



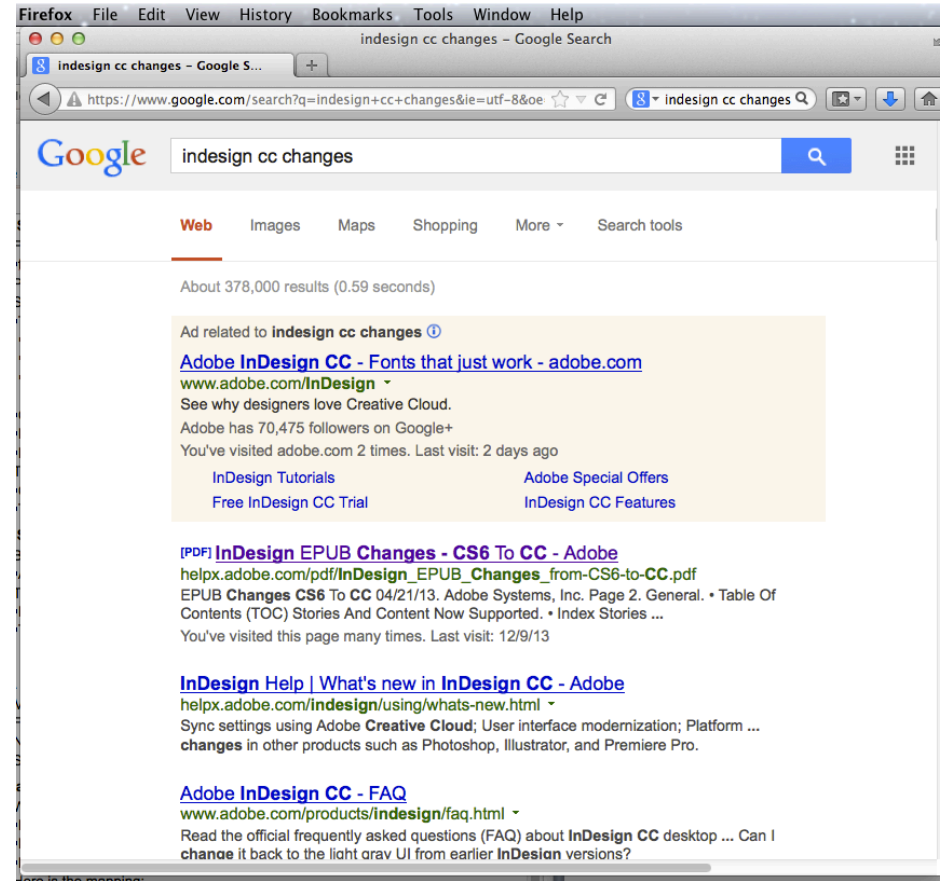
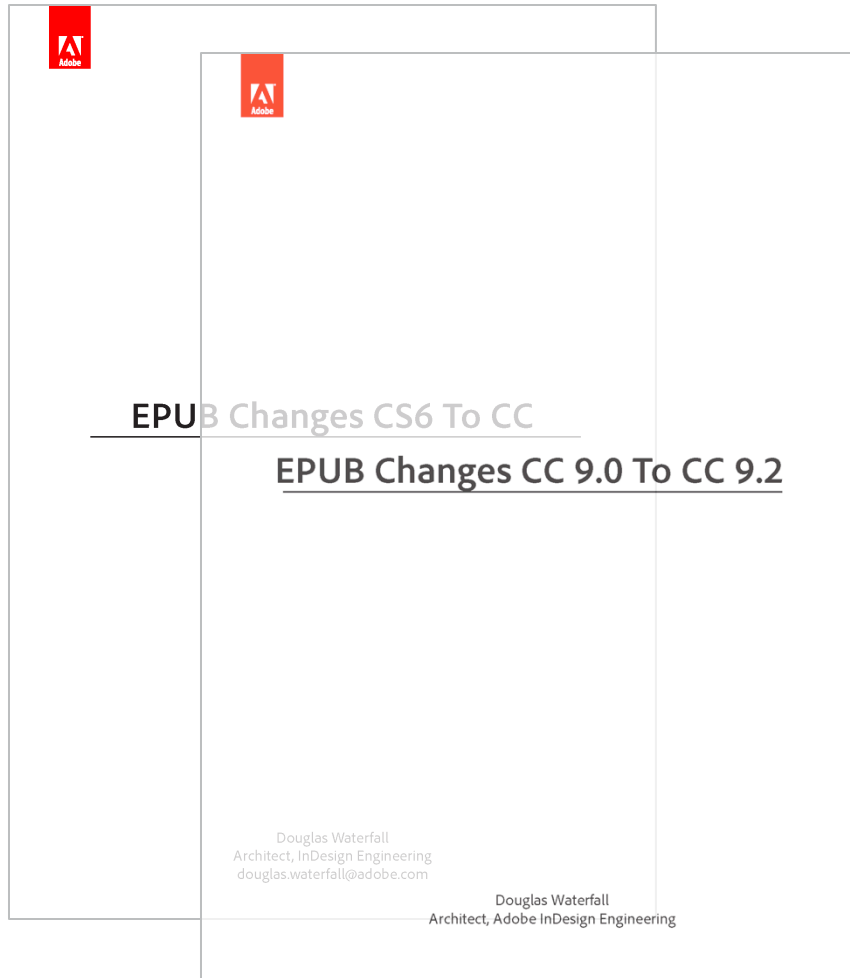
Advanced EPUB creation for iPad with Adobe InDesign CC

Douglas Waterfall | Architect, InDesign Engineering (douglas.waterfall@adobe.com)



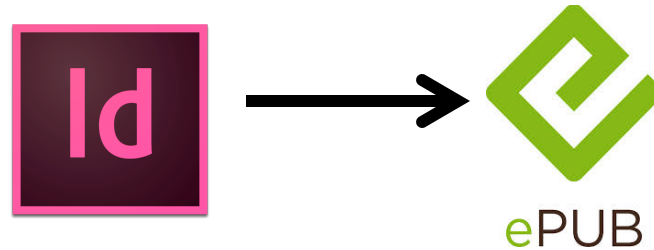
The Perils Of Asking An Engineer To Explain How It Works...

How To Find The List Of EPUB Changes In CC



helpx.adobe.com/pdf/InDesign_EPUB_Changes_from-CS6-to-CC.pdf
helpx.adobe.com/pdf/InDesign_EPUB_Changes_CC_9.0_to_9.2.pdf

Soon!



- The history of EPUB within InDesign
- Our Mapping Philosophy
- A Peek Behind the Curtain:
 - HTML Markup
 - CSS
 - Packaging
- Questions

The history of EPUB within InDesign

Continuing to build on more than seven years of EPUB creation...



InDesign CS3

2007

Initial support for EPUB export via script



InDesign CS4

2008

'Export for Digital Editions'



InDesign CS5

2010



InDesign CS5.5

2011

Complete rewrite of EPUB export to now be native feature of InDesign



InDesign CS6

2012

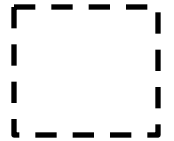
First EPUB3 support



InDesign CC (9.1)

2013

More than 30 new and enhanced EPUB features



InDesign CC (9.2)

2014

More Good Stuff

Our Mapping Philosophy

Our Mapping Philosophy (cont)

- Be realistic about what we can and cannot control
- Prefer vanilla EPUB over device specific versions
- Semantic markup is more important than *making it look like InDesign*
- Prefer 1:1 mapping of attributes to CSS properties
- Map as much of InDesign's document paradigm as we can
- Add EPUB specific controls over our markup & CSS & packaging

- *Continue to reduce the number of reasons you have edit the EPUB*

A Peek Behind The Curtain: HTML Markup

- `<body>` tag
- Objects & Groups
- Object Content

Markup: <body> tag

```
<body id="myINDDFileName" xml:lang="en-US" >
```

- What is up with the `id`?
 - Based on the filename of the INDD file
 - Why is it there?
- Language calculation

Markup: Objects & Groups: These Are All Objects

Above

Left Body+Green Body Right

Below#

253

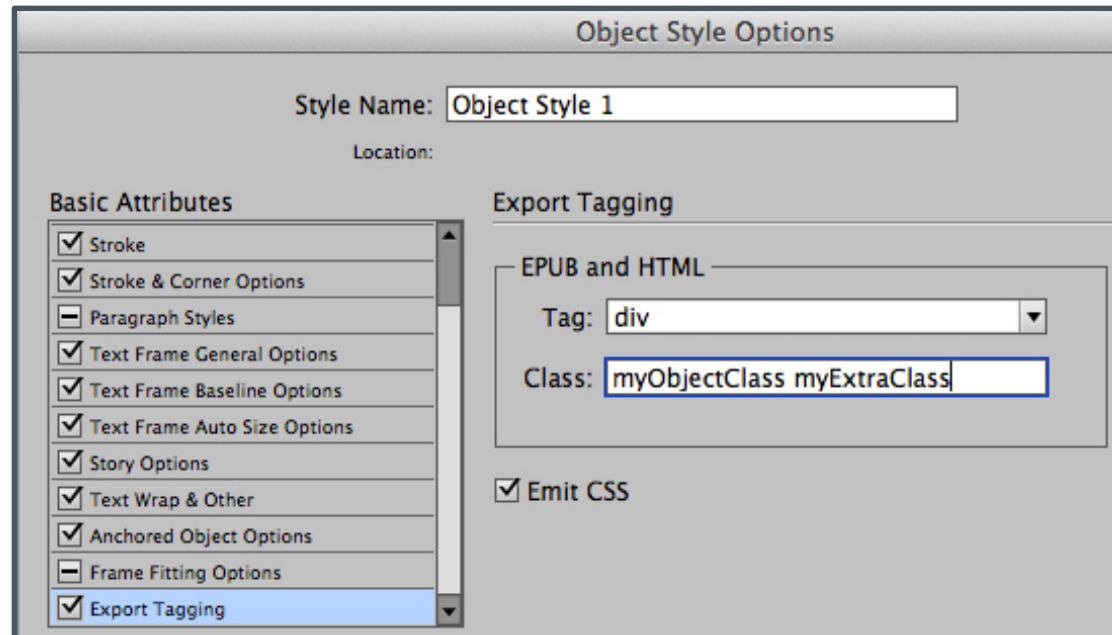
Text on the Path is Fun!

TV

Hello, World!

```
<div class="myObjectClass myExtraClass" >  
    some content  
</div>
```

- Tag and Class come from Object Style Export Tagging Tab
- Multiple classes allowed, only first will be used to Emit CSS





Graphics

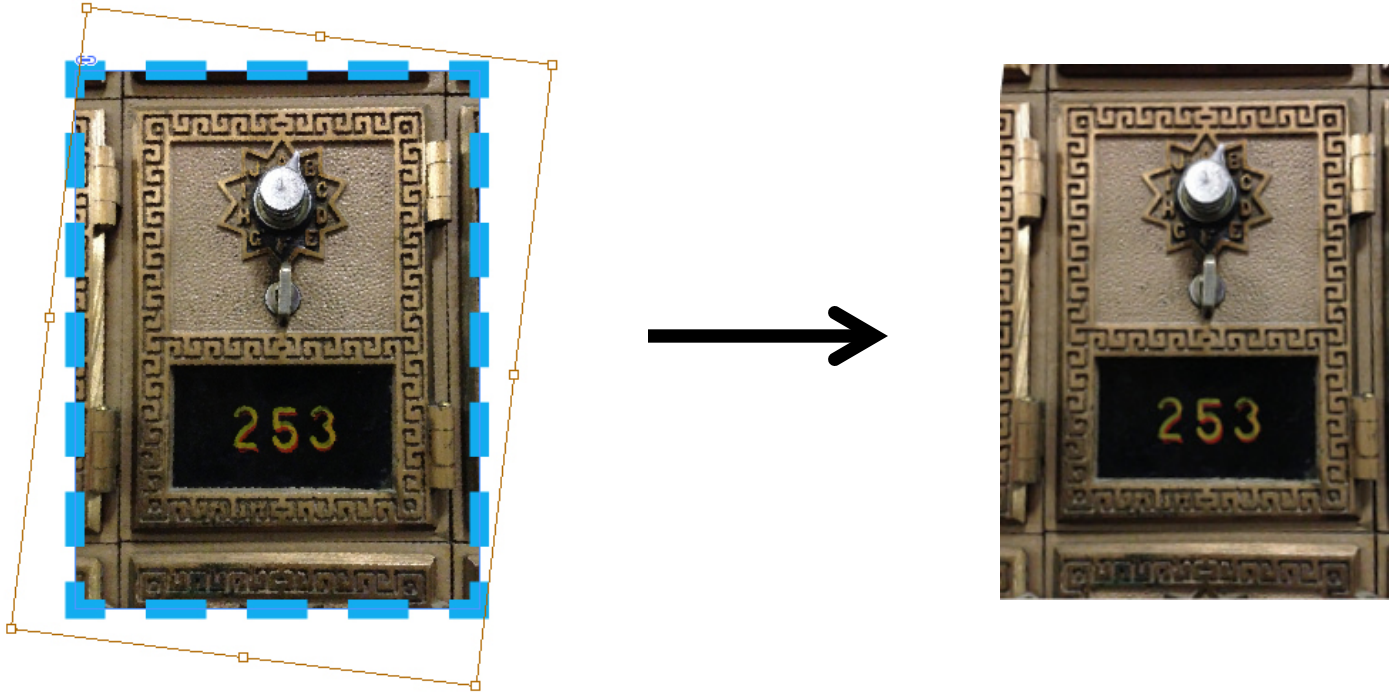
Audio & Video

Text

```
<div class="myObjectClass myExtraClass" >  
      
</div>
```

- Alt Text comes from the Object Export Options Dialog
- The Image is created from the Graphic to be EPUB compatible and with the desired resolution.

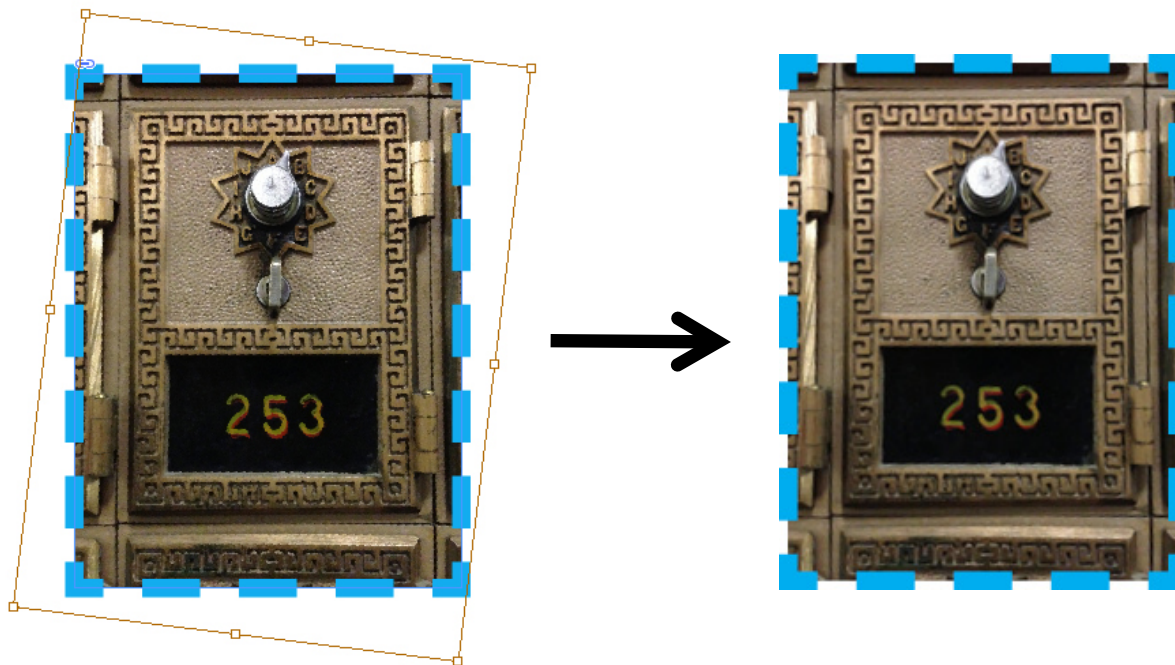
Markup: Graphic Conversion



- Controlled by Export Dialog Image Tab
- Rotation and Scaling flatten into the new image
- But happened to the blue stroke?

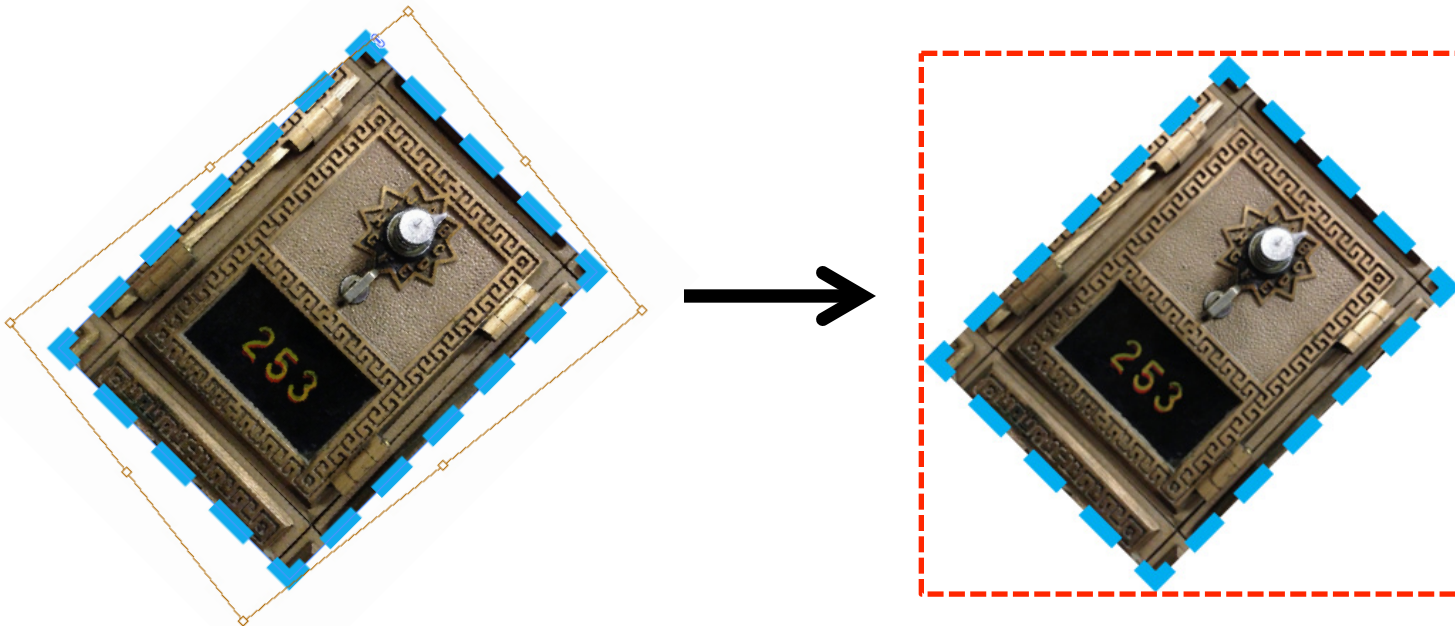
Markup: Graphics - Preserve Appearance From Layout

- Object and Content are different both in InDesign and HTML
 - The blue stroke was on the Object not the Graphic
- To make the generated image include the Object Graphic Attribute use *Preserve Appearance From Layout* in the Export Dialog Image Tab.
 - In some cases InDesign will do this automatically - ex: Text On Path on Image



Markup: Graphics – (More) Preserve Appearance From Layout

- What happens if the Object is Rotated?
 - Few EPUB2 Reading Systems support CSS transforms
 - Converted image (the red box) "burns in" the rotation
 - Next update no longer forces this automatically



Markup: Graphics – Why So Complicated?

- InDesign supports more Graphic formats than EPUB
- Graphics can be transformed and cropped in InDesign
- Objects can be transformed and contain fill
- Choices and Tradeoffs
 - Some advantages to having CSS handle stroke and fill on the Object
 - ...though we currently don't emit CSS for this
 - Next update will have an additional control to *Use Existing Image*

Markup: Audio & Video

```
<div class="myObjectClass myExtraClass" >
  <video id="AUD_MW_E.264.mp4" poster="image/243.jpg"
    autoplay="autoplay" controls="controls">
    <source src="video/AUD_MW_E.264.mp4" type="video/mp4" />
  </video>
</div>
```

- Poster and autoplay come from the Media Palette

```
<div class="myObjectClass myExtraClass" >
  <audio id="bear.mp3" autoplay="autoplay" loop="loop"
    controls="controls">
    <source src="audio/bear.mp3" type="audio/mpeg" />
  </audio>
</div>
```

- Autoplay and loop also come from the Media Palette
 - Poster appears *NOT* to be emitted – something wrong there

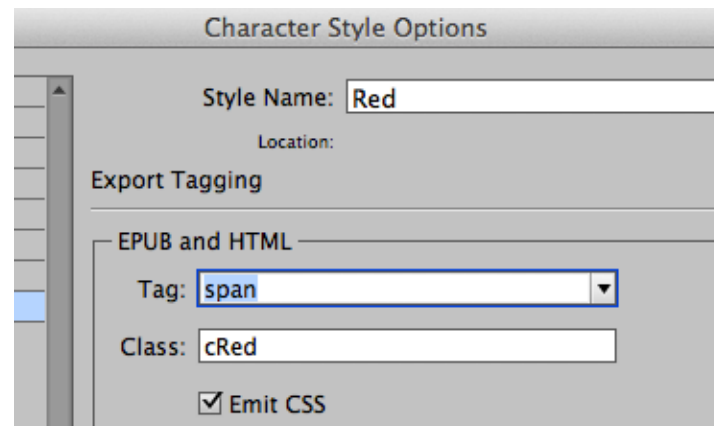
- Character Styles and Overrides
- Paragraphs, Paragraph Styles and Overrides
 - Nested Styles
 - Lists
 - Anchored Objects
- Tables
- Other...

Markup: Character Styles and Overrides

Hello, World!

```
<span class="cRed" {xml:lang=} >
  H
  <span class="CharOverride-1" >e</span>
  llo
</span>
World!
```

- Tag and class come from Character Style Export Tagging Tab
- User applied overrides are named *CharOverride-#*
- `xml:lang` is applied as tag attribute as needed

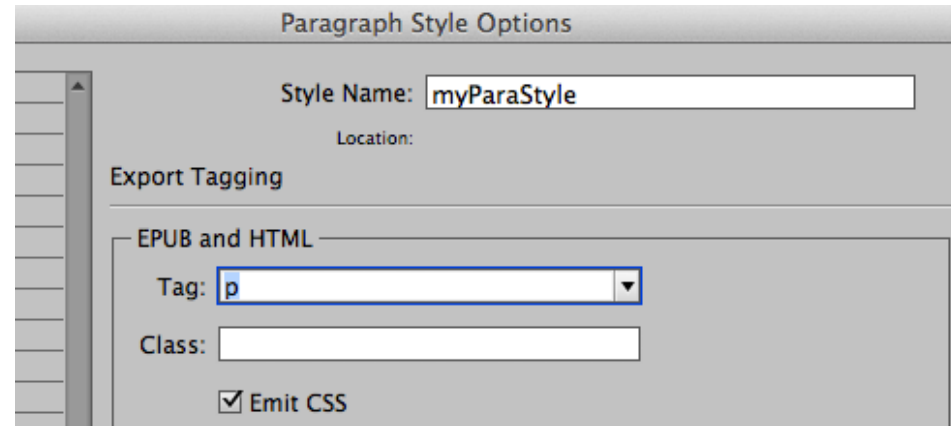


Markup: Paragraphs, Paragraph Styles and Overrides

Hello, World!

```
<p class="myParaStyle ParaOverride-1"  
    {xml:lang=} {dir=}>  
    Hello, World!  
</p>
```

- Tag and class come from Paragraph Style Export Tagging Tab
 - If no Class specified, the Style Name is used.
- User applied overrides are named *ParaOverride-#*
- `xml:lang` and `dir` are applied as tag attributes as needed



Markup: Paragraph/Nested Styles

Hello, World!

```
<p class="myParaStyle" >  
    <span class="cRed cItalic" >Hello</span>, World!  
</p>
```

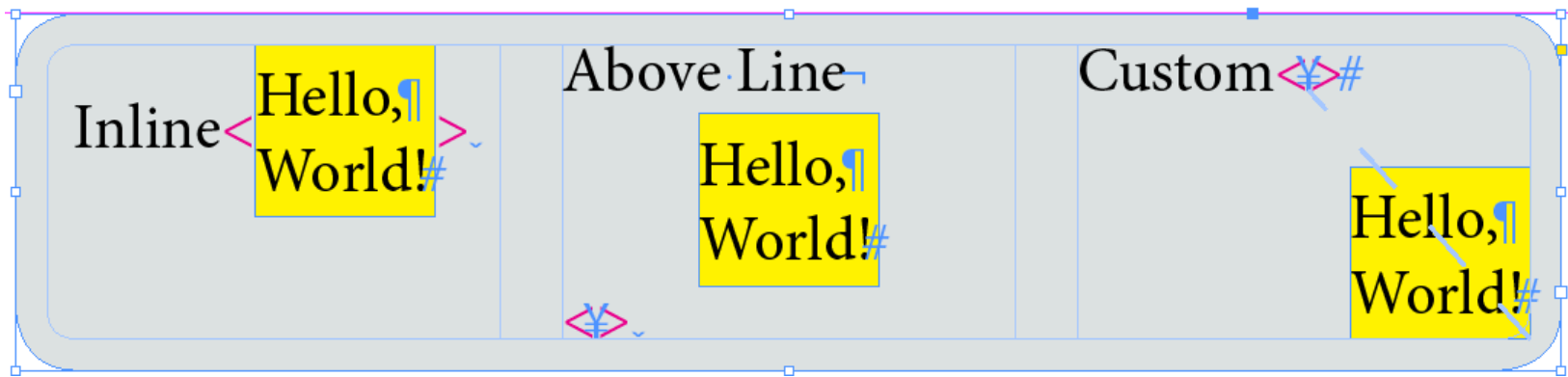
- Nested Styles create the opportunity for more than one Character Style to be applied on the same span.
 - Nested Style applies Character Style "cRed" to first word
 - User applies Character Style "cItalic" to same word
 - User Character Overrides could have been applied too
- Which wins if both Character Styles specified a color...?

- Bullet 1
 - Nested Bullet
- Bullet 2

```
<ul>
  <li class="bl">Bullet 1
    <ul>
      <li class="nbl">Nested Bullet</li>
    </ul>
  </li>
  <li class="bl">Bullet 2</li>
</ul>
```

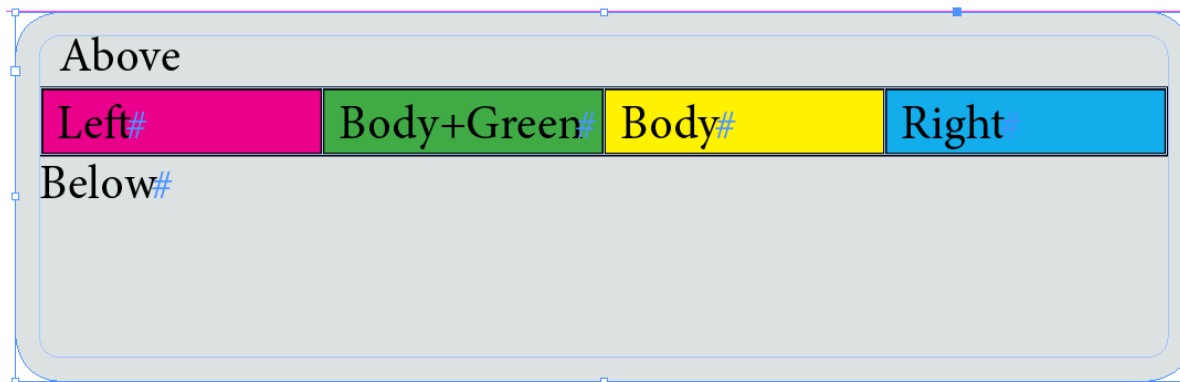
- InDesign Lists are controlled by a set of Paragraph Attributes
 - There is no internal hierarchy/level tracking mechanism
 - We calculate the hierarchy with a heuristic based on the Left Indent and the List Attributes - do not use Tabs!

Markup: Paragraph/Anchored Objects



- Three types of Anchored Objects – Inline, Above Line and Custom
 - (Red <> shows the position of the anchor)
- HTML markup rules constrain what we can do - no <div> in the middle of paragraphs only
 - Above Line and Custom *have* to be moved outside of the Paragraph
 - An example where we cannot maintain semantic mapping
 - Inline Objects must be rasterized as if *Preserve Appearance From Layout* – including Text Frames!
 - What the difference between an Inline *Glyph* and Inline Graphic?
 - Be nice to have a control to deal with this ambiguity

Markup: Tables



- Tables cannot be a child of a `<p>` tag, same problem as `<div>`s
 - *Have* to be moved outside of the Paragraph thus we cannot maintain semantic mapping
- No Export Tagging support for Tables yet
 - No tag control needed anyway
 - Class named based on Table/Cell Style name
- Table Styles can apply Cell Styles based on *Regions*, here we have *left column (red)*, *right column (blue)* and *body rows (yellow)*
- User can also apply a Cell Style too – here it is **green**.
 - So more than one Cell Style/Class is possible, plus Cell Overrides too

Markup: Other Text Markup

- Hyperlinks
- Cross References
- Japanese Specific
 - Ruby
 - Tate-chu-yoko (TCY)

CSS creates the visual styling within the framework of our markup

We create CSS by mapping InDesign *Attributes* to *CSS Properties*

- Challenges
- Style To Class Mapping Controls
 - Export Tagging Emit CSS
 - Export Dialog CSS Options
- Special Topics

CSS: Challenges

- CSS Properties are mere shadow of what InDesign Attributes can do
 - Over 255 InDesign Paragraph and Character Attributes alone!
 - Object Graphic Attributes, Media Attributes, Table Attributes...oh! my!
 - Those you think will behave the same do not – ex: Cell Strokes
- Inconsistent CSS Property support by Reading Systems
 - Ex: page-break-before, table headers & footers
- Some CSS Properties are controlled by the Reading System
 - Font, color, text size, image size (ex: iBooks)
- CSS Class Priority Order not the same as InDesign Style/Override Order
- Some InDesign Attributes map to HTML markup and tag attributes
 - Ex: Lists, Ruby, TCY, Language, Direction
- Not all Attributes are held by InDesign Styles
 - Ex: Object width/height/transform, Table row height/column width, J Frame Grids
- Should we map everything we can...and literally?
 - Ex: Paragraph Text Color, Pixels vs Ems

CSS: Export Tagging Emit CSS

Epub and HTML

Tag:

Class:

Emit CSS

- Emit CSS
 - Controls if we should create a CSS Class from the InDesign Style Attributes
 - Independent of the markup
 - Allows you to have two Styles map to one CSS Class

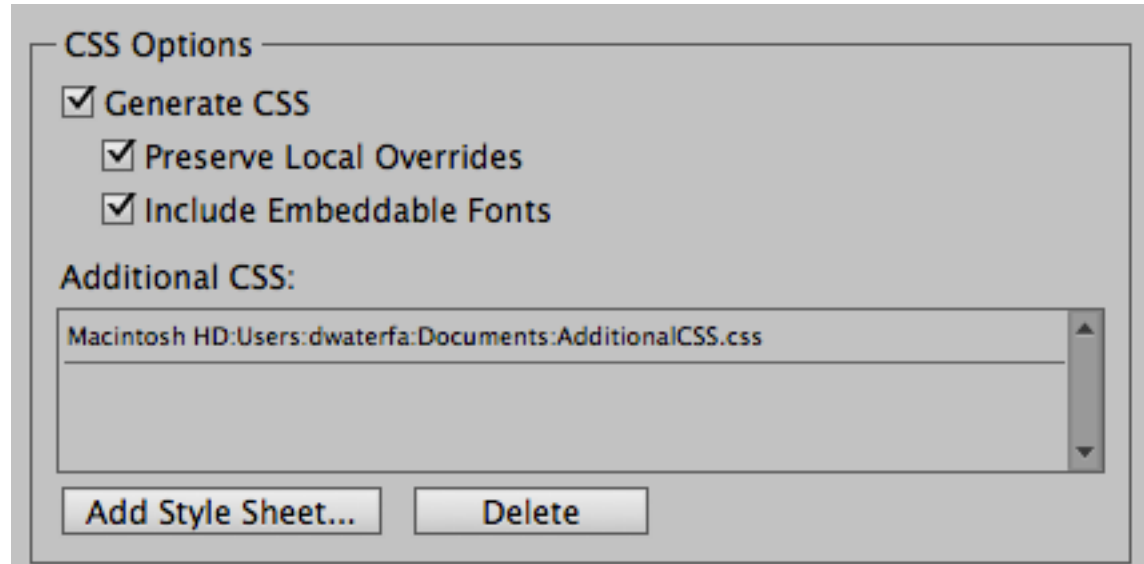
Style: "Red"
Class: "Blue"
Emit CSS: NO

Style: "Blue"
Class: "Blue"
Emit CSS: Yes



```
div.Blue { color: blue; }
```


CSS: Export Dialog CSS Options

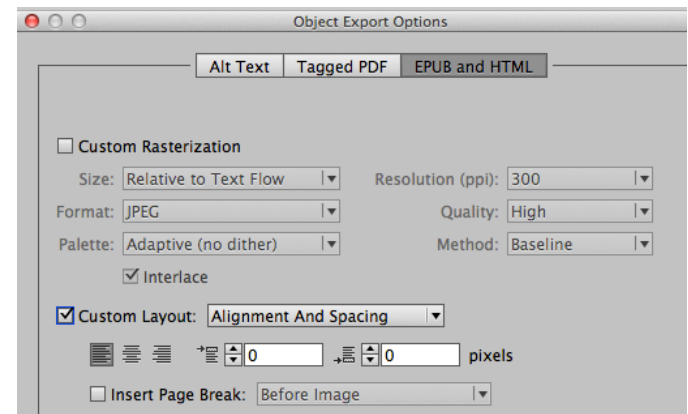
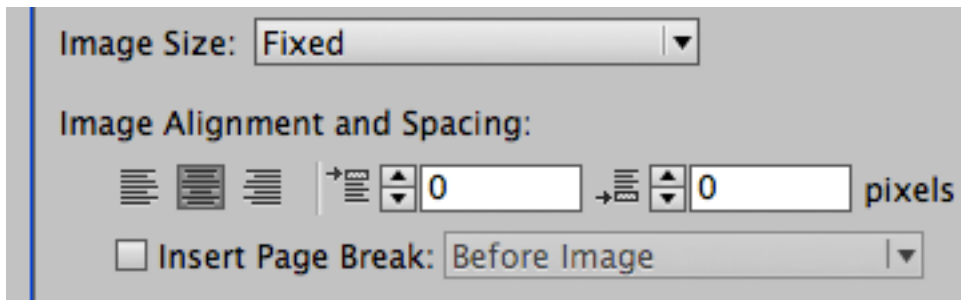


- Generate CSS – OFF
 - We will not create any CSS file
 - Expert users only...
 - We will not embed any fonts
- Additional CSS
 - We just add your file to the package and link to every HTML file...after ours
 - Useful for customization

- Objects
- Text
- Book

CSS: Special Topics/Objects

- No Graphic Attribute Support yet...
 - Stroke, fill, rounded corners are up next
- Width & Height & Transform are not InDesign Attributes
 - By default *width-in-points* is mapped to *width-in-pixels*
 - Mapped to `_idGenPageItem-#` Classes
- CSS Specific Layout Controls
 - Gives more control over reflowable composition and positioning
 - Float/Alignment/Space Before & After/Page Break
 - Mapped to `_idGenObjectLayout-#` Classes



- Fonts
 - `font-family:"Minion Pro", serif;`
- Paragraph Left/Right/First Indent and Space Before/After are in px
- Some reasons for `_idGen` Classes
 - Text size in `em @12pt = 1 em`
 - Paragraph Style says 24 pts => `font-size:2em;`
 - Character Style says 24 pts => `font-size:2em;`
 - Oops: CSS `font-size` is inherited!
 - `span._idGenCharOverride-1 {font-size:1em; }`
 - Convert Lists To Text
 - Drop Caps
 - Nested Style order
- Table & Cell CSS remains incomplete
 - Cell Strokes, Alternating Row & Column Strokes, Vertical Alignment
 - Table Style CSS not being generated

- Books are just a simple collection of independent InDesign documents
 - An InDesign documents does not know that it is part of a particular Book
- The Style Source defines the Styles which will become the CSS Classes
 - Make sure you Style Sync your documents and are thus all will be consistent
 - The first Style we see by a particular name “wins”

A Peek Behind The Curtain: Packaging

A Peek Behind The Curtain: Packaging

- Metadata
- Font Embedding
- Navigation Table Of Contents (aka TOC)
- Splitting Document

Packaging: Metadata

- Where is this so called *Document Metadata*?
 - Document XMP Data
 - File/File Info.../Description
 - For a Book export, the Style Source document.
- Here is the mapping:

EPUB Metadata

Include Document Metadata

Publisher:

Unique ID:

Description | IPTC | IPTC Extension | Camera Data | GPS Data | Video Data | Audio D ▶ ▼

Document Title:

Author:

Author Title:

Description:

Rating: ★ ★ ★ ★ ★

Description Writer:

Keywords:

ⓘ Semicolons or commas can be used to separate multiple values

Copyright Status: ▼

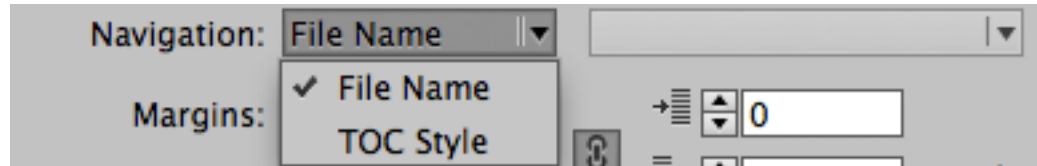
Copyright Notice:

Copyright Info URL:

Packaging: Font Embedding

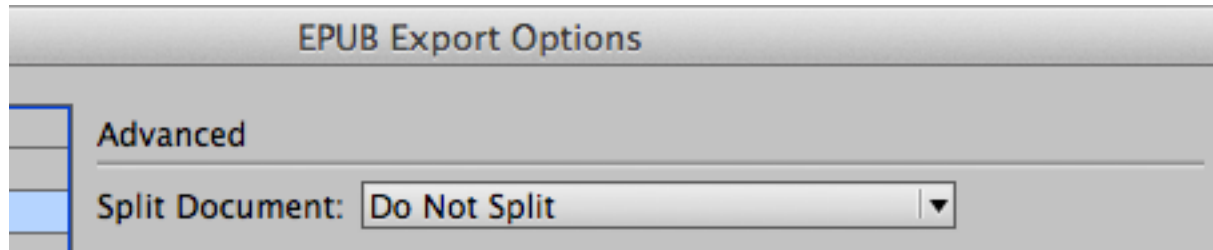
- Adobe Legal
- Subsetting
 - Only put in code points for what was used
- Obfuscation
 - IDPF defined, uses the `dc:identifier` unique ID metadata
- Compression
 - umh, make it smaller

Packaging: Navigation Table of Contents (aka TOC)



- Navigation TOC vs Document TOC
- By "File Name"
 - Based on the names of the XHTML files we put in the package
 - Really only useful for Book
- TOC Style
 - This is how to build a multi-level TOC
 - Can be different than the TOC Style used for the Print version
 - To create a single-level TOC, only put in one Paragraph Style in the TOC Style

Packaging: Splitting Document



- Why
 - Do Not Split
 - Based on Paragraph Style Export Tags
 - { Paragraph Style }

Questions?

douglas.waterfall@adobe.com



Adobe