, 21
- Dokumenten Bibliothek, 21

Kontextmenü in Webpart, 23

—N—

Nutzen, 6

—O—

osct Datei

- Vorgehen, 7

osct Datei erstellen, 7

osct-Datei

- in Dokumenten-Bibliothek hochladen, 19

- Site Content Type erstellen, 19

osct-Datei

- Funktionen, 7

- Funktionsumfang, 8

- Syntax, 7

osct-Datei

- Definition der Content Typen-Gruppen, 20

—S—

Systeme, 6

—T—

Typografische Konventionen, 5

—U—

Überblick, 6

officeatwork AG
Bundesplatz 12
6300 Zug, Switzerland

T +41 41 544 7100

www.officeatwork.com
mail@officeatwork.com