

## 総論

## デスクトップパブリッシング



### 1. DTP の現状と動向†

村山 登竹

#### 1. はじめに

まず、DTP (Desktop Publishing) という奇妙な言葉のセマンティックスを明白にするとすれば<sup>1)</sup>,

(1) 著者のデスクまわりで、

(2) コンピュータの助けを借りて出版することであり、CAP (Computer Aided Publishing) と呼んでもよい。しかし、EP (Electronic Publishing) という紙以外のメディアによる出版、たとえば CD-ROM 出版などと混同するおそれがあるので避けたほうがよい。

「著者のデスクまわり」にあるコンピュータはあまり大きなものではないので PC (Personal Computer) か WS (Work Station) を使う出版と定義してもよい。

さて、従来 (そして今も) 欧米では著者が自分で出版しようとするれば、タイプライタで原稿を作り複写機を使って「出版」してきた。したがって、文字の大きさや間隔や種類に制限があった。しかしベクトルフォントとレーザプリンタ (LBP) の開発によりこの制約がとれて、いろいろな書体の文字を自由なレイアウトで出力できる DTP が可能になったのである。また、印刷会社による商業出版の原稿として著者の作ったデータがそのまま使用できるようになってきた。アメリカの学会などでは紙ではなくデジタルデータをそのまま学会に投稿する「電子投稿」が、AMS (American Mathematical Society) を創始として盛んになってきたが、我が国ではまだのようである。DTP は著者にセンスがあれば意図した美しい文書が作成できるという利点もあるが、その反面著者にセンスがなければとんでもないものができてしまい時間と紙の浪費となる。中学生のころから長年自分でタイプライタを

表-1 表示、出力装置の情報容量 (bit/mm<sup>2</sup>)

装置名	ドット数 ( $X_d \cdot Y_d$ )	階調情報 $\log_2 D$	計 (bit/mm <sup>2</sup> )
ドットプリンタ	25 (5×5)	1	25
CRT	25 (5×5)	5	125
レーザプリンタ	256 (16×16)	1	256
新聞印刷機	2500 (50×50)	1	2500
雑誌印刷機	10000 (100×100)	2	20000

$X_d$ : 横ドット数,  $Y_d$ : 縦ドット数,  $D$ : 階調数

叩いてきた欧米と日本の差はそこにあり、いくらよい DTP システムができて欧米のように普及するかどうかは疑問である。またアメリカでも、著者がはじめのうちこそ好奇心でやっていたが、そろそろあきてしまってタイピストや秘書にまかせっぱなしになってきている。しかし頼んだ人が自分でもできる欧米と自分ではできない日本では大ちがいではある。

また出力品質については、現在では文字と図の部分はほぼ商業印刷なみであるが写真の部分はまだ商業印刷に及ばない。出力品質の客観評価量として、情報容量 (IC=Information Capacity), つまり単位面積あたりの情報量を用いて比較したのが表-1 である<sup>1), 2)</sup>。

このように社会的習慣と出力品質の両方のハンディキャップはあるが、とにかく我が国にも DTP の夜明けがきたようである。

#### 2. 印刷、出版技術の歴史

印刷、出版技術の歴史における日本の貢献度は驚くほど大きい。

まず現存する世界最古の印刷物といわれるのが法隆寺に残る百万塔陀羅尼経である (765年, 図-1(●))。その中で、早くも漢数字により文章の段落と順序を示すマークアップ手法がとられ、SGML (Standard Generalized Markup Language) のようなアイデアが見られる。版下も木ではなく鉛か銅を使用して耐久性を上

† DTP Today and Future by Noboru MURAYAMA (RICOHR & D Center).

竹 リコー中央研究所

げている。

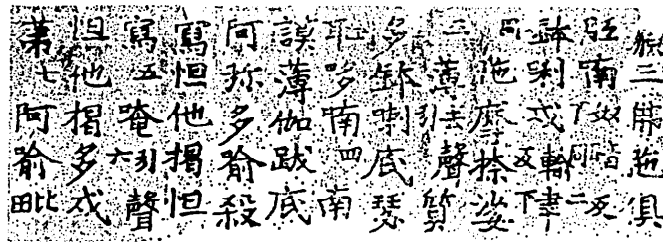
歴史に残る第二の技術革新は図-1(b)に示すようなドイツ人 Gutenberg による活版印刷による聖書の出版である (1445 年)。Gutenberg のように文字を一つ一つ別個に活字にして組み合わせるというアイデアは近世まで遂に漢字文化圏内ではみられなかった。

第三の技術革新は文字をフィルムに置いておいて光学的に原稿を作るというフィルムフォントのアイデアで、日本の石井茂吉、森沢信夫両氏により 1924 年

(大正 13 年) から 1929 年 (昭和 4 年) までかかって実用化された。

第四の技術革新は現代であり、文字は半導体メモリに入れられ、コンピュータにより選択、処理される。この第四の技術革新は 30 年前に始まったばかりで、まだ進行中であり、デジタルフォントはフィルムフォントや活字フォントと共存している (図-1(c))。

なお、図-3 のように半導体メモリの中にデジタルフォントを走査線単位に記憶するドットフォントの基

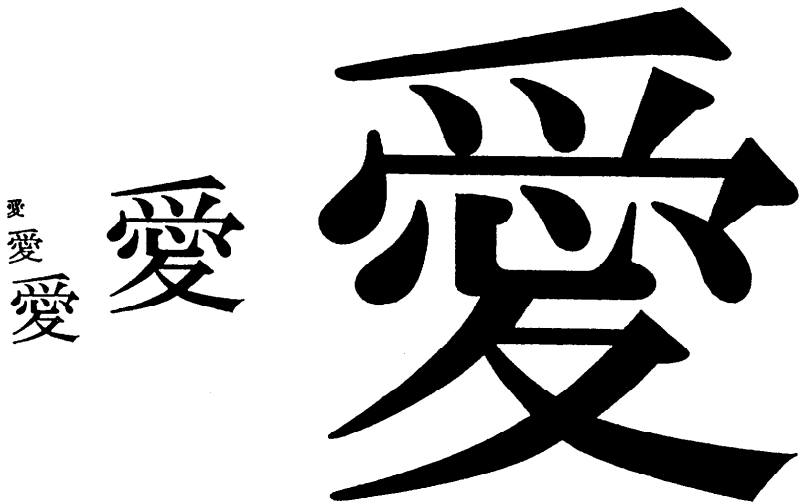


(a) 法隆寺の百万塔陀羅尼經 (世界最古の印刷物: 764 年)

gentes: incipientibus ab iherosolima. Et ego mit-  
tam pmissum pacis meae in uos: uos  
autem sedete in ciuitate. quoadulq; indu-  
amini uirtute re alto. Adusit autem eos  
foras in bethaniam: et eleuatis mani-  
bus suis benedixit eis. Et factu est du  
benedicebat illis recessit ab eis: et factu  
est in celum. Et ipsi adorantes regres-  
si sunt in iherusalem cum gaudio ma-

in epistola per multa signa exprimen-  
ta. pmissis autem descendente i bethaniam  
sepulture sue locu facta oratione. po-  
situs est ad patres suos: tam egreane?  
a dolore mortis q; a corruptione car-  
nis inuenitur aliquis. Tamen post o-  
mnes euangelii scriptas: et hoc uirgini  
debetur. Quoniam tamen uel scriptas re-  
prie disposito. uel librorum ordinando ita  
a nobis per singula non reponitur:

(b) Gutenberg の活版印刷 (1445 年)



(c) アウトラインフォント (1989 年)

図-1 フォントの技術革新の歴史

## 出力サンプル

不思議の国のアリス  
 不思議の国のありす ①

Alice's Adventure in Wonderland ②

不思議の国のアリス ③

不思議の国のありす ④

Alice's Adventure in Wonderland

不思議の国のアリス

不思議の国のありす

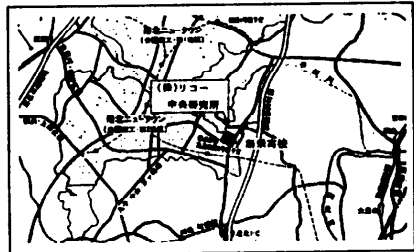
Alice's Adventure in Wonderland

不思議の国のアリス

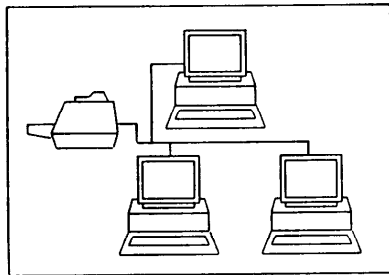
不思議の国のありす



⑥



⑤



⑦

JIS漢字コード表 ⑧

	0	1	2	3	4	5	6	7	8	9	A	B	C
2120									:	:	?	!	
2130													
2140													
2150													
2160													
2170													

スキャナーやレーザープリンタをサポートし、  
 テキスト、イメージ、グラフィック、罫線を  
 編集できるエディタ

$$H(j\omega) = \sum_{k=-\infty}^{\infty} h_n^*(k) e^{-j\omega k} \quad \text{⑨}$$

## MODEM ⑩

NAME	DATA.R (bps)	FREQ (Hz)	BAUD (sps)	MODULATION METHOD	SIGNAL STRUCTURE
V26bis	2400	1800	1200	4 φ PSK 00: 45° 01: 135° 11: 225° 10: 315°	

図-2 統合出力サンプル

本特許は 1965 年に筆者と三沢隆氏（現船井電気研究所長）によりなされた。また、文字の輪郭線を記憶するアウトラインフォントの基本特許と輪郭線の中をペイントするスキャンラインコンバージョンの基本特許は 1970 年長谷川実郎氏（当時日本電子産業、現日電オフィスシステム常務）によりなされた。しかしあまり早すぎて筆者のは期限切れ、長谷川特許ももうすぐ切れる。

このように、印刷出版の歴史は文字の処理がメインであった。現在、DTP はアウトラインフォントとレーザプリンタの実用化により文章はなんとかきれいになった。しかし、図形と写真の処理はまだまだこれからである。

### 3. 文書処理

印刷物の内容を大分類すると図-2 のように文章（テキスト）、図形（グラフィクス）、写真（イメージ）が

混在している。この中で文章の処理が歴史的にも重要であった。近年の画像映像文化を反映して図やイメージの比率が高まってきてはいるものの、まだ文章の重要性が逆転してはいない。

文章の処理は大きく次の三つの処理に分けられる。

(1) 文章の配列や大きさを読みやすくするレイアウト処理

(2) 文字の画像品質のための文字処理

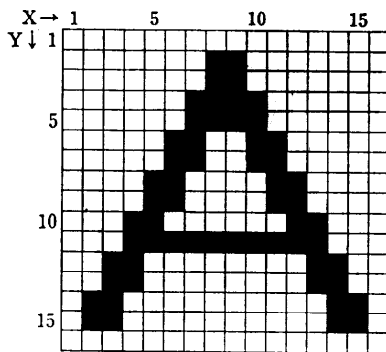
(3) 文章の論理的内容の処理

上記(1)(2)は文章の体裁の処理つまり物理的処理であり、(3)は文章の内容つまり論理的処理である。

#### 3.1 レイアウト処理

文章は章や節などの階層構造をもっている。また、本文のほか、図-2 の(9)の数式、(10)の表のように論理的処理と物理的処理の両方が必要なものもある。

文章のレイアウト処理では対照的に異なる二つの方式がある。



(a) 16x16 "A"

Lin	Edges	MH-Code	MH-Bits	MMR-Code	MMR-Bits
1	1	W(16)	6	W(16)	6
+2	3+	W(7) B(2) W(7)	10	+H(W(7) B(2)) V(0)	10
3	3	W(7) B(2) W(7)	10	V(0) V(0) V(0)	3
4	3	W(6) B(4) W(6)	11	VL(1) VR(1) V(0)	7
5	3	W(6) B(4) W(6)	11	V(0) V(0) V(0)	3
+6	5+	W(5) B(2) W(2) B(2) W(5)	16	+VL(1) H{B(2) W(2)} VR(1) V(0)	16
7	5	W(5) B(2) W(2) B(2) W(5)	16	V(0) V(0) V(0) V(0) V(0)	5
8	5	W(4) B(2) W(4) B(2) W(4)	16	VL(1) VL(1) VR(1) VR(1) V(0)	13
9	5	W(4) B(2) W(4) B(2) W(4)	16	V(0) V(0) V(0) V(0) V(0)	5
10	5	W(3) B(2) W(6) B(2) W(3)	16	VL(1) VL(1) VR(1) VR(1) V(0)	13
11*	3*	W(3) B(10) W(3)	15	*V(0) PV(0) V(0)	7
+12	5+	W(2) B(2) W(8) B(2) W(2)	17	+VL(1) H{B(2) W(8)} VR(1) V(0)	17
13	5	W(2) B(2) W(8) B(2) W(2)	17	V(0) V(0) V(0) V(0) V(0)	5
14	5	W(1) W(2) B(10) W(2) W(1)	21	VL(1) VL(1) VR(1) VR(1) V(0)	13
15	5	W(1) B(2) W(10) B(2) W(1)	21	V(0) V(0) V(0) V(0) V(0)	5
16*	1*	W(16)	6	*PPV(0)	9
Tot.	62		225		137

(b) MH, MMR Code

図-3 "A" の MH-Code と MMR-Code

(1) WYSIWYG 方式

(What You See Is What You Get 方式)

文章データのみが画面表示される。いわゆるワープロ (WP) ではよく使用される。データとその処理コマンドが混合して表示されないのが分かりやすいが、プリンタでの再処理が困難である。

(2) プログラム方式

文章の処理をするコマンドと処理をされる文章データが混合して表示される。印刷結果を想像しにくい、コンピュータプログラムなのでプリンタで再処理が容易である。

TeX (テフまたはテック) がこの代表である。

タイプライタや WP では上記(1)が多かったが商用印刷、出版では上記(2)が要求される。この中間の存在である DTP では両方の方式を併用している。

しかし、最初はめんどくでも出来上りの美しさや互換性、再処理の容易さなどから上記(2)の方式がだんだん増える傾向にあるが、どちらも一長一短がある。たとえば図-2の(9)の数式は TeX では次のように入力する。

$$SH(j\omega) = \sum_{k=-\infty}^{\infty} h_n^m(k) e^{-j\omega k}$$

一方、WYSIWYG 方式の例では、

$$H(j\omega) = \sum_{k=-\infty}^{\infty} h(k) e^{-j\omega k}$$

と入力して、サイズや行間を調節する。

いずれもあまり楽ではない。

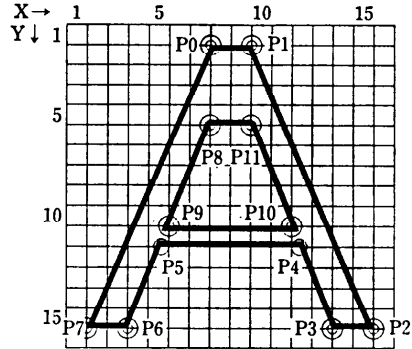
3.2 文字処理

文字の形がデザインとして統一されたのが書体 (フォント) であり、文字一つ一つの形状が字體 (グリフ) である。フォントの表現形式としては次の三つがある。

- (1) ドットフォント
- (2) ベクトルフォント
  - 2-1 輪郭線 (アウトライン) フォント
  - 2-2 骨格フォント

図-3 に 16×16 ドットの "A", 図-4 にアウトラインフォントの "A" を示す。

図-3 の表はドットフォント "A" をファクシミリ標準符号 (MH, MMR) で圧縮したものでこのような圧縮形式をランレングスフォントと呼ぶこともあ



(a) "A" の輪郭特徴点 Pi

```
Code=[PLN(P0P1P2P3rP4rP5P6P7P0)|
      PLN(P8P9P10P11P8)]
      =[PLN((7,1)(9,1)(15,15)(13,15)r(11.5,11)r(4.5,11)
            (3,15)(1,15)(7,1))|
        PLN(7,5)(5,10)(11,10)(9,5)(7,5)]
```

PLN(Polyline) Code=01

r(Rational Point) Code=1111+2bit

$$\{C\} = 2 \times \{PLN\} + 2 \times \{r\} + (12+2) \times \{Pi\}$$

$$= 2 \times 2 + 2 \times 6 + 14 \times 2 \times \log 16$$

$$= 128$$

(b) Contour Code

図-4 "A" の輪郭符号化

る。符号量はドット、ランレングス、ベクトルの順に減少し、特にベクトルフォントの符号量はサイズにほとんど無関係なので大きなサイズの文字ほど有利となる。

また、ベクトルフォントは図-5、図-6のようなアフィン変換が可能なので DTP では盛んに使用される。しかし、サイズが小さいときはドットフォントが一番きれいなのでドットフォントも併用される。

アウトラインフォントの生成式は、従来の欧文では円弧や楕円が多かったが PDL (Page Description Language) からの要求もあって最近ではベジェ曲線に代表されるパラメトリック多項式曲線が用いられ、特に複雑な曲線が多い漢字やひらがなでこの傾向が強い。

この n 次ベジェ曲線 B は n+1 個の特徴点 Pi から次の式で生成される (図-7(c))。

$$B = \sum_{i=0}^n C_i (1-t)^{n-1} t^i P_i \quad (1)$$

• Ci: 2項係数, 0 ≤ t ≤ 1; B, Pi ∈ R²

また、曲線のふくらみが自由に変えられる 3次拡張ベジェ曲線 BE は次の式で生成される<sup>7),13)-15)</sup> (図-7(d))。

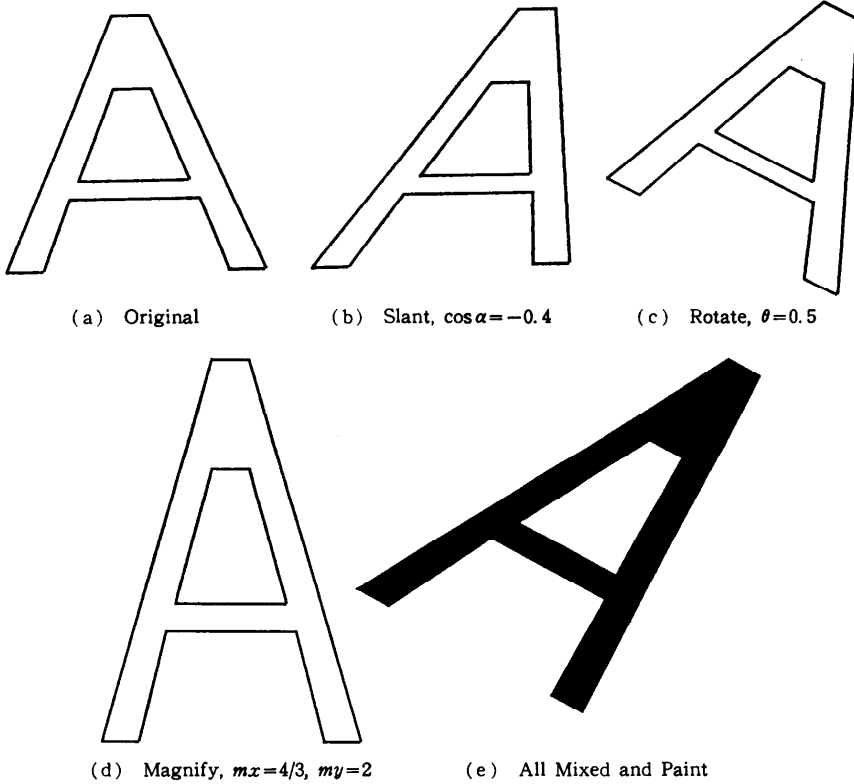


図-5 アフィン変換

$$BE = \sum_{i=0}^3 (1-t)^{n-i} t^i C_i \quad (2)$$

$$C_0 = P_0, C_1 = cP_1 - (c-3)P_0$$

$$C_3 = P_3, C_2 = dP_2 - (d-3)P_3$$

c, d: ふくらみ係数; c, d ∈ R<sup>1</sup>

ベジェ曲線の問題点は数学的に正確な円弧を描くことができないことであるが、三つの解決法がある。

- (1) 有理多項式にする<sup>8),12)</sup>.
- (2) 円弧を分割してベジェ曲線で書く.
- (3) 多重多項式で生成する<sup>14)</sup>.

図-8, 図-9 は“つ”を拡張ベジェ曲線で表現した例で文字の太さ (weight) を変えた例である。

文字輪郭線の高品質、高速展開技術と輪郭線間のペイント技術も曲線の曲率適応分割や幾何学的展開、ペイントの高速化などで 2000 文字/秒以上の高速、高品質文字生成が可能となった。一般的には RISC-CPU などの汎用高速 CPU を使用するよりも、2次元 DMA や 2 ビットマッププロセッサ<sup>10),15)</sup>などの専用プロセッサのほうが桁違いに高速化できる。



図-6 “a”と“泳”の自動ベクトル化とアフィン変換

### 3.3 文章の論理処理

欧文では Spell Checker や Semantics Evaluator などにより単語の誤りを指摘したり文章の論理的適切さを評価、採点したりする処理ソフトウェアがある。

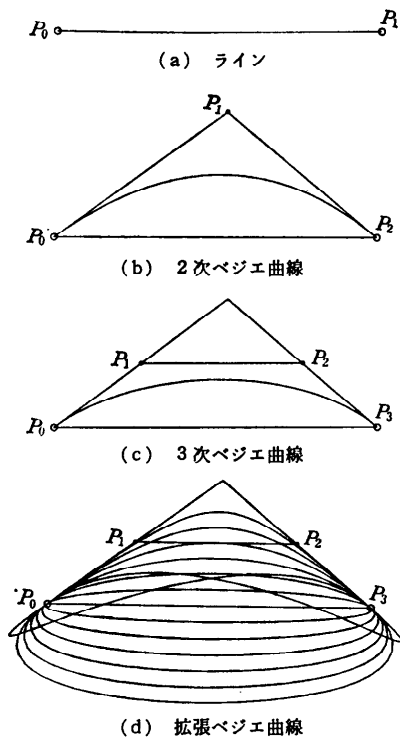


図-7 ベジェ曲線と拡張ベジェ曲線

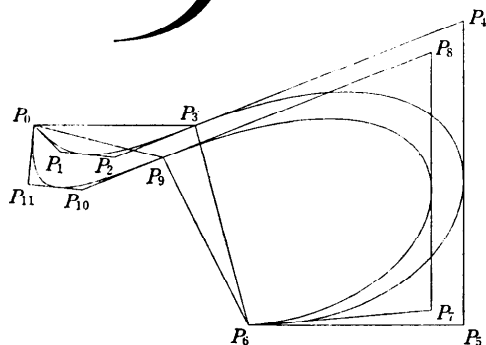


図-8 拡張ベジェ曲線による“つ”の生成

しかし、日本語は単語の切れ目さえ明確ではなく、Semantics もあいまいなこともあり、まだこのような

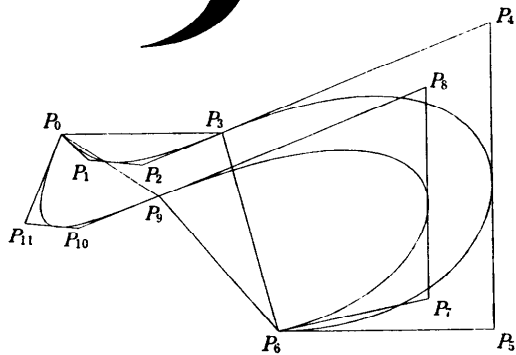


図-9 拡張ベジェ曲線による“つ”の生成

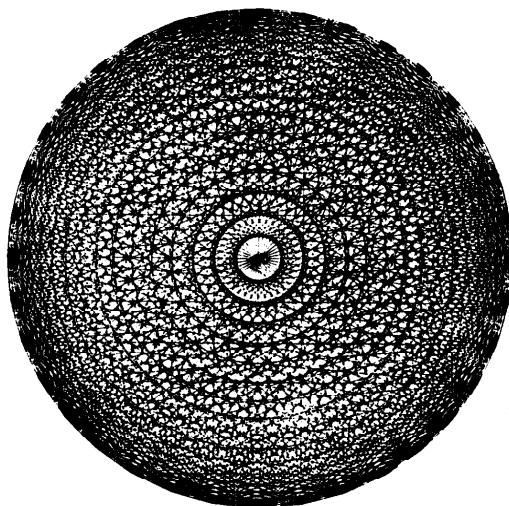


図-10 線輪

処理ソフトウェアが少ない。せめてかな漢字変換での珍変換（賃変換、賃返還、朕変換、ちん返還、朕返還、鎮変換など）の Checker の出現が待たれるところである。

### 3.4 図形（グラフィクス）とグラフの処理

図形の処理はアウトラインフォントと同じで

- (1) ライン
- (2) 円弧

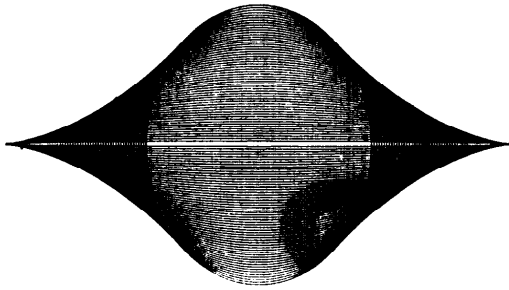


図-11 楕円宇宙

(3) ベジェ曲線

が描ければよい。また、グラフは円、折れ線、棒、レーダチャートの4種が必要である。数式曲線は、座標軸目もりが等間隔と対数の2種で4つの組合せが必要である。円グラフ(パイチャート)や棒グラフでは内部に任意パターンのタイリングが必要である。

図-10は直線描画、図-11は楕円、図-12はタイリング処理、図-13は数式曲線の例である。

3.5 イメージ処理

DTPの最終出力機はレーザープリンタなどを前提としているので、写真なども疑似中間調出力つまり網点やディザを用いたハーフトーン画像となる。

たとえば64階調のハーフトーン出力のときに、母マトリックスサイズを8×8として、出力サブマトリックスサイズを8×8、4×4、3×3、2×2、1×1とすれば図-14のように簡単に画像サイズの変更ができる。

商用印刷では網点の母マトリックスは12×12以上であるがDTPでは母マトリックスは8×8程度が多い。また、図-14の例では網点パターンの種類は4種類選択可能である。

3.6 統合編集

図-2のように文書はテキスト、図形、画像の3種類から構成されるが、このうちテキストのみは物理的制約が少なく、たとえば複数ページに分割可能であるが、図と画像は分割不能である。そこで統合エディタでは図と画像のエリアを予約してテキストは予約エリアを避けて流し込む。この場合、テキスト、図形、画像のソースファイルは独立していて、統合ファイルにはそのファイル名やレイアウト情報だけが必要である。

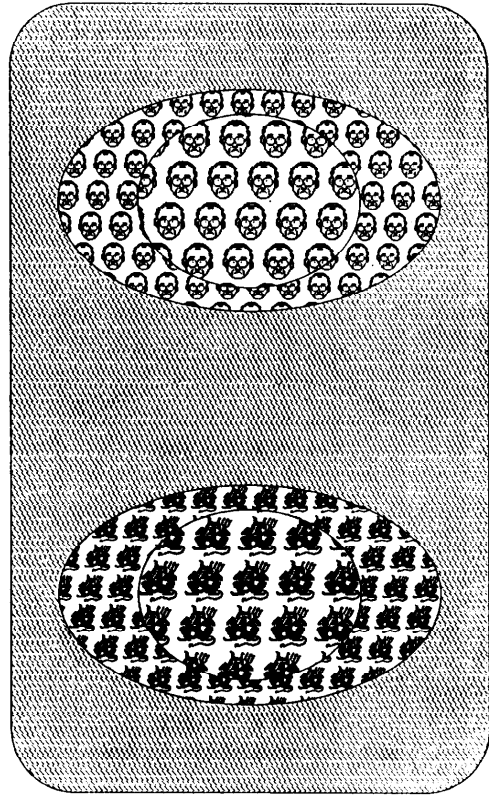


図-12 タイリング

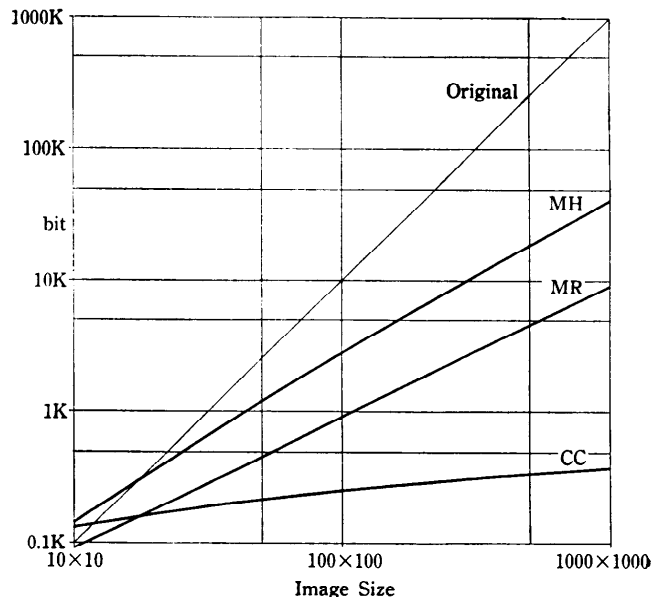


図-13 "A"のサイズと符号量





図-14 網点出力 (サブマトリックス法)

#### 4. DTP の標準化

DTP は品質的にも機能的にも中間的なものでありそれ自身を対象とした標準はないが、次の2系統の標準が関係している。

(1) タイプライタ、WP (Word Processor)、ファクシミリなどの OA 機器の互換性を目的としたオフィス文書のための標準

(2) 印刷、出版関係の標準

前者はオフィスドキュメントアーキテクチャ (ISO) またはオープンドキュメントアーキテクチャ (CCITT) であり、後者は SGML-DSSSL-SPDL-FONT の印刷関連標準である。全体の関係を図-15 に示す。

##### 4.1 ODA (ISO 8613)

(ISO: Office Document Architecture)

(CCITT: Open Document Architecture)

オフィス文書のアーキテクチャ (定義) とそのインタフェースを定める。

なお、ISO も CCITT と同じように Open Document Architecture と名称変更の予定である。

##### 4.2 SGML (ISO 8879)

(Standard Generalized Markup Language)

文書の項目のマークアップ用言語である。

##### 4.3 DSSSL (審議中)

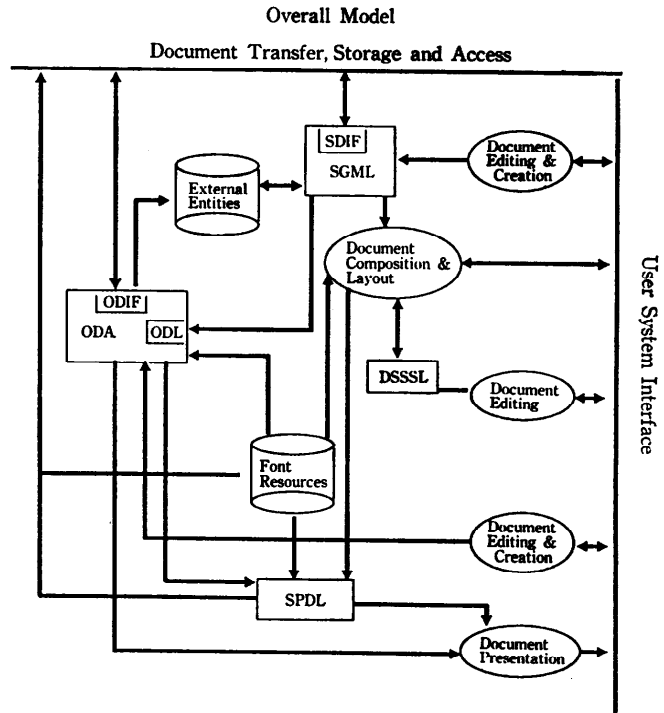
(Document Style Semantics and Specification Language)

SGML でマークアップされた項目の形や意味を記述する言語。

##### 4.4 SPDL (審議中)

(Standard Page Description Language)

プリンタに出力するときの記述言語。最終出力なので編集はできない。デファクトとしては Adobe 社の PostScript<sup>6)</sup> があるが Microsoft 社の TrueImage も有力な新興勢力である。なお、前者の文字生成は3次ベジェ曲線によるが後者は Conic (2次) ベジェスプラインによる<sup>8)</sup>。



SGML Standard Generalized Markup Language  
 SDIF SGML Document Interchange Format  
 ODIF Office Document Interchange Format  
 ODL Office Document Language  
 ODA Office Document Architecture  
 DSSSL Document Style Semantics and Specification Language  
 SPDL Standard Page Description Language

図-15 標準の統合モデル

##### 4.5 フォント (ISO 9541, 1-3), グリフ (ISO 10036)

9541-1 で定義 (アーキテクチャ), 9541-2 でフォーマット, 9541-3 でフォント形状の記述を定める。アメリカの AFII (Association for Font Information Interchange) がフォント著作権の有料登録をし、日本では規格協会の文字フォント開発普及センタが登録業務をしていて、互換性を上げるためにフォントのテキストフォーマット<sup>10)</sup>を提唱している。

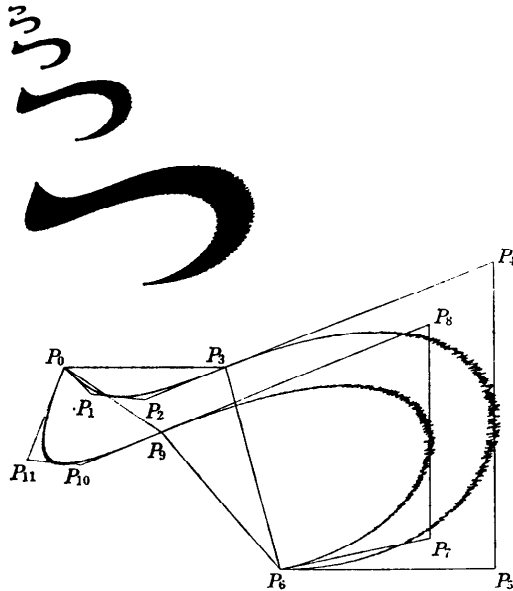
なお、グリフ (glyph) とは文字一つ一つの形、つまり字形のことであり、フォントはその集合、つまり書体のことである。

### 5. DTP の将来

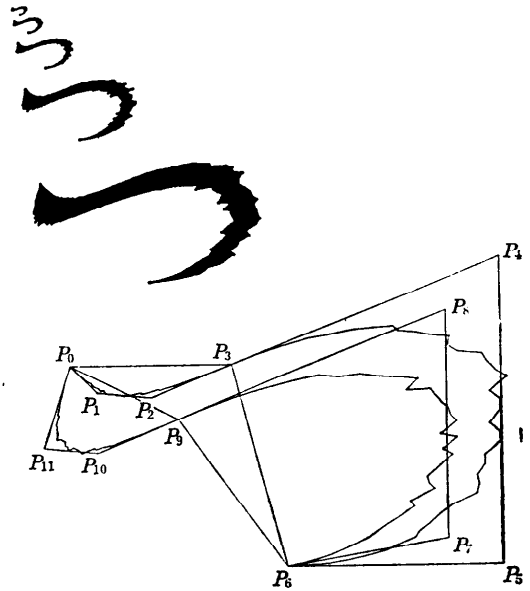
#### 5.1 フルカラー画像の統合

もし図-2 のような文書を作成しようとするときテキストだけなら TeX で編集して PDL で出力すればよいが、図形やイメージは別にして統合編集する必要

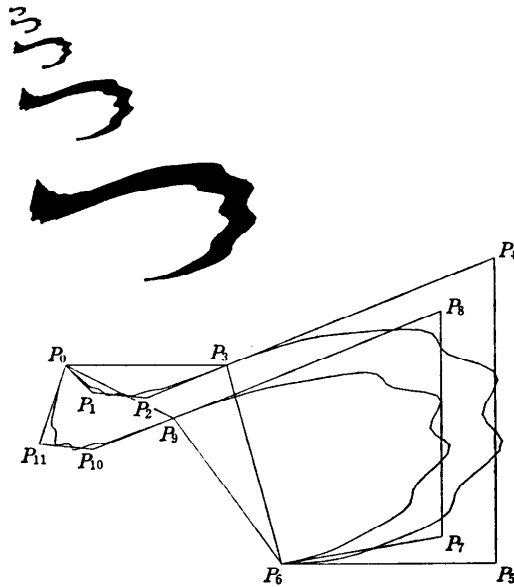
がある。特に、イメージの部分のフォーマットの標準はまだない。レーザープリンタやイメージスキャナの独占的供給元である日本からよい提案が待たれるところである。幸いにも膨大な情報量があるフルカラー画像の圧縮符号化標準はほぼ完成したので今後の DTP への応用が期待される。



(a) 拡張ベジエ曲線によるフォント展開 (乱数変調)



(b) 拡張ベジエ曲線によるフォント展開 (乱数変調)



(c) 拡張ベジエ曲線によるフォント展開 (正弦変調)

図-16

## 5.2 WYSIWYG 方式とコマンド方式の併用

日本独特の文字入力法としてかな漢字変換があるが、これを発展させて数式変換や、英日、日英変換などを含め、WYSIWYG とプログラム方式のよい点を併用すれば大変使いやすくなる。

## 5.3 互換性

現在の WP は他社どころか自社の製品間でも互換性がとれないほどメーカーエゴが強いが、今後の DTP ではユーザ要求と、メーカーエゴの両立が期待される場所である。なによりも一度作った文書はいつでも再利用できるようにネットワークやデータベースの互換性の確立がまず第一歩であり、そのためにも当学会などが率先して電子投稿などの試みをすべきであろう。

## 5.4 社会環境

日本のオフィスが能率に無関係な悪平等を脱皮し、生産性の高い人が高い給料をもらうようになればキーボードアレルギーもなくなるだろう。

一部の大学ではレポート作成にコンピュータターミナルを使用する動きがあるが、これが誘因となって幼稚園や小学校でアメリカのようにキーボード叩きが始まり、キーボード塾の出現の可能性もある。

## 5.5 現代フォントの開発

フォントの分野では今後も活字、光(写植)、デジタルの3種併存が数十年は続くが次第にデジタルの比率が上がるだろう。

また、“宋”、“明”や“ゴシック”などのクラシックフォントに加えて CAD による豊富な現代フォントの出現も期待される場所である。図-16 は“つ”を変形した例である。

## 5.6 処理速度

DTP の処理速度の改善は今後の大きな問題である。しかしこの点でも、幾何学的フォント展開により加算とビットシフトだけで文字生成する方法、2次、3次ベジェ曲線の DDA (Digital Differential Algorithm)、2面のビットマップによる高速ペイント法などを採用した RIP (Raster Image Processor) により 10-100 倍もの高速化が達成できる見通しができてきた。

## 6. おわりに

DTP の現状と動向について述べた。このように DTP は今後解決すべき多くの技術課題を抱えている。

画像品質と文字品質の向上はもちろんのことであるが、一度作った文書は機種に関係なく互換性がとれる

ようにユーザの声を反映する必要がある。

また、標準化が進んだとしても、プロフェッショナルでない人が使用する DTP では、文書の校正などはとても大変なので、自動校正支援システムが必要である。今までコンピュータによる文書処理分野の支援は必ずしも十分ではなかったが、最適に近い美しい文書をコンピュータの支援でほぼ自動的に作れるようにするのが今後の課題である。

最後にサンプル作成に協力してくれた稲庭智子さん、広江泰俊君、笠原龍夫君に感謝いたします。

## 参考文献

- 1) 村山：デスクトップパブリッシングの技術，画像電子学会誌，Vol. 17, No. 3, pp. 151-160 (1988).
- 2) 村山，江成：DTP 用フォント，画像電子学会誌，Vol. 18, No. 3, pp. 97-107 (1989).
- 3) Knuth, D. E.: The TEXbook, Addison-Wesley (1984).
- 4) Knuth D. E.: The METAFONTbook, Addison-Wesley (1986).
- 5) Bézier, P.: Emploi des Machine a Commande Numerique, Masson & C<sup>ie</sup>, Paris (1970).
- 6) Adobe Systems Inc.: PostScript, Addison-Wesley (1985).
- 7) Murayama, N.: A Polynomial for Automatic Contour Vector Coding, Trans. IEICE, Vol. E 72, No. 5, pp. 661-669 (1989).
- 8) Pavlidis, P.: Curve Fitting with Conic Splines, ACM Trans. on Graphics, Vol. 2, No. 1, pp. 1-31 (1983).
- 9) Karow, P.: Digital Formats for Typefaces, URW Verlag, Hamburg (1987).
- 10) 日本規格協会：高品質フォントの標準化調査研究報告書 (1989, 1990).
- 11) 大野：TEX 入門，共立出版 (1989).
- 12) 斎藤，穂坂：拡張 2 次有理ベジェ曲線による高品位文字フォントの生成とその特徴，情報処理学会論文誌，Vol. 31, No. 4, pp. 552-570 (1990).
- 13) 村山：拡張ベジェ曲線による整数図形の輪郭線符号化，電子情報通信学会画像符号化研究会，PCSJ '89 予稿集，pp. 15-16 (1989).
- 14) 村山：曲線生成のための多重多項式，情報処理学会論文誌，Vol. 31, No. 8, pp. 1213-1220 (1990).
- 15) 村山：拡張 3 次ベジェ曲線によるフォント生成，情報処理学会グラフィクスと CAD 研究会予稿，45-1, pp. 1-8 (1990).
- 16) ISO: 8613 (ODA), 8879 (SGML), 9541 (Font), 10036 (Glyph).

(平成 2 年 8 月 2 日受付)