



講義ノートの周辺

Part 2 情報と社会

3. 情報の種類と符号化

森 隆一

$\alpha\beta\gamma\delta\epsilon(\epsilon)\zeta\eta\theta(\vartheta)\iota\kappa\lambda\mu\nu\xi\omicron\pi(\varpi)\rho(\varrho)\sigma\varsigma\tau\upsilon\phi(\varphi)\chi\psi\omega$

Γ Δ Θ Λ Ε Π Σ Υ Φ Ψ Ω

ABCDEFGHIJKLMN OPQRSTUVWXYZ

abcdefghijklmnopqrstuvwxyz

abcdefghijklmnopqrstuvwxyz

ABCDEFGHIJKLMNOPQRSTUVWXYZ



目次

第3章	情報の種類と符号化	51
3.1	情報の格納	51
3.2	文字	56
3.3	自然数	60
3.4	整数	62
3.5	実数	66
3.6	情報量	69
3.7	音声	76
3.8	画像	82
3.9	動画	88

第3章 情報の種類と符号化

3.1 情報の格納

情報とは、“一定の約束に基づいて、人間がデータに与えた意味”とした。

ここでは、“コンピュータで利用できるように一定の約束に基づいて、データを符号化したもの”とする。

各種のデータに対して、それぞれの符号化の規格が幾つかある。

コンピュータ内では、情報はメモリに格納される。

メモリの説明には次図のような幾つかの箱を並べたものでなされている。



1つ1つの箱をビット (bit) というが、箱の内容を表すこともある。各ビットには0または1が入り、対応するメモリの状態を表す。

8この箱を一まとめにして扱うことが一般的である。これを1バイト (Byte) という。このとき、左端のビットを最上位ビット、右端のビットを最下位ビットという。また、8個を左の4個と右の4個に分け、左を上位ビット、右を下位ビットともいう。

ここで、自然数の2進記数法に触れておく。

2進記数法とは、0と1の列で数を表す方法である。すなわち、右から k 番目に1があれば、 2^{k-1} があるとみなすことである。

たとえば、11010は、右から2番目と4番目に5番目に1があるから、 2^1 と $2^3 = 8$ と 2^4 とを加えて、26となる。これを式にしてみれば

$$2^1 + 2^3 + 2^4 = 0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4$$

を計算したことになる。

このとき、番号は0から始めた方が式は少し簡単になる。

親指を0位、人差し指を1位、中指を2位、薬指を3位、小指を4位、とし、立てたときに1 曲げたときに0とすれば、5桁の2進数に対応することができる。グーは00000に対応し、パーは11111に対応する。

とすれば、片手で0から $2^5 - 1 = 31$ まで表すことができる。

両手を使えば、0から $2^{10} - 1 = 1023$ まで表すことができる。また、上の11010には、小指・薬指・人差し指をたて、親指・中指を曲げた状態が対応する。

0と1の5個の列の総数は $2^5 = 32$ であるので。片手では0から31までの数を表すことができる。さらに、両手を使えば、指は10本になるので、0から $2^{10} - 1 = 1023$ までの数を表すことができる。

ただし、薬指と中指を単独で動かすのは難しい。右利きの場合、右手はある程度動かせても、左手動かすのは難しい。

8個の箱では、 $2^8 = 256$ であるから、0から255までの自然数を扱う

ことができる。

箱の数を語調ということもある。語調が w のときは、0 から $2^w - 1$ までの数を扱うことができる。

ここで、初めに考えた 8 ビットの箱を考える。今回は、各ビットの値を、右から、 b_7, b_6, \dots, b_0 を書き加えることにする。

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
-------	-------	-------	-------	-------	-------	-------	-------

このとき、 b_7 が最上位ビット、 b_0 が最下位ビットとなる。また、 $b_7b_6b_5b_4$ が上位ビット、 $b_3b_2b_1b_0$ 下位ビットとなる。

p 進記数法にも触れておく。2 進記数法や 10 進 4 記数法の。2 や 10 を p におきかえたものである。 n 桁の p 進数を $a_{n-1}a_{n-2}\dots a_1a_0$ とすれば、各 a_i は $0, 1, \dots, p-1$ の数となる。 $\dots 0, 1, \dots, p-1$ の数で、 $\sum_{k=0}^{n-1} a_k p^k$ を表す。

ぎゃくに、 x に対する $a_{n-1}a_{n-2}\dots a_1a_0$ を求めるには、 p による割り算を余りが 0 になるまで実行し、得られた余りを順番に a_0, a_1, \dots とすればよい。

ここで、 $p = 4$ の場合を考える。この場合は、 $0 \sim 2^4 - 1 = 15$ の数に対応させられる。

16 進表示では、16 個の 1 文字の数字を準備する必要がある。数字とアルファベットしかない米語では、 $0, 1, \dots, 9$ の次は、アルファベットを使う

しかなく、 $0,1,\dots,9,A,B,C,D,E,F$ を (16進) 数とした。(16進数)

これを用いて、上位4ビットを16進の第1位、下位4ビットを第0位とすることにより、00000000 から 11111111 までの2進8桁の数を 00 から FF までの16進2桁の数として表すことができる。

コンピュータでは、すべての情報がビット列として保持される。実際には、ビット列に対応する回路の状態である。

ビットの列ではあるが、一定の長さのビット列を単位として扱う。この長さ w を (機械) 語長という。語長は2の倍数が使われ、少し前までは32であったが、最近では64となった。機械の制御では $8 \cdot 16$ も使われているらしい。

プログラムにおいては、変数に格納される。というよりも、プログラムは変数に対する処置を並べたものといえる。したがって、多くの言語では、変数の型を指定することからプログラムは始まる。変数の型には、非負整数 (自然数)・整数 (・有理数)・実数 (固定・浮動小数点)・文字などがある。

これは、数によって格納の方法が異なる、言い換えれば、各ビットの意味が異なることによる。したがって、演算も同じ型同士で行う必要がある。次節以降である程度のことを説明していく。

自然数 (非負整数) は2進記数そのものである。

筆者がプログラムを行っていたのは16ビットから32ビットに移行した

時期で、パソコン等では、整数は2バイトで表している。

$2^{16} = 65,536 = 6,5536$ である。

当時はワーク・ステーションと呼ばれるクラスがあり、ここでは、4バイト用いるのが標準のようだ。パソコンでも unsigned long を用いれば、4バイトが使えた。

$2^{32} = 4,294,967,296 = 42,9496,7296$ である。

42億9496万7296 四十二億九千四百九十六万七千二百九十六

Windows10 の電卓では、 2^{64} は実数計算となり、
1.844674407370955e+19 と表示される。

1844,6744,0737,0955,0000 1844京6744兆0737億0955万

2^{128} は 3.402823669209385e+38 で、3402澗8236溝6920穰9385ジョ
となる。

ここまでは、漢字の単位が準備されている。3文字の単位は初めの1文字を用いれば、残りは、正・載・極・恒・阿・那・不・無の8つで、 2^{256} には足りなくなる。

ハード・ディスクやUSBメモリなどの補助記憶装置外部情報はファイルに格納される。各ファイルは‘ファイル名+拡張子’で区別される。拡張子は txt, doc, jpg, bmp, exe, tex, などがある。

ファイルは‘ディレクトリ’に分類され、格納される。

3.2 文字

コンピュータが事務処理に導入してしばらくは、英数字とカタカナしか扱えなかった。また、プリンターはドット・ライン・プリンターであった。印刷用紙も両端に穴の列があるコンピュータ用紙が主であった。

ラインというのは、行を印刷するという意味である。したがって、縦に印刷することは非常に難しい。

次は、Eメールの初期によく起きた文字化けである。

```
SJ2bIAcLUwDK+hwM8bMcJBjz0UhfXjlIjWZykIjM5CADGecg3c44B5nEOAepy
DgHicRcDlJjxz1IjWZycMFtLQepxOocpNtYm4Nsf9M0ysEF61s0MheMc5CKrM
5BG1PQWT1yk0JPvBd95CA77XE00pBZnYNumuWql0Numk+f1oNEx0qbsJ965B7
Fh96x5Z6jM6j47qh1JbWt+VjWhptF5zzydcE7LV8HzsF8XQKp+boE0vKVeuno
1SBfqcTFcQ5rWem2jMEtKzd5+pGvC1HQ8nXgJFnTjs4ZN00WvDNo2juUvmnvU
aRNe8tkqnxyrCCWvvtUCnvK4CC7F8acEQB+i8NOKIAPZYGHFHE9PggnozxdJ
```

この文字化けは漢字コードの設定ミスによる。

文字の符号化としては、モールス符号や手旗信号などが古くから使われている。Wikipedia「モールス符号」では“短点 (·) と長点 (-) を組み合わせて文字を表現する。表現する文字種の違いにより、複数の規格がある。”ということである。さらに、幾つかの規格の表が挙げられている。

文字の符号化は ASCII コードが基本となる。ASCII は American Standard Code for Information Interchange の頭文字をとったものであり、半分は全てのコードに共通である。

ASCII コードを直接扱ったのはプログラムをする人の一部しかいなかったと思われる。

次の表は ASCII コードのうち、16進表示で1位が 1 から 4 までの部分で、全体の $\frac{1}{4}$ の 64 個である。

10進	16進	文字	10進	16進	文字	10進	16進	文字	10進	16進	文字
16	10	D _E	32	20		48	30	0	64	40	@
17	11	D ₁	33	21	!	49	31	1	65	41	A
18	12	D ₂	34	22	”	50	32	2	66	42	B
19	13	D ₃	35	23	#	51	33	3	67	43	C
20	14	D ₄	36	24	\$	52	34	4	68	44	D
21	15	N _K	37	25	%	53	35	5	69	45	E
22	16	S _N	38	26	&	54	36	6	70	46	F
23	17	E _B	39	27	'	55	37	7	71	47	G
24	18	C _N	40	28	(56	38	8	72	48	H
25	19	E _M	41	29)	57	39	9	73	49	I
26	1A	S _B	42	2A	*	58	3A	:	74	4A	J
27	1B	E _C	43	2B	+	59	3B	;	75	4B	K
28	1C	→	44	2C	'	60	3C	<	76	4C	L
29	1D	←	45	2D	-	61	3D	=	77	4D	M
30	1E	↑	46	2E	.	62	3E	>	78	4E	N
31	1F	↓	47	2F	/	63	3F	?	79	4F	O

16進の番号を用いれば、10 から 1B までは制御文字である。

最上位ビットが 1 のコード表 (16進で 80 から FF まで) には、カタカナや罫線記号が含まれているように、国により異なる。この意味で、日本語 ASCII コードと呼ばれている。

昔のプログラムの本には、このコード表が必ずあった。

漢字は2バイトを用いる。したがって、 $256 \times 256 = 65,536$ 個の文字を扱

うことが可能である。このうち、初めの1バイト分は、日本語 ASCII コードが占めている。

この対応を漢字コードといい、JIS、shift JIS、new JIS、uni code、euc code などがある。

JIS と shift JIS の経緯は何かで読んだことはあるが、覚えていない。覚えているのは、JIS は CP/M-86 系で用いられ、shift JIS は DOS 系 (MSDOS、PCDOS、DOSV) で用いられたということだけである。パソコンでも Unix が使えるようになり、uni code がつくられた。euc code は名前からして EU code を連想させる。

漢字には、教育漢字と当用漢字という指定があった。さらに、人名漢字というものも指定されている。

ある本で、第1水準 2965、第2水準 3388、計 6353 の指定があり、第1水準には、漢字の他に、ひらがな 83、カタカナ 86、数字 10、英字 52、特殊記号 147、ギリシャ文字 48、ロシア文字 66、罫線文字 32、計 524 が含まれると書かれていた。

さらに、幾つかが追加された。第3水準 (1,249 字) ・第4水準 (2,436 字) が指定されているとのことである。

Windows10 では IME パッド 文字一覧に漢字コードを見ることができ、U とあるから uni code と思われる。

Windows 10 では ‘環境依存文字’ というものがある。‘京・垓・ジョ’ のジョはこの環境依存文字ではフォントがあるが、筆者の本稿を作成する環

境では扱えない。四庫全書 (あるいは維基文庫) の漢字はいくつあるかはわからないが、“異体字をを考慮すれば、32万字は必要である”ということを知った記憶がある。“これでも異体字を全て表すには不十分である”も続いていた気がする。4バイトでは 65,536 で足りないが、6バイトでは 16,777,216 となる。現在は64ビット = 8バイトCPUであるから、8バイトは 4,294,967,269 となり、43億弱のフォントが扱えることになる。

漢字以外では1言語100フォント、書体・異体字を含めてその100倍の1万フォントで間に合うのではないかと考える。

8バイトフォントでは、漢字以外の全ての文字の全ての書体・異体字と、漢字の幾つかの書体・異体字が収録して相当残ると思われる。この部分には、交通標識・地図記号・図形記号などの図形フォント、および、4バイトコードを2・3書体の収録できる。

このようなフォント体系の作成に携われる人材が揃っているのは、日本と中国のみであろうと思っている。日台で進めるのが問題が少ないかと思う。

3.3 自然数

自然数は 2 バイト (16 ビット) または 4 バイト (32 ビット) の 2 進数として扱われる。

加法は 2 進加法である。上の桁では長くなるので、1 バイト (8 ビット) の場合で考えてみる。

まず、 $2 + 1$ を考える。2 は (00000010) で 1 は (00000001) である。

第 0 位と第 1 位は、ともに、0 と 1 であるから、足して 1 になる。

結果は (00000010) + (00000001) = (00000011)

$255 + 1$ を考える。255 は (11111111) で 1 は (00000001) である。

第 0 位はともに 1 であるから、0 と桁上がり 1 となる。

第 1 位は 1 と 0 であるが、桁上がりがあるから、

0 と桁上がり 1 となる。

第 2 位から第 6 位も同様に 0 と桁上がり 1 となる。

第 7 位も上と同じであるが、桁上がりが問題である

9 ビットならば (100000000) となるところであるが、

この処理ができないので、ほっておくことにする

結果は (11111111) + (00000001) = (00000000) となる

これは、 $\text{mod } 2^8$ の演算が行われたことになる

1 バイト自然数を使えるかどうかはコンパイラにより異なる。

C では char で宣言可能

p 進位取り記法では $0, 1, \dots, p-1$ の n 個の列 $(a_{n-1}a_{n-2}\cdots a_1a_0)$ には $x = \sum_{k=0}^{n-1} a_k p^k$ が対応する。

逆に、 x に対する $(a_{n-1}a_{n-2}\cdots a_0)$ を求めるには、 p による割り算を実行し、得られた余りを順番に a_0, a_1, \dots, a_{n-1} とすればよい。

日本語の数詞

ひい、ふう、みい、よう、いつ、むう、なな、やあ、ここ、とおを考えてみる。

‘はち’に関しては、末広がり、八十島、ウソ八百などの沢山ある意味に使われている。これは、 $1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot$ 沢山 という概念のあったことの名残とも思える。さらに、 $1 \cdot 2 \cdot 3 \cdot$ 沢山もあったかもしれない。

秒 分 時は60進となっている。60の訳数は $1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 10 \cdot 12 \cdot 15 \cdot 20 \cdot 30$ と多い。逆数を取れば、

$$\frac{1}{60} \cdot \frac{1}{30} \cdot \frac{1}{20} \cdot \frac{1}{15} \cdot \frac{1}{12} \cdot \frac{1}{10} \cdot \frac{1}{6} \cdot \frac{1}{5} \cdot \frac{1}{4} \cdot \frac{1}{3} \cdot \frac{1}{2}$$

が整数となることである。

演習を60等分し、各分店を通る半径を60本放射状に引いておく。各半径の先に、 $\frac{p}{q}$ ($q = 3, 4, 5, 6, 10, p = 1, 2, \dots, q-1$) を書いておけば、ケーキを切るときに役立つかもしれない。

3.4 整数

負の数は中学校で習うことになっている。

中学校学習指導要領(平成29年度)解説 数学編では

正の数と負の数を用いることによって，

- ・反対の方向や性質を数で表すことができること
- ・ある値を基準とした数の大小の比較ができること
- ・数を数直線上に表すことができること
- ・減法がいつでも可能になること
- ・加法と減法を統一的に表すことができること

様々な事象における問題解決の場面において，正の数と負の数を用いて変化や状況を分かりやすく表したり，能率的に処理したり，その意味を読み取ったりすることができるようにする。例えば，図書館の本の1日の貸出冊数など，設定した目標値を基準として，その目標値からの増減を正の数と負の数を用いて表すことにより，目標の達成状況などを明確に示したり把握したりすることや，仮平均を定めて処理することにより，効率よく平均を求めるなどの活動を設定することが考えられる。このように様々な事象を正の数と負の数を用いて考察し表現することで，それらを活用することができるようにする。

と書かれている。

整数の符号化は符号と絶対値方式と補数方式の2通りがある。

まず、符号と絶対値方式は簡単で、最上位ビットを符号に用いる。8ビット

トの場合では、ビット列を $(b_7b_6\dots b_1b_0)$ とするとき、 $b_7 = 0$ の場合は +、 $b_7 = 1$ の場合は - とし、 $b_6\dots b_1b_0$ を絶対値とする方法で、 $(-1)^{b_7} \sum_{k=0}^6 b_k 2^k$ が対応する。

-127 から 127 までの数を表すことが出来る。

一方、補数方式は最上位ビットに -2^7 (-2^{w-1}) の位取りを与える。このとき、ビット列の表す数は $-b_7 128 + \sum_{i=0}^6 b_i 2^i$ となり、-128 から 127 までの数を表すことが出来る。

もう少し、補数方式をみていくため、自然数 n が与えられたとする。

$-n$ を補数方式で表わしたものを $1b_6\dots b_0$ 、 $b_6\dots b_0$ の表す数を x とする。

$$-n = -2^7 + x$$

両辺に 2^8 を足せば、

$$2^8 = n + (2^7 + x)$$

ここで、右辺の $(2^7 + x)$ は $1b_6\dots b_0$ 、を普通の2進位取り記法としたときに、対応する自然数となる。これから、 $-n$ の表現 $1b_6\dots b_0$ は、2進位取り記法で n の 2^8 に関する補数を表したものとなる。したがって、 n の2進表記で、全てのビットを反転すれば、 $-n$ の補数方式の表記が得られることになる。。

補数方式の利点は、ビット列を普通の2進表記と違って、加減算をした

結果が、実際に加減算と同じになる。

これを幾つかの例に当てはめてみる。

$$-1 + 2 = 1 \quad \longleftrightarrow \quad (11111111) - (00000010) = (00000001)$$

$$-1 - (-2) = 1 \quad \longleftrightarrow \quad (11111111) - (11111110) = (00000001)$$

$$0 - 1 = -1 \quad \longleftrightarrow \quad (00000000) - (11111111) = (11111111)$$

“いつでも (1 00000000) を借りてくることができる” と考える。

すなわち、 $(\text{mod } 2^8)$ で考える。

幾つかのコンパイラと幾つかの変数型で、 $\sum_{k=1}^n k$ および $n!$ のプログラムを作成し、さらに、 $n!$ の値を変えて結果を見てみると面白い。

プログラムでは、整数の次は実数になるが、数学では分数(有理数)を習う。分数は分母と分子という、正の整数と整数の組で表される。これを用いるには、上記2成分をもつ変数をもつ変数に対する演算を造ることが必要となる。

問題は2つある。1つは、CPU にセットされている整数と実数の演算に対して、プログラムで造る演算は実行速度が遅いことである。もう1つは、必要に応じて、ビット列を長くすることと、長さの異なる変数の演算のプログラムの作成である。

このプログラムには、数学科の学生に対するプログラム演習の範囲を超えるものと思われた。それよりもマルコフ連鎖のプログラムのほうがプログラム演習のテーマとしては適していると考えた。また、この頃 Math-

ematica という数式処理が可能なソフトが話題になり、これを用いれば、任意の桁の分数演算ができた。これらにより、任意の桁の分数演算には挑戦しなかった。

複素数については、その必要性からか、FORTRAN に組み込まれている。

3.5 実数

実数の符号化としては、よく知られている固定小数点方式とマイナーな浮動小数点方式がある。

まずは、非負実数の固定小数点方式から述べる。

これは、与えられた k を用いて、 $\dots b_k$ の右に小数点をおいたとする記数法である。すわわち、 w 語のビット列 $b_{w-1}b_{w-2}\dots b_1b_0$ は w 進数で $b_{w-1}\dots b_k.b_{k-1}$ を意味し、 $\sum_{i=0}^{w-1} b_i 2^{i-k}$ が対応する。

これを言い換えれば、 2^{-k} を最小単位として、その $\sum_{i=0}^{w-1} b_i 2^i$ 倍を表すと言いかえられる。加減算は、 $\text{mod } 2^{w-k}$ の意味で正確に実行できる。乗算は、小数点以下を k 桁にする必要がある。 $k+1$ 桁以下の処理は、0 捨1入と思われるが、確かめていない。

実数では、補数方式と浮動小数点方式が用いられているということである。

補数方式は、固定小数点方式で、最上位ビットに -2^{w-1-k} の位取りを与え、他の i ビットには 2^{i-k} の位取りを与える方法である。したがって、表すことのできる数は

$$-b_{w-1}2^{w-1-k} + \sum_{i=0}^{w-2} b_i 2^{i-k} \quad \text{となる。}$$

この場合扱える数は、 -2^{w-1-k} 以上で 2^{w-1-k} 未満で、 2^{-k} を単位としたものとなる。

$w = 24, k = 16$ の場合は、 $-2^7 = -128$ 以上で $2^7 = 128$ 未満で、 $2^{-16} = \frac{1}{65536} \simeq 0.000015258$ を単位とすることになる。

補数方式では、扱える数に制限があることが問題となる。

また、無理数を有限の範囲で扱うことから、扱える数のうちで近いもの(近似値)で代用することになる。

$k = w - 1$ の場合は、 $-b_{w-1} + \sum_{i=0}^{w-2} b_i 2^{i+1-w}$ となり、 $-1 \leq x < 1$ の

実数を 2^{-w+1} を単位として計ったものになる。

$-x$ は 2 の補数となる。

一般的に用いられているのは浮動小数点方式で、指数表記と呼ばれる符号化である。これは $b2^a$ の形で表す。 b を仮数、 a を指数という

b には上で述べた $k = w - 1$ の固定小数点表記を用いる。ただし、 $b_{w-2} = 1$ となるようにする。また、 a は補数表記された整数である。 b に w ビット、 a に v ビット用いるとすれば、 $b_{w-1} \cdots b_0 a_{v-1} \cdots a_0$ となる。

実数に関しては有効数字、不純物の少なさを表すのに 0 が 個並ぶという表現が用いられている。これは 10 進数による指数表記に基づくものである。これを書くときには、加数都市数の間に e または E を挟む。たとえば、 2^{w-1} を 10 進表記したときの桁数を精度あるいは有効桁数という。不純物に関しては、10 進で a が $-$ 以下となることである。

10 進表記と 2 進表記を関係づけるものとして $2^{10} = 1024$ という式があ

る。これから、10進4桁は2進10桁に相当することになる。有効数字12桁とするためには2進で30桁(30ビット)必要ということになる。

16ビット CPU 時代のコンパイラ (Basic, Pascal, C) の実数は6バイト(倍精度が8バイト)であった。コンパイラのマニュアルには、実数変数の各ビットの仕様図が書かれていた。もっとも、当時は上の内容を理解していなかった。というよりも、理解しようとしなかった。

理解しなくてもプログラムの練習には可能であったことと、当時のマニュアルの多くは入門者にとっては読み易いものではなかった。多くは、マニュアルとは言えるものでなく、仕様書にメモを書き加えたといしか言いようのないものであった。

Turbp Pascal と Turbo C のマニュアルは一線を描くものであった。この sample program を作成すれば、普通の入門書より効率よく学ぶことができた。日本でこれに近いものは、改変された一太郎のマニュアルであった。これに関しては、雑誌で次の記事を読んだ。

“マニュアルの校正を頼まれた事務系の女性社員が、こんなものは読めたものではない、との感想を言ったところ、それならば、お前がつくってみろ、といわれ、社内で聞きまくって書き上げた。”

3.6 情報量

物の量を量るということを考えてみる。‘はかる’の変換の候補としては、測る・量る・計る・図る・諮る・謀るが表示される。

「日本大百科全書(ニッポニカ)「度量衡」の意味・わかりやすい解説」では、“長さ、面積、体積および質量の単位、標準、ならびにこれらの計量器について定められた慣習や制度をいう。かつては英語で weights and measures にあたるとされていたが、最近では、英語の metrology に相当する。国際計量用語集(VIM)では metrology は‘測定の科学とその科学の応用’と広い意味に定義されている。”

また、「同度量衡の語源」では“漢書律歷志に「度は分、寸、尺、丈、引である。これが長短を計る手だてである。もと黄鐘の長さにおこる」とある。分、寸、尺、丈、引は単位で、黄鐘はこれらの単位の基礎となる標準 standard である。尺はもともと手幅が起源であるが、人間の手の大きさは一定しない。そこで、定まった音律の笛は一定の長さをもつということを利用して標準を定めたのである。黄鐘は古代中国の十二音律の基準音で、この音律を出す笛の長さを9寸、90分とした。そして10寸が尺、10尺が丈、10丈が引である。

これに続いて「量は龠、合、升、斗、斛である。これが量の多少を計る手だてである。もと黄鐘の龠(管)におこる。度によってその容量を正す。北方に産する平均的な大きさの秬黍1200粒で満つる。井戸水を用い水準を正したときの容量が1龠である。龠を二つあわせて1合とし、10合を1

升、10升を1斗、10斗を1斛として五つの量が定まる」とある。体積の単位も標準は黄鐘管である。龠は竹の管の意で、黄鐘竹管の体積が1龠であるが、小さすぎるので二つあわせて合としたということである。

さらに続いて権は銖、両、斤、鈞、石であり、物を計るに衡を水平にして軽重を知るものである。もと黄鐘の量に応じる重さにおこる。1龠には1200の秬黍が入り、この重さが12銖である。これを二つあわせて両とする。24銖が両である。16両を1斤とし、30斤を1鈞、4鈞を1石とする」とある。黄鐘竹管を満たす1200粒の秬黍を質量の標準としたわけである。質量を計るには権すなわち分銅と衡すなわち天秤がいる。衡はくびきに由来し、転じて秤ざおとなった。

以上が語源であるが、度量衡はまた度量権衡ともいわれる。この漢書の内容は中国の計量制度の基準をなし、それが前漢において確立したことを示している。なお重さまたは重量という語は、今日では力と同じ量とされ、衡は質量の計量のこととされている。英語では mass であるが、一般には重量 weight が慣用されている。”と解説されている。

測る・量る対象は、の長さ、面積、体積および質量とされている。漢書で引用されているのは、長さ・体積・質量である。長さ・重さ・かさ(体積)の順に測り方を身といくことにする。

長さはを測る道具としては物差し(メジャー)が挙げられる。すなわち、長さがわかっているものと比較することで測ることになる。

物差しは、普通 mm 単位のメモリが打たれている。mm の $\frac{1}{10}$ まで読

み取れるようにしたもののノギスである。

重さを測る道具としては、天秤ばかりが古くから用いられている。これは重さのわかっているものと比べる方法である。

次にバネはばりが現れた。加えられた力に応じてバネが伸縮するという法則を利用したものである。由来は調べてないが、時計の出現と同じ頃と思っている。

最近では電子ばかりが登場した。これは、圧電素子という加えられた力に応じて電気を発生するという素子を用いている。液晶表示をすれば、可動部分はないことになる。

かさ (体積) の単位としては、1 辺が 1cm の正立方体の体積を 1cc としている。1合は 180cc、1升は 1.8 ℓ とされている。1升瓶から 1.8 ℓ 瓶に変わったのは、筆者では、物心ついて暫くした頃である。“酔っ払いが 1.8 ℓ 瓶をぶら下げてでは、酔いが醒めてしまう” というような話を聞いた気がする。

時計は何を測っているのか。長年使われていた機械式時計は、ゼンマイの戻る性質を利用して、最初の歯車を回転させ、これに幾つかの歯車を連動させることが基本となっている。以下は、筆者の理解した大雑把な仕組みである。

最初の歯車の1回転に対し $\frac{1}{60}$ 回転する2番目の歯車を準備する。これは、秒針になる。3番目も同様に2番目の歯車の1回転に対し $\frac{1}{60}$ 回転するもので、分針になる。4番目は3番目の1回転に対し $\frac{1}{12}$ 回転するもので、

時針なる。

ここまでのことを使われ方から見ることにする。使われ方のソースは Youtube の DIY と料理の動画である。

料理の動画では、はかりとメジャー・カップと軽量スプーンがおかれ、多くの場合、実際に使われている。

後者はかさ (体積) を測るもので、主として液体に用いられる。例外として、米はかさで測っている。一方、穀物を挽いて粉にしたものは重さで測っている。前者は重さを測るもので、肉など変形はするけど形のあるもの測っている。一方、長さに関しては適当な場合が殆どである。

DIY のうち小屋づくりを例にとれば、色々な量が測られている。極稀な場合を除き、建物は矩形の土台の上に建てられる。

初めに行うのは、この土台を作る場所を指定することで、‘縄張り’ とよばれている。かつては経緯儀と呼ばれる道具か水盛りと呼ばれる方法で行われた。

skylinkjapan.com > [blog](#) > [survey-equipment](#) 「測量に使われる機器の種類とそれぞれのメリットデメリット」では、

セオドライト (別名 トランシット) は、経緯儀とも呼ばれる、角度を測定する機械のことです。

トータルステーションとは、光波距離計と電子トランシットを組み合わせた測量機です。一台の機械で角度と距離を同時に計測することが出来ます。ターゲット (プリズム) を使用して2名で作業を行うものと、ター

ゲットを使用しないレーザー光で測定するモデルがあります。ターゲットを自動追尾してくれるモデルもあり、その場合一人で作業を行うことが可能です。また測量業務に必要な電子野帳のプログラムが内蔵された機種もあります。

と書かれている。

束石の上に小さい小屋を造る場合は、束石に土台を架け水準器で束石の高さを調整することでも可能である。最近ではレーザー水平器を用いて、水平・垂直を測ることが行われている。

縄張りの次の作業は、コンクリート作業である。コンクリ-・砂・砂利の比率が問題とおなるが、目分量で行われていることが多いが、かさ(体積)で配分しているようである。粒と粉のあつかいであろうか。

この後は、木工や塗装などが行われる。後者で測ることが必要となるのは塗料の混合であろう。一方、プレハブ住宅以外では、集めた素材を加工して用いることが基本となる。このとき、必要となるのは、長さを測ることと直角を描くことである。長さを測るには、メジャーを用いるもののほかに、現物合わせという方法もある。ベテランの大工は、細かい細工では両者を併用していた。すなわち、少し大きめに切ったものを現場合わせて修正していた。

情報の量(の大きさ)を考えてみる。

コンピュータで扱う情報は、0と1の列であると述べた。コンピュータでは、対応する素子(メモリー)の状態を0または1に対応する状態に変化させることで、取り込まれる。この対応により、1つのメモリーを1つの箱とし、そこに0または1が格納されていると見做せることになる。

そこで、箱の数、すなわち、列の長さを情報の量とすればよいことになる。しかし、度量衡で扱ったものとは異なり、0または1の列に変える方法により、その値は異なってくる。

1つ1つの箱をを1ビット(bit)と呼ぶ。これは2つの状態を取り得るという意味も入っている。また、8ビットを1バイト(byte)という。記号は、ビットは小文字のb、バイトは大文字のBを用いる。

3桁毎に単位が設けられ、

キロ・メガ・ギガ・テラ・ペタ・エクサ・ゼタ・ヨタ

が準備されている。コンピュータでは、2進法が基本となっているので、 $2^{10} = 1024$ を 1000 に当てている。現在、ハード・ディスクの容量がテラバイト TB 単位になっている。

日本(中国)では4桁ごとに単位が設けられ、漢字1つのものは

万・億・兆・京・垓・ジヨ・穰・溝・澗・正・載・極・恒・阿・那・不・無

が準備されている。スーパー・コンピュータで‘京’というものがある。これは、詳細は調べてないが、浮動小数点表示された実数の演算を1秒に1京回実行できるコンピュータという意味である。

キロ・メガ系と万・億系の簡単な対応を示す。

キロ	メガ	ギガ	テラ	ペタ	エクサ	ゼタ	ヨタ
0.1万	100万	10億	兆	0.1京	100京	10垓	ジヨ

ここでは、キロ・メガ系4単位が万・億系3単位と一致する。

3.7 音声

音は縦波であることと、磁器テープとレコードのアナログな保存については 2.1 節で述べた。

音声もデジタル化という過程を通して符号化される。PCM 録音というレコードを買った頃は上のような知識はなかった。

このレコードは、筆者の装置でも音がよいことは聞き取れた。一方価格は2-3割高く、録音時間は半分ほどであった。普通の LP は大きい音や可聴域より低い周波数の音をカットしていた。これは溝の幅を狭くすることになったが、このPCM 録音の LP はこのカットをせずに作成したということであった。

PCM 録音のメリットの1つに、テープヒスがないことである。これが顕著に表れるのは、小さい音である。

ここで「PCM 録音」で検索し、次の解説を見つけた。

PCM デジタル録音の歴史 橋本 尚久 2016年10月8日 18:45

音楽用のコンパクトディスク (CD) が発売されたのが1982年ですが、その10年前に既にデジタル録音は行われていました。実は世界初のデジタル録音を成功させたのは日本で、会社は日本コロムビア、今の DENON でした。

この時、日本コロムビアが実用化したのは映像記録用 (VTR 用) の2インチテープに13bit・47.25kHzのPCM デジタルの音楽を記録できる業務用PCM レコーダーの「DN-023R」という機種でした。この当時はまだ

デジタル記録に最適な量子化ビット数やサンプリング周波数がよく分かっておらず、手さぐり状態だったので、13bit・47.25kHzという中途半端なフォーマットになっていたようです。

この47.25kHzというなんとも中途半端なサンプリング周波数も、VTRの水平同期周波数15.75kHzのちょうど3倍でこういう数値が選ばれたようです。1974年の2号機では量子化ビット数が13bitが14bitになり、1979年の4号機でようやく16bitになります。

ソニーもPCM録音技術の開発を行っていて、1977年にPCM-1というVTRへの録音機を発売しました。この後オランダのフィリップス社の光ディスク技術を取り入れて誕生したのがCDです。当初のフィリップス案では直径11.5cmで14bitの60分だったのですが、ソニーはあくまで16bitを主張、かの有名なオーケストラ指揮者のカラヤンはベートーヴェンの交響曲第9番「合唱」がまるまる収録できる74分を希望、16bitで74分にするため、CDの直径は12cmになったそうです。この16bitにしたことが後のPCでの処理に大変都合が良く、CD-ROM用としても広く使われることになりました。

と書かれている。この記事から、1972年にPCM録音が開発され、1982年にCDが発売されたということである。

この間に、PCM録音の技術の応用として、DATが発売された。 $\frac{1}{4}$ インチのカセット式磁気テープを記録媒体とするものであったが、あまり普及しなかった。

CD-ROM もこの後普及した。Windows 3.1 の供給媒体は CD-ROM とフロッピーの選択が可能であったが、Windows 95 では CD-ROM のみであった。

ネットで音楽が聴けるようになったが、これも何時頃かは覚えていない。2002年のハーゲンでの海外研修では音楽を FM で聞いていた気がするが、2012年のダルムシュタットでは、宿舎で聞いていた人がいたので、この間であろう。初めは、数分でしか容量がなかったようで、クラシックは細切れで聞いていたが、2010年代には制限がなくなった。最近は曲の途中でも広告が入るようになった。

「音声 符号化 規格」で検索したら、MP3、WAV、AAC、AIFF、WMA、FLAC などの規格があることがわかった。この中で、MP3 は音質の良い規格であるが、当時の筆者の環境では再生が間に合わず、読み込みを待つことが多かった。

音は縦波である。各楽器がそれぞれの音波を発振し、我々はこれらが合成されたものを聴くことになる。1点での音圧(電流)の変化をグラフにしたものを見る装置がオシロスコープである。潜水艦の戦闘を扱った動画では必ずと言っていいほど現れる。

音波の変化は、連続曲線で表される。この曲線を再現できれば、音が再現されることになる。数学の理論で、

“波を $\frac{a_0}{2} + \sum_n^{\infty} (a_n \cos nx + b_n \sin nx)$ と表す”

ことができるかどうかを扱うものがある。これは、フーリエ解析と呼ばれる分野の出発点である。

上に述べた性質を定理としようとも考えたが、数学をやっていたものとしては、関数に対する条件を述べないといごごちが悪い。

関数を考える場合は定義域を指定する。ここでは閉区間 $[0, 2\pi]$ とこの上で関数 $f(x)$ を考えるとする。

上の(関数の)級数が収束して $f(x)$ になるための条件が求められている。この条件を調べたが、簡単に述べられるには理解できなかった。

このような場合、講義では次のように処理されている。

定理をキチンと述べて証明しない

条件を付けて証明する

応用とか歴史の話でお茶を濁す

ということで、条件が満たされた場合

$$a_n = \frac{1}{\pi} \int_0^{\pi} f(x) \cos nx \, dx \quad b_n = \frac{1}{\pi} \int_0^{\pi} f(x) \sin nx \, dx$$

となる。 a_n 、 b_n はフーリエ係数と呼ばれている。

音叉というものがある。Wikipedia「音叉」では、

音叉 (Tuning fork) とは、正弦波を描く特定の周波数の音または固有の

振動数を発するU字状(2又)に別れた金属製の器具である。そして全体で見るとy字型である。

概説: 音叉の素材は鋼(炭素鋼)、軽量化のためにアルミニウムやジュラルミンが用いられることもある。全体的にU字形をしており、底部に柄が付いている。腕の部分を叩くなどして振動させると音を発する。それ自体の音は極めて弱いため、音を聞くには柄の部分を耳に近づけたり歯でくわえたりするか柄を共鳴しやすいものに触れさせる。音叉の発する音はほぼ純音である。叩いた直後にはさまざまな上音を含んでいるがこの形では基音以外の音は持続し得ないのですぐに消え去り、純音が得られるのである。

理科実験などでは共鳴箱付音叉が用いられる。共鳴箱は共鳴により大きな音を発生するために音叉の下に取り付ける箱で、この箱は1つの面が空いており、そこから音が出るようになっている。

音楽用の音叉は楽器の調音や発声の基音を確認するのに用いる。医療用の音叉は聴覚検査のほか脳外科の感覚検査や糖尿病の進行度を調べるためのフットケア用に用いる。ドップラー効果を利用したスピードチェッカーの校正に音叉が用いられている。

周波数シンセサイザが普及するまでは、発振回路の信号源として音叉発振器が利用された。また、音叉型水晶振動子がクォーツ時計などに利用されている。

と書かれている。

単振動の合成を行う装置としてシンセサイザーがある。このもとになった動詞 *synthesize* は ‘合成する’ という意味をもつ。

音のデジタル化の各規格を見ると、色々の解説が書かれている。

ここで、筆者の描いているデジタル化のイメージを述べておく。

簡単な関数の中に、階段関数というものがある。これは、定義域を互いに素な区間の和に分けて、各少区間上では定数となる関数である。このうち、2進区間 $[k2^{-n}, (k+1)2^{-n})$ で定数をとるものを、2進階段関数とよぶ。さらに、関数のとる値も $j2^{-m}$ に制限する。

デジタル化は、近似する2進階段関数を求めることと違って大差ない。言い換えれば、 n と m を定めておいて、もとの関数 $f(x)$ を、各区間 $[k2^{-n}, (k+1)2^{-n})$ で $j2^{-m}$ に置き換えることである。

問題は j をどうきめるかである。

2進階段関数としては、ウォルシュ関数やハール・ウェーブレットなどが知られている。

3.8 画像

画像をコンピュータで処理している分野としては写真 (デジカメ)、図形 (イラスト、CG)、製図 (CAD、JWCAD) などで、芸術としての絵画以外はコンピュータで処理に置き換えられている。

まずは、画像のデジタル化に関して、筆者の抱いているイメージを述べておこう。

図形は2次元のためである。モニタを連想すれば、長方形を2進正方形 $[k2^{-n}, (k+1)2^{-n}] \times [\ell2^{-n}, (\ell+1)2^{-n}]$ に分割し、そこでの3原色の各成分の値を2進で保存すればよい。

この小正方形の個数が画素数とされている。また、縦の個数 \times 横の個数を解像度ともいう。

3原色は赤 (Red) \cdot 緑 (Green) \cdot 青 (Blue) である。3原色の各成分の値を指定する方法を RGB 方式という。

これに対し、シアン (Cyan) \cdot マゼンタ (Magenta) \cdot イエロー (Yellow) の3色の値を指定する方法を CMY 方式という。理由は理解していないが、プリンタはこちらを採用している。

もう1つ、色相 (Hue) \cdot 彩度 (Saturation) \cdot 明度 (Value) を指定する方法を HSB 方式という。

3つの数値 それぞれに 8 bit = 1 byte が割り当てられている。

写真を (数十倍の) ルーペで見れば、色のついた円が幾つか見られる。

すなわち、写真は点画といえる。

特に飾りたいと思うものを覗いて、画像の多くはディスプレイで見るとことになる。ディスプレイの解像度の規格として、VGA (Video Graphics Array) というものがある。

始めてみたディスプレイは VGA (640x480) であり、ブラウン管であった。ガラ携の解像度はこれであった。また、10年前のパソコンがこんなに小さくなったとも思った。

SVGA (800x600) は記憶が薄い。画素数は VGA が 307,200 で SVGA が 480,000 で 1.5 倍である。この程度では、SVGA の解像度がなければ仕事の効率が落ちる人以外は、無理して買う気にはならなかったのではないと思う。

XGA (1024x768) は、筆者が初めてパソコンを購入したとき、標準解像度のようにであった。漢字を表示するには 32x32 の画素が必要であったこともあり、高解像度のモニターが求められた。

後に発売されたプロジェクターもこの解像度の期間が相当続いた。

1990年代に、ブラウン管モニターは液晶モニターに換った。カラー液晶 XGA モニターの最初の販売価格は100万円であった。また、解像度は忘れたが、カラー液晶モニター搭載のノート・パソコンは40万円であった。

これらに続いて、XGA系のディスプレイは、

WXGA (1280x768)、SXGA (1280x1024)、UXGA (1600x1200)、

WUXGA (1920x1200)、QXGA (2048x1536)、WQXGA (2560x1600)

と続いた。この系列は、横縦の比が 4:3 で、W の付くものは、この比を 8:5 と横長にしたものである。

また HD のつく系列があり、

FHD (1920x1080)、WQHD (2560x1440)、QFHD (3840x2160)、
UHD (7680x4320)

の種別があった。これは、HiVision テレビ放送 (衛星放送) の規格に応じたものである。動画再生が広がるにつれ、VGA 系から HD 系に替った。横縦の比は 16:9 である。この後は、HD を付けるのはやめて、FHD を単位の、画素数を測り、

1K (1920x1080)、2K (2560x1440)、4K (3840x2160)、
8K (7680x4320)

が用いられている。

まえに、“4000x3000 が実現されれば写真がデジタル画像に代ってしまう”と述べたが、これについてももう少し考えてみる。‘石仏あれこれ’や‘写真アルバムから’での写真は、引き伸ばした写真をスキャンしたものを、Word の画面で少し縮小する程度で納まるように解像度を下げて用いた。この過程で、上の予想を次のように修正する必要があると感じた。“A3 の画面に 4000x3000 が実現されれば写真がデジタル画像に代ってしまう”。

A3 の規格を見たら、mm 単位で、A3 は 420x297、A3 ノビは 483x329 とあった。一方、筆者の使っている FHD パソコンは、縦 21.7cm で、2.4

で割れば 9 吋である。1080 を 9 で割ると 120dpi (dots per inch) となる。A3 の縦 297mm に同様の計算をすれば、12.4 吋で、3000 を 12.4 で割れば 240dpi となる。

これは、今の画素を縦横ともに密度を倍にしたもので、4K と 8K の間である。4K 有機 EL モニタが市販されているので、研究段階では実現されているのかもしれない。

初期 (10 年前) のデジカメ 100 万 ~ 300 万 画素であった。

筆者の初めて買ったエプソンのデジカメの画素数は $1024 \times 768 = 768K$ である。これを 3 倍すれば、2.3M となる。これより、各点の色を指定するビットマップ方式では、2.3MB の容量となる。

実際には、各写真の容量は 150KB 前後となっている。この比は $\frac{2300}{150} \sim \frac{1}{15}$ となる。

次の機種でも、画素数は 1024×768 であるが、各写真の容量は 300KB 前後となっている。フラッシュ・メモリの容量が増え、設定を変えたためかもしれない。

さらに、次の機種では、解像度 4000×3000 と上の辺のが約 4 倍で、1200 万画素となる。各写真の容量は 3MB 前後となっている。オーダー的には、ほぼ 16 倍である。

これは、画像圧縮が行われていることをしめす。

画像圧縮の規格としては JPEG、JPEG2000、GIF、PNG、RAW な

どが知られている。

圧縮の方法は、記事を見ても理解できていない。標語的に、JPEG は重み付フーリエ級数、JPEG2000 は Daubechies ウェーブレットを利用しているということである。

上で述べた画像ファイルの規格の説明を引用する。

JPEG: デジタルカメラで撮影したデータなどに使われており、画像規格の中でも、もっともポピュラーな規格です。アプリとの互換性も高くパソコンに取り込んで利用しやすい規格ですが、記録するときに圧縮して保存しているため、圧縮率を高くしてしまうと画質が劣化してしまいます。

BMP: Windows で標準に使用される画像形式です。圧縮されていないため、データ容量が大きくなります。

GIF: インターネットの Web ページの画像としてよく使われている規格です。扱える色数が 256 色のため、色数が少ないイラスト等に適しています。また、透明色を設定できるため背景の色を反映することができます。年賀状などの切り抜き写真に便利です。

PNG: GIF と同様にインターネットの Web ページの画像としてよく使われています。透明色の利用もでき、GIF よりも扱える色数が多いですが、品質劣化のないデータ圧縮が可能なのが特長です。

RAW: 「RAW」は「生」という意味で、何の加工もされていないデー

タです。そのため、ユーザーが色味や明るさを調整することができます。画質にこだわるプロカメラマンの間では浸透している規格です。デジタルカメラのメーカーごとにデータ形式が異なり、対応するアプリもデータ形式ごとに違うため注意が必要です。

3.9 動画

Wikiprdia「残像効果」では、

テレビ放送や映画は、一連の静止画を高速に切り替え表示（映画では1秒間に24枚、NTSC方式のテレビ放送では1秒間に30枚）することによって動画の知覚を起こさせている。現行の方式では、一つの静止画から次の静止画への切り替えは上描きによっておこなわれるが、CRT（テレビの場合）やフィルム（映画の場合）が使われていた時代の技術では、前の静止画の表示が消えている間に次の静止画の表示をおこなわなければならなかった。それにもかかわらず表示されている画像が点滅しているようには見えていなかったのは、これと同等の現象によるものである。ただし、それがまるで動いているように見えるのは、仮現運動とよばれる別の現象である。

Wikiprdia「残像効果」では、

ファイ現象とは、静止画の連続によって引き起こされる仮想の運動を知覚する現象のこと。仮現運動を引き起こす代表的な現象。1912年にマックス・ヴェルトハイマーが発見した。映像やフィルムアニメーションの知覚に関する議論ではよくベータ運動と混同されがちだが、これは定義としてはまた別の現象であって映像の知覚とは直接関わるものではない。

と書かれていて、“残像に基づき、絵を連続して表示させれば、動いて見える”というように単純ではないようだが、映画では1秒間に24枚、テレビ放送では1秒間に30枚表示できれば、動画となるということである。

YouTube で画質を見ると、次のような項目の選択画面になる。

144p 240p 360p 480p 720p 1080p60HD 1440p60HD

ここで、p より前の数字は横方向の画素数を表している。少し前では、720p にも HD が付いていた。上の列で p の次に書かれている 60 は、1秒間に60枚の絵を表示することを意味している。

ここで、解像度 FHD 1920x1080 の画像の JPG での容量を概算してみる。まず、 $1920 \times 1080 = 1.9M$ である。これを 3倍して 15 で割れば、0.38M となる。30倍すれば 11.4MB となり、3600倍すれば、41GB となる。

CD の容量が 500MB、DVD の容量は 5GB と記憶している。この上に、BRD という規格もある。

動画の規格としては次のものが知られている。

MPEG-4、AVI、WMV、MOV、FLV、MPEG-2、VOB、ASF

最後に、ゼネラルアサヒ「動画のファイル形式とは？代表的な形式や種類を紹介！」での寸評を引用する。

MP4: MP4は、高い圧縮率を持ち、比較的小さなファイルサイズで高画質な映像を保存でき、ストリーミングサービスや動画編集ソフトなど幅広いデバイスに対応しており、最新の機器やプログラムへの互換性を確保しています。また、圧縮後の動画品質も他のファイル形式と比べて高く、圧縮した際の画質の低下をできる限り防ぎたい場合におすすめです。

AVI: AVIは、Windowsで標準的に用いられているファイル形式です。

利点はいくつかあり、画質や音質が良かったりファイルを実行するために特定のハードウェアやソフトが必要なかったりなどが挙げられます。一方、ファイルのサイズが大きくなりやすく、また圧縮時には画質が低下しやすい傾向にあります。

MOV: MOVは、Apple社のQuickTime Playerのためのフォーマットです。そのためMacとの相性は抜群ですが、一方でWindowsとの相性が良くありません。高品質な映像と音声を保存できますが、MP4に比べて圧縮率は低く、ファイルサイズが大きくなります。

Windowsで動画を扱う際にはMP4やAVIといった拡張子で使うのが良いでしょう。

WMV: WMVは、Windows Media Playerで再生するためのフォーマットです。高い圧縮率を持ち、小さなファイルサイズで高品質な動画を保存でき、ストリーミングに対応しています。YouTubeやVimeoなどに投稿しても問題なく再生できるため、インターネット上へ動画をアップロードするのに使用できます。ただし、Microsoftによるストリーミングサポートが終了しているため、今後の商業展開はないでしょう。

FLV: FLVは、Adobe Flash Player用のフォーマットで、YouTubeやニコニコ動画などの動画共有サイトで採用されています。Web上でのストリーミング再生に適しており、低速回線でも再生が可能です。MP4に比べて圧縮率が低く、ファイルサイズは大きくなります。また現在ではAdobe社によるFlashのサポート終了に伴い、使用機会は少なくなりました。

WebM: WebMは、Googleが開発したWeb用の動画ファイル形式です。オープンソースで、VP9ビデオコーデックを使用しています。高品質な動画と音声を保存でき、ストリーミング再生に適しています。しかし、一部のメディアプレイヤーでは再生が不可能な場合があります。

MPEG2(.mpg): MPEG2は、高画質を目指した動画圧縮フォーマットであり、ISO/IECのMPEG委員会が策定した標準規格です。MPEG1の後継として開発されました。再生品質が高いことが魅力で、DVDや地上波デジタルテレビなどで用いられています。

厳密にいうとMPEG2-TSとMPEG2-PSという2種類に分けられます。また後継仕様としてMPEG4がありますが、画質を重視する場合にはMPEG2が適しています。しかしファイルサイズが大きく圧縮には向かないため、Web上での共有や再生には向いていません。

MKV: MKVは、海外でハイビジョンビデオなどに用いられているファイル形式で、1つのファイル内に映像と音声と字幕データを格納できるのが特徴です。複数の音声や字幕を格納できることもあり、海外向けに動画を作りたい場合にはおすすめです。ただし、高画質で再生できますが、MP4と比べてファイルサイズが大きくなり、ストリーミングには向いていません。

ASF: ASFは、AVIの良い点を引き継ぎ、デメリットを改善したAVIの後継と言えます。ストリーミング再生や保存に特化して作られており、WMVと同様にDRM機能が付いているなど、高い利便性を備えています。

す。知名度が低いのと、エンコードできるソフトが少ない点がマイナスとなっています。

VOB: VOBは、Video OBject fileの略称で、DVDビデオで動画や音声のデータを記録するためのファイル形式です。複数のファイルを特定のディレクトリ構造に格納した形式になっており、目次のチャプターメニューから好きなシーンにジャンプして再生が可能です。VLCメディアプレイヤーなどのソフトでも再生できます。