

# Tutorial: Informationsmodell festlegen

Version: 1  
Autor: Daniel Neumann

# Inhaltsverzeichnis

1	Beschreibung des Modells.....	2
2	Umsetzen des Modells.....	2
2.1	Aufbau des Editors.....	2
2.1.1	Rahmen.....	3
2.1.2	Navigationsfenster.....	3
2.1.3	Logobereich.....	3
2.1.4	Strukturbereich.....	3
2.1.5	Kontextmenü.....	4
2.1.6	Verschieben und Sortieren von Objekten.....	4
2.1.7	Wechsel des Verwaltungsmodus.....	5
2.1.8	Objektfenster.....	5
2.2	Anlegen von Schemata.....	6
2.3	Kindelemente.....	8
2.4	Arbeiten mit Objektinstanzen.....	9
2.5	Erweiterung des Schemas um weitere Felder.....	11
2.5.1	Zeichenkette (String).....	11
2.5.2	Richtext-Feld (Flow).....	12
2.5.3	Bild (binaryReference).....	14
2.6	Anlegen von Zitaten.....	15
2.7	Weitere Schemata.....	16

## Voraussetzungen

Um dieses Tutorial durchzuarbeiten, sollten Sie über Grundkenntnisse von XML-Schema verfügen. Möchten Sie zu diesen Themen zunächst ein Tutorial durcharbeiten, empfehlen wir Ihnen an dieser Stelle das von w3schools.com.

> [XML Schema Tutorial](#)

## Beschreibung

In der ersten Folge des Tutorials öffnen wir ein neu installiertes onion.net System. Wir definieren in der Schemaverwaltung des onion.net Editors die XML-Schemata für Zitate und Zitatsammlungen. Aus dieser Information generiert der onion.net Editor automatisch die Pflegemasken.

**Hinweis:** XML Schema ist *die* Beschreibungssprache für XML Dokumente. Durch die automatisch generierten Pflegemasken des onion.net Editors hilft Ihnen das System, XML Schema Stück für Stück zu erlernen. Dank der fest verankerten Schema- und Datenkonsistenz können selbst Einsteiger ein onion.net Live-System nicht aus der Ruhe bringen.

## Zeichenerklärung



Text, der grün umrandet und mit dem Pfeil-Symbol gekennzeichnet ist, enthält konkrete Anweisungen, was als nächstes zu tun ist.



Texte in solchen Kästchen enthalten Tipps und Tricks.

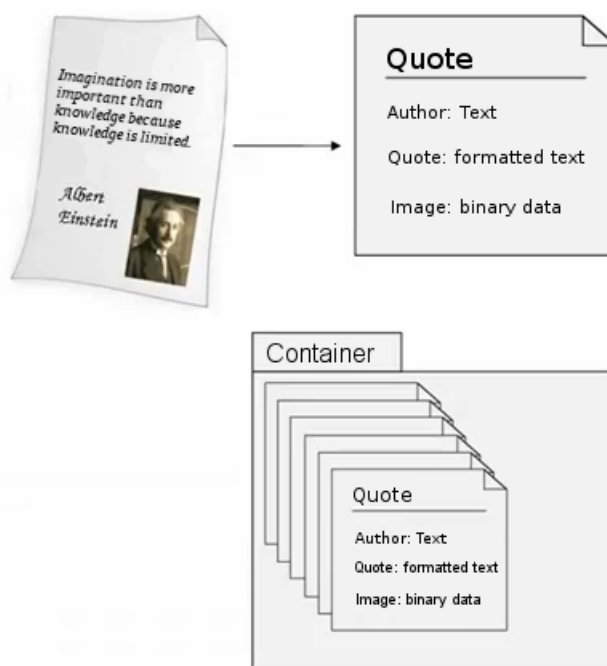
Quellcode wird in solch blauen Boxen dargestellt.

## 1 Beschreibung des Modells

Zunächst überlegen wir uns, über welche Attribute ein Zitat verfügen soll. Um es für den Anfang nicht zu komplex werden zu lassen, besteht ein Zitat in diesem Tutorial aus drei Feldern: dem Autoren, einem formatierbaren Zitattext und optional einem Bild des Autoren.



Die Zitate sollen in einer Zitat-Ablage gespeichert werden können. Damit es bei einem großen Umfang von Zitaten nicht unübersichtlich wird, sollen Zitate in Zitatsammlungen zusammengefasst werden können. Diese Zitatsammlungen werden wiederum in einer eigenen Ablage gespeichert.



*Beschreibung: Das in den Tutorials der Zitatverwaltung verwendete Datenmodell*

## 2 Umsetzen des Modells

Nachdem das Modell in Schriftform existiert, soll es in onion.net übertragen werden. Starten Sie nun Ihr neu installiertes onion.net-System.

Es folgt ein Kapitel aus dem Editor-Handbuch mit den grundlegenden Erläuterungen zu dem Editor.

### 2.1 Aufbau des Editors

Im onion.net-Editor sind drei voneinander unabhängige Verwaltungen unter einer gemeinsamen Benutzeroberfläche vereint: Inhaltsverwaltung, Benutzer- und Gruppenverwaltung sowie Modellverwaltung.

In den Tutorials wird die Inhaltsverwaltung auch als Detailansicht, die Modellverwaltung als Schemaverwaltung bezeichnet.

Im Folgenden werden die grafischen Elemente der Benutzeroberfläche des onion.net-Editors erläutert.

### **2.1.1 Rahmen**

Der Rahmen umfasst die im Folgenden vorgestellten Fenster des onion.net-Editors. Die Rahmenelemente lassen sich verschieben, so dass Benutzer sowohl das ganze Editorfenster als auch jedes Teilfenster verkleinern oder vergrößern können. Über die Beenden-Funktion unten rechts lässt sich der Editor schließen.

### **2.1.2 Navigationsfenster**

Das Navigationsfenster erlaubt die Navigation innerhalb der zu verwaltenden Objekte. Es besteht aus drei Bereichen: Logobereich, Strukturbereich und Wechsel des Verwaltungsmodus.

### **2.1.3 Logobereich**

Das Logo gehört zu den Editor-Anpassungen. Darüber hinaus bringt ein Klick auf das Logo den Nutzer wieder zur Startseite des onion.net-Editors (siehe Abschnitt 1.3.1.8).

### **2.1.4 Strukturbereich**

Der Strukturbereich zeigt alle Objekte in Form eines Baums an, die der angemeldete Nutzer im jeweiligen Bereich aufgrund seiner Gruppenrechte bearbeiten darf.

Ein Baum ist ein spezieller Graph, einfach ausgedrückt eine hierarchisch geordnete Menge von Objekten, die einer gemeinsamen Wurzel entspringen. Es gibt also ein Objekt, unter dem alle anderen Objekte liegen (das so genannte Systemroot-Objekt). Die Beziehung der Objekte zueinander kann man sich vorstellen wie das Verhältnis verwandter Familien zueinander in einem Familienstammbaum. Es gibt Vorfahren, Nachfahren, Eltern, Kinder und Geschwister.

Die einzelnen Äste des Objektbaums lassen sich auf- und zuklappen. Dabei wird der Zustand der einzelnen Äste gespeichert. Wird ein weiter aufgeächter Ast zugeklappt, so steht die Auffächerung wieder zur Verfügung, wenn der Ast erneut aufgeklappt wird.

Bereits beim ersten Blick auf den Editor fällt auf, dass es – anders als in den meisten anderen Informations- oder Content-Management-Systemen – sehr wenig Schaltflächen, Knöpfe und Funktionsleisten gibt. Die Oberfläche ist nicht überfrachtet, sondern schlank und funktional. Um dies zu ermöglichen, macht onion.net intensiven Gebrauch von Kontextmenüs, sowie dem Ziehen und Ablegen von Objekten.

### 2.1.5 Kontextmenü

Kontextmenüs sind aus der alltäglichen Arbeit am Computer bekannt. Bei grafischen Benutzeroberflächen ist ein Kontextmenü ein Interaktionsobjekt, das dem Benutzer zu einem bestimmten Kontext verschiedene Aktionen zur Auswahl anbietet. Im Allgemeinen wird das Kontextmenü per Klick mit der zweiten (rechten) Maustaste als Popup in der Nähe des Mauszeigers geöffnet.

Das Kontextmenü ist in mehrere Abschnitte unterteilt. Im Allgemeinen gibt es die Abschnitte Neu und Funktionen, häufig auch den Abschnitt Extras. Da der onion.net-Editor kontextsensitiv ist, werden abhängig vom angeklickten Objekt spezielle Kontextmenüs angezeigt. Nicht alle der folgenden Kontextmenüeinträge sind daher für jedes Objekt verfügbar.

- Alle Objekttypen, die unterhalb eines Objekts angelegt werden können, erscheinen in dessen Kontextmenü im Abschnitt Neu. Um ein neues Objekt anzulegen, braucht man folglich lediglich das Kontextmenü aufzurufen und den entsprechenden Objekttyp auszuwählen.
- Kopieren, einfügen, löschen oder umbenennen lässt sich ein Objekt über den Kontextmenüabschnitt Funktionen. An dieser Stelle verbergen sich in der Inhaltsverwaltung auch der Aufruf des Recherchefensters und die Vorschau.
- Für jeden Objekttyp lassen sich bestimmte, eigene Funktionen programmieren, die im Abschnitt Extras aufgeführt werden.

### 2.1.6 Verschieben und Sortieren von Objekten

Ziehen und Ablegen (Drag & Drop) ist eine Methode zum Bewegen grafischer Elemente in einer (Web-)Anwendung mittels einer Maus. Ein Element kann gezogen und über einem möglichen Ziel losgelassen werden.

Ziehen und Ablegen gehört zu den wichtigsten Steuerungselementen des onion.net-Editors und wird für zwei unterschiedliche Zwecke eingesetzt:

- Organisation von Struktur: Änderung der Reihenfolge und der Verschachtelung der Objekte im Objektbaum.
- Verlinkung: Einfaches Setzen von Links innerhalb des Freitexteditors oder bei Verweisfeldern.

Dabei gibt das System an allen signifikanten Stellen einer Ziehen-und-Ablegen-Operation eine sofortige Rückmeldung:

- Markieren des Objekts
- Visualisierung des Ziehens
- Anzeige, wo das Objekt fallen gelassen werden kann
- Rückmeldung beim Fallen lassen

### 2.1.7 Wechsel des Verwaltungsmodus

Der onion.net-Editor erfüllt drei Funktionen:

- > Inhaltsverwaltung
- > Benutzer- und Gruppenverwaltung
- > Modellverwaltung

Über die drei nebeneinander liegenden Schaltflächen kann der Nutzer von einem Modus in den anderen wechseln. Dabei sieht jeder Nutzer nur die Schaltflächen, für deren Modus er Zugriffsrechte besitzt. Das ist wichtig, damit jeder Nutzer sich auf die Bereiche und Objekte konzentrieren kann, die für seine Tätigkeit wichtig sind.

Mit einem Klick auf das Papierkorb-Symbol öffnet man das Recherchefenster. Hier kann man Löschprozessen suchen und Dokumente wiederherstellen.

### 2.1.8 Objektfenster

Wenn ein Redakteur den onion.net-Editor aufruft, zeigt das Objektfenster zunächst eine Startseite an, auf der drei Informationen zu sehen sind:

- > die Version des Editors,
- > der Name des angemeldeten Benutzers sowie
- > die Anzahl der von diesem Benutzer ausgeliehenen Dokumente. Ein Klick auf »Ausgeliehene Dokumente« ruft das onion.net-Recherchefenster auf, das alle ausgeliehenen Dokumente aufführt.

Wird ein Objekt im Strukturbereich angeklickt, verändert sich die Ansicht im Objektfenster. Das Objekt kann nun bearbeitet oder verschiedene Aspekte des Objekts betrachtet werden. Der Inhalte und Funktionen des Objektfensters unterscheiden sich in den drei Modi voneinander. Sie werden in den nächsten Abschnitten beschrieben.

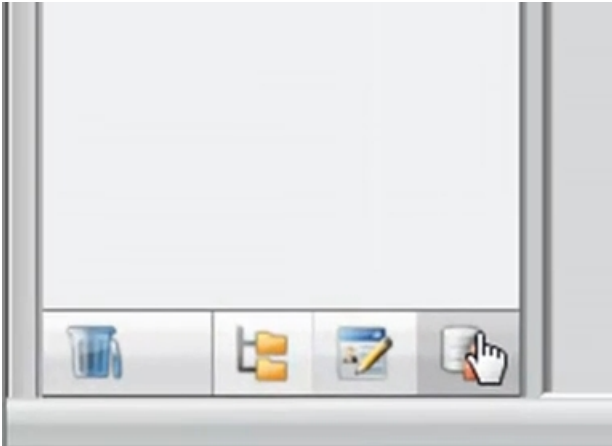
Das Objektfenster kann auch allein stehend in einem neuen Fenster geöffnet werden. Dazu müssen Online-Redakteure beim Klick auf das entsprechende Objekt im Strukturbereich die UMSCHALT-Taste gedrückt halten.

Die Ansicht, die sich zunächst öffnet, heißt „Detailansicht“ und ist in zwei Spalten unterteilt. Links sehen Sie eine Baumansicht, rechts den jeweiligen Detailbereich des in der Baumansicht ausgewählten Elements.

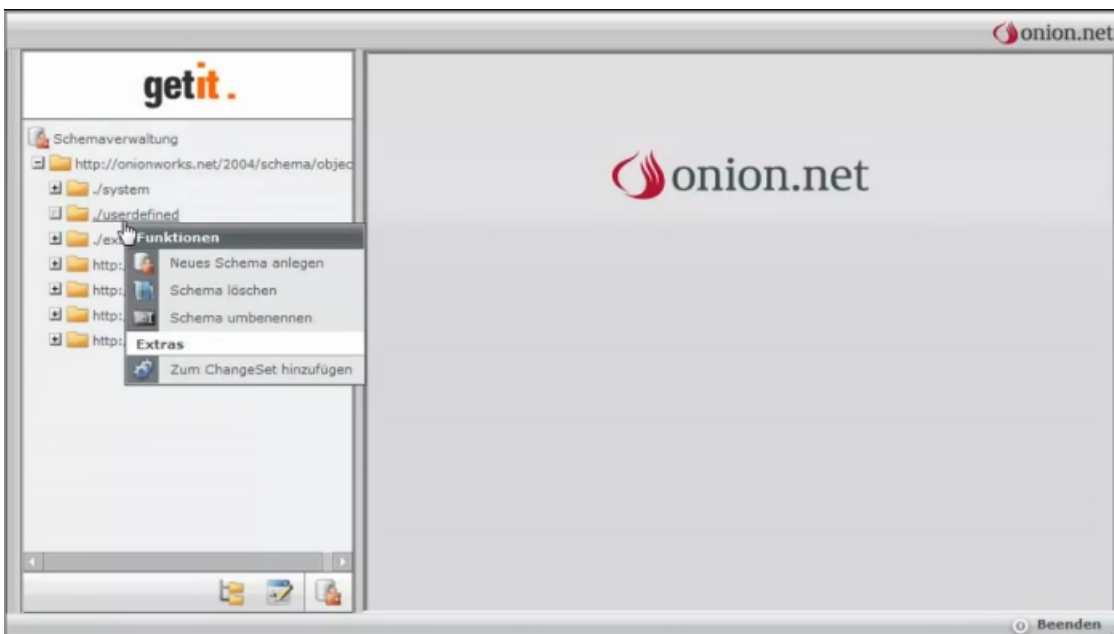
In der Detailansicht werden konkrete Objekte angelegt und bearbeitet sowie die Transformationen für die Ausgabe der Objekte in unterschiedliche Medienkanäle verwaltet. Doch damit wir in diesem Tutorial die ersten Objekte – in diesem Fall Zitate – anlegen können, muss ein dieses Objekt beschreibendes Schema existieren. Dieses wird in in der Schemaverwaltung des Editors angelegt.

## 2.2 Anlegen von Schemata

Zur Bearbeitung des Informationsmodells wechseln wir in die Schemaverwaltung. Diese erreichen Sie über einen Klick auf das rechte Symbol in der linken Spalte des Editors.



Das onion.net-Typsystem ist hierarchisch gegliedert. Benutzerdefinierte Schemata, wie die, die wir für die Zitatverwaltung benötigen, werden unterhalb des Schemas „userdefined“ angelegt. Ein Rechtsklick öffnet die kontextsensitiven Funktionen.

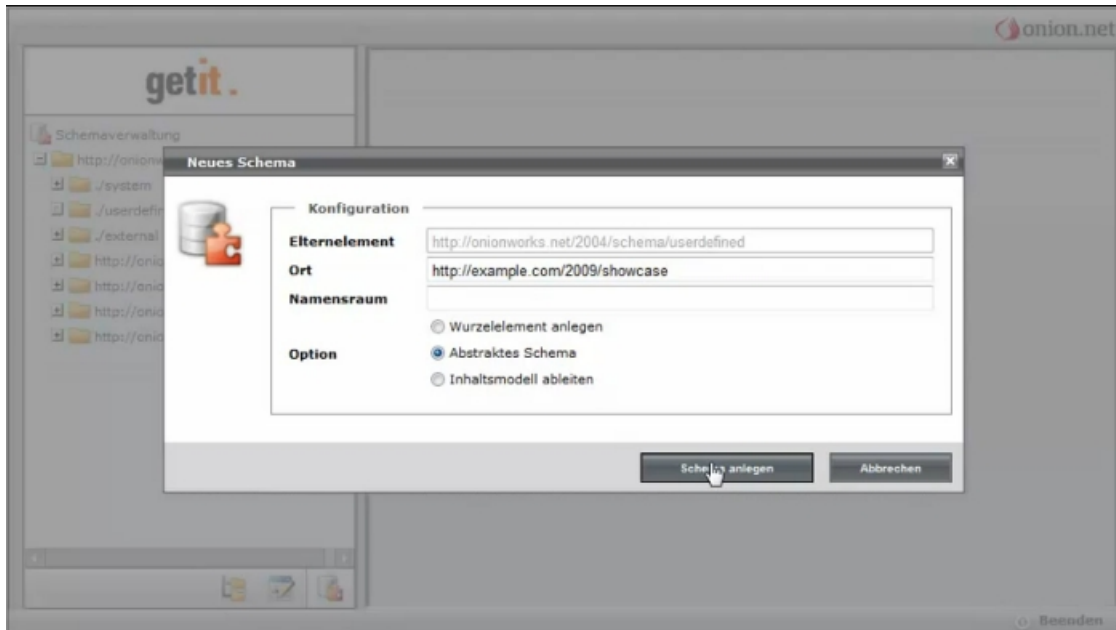


Klicken Sie im Kontextmenü auf die Funktion „Neues Schema anlegen“.

Als erstes definieren wir ein abstraktes Schema „http://example.com/2009/showcase“. Dieses dient als Basistyp für alle unsere Zitatdatentypen. Da es abstrakt ist, kann davon kein konkretes Objekt erzeugt werden und dient lediglich der Gruppierung innerhalb der Schemaverwaltung.



Tragen Sie dieses in der Maske „Neues Schema“ unter „Ort“ ein. Unter „Option“ wählen Sie „Abstraktes Schema“ aus. Ein Klick auf „Schema anlegen“ erzeugt das abstrakte Schema und öffnet automatisch dessen Bearbeitungseditor.



Unterhalb unseres Basistyps legen wir nun ein weiteres abstraktes Schema „quotations“ an. Dort sollen die Zitatablage (container) und das Zitat (quotation) Platz finden. Bei den letzteren beiden handelt es sich um konkrete Schemata, von denen auch Objekte erstellt werden können. Gehen Sie dazu vor wie bei dem vorherigen abstrakten Schema.



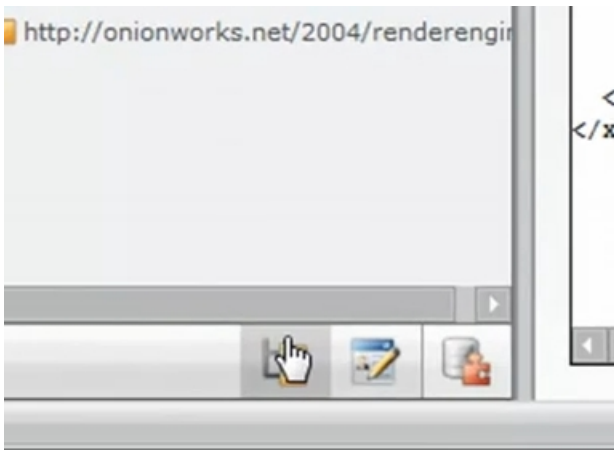
Als Ort für dieses abstrakte Schema „quotations“ wählen Sie „http://example.com/2009/showcase/quotations“.  
Für die Zitatablage „container“ erstellen Sie ein neues Schema mit dem Ort „http://example.com/2009/showcase/quotations/container“. Hierbei handelt es sich jedoch um Wurzelemente.

Die Option „Wurzelemente“ generiert ein minimales XML-Schema mit einem Wurzelement. Das minimale Schema können Sie nach dem Anlegen direkt im rechten Editorfenster einsehen.




Wir wechseln nun zurück in die Detailansicht.

Diese erreichen Sie über einen Klick auf das linke Symbol in der linken Spalte des Editors.



## 2.3 Kindelemente

Unterhalb der Systemwurzel, gekennzeichnet durch das onion.net-Logo , soll die Zitatablage angelegt werden können. Jedes Schema definiert über ein einfaches Kindschema, was angelegt werden darf.



Um die Liste der anlegbaren Typen anzuzeigen, klicken Sie mit der rechten Maustaste auf das Wurzelement. Sie sehen nun die Liste der möglichen Kindschemata.

Die Zitatablage soll nun als mögliches Kindschema unter der Systemwurzel eingefügt werden.



Tipp: Durch Alt-Rechtsklick erscheint im Kontextmenü eine Option, um direkt zum Schema zu navigieren.

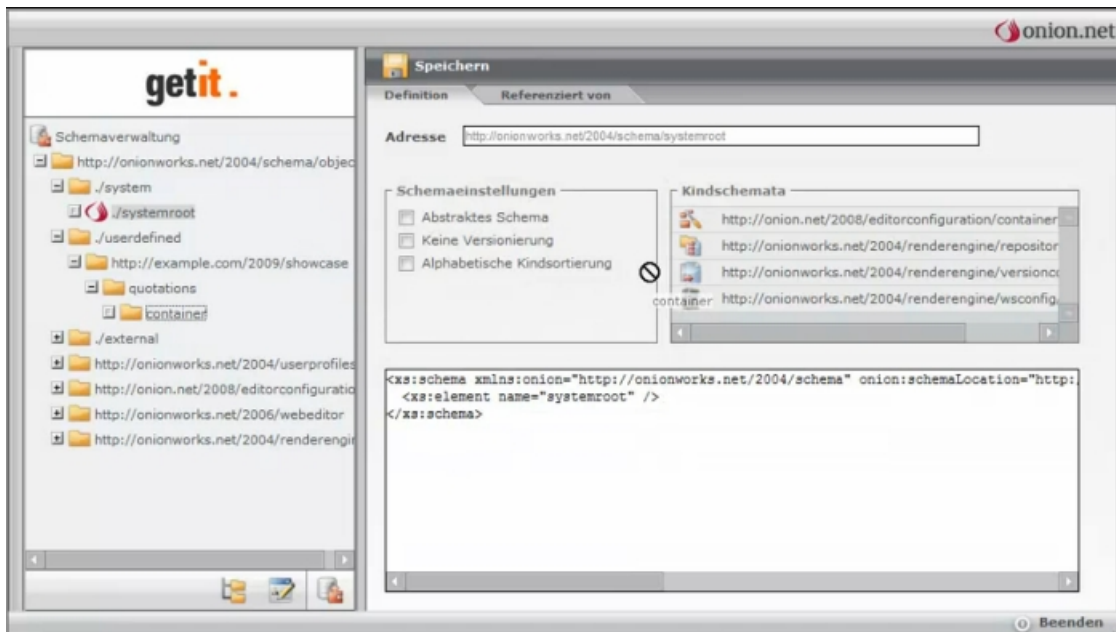


Klicken Sie im Kontextmenü unter dem dadurch neu erscheinenden Punkt „Extras“ auf „Zum Schema navigieren“. Dies wählt in der Schemaverwaltung automatisch das richtige Schema zur Bearbeitung aus.

Wir befinden uns nun im Schema der Systemwurzel (systemroot). Rechts befinden sich die zulässigen Kindschemata. Wir fügen dort unsere Ablage (container) hinzu.



Dazu ziehen Sie das Schema „container“ aus der Baumansicht links in das Feld „Kindschemata“ auf der rechten Seite. Klicken Sie dazu das Schema „container“ nicht vorher an, damit sich die Detailansicht auf der rechten Seite nicht ändert. Das Schema „container“ befindet sich nun mit in der Liste der Kindschemata. Speichern Sie nun die Änderung durch einen Klick auf das „Speichern“-Symbol am oberen Rand des Editors.



## 2.4 Arbeiten mit Objektinstanzen

Diese Änderung können Sie nun direkt überprüfen, indem Sie erneut in die Detailansicht wechseln und mit rechts auf die Systemwurzel klicken. Das System bietet uns nun an, eine Instanz unserer Ablage anzulegen.



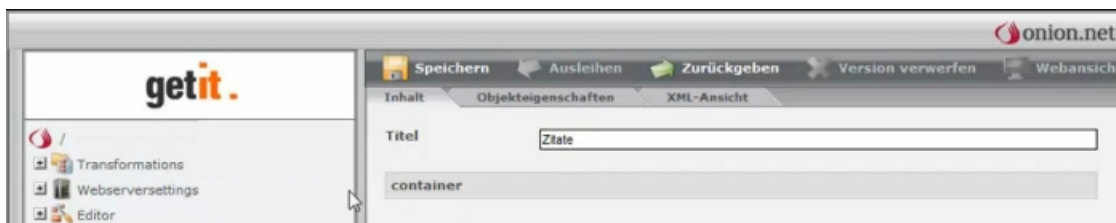
Durch einen Klick legen Sie eine neue Instanz an.

Auf der rechten Seite sehen Sie nun die Attribut-Felder, die Sie für dieses Schema vergeben können. In diesem Fall ist es der Titel der Objektes. Die Instanz wird erst dann tatsächlich angelegt, wenn Sie auf „Speichern“ bzw. „Zurückgeben“ klicken.



### Exkurs: „Ausleihen“, „speichern“, „zurückgeben“ und „Version verwerfen“

Wenn Sie ein Objekt ausleihen, bedeutet das, dass Sie dieses Objekt bearbeiten und es in dieser Zeit kein anderer Nutzer des Systems zum Bearbeiten ausleihen kann. Darüber hinaus wird auch eine neue Version des Objektes erzeugt. Änderungen können Sie über die „Speichern“-Schaltfläche sichern. Möchten Sie Ihre Änderungen verwerfen, die Sie seit dem letzten Ausleihen gemacht (und auch gespeichert) haben, nutzen Sie die Schaltfläche „Version verwerfen“. Dies stellt den Zustand des Objekts wieder her, den es vor dem Ausleihen gehabt hat. Erst, wenn Sie auf „zurückgeben“ klicken, wird das Objekt wieder für andere Nutzer bearbeitbar. Ein Klick auf „zurückgeben“ speichert auch automatisch die Änderungen.



Wählen Sie als Titel „Zitate“ und geben Sie das Objekt zurück.

In der Baumansicht sehen Sie nun das neue Objekt „Zitate“.



Tipp: Sie können in der Baumansicht Objekte neu sortieren, indem Sie diese per Drag & Drop an die neue Position legen.

Unterhalb unserer Ablage („Zitate“) wollen wir nun Zitate erstellen können. Das Schema „Zitat“ (quotation) haben wir noch nicht angelegt. Daher holen wir es an dieser Stelle nach.



Legen Sie nun unterhalb des „quotations“-Schemas das Schema „quotation“ als neues Wurzelement an. Als Ort verwenden Sie „<http://example.com/2009/showcase/quotations/quotation>“

Wir fügen nun den Zitatdatentyp (quotation) zu den Kindschemata der Ablage (container) hinzu.



Wählen Sie dazu das Schema „container“ aus und ziehen Sie dann das Schema „quotation“ in das Feld „Kindschemata“ des „containers“. Speichern Sie dann die Änderungen.

## 2.5 Erweiterung des Schemas um weitere Felder

Das System bietet uns nun an, Zitate unterhalb der Ablage anzulegen. Doch legen Sie nun noch keines an, denn dem Zitat fehlt es noch an der Möglichkeit, Inhalte zu erfassen. Dies erledigen wir im Schema von „quotation“. Wir beginnen mit einem Inhaltselement für die Erfassung des Zitatautors.



Tipp: Wenn Sie sich bisher noch nicht mit XML-Schemata auseinandergesetzt haben, ist nun ein guter Zeitpunkt, sich ein passendes Tutorial dazu durchzulesen. Ein gutes, allerdings englischsprachiges, Tutorial finden Sie auf den Seiten von [w3schools.org](http://www.w3schools.com/schema/schema_intro.asp): Introduction to XML Schemas ([http://www.w3schools.com/schema/schema\\_intro.asp](http://www.w3schools.com/schema/schema_intro.asp))

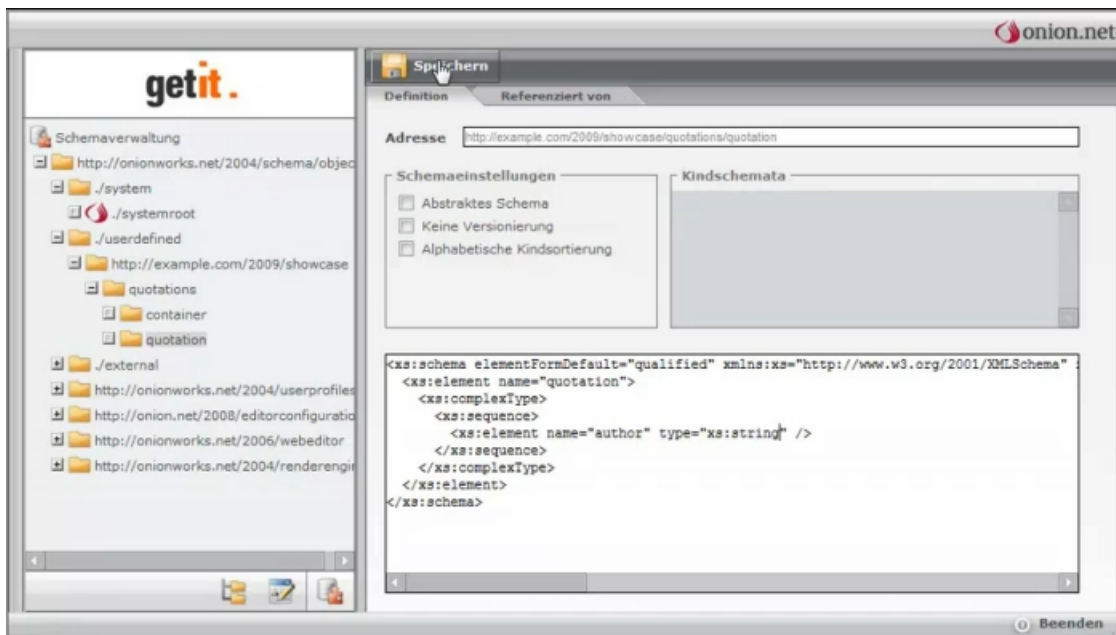
### 2.5.1 Zeichenkette (String)

Innerhalb des `<xs:complexType>`-Elements wird das leere `<xs:sequence>`-Element entfernt und durch folgende Sequenz ersetzt:

```
<xs:sequence>
  <xs:element name="author" type="xs:string" />
</xs:sequence>
```



Tipp: Der Editor verfügt über Codevervollständigung. Schreiben Sie einfach „<xs:e“ und drücken dann STRG+Leertaste. Der Editor schlägt Ihnen nun alle Möglichkeiten mit diesen Anfangsbuchstaben vor. Wählen Sie nun `<xs:element>` aus der Liste aus, wird dieses Element mit seinen Attributen in den Editor übernommen.



Speichern Sie nun das Schema. Im Datenbereich entsteht für jedes Schema-Element automatisch eine passende Pflegemöglichkeit. In diesem Fall sehen Sie neben dem obligatorischen Feld „Titel“ das neue Feld „author“, in das Sie eine Zeichenkette eingeben können. Speichern Sie diesen Test noch nicht ab, da wir noch die beiden weiteren Felder hinzufügen werden.

### 2.5.2 Richttext-Feld (Flow)

Das nächste Feld, das das Zitat benötigt, ist der Zitattext an sich. Der Zitattext selbst soll Formatierungen enthalten können. Wir nutzen zu diesem Zweck den Flow-Datentyp aus XHTML Strict.

Erweitern Sie die `<xs:sequence>` um ein weiteres Element:

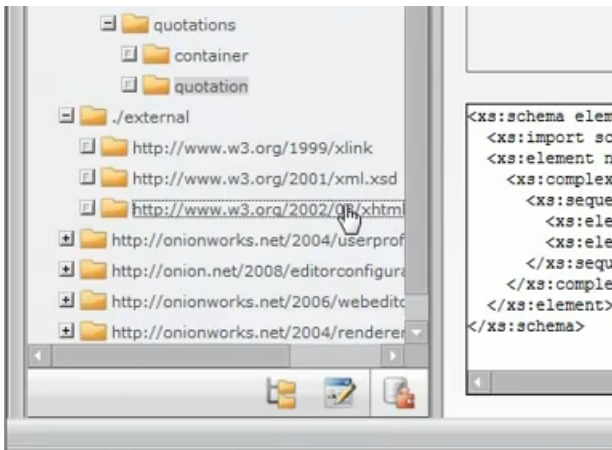
```
<xs:element name="quote" type="xhtml:Flow" />
```

Da der Namensraum „xhtml“ noch nicht definiert worden ist, geben wir dem Schema nun die benötigten Informationen mit. Dazu erweitern wir das `<xs:schema>`-Element um den Namensraum xhtml:



Tipp: weitere Informationen über Namensräume finden Sie auch bei [www.w3schools.com](http://www.w3schools.com/xml/xml_namespaces.asp): XML Namespaces  
([http://www.w3schools.com/xml/xml\\_namespaces.asp](http://www.w3schools.com/xml/xml_namespaces.asp))

Das XHTML-Schema befindet sich bereits im System. Datentypen aus anderen Schemas können einfach importiert werden. Das XHTML-Schema finden Sie im Baum neben `./userdefined` unter `./external`. Dort liegen auch noch weitere Schemata, von denen wir gleich noch den XLINK benötigen.



Importieren Sie das XHTML-Schema über die Anweisung `<xs:import schemaLocation="" namespace="" />` als erstes Element innerhalb von `xs:schema`. Ziehen Sie nun aus dem Baum das Schema „`http://www.w3.org/2002/08/xhtml1-strict.xsd`“ in das Attribut „`schemaLocation`“. In das Attribut „`namespace`“ tragen Sie „`http://www.w3.org/1999/xhtml`“ ein.

Insgesamt sieht das Schema nach diesem Schritt wie folgt aus. Die Änderungen sind hervorgehoben.

```

<xs:schema
  xmlns:onion="http://onionworks.net/2004/schema"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  elementFormDefault="qualified"
  onion:schemaLocation="http://example.com/2009/showcase/quotations/quotation"
>
  <xs:import schemaLocation="http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd"
  namespace="http://www.w3.org/1999/xhtml" />
  <xs:element name="quotation">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="author" type="xs:string" />
        <xs:element name="quote" type="xhtml:Flow" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```



Speichern Sie nun die Änderungen.

Wenn Sie jetzt ein neues Zitat anlegen, sehen Sie, wie sich unterhalb des „author“-Feldes ein Richtexteditor öffnet, in das der Zitattext (quote) mit Formatierungen eingegeben werden kann. Speichern Sie das Zitat auch jetzt noch nicht ab, da wir direkt das dritte Feld einfügen werden.

### 2.5.3 Bild (binaryReference)

Im nächsten Schritt ergänzen wir das Zitat um ein optionales Bild. Der notwendige Datentyp (binaryReference) ist im XLINK-Schema definiert.



Dazu erweitern Sie die `<xs:sequence>` um folgendes Element:

```
<xs:element
  name="image"
  type="xlink:binaryReference"
  minOccurs="0"
/>
```

Das Attribut „minOccurs“ bedeutet an dieser Stelle, dass ein von diesem Typ angelegtes Objekt mindestens so viele Elemente von diesem Typ haben muss, wie hier angegeben. Der Wert 0 macht dieses Element optional.



Wegen des neuen Namensraums brauchen wir eine weitere Import-Anweisung. Genau wie das XHTML-Schema befindet sich das XLINK-Schema unter `./external` und kann per Drag & Drop in das Attribut „schemaLocation“ und „namespace“ eingefügt werden:

```
<xs:import schemaLocation="http://www.w3.org/1999/xlink"
namespace="http://www.w3.org/1999/xlink" />
```

Unser Zitatdatentyp ist fertig. Das komplette Schema sieht nun wie folgt aus:

```
<xs:schema
  xmlns:onion="http://onionworks.net/2004/schema"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  elementFormDefault="qualified"
  onion:schemaLocation="http://example.com/2009/showcase/quotations/quotation"
>
  <xs:import schemaLocation="http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd"
  namespace="http://www.w3.org/1999/xhtml" />
  <xs:import schemaLocation="http://www.w3.org/1999/xlink"
  namespace="http://www.w3.org/1999/xlink" />
  <xs:element name="quotation">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="author" type="xs:string" />
        <xs:element name="quote" type="xhtml:Flow" />
        <xs:element
          name="image"
          type="xlink:binaryReference"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## 2.6 Anlegen von Zitaten

Wir pflegen nun drei Zitate ein. In diesem Tutorial verwenden wir diese drei Zitate:

### **Titel: Zitat Albert Einstein**

Autor: Albert Einstein

Zitat: Phantasie ist wichtiger als Wissen, denn Wissen ist begrenzt.

### **Titel: Zitat Konrad Adenauer**

Autor: Konrad Adenauer

Zitat: Wir leben alle unter dem gleichen Himmel, aber wir haben nicht alle den gleichen Horizont.

### **Titel: Zitat Thomas Carlyle**

Autor: Thomas Carlyle

Zitat: Der schlimmste aller Fehler ist, sich keines solchen bewusst zu sein.



Wie schon zuvor wechseln Sie dazu in die Detailansicht und klicken mit rechts auf Ihr „Zitate“-Element. Unter dem Punkt „neu“ können Sie ein neues Objekt des Schemas „quotation“ anlegen. Füllen Sie alle Felder aus und laden Sie auch ein passendes Bild hoch. Wenn Sie mit einem Zitat fertig sind, klicken Sie „Zurückgeben“.

## 2.7 Weitere Schemata

Nun fehlen uns noch die Datentypen, um Zitatsammlungen erstellen zu können. Dazu wechseln Sie wieder in die Schemaverwaltung. Der Ablauf ist nun ziemlich ähnlich. Wir benötigen eine Ablage (container) für die Sammlungen (collection). Die Datentypen für Sammlungen gruppieren wir unterhalb eines abstrakten Typs „collections“.



Tipp: da wir nun schon mit recht viele Schemata arbeiten, ist es sinnvoll, sich noch einmal das Datenmodell zu vergegenwärtigen:

Datentyp	Art (Option)	Schema Location (Ort)
Basistyp	abstraktes Schema	<a href="http://example.com/2009/showcase/">http://example.com/2009/showcase/</a>
Zitate	abstraktes Schema	quotations
Ablage	Wurzelement	quotations/container
Zitat	Wurzelement	quotations/quotation
Sammlungen	abstraktes Schema	collections
Ablage	Wurzelement	collections/container
Sammlung	Wurzelement	collections/collection



Legen Sie dazu unterhalb von <http://example.com/2009/showcase/> ein neues abstraktes Schema „collections“ an. In diesem Schema wiederum legen Sie das Wurzelement „container“ an.

Die Ablage (container) soll wieder unterhalb der Systemwurzel angelegt werden können. Wir fügen es zum Kindschema hinzu.



Klicken Sie dazu das Element `./systemroot` an und ziehen Sie nun den soeben erstellten container in das Feld Kindschemata. Speichern Sie die Änderung.



Legen Sie unter der Systemwurzel in der Detailansicht ein neues Objekt „Zitatsammlungen“ an und geben das Objekt zurück.

Nun legen wir den Datentyp für die Sammlung (collection) an. Diese fügen wir anschließend zum Kindschema der Ablage hinzu.



In der Schemaverwaltung klicken Sie nun mit rechts auf das Schema „collections“ und legen ein neues Wurzelement „collection“ an. Damit eine Ablage unterhalb der Sammlungen (container) erstellt werden kann, wählen Sie „container“ aus und ziehen das neue Schema „collection“ in das Feld Kindschemata und speichern die Änderung.

Die Sammlung soll auf mindestens zwei Zitate verweisen. XML Schema ermöglicht dies durch die Angabe von `minOccurs` und `maxOccurs`.



Dazu ersetzen Sie nun das leere `<xs:sequence>`-Element durch folgende Objekt-Referenz:

```
<xs:element
  name="quotation"
  type="xlink:objectReference"
  minOccurs="2"
  maxOccurs="unbounded"
/>
```

Der Wert „unbounded“ steht für „unbegrenzt“.

Da wir wieder einen XLINK benutzen, müssen wir dessen Datentyp importieren. Dies machen wir in der zweiten Zeile des Schemas durch einfügen des Tags:

```
<xs:import schemaLocation="" namespace="" />
```



Um Tipparbeit zu ersparen, ziehen Sie nun wieder das XLINK-Schema aus der Baumstruktur in die Attribute „schemaLocation“ und „namespace“. Speichern Sie nun das Schema

Nun können unterhalb der eben angelegten Ablage „Zitatsammlungen“ konkrete Sammlungen angelegt werden. Legen Sie nun eine solche Sammlung an.



Wechseln Sie nun in die Detailansicht und klicken mit rechts auf die Ablage „Zitatsammlungen“. Legen Sie eine neue collection an und geben ihr den Titel „Bekannte Persönlichkeiten“.

Sie sehen nun einen Bereich mit dem Namen „collection“, in das sich zwei Verweise ziehen lassen. Darunter befindet sich die Funktion „Neues quotation hinzufügen“. Rechts neben den Verweisen gibt es jeweils eine Funktionsbox, die durch anklicken ein Kontextmenü öffnet. Darin haben Sie die Möglichkeit, die Einträge zu sortieren oder zu löschen.



Ziehen Sie nun die Zitate von Albert Einstein und Konrad Adenauer in die beiden vorhandenen Felder. Klicken Sie dann auf „Neues quotation hinzufügen“ und nehmen Sie noch das Zitat von Thomas Carlyle mit in die Sammlung. Ändern Sie die Sortierung über die Funktionsboxen an der rechten Seite nun so, dass Konrad Adenauer am Anfang steht. Geben Sie nun das Objekt zurück.

Zuletzt ergänzen wir das Schema um die Eigenschaft „interval“. Damit soll ein Redakteur steuern können, wie lange ein Zitat angezeigt wird. Dies ist für die spätere Ausgabe des Zitats z.B. auf einer Website interessant.



Wechseln Sie zurück in die Schemaverwaltung und wählen die „collection“ unterhalb von „collections“ aus. In der Definition fügen Sie nach dem schließenden `<xs:sequence>`-Element ein Attribut-Element hinzu.

```
<xs:attribute  
  name="interval"  
  type="xs:positiveInteger"  
  default="5"  
>
```



Die Codevervollständigung funktioniert auch innerhalb von Attributen. Sie können also im Attribut „type“ STRG+Leertaste drücken und bekommen die vorhandenen Möglichkeiten angezeigt.



Wechseln Sie nun zurück zu Ihrer Sammlung „Bekannte Persönlichkeiten“. Sie sehen nun oberhalb der drei Zitate das neue Attribut „interval“ mit dem Vorgabewert „5“

Geschafft!

Im nächsten Teil kümmern wir uns um die Verwertung unserer Zitatsammlung.