G53DOC Revision

Steven R. Bagley

Today

- Skim over the content of the whole course
- Weighted toward the earlier stuff in the course
- Please ask questions on anything you are not sure on…

Text

- Started off looking at how text is encoded
- Baudot code
  - Uses a shift code to select different character sets of characters
- ASCII
- Unicode

00	
01	T
02	CARRIAGE RETURN
03	O
04	SPACE
05	H
06	N
07	M

08	LINE FEED
09	L
0A	R
0B	G
0C	I
0D	P
0E	C
0F	V

10	E
11	Z
12	D
13	B
14	S
15	Y
16	F
17	X

18	A
19	W
1A	J
1B	FIGURE SHIFT
1C	U
1D	Q
1E	K
1F	LETTERS SHIFT

LETTER CASE

Position zero unused

00	
01	5
02	CARRIAGE RETURN
03	9
04	SPACE
05	#
06	,
07	.

08	LINE FEED
09	)
0A	4
0B	&
0C	8
0D	0
0E	:
0F	=

10	3
11	+
12	WHO ARE YOU
13	?
14	.
15	6
16	$
17	/

18	-
19	2
1A	BELL
1B	FIGURE SHIFT
1C	7
1D	1
1E	(
1F	LETTERS SHIFT

FIGURE CASE

Position zero unused
Explain how we can write a message in Baudot code
Use whiteboard

USASCII code chart

b_7 b_6 b_5 →	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
Bits ↓ b_4 b_3 b_2 b_1 ↓ Row ↓ Column →	0	1	2	3	4	5	6	7
0 0 0 0	0	NUL	DLE	SP	0	@	P	`	p
0 0 0 1	1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	2	STX	DC2	"	2	B	R	b	r
0 0 1 1	3	ETX	DC3	#	3	C	S	c	s
0 1 0 0	4	EOT	DC4	$	4	D	T	d	t
0 1 0 1	5	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	6	ACK	SYN	&	6	F	V	f	v
0 1 1 1	7	BEL	ETB	'	7	G	W	g	w
1 0 0 0	8	BS	CAN	(	8	H	X	h	x
1 0 0 1	9	HT	EM	)	9	I	Y	i	y
1 0 1 0	10	LF	SUB	*	:	J	Z	j	z
1 0 1 1	11	VT	ESC	+	;	K	[	k	{
1 1 0 0	12	FF	FS	,	<	L	\	l	|
1 1 0 1	13	CR	GS	-	=	M	]	m	}
1 1 1 0	14	SO	RS	.	>	N	^	n	~
1 1 1 1	15	SI	US	/	?	O	_	o	DEL

Go through some of the ordered parts

UTF-8

- Designed by Ken Thompson and Rob Pike over dinner on the back of an envelope
- Aimed to solve the problems of UTF-1
- Self-syncing
- Type of each byte uniquely decodable
- Varies in length from 1-4 bytes

Yes really see -- http://www.cl.cam.ac.uk/~mgk25/ucs/utf-8-history.txtOr
Not quite as compact as UTF-1

UTF-8

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4
7	U+007F	0xxxxxxx			

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4
7	U+007F	0xxxxxxx			
11	U+07FF	110xxxxx	10xxxxxx		

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4
7	U+007F	0xxxxxxx			
11	U+07FF	110xxxxx	10xxxxxx		
16	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8

Bits	Last Code Point	Byte 1	Byte 2	Byte 3	Byte 4
7	U+007F	0xxxxxxx			
11	U+07FF	110xxxxx	10xxxxxx		
16	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
21	U+1FFFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

x are bits from the original code points
Show how to encode the € symbol (U+20AC)

UTF-8 self syncing

- Top two bits tell us the type of byte
  - Bit `7` is `0`, a *single byte* character
  - Bit `6-7` is `11`, start of a MBCS, a *leading byte*
  - Bit `6-7` is `10`, part of a MBCS, a *continuing byte*
- Can easily find our place in the stream for decoding…

MBCS Multibyte character

Text Search

- Looked at some text search algorithms
- Simple naive search
- Boyer-Moore String Search
- Burrows-Wheeler Transform

Observations

- Boyer-Moore made several observations about possible mismatchs
- These observations enable us to slide *pattern* ahead more than one character at a time

Observation One

- If the mismatching *char* in *text*, does not occur in *pattern*:
- Then we know there's no possibility of *pattern* matching at `0`, `1`, `2`, ..., `length(pattern)`
- Since this would require the character to be part of pattern
- Can slide *pattern* down `length(pattern)` chars

Observation Two

- More generally, even if *char* does occur in *pattern*
- We can still slide *pattern* so that the *char* aligns with the rightmost occurrence of *char* in pattern
- If we slide it any less, then it still won't be a match…

char is the mismatching char in text

Observation 3a

- Third observation they made takes place when a character is matched
- Continue backing up until we match all of *pattern* — and so have found it
- Or a mismatch occurs after matching *m* characters…

Observation 3a

- Using the same reasoning as before, we can obtain a value *k* to slide *pattern*
- If the right-most char is to the right of the mismatch, then we'd have to slide the pattern backwards to align it
- This is worthless, so…
- In this case, $k = 1$

k is based on the rightmost occurence of char in pattern as before

Observation 3a

- On the other hand, if it is to the left of the mismatch, then $k = delta_1 - m$
- In either case, we can slide pattern down *k* characters
- And continue from the end of pattern again

delta is the distance from the end of pattern of the rightmost occurence
m is the number of characters matched

Observation 3b

- But we can do better than that…
- We know that the next *m* characters of *text* match the final *m* characters of *pattern*
- Call this *subpat*
- Also know that this occurrence of *subpat* is preceded by *char*

Observation 3b

- Roughly speaking…
- Slide *pattern* down some so the discovered *subpat* is aligned by the rightmost occurrence of *subpat* in *pattern* not preceded by *char*
- Must allow the right most plausible reoccurrence of *sub*pat to fall off the left end of *pattern*

Observation 3

- In the case, where we have matched *m* characters we want to slide either
  - 1 character
  - $delta_1$ characters
  - $delta_2(j)$ characters
- Just chose the maximum of the three…

Observations

- These observations massively reduced the number of comparisons we do
- In this example, we only make 14 references to *text*
- Seven of which were verifying the final match…

BWT and Suffix Arrays

- Another approach makes use of the Burrows-Wheeler Transform
- Easier to understand by consider a related approach Suffix Arrays first
- BWT effectively used to build up the suffix array…

0	MISSISSIPPI
1	ISSISSIPPI
2	SSISSIPPI
3	SISSIPPI
4	ISSIPPI
5	SSIPPI
6	SIPPI
7	IPPI
8	PPI
9	PI
10	I

All possible suffixes of Mississippi

Suffix Array Search

- Just as slow as our original naive algorithm
- Also need to build up array (more RAM)
- That's true — but if you sort the suffix array, you can use binary search
- Finds *pattern* in O(`m log n`) time
- Also, finds all occurrences

more RAM than the naive approach, but probably about the same as Boyer-Moore

M	I	S	S	I	S	S	I	P	P	I

M	I	P	P	I	S	S	I	S	S	I
I	P	P	I	S	S	I	S	S	I	M

M	I	S	S	I	S	S	I	P	P	I
I	S	S	I	S	S	I	P	P	I	M
S	S	I	S	S	I	P	P	I	M	I

M	I	S	S	I	S	S	I	P	P	I
I	S	S	I	S	S	I	P	P	I	M
S	S	I	S	S	I	P	P	I	M	I
S	I	S	S	I	P	P	I	M	I	S
I	S	S	I	P	P	I	M	I	S	S
S	S	I	P	P	I	M	I	S	S	I
S	I	P	P	I	M	I	S	S	I	S
I	P	P	I	M	I	S	S	I	S	S
P	P	I	M	I	S	S	I	S	S	I
P	I	M	I	S	S	I	S	S	I	P
I	M	I	S	S	I	S	S	I	P	P

I	M	I	S	S	I	S	S	I	P	P
I	P	P	I	M	I	S	S	I	S	S
I	S	S	I	P	P	I	M	I	S	S
I	S	S	I	S	S	I	P	P	I	M
M	I	S	S	I	S	S	I	P	P	I
P	I	M	I	S	S	I	S	S	I	P
P	P	I	M	I	S	S	I	S	S	I
S	I	P	P	I	M	I	S	S	I	S
S	I	S	S	I	P	P	I	M	I	S
S	S	I	P	P	I	M	I	S	S	I
S	S	I	S	S	I	P	P	I	M	I

Lexicogrpahically sorted version!

I	M	I	S	S	I	S	S	I	P	P
I	P	P	I	M	I	S	S	I	S	S
I	S	S	I	P	P	I	M	I	S	S
I	S	S	I	S	S	I	P	P	I	M
M	I	S	S	I	S	S	I	P	P	I
P	I	M	I	S	S	I	S	S	I	P
P	P	I	M	I	S	S	I	S	S	I
S	I	P	P	I	M	I	S	S	I	S
S	I	S	S	I	P	P	I	M	I	S
S	S	I	P	P	I	M	I	S	S	I
S	S	I	S	S	I	P	P	I	M	I

Last column taken and used as basis for compression

BWT search

- The BWT matrix will be bigger than the Suffix array
- But fortunately, there is a way to build up any line of the sorted matrix in linear time
- Using just the last line (and a couple of easily precomputed tables)

BWT search

- Therefore, using the last line of the matrix
  - Which could well all ready exist if we've compressed *text*
  - The reconstruction algorithm and a binary search
  - We can find all occurrences of a *pattern* very quickly

e.g. on disk

PostScript

- Spent some time looking at PostScript
- The Adobe Graphics Model
- How to write PostScript software

Adobe Graphics Model

- Based around the idea of painting marks on a blank page
- Any mark made obscures any previous mark underneath it
- Although modern versions allow for transparency

Adobe Graphics Model

- Painted Marks may be:
  - Character shapes (glyphs)
  - Geometric shapes or lines (paths)
  - Sampled images (bitmaps)

Spaces

- AGM makes use of the notion of co-ordinate *spaces*
- Main two are
  - *User Space*
  - *Device Space*
- Others include *text* space, *glyph* space, *image* space and *form* space

User Space

- User space defines a specific co-ordinate space
- Abstracts the AGM from any specific device
- Resolution, origin etc. all vary across devices
- If the AGM didn't abstract the device away then a page would look vastly different on each device…

Current Transformation Matrix

- AGM handles conversion from *User Space* to *Device Space* automatically
- Every co-ordinate is mathematically transformed to device space by multiplying it with a transformation matrix
- Referred to as the *Current Transformation Matrix*

User to Device Space

- By combining separate affine transformations into a single matrix
- We can develop a single TM that can convert from user space to device space
- Usually a combination of translates and scales

US to DS

- Suppose we have 300dpi printer which has an origin at the top of the page going down
- Need to:
  - Scale user space co-ordinates by 300/72
  - Invert the y-axis (can do this by scaling the y-axis alone by -1)
  - Translate the origin so it's at the top of the page

Assume page height is A4 so 595x842
Work through example
Show the three matrices, then combine them in the right order

US to DS example

$$\begin{bmatrix} \frac{300}{72} & 0 & 0 \\ 0 & \frac{300}{72} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 842 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{300}{72} & 0 & 0 \\ 0 & -\frac{300}{72} & 0 \\ 0 & 842 & 1 \end{bmatrix}$$

Start by moving the origin to the top-left (in user space)
Then scale y by -1 (now origin is the top left)
Finally can scale by 300/72 to adjust for device resolution

Operators

- Saw how to construct simple PostScript programs that draw things
- Graphical Operators
- Programmatic operators
- How to redefine operators

PDF

- Looked at PDF
- Saw how it arose from the ‘graph-binder’ technique developed by John Warnock
- PDF is declarative
- Constructed from a series of objects
- Indirect and direct objects

```
%Start building the page
-10 -10 translate
gsave
s24o setfont
leftmar 476 add top 18 add moveto (19)show s24 setfont (79)show
s10b setfont
  (SCHEDULE  E)leftmar -2.5 tshow
  ((Form 1040))leftmar -1.75 tshow
s6 setfont
  (Department of the Treasury) leftmar -1 tshow
  (Internal Revenue Service) leftmar -.3 tshow
s16b setfont
  (Supplemental Income Schedule)208 -2.5 tshow
s8bc setfont
  ((From pensions and annuities, rents and royalties, partnerships, estates and
trusts, etc.))174 -1.3 tshow
  (Attach to Form 1040) 188 -.33 tshow
  (See Instructions for Schedule E (Form 1040).) 328 -.33 tshow
  174 top 5 add triangle
  314 top 5 add triangle
s7 setfont
  (Name(s) as shown on Form 1040)leftmar 1 tshow
  (Your social security number) 476 1 tshow
%draw all full horizontal lines.
0 fullrule
(Part I) leftmar top 38 sub bbox
s10b setfont
(Pension and Annuity Income.)102 3 tshow
s8 setfont
(If fully taxable, do not complete this part. Enter amount on Form 1040, line
17.)250 3 tshow
(For one pension or annuity not fully taxable, complete this part. If you have more
than one pension or annuity that is not fully taxable,)48 4 tshow
(attach a separate sheet listing each one with the appropriate data and enter
combined total of taxable parts on line 4.)48 4.8 tshow
2 fullrule
5 fullrule
s8b setfont
(1a)52 6 tshow
s8 setfont
(Did you and your employer contribute to the pension or annuity?)68 6 510 tlshow
514 top 6 pica mul sub YesNoBox
s8b setfont
(b)57 7 tshow
s8c setfont
(If "yes," do you expect to get back your contribution within 3 years of the date
you receive the first payment?)68 7 510 tlshow
514 top 7 pica mul sub YesNoBox
s8b setfont
(c)57 8 tshow
182 top 96 sub triangle
($.............................,)192 8 tshow
474 top 96 sub triangle
s8b setfont
(1d) 492 8 tshow
s8c setfont
(If "Yes," show: Your contribution)68 8 tshow
s8b setfont
(d) 288 8 tshow
s8c setfont
(Contribution received in prior years)294 8 470 tlshow
s8b setfont
(2)52 9 tshow
(2)492 9 tshow
s8 setfont
(Amount received this year) 68 9 484 tlshow
s8b setfont
(3)
52 10 tshow
(3)492 10 tshow
s8 setfont
(Amount on line 2 that is not taxable)68 10 484 tlshow
s8b setfont
(4)52 11 tshow
(4) 492 11 tshow
s8 setfont
(Taxable part (subtract line 3 from line 2). Enter here and include in line 18
below)68 11 484 tlshow
11 fullrule
(Part II) leftmar top 145 sub bbox
s10b setfont
(Rent and Royalty Income or Loss. If you need more space, attach a separate
sheet.)102 12 tshow

s8b setfont
(5a)52 13 tshow
s8c setfont
(Have you claimed expenses connected with your vacation home (or other dwelling
unit) rented to others (see Instructions)?)68 13 512 tlshow
```

SCHEDULE E (Form 1040)
Department of the Treasury
Internal Revenue Service

Supplemental Income Schedule

(From pensions and annuities, rents and royalties, partnerships, estates and trusts, etc.)

▶ **Attach to Form 1040** ▶ **See Instructions for Schedule E (Form 1040).**

1979

Name(s) as shown on Form 1040 | Your social security number

Part I Pension and Annuity Income.

If fully taxable, do not complete this part. Enter amount on Form 1040, line 17.

For one pension or annuity not fully taxable, complete this part. If you have more than one pension or annuity that is not fully taxable, attach a separate sheet listing each one with the appropriate data and enter combined total of taxable parts on line 4.

1a	Did you and your employer contribute to the pension or annuity?	☐ Yes ☐ No	
b	If "yes," do you expect to get back your contribution within 3 years of the date you receive the first payment?	☐ Yes ☐ No	
c	If "Yes," show: Your contribution ▶ $....., d Contribution received in prior years ▶	1d	
2	Amount received this year	2	
3	Amount on line 2 that is not taxable	3	
4	Taxable part (subtract line 3 from line 2). Enter here and include in line 18 below	4	

Part II Rent and Royalty Income or Loss. If you need more space, attach a separate sheet.

5a	Have you claimed expenses connected with your vacation home (or other dwelling unit) rented to others (see Instructions)?	☐ Yes ☐ No
b	If "yes," did you or a member of your family occupy the vacation home (or other dwelling unit) for more that 14 days during the tax year?	☐ Yes ☐ No
6a	Did you elect to claim amortization (under section 191) or depreciation (under section 167(o)) for a rehabilitated certified historic structure (see instructions)?	☐ Yes ☐ No
b	Amortizable basis (see Instructions) ▶	

	(a) Property code (describe in Part V)	(b) Total amount of rents	(c) Total amount of royalties	(d) Depreciation (explain in Part VI) or depletion (attach computation)	(e) Other expenses (explain in Part VII)	(f) loss	(g) income
	Property A.						
	Property B.						
	Property C.						
	Property D.						
7	Property E. Amounts from Form 4835 . .						
8	Totals . .					( )	

9 Total rent and royalty income or (loss). Combine amounts in columns (f) and (g), line 8. Enter here and include in line 18 below — 9

Part III Income or Losses from --

	(a) Name	(b) Employer identification number	(c) loss	(d) income
Partnerships				
	10 Add amounts in columns (c) and (d) and enter here — 10		( )	
	11 Combine amounts in column (c) and (d), line 10, and enter net income or (loss) — 11			
	12 Additional first-year depreciation — 12		( )	
	13 Total Partnership income or (loss). Combine lines 11 and 12. Enter here and include in line 18 below — 13			
Estates or Trusts				
	14 Add amounts in columns (c) and (d) and enter here — 14		( )	
	15 Total estate or trust income or (loss). Combine amounts in columns (c) and (d), line 14. Enter here and include in line 18 below — 15			
Small Business Corporations				
	16 Add amount in columns (c) and (d) and enter here — 16		( )	
	17 Total small business corporation income or (loss). Combine amounts in columns (c) and (d), line 16. Enter here and include in line 18 below — 17			

Part IV

18 **TOTAL** income or (loss). Combine lines 4, 9, 13, 15, and 17. Enter here and on Form 1040, line 18. — 18

19 Enter your share of gross farming and fishing income applicable to Parts II and III. — 19 — **E**

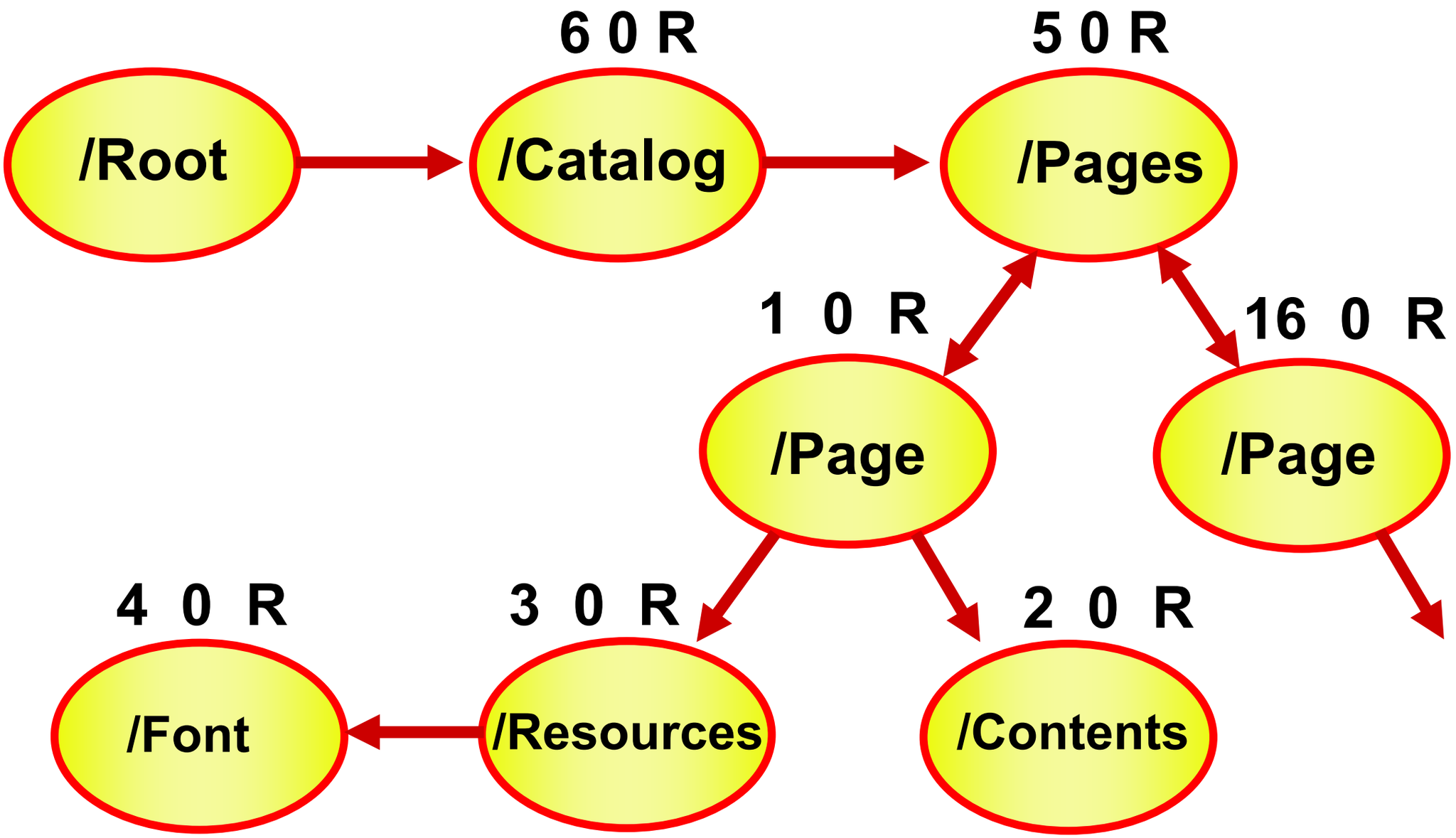
```
-10 -10 T
G
fff17 sf
522 734 M
(19)S
fff18 sf
(79)S
fff6 sf
46 748.0 M
(SCHEDULE  E)S
46 739.0 M
(\(Form 1040\))S
fff8 sf
46 730 M
(Department of the Treasury)S
46 721.6 M
(Internal Revenue Service)S
fff16 sf
208 748.0 M
(Supplemental Income Schedule)S
fff13 sf
174 733.6 M
(\(From pensions and annuities, rents and royalties, partnerships, estates and trusts,
etc.\))S
188 721.96 M
(Attach to Form 1040)S
328 721.96 M
(See Instructions for Schedule E \(Form 1040\).)S
G
174 721 T
0 0 M
8 3.5 L
0 7 L
CP
F
GR
G
314 721 T
0 0 M
8 3.5 L
0 7 L
CP
F
GR
fff14 sf
46 706 M
(Name\(s\) as shown on Form 1040)S
476 706 M
(Your social security number)S
1 SL
N
580.0 716.0 M
46.0 716.0 L
SK
G
46 678 T
1 0 T
N
0 0 M
0 10 L
48 10 L
48 0 L
CP
F
1 SG
fff6 sf
12 2 M
(Part I)S
GR
fff6 sf
102 682 M
(Pension and Annuity Income.)S
fff9 sf
250 682 M
(If fully taxable, do not complete this part. Enter amount on Form 1040, line 17.)S
48 670 M
(For one pension or annuity not fully taxable, complete this part. If you have more than
one pension or annuity that is not fully taxable,)S
48 660.4 M
(attach a separate sheet listing each one with the appropriate data and enter combined
total of taxable parts on line 4.)S
1 SL
N
580.0 692.0 M
46.0 692.0 L
SK
1 SL
N
580.0 656.0 M
46.0 656.0 L
SK
fff10 sf
52 646 M
```

```
%Start building the page
-10 -10 translate
gsave
s24o setfont
leftmar 476 add top 18 add moveto (19)show s24 setfont
s10b setfont
  (SCHEDULE  E)leftmar -2.5 tshow
  ((Form 1040))leftmar -1.75 tshow
s6 setfont
  (Department of the Treasury) leftmar -1 tshow
  (Internal Revenue Service) leftmar -.3 tshow
s16b setfont
  (Supplemental Income Schedule)208 -2.5 tshow
s8bc setfont
  ((From pensions and annuities, rents and royalties, p
trusts, etc.))174 -1.3 tshow
  (Attach to Form 1040) 188 -.33 tshow
  (See Instructions for Schedule E (Form 1040).) 328 -.
  174 top 5 add triangle
  314 top 5 add triangle
s7 setfont
  (Name(s) as shown on Form 1040)leftmar 1 tshow
  (Your social security number) 476 1 tshow
%draw all full horizontal lines.
0 fullrule
(Part I) leftmar top 38 sub bbox
s10b setfont
(Pension and Annuity Income.)102 3 tshow
s8 setfont
(If fully taxable, do not complete this part. Enter amo
17.)250 3 tshow
(For one pension or annuity not fully taxable, complete
than one pension or annuity that is not fully taxable,)
(attach a separate sheet listing each one with the appr
combined total of taxable parts on line 4.)48 4.8 tshow
2 fullrule
5 fullrule
s8b setfont
(1a)52 6 tshow
s8 setfont
(Did you and your employer contribute to the pension or
514 top 6 pica mul sub YesNoBox
s8b setfont
(b)57 7 tshow
s8c setfont
(If "yes," do you expect to get back your contribution
you receive the first payment?)68 7 510 tlshow
514 top 7 pica mul sub YesNoBox
s8b setfont
(c)57 8 tshow
182 top 96 sub triangle
($.............................,)192 8 tshow
474 top 96 sub triangle
```

```
-10 -10 T
G
fff17 sf
522 734 M
(19)S
fff18 sf
(79)S
fff6 sf
46 748.0 M
(SCHEDULE  E)S
46 739.0 M
(\(Form 1040\))S
fff8 sf
46 730 M
(Department of the Treasury)S
46 721.6 M
(Internal Revenue Service)S
fff16 sf
208 748.0 M
(Supplemental Income Schedule)S
fff13 sf
174 733.6 M
(\(From pensions and annuities, rents and royalties, partnerships, estates and trusts,
etc.\))S
188 721.96 M
(Attach to Form 1040)S
328 721.96 M
(See Instructions for Schedule E \(Form 1040\).)S
G
174 721 T
0 0 M
8 3.5 L
0 7 L
CP
F
GR
G
314 721 T
0 0 M
8 3.5 L
0 7 L
CP
F
GR
fff14 sf
46 706 M
(Name\(s\) as shown on Form 1040)S
476 706 M
(Your social security number)S
1 SL
N
580.0 716.0 M
46.0 716.0 L
SK
G
```

Note how they've reduced things to single letters

PDF does the same thing with its operators -- they are generally single or double letters



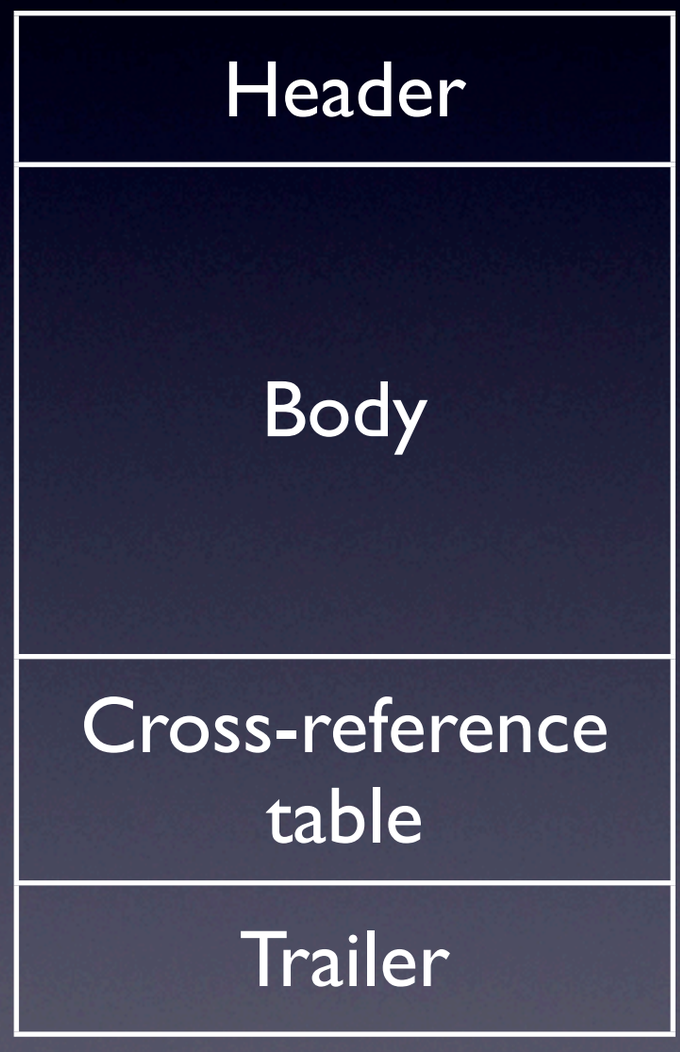
Objects in PDF form a graph, but how are they stored in the files

```
%PDF-1.2
%âãÏÓ
2 0 obj
<< /Length 74
>>
stream
BT
/F1 1 Tf
36 0 0 36 300
300 Tm
0 g
0 Tc 0 Tw
(Hello World)
Tj
ET
endstream
endobj
3 0 obj
<<
/ProcSet [/
PDF /Text ]
/Font <<
/F1 4 0 R
>>
>>
endobj
4 0 obj
<<
/Type /Font
/Subtype /
Type1
/Name /F1
/BaseFont /
Helvetica
>>
endobj
1 0 obj
<<
/Type /Page
/Parent 6 0 R
/Resources 3 0
R
/Contents 2 0
R
>>
endobj
6 0 obj
<<
/Type /Pages
/Kids [1 0 R]
/Count 1
/MediaBox [0 0
594 842]
>>
endobj
8 0 obj
<<
/Type /Catalog
/Pages 6 0 R
>>
endobj
9 0 obj
<<
/CreationDate
(D:
20130225112108
)
/Producer
(Acrobat
Distiller
Command 3.01
for Solaris
2.3 and later
\(SPARC\))
>>
endobj
xref
0 10
0000000000 65535 f
0000000507 00000 n
0000000016 00000 n
0000000140 00000 n
0000000427 00000 n
0000000356 00000 n
0000000587 00000 n
0000000234 00000 n
0000000668 00000 n
0000000717 00000 n
trailer
<<
/Size 10
/Root 8 0 R
/Info 9 0 R
/ID
[<7b67e06cbb92
a9ae2dba172c60
c1b8e7><7b67e0
6cbb92a9ae2dba
172c60c1b8e7>]
>>
startxref
851
%%EOF
```

This has been tweaked to show it all on screen so some bits are missing
And the xref is wrong…
Highlight how it references other objects

PDF File Structure

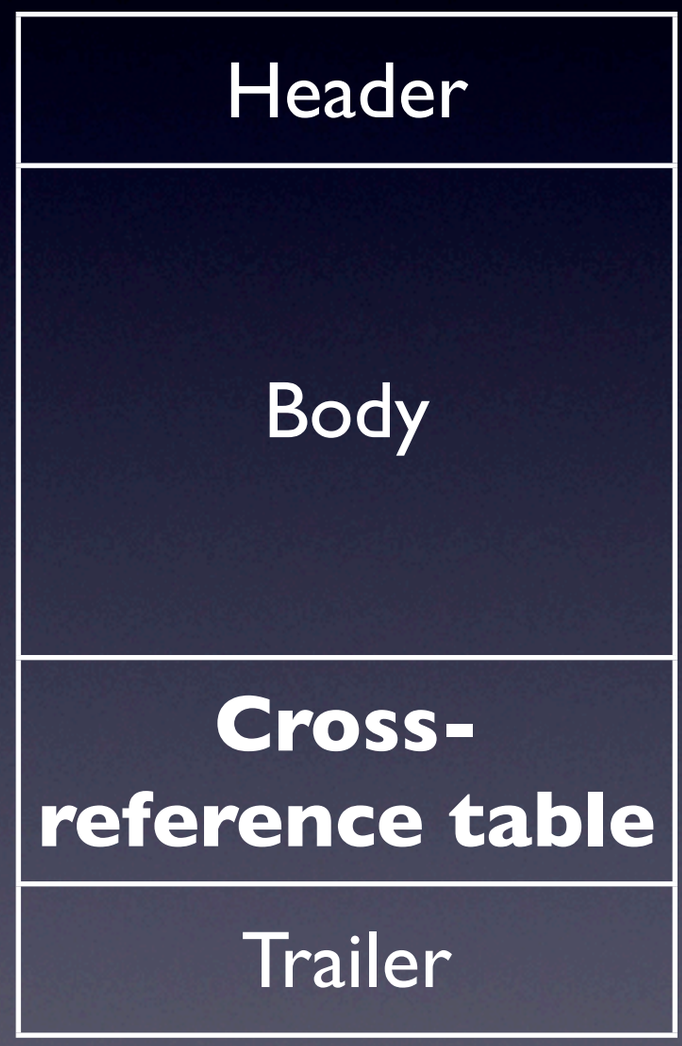
- PDF files are constructed from four parts
- The header contains the version of the PDF
  `%PDF-1.4`
- Followed by four bytes of high-ASCII to force the file to be treated as binary



% used to mean start of comment

Cross-reference table

- Designed to allow quick access to any object in the file
- Maps object id to the position within the file
- Enables access to PDF without parsing all of it
- Even if bigger than memory…



% used to mean start of comment

Cross-reference table

`xref` ← marks start of xref

```
0 10
0000000000 65535 f
0000000513 00000 n
0000000016 00000 n
0000000146 00000 n
0000000433 00000 n
0000000362 00000 n
0000000593 00000 n
0000000240 00000 n
0000000674 00000 n
0000000723 00000 n
```

Line Breaking

- Spent some time looking at line breaking algorithms
- Simple approaches tend to lead to nasty breaks
- Studied the TeX line breaking algorithm in detail…

TeX Line breaking

- Models the text in terms of:
  - Boxes — content
  - Glue — space between content that can shrink or stretch
  - Penalties — for breaking at certain points
- Tries to minimise the cost of line-breaking at certain points

Fonts

- Spent some time looking at fonts
- Bitmap fonts
- Outline Fonts

EBooks

- Looked at how EBooks require a different approach
- Since the shape and size of the display isn't known until it is read
- Have to use formats that are capable of easily being reflowed

EPUB

- Looked at the structure of EPUB 3
- Saw how the content is ordered
  - Both for the system to render
    - But also in terms of providing navigation aids to the user
- Aims to declare stuff as much as possible rather than programming it