

グラフィクス標準化と日本語機能

藤村 是明

電子技術総合研究所

情報処理の各分野・各システム内での日本語処理機能の実現は急速に進められているが、各分野間の統合性・各システム間の相互運用性への配慮が不足している。グラフィクスの分野においても、統合性・相互運用性を保証すべき標準化に際し、日本語機能に関しての検討・具体化は不十分であった。本論文は、統合性・相互運用性を備えた日本語グラフィクス機能の実現のための議論の手引きを目指したものである。具体的には、グラフィクス標準化における文字列機能の検討、日本語機能実現に関わる諸問題の整理、PHIGS標準化に際しての各国語サポートに関する日本提案の紹介を行なう。

Computer Graphics Standards and Japanese Text Processing

Koreaki FUJIMURA

Computer Science Division, Electrotechnical Laboratory
1-1-4, Umezono, Sakura-mura, Niihari-gun, Ibaraki, 305 JAPAN

Japanese word processing is rapidly spreading in many applications and in many information systems, while it is very difficult to integrate the applications and they are not interoperable among different systems. In the area of computer graphics, though integration capabilities and interoperability are improved by the current international standardization efforts, Japanese text support is left un-standardized. This paper is intended to present the basis for discussions about Japanese text support in computer graphics standards. It contains a survey of text support in the computer graphic standards, the problems of Japanese text support and a summary of the proposal for international text support in PHIGS by the Japanese standardization group.

1. はじめに

情報処理の各分野の統合性(分野間の一貫性)、各処理システム間での相互運用性の低さが問題となつて来しいが、日本語機能が入ればさらに深刻な状況が生じる。日本語ワードプロセッシングプログラムで作成したファイルと、同じシステム内の日本語FORTRANで読むことができるという統合性の欠如や、同一メーカーの同一機種上のオペレーティングシステム間で、一々コード変換が必要となるという相互運用性の低さは、多くの利用者の嘆きの種である。

情報処理一般の統合性・相互運用性の保証については、国際的協力の中での解決が可能であるが、日本語機能に肉しては、日本がほとんど単独でその解決にあたらねばならない。それぞれの分野において国際的標準化に加わる一方、日本語機能をその中に盛り込んでゆくのは容易なことではない。

グラフィックスの分野においても、正に上のような状況にある。GKS⁽¹⁾を筆頭とする国際的標準化活動が着々と進んでい一方、ビジネスグラフのような個々の応用プログラム中での日本語機能の実現が、日本語機能を標準化する必要性を高めている。

本論文は、現在のグラフィックス標準化の中に日本語機能を組み込んで行くため、広い範囲の人が議論に加わる際の簡便な手引きとなることを意図して書かれている。第2章では、現行のグラフィックス規格(案)の中での文字列機能の概要を示す。第3章では、日本語機能を組み込むにあつての、様々な問題点の洗い出しを行なう。第4章では、PHIGS国際規格化に際しての各国語サポート方針に関する日本提案の紹介を行なう。

2. グラフィックス規格と文字列機能

この章では、GKSを始めとするグラフィックス諸標準の中での文字列機能の概要の紹介を行なう(各標準の全体的な概要については、IEEE Computer Graphics and Applicationsの特集号の各論文(21~29)を参照のこと)。ここでいうグラフィックス諸標準とは、GKS、GKS-3D、PHIGS⁽⁷⁾、CGI⁽⁵⁾、CGM⁽⁴⁾である(図1参照)。このうち、GKS、GKS-3D、PHIGSは、“in which the tasks users wish to perform are more important than how these tasks will be decomposed into executable elements”⁽⁶⁾ という性格を持つ“task oriented systems”であり、装置およびファイルの一般化であるCGI、CGMとは、文字列の扱いにおいても大きく異なる。以下、前者の代表としてのGKS及び後者のCGI・CGMのそれぞれにおける文字列機能について検討する。

2.1. GKSの文字列機能

GKS⁽¹⁾の基本概念であるワークステーションは、抽象化された出力と抽象化された入力とを構成要素とするが、その双方に文字列機能が含まれている。

抽象化された出力は、出力基本要素(output primitives)と、出力基本要素属性(output primitive attributes)から成る。出力基本要素のTEXTが文字列を生成(表示)するものであり、GKS関数としては、

TEXT (レベル)のa

入力: 文字列の位置(デフォルト)

入力: 文字列 (;)S

が対応している。デフォルトは、Sはそれそれ点型、文字列型である。グラフィックス標準の特徴の一つとして、抽象的な関数が、具体的なプログラミング言語の複数に対応する言語結合(Language Binding)⁽⁷⁾があげられる。TEXTは、

FORTRAN 上位水準 (- Full FORTRAN 77 version) では、

SUBROUTINE GTX (PX, PY, CHARS)

Input Parameters:

REAL PX, PY

CHARACTER*(*) CHARS

この形式をとり、FORTRAN 基本水準 (FORTRAN 77 Subset version) では、

SUBROUTINE GTXS (PX, PY, LSTR, CHARS)

Input Parameters:

REAL PX, PY

INTEGER LSTR (length)

CHARACTER*80 CHARS

とある。また Pascal level 1 では、

procedure GText (

text position : GRpoint;

charstring : packed array [

min..max : GInt1] of char);

とあり、

Pascal level 0 では、

procedure GTextString (

text position : GRpoint;

string length : GInt1;

charstring : GString);

とある。ここで GRpoint は x, y (real)

の二成分を持つコードであり、

GInt1 は、0 以上、1 以上の

整数、GString は packed array

[1..GMaxstring] of char である。

このように、抽象的データ型としての文字列 S の定義が、

S string number of characters

and character sequence

とあり、このように、一々文字数を指定する

必要のない言語では、この項は省略

される。ただし、同じ FORTRAN 77 上位

水準にあっても、文字列入力の場合は、

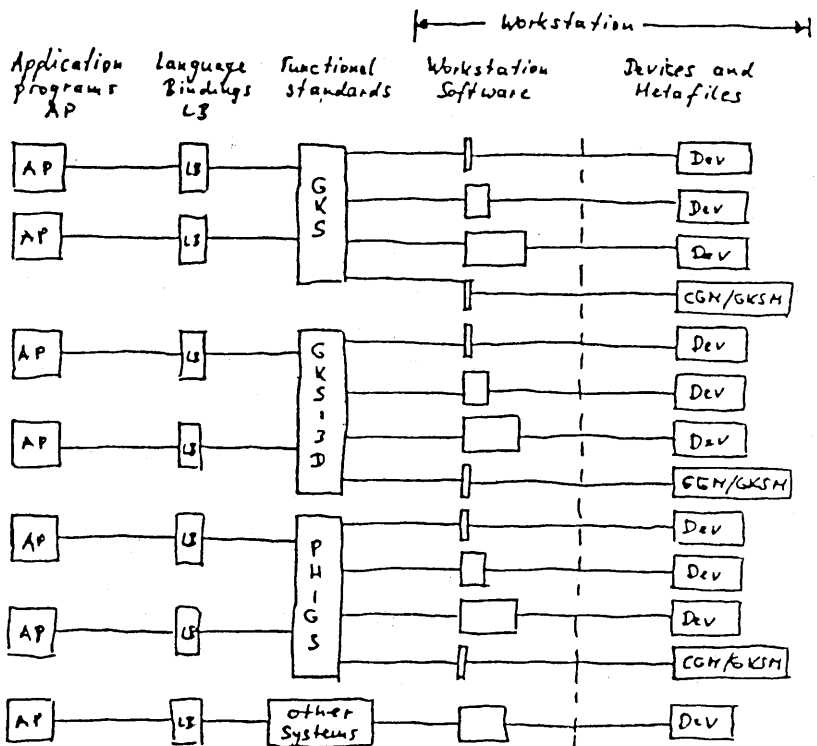


図 1. グラフィクス諸標準の関係

(Figure 1. Computer graphics standards interrelationships)
CGI 2nd Initial Draft (WG2 N356, 11/85) Part 1

CGI
Interface

character string

→ INTEGER LOSTR
number of characters returned
CHARACTER*(*) STR string
と対応づけられ、文字数は省略される。

文字列の出力として、TEXTの他に、MESSAGE 関数があり、プログラムからオペレータにメッセージを送るのに使われるが図形要素ではない。また、グラフィック入力の装置の機能指定 (INITIALIZE xxx 関数を用いる) で、入力促進 (prompt) やエコーの方法として文字列出力が行われることがある。

文字列出力の“見え方”を決定する出力基本要素属性については、日本語機能との関わりのあるところで述べることにする。

抽象化された入力は論理入力装置で考えられるが、その入力値の一つとして、文字列入力がある。要求・抽出及び専断といった制作用モードの違いに応じて、REQUEST/SAMPLE/GET STRING の3種の関数を用意されている。また入力装置の初期設定の関数 INITIALIZE STRING での入力引数で文字列の初期値を設定することができる。当り前のことだが、表示される文字列の源としては、以上のグラフィックス入力装置の他、プログラム上の文字定数、FORTRAN の READ 文のように入力装置以外からの入力機能、さらには各種のサブルーチン等の出力が考えられる。

2. 2. CGI, CGM の文字列機能

CGI, CGM における文字列機能はいくつかの点で GKS (GKS-3D, PHIGS) のような “task-oriented” 標準と異なっている。

第一に、CGI, CGM では、出力基本要素として、TEXT, RESTRICTED TEXT, APPEND TEXT の3関数があり、GKS に比べ、途中で属性の変化する表

示や、枠内表示を可能にしている。

次に、CGI, CGM では ISO 2022 (JIS-C-6228) の情報交換用符号の拡張法に基づき、複数文字集合という概念を積極的に採用している。すなわち、前述の TEXT 系3関数の文字列引数については、CHARACTER CODING ANNOUNCER 関数で、BASIC/EXTENDED 7/8 BIT の符号化手法指定を行ない、CHARACTER SET LIST 関数で各文字集合指示 (designate) の種別文字を悉に登録し、CHARACTER SET INDEX 関数及び ALTERNATE CHARACTER SET INDEX 関数で、その巻の番号により G0 及び G1, G2 待機所への指示をし、文字列内部の SI, SO 等の制御文字で G0~G3 の呼び出し (invoke) を行うことにより、符号拡張を可能にしている (符号拡張については文献(10)~(12)を参照のこと)。一方 GKS 等では、表示可能文字についての規定は無く、むしろ書体 (text font) の番号が ISO 646 (JIS 6220) で定義されている文字を視覚的に表現できることを規定しているだけである。

GKS と CGI, CGM における文字集合の考え方の差は、図1に示したようなそれ以外の位置づけをその予言したものであり、プログラミング言語の世界における符号化独立性と情報交換の世界の符号化依存性との差である (CGI に対して言語結合を考えると⁽⁵⁾、符号化独立であるべき OS 応用層のユーティリティとも考えられる。事務文書体系 - Office Document Architecture = DPO613 一の中で CGM を採用するとともにどういうことになったのか?)。

CGI, CGM と GKS の文字列機能は、入力においても若干の差があるが、新関数の関数で省略する。

3. 日本語機能の諸問題

3.1. プログラミング言語

QKSに対する言語統合の例からも解るように、グラフィクスにおいて日本語機能を実現する場合、基となるプログラミング言語自体の日本語処理能力が問題となる。

日本語処理能力を備えたプログラミング言語については多くの考察がなされているが⁽¹²⁾⁽¹³⁾、データの内部形式に着目すると、次のように分かれる。

① 文字型の複教化によるもの：これは日本語処理FORTRANのJIS原案⁽¹⁶⁾のように、従来の1バイト1文字を暗黙に仮定した文字型の他に、新しいデータの型として、多バイト1文字の型を用意するもので、混在は許されない。

② 1文字の多バイト化：これは、従来の1バイトで扱っていた文字も含め、全てを2バイト(3や4も考えられている)で扱うというものである。

③ シフト方式：これは、1バイト符号と2バイト符号とをシフトコード(狭義のSI, SOとは限らない)で切り替えるものである。

④ ビット化：8ビット目を利用して、シフトコードを用いずに、1バイト符号と2バイト符号とを混在させる。シフトJISコード、ATT拡張UNIXコード、日本DECコード、シフトJIS汎用漢字コード⁽¹²⁾⁽¹⁵⁾等が考えられている。

以上が、現在の日本語処理プログラミング言語におけるデータ内部形式であるが、抽象的な言語機能という点からすれば、①の複教化文字型と②~④の拡張文字型とに大別される。

3.2. 日本語文字列の表示様相

QKSやCGI・CGMでは、文字列表示のために多くの属性を用意している。

これらの属性が日本語表示にとって適当かどうかを完全に論じる余裕/能力は筆者にないが、簡単に問題点を整理して見ると、

① 縦書きは可能か：QKSの文字属性の説明図(Fig. 4-6)を見れば、TEXT PATH=DOWNとするだけで、一応のことは出来るように見える。

ただし、この場合JIS-C-6232(16ビット字形)やJIS-C-6234(24ビット字形)で規定(している)ようにあいう...などの字については、縦書きの字形を別に用意する必要がある。

② 日本語文字と欧文文字の混在：見やすいようにするためには、日本語文字をベースラインからキャプラインまで一パイに使って表示することが望ましいが、このようにすると、縦書きの際の文字列配置が面倒になる。

③ QKS等では、文字の幅は完全にワークステーション依存としている為、日本語文字と欧文文字との比率といった実際上は重要な問題は隠されている。

④ QKS等では復帰改行を始めとして、すべての制御文字が及んで無規定としているので、ワークステーション非依存の表現効果は、せいぜい見出し程度の文字列にしか期待できない。このため本格的な文書作成の際、日本語固有の問題として取り上げられる。禁則処理、ルビ、割注、半角・全角・倍角の使い分け、縦書きの際の欧文文字の扱いなどは、QKSレベルのグラフィクスにとっては問題外となる。

4. PHIGSにおける各国語サポート

に関する日本提案

グラフィックスの国際標準化は、ISO TC97(情報処理) / SC21(開放型システムにおける情報の流通とその管理) / WG2で行われていゝ。これに対応するものゝ情報処理学会規格委員会 SC21 / WG2小委員会(服部幸英主席以下WG2委員と略)である。WG2委では、現在標準化が進められていゝ PHIGSの中に各国語サポート機能が入れられれば、既に標準化の済んだ GKSの持来の見直しの際に同様の機能が取り入れられようとの見直しを待つて、日本語を含む各国語サポート機能の標準への組み込みに取り組んでいゝ。具体的には、WG2委の下に、J-PHIGSという十数名のアドホックグループ(宇野栄代表)を設け、PHIGS一般についての検討と並行して、多国語サポートの提案作成を行ない、本年(1986)9月の Egham 会議に提出した。

この提案(International Text Support in PHIGS)は、A4版14ページにのぼるもので、ここではその骨子のみを紹介する。中は"introduction"から"conclusion"までの5項から成る。"introduction"では、各国語機能が持つべき表示の問題ではなく、他の分野との統合性も考えに入れべきと述べた後、次の様子を基本要求

- ① PHIGSでの文字列は、情報交換ではなく内部処理の対象であり、各システム間で異なった内部表現が許され、符号化独立でなければならぬ。
- ② PHIGSでの各国語サポートは、プログラミング言語に埋め込まれるものだから、持来の各国語サポートプログラミング言語標準に適合できるものでなければならぬ。
- ③ PHIGSの応用プログラムは、データベース管理システムのような他

のプログラム・インタフェースも使う可能性が示るので、それらと矛盾してはならない、また GKS、CAI、CAM との一貫性も要求される(筆者注: CAI・CAM との一貫性は必ずしも必要とされるか?)、と示している。

次に"Character-set"というタイトルで、符号化とは別にも連動していゝ複数の文字集合の概念を導入し、各文字列に対して、文字集合指定が特性として与えられることを述べていゝ。この結果、文字列が現在の GKS での文字数と文字連鎖の2つ組でなく、文字集合指定、文字数、文字連鎖の3つ組として概念的には扱われる。ただし、実現方法としては3つ組と同等の情報は何らかの形で備わっていかねばいゝ。

"Range of support"の内容は、TEXT 主力基本要素だけでなく、文字列を扱うところのすべてで、各国語文字列を扱えるようにすべきであることと述べられている。

"Language binding consideration"では、これまでの文字列の概念的な2つ組が、個別の言語結合においては異なる文字連鎖だけでなく十分な情報が提供されたのと同様に、先に提案した概念的な3つ組も言語結合の上では違った形をとり得るとし、次のよう言語結合の別を示している。

- ① 言語上の文字列自身に文字集合指定と文字数の情報が含まれ、引数としては、1つで済む場合。
- ② 文字集合指定子と終端情報を含んだ文字列として引数とすべき場合。
- ③ 文字集合指定子と文字数と文字列として引数とすべき場合。
- ④ 言語結合特有の文字集合指定関数を用意し、個々の関数呼び出しでは、そのときの文字集合指定を参照すべき場合。

以上の4ケースが今後の言語結合決定の際の参考となることを示し、PHIGS本体部では、抽象的3つ組を用いることと提案している。

"Conclusion"では、これまでの記述を要約して、

- ① 文字列データは3つ組として考えられること
- ② 文字集合は、プログラミング言語依存であり、言語結合ドキュメント内で定義されること
- ③ 入力文字列の文字集合は、INITIALIZED時に指定されるものに依ること(上記の要約では、これについて述べていない)。

を挙げている。また、文字列属性と文字集合との関係、他のグラフィクス標準への応用について付未検討であるとしている。

ここで以上の提案の作成経緯について述べてみる。J-PHIGSで日本語サポートの話が始めたときは、各自が自分の主に使っている言語の機能の知識だけを語を述べようとしていた。3.1で示したように、プログラミング言語の諸形態のどれに対しても結合できる概念がPHIGS本体では必要とされ、それがそのまゝの形で言語結合と互に必要がないことが合意され、互に互に結合時同かかったが、最後の段階でも複数文字集合切替機能はPHIGS本体内で定義するのが良いという意見が残った。

筆者としても、最近になって3つ組の言語結合形態として、関数名中に文字集合指定情報を入れる可能性に気付くなど、上記の提案がまだいろいろと問題を残していると考えている。

(*) 複数文字型など考慮する必要がないという声もあるが、日本語FORTRANの普及および国際標準化提案の動き⁽¹⁷⁾と尊重するところから、

5. おわりに

日本語処理については、各分野内での実践が先行している。読者の中からは、貴重で示唆が、グラフィクス標準化と日本語機能の問題に対して寄せられることを期待する。

なお、この提案作成の後、情報処理の各分野で同様の問題があり、統一的な日本の態度について検討する必要性が確認され、情報処理学会規格委員会(今月 規格調査会に組織変更)内に日本語機能専門委員会を設置し、「グラフィクス、オペレーティング・システム、データベース、文書処理及び交換などにおける日本語機能の統一的取扱について検討し、これを国際標準に反映させる」ことが決定されている。

最後に一言、説明をしたい。本資料は本来、J-PHIGSの活動の柱である日本IBM東京基礎研究所の宇野栄氏、浦北和哉氏によってまとめられるべきものであったが、宇野氏が長期の木国在外研究に行かれたので、グラフィクス標準化への関わりが未だ浅い筆者がまとめることになってしまった。ここまでの執筆、乱文に、「こんな人間が日本語処理について論ずるのか」と嘆かれる方も多いと思うが、事の本質上、あえて拙速の手段をとったことを差し引いて考えて頂きたい。

参考文献

- (1) Graphics Kernel System (GKS) Functional Description, ISO IS 7942, July 1985
- (2) P.R.Bono: Guest Editor's Introduction, IEEE Computer Graphics and Applications, Vol.6, No.8, pp.12-16, Aug. 1986
- (3) G.S.Carson and E.McGinnis: The Reference Model for Computer Graphics, *ibid.*, pp.17-23
- (4) L.Henderson, M.Journey and C.Osland: The Computer Graphics Metafile, *ibid.*, pp.24-32
- (5) T.Powers, A.Frankel and D.Arnold: The Computer Graphics Virtual Interface, *ibid.*, pp.33-41
- (6) R.F.Puk, J.I.McConnell: GKS-3D:A Three-Dimensional Extension to the Graphics Kernel System, *ibid.*, pp.42-49
- (7) D.Shuey, D.Bailey and T.P.Morrissey: PHIGS: A Standard, Dynamic, Interactive Interface, *ibid.*, pp.50-57
(邦訳): 装置に独立でモジュール性を備えたグラフィクス・インタフェースPHIGS、日経エレクトロニクス、1986.10.20 (No.406)、pp.139-151
- (8) M.R.Sparks and J.R.Gallop: Language Bindings for Computer Graphics Standards, *ibid.*, pp.58-65
- (9) M.W.Skall: NBS's Role in Computer Graphics Standards, *ibid.*, pp.66-70
- (10) 和田英一: エディタとテキスト処理①文字セットとそのコード、*bit*、1982.4
- (11) 和田英一: エディタとテキスト処理①コードの拡張、*bit*、1982.5
- (12) 飯村二郎: 機種間インタフェース、*bit*、1985.4 (別冊 ワードプロセッサの日本語処理)
- (13) 木下恂: 標準プログラム言語における日本語処理、情報処理、Vol.26, No.3, pp.226-232, Mar.1985
- (14) 尹、高木、牛島: 日本語テキストにおけるパターンマッチング手法の比較と改善、情処研資、記号処理36-6/プログラミング言語5-6、Mar.1986
- (15) 永栄繁樹: 日本語処理機能を備えたUnixの標準化が進む、日経エレクトロニクス、1986.1.27 (No.387)
- (16) 西村恕彦: FORTRAN日本語処理のJIS原案、情処研資、記号処理36-6/プログラミング言語5-6、Mar.1986
- (17) 規格委員会: 1985年における規格委員会の活動、情報処理、Vol.27, No.7, pp.780-809, 1986.7
- (18) J.D.Becker: Multilingual Word Processing, Scientific American, Vol.251, No.1, pp.82-93, Jul.1984
森健一(訳): 多言語ワードプロセッサの開発、サイエンス、Vol.14, No.9, pp.34-46, Sep.1984