

InDesign

OCF  
XML

IDMS

IDML

Transformation

Workflows

XSLT

ICML

# Was kann IDML denn so?

20. November 2015

#markupforum

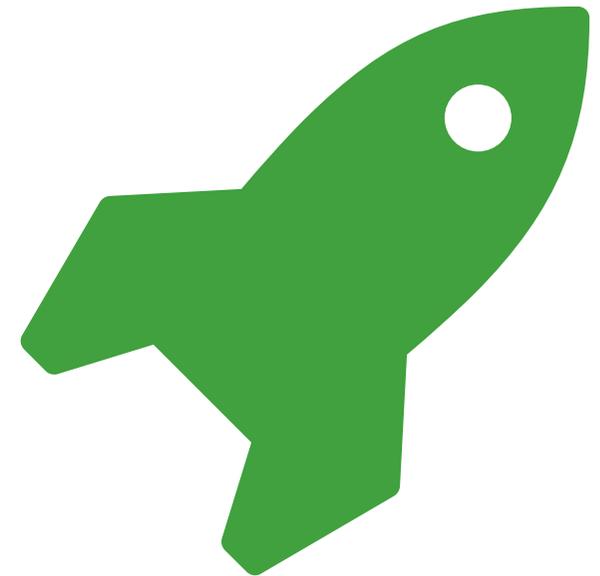
Kontakt: [gregor.fellenz@publishingx.de](mailto:gregor.fellenz@publishingx.de)

Folien: <http://www.publishingx.de/dokumente>

# InDesign

„The Greatest Software Product of All Time“

*Max Dunn, 2013*



## Killer feature

Einfache, manuelle Bearbeitung von Dokumenten und Templates.

## Wo ist das XML versteckt?

- XML kann nativ in Dokumenten verwendet werden.
- Mit XML kann das Dateiformat generiert werden.
- Metadaten werden als XML gespeichert.
- Einige Voreinstellungen werden in XML gespeichert.



# Die IDML-Familie

Mit **InDesign CS4** wurde das datenzentrierte XML-Format **IDML** (InDesign Markup Language) eingeführt. Familie mit **IDMS**, **ICML** und **ICMA**.

## Verwendung

- **Austauschformat** zwischen verschiedenen InDesign-Versionen.
- **Katalogproduktion**: IDML aus XML-Daten bzw. Datenbanken automatisiert erstellen.
- **Templating** für Web2Print-Lösungen.
- **Übersetzungsworkflows** und Translation Memory Systeme (TMS).
- **Snippets** und **InCopy**

## Designprinzip

Eine vollständige Repräsentation des InDesign Dateiformats in XML.

- Lesbar im Texteditor.
- Ohne gerenderte Voransichten.
- Ohne Umbruchinformationen der Render-Engine.
- Automatisierter Aufbau von **Dokumenten** mit XML-Daten.
- Maßeinheit ist immer **Points** (= 72 pt pro Inch).
- Minimale Angaben reichen aus (IDM-Voreinstellungen).
- Erstellt sich aus den gleichen Informationen wie das Scripting Object Model.

## IDML, IDMS, ICML und ICMA – Was, wann wozu?

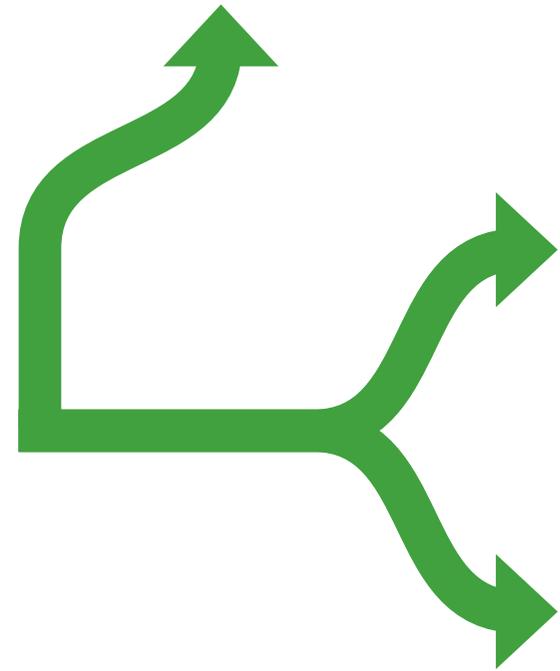
- **IDML** (InDesign Markup Language)  
ZIP-Container.  
Vollständiger Seitenaufbau möglich. Gepackt als ZIP-Container.
- **IDMS** (InDesign Markup Snippet)  
Zur Wiederverwendung von in InDesign gestalteten Objekten.
- **ICML** (InCopy Markup Language)  
Inhalte in InCopy bearbeiten.
- **ICMA** (InDesign Markup Assignment)  
Hier können Inhalte einem Autor zur Bearbeitung in InCopy zugewiesen werden.
- **IDPP** (InDesign Preflight Profile)  
InDesign Preflight-Profile.

# Datenformat IDML

## Aufbau

ZIP-Container, der verlinkte XML-Dateien enthält.

- Die Typdefinition **mimetype**
- Übersichtsdatei **designmap.xml**
- Dateiaufbau (Oasis OCF) im Ordner **META-INF**
- Eine Datei pro Musterseite im Ordner **MasterSpreads**
- Infos zu Formaten, Schriften, Farben, Voreinstellungen im Ordner **Resources**
- Die Inhaltsseiten und deren Objekte im Ordner **Spreads**
- Der Eigentliche Inhalt im Ordner **Stories**
- Die XML-Tags und BackingStory des Dokuments befinden sich im Ordner **XML**



## IDML Defaults

IDML-Dateien können recht kurz geschrieben werden.

- Nur notwendiges muss definiert werden!
- Alle nicht enthaltenen Informationen werden der Voreinstellungen-Datei entnommen.  
`InDesign Programmordner/Presets/Default/Predef.iddx`
- Entspricht einer Neuinstallation und garantiert damit konsistente Ergebnisse unabhängig von den Benutzereinstellungen.

# Validierung



- Relax NG Schema

## Es gibt kein Standard Schema

IDML bildet immer die installierten Plugins ab,  
d.h. man muss ein Schema für die InDesign-Instanz erstellen.

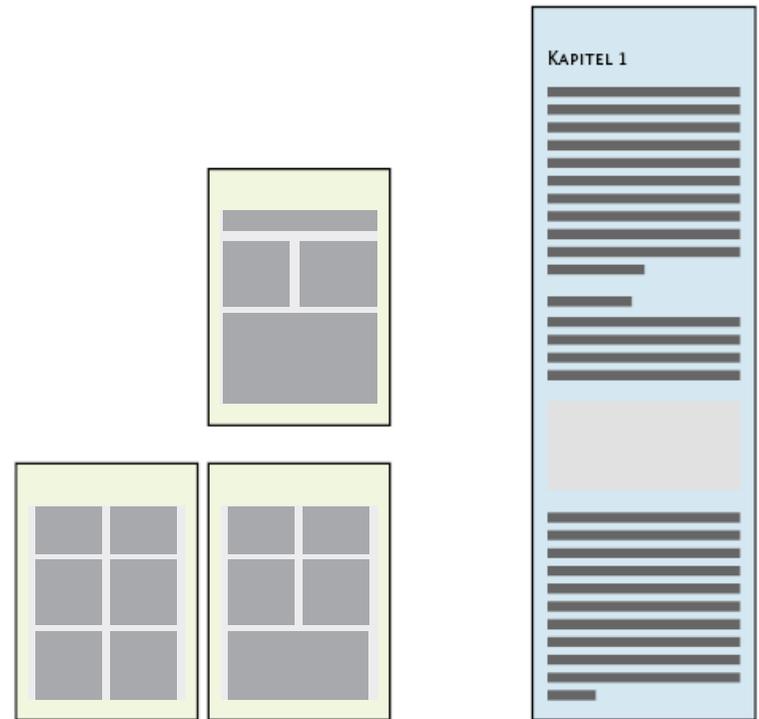
```
app.generateIDMLSchema(Folder("/idml-schema/snippet"), false); // Single File
```

- Validierung von Einzeldateien

```
app.generateIDMLSchema(Folder("/idml-schema/package"), true); // Container
```

- Container Format validieren mit IDMLTools
- CS6 <http://www.adobe.com/devnet/indesign/sdk/>
- CC Bewerbung für das *InDesign SDK Access Program*

```
validate.bat "c:\idml-schema\package" test.idml
```



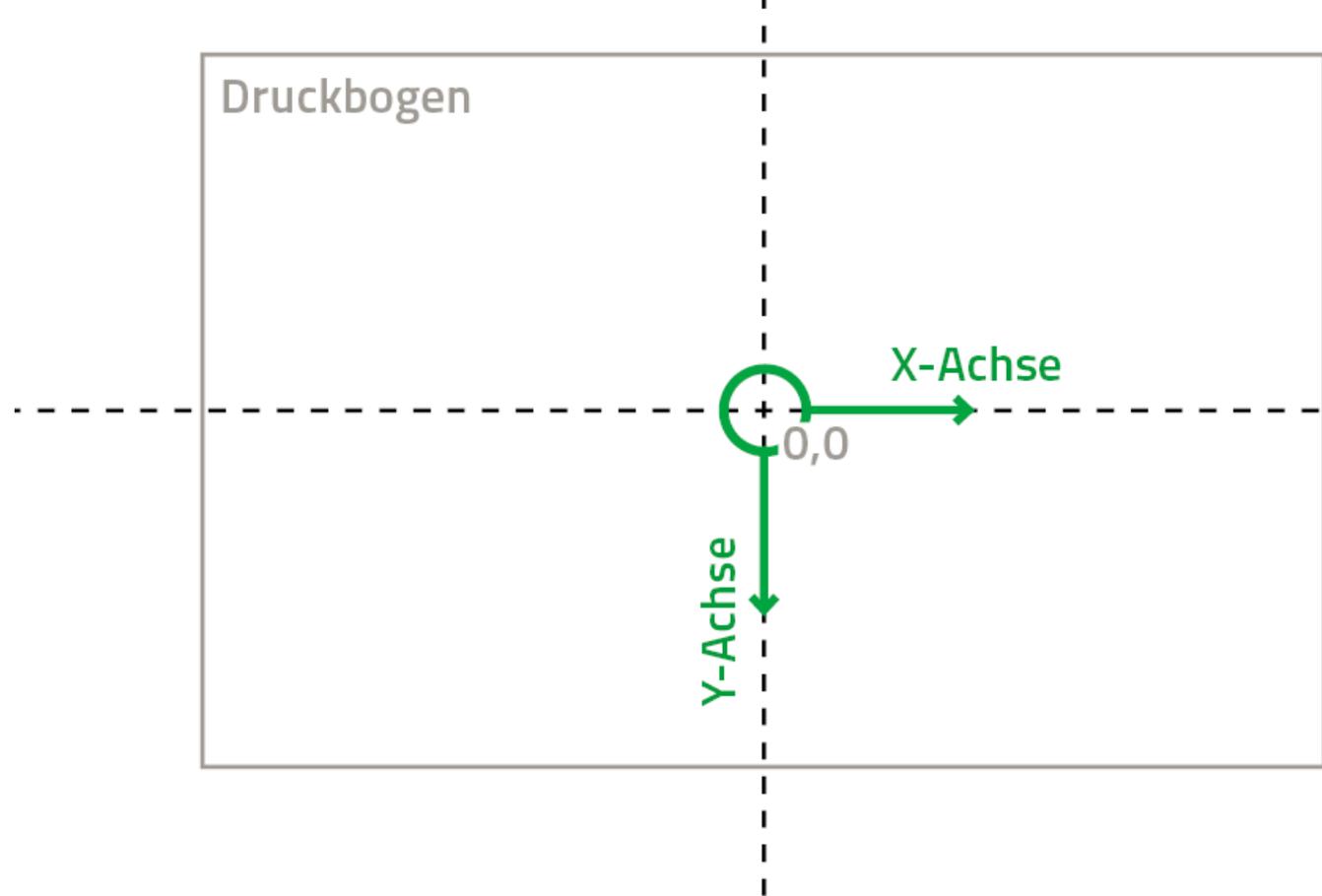
# Was geht nicht so gut?

- Die Render-Engine ist bei der Erstellung nicht verfügbar.
- Layoutabhängige Berechnungen schwierig oder unmöglich.
- Lösung: **Mehrere Durchläufe** oder **Skripting**.



# Ein Snippet in der Detailansicht

- Dateiformat **IDMS**
- Beim Export werden alle **Layoutinformationen**, die für den Aufbau relevant sind, exportiert.
- Zusätzlich noch ein **Vorschaubild**.
- Die meisten Informationen werden aus dem Dokument, in dem das Snippet platziert wird, übernommen.

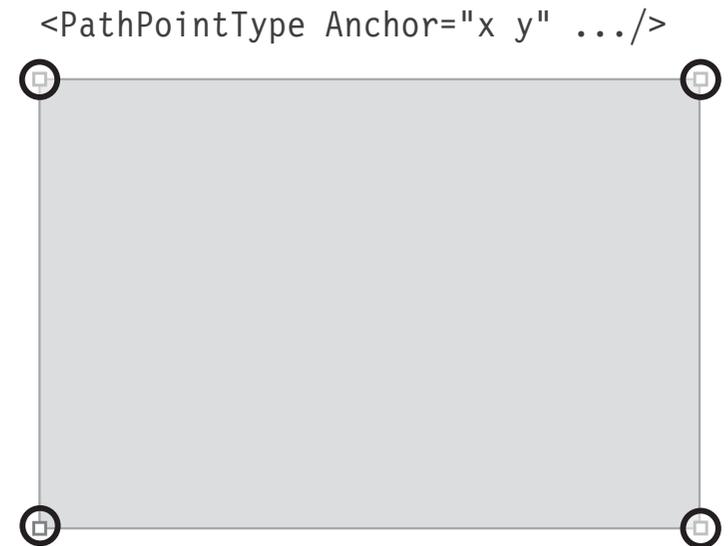


# Objekte bauen

- Generell gehören alle Seitenobjekte wie Textrahmen oder Rechteck zum Druckbogen `<Spread>` – nicht zu `<Page>`.
- **Koordinaten**
  - Mittelpunkt: Zentrum des Spreads – Problem mehrere Seiten pro Spread.
  - Alles ist in Punkt: Umrechnung:  $(\text{mm} / 25.4) * 72$

## Rechteck

- Element `<Rectangle>`
  - Position wird über `<PathPointArray>` festgelegt.
  - Rechteck mit vier Punkten `<PathPointType Anchor="x y" .../>`
- Mit dem Attribut `ItemTransform="MATRIX"` anhand von Matrixtransformationen relativ zum Mittelpunkt des Spreads verändert/verschoben.
  - Identity: `ItemTransform="1 0 0 1 0 0"`
  - Verschieben: `ItemTransform="1 0 0 1 tx ty"`
  - Skalieren: `ItemTransform="sx 0 0 sy 0 0"`
  - Rotation: `ItemTransform="cos(θ) sin(θ) -sin(θ) cos(θ) 0 0"`





# Bild einladen

- Element `<Image>` als Kind von `<Rectangle>`
- Hier gibt es zur Abwechslung eine BoundingBox: `<GraphicBounds>`
  - Bilder werden automatisch proportional passend in die vorgegebene BoundingBox skaliert.
- Verknüpfung zur Bilddatei über Element `<Link LinkResourceURI="Path" .../>`

# Rahmen markieren

- Rahmen die beim **postprocessing** verarbeitet werden sollen, können mit Labels markiert werden.

- Kindelement von `<Properties>`

```
<Label>  
  <KeyValuePair Key="px:Type" Value="Bild"/>  
</Label>
```

- Auslesen via Skripting

```
OBJECT.extractLabel(labelName);
```



# Textrahmen

- Rahmenobjekt `<TextFrame>` (Aufbau wie `<Rectangle>`).
- Der eigentliche Text ist einem separaten `<Story>` Element.
- Verknüpfung über ID mit Attribut `ParentStory="ID"` von `<Story Self="ID"/>`

```
<ParagraphStyleRange AppliedParagraphStyle="ParagraphStyle/af">  
  <CharacterStyleRange AppliedCharacterStyle="CharacterStyle/zf">  
    <Content>Text</Content>  
    <Br/>  
  </CharacterStyleRange>  
</ParagraphStyleRange>
```

# Story

- Textabschnitte für einen oder mehrere (verknüpfte) Textrahmen.
- Eigenwilliger Aufbau von Bereichen die mit gleichen Formaten ausgezeichnet sind.
  - Elemente `<ParagraphStyleRange>` und `<CharacterStyleRange>`
  - Textinhalt im Element `<Content>`
  - Absatzende mit `<Br/>`
- Definition der Formate zu Beginn **notwendig!**

# ICML

- ICML enthält:
  - Inhalt eines zusammenhängenden Fließtextes (Textabschnitt).
  - Zugewiesene Zeichen-, Absatz- und Objektformate.
  - Platzierte Inhalte wie Abbildungen.
- ICML enthält nicht:
  - Seiten
  - Rahmen (verankerte Rahmenobjekte sind möglich)
- Ideal um manuell erstellte Templates mit textlichen Inhalt zu befüllen.
- Formateinstellungen der Datei werden vom Dokument **überschrieben**.


# Tabellenmodell

InDesign hat ein eigenes, etwas ungewöhnliches, Tabellenmodell.

- Im Element `<Table>` werden die Anzahl der Zeilen `BodyRowCount`, Spalten `BodyColumnCount`, Kopfzeilen `HeaderRowCount` und Fußzeilen `FooterRowCount` über Attribute spezifiziert.
- Alle Zellen hintereinander mit dem Element `<Cell>` erstellt. Die Position wird vom Attribut `Name` spezifiziert.  
Aufteilung: Spaltenzahl:Zeilenzahl (Zählung von 0)  
Beispiel `2:0` -> Position 2. Spalte 1. Zeile
- Zellen können mit den Attributen `RowSpan` und `ColumnSpan` verbunden werden.
- Zeilenhöhe und Spaltenbreiten können mit den Elementen `<Row>` und `<Column>` eingestellt werden.
- Zellen enthalten wiederum Texte die über `<ParagraphStyleRange>`.

# IDML Datenextraktion

Text- und Formatinformationen aus einer IDML-Datei auslesen.

Ziel: Weiterverarbeitung Web, EPUB oder Übersetzung.

## Workflow für Text

1. IDML Container entpacken.
2. Die Referenzen auf Spreads (Druckbögen) und Stories (Textabschnitte) aus der `designmap.xml` suchen und in zusammenführen.
3. Textdatenmodell auflösen und in flache Struktur überführen.
4. Block-Level Elemente gruppieren.

# Was geht sonst noch?

- **Skripting:** Aus IDMS oder IDML Properties heraussuchen.
- **Änderungsvergleich** Snippet oder IDML.
- Snippets als **Bausteine** für den Seitenaufbau.

## Tools

- idmlLib <http://idmlib.com>
- transpect <http://transpect.le-tex.de>
- ickmull <https://code.google.com/p/ickmull>
- DITA-OT <https://github.com/dita4publishers/org.dita4publishers.dita2indesign>
- Pressbooks <http://pressbooks.com>

**Vielen Dank für Ihre Aufmerksamkeit!**

**Fragen und Anregungen?**

E-Mail: [gregor.fellenz@publishingx.de](mailto:gregor.fellenz@publishingx.de)

Twitter: [grefel](#)

Blog: <http://www.indesignjs.de>