

# **WebCenter Content – Wissenstransfer für den erfolgreichen Einsatz**

**Vlad Flore  
virtual7  
D-76185 Karlsruhe**

## **Keywords**

WebCenter Content, Web Content Management, Dokumentenverwaltung, Batch Loader, Metadaten, Filesystem, Archiver, Kriterien-Workflows, Remote Intradoc Client (RIDC), Such-Engine, Profile, Regeln

## **Einleitung**

WebCenter Content ist ein erstaunliches System, das mit vielen Funktionalitäten ausgestattet ist, die einen großen Teil der Anwendungsfälle, die von Kunden gefordert werden, abdecken kann.

Durch seine Funktionen wird es zur ersten Wahl, wenn man an ein Enterprise Content Management denkt.

Seine Architektur erlaubt dem Entwickler sowohl neue Funktionalitäten hinzuzufügen, als auch schon existierende Funktionalitäten entweder zu erweitern oder vollständig zu überschreiben.

Ein Großteil der Arbeit im System kann ohne viel Schreiben von Code oder sogar komplett ohne Code durchgeführt werden, die Möglichkeit Funktionen in andere Anwendungen zu integrieren oder Inhalt, den man im System hinterlegt hat kann von externen Anwendungen verwendet werden, dies alles macht WebCenter Content zu einem System, das Anerkennung verdient hat und das auch Anerkennung verlangt.

Trotz seiner Flexibilität, gibt es Zeiten in denen man einen tieferen Einblick in eine Funktionalität benötigt, um mit dieser Funktion richtig und später einfacher arbeiten zu können. Manchmal erfordert dies einfach nur eine Konfiguration, die gesetzt werden muss, ein anderes Mal wird eine Anpassung, durch die Entwicklung einer Komponente nötig.

Was auch immer der Grund ist, WebCenter Content ermöglicht all dies und erfordert dadurch Engagement und Verantwortung von jedem, der damit arbeitet.

Die großen Stärken des Systems stecken aber vielmehr in seinem Inneren. Im Rahmen meines Vortrags werde ich auf einige dieser Stärken eingehen und Ihnen zeigen wie einfach es ist damit zu arbeiten, trotz der Komplexität, die auf den ersten Blick erscheint.

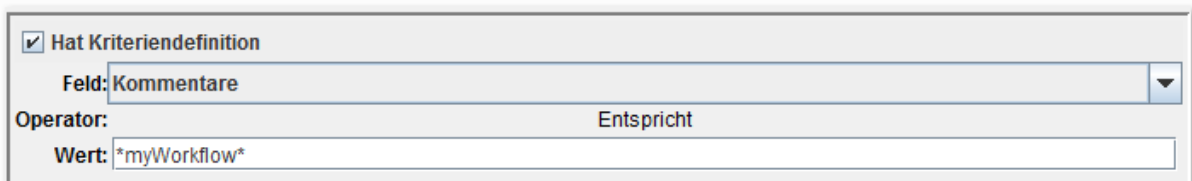
## Kriterien-Workflows

WebCenter Content bietet die Funktionalität ein Dokument in einen Workflow hinzuzufügen. Hat der Workflow ein Kriterium (außer der Sicherheitsgruppe), sprechen wir von einem Kriterien-Workflow. Die Kriteriendefinition, die man für einen Kriterien-Workflow erstellen kann, basiert auf einem anderen Metadatum, das auch benutzerdefiniert sein kann.

Die Kriteriendefinition verwendet den Operator „Matches“, das heißt, das Metadatum muss genau den angegebenen Wert haben, um die Kriterien zu erfüllen. Zumindest ist das die erste Vermutung!

Diese Vermutung ist jedoch nicht ganz richtig, da der Operator „Matches“ („Entspricht“) auch die Rolle vom Operator „like“ einnimmt. Dies würde bedeuten, dass Platzhalter verwendet werden können. In diesem Fall kann man die Wildcard \* (Stern) anwenden, um der Kriteriendefinition mehr Gewichtung zu geben.

Die Anforderung alle Dokumente, die im Feld „Kommentare“ das Wort „MyWorkflow“ enthalten in einen Workflow hinzuzufügen, kann jetzt sehr einfach implementiert werden. Man muss hierfür nur die Konstruktion „\*myWorkflow\*“ als Wert für das Feld „Kommentare“ in die Kriteriendefinition eingeben. Dabei ist die Bedingung von der Schreibweise unabhängig.



<input checked="" type="checkbox"/> Hat Kriteriendefinition
Feld: <input type="text" value="Kommentare"/>
Operator: <input type="text" value="Entspricht"/>
Wert: <input type="text" value="*myWorkflow*"/>

Abbildung 1: Kriteriendefinition für einen Kriterien-Workflow

Ein Dokument kann einem Kriterien-Workflow **nur** hinzugefügt werden beim Erstellen einer neuen Revision des Dokumentes. Was passiert aber wenn keine neue Revision erstellt und nur die Aktualisierungs-Aktion durchgeführt werden soll? In diesem Fall, nur bei der Aktualisierungs-Aktion eines Dokumentes, wird das Dokument **nicht** dem Workflow hinzugefügt.

Da WebCenter Content die Möglichkeit bietet seine Funktionen durch die Erstellung einer Komponente zu beeinflussen, ist es möglich die Aktualisierungs-Aktion zu erweitern und anhand des Custom-Codes das Dokument in den Workflow zu pushen, ohne eine neue Revision zu erstellen.

Der neue Code kann in einen Sub-Service gepackt und anhand eines Service aufgerufen werden, dies hat zum Ergebnis, dass die Abhängigkeit vom Erstellen einer neuen Revision umgangen wird.

Der neue Service kann nun anhand RIDC aufgerufen oder als extra Schritt bei anderen Services verwendet werden oder sogar direkt auf der Content-Informationen-Seite eines Dokumentes eingebunden und aufgerufen werden.

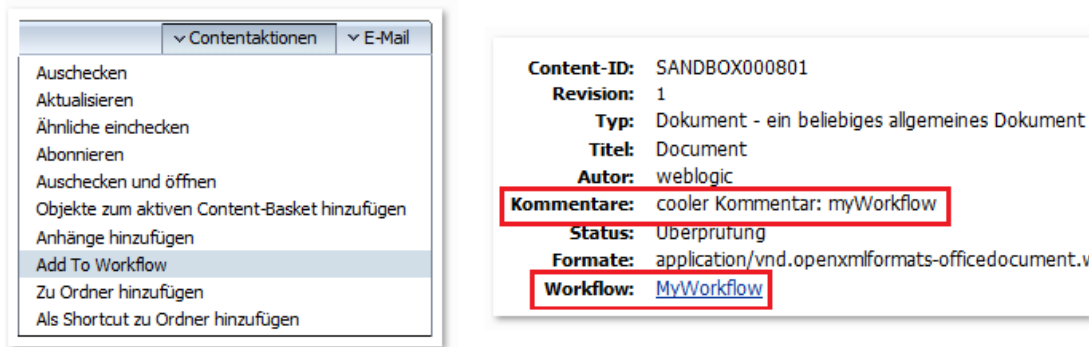


Abbildung 2: Neue Aktion im Content-Aktionen Menü und das Ergebnis dieser Aktion

## Archiver und seine spezielle Stärke

Das Archiver-Tool bietet die Funktionalität Inhalt (sowie Dokumente, die im Content Server vorliegen, als auch Daten von Tabellen) zu exportieren und importieren. Andere Aktionen, die der Archiver durchführen kann sind: Replikation oder Übertragung.

Das ist aber nicht alles was der Archiver zur Verfügung stellt. Der Inhalt kann bei der Import-Aktion verarbeitet und Eingabewerte (alle oder nur bestimmte Werte) können durch andere ersetzt werden.

Die Wertzuordnung konfiguriert man für jedes Archiv im Reiter „Importkonfigurationen“. Man könnte sagen, dass die Wertzuordnung nicht flexible ist, dass keine dynamischen Daten generiert und als Ausgabewert verwendet werden können. Das ist aber nicht ganz korrekt.

Im Ausgabewert-Feld kann man IdocScript verwenden. Methoden-Aufrufe, die ein Ergebnis zurück liefern, das als neuer Wert für die Zuordnung verwendet werden kann. Diese Möglichkeit bringt dem Archiver viel Flexibilität. Zusätzlich können auch benutzerdefinierte IdocScript-Methoden verwendet werden.

Ein wichtiger Punkt hier ist, dass nicht alle IdocScript-Methoden aufgerufen werden können, das heißt bei manchen Methoden wird einen Fehler ausgegeben. Das wird bei Methoden passieren, bei denen es keinen Zugriff auf das Service Objekt gibt.

Als Beispiel, man kann alle Werte in „xComments“ (Kommentare) Metadaten konvertieren und durch etwas wie folgt ersetzen:

```
<${#active.xComments$}><${myCustomIdocScript () $}><${dateCurrent () $}>
```

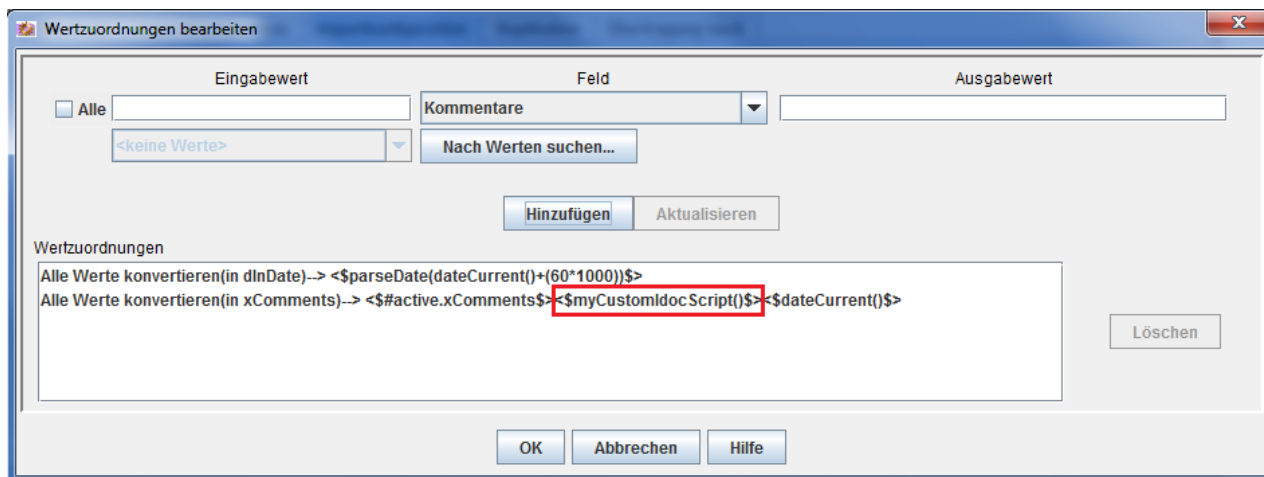


Abbildung 3: Wertzuordnung beim Import eines Archives

Wenn man viel mit Archiver arbeitet, steht man dem Problem gegenüber, dass mehrere Archive nicht gleichzeitig laufen können. Manchmal gibt es die Anforderung mehrere Archive sequentiell laufen zu lassen, natürlich bedeutet dies, dass der Status jedes Archives bekannt sein muss, um das nächste Archiv anstoßen zu können. Das heißt, Archiver muss diese Information in Erfahrung bringen.

Es gibt eine ganz einfachere Methode den aktuellen Archiver-Status herauszufinden: `ArchiverMonitor.isActive()`. Ruft man die Methode auf, erhält man die Information, ob Archiver bereits läuft.

### Eine Alternative zum Archiver für die Inhaltsverwaltung

Das Archiver-Tool bietet die Funktionalität Inhalt (sowie Dokumente, die im Content Server vorliegen, als auch Daten von den Tabellen), unter anderem zu exportieren und zu importieren. Wenn der Inhalt nur in geringem Maße abgeändert wird, ist Archiver die beste Lösung hierfür.

Ist das Archiver-Tool auch die beste Lösung für Inhalt, der sehr häufig geändert wird? Gehen wir von einer Situation aus, in der ein Entwicklungsteam mehrere Dokumente erarbeiten muss. Das bedeutet, dass jedes Dokument auf dem neuesten Stand sein muss bevor es von einem Entwickler geändert wird. Damit man den letzten Stand für jedes Dokument hat, muss man zuerst ein Archiv angelegen, das Archiv exportieren und zum Entwicklungssystem, wo die Dokumente nochmal geändert werden, wieder importieren. Diese Reihenfolge von Aktionen muss von jedem Entwickler durchgeführt werden.

Dieser Prozess ist einerseits sehr aufwändig und andererseits für Fehler anfällig.

Ein besserer Ansatz hierfür ist die Verwendung eines Repository-Systems, in dem die Dokumente hinterlegt werden können, um eine Versionierung zu ermöglichen.

Das bedeutet, dass die Dokumente zuerst aus dem Content Server exportiert, dann auf das Dateisystem abgespeichert und wieder in den Content Server importiert werden müssen.

Dieser Vorgang muss umgesetzt werden. Ein Java-Tool kann entwickelt werden, um den Export bzw. den Import zu ermöglichen. Zusätzlich muss ein Versionierungs-System verwendet und konfiguriert werden.

Der Export Vorgang muss so umgesetzt werden, dass jedes Dokument mit seinen zugehörigen Metadaten abgespeichert werden kann. Jeder Metadatumswert kann zum Beispiel als einzelner Ordner interpretiert werden. So kann man sowohl das Dokument, als auch seine Metadaten als Pfad auf dem Dateisystem abspeichern.

Der Import Vorgang kann auf verschiedene Weise umgesetzt werden. Entweder mit Hilfe des BatchLoader-Tools oder ganz von Anfang an. Das bedeutet, dass ein Java-Tool entwickelt werden könnte, um die Dokumente und seine Metadaten wieder in den Content Server einzuchecken.

Der Trick hierbei ist den richtigen Service herauszufinden. Eine standardmäßige Einstellung ist das Dokument auszuchecken und wieder mit den zugehörigen Metadaten einzuchecken. Das ist riskant und kann zu Problemen führen.

Eine bessere Einstellung ist der Service, der vom BatchLoader verwendet wird, um die Dokumente wieder einzuchecken.

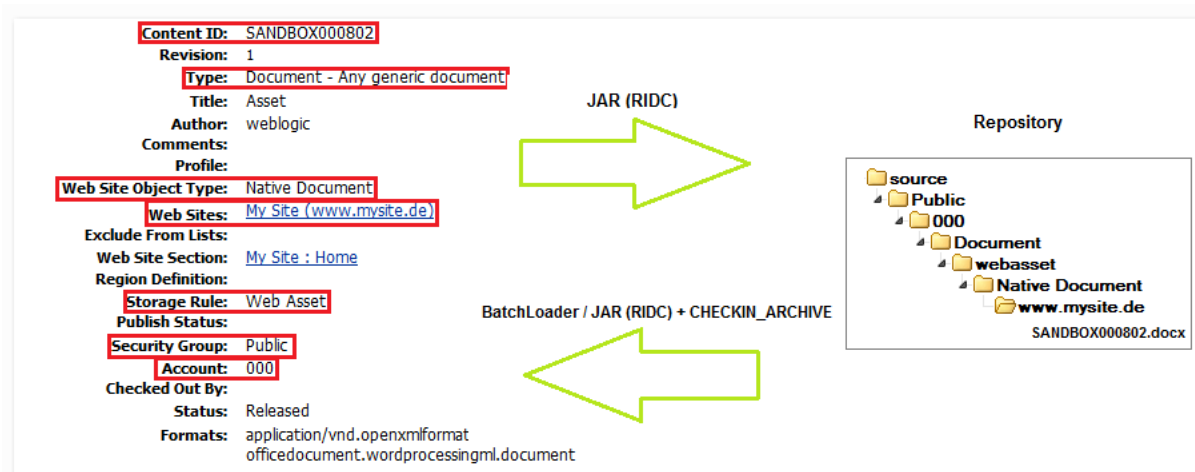


Abbildung 4: Vorgang zur besseren Verwaltung von Inhalt in WebCenter Content

## **Datenbank Ansichten und der Konfigurationsmanager**

Der Konfigurationsmanager bietet die Funktionalität Ansichten zu erstellen, die als Datenquellen für Metadaten verwendet werden können. Die Ansichten basieren auf Datenbank-Tabellen, das bedeutet, dass eine Ansicht die Daten aus einer Tabelle beinhaltet. Die gleiche Tabelle kann Daten für mehrere Ansichten zur Verfügung stellen. In diesem Fall sprechen wir von einem eins-zu-mehreren Verhältnis (eine Tabelle, mehrere Ansichten).

Natürlich taucht jetzt die Frage auf, wie man eine Ansicht basierend auf mehreren Datenbanktabellen erstellen kann? Die Lösung ist ganz simple. Man erstellt eine Datenbankansicht und verwendet diese Ansicht als Datenquelle für die WebCenter Content Ansicht.

Man kann aber nur Datenbanktabellen in den Konfigurationsmanager hinzufügen und verwenden! Das ist nicht ganz richtig, obwohl das die erste Vermutung sein wird, insbesondere nachdem man die Datenbankansicht erstellt hat und die Ansicht nicht in den Konfigurationsmanager hinzufügen kann.

Der Grund dafür ist eine Konfiguration die nicht gesetzt wurde und zwar `EBRIncludeViewsInTableList=1`. Setzt man die Konfiguration in der Datei `config.cfg`, so wird die Datenbankansicht zur Verfügung gestellt, um sie in den Konfigurationsmanager hinzufügen zu können. Das bedeutet hier, dass die Ansicht als eine Tabelle betrachtet wird.

So kann man ein Metadatum basierend auf einer Ansicht, die Daten aus mehreren Tabellen abholt im WebCenter Content erstellen.

## **Oracle Text Search und die *idcnull* Zauberei**

WebCenter Content unterstützt mehrere Such-Engines - `DATABASE.METADATA`, `DATABASE.FULLTEXT` und `OracleTextSearch`. Wenn `OracleTextSearch` oder `DATABASE.FULLTEXT` als Search-Engine verwendet wird, kann man nicht nur nach Inhalt in den Metadaten suchen, sondern auch nach Inhalt in den Dokumenten die in WebCenter Content vorliegen.

Am häufigsten gebraucht ist die Suche nach einem bestimmten Text, der als Wert für ein Metadatum gesetzt wurde. Dies kann eingestellt werden anhand des erweiterten Formulars. Wird ein Wert für ein Metadatum gesetzt, so wird das Metadatum für den Aufbau der Abfrage verwendet.

Sucht man nach dem Text „Konferenz“ im Feld „Kommentare“ wird eine Abfrage wie folgt aufgebaut:  
`xComments <matches> `Konferenz``

Was passiert aber wenn man überhaupt keinen Text in das Feld einträgt? Das Feld wird nicht für den Aufbau der Abfrage benutzt, das bedeutet, dass bei der Suche das Metadatum komplett ignoriert wird.

Wie kann man dann nach einem leeren Wert suchen? Die Lösung dafür besteht aus einem Wort: `idcnull`.

Ist der Wert „idcnul“ vorhanden, so wird er durch einen leeren Wert ersetzt. Eine Abfrage wie `xComments <matches> `idcnul`` wird als `xComments <matches> ``` interpretiert. Mit der Hilfe dieses Wortes kann man nach leeren Werten suchen.

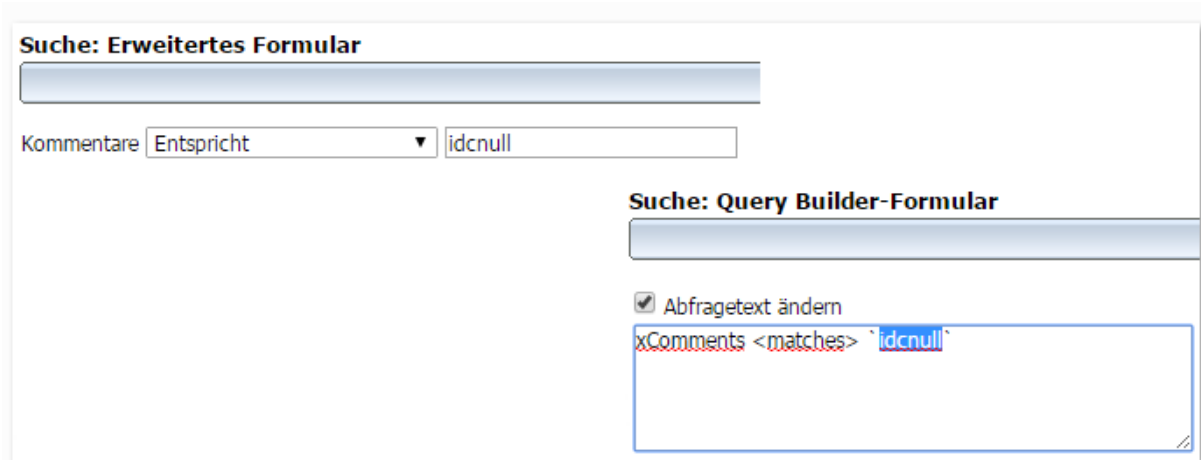


Abbildung 5: Such-Formulare

### Verwendung von einem benutzerdefinierten Include für ein Metadatum

Der Konfigurationsmanager bietet auch die Möglichkeit Profile und Regeln zu definieren und anhand dieser Formulare (Eincheckformular, Aktualisierungsformular, Suchformular) zu ändern.

Eine ganz spezielle Funktion ist die Möglichkeit ein benutzerdefiniertes Include für die Darstellung eines Metadatums zu verwenden. Es kann ein Profil definiert und, Regeln können hinzugefügt werden. Mit diesen Regeln erhält man durch custom-Includes neue Darstellungen für die Metadaten.

Gehen wir einmal von dem folgenden Problem aus: der Status eines Dokumentes soll auf der Content-Informationen-Seite als Icon dargestellt werden und auf der anderen (Eincheckformular, Aktualisierungsformular, Suchformular) als normaler Text.

Die möglichen Werte für das Metadatum sind: „published“, „unpublished“, „approved“. Für jeden Status soll auf der Content-Informationen- Seite das entsprechende Icon angezeigt werden.

Die Lösung hierfür kann wie folgt umgesetzt werden:

- eine globale Regel definieren, die nur auf der Content-Informationen-Seite aktiviert wird
- das Include definieren/umsetzen und für das Metadatum in der Regel verwenden



Abbildung 6: Benutzerdefiniertes Include verwenden


Unser Include könnte wie folgt aussehen:


```
<@dynamichtml mycustom_include@>
  <td class="idcFieldCaption idcInfoCaption">Publish Status:</td>
  <td class="idcFieldEntry idcInfoEntry">
    <$if strEquals(#active.xStatus,'published')$>
      
    <$elseif strEquals(#active.xStatus,'unpublished')$>
      
    <$elseif strEquals(#active.xStatus,'approved')$>
      
    <$endif$>
  </td>
</end@>
```

Abbildung 7: Benutzerdefiniertes Include

und das Ergebnis:

### Content Information

<b>Content ID:</b>	SANDBOX000801
<b>Revision:</b>	1
<b>Type:</b>	Document - Any generic document
<b>Title:</b>	Document
<b>Author:</b>	weblogic
<b>Comments:</b>	
<b>Profile:</b>	
<b>Web Site Object Type:</b>	
<b>Web Sites:</b>	
<b>Exclude From Lists:</b>	
<b>Web Site Section:</b>	
<b>Region Definition:</b>	
<b>Publish Status:</b>	
<b>Security Group:</b>	Public

Publish Status:  

- unpublished
- published
- approved

Abbildung 8: Zwei Darstellungen für dasselbe Metadatum



## **Filtern von Inhalten für eine WebCenter Content Ansicht auf Basis der Standard Security**

Die WebCenter Content Ansichten, die zusammen mit den Optionslisten als Datenquellen für Listenbasierte Metadaten verwendet werden können, bieten eine Funktion die angewendet werden kann, um eine bestimmte Sicherheitsebene zu erzwingen. Dies hat zur Folge, dass bestimmte Werte aus-/eingebledet werden können.

Das System bietet drei verschiedene Sicherheitsebenen und zwar: keine Sicherheit, Standarddokumentensicherheit und die benutzerdefinierte Sicherheit.

Die Standarddokumentensicherheit basiert auf einem sicherheitsrelevanten Metadatum: die Sicherheitsgruppe oder der Account. Hat der Benutzer Zugriff auf die Sicherheitsgruppe oder auf den Account, so kann man die Werte in der Ansicht verwenden. Dies bedeutet, dass jeder Wert in der Ansicht durch eine andere Information gesteuert werden kann. Diese Information muss Teil der Ansicht sein (wie zum Beispiel eine Spalte, die nicht auf der Oberfläche angezeigt werden soll). Außerdem muss die Information entweder als Sicherheitsgruppe oder als Account betrachtet werden (dies muss in der hda-Datei der Ansicht eingetragen werden).

Eine größere Flexibilität bietet die dritte Option, die benutzerdefinierte Sicherheit. Hier kann man eine ganz neue Sicherheit definieren. Dies ist möglich durch die Java-Klasse, die verwendet werden kann zur Definition einer ganz neuen Sicherheit.

Zur Erläuterung ein Beispiel. Gehen wir davon aus, dass aus einer Liste mit Zahlen nur die geraden Zahlen eingebledet werden sollen. Durch die benutzerdefinierte Java-Klasse kann man diese Funktionalität sehr einfach hinzufügen.

Die Sicherheitsoptionen für die Ansicht greifen sofort nachdem die Anpassungen gemacht wurden.



## **Fazit**

WebCenter Content ist viel mehr als nur eine Dateiablage, seine Funktionen, die Fähigkeit seinen Inhalt zu bearbeiten und zur Verfügung zu stellen, die Flexibilität die das System bietet, um Anpassungen und neue Funktionen hinzuzufügen, machen es zur ersten Wahl, wenn man über Web Content Management spricht.

Mit manchen Funktionen kommt man nicht ganz so schnell zurecht und man muss daher viel Zeit investieren, um die Art und Weise des Systems zu verstehen. Dies ist aber nicht umsonst, denn nachdem man eine Funktion richtig und gut verstanden hat, kann man sie für verschiedene andere Anwendungsfälle verwenden und somit viel Zeit sparen.

WebCenter Content ist ein starkes System mit großartigen Funktionen, die ohne viel Aufwand zur Verfügung gestellt werden können, um die Benutzerzufriedenheit zu erhöhen. Arbeitet man mit diesem System, muss man verstehen, dass die Arbeit nicht immer leicht sein wird. Trotzdem ist der Spaß garantiert!

## **Kontaktadresse:**

### **Vlad Flore**

virtual7 GmbH  
Zeppelinstraße 2  
D-76185 Karlsruhe

Phone: +49(0)721-6190170  
Fax: +49(0)721-61901729  
Email: vlad.flore@virtual7.ro  
Internet: www.virtual7.de