

InDesign Skripting und XML

19. Juni 2010

10:00 Uhr bis 17:00 Uhr

Raum 171

Gregor Fellenz

Mediengestalter für Digital- und Printmedien

Studium Druck- und Medientechnologie, HdM Stuttgart

pagina GmbH, Tübingen

gregor.fellenz@indesignblog.com

InDesign

<xml/>

XML-Rules

JavaScript

Übersicht

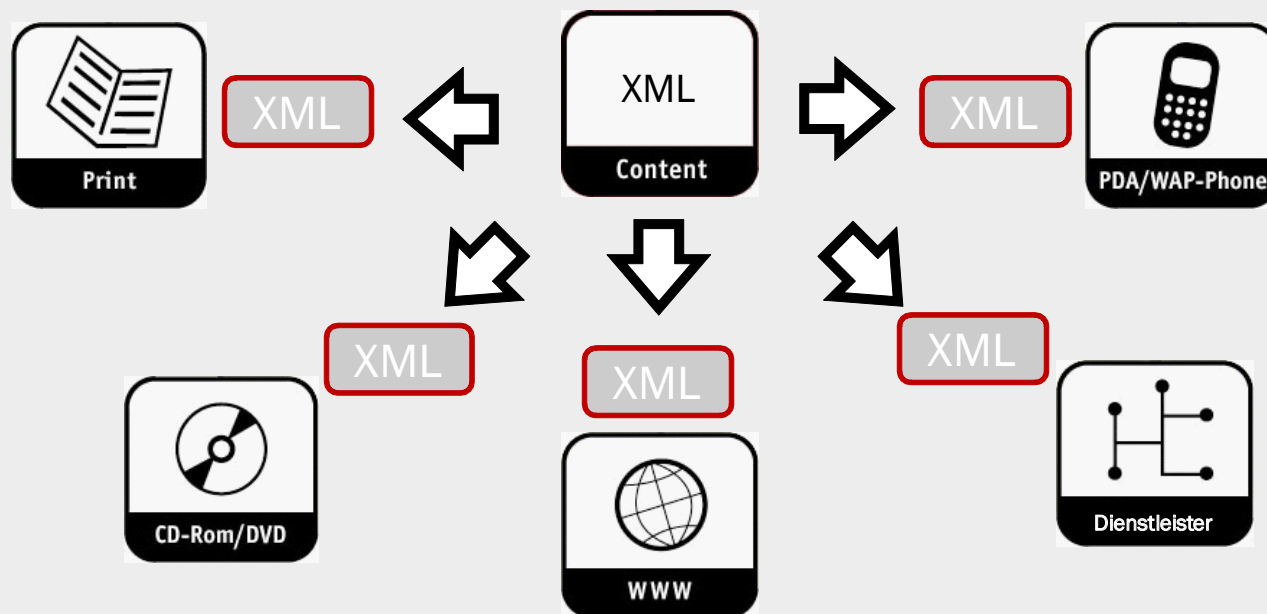
- **DTP und XML** – Boardmittel von InDesign kennenlernen
- Zusammenspiel von **strukturierten Daten** und Satzautomation
- **XML Rules** – der XPath zum Erfolg
- Den **XML-Funktionsumfang** von InDesign erweitern
- **Workflows** – Muss es immer Roundtripping sein?

Die Folien zum Download:

<http://www.indesignblog.com/hdm>

Warum überhaupt XML

- **Weil wir nichts anderes haben?**
- **Datenchaos:** Formate, Datenprobleme, Konvertierungsprobleme
- **Wiederverwertbarkeit:** Langzeitarchivierbarkeit, Kompatibilität
- **Produktion und Standardisierung:** Medienbrüche beachten
- **Mehrfachverwertung der Daten:** Print, E-Book, Smartphones, Tablets, Internet, DVD



Extensible Markup Language

Standardisiertes Dateiformat

- Austausch von Dokumenten, Definition von Datenstrukturen
- Sprache zur Beschreibung von **Dokumentstrukturen** bzw. eine Metasprache, die eine abstrakte Syntax für Auszeichnungssprachen definiert
- **Dokumentzentriert vs. datenzentriert**
- Keine Datenbank

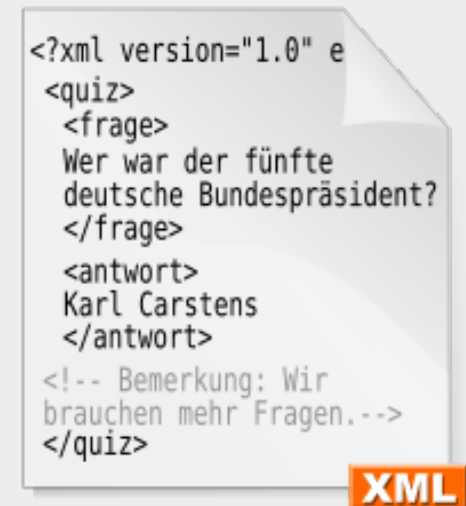
Auch im Bereich Publishing?

Mit Strukturierung wurde schon immer das Erfassen des Informationsgehaltes erleichtert.

Umgesetzt durch Formatierung wird indirekt die Struktur vom Leser wahrgenommen.

Aber: XML ist nur eine Sprache

Die Kunst, Information zu strukturieren, wird durch XML nicht ersetzt.



Beispiel

XML-Repräsentation eines Rezeptes

Öffnen Sie die Dateien `01_kochbuch_beispiel.xml` und `02_kochbuch.dtd` aus dem Ordner `10_xml_grundlagen` in einem XML-Editor

- ! Analysieren Sie die DTD anschauen
- ! Fügen Sie der XML-Instanz ein neues Rezept hinzu

DTP vs. XML

Wesentliches Feature von DTP-Programmen ist die **freie Gestaltung** von Layouts

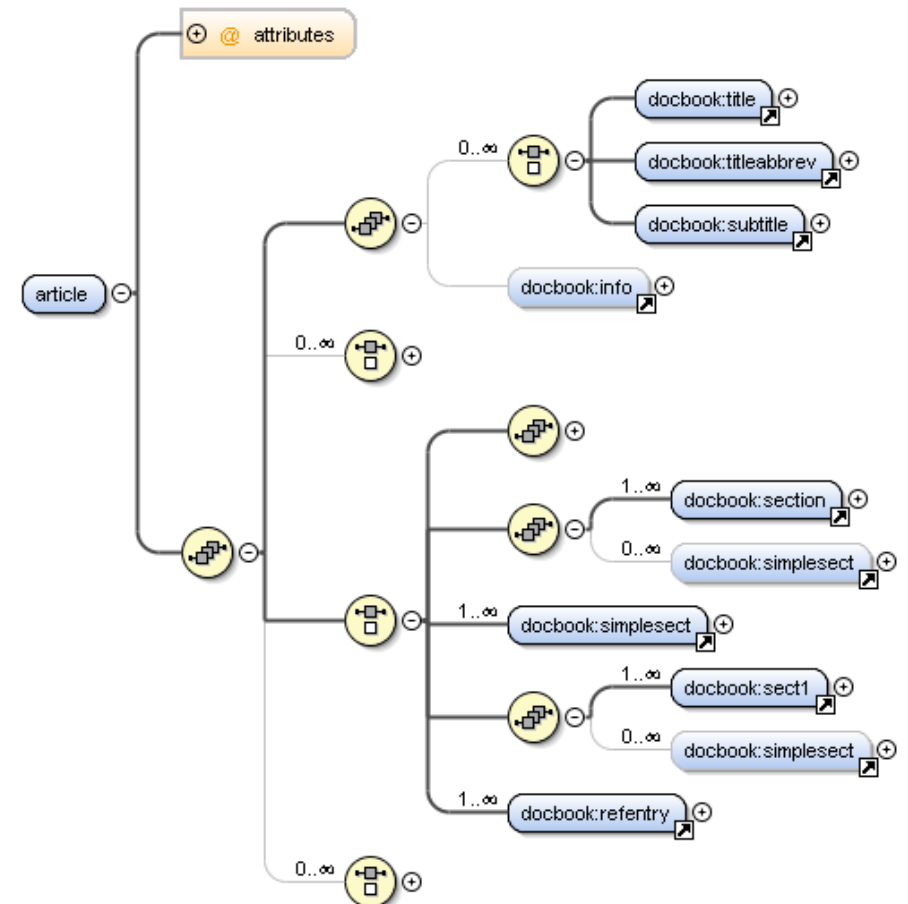
- Seiten- und Rahmenbasiertes arbeiten
- Keine Festlegung des Datenstroms „Main Story“
- Arbeitsweise vom Endprodukt (Druck) inspiriert

XML-Dokumente sind **stark strukturiert**

- Hierarchien und Kapitelstrukturen
- Ein einziges Wurzelelement bestimmt die Datenstruktur und das Datendesign
- Grammatiken (DTD) von den Inhalten (Struktur und Semantik) inspiriert
- Vergleiche dazu: Office Open XML, IDML, XSL-FO etc. mit DocBook, xhtml, Verlags DTD

InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML



InDesign und XML

*InDesign ist eine von **vielen Applikationen** die XML unterstützt.*

- **Richtig erstellte** InDesign-Dokumente können als XML weiterverwendet werden.
- Nach **entsprechender Anpassung** kann InDesign beliebige XML-Dokumente formatieren.

Wozu und für wen eigentlich?

- Dokumente **liegen als XML** vor.
- Ähnliche Strukturen in **wiederkehrenden** Aufträgen/Projekten.
- Dokumente sollen **weiterverwendet** werden.
- Daten müssen **als XML abgeliefert** werden.
- Bereitschaft sich mit der **Technologie** auseinanderzusetzen.

InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

Typischer Workflow



- **Wo** kommen die Daten her?
- Ist **Roundtripping** gewünscht?
- Welche **Zielformate** sollen bedient werden?

Oder nur Teilstrecken...

- XML-Dokumente vorhanden → **Import**
- InDesign Dokumente vorhanden → **Export**

XML in InDesign verwenden

- **Grundsätzlich:** Dokument- und XML-Inhalte auseinanderhalten.

- **Struktur Palette**

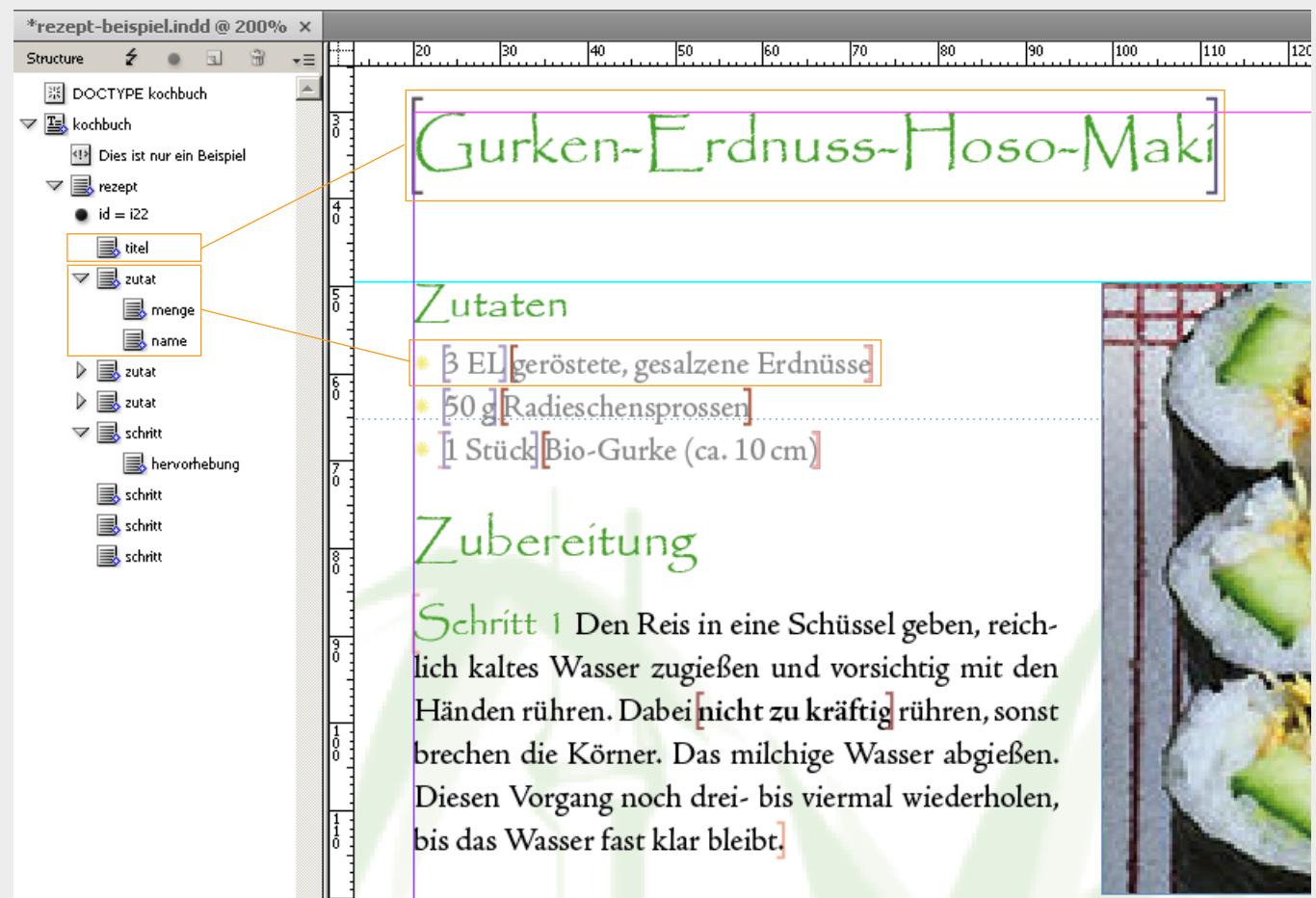
Ansicht -> Struktur ->
strukturpalette [Alt +
STRG + 1]

Übersicht über die XML-
Inhalte, Verknüpfungsstatus,
Strukturänderungen,
Attribute, validieren

- **Navigation** in der
Strukturpalette über
Shortcuts

- Tag-Marken für Rahmen bzw.
Text einblenden

Ansicht -> Struktur



Übersicht aller Icons der Strukturpalette

Icon	Beschreibung
	Element (nicht platziert)
	DOCTYPE Element
	Textabschnitt
	Text
	Text (nicht platziert)
	Abbildung
	Abbildung (nicht platziert)
	Tabelle
	Tabellenzelle
	Fußzelle (Tabelle)
	Kopfzelle (Tabelle)
	Attribut
	Processing Instruction
	Kommentar

InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

■ Tags Palette

Fenster -> Tags

Elemente anlegen, verwalten und zuweisen

■ Texteditor

Bearbeiten -> Im Textmodus bearbeiten

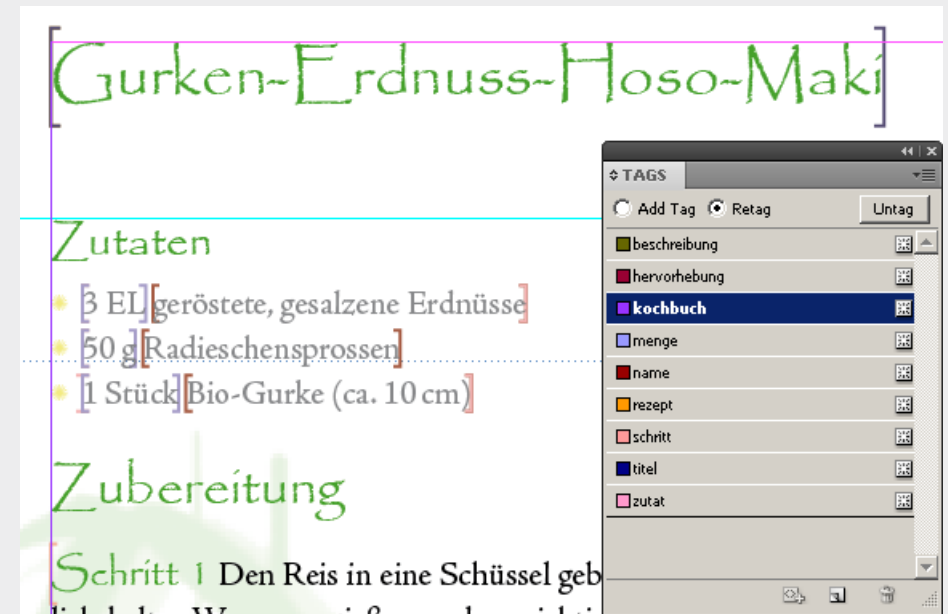
[STRG + Y]

Mit eingblendeten Tag-Marken gute Ansicht der Textauszeichnung

```
<rezept> <titel> Gurken-Erdnuss-Hoso-Maki </titel>

Zutaten
<zutat> <menge> 3 EL </menge> <name> geröstete, gesalzene
Erdnüsse </name> </zutat>
<zutat> <menge> 50 g </menge>
<name> Radieschensprossen </name>
</zutat> <zutat> <menge> 1 Stück </menge> <name> Bio-Gurke (ca.
10 cm) </name> </zutat>

Zubereitung
<schritt> Schritt 1 Den Reis in eine Schüssel geben,
reichlich kaltes Wasser zugießen und vorsichtig mit den
Händen rühren. Dabei <hervorhebung> nicht zu
kräftig </hervorhebung> rühren, sonst brechen die Körner. Das
milchige Wasser abgießen. Diesen Vorgang noch drei- bis
viermal wiederholen, bis das Wasser fast klar
bleibt. </schritt>
```



InDesign Satzautomation

Übungsaufgabe

XML Funktionen

Öffnen Sie die Datei 03_rezept-beispiel.indd in InDesign

! Machen Sie sich mit der **Strukturpalette** und den **XML-Ansichtsoptionen** vertraut

InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

Grundfunktionen

Elemente können mit Absatz- und Formatvorlagen **automatisch formatiert** werden.

- **Formate zu Tags** zuordnen

Flyout Strukturpalette -> Formate zu Tags zuordnen

Formaten werden Tags zugeordnet.

- Tags können auf Formaten zugewiesen werden

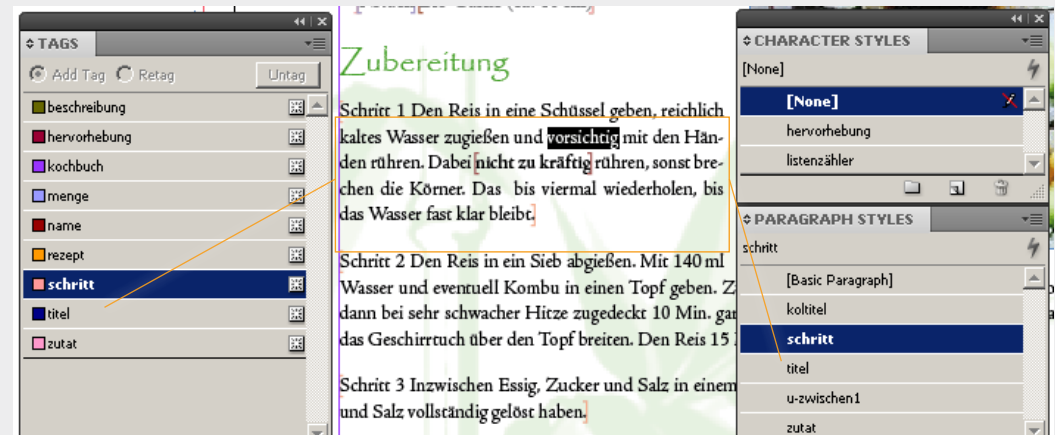
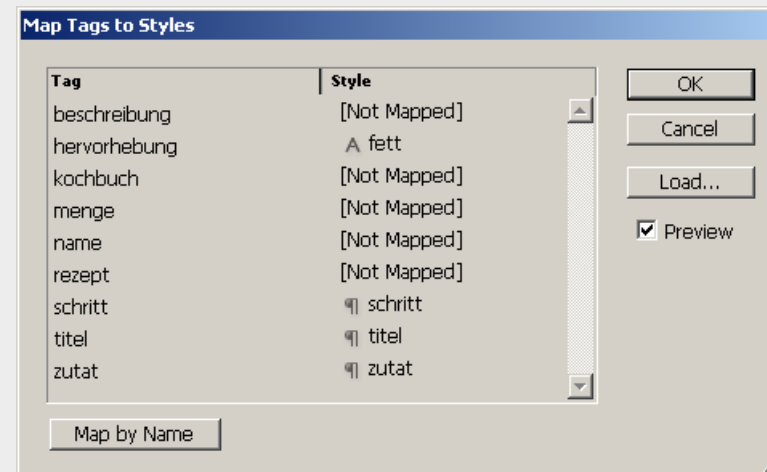
- Weitere Möglichkeiten:

- XML-Rules
- Mit einer XSL-Transformation Elementen über **Attribute** Styles zuordnen.

Attribut	Beschreibung
----------	--------------

aid:cstyle	Zeichenformat
------------	---------------

aid:pstyle	Absatzformat
------------	--------------



- Namespace nicht vergessen:

xmlns:aid="http://ns.adobe.com/AdobeInDesign/4.0/" [InDesign CS2]

xmlns:aid5="http://ns.adobe.com/AdobeInDesign/5.0/" [InDesign CS3]

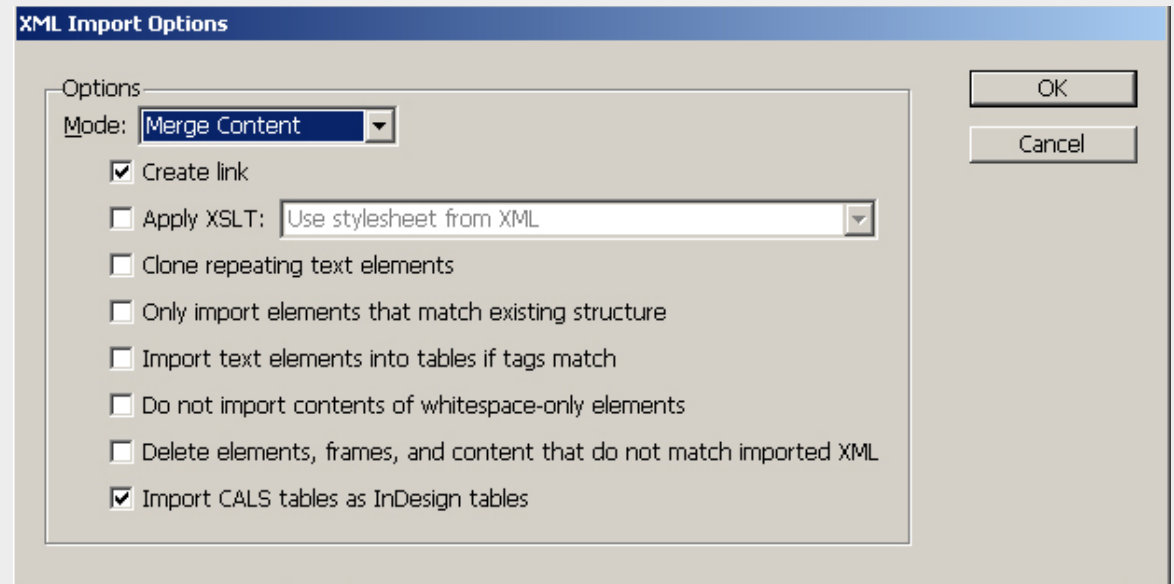
InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

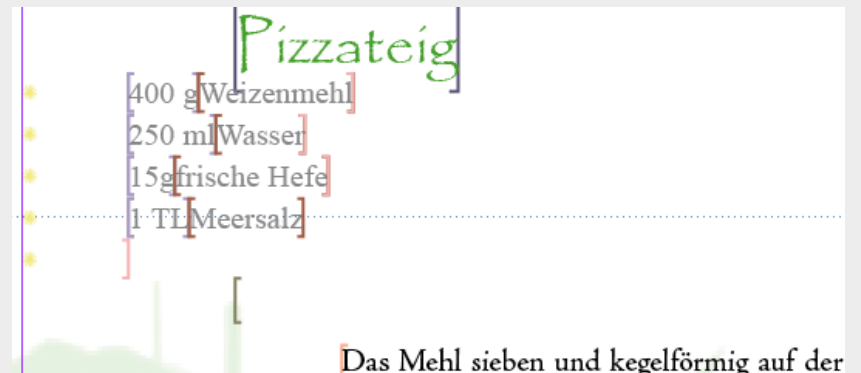
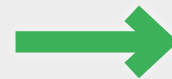
XML Import

Datei -> XML importieren

- Viele kaum brauchbare Funktionen
 - **XSLT Anwenden**
nur Subset von 1.0
 - **Wiederholte Textelemente kopieren** nur für stark regelbasierte Layouts
 - **Whitespace** wird vollständig **importiert** und dargestellt **oder** wird vollständig ignoriert.



```
<cochbuch>
  <rezept id="i1">
    <titel>Pizzateig</titel>
    <zutat><menge>400 g</menge><name>Wei
    <zutat><menge>250 ml</menge><name>We
    <zutat><menge>15g</menge><name>frische
    <zutat><menge>1 TL</menge><name>Meer
  </zutat>
```



Um das Whitespace Problem zu umgehen kann ein angepasstes Importskript verwendet werden.
Download unter: <http://www.indesignblog.com/>

XML Import

Öffnen Sie die Datei 04_rezept-template.indd in InDesign

! Importieren Sie die Datei 01_kochbuch_beispiel.xml

Verknüpfen Sie Tags und Formate

Machen Sie sich mit den verschiedenen Möglichkeiten des Imports vertraut.

XML Skripting

! **Analysieren** Sie das Import Skript.

! Mit dem folgenden Code können Container Elemente in eigene Textrahmen + Objektstil umgesetzt werden:

```
_node.insertTextAsContent("\r", XMLElementPosition.afterElement);  
var _atf = _node.placeIntoInlineFrame ([15,50]).getElements()[0];  
_atf.appliedObjectStyle = _oStyle;  
_atf.fit(FitOptions.frameToContent);
```

! **Erweitern** Sie den Import für Objektstile.

Tabellen

- InDesign unterstützt das CALS-Tabellenformat und sein eigenes Tabellenformat.
- **InDesign Tabellenformat** Eher Softwarenah? Aufwändige Konvertierung notwendig.
- **CALS Tabellen** OASIS Standard, kein xhtml aber Konvertierung vorhanden.
www.oasis-open.org/specs/tablemodels.php
- Formatzuweisungen über **Tags zu Formaten** zuweisen
- Formatierung mit einer XSL-Transformation über **Attribute**:

Attribut	Beschreibung
aid:table	defines that the element is a table.
aid:trows	total number of rows the table contains
aid:tcols	total number of columns the table contains
aid5:tablestyle	Table Style that formats the table [ab CS3]
aid:thead	defines cells as being part of a header row.
aid:tfooter	defines cells as being part of a footer row
aid5:cellstyle	Cell Style that formats table cells [ab CS3]
aid:crows	number of rows across which the cell spans
aid:ccols	number of columns across which the cell spans
aid:ccolwidth	column width of cell, measured in points.

Bilder

Bilder werden automatisch gesucht wenn ein href Attribut vorhanden ist

- **Rudimentäres Handling** über href Attribut.

```
<bild href="file://path/images/testfile.jpg" />
```

Nur Import, Keine Größen, Ausschnitte etc.

- Nur **verlinkte Bilder** möglich – Einbetten ist grundsätzlich nicht empfehlenswert.

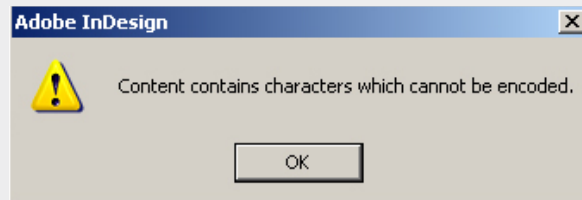
InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

XML Export aus InDesign

■ Remap Special Characters

Spezialzeichen werden als Leerzeichen ins XML geschrieben.

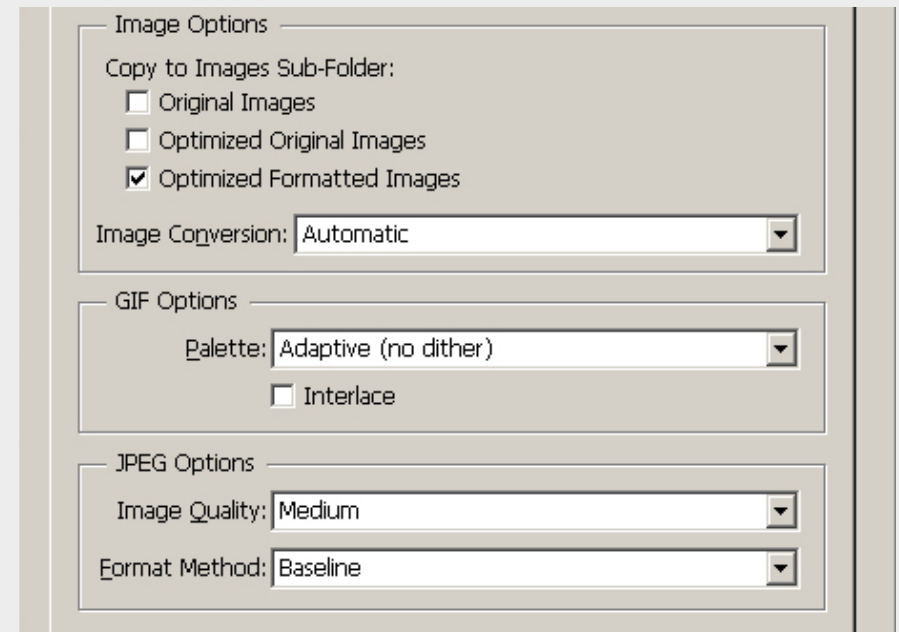
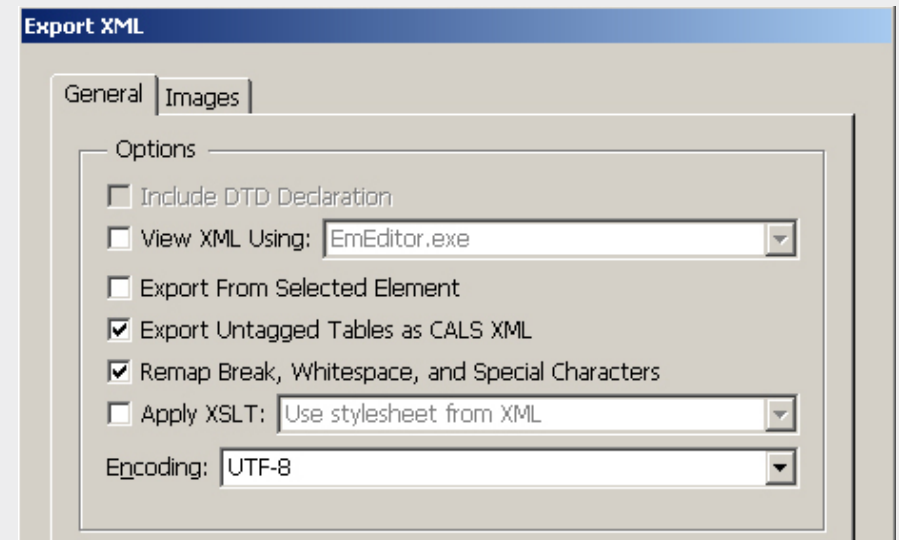


■ Tabellen

Tabellen können als CALS-Tabellen exportiert werden

■ Bilder

Werden auch exportiert, Downsampling möglich.



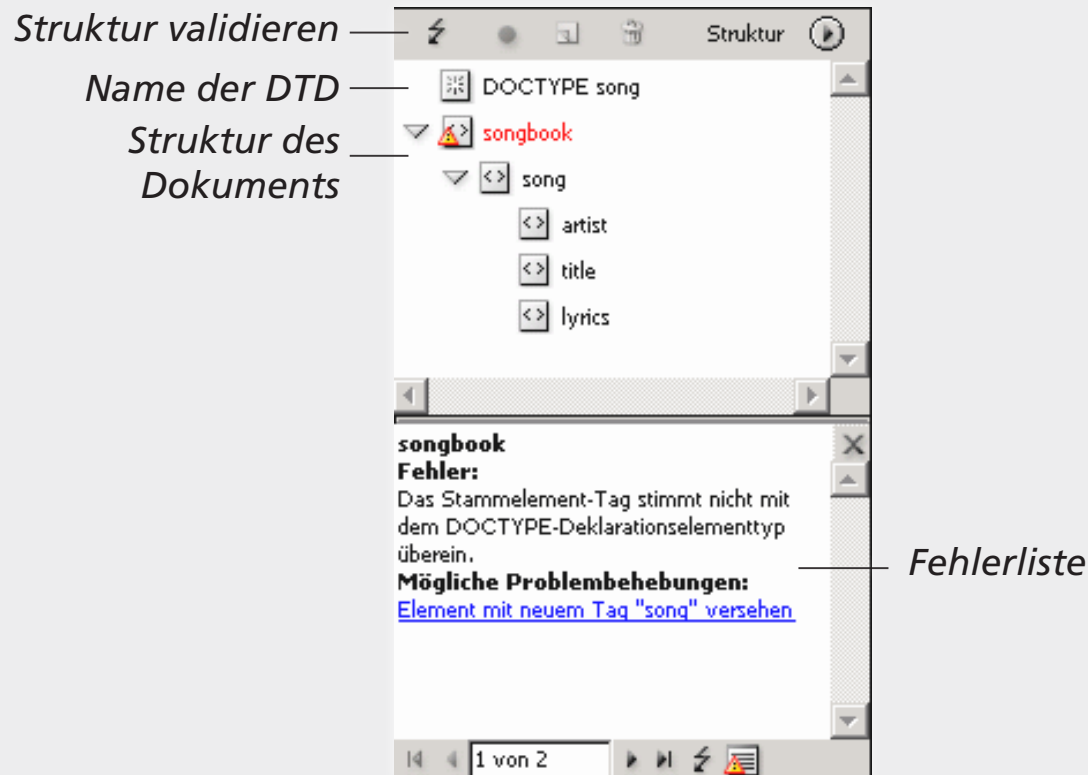
Weitere XML-Funktionen

■ DTD importieren

InDesign kann eine DTD (Document Type Definition) importieren und gegen diese validieren.

■ DTD **validieren**

Ab dem Stammelement (Root-Element)
oder ab einem ausgewählten Element gegen die DTD validieren.



XML Fazit

- Struktur muss in InDesign **abbildbar** sein. XML-Daten müssen **vorhanden** sein, bzw. **generiert** werden können.
- Dokumente können danach wie gewohnt weiterverwendet werden, Layoutfeintuning durch Mediengestalter
- **Bearbeitungsmöglichkeiten** verbesserungswürdig.
- **Export** nicht optimal – Nachbearbeitung oftmals notwendig.
- **XHTML** Export oder **EPub** Export besser als vermutet
Weitere Transformationen anhand dieser Daten
- Anleitung für EPub: <http://www.sachaheck.net/blog/indesign/indesign-ebook-leitfaden>

Zukunft für InDesign mit XML?

- Entwicklung in CS5 enttäuschend
- Adobes Strategie für Medienbrüche setzt nicht auf crossmediale XML-Workflows
 - Flash und Interaktivität vs. EBook bzw. fließende Formate
 - Doppelproduktion realistischer?
- Creative Suite vs. Technical Suite (FrameMaker)

XML und Scripting

XML und Scripting ergänzen sich sehr gut. Es können fast beliebige Layouts aus strukturierten Daten erstellt werden.

Die Verwendung von XML beim Layoutaufbau bietet ein hohes Automatisierungspotenzial

- **XML-Daten** sind strukturiert, wiederkehrend, vorhersehbar.
- **Scripte** benötigen gleiche Strukturen um effizient arbeiten zu können.
- XML-Unterstützung kann erheblich verbessert und erweitert werden.
- Mit **XML-Rules** und **XPath** können kontextabhängige Abfragen definiert werden.

XML und Skripting

- XML Dokument **importieren**:
`Document.importXML (from: File)` die `xmlImportPreferences` vorher einstellen!
DTD laden über: `Document.importDtd (from: File)`
- **Root Element** ansteuern
`Document.xmlElements[0]`
- Element vs. MarkupTag
`XMLElement.markupTag.name`
- Selber **parsen**
`for (i = 0; i < XMLElement.xmlElements.length; i++) { ... }`
- **Attribute**
ansteuern: `XMLElement.xmlAttributes.itemByName (name: string)`
auswerten: `xmlAttributes.value();`
hinzufügen: `XMLElement.xmlAttributes.add(name: string, value: string)`
- XML Elemente **platzieren**
`XMLElement.placeXML(PageItem);` oder `placeIntoInlineFrame (dimensions [breite,höhe])`
- Seit CS4: XPath Ausdrücke auswerten:
`Array of XMLItem XMLElement.evaluateXPathExpression ("XPath als String")`

Rezepte formatieren

Im Ordner 15_xml_und_skripting befinden sich die Dateien 01_kochbuch_beispiel.xml und beispiel.indd.

- ! Verwenden Sie das Skript basics.jsx als Grundlage
- ! Adressieren Sie die einzelnen Objekte und machen Sie sich mit dem Umgang mit XML-Elementen vertraut.
- ! Formatieren Sie das Rezept mit Absatzformatvorlagen
- ! Fügen Sie bei jedem vegetarischem Gericht das Bild vegi.jpg hinzu.

XML-Rules

XML-Rules eignen sich dazu dynamisch Layouts zu erstellen und formatieren.

- **XML-Rule-Sets** sind kleine Scripte die bei bestimmten XML-Knoten Aktionen auslösen.
- **XPath** gesteuert – Wenn die Regel zutrifft, wird eine bestimmte Aktion ausgelöst.
- Gehen über das einfache Mapping von Stilen zu Tags hinaus
- **Beispiele**
 - Erster Absatz nach einer Überschrift wird nicht eingerückt
 - Bei einer bestimmten Hierarchie-Ebene wird eine neue rechte Seite erstellt
 - Infokasten oder Katalogrecord ansteuern

Exkurs: XPath

Die **XML Path Language** (XPath) ist eine Abfragesprache um Teile eines XML-Dokumentes zu adressieren.

- Grundlage für **XSLT** – wird für das Template Matching verwendet.
Beim transformieren von Baumstrukturen bietet sich eine Syntax für die Selektion von Teilbäumen an
- Ein XPath-Ausdruck setzt sich zusammen aus:
 - Folge von Lokalisierungsschritten (mit dem /-Zeichen getrennt)
 - Optional können Filter zur Reduktion des Ergebnisses verwendet werden
 - Führender Slash (/) navigiert immer zum Document Root
- In XML-Dokumenten navigieren: Ein **Lokalisierungsschritt** besteht aus Achse und Knotentest
- Beispiele für XPath Ausdrücke
 - `/dok/kap` selektiert alle kap-Elemente innerhalb eines dok Elements
 - `/dok/kap[1]` selektiert das erste kap-Element innerhalb eines dok Elements
 - `//abs` selektiert alle abs-Elemente auf allen Ebenen
 - `//kap[@freigabe="true"]` selektiert alle kap-Elemente deren Attribut freigabe = true ist

XML-Rules HowTo

Für XML-Rules benötigt man:

1. XML-Rule Prozessor (scripting object)
2. Glue Code Scripts (von Adobe ausgeliefert)
3. Die XML-Rule

Anatomie einer XML-Rule

1. Eindeutiger Rule-Name
2. XPath Ausdruck
`/adressbook/adress/name/
titel`
3. Funktion die auf die Ergebnisse
angewendet wird
`this.apply function ...`

Codebeispiel:

```
function RuleName() {  
    this.name = "RuleNameAsString";  
    this.xpath = "XPath";  
    this.apply = function (elem, ruleSet, ruleProc){  
        //Do something here.  
        //Return true to stop further processing  
        of the XML element  
        return true;  
    }; // end of Apply function  
}
```

Glue Code und drumherum

Code von Adobe der das Handling von XML-Rules vereinfacht

- Der Glue Code muss eingebunden werden:

```
#include „glue code.jsx“
```

- Ein Rule Set erstellen

```
var ruleSet = new Array (new nameXMLRule1 ,new nameXMLRule2);
```

- Rule Set auf einem XML-Element aufrufen

```
__processRuleSet(root, ruleSet)
```

- Falls Elemente verschoben oder gelöscht werden:

```
__skipChildren(ruleProcessor)
```

Innerhalb der apply Funktion vor dem Löschen aufrufen!

- Innerhalb der Apply Funktion

```
return true; //Return true to stop further processing of the XML element
```

```
return false; //Let other XML rules process the element.
```

- Praktische Funktionen:

```
XMLElement.insertTextAsContent("\r", XMLElementPosition.afterElement);
```

```
XMLElement.applyParagraphStyle(ParagraphStyle);
```

Erstes Beispiel:

Die Daten liegen im Ordner 20_xml_rules

Fußnoten aus der XML-Struktur in InDesign Fußnoten umwandeln. Beispieldatei 01_footnote.indd laden und Skript XML2Footnotes.jsx analysieren.

XML Rules für den Bildimport

Daten 02_text.indd, 02_text_mit_abb.xml und PlaceAbbildung.jsx

- ! Schreiben Sie zunächst ein Skript das beim ersten Absatz nach einer Überschrift den Einzug entfernt.
- ! Erstellen Sie eine XML-Rule die das Bild aus dem Element abb importiert.
- ! Beim Bildimport sollen die Attribute für Höhe, Breite ausgewertet sowie eine Bildlegende automatisch erstellt werden.

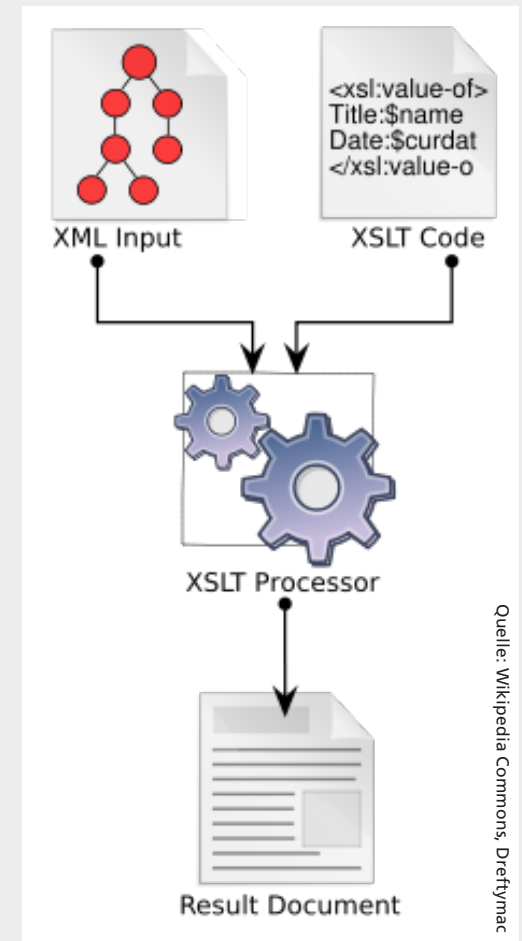
Exkurs: Extensible Stylesheet Transformation Language (XSLT)

*XSLT ist eine **Programmiersprache** zur Transformation von XML-Dokumenten, Teil der Extensible Stylesheet Language (XSL).*

- **XSLT-Stylesheets** sind nach den Regeln des XML-Standards aufgebaut.
- Baut auf der logischen **Baumstruktur** eines XML-Dokumentes auf.
- **XSLT-Prozessoren** (Saxon, MSXSL, ...) verarbeiten XML-Dokumente anhand der im XSLT-Stylesheet definierten Regeln zur gewünschten Ausgabestruktur.

Notwendigkeit von XML-Transformationen

- Vielfältige XML-Datenformate von verschiedensten Programmen verarbeiten.
- **Datenanalyse** und **Dateninterpretation**
- **Split/Join** von Strukturen zur Weiterverarbeitung
- Umstrukturierung
- ...



Funktionen erweitern

Die XML-Funktionen von InDesign sind leider nicht besonders weitgehend.

- Blocklevel-Elemente aufteilen
- Character Data innerhalb von Element only Kontexten

XML-Rule für Blocklevel Elemente

Die Bearbeitung von XML innerhalb von InDesign ist verbesserungswürdig.

- ! Versuchen sie zunächst ein Blocklevel Element in Zwei Teile aufzuteilen
- ! Versuchen Sie eine Funktion zu schreiben, die das XML-Element korrekt zerteilt.