PDF in Smalltalk

Christian Haider

Introduction

- PDF is
  - a graphics Model
  - a document Format

Graphics

- 2D Vector Graphics
- Mathematical
  - Paths
  - Coordinate transformations
- Dominant Model
  - PostScript, SVG, …
- Advanced
  - Transparency

Documents

- Faithful Reproduction
  - Abstracts from OS ‘s and Printers
  - Fonts are embedded
- Elaborate Object Model for Documents
  - Interactive
  - Linkable graphics Content
- No execution Model
  - no programming like PostScript

Standard

- ISO 32000-2008 Standard
  - PDF-1.7 (Acrobat 8)
  - Last Standard; progress through extensions
- ~ 750 Pages
  - 79 Indispensable References
- Well written
  - must have for doing anything PDF

Open Source

- PDF is important
- PDF is there
- PDF is big
- PDF is free: MIT Licence

Overview

- File format
  - Updates
- Object Model
  - Object Types
  - Document Structure
- Graphics
  - Vector Graphics
  - Text and Fonts
  - Transparency

File Structure

```
%PDF-1.4
```

```
endobj
5 0 obj
   (A String)
endobj
6 0 obj …
```

```
0000000081 00000 n
0000000248 00001 n
0000000000 00000 f
```

```
trailer <<
   /Size 22
   /Root 1 0 R >>
startxref
18799
%%EOF
```

- Header
- List of Objects

- Reference Table
  - File Position of each Object
- Trailer
  - Reference Table Size and Location
  - /Root

Show minimal

Updates

- Original stays unchanged
  - Can be signed
- New Objects are appended
- Objects can be overwritten
  - Versions
- New XRef Table for new Objects
- Can be Many

+ / -

- Can
  - Reading any valid PDF
    - Updated PDFs (many Xref tables)
  - Writing Objects as new File
    - Only 1 XRef Table
- Can´t do
  - Recreating XRef Table
  - Updating PDFs with incremental Changes
  - Linearizing for the Web

Object Model

- Basic Values
  - null, true, false
  - Numbers
    - Integer or Real; only decimal, no exponents
  - Strings
    - Encoding: PDFDoc, Font, Unicode
    - Date (utc String)
  - Names
    - Like Smalltalk Symbols
- Arrays

```
42 3.14 +7.5 -.3
```

```
(a String)
(with \n new Line)
(with char \245)
<901FA3>
```

```
(D:20110824103O+02'00)
```

```
/Root /with#20space
```

```
[3.14 (Pi) [/Math]]
```

Dictionaries

```
<<
   /name (a String)
   /id 12345
   /properties << /active 6 0 R >>
>>
```

- Unordered collection of Associations
- Unique Names as Keys
- Values are either Objects or References
- Null cannot be a Value (same as absent Key)

- The Root of all other object Types

Streams

```
<< /Length 10 >>
stream
(a String)
endstream
```

- Dictionary with arbitrary data
  - Dictionary must be direct
  - Unlimited data
  - Must be indirect

```
<< /Length 1835
   /Filter /FlateDecode >>
stream
…Binary content…
endstream
```

- Can have Filters to compress or encrypt
  - Cascaded -> [/FlateDecode /Crypt]

```
<< /Type /XRef
   /Size …
   /Root … >>
```

- XRefStreams
  - Replaces XRef Tables
  - Very compact
- Object Streams

Stream Filter

- Compression
  - **/FlateDecode** % zlib (smaller), everywhere, Predictor
  - /LZWDecode % zlib (faster), Predictor
  - /RunLengthDecode
  - /CCITTFaxDecode % B/W Pictures
  - /JBIG2Decode % B/W Pictures
  - /DCTDecode % JPEG (approximates)
  - /JPXDecode % JPEG2000 (loss less)
- /Crypt
- Development
  - /ASCIIHexDecode
  - /ASCII85Decode

Implementation

- PDF Classes in Smalltalk
  - PDF Objects implement #content
  - Smalltalk Objects implement #asPDF
  - In separate namespace PDF
  - Same names as in the spec (if possible)
    - Dictionary, Array, String, Date etc.
  - Some Classes may be aliased
    - Name, Number, Boolean, null
  - Can be confusing

+ / -

- Can
  - Read all object Types
  - Write any Object
  - Can use /FlateDecode for Reading and Writing
- Cannot
  - No picture oriented stream filters

Speaking PDF

- With this, we can read any PDF
- We can use PDF instead of Smalltalk
  - Would be cool to have that in Smalltalk…
- We can specify the PDFs by configuring the Dictionaries
- Domain Language PDF

Object Model: Documents

- /Root
  - **/Type /Catalog** % required
  - **/Pages**
  - /Outlines
  - /StructTreeRoot
  - /MetaData % XML
  - /Names
  - ….
- /Page(s)
  - **/MediaBox** [0 0 595 842]
  - /Contents % Stream of graphics Operators
  - **/Resources** % Fonts, Images, Color Spaces

create minimal

Domain Objects

- Subclass of Dictionary or Stream
  - May be typed explicitly with /Type
    - TypedDictionary and TypedStream
  - Has Version
  - Has Documentation
- Typed Attributes
  - Type(s)
  - direct or indirect
  - required/optional
  - Version
  - Documentation

Typing

- Explicit with /Type
- Implied by attribute Type
  - specialized when assigning to an Attribute

- Checks when reading
  - Checks compatibility => Error
  - Specializes Objects
- Reads lazy

PDF Explorer

- A good Writer needs a good Reader
  - and vice versa
- Shows the Contents of a PDF on the object Level
- Uses meta Data about Attributes (Version, Doc, required etc.)

Show PDFExlorer

+ / -

- Can
  - Infer the implemented Types
  - Detect type Errors
  - Infer Version
  - Show Documentation
- Cannot
  - Not all type restrictions are implemented
  - edit

Time: 30 min

Graphics

- Stream of Operators with Parameters
- Executed in sequence to produce Graphics
- /GraphicsState
  - holds all (28) Attributes for the current Operation
  - Can be stacked (nested)
- Operations (73)
  - 15 groups of Functionality
    - GraphicsState, Color, Marking…
    - Paths, clipping, Text, painting…

Lines and Paths

```
0 0.5 0.5 0 K
3 w
10 100 m
300 500 l
S

0.5 0 0 0.5 k
20 40 m
20 80 l
40 80 l
40 40 l
f
```

- Line
- Filled Rectangle

+ / -

- Have
  - Read and write Operations with Parameters
  - Bare Metal
  - Only /DeviceCMYK and /DeviceGray
- Don´t have
  - GraphicsState
  - Enforcing correct order of Operations
    - Examples: marking, text…
  - No /DeviceRGB or any other colour Spaces
  - Higher Abstractions (publicly)
    - Graphical Objects
    - Text Objects

Text

```
BT
/F13 12 Tf
288 720 Td
(Hello World) Tj
ET
```

- Paints Chars from a Font

```
/Resources <<
 /Font << /F13 23 0 R >>
>>

23 0 obj
<< /Type /Font
   /Subtype /Type1
   /BaseFont /Helvetica >>
endobj
```

- Needs /Font Resource
  - Type-1
  - TrueType
  - OpenType

Create Text

About Fonts

- Occupied me last Year
- Varieties of vector Fonts
  - PostScript Type 1
  - TrueType
  - OpenType (PS /TT)
- 14 PDF Standard Fonts (Type 1)

```
<< /Type /Font
   /Subtype /Type1
   /BaseFont /DDPEFM+Tahoma
   /FirstChar 32
   /LastChar 169
   /Widths [278 …]
   /FontDescriptor 4 0 R
   /Encoding /WinAnsiEncoding >>
```

- Font

```
4 0 obj
<< /Type /FontDescriptor
   /FontName /DDPEFM+Tahoma
   /Flags 32
   /FontBBox [-166 -225 1000 931]
   /ItalicAngle 0
   /Ascent 718
   /Descent -207
   /CapHeight 718
   /StemV 88
   /FontFile3 5 0 R>>
```

- Descriptor

```
5 0 obj
<< /Length 3723
   /Subtype /Type1C >>
stream … endstream
```

- File

+ / -

- Have
  - Font Explorer
  - OpenType (PostScript kind)
  - Type-1 (last minute implementation ☺)
    - Standard 14 Fonts
    - Custom (one free example Font is included)
  - Tabular Glyphs
- Don´t have
  - TrueType, OpenType (TT)
  - Subsetting
    - Allows to publish custom graphics
  - Kerning, Ligatures
  - General way to access alternative Glyphs
  - Advanced Typography (as possible with OpenType)

Show FontExplorer

Transparency

- More and more useful: Gradients, Shadows… and everywhere
- Approach
  - Combine the colors from different layers
  - Usually done on pixel level
  - PDF on the graphics Level
- How to?
  - Create Graphics with own contents stream
  - Paint Graphics onto another Graphics using the right attributes

Implementation

- Graphic Editor needs Screen Output
  - Fonts
  - Transparency
- VisualWorks 7.8
- Directly implemented in Windows GDI(+)
  - Text output with pixel level adjustments
  - Graphics (planed)
  - Only Windows

+ / -

- Have
  - Font support for Windows
- Don´t have
  - Transparency
  - Font support for
    - TrueType
    - non-Windows platforms

Documentation

- Class Documentation from the Spec
- Attribute Documentation from the Spec
- Extracted Properties of Attributes and made them operational
- Docuware – tight connection between doc and code

Extending

- Subclass (Typed)Dictionary or (Typed)Stream
  - Use name from the Spec
  - Add PDF Documentation to the class comment
- Add Attributes
  - Add class method named with attribute Name
  - Add PDF Documentation as comment
  - Extract Pragmas from docu
  - Implement the access (with or without Default)
  - Add your Logic

```
Pages
   <typeIndirect: #Pages>
   <required>
   <attribute: 4 documentation: 'The page tree node that shall be
the root of the document's page tree.'>

   ^self objectAt: #Pages
```

Show code

+ / -

- Have
  - Good places for Doc
  - Good operational Annotations
  - Easy to extent
- Don´t have
  - No class doc
  - No PDF Reference link
  - Not all dependencies are implemented
    - requiredIf: version = x and: attribute /y notNil

Package Structure – load Order

- Fonts
  - (Fonts for Windows)
- PDF

- Prerequisites
  - Values

To do

- Support porting
  - To Pharo, Squeak, VA, Smalltalk/X, Dolpin …
  - Problem with Namespaces, Pragmas?
- Fonts
  - Subsetting, Kerning, Ligatures
- PostScript Interpreter
- GraphicsState
- Smalltalk source parser for PDF

Summary

What do I have?

- Writer for smallCharts
  - Driven by customer Demand
  - Vector Graphics with custom Fonts
- Bare metal implementation
  - Strictly implementing the Spec
  - Object Model

- Implementation in VisualWorks 7.8
  - On Windows

What I don´t have

- Relaxed Reader
  - Not error tolerant at all (unlike Acrobat)
- No Bitmaps, no Reports, no Tables
- No Encryption, no signing
- No non-latin Languages
- No pluggable GraphicsContext
- No rendering/painting
  - Acrobat
  - Ghostscript
- No screen support for other Platforms
- Ports to other Smalltalks

Projects – What to do with it?

- Vector graphics Editor
- Online PDF Generation
- PDF Tools and Verifier
- Renderer
- Embedding Viewer
  - Ghostscript / Acrobat

References

- PDF Specification
  http://www.adobe.com/devnet/pdf/pdf_reference.html

- Project Page (Docs, Forum, FileOuts…)
  http://pdf4smalltalk.origo.ethz.ch/

- Cincom Public Store
  http://www.cincomsmalltalk.com/CincomSmalltalkWiki/PostgreSQL+Access+Page