ディジタルドキュメント (7)

高久雅生

2015年5月28日（木）3･4時限

本日のお品書き

- （前回の復習）
- （前回講義での質問・要望への回答：２件）
- テキスト処理とドキュメント形式
  - ✓ドキュメント形式（フォーマット）とは
  - ✓テキストフォーマットとは
    - 文字コード
  - ✓電子メールのドキュメント形式
  - ✓軽量テキストフォーマット
  - ✓ページ記述型フォーマット

（前回の復習 = ふりかえり）

- マークアップ言語とデジタルドキュメント
  - ✓メタ言語SGMLとXML
    - 整形式
    - メタ言語とスキーマ
  - ✓様々な応用
- セマンティックウェブとデジタルドキュメント
  - ✓Semantic Webの基盤技術
  - ✓オープンデータとメタデータ，ライセンス
  - ✓Linked Open Data (LOD)
  - ✓デジタルドキュメントのワークフロー

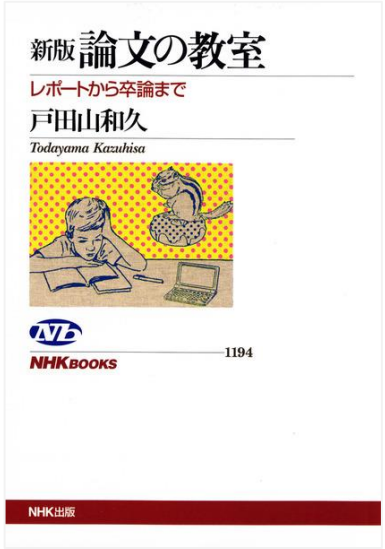
質問･要望へのお返事 (1)

- Aを取るような模範的なレポートを例として紹介していただけると今後のレポートの参考になります。

次回以降、検討します。なお、文例等については、レポートの作成法･作文技術の参考文献も確認するようにしてください。

質問・要望へのお返事 (1-2)

- ✓ 木下是雄. 理科系の作文技術. 中央公論新社, 1981, 244p. （中公新書, 624）
- ✓ 井上千以子. 思考を鍛えるレポート・論文作成法. 第2版, 慶応大学出版会, 2014, 156p.
- ✓ 戸田山和久. 新版 論文の教室: レポートから卒論まで. NHK出版, 2012, 313p.

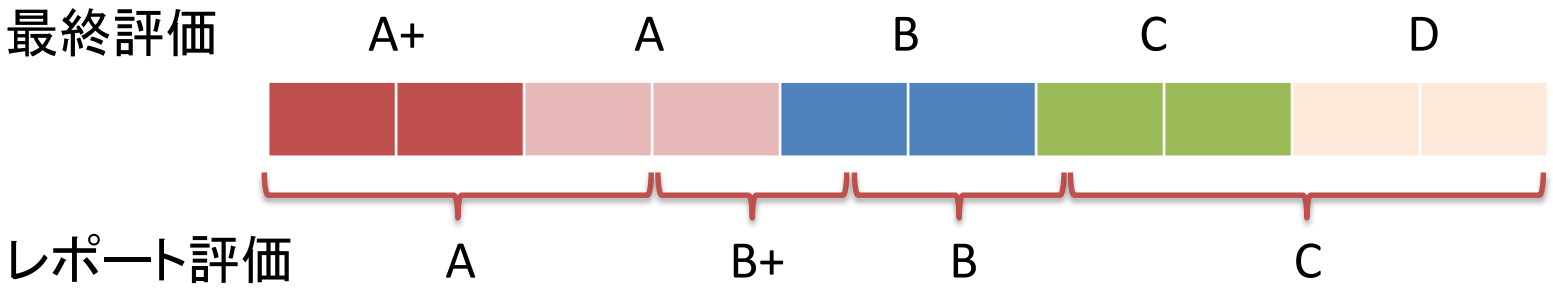


参考：http://bit.ly/1gLEYLC
（附属図書館: 816.5）

質問･要望へのお返事 (2)

授業の評価は、A+, A, B, C, D（不可）で評価されると思うのですが、なぜレポートの評価は、A, B+, B, C なのでしょうか?
A+がA，AがB+に対応しているということでしょうか。

最終評価をおこなう際に総合的な評価結果を補正する余地を残すためにわざとずらしています。



ドキュメントフォーマット (1)

ドキュメントフォーマットとは？

- デジタル文書の内容を機械的に解釈するための「決まりごと」「様式」「形式」
  - ✓データをどのように並べるか（配置）
  - ✓内容解釈のためのデータ・情報は、デジタル情報として、符号化（エンコード）される
- 例：
  - ✓プレインテキスト？
    - 文書内容を文字コードにより符号化し、テキスト（文字の連なり）として解釈できるようにしたもの。
  - ✓画像形式？
    - 表現すべき画像要素を、ピクセル・描画要素単位の情報として符号化し、2次元画像として解釈できるようにしたもの。

ドキュメントフォーマットの切り口 (1)

- テキスト (text) vs バイナリー (binary)
  - ✓ビットデータ
  - ✓文字コードによる解釈
  - ✓外字
- フォーマットの指定・識別・判別
- シンプルコンテンツ vs 複合メディア
  - ✓埋め込みコンテンツ
  - ✓ハイパーリンク
- メタデータ
  - ✓埋め込みメタデータ
  - ✓外部メタデータ記述
- 文書レイアウト
  - ✓ページ概念
- 文書内の書式要素
  - ✓見栄え / スタイル
  - ✓フォント
- ファイル vs ストリーム
  - ✓データの保存・蓄積と配信

ドキュメントフォーマットの切り口 (2)

- オープンフォーマット
  - ✓ 移植可能性 / ソフトウェア独立性
  - ✓ Free / proprietary
- 標準化
  - ✓ デファクト標準とデジュール標準 (“de facto” vs “de jure”)
- 文書フォーマットのバージョン
- フォーマット変換
  - ✓ プレインテキスト → HTML
  - ✓ LaTeX → PDF
- 圧縮
  - ✓ 可逆 vs 非可逆
- セキュリティ
  - ✓ パスワード
  - ✓ 電子署名
- 長期保存
- デジタルフォレンジック

プレインテキストとは？

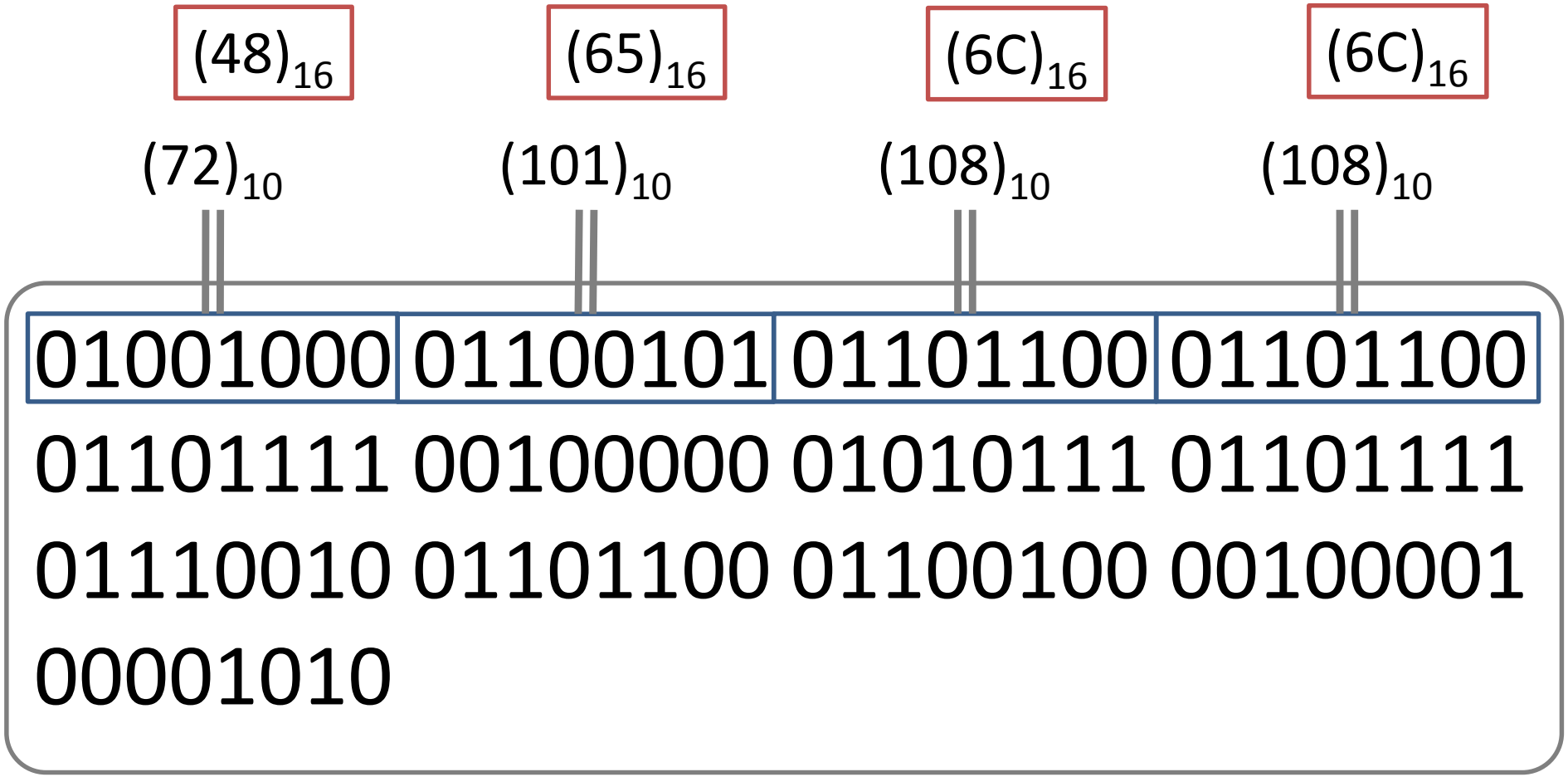
- 文書内容を文字コードにより符号化し、テキスト（文字の連なり）として解釈できるようにしたもの。
- もっとも基本的なドキュメントフォーマットの一つ
  - ✓ドキュメントフォーマットの基礎
  - ✓（別の定義：バイナリフォーマットではない）
- 特徴：
  - ✓ほぼあらゆる環境で特別なソフトウェア無しに用いることが出来る
  - ✓テキストフォーマットだけでは、書式要素を保持しない
  - ✓複合的なオブジェクトとの関連付けはそれ単体ではできない

プレインテキスト読解例 (1)

0100100001100101011011000110110001101111100100000010101101110111100100110110001100101000010000

プレインテキスト読解例 (2)

- 先頭からバイト(byte)単位で解読してみよう。
  - ✓1バイト = 8ビット (bit)



プレインテキスト読解例 (3)

- 文字コードと照合
- ASCIIコード表を基にしてみる

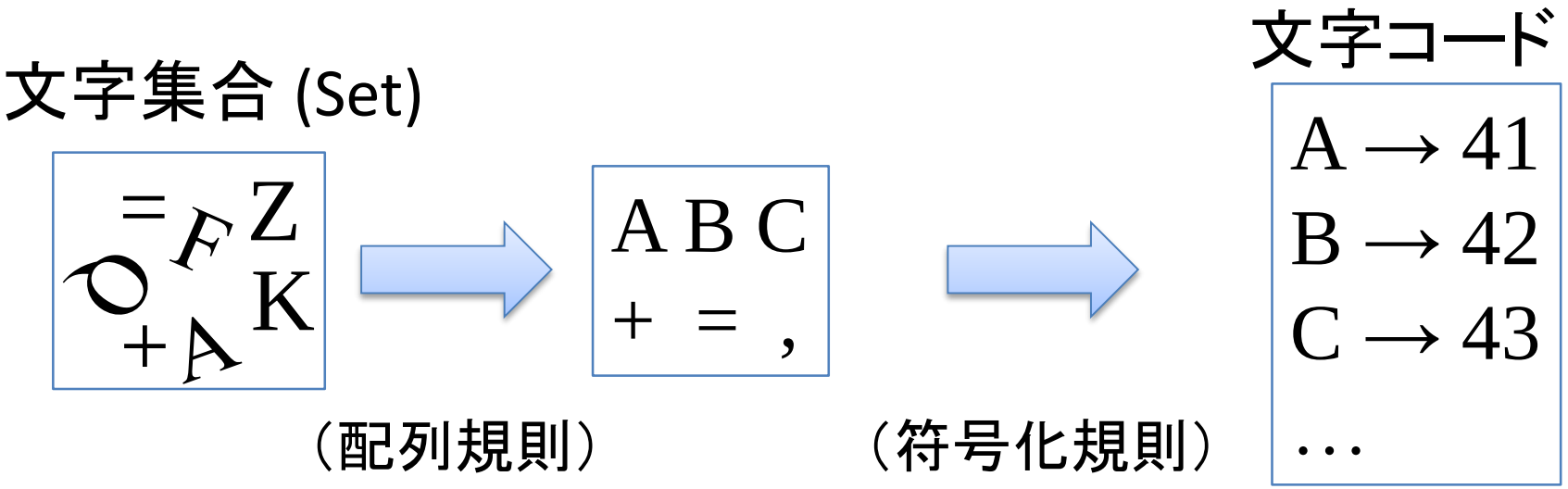
H e l l o　W o r l d ! ¥n

48 65 6C 6C 6F 20 57 6F 72 6C 21 64 0A

01001000 01100101 01101100 01101100
01101111 00100000 01010111 01101111
01110010 01101100 01100100 00100001
00001010

※文字コードとは？

- 文字を表現する集合を配列し、それを符号化する方式を定めたもの
  - ✓文字集合 (Character Set)
  - ✓文字符号化方式 (Character Encoding)



ASCIIコード表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	“	#	$	%	&	‘	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	¥	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	M	n	o
7	P	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

文字コードの実際

- 欧米語の場合
  - ✓ ASCII
  - ✓ ISO-8859-1
- 日本語の場合
  - ✓ JIS X 0208
  - ✓ JIS X 0201
  - ✓ …
- その他
  - ✓ Unicode

- よく使われる文字コード
  - ✓ ASCII
  - ✓ UTF-8
  - ✓ Shift_JIS
  - ✓ EUC-JP
  - ✓ ISO-2022-JP
  - ✓ ISO-8859-1

文字コードにおける制御文字

- 計算機上で表現される「文字」には、情報交換の役割があり、たとえば、下記のような特殊な役割をもつ制御文字が含まれている:
  - ✓空白文字
  - ✓タブ
  - ✓改行文字
  - ✓削除記号（ BS, DEL ）
- これらの文字以外の表現は、歴史的な文脈で必要とされたという経緯に加え、テキストのみで文書構造を表現するためにも用いられる。

00101101001011010010000000010101011100111100010101101011101011
01000001011001011101101110000001011100000100000001010000100110101
10000101110011011000010110111100100000010101000110000101101011011
00001011010110111010100101001001000000010000000100000001000001010
00101010100000110010001100000011000100110011111000111101011110011
10010110111111011101010010011001011110000111101111011000111110010
00110000101110011111011001111011000101001001100101011011000011011110
01010010011101010101001001101111010100100101101111010010010111111
10100001101000110000101000101111001011110010000000100000110000111
10111101100011111001000110000101110011111011001111011000101111111
01111010111101111100011011010011011011101111011110000110010101111
10011101001011110000110100101110001111010010110100011101001011010
00101011100010100110101101011110011010110111110011110000101000101
11100101111001000000100000010001010110110101100001011010010110110
00001110100010000001101101011000010111001101100001011011110100000
00111001101101100011010010111001100101110011101000111001101110101
01101011011101010110001001100001001011100110000101100011001011100
11010100111000000001010001011110010111100100000001000000101010001
10010101101100001110100010000000110000001100100011100100101101001
11000001101010011100100101101001100010011001100111001001101000000
1010

「文字化け」とは?

- 文字コードを解釈するソフトウェアが元の文字コードを復号化できない場合に発生する不整合
  - ✓元の文字コード：Shift_JIS
  - ✓ソフトウェア側：UTF-8

プレインテキストに基づくドキュメントフォーマット

- プレインテキストに基づくドキュメントフォーマットには多くの事例がある
  - ✓Eメール
  - ✓HTMLファイル（ウェブ）
  - ✓SGML / XML
  - ✓LaTeX

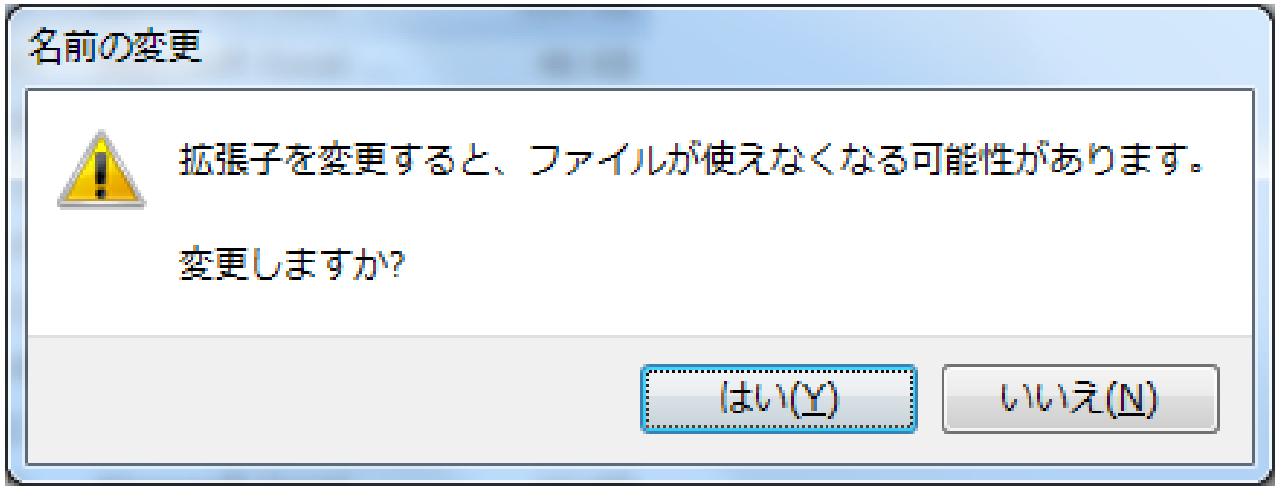
ドキュメントフォーマットの識別，判別

- メタデータとして付与
  - ✓拡張子
- 内容で判断
  - ✓マジックナンバー（File magic）
- 通信・転送プロトコル上で指定
  - ✓MIME Type (IANA)
  - ✓その他

フォーマットの識別：ファイル拡張子

ファイル名の末尾を「.」（ピリオド）で区切り、ファイル形式を示す2～3文字からなる文字列を付与

- テキストファイル: text.txt
- HTMLファイル： index.html, welcome.htm
- PDFファイル： example.pdf
- などなど。



フォーマットの識別：ファイルマジックナンバー

- 古典的なヒューリスティック手法
- ファイルの冒頭に分かりやすいフォーマットを示すバイト列ヘッダを挿入しておく方式
- 例：
  - ✓XML形式：<?xml version=“ …
  - ✓GIF形式（画像）：GIF87a または GIF89a

フォーマットの識別：MIMEタイプ (MIME type)

- Multipurpose Internet Mail Extensions (MIME)
- コンテンツ用のフォーマット指定タイプ
  - ✓ Internet Media Type, Content Type
- ウェブやEメールでのドキュメント形式の指定に使われる
- IANAによる公式登録リスト：http://www.iana.org/assignments/media-types
- `[type]/[subtype]`
  - ✓ `text/plain, text/html, application/pdf, image/gif, vnd.openxmlformats-officedocument.presentationml.presentation`
- タイプ
  - ✓ application, audio, image, message, model, multipart, text, video

メールフォーマット

- Eメール：最も古典的なインターネットアプリケーションの一例
- テキスト形式によるフォーマットの例
  - ✓基本的に、テキストデータのみでやり取りする
    - （一種、テキスト形式への符号化がなされる）
- オープンフォーマット
  - ✓様々なソフトウェア（メーラー）がメール内容を扱う
  - ✓表示、作成、送信
- 送信者と受信者は異なる環境にいることを前提
  - ✓全世界で幅広く分散して使われており、新しい仕様に改善しようとしても一度に一斉に切り替えることは不可能
- RFC 5322 “Internet Message Format” (2008): http://tools.ietf.org/html/rfc5322
  - ✓RFC 2822 (2001) ← RFC 822 (1982)

メールフォーマットの例

Date: Thu, 20 Sep 2001 07:00:00 JST
From: 首相官邸 <koizumi@mmz.kantei.go.jp>
Reply-To: koizumi@mmz.kantei.go.jp
Subject:【小泉内閣メールマガジン 2001/09/20】難局にひるまず立ち向かおう！！
To: koizumi-ml-user@mmz.kantei.go.jp

小泉内閣メールマガジン ========== 2001/09/20

★☆　今週のキーワード「補正予算」 ☆★
　小泉総理が平成１３年度補正予算の編成について準備を進めるよう指示をしました。（解説は最後に）

□　目次

［らいおんはーと　～　小泉総理のメッセージ］
●　難局にひるまず立ち向かおう！！

メールフォーマットの例

メールヘッダ

Date: Thu, 20 Sep 2001 07:00:00 JST ⏎
From: 首相官邸 <koizumi@mmz.kantei.go.jp> ⏎
Reply-To: koizumi@mmz.kantei.go.jp ⏎
Subject:【小泉内閣メールマガジン 2001/09/20】難局にひるまず立ち向かおう！！ ⏎
To: koizumi-ml-user@mmz.kantei.go.jp ⏎

メール本文

小泉内閣メールマガジン ========== 2001/09/20

★☆　今週のキーワード「補正予算」　☆★
　小泉総理が平成13年度補正予算の編成について準備を進めるよう指示をしました。（解説は最後に）

□　目次

［らいおんはーと　～　小泉総理のメッセージ］
●　難局にひるまず立ち向かおう！！

メールフォーマットの例（ヘッダ部）

```
Received: from goat.ipc.ulis.ac.jp (goat.ipc.ulis.ac.jp [133.51.184.12])
        by nile.ulis.ac.jp (8.9.3+Sun/8.9.1) with ESMTP id HAA06473
        for <masao@nile.ulis.ac.jp>; Thu, 20 Sep 2001 07:06:10 +0900 (JST)
Received: from mails1.mmz.kantei.go.jp (mails1.mmz.kantei.go.jp [202.212.115.161])
        by goat.ipc.ulis.ac.jp (8.8.6 (PHNE_14041)/3.7W) with SMTP id HAA02715
        for <masao@ulis.ac.jp>; Thu, 20 Sep 2001 07:08:52 +0900 (JST)
Received: (qmail 6408 invoked by uid 113); 20 Sep 2001 07:03:39 +0900
Received: from localhost (HELO mails1.mmz.kantei.go.jp) (sendmail-bs@127.0.0.1)
  by localhost with SMTP; 20 Sep 2001 07:03:39 +0900
Received: (qmail 6325 invoked by uid 200); 20 Sep 2001 07:00:00 +0900
Date: 20 Sep 2001 07:00:00 +0900
From: =?ISO-2022-JP?B?GyRCPHNBajQxRSEbKEI=?= <koizumi@mmz.kantei.go.jp>
Reply-To: koizumi@mmz.kantei.go.jp
Subject: =?iso-2022-jp?B?GyRCIVo+LkB0RmIzVSVhITwlayVeJSwlOCVzGyhKIDIwMDEv?=
        =?iso-2022-jp?B?MDkvMjAbJEIhW0ZxNkkkSyRSJGskXiQ6TikkQTh+JCskKiQmISobKEo=?=
        =?iso-2022-jp?B?GyRCISobKEo=?=
To: koizumi-ml-user@mmz.kantei.go.jp
Message-Id: <20010919220000.6323.qmail@msgs1.mmz.kantei.go.jp>
Mime-Version: 1.0
Content-Type: text/plain;
        charset="iso-2022-jp"
Precedence: bulk
Content-Length: 12860
```

メールサーバ転送の記録

時刻

送信者

返信先の指定

件名

宛先

メッセージID

MIMEバージョン

フォーマット指定

メールマガジン配信指定

本文長

ここまで、メールヘッダ

メールフォーマットの例（ヘッダ部）(2)

- Date
- From
- To
- Subject
- Message-Id
- Received
- Reply-To
- Content-Type
- MIME-Version
- Precedence
- Content-Length

複合型ドキュメントとしてのメールフォーマット

- 複数のドキュメントをテキストフォーマットの中に埋め込む
  - ✓マルチパート：添付ファイルの実現
- MIME：Multipurpose Internet Mail Extensions
  - ✓メール用の交換フォーマットにおいて、添付ファイルやプレインテキスト以外の形式でのやりとりを行うための拡張仕様
- メールヘッダの拡張
  - ✓MIME-Version
  - ✓Content-Type
  - ✓Content-Disposition
  - ✓Content-Transfer-Encoding

添付ファイル付きのメールフォーマット例

```
Received: from SINPRD0310MB391.apcprd03.prod.outlook.com ([169.254.5.109]) by
    SINPRD0310HT002.apcprd03.prod.outlook.com ([10.255.17.37]) with mapi id
    14.16.0311.000; Thu, 30 May 2013 02:38:55 +0000
Content-Type: multipart/mixed; boundary="_002_9BB0C5FDD26EB142B3B76E53820B0FB9020BC402SINPRD0310MB391_"
Content-Transfer-Encoding: binary
From: Masao Takaku <masao@slis.tsukuba.ac.jp>
To: Masao Takaku <masao@slis.tsukuba.ac.jp>
Subject: =?iso-2022-jp?B?GyRCJUYlOSVIGyhC?=
Date: Thu, 30 May 2013 02:38:54 +0000
Message-ID: <9BB0C5FDD26EB142B3B76E53820B0FB9020BC402@SINPRD0310MB391.apcprd03.prod.outlook.com>

--_002_9BB0C5FDD26EB142B3B76E53820B0FB9020BC402SINPRD0310MB391_
Content-Type: text/plain; charset="iso-2022-jp"

これはテストです。

高久

--_002_9BB0C5FDD26EB142B3B76E53820B0FB9020BC402SINPRD0310MB391_
Content-Type: text/plain; name="20130529-klis-tokuron.txt"
Content-Description: 20130529-klis-tokuron.txt
Content-Disposition: attachment; filename="20130529-klis-tokuron.txt"
Content-Transfer-Encoding: base64

jcCSa4nvgUaR5Yp3iUANCg0KgUWOv5biDQoNCi0gII6ObYzjivqJ25L2gsyQaZhIgs2Cx4KkgsiC
wYLEgtyCt4KpgUgNCi0gk/yOjoLMjduCzINlgVuDfYLGjKSLhoNlgVuDfYLNlc+CpoLEguCR5Y/k
lXaCxYK3gqmBSA0KLSCM9paxiPWOdZZdgsaCqYLBgsSDQYOKgsWCt4KpgUgNCg0K

--_002_9BB0C5FDD26EB142B3B76E53820B0FB9020BC402SINPRD0310MB391_--
```

- Content-Type: multipart/mixed; boundary="_002_9BB0C5FDD26EB142B3B76E53820B0FB9020BC402SINPRD0310MB391_”

複数パートの区切り文字を全体で指定

- 各パートでの内容種別
  1. Content-Type: text/plain; charset="iso-2022-jp“
  2. Content-Type: text/plain; name="20130529-klis-tokuron.txt”

     Content-Disposition: attachment; filename="20130529-klis-tokuron.txt";

軽量テキストフォーマット

- 各種Wiki/Blog記法
  - ✓MediaWiki
  - ✓Markdown
- HTML
- LaTeX

Wiki記法

- WikiWikiWeb等により提唱
- 簡易なテキストフォーマット＋スタイル整形
  - ✓文書要素：段落、見出し、リンク、図表、強調
  - ✓HTMLへの変換により、Webでの発信や共有を前提とした軽量マークアップ言語の典型
- 標準化の欠如
  - ✓ウィキソフトウェアごとに多くの亜流（数百種?）
- 例：
  - ✓'''強調'''
    - → 強調
  - ✓[http://www.tsukuba.ac.jp/ 筑波大学]
    - → 筑波大学

MediaWiki記法

- https://ja.wikipedia.org/wiki/Help:%E6%97%A9%E8%A6%8B%E8%A1%A8
- 世界最大の百科事典Wikipediaの執筆に用いられている記法
- 特徴
  - ✓Web上での発信用途
  - ✓誰でも記述できるようHTMLを簡易化した記法
  - ✓図表や数式の表記まで対応
  - ✓HTMLマークアップにもそのまま対応
  - ✓テンプレート（transclude）

Markdown記法

- http://daringfireball.net/projects/markdown/
- Web上で提供されるソフトウェア開発やマニュアル、ブログ等で用いられる軽量テキストフォーマット
- 特徴
  - ✓Web上での発信に加えて、HTML以外のフォーマットのサポートも。
  - ✓複数の処理系の存在
    - （処理系ごとの互換性は不完全）
- 例:
  - ✓**強調**
    - → 強調
  - ✓[筑波大学](http://www.tsukuba.ac.jp)
    - → 筑波大学

LaTeX

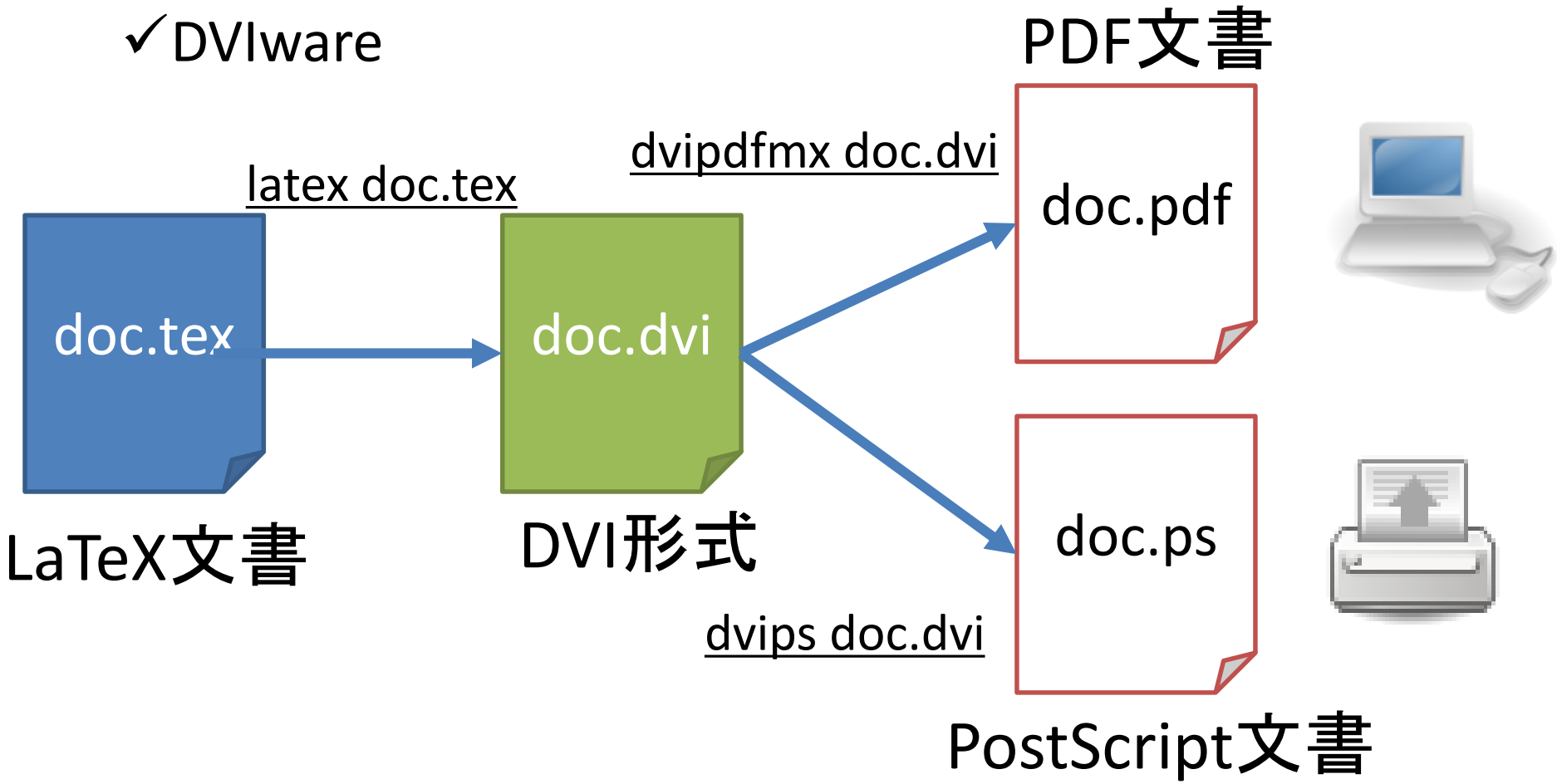
- テキストベースの文書記述言語
- 組版ツール
- （発音：ラテフ, ラテック, レイテック）
- 組版ツールTeX by Donald Knuth
  - ✓1978年から開発 ←『The Art of Computer Programming』
  - ✓数式対応に優れている
  - ✓豊富な拡張機能（マクロ, フォント）
    - 一種のプログラミング言語
- LaTeXは、TeXを文書構造記述に拡張したもの
  - ✓日本語対応版：pLaTeX

LaTeX文書の構成要素

- メタデータ、文書本文
  - ✓¥title{}, ¥author{}, ¥maketitle
  - ✓¥begin{document} … ¥end{document}
- 章節構造
  - ✓¥section{}, ¥subsection{}, ¥subsubsection{}
- 参照構造
  - ✓¥label{} ⇔ ¥ref{}
- 文献参照
  - ✓¥cite{} ⇔ ¥bibitem{}

LaTeX文書の組版（フォーマット変換）

- LaTeX → DVI → PostScript, PDF
  - ✓DVI (Device Independent)
  - ✓DVIware



LaTeXの実用例

- 理論計算機科学者による開発
  - ✓計算機科学、数学、物理学等の学術文献（論文）での利用が多い
- 例：
  - ✓ACM論文執筆用テンプレート: http://www.acm.org/sigs/publications/proceedings-templates
  - ✓Springer論文執筆用テンプレート: http://www.springer.com/authors/book+authors?SGWID=0-154102-12-970131-0

PDF (Portable Document Format)

- Portable: 持ち運び可能な…
- Document Format: 文書形式
- コンピュータの機種や環境によらず、オリジナルのイメージをかなりの程度正確に再生できる。
- 歴史的には、PostScript（プリンタ用ベクタ描画言語）が源流として存在
- 印刷媒体＋コンピュータ上でのデータ交換
  - ✓ディスプレイモニタ上での表示
  - ✓文書情報（メタデータ）
  - ✓ページ送り;ランダムアクセス
- 異なる環境でもレイアウトがほぼ一定

PDFの歴史

- 1993年 Adobe Systems社により開発
  - ✓PDF version 1.0
  - ✓Acrobat 1.0同時発売
- 1994年 Acrobat Reader無償配布
- 1995年 Netscape Navigator用プラグイン配布
  - ✓Web普及への戦略
- 2008年 ISOによる標準化
- ※ただし、PDF以前の歴史が重要
  - ✓PostScriptとDTP技術

PostScript

- PostScript
  - ✓Adobe Systems（アドビ・システムズ）社が開発したページ記述言語
  - ✓描画をベクトル化することにより高い表現力と品質の向上を実現
- 1985年 PostScriptプリンター(Apple LaserWriter)およびイメージセッタ(Linotype社)の開発
- 1989年 日本語PostScriptプリンタの登場
- DTP(Desktop Publishing)の爆発的普及に貢献

PostScriptからPDFへ

- PDFはPostScriptを元に策定
- PostScriptのターゲットは印刷（紙）
- PDFはコンピュータ上でのデータ交換
  - ✓ディスプレイでの表示
  - ✓文書情報（メタデータ）の扱い
  - ✓ページ
  - ✓ランダムアクセス

PDFの特徴

- 表示・印刷：異なる環境でもレイアウトがほぼ一定
- セキュリティ：設定可能
- ファイルサイズ：小
  - ✓データ圧縮：テキスト、図画像
- 閲覧支援：しおり，リンク，コメント，注釈
- 検索：透明テキスト
- インタラクティブ：フォーム機能により入力欄を設定可能
- アクセシビリティ：音声化
- マルチメディア：対応

PDFの印刷と表示

- 表示用アプリケーション
  - ✓無償配布⇒OSに依存しない表示
    - MS Word, Excel等はOSが変わるとレイアウトも変わる
    - HTMLはウィンドウサイズにより左右される
- PDFの罠
  - ✓Adobe Readerのバージョンが異なると正しく表示できない
  - ✓同一フォントを持っていないと文字化けやレイアウトが崩れる場合も
    - 解決方法：フォントの埋め込み
    - （課題）全体のファイルサイズが大きくなる。フォントのライセンス問題。

PDFのセキュリティ機能

- 閲覧パスワード
- 編集パスワード
  - ✓編集
  - ✓印刷
  - ✓テキストと画像のコピーを制限
  - ✓※PDF言語仕様上は、単なる“紳士協定”であることに注意…
- 電子署名による作成・配布者の認証も可能

PDFファイルの作成

- PDFはPostScriptを元に策定された
- アドビシステムズのAcrobat
  - ✓印刷機能にAdobe PDFプリンタを追加
  - ✓印刷機能においてPDFファイルを作成する
  - ✓文書の編集は一般的に各種アプリケーションにまかせている
- Mac OS Xは各種アプリケーションから印刷でPDF化可能
- 最近は各種アプリケーションが直接PDF出力をサポート

PDFファイルの実際の作成

- Adobe製品⇒そのままPDF化可能
- Microsoft Office
  - ✓プリンタとして出力
  - ✓そのままPDF化可能（MS Office 2007から）
- LaTeX
  - ✓PDFLaTeX：LaTeXソースからそのままPDFへ
  - ✓ dvipdfmx：dvi形式からPDFへ
  - ✓それ以外：dvi形式からPostScript形式にしてから、別途PDFに変換する
- 紙 ⇒ スキャン（ + OCR ）
  - ✓有名な製品ScanSnap（富士通）

PDFの課題

- 複雑な仕様
  - ✓PDF 1.7 (ISO 32000-1, 2008) は750ページ以上（付録含む）
    - http://www.adobe.com/jp/devnet/pdf/pdf_reference.html
  - ✓ISO標準化以前はAdobe一社による管理→デファクト標準の部分も一部残る
- ディスプレイでの閲覧支援とは言うものの…
- しおり，リンク，コメント，注釈に関しては、作成時につける
  - ✓Adobe Readerで閲覧しながら付けることはできない
  - ✓専用アプリケーションソフトウェアが必要

PDFの構造

- 一言で言えば、とても複雑
- ランダムアクセスを許すために、文書要素（オブジェクト）を分割して記述している
  - ✓逆に言えば、常に数百・数千ページあっても、簡単にページジャンプを実現できる。
  - ✓相互参照により、閲覧・編集ソフトウェアが文書構造を再構成しなおして表示する。
- オブジェクト（Object）
  - ✓記述のための要素。PDF文書の最も基本的な構成要素
  - ✓文字列、数値、配列、画像データ、フォント等。

PDFの構造 (2)

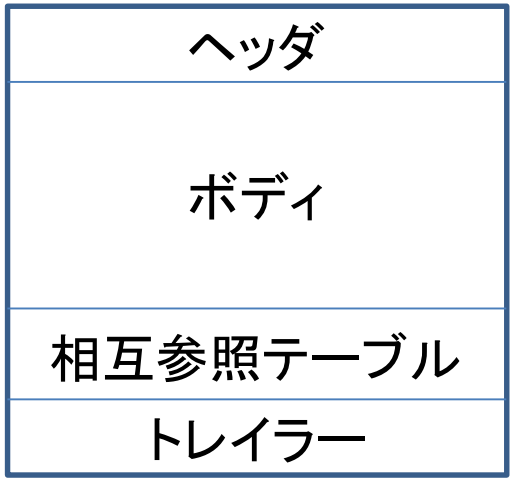
- ファイル構成 (PDF: 7.5 File Structure, p.38) :
- ヘッダ (header)

  %PDF-1.4
- ボディ (body)

  …
- 相互参照テーブル (cross-reference table)

  xref
  0 45
  0000000000 65535 f
  0000003343 00000 n
  0000003216 00000 n
  …
- トレイラー (trailer)

  trailer
  <</Root 1 0 R/Info 2 0 R/Size 45>>
  startxref
  17538
  %%EOF



※参考文献： John Whitington. PDF構造解説. 村上雅章訳. オライリー・ジャパン, 2012, 225p.

PDFの構造 (3)

- 文書構造 (PDF: 7.7 Document Structure, p.70)
  - ✓Objectsの階層関係
  - ✓参照関係で表現
  - ✓トレイラー → ドキュメントカタログ → ページリスト → 各ページ （次スライド）

- コンテントストリーム (PDF: 7.9 Content Streams and Resources, p.81)
  - ✓ページの見た目やグラフィカルな要素を記述する命令．文書構造とは独立

1 0 obj

<</Pages 18 0 R/Type/Catalog>>

endobj

18 0 obj

<</Type/Pages/Count 1/Kids[3 0 R]/MediaBox[0 0 595.28 841.89]>>

endobj

3 0 obj

<</Resources 17 0 R/Type/Page/Parent 18 0 R/Contents[16 0 R]>>

endobj

（PDFフォーマットに関する参考文献）

- Adobe Systems. Document management — Portable document format — Part 1: PDF 1.7. First Edition. 2008, 748p. http://www.adobe.com/jp/devnet/pdf/pdf_reference.html
- John Whitington. PDF構造解説. 村上雅章訳. オライリー・ジャパン, 2012, 225p.
- 荒井文吉. 手書きPDF入門. http://www.kobu.com/docs/pdf/pdfxhand.htm

まとめ

- ドキュメントフォーマットとは？
- ドキュメントフォーマットの切り口
- フォーマットの識別、判別
- プレインテキストのフォーマット
  - ✓テキスト / 文字コード
  - ✓ドキュメントフォーマットの例
    - Eメールのフォーマット
    - 軽量テキストフォーマット
- ページ記述型フォーマット
  - ✓PostScript
  - ✓PDF