

PHIGS89概説(1)

佐藤, 周行
九州大学大型計算機センター研究開発部

<https://doi.org/10.15017/1470215>

出版情報 : 九州大学大型計算機センター広報. 25 (5), pp. 393-404, 1992-09-25. 九州大学大型計算機センター
バージョン :
権利関係 :

PHIGS89 概説 (I)

佐藤周行*

1 はじめに

1.1 PHIGS とは

PHIGS(the Programmer's Hierarchical Interactive Graphics System) はその名前の示す通り「階層的データ構造をもち、かつ対話性を備えたプログラム開発者のための図形処理システム」である。ISO の規格が 1989 年に定められ、GKS などとともに標準的な図形処理システムのひとつになっている。富士通 MSP の上でも F6247 グラフィックディスプレイを出力ワークステーションとして PHIGS システムが構築されている。

その他にも X Window System, Sunview Window System など様々なシステムの上に PHIGS が実現されている。

そのおおまかな特徴として次のことがあげられる。

- 2D,3D グラフィックスの実現。
- 図形データの作成 / 編集、図形出力、対話的入力などの豊富な機能の提供。
- 階層的な図形データ構造の提供。
- 座標変換ユーティリティの充実。

PHIGS グラフィックスシステムの応用分野は多岐にわたるが、特に建築、機械工学などにおける CAD, CAM などが考えられる。また、様々な角度から 3 次元分子、結晶をながめる必要のある化学関連分野も有力な応用分野になるだろう。一例として、本センターで提供している MENTAT システム [4] がある。センター提供の MENTAT は Sun-Phigs システムの上に構築されている。

PHIGS は具体的にサブルーチンパッケージとして実現されている。その対象となる言語として Fortran と C がある。

本センターでは 1992 年 4 月から MSP + F6247 ディスプレイ上での PHIGS システムを公開している [5]。本稿は主としてその解説を目的とするものである。

1.2 PHIGS の概念

まず、概念図を図 1 にしめす。

PHIGS において図形データは CSS(Centralized Structure Store) と呼ばれるデータベースで管理されており、図形の編集は CSS に対する操作という形で与えられる。CSS における図形情報の単位を「ストラクチャ」(structure) という。CSS における「モデル」(model) はストラクチャのネットワークという形で「定義」される。

平成 4 年 7 月 25 日 受理

*九州大学大型計算機センター研究開発部

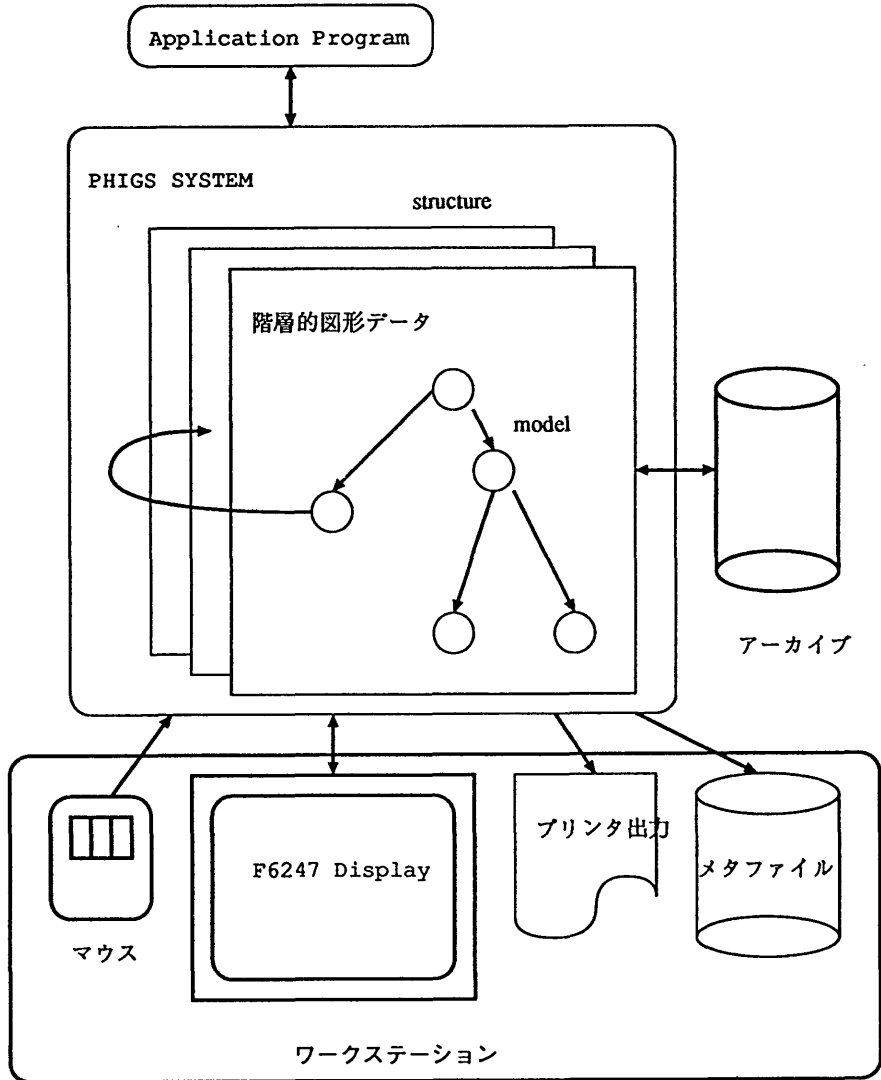


図 1: PHIGS の概念

図形データの表示 (画面、プリンタなど)、入力 (マウス、タブレットなど) に用いるデバイスを PHIGS ではまとめて **workstation** と呼ぶ¹。

さて、モデルのワークステーションへの表示を図形出力という。図形出力とはモデルの持つネットワークをたどっていき、表示するプロセス (traversal) である。

1.3 この連載の予定

今後の PHIGS の説明は以下のトピックスを簡単にたどっていく予定である。

1. 出力の基本要素
2. その属性
3. モデルの作成
4. モデルの表示
5. モデルの編集
6. 座標系について
7. メタファイルとアーカイブ

なお、今後数回にわたって連載する (予定である) が、PHIGS の包括的なマニュアルにするつもりはない。PHIGS を理解する上でのポイントを押えるのが連載の目標である。

具体的にプログラムを書くにはマニュアル [1, 2, 3] は必携である。この連載はマニュアルを読むための前提知識の解説と考えていただきたい。

今回は F6247 ディスプレイ で具体的に簡単な図形を出力するまでを解説する。次回は基本的な図形を組み合わせ一つ一つの図形を作成 / 編集することと、ディスプレイ以外のプリンタやデータセットへ図形を書き出すことを解説する予定である。それ以降は座標系の解説その他をする予定である。

2 センターで利用可能な PHIGS システム

現在センターでは MSP 上で PHIGS89 システムを公開している。これは Fortran のサブルーチンの形で与えられている。グラフィックス用の専用端末として センターオープン機器室内に F6427 システムを用意している。現在 PHIGS はこの専用ディスプレイ上でのみ利用可能である。

もうひとつ、SunView 上で動作する PHIGS-PLUS システムを公開している。これは C のサブルーチンの形で与えられている。グラフィックス用のデバイスとして センターオープン機器室内に Sun システム (qviss) を用意している。

PHIGS の説明は言語結合層までいかずに説明することも可能であるが、ユーザ層を考え、基本的に MSP 上のシステムを例にとって説明することとする。Sun 上での PHIGS も筆者に余裕があれば 解説を試みるが、利用者は、各自マニュアルを見て独習する覚悟をされたい。

3 PHIGS のアウトライン

PHIGS 自身はかなり抽象的な機能で定義、説明されている。それを特定の言語にライブラリなどとして実装する時に、機能と (ターゲットとなる言語の) ライブラリ関数のインターフェイスを別に定める (これも規格の一部!!)。このインターフェイスを言語結合層 (language binding) と言う。

以下、MSP で提供されている Fortran とのインターフェイスを用いてプログラムを記述し、機能面での抽象的な説明は必要に応じてコメントとしてプログラムにはさみこむことにする。

¹これは PHIGS の用語である。通常用語と混同しないこと

3.1 PHIGS プログラムの骨格 (I)

3.1.1 PHIGS の開始と終了

PHIGS を開始する時には必ず PHIGS を「開始」し、終了する時は PHIGS を「終了」する必要がある。PHIGS の開始に対応するサブルーチンは POPPH(Phigs OPen PHigs)、終了に対応するそれは PCLPH(Phigs CLose PHigs) である。

popph(errfile, bfa) (integer errfile,bfa)

PHIGS を「開く。」 errfile はエラーのログファイルを指定する。以下では errfile を 99 に固定する (phigs89 コマンドの環境設定に従えば、これでエラーは ERRFILE に書き出される)。
bfa には適当な値を指定する (現在は特に意味のないパラメタである)。

pclph PHIGS を「閉じる。」

例 1 (PHIGS の骨格 (I))

```
C Open PHigs
    call popph(99, 0)
C Phigs session
    .
    .
    .
C Close Phigs
    call pclph
    stop
    end
```

3.1.2 ワークステーションのオープンとクローズ

PHIGS のセッションは入出力デバイスであるワークステーションを開くことから始まる。ワークステーションは PHIGS の終了前にクローズすること。対応するサブルーチンは POPWK (Phigs OPen WorKstation) と PCLWK(Phigs CLose WorKstation) である。

popwk(wkid, conid, wtype) (integer wkid,conid,wtype)

ワークステーション wkid を開く。wkid はワークステーション ID としてプログラムで指定する。conid で wtype ワークステーション wkid の種類を指定する。ここでは conid は 1 に、wtype は 1 に固定する (phigs89 コマンドの環境設定に従えば、この指定で F6247 ディスプレイが使用可能になる)。

pclwk(wkid) ワークステーション wkid を閉じる。

例 2 (PHIGS の骨格 (II))

```
C Open PHigs
    call popph(99, 0)
C Phigs session
    call popwk(0,1,1)
    .
    .
    .
    call pclwk(0)
```

```
C Close Phigs
  call pclph
  stop
  end
```

さて、残りは PHIGS のセッション中で具体的に仕事をする方法を覚えることである。

3.2 MSP での PHIGS の実行 (I) – F6247 編

センターでは PHIGS の実行環境のインターフェイスを phigs89 コマンドとして提供している [5]。

3.2.1 PHIGS89

現在、F6247 ディスプレイ上で PHIGS システムが利用可能である。F6247 利用を想定した環境設定を自動的にしてくれる。

例 3 (使用例)

```
READY
phigs89 phigs.fort(eg1)
.
.
.
```

3.2.2 F6247 ディスプレイ使用上の注意

F6247 ディスプレイは PHIGS 専用端末である。2 画面を持ち、一つがテキスト画面、もうひとつが PHIGS グラフィックス画面である。画面の切替えは キーボード左端にある

JUMP SCREEN

 キーで行なう (トグルになっている)。テキスト画面は現在日本語表示ができない。

このセクションを閉じる前に完全なプログラムを一つ例としてあげておこう。読者はこれから PHIGS プログラムの「基本形」を読みとられたい。

例 4 (完全なプログラムの例) ごく単純なプログラムである。F6247 ディスプレイに長方形 (実は正方形を斜めから見たもの) が一瞬出てきて終了する。

```
      program rect
C
      integer strct, lntype
      real px(5),py(5),pz(5)
      real wsc
C
C data value
      data strct/1/,lntype/1/
      data px/0.25,0.75,0.75,0.25,0.25/
      data py/0.50,0.50,0.20,0.20,0.50/
      data pz/0.20,0.20,0.60,0.60,0.20/
      data wsc/3.0/
C
C open phigs
      call popph(99,0)
```

```

C open F6247
    call popwk(0,1,1)
C set polyline-data in CCS
    call popst(strct)
    call psplci(1)
    call psln(lntype)
    call pslwsc(wsc)
    call ppl3(5,px,py,pz)
    call pclst
C post the structre
    call ppost(0,strct,0.0)
C epilogue
    call pclwk(0)
    call pclph
    stop
    end

```

4 図形出力

4.1 CSS

PHIGS では図形データを管理するデータベースを CSS(Centralized Structure Store)と呼ぶ。図形の編集 / 出力は必ず CSS を対象に行なわれなければならない。

popst(st) (integer st) CSS st を開く。CSS の識別子(整数)はプログラム側で与える。

pclst 現在開いている CSS を閉じる。

なお、編集の対象になる CSS は同時に2つ以上開くことはできない。

CSS を表示するには CSS を 出力ワークステーションに「ポスト」しなければならない。

ppost(wk,st,pr) (integer wk,st,real pr)

出力ワークステーション wk に CSS st を優先度 pr でポスト(表示)する。

最終的な PHIGS プログラムの骨格は以下のようなことになることがわかるであろう。

例 5 (PHIGS の骨格 (III))

```

C Open PHigs
    call popph(99, 0)
C Phigs session    wk integer
    call popwk(wk,1,1)
C Open CSS        st integer
    call popst(st)
C Edit CSS
    .
    .
    .
C Close CSS
    call pclst

```

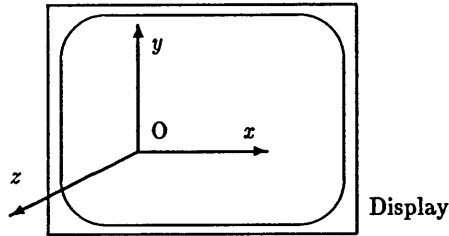


図 2: 右手系デカルト座標

```

C Post CSS to a Workstation
  call ppost(wk,st,0.0)
C Close Workstation
  call pclwk(wk)
C Close Phigs
  call pclph
  stop
  end

```

4.2 出力基本要素

で、図形データを構成する基本要素である。規格では以下のものを基本要素と定める。

Polyline 折れ線

Polymarker マーカ列

Fill Area 領域

Fill Area Set 領域セット

Text 文字列

Annotation Text Relative 注釈文字列

Cell Array セル配列

Generalized Drawing Primitive(GDP) その他、実装に依存する要素。

MSP でサービスしている PHIGS89 では GDP として 1. 3次元折れ線セット, 2. 2次元折れ線セット, 3. 円, 4. 円弧を提供している。

4.2.1 座標系について

以後、各要素について解説するが、その前に座標系について一言。PHIGS において座標は右手系 3次元デカルト座標(図 2参照)を用いる。

PHIGS は 3次元グラフィックスシステムであるからビューに関連する座標系はかなりの複雑度を持ったものになっている。が、しばらくは気にすることはない(範囲で話をすすめる)。

5 各要素の説明

5.1 Polyline

点を $n(n \geq 2)$ 個指定して、それらを結ぶ $(n - 1)$ 個の折れ線を定義する。折れ線は 2 次元または 3 次元で指定する。

PPL3(Phigs PolyLine 3) 点を N 個指定して、それらを結ぶ $N - 1$ 個の折れ線を定義する。

```
PPL3(N, PX, PY, PZ)
INTEGER N
REAL    PX(N),PY(N),PZ(N)
```

点は配列 PX、PY、PZ でそれぞれ x, y, z 座標を指定する。

PPL(Phigs PolyLine) 上の 2 次元版

プログラムの具体例は 例 4 参照。

5.2 Polymarker

点を $n(n \geq 1)$ 個指定して、その点にマーカを置く。結果としてマーカの列ができる。2 次元または 3 次元で指定する。

PPM3(Phigs PolyMarker 3) 点を N 個指定して、それらの上にマーカを置く。

```
PPM3(N, PX, PY, PZ)
INTEGER N
REAL    PX(N),PY(N),PZ(N)
```

点は配列 PX、PY、PZ でそれぞれ x, y, z 座標を指定する。

PPM(Phigs PolyMarker) 上の 2 次元版

属性について

Polyline ではふむふむ、と納得したつもりでここまで来た読者もいるかと思うが(そうでなかったらごめんなさい)、さすがに Polymarker ではつまったはずである(といいなあ)。つまり、「マーカ」とは何か?、どんな形なのか? 1 種類しかないのか? 等々。

実は PHIGS では基本要素でその位置 (= 幾何的情報) を指定し、その他に幾何的または非幾何的な属性を指定することでグラフィックス出力が得られることになっている。

例えば、PolyLine の属性として以下のものが定義されている。

属性	機能	設定サブルーチン
line type	線種	PSLN
linewidth scale factor	線幅	PSLWSC
colour index	線色	PSPLCI

PolyMarker の属性として以下のものが定義されている。

属性	機能	設定サブルーチン
marker type	マーカ種	PSMK
marker size scale factor	倍率	PSMKSC
colour index	マーカ色	PSPMCI

設定に当たっての引数、引数の意味その他についてはマニュアルを参照すること。

例としてマーカ種をいろいろ設定して表示するプログラムをあげる。読者は F6247 ディスプレイの前に座って何が出てくるかを確認されたい。

```

    program marker
C
    integer str
    real px(5),py(5),pz(5)
C data value
    data str/0/
    data px/0.50,0.60,0.70,0.80,0.90/
    data py/0.40,0.50,0.60,0.70,0.80/,pz/5*0.50/
C prologue
    call popph(99,0)
    call popwk(0,1,1)
C open structure
    call popst(str)
C set marker of various shape
    call psmksc(5.0)
    do 200 i=1,5
    call psmk(i)
    call ppm3(1,px(i),py(i),pz(i))
200 continue
    call pclst
C post the structure
    call ppost(0,str,0.0)
C read some input from a keyboard -- synchronization
    read(5,*) a
C epilogue
    call pclwk(0)
    call pclph
    stop
end

```

5.3 Fill Area

さて、話を戻す。次は領域である。

PFA3(Phigs FillArea 3) 点を N 個指定して、それらを結ぶ折れ線を境界とする平面領域を定義する (平面を形成するように点を指定するのはプログラム側の責任である)。

```

PFA3(N, PX, PY, PZ)
INTEGER N
REAL    PX(N),PY(N),PZ(N)

```

点は配列 PX、PY、PZ でそれぞれ x, y, z 座標を指定する。

PFAS(Phigs FillArea) 上の 2 次元版。

5.4 Fill Area Set

領域セット。平面領域を有限個集め、それらの成す 3 次元 / 2 次元の点集合のことを言う。

PFAS3(Phigs FillArea Set 3) 領域を *NPL* 個指定してそれらからなる領域セットを定義する。

```
PFAS3(NPL,IX,PX,PY,PZ)
INTEGER NPL
INTEGER IX(NPL)
REAL PX(*),PY(*),PZ(*)
```

ここで i 番目の領域の x 座標は $PX(IX(i-1)+1)$ から $PX(IX(i))$ までで定義される²。 y 座標と z 座標についても同様である。

PFAS 上記の 2 次元版。

5.5 Text

文字列。PHIGS での文字列の扱いは複雑を極める。基本的な幾何情報の指定は単純であるが、非幾何的な属性が 11、幾何的な属性が 6 種類あり、それぞれの組合せにより多様な図形が出力できる。文字列の属性について語りはじめると紙数がたりないのでばっさりと省略する。

PTX3(Phigs TeXt 3) 文字列、文字列表示の基準点(通常左端)(text point)と、方向ベクトル(direction vector)を指定して、3 次元空間内に文字列を指定する。その具体的な意味は図 3 参照。

```
PTX3(PX,PY,PZ,TDX,TDY,TDZ,CHARS)
REAL PX,PY,PZ
REAL TDX(2),TDY(2),TDZ(2)
CHARAGER*(*) CHARS
```

PX,PY,PZ で表示基準点(text point)を、TDX(1),TDY(1),TDZ(1) と TDX(2),TDY(2),TDZ(2) で方向ベクトル(direction vector)を、CHARS で文字列を指定する。

PTX(Phigs TeXt) 上記の 2 次元版。

5.6 Annotation Text Relative

注釈文字列を指定する。図形を変換すると、文字列はその変換に従う(ので裏返ったりすることがおきる)が、注釈文字列はその変換に従わず、「正しい」表示が得られる。

PATR3(Phigs Annotation Text Relative 3)

```
PATR3(RPX,RPY,RPZ,APX,APY,APZ,CHARS)
REAL RPX,RPY,RPZ
REAL APX,APY,APZ
CHARACTER*(*) CHARS
```

²1 番目のみ PX(0) から PX(IX(1)) まで

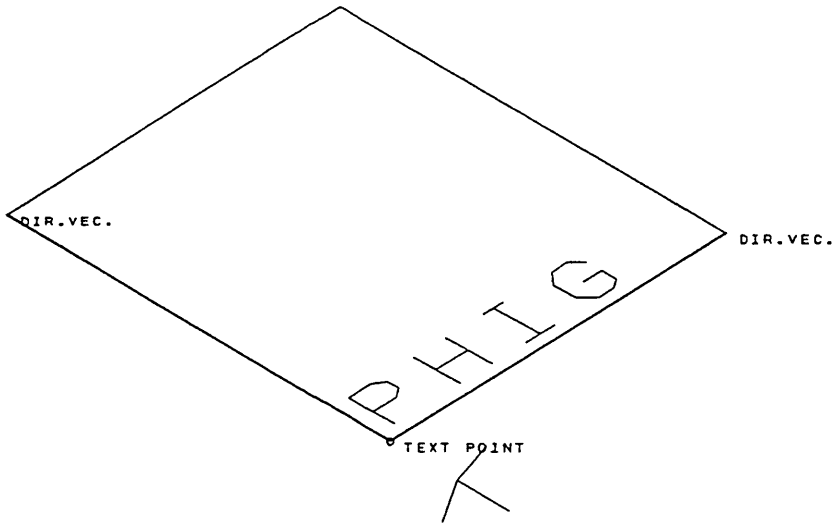


図 3: PTX3 の引数の意味

注釈文字列の参照点を RPX,RPY,RPZ で、注釈文字列位置の基準点を (参照点からの相対位置で) APX,APY,APZ で、注釈文字列自体を CHARS で指定する。

PATR(Phigs Annotation Text Relative) 上の 2次元版。

次のプログラムは図 3 のソースの一部である。「文字列」と「注釈文字列」が座標変換の影響をどう受けるのかがわかるだろう (プログラム中 不完全な部分は次回以降説明する)。

例 6 (図 3 のソースの一部) サブルーチン pfa3, ptx3, patr3 が新たに出現している。

```

program ptxsmp1
C
  parameter (PI=3.14159265)
  integer strct
  real pxa(4),pya(4),pza(4),pdx(2),pdy(2),pdtz(2),pinv(2),pvar(2)
  real xfrmt1(4,4),xfrmt2(4,4),xfrmt3(4,4)
  integer errind
C data value
  data strct/1/
  data pxa/-0.3,0.1,0.2,-0.2/,pya/0.2,0.1,0.5,0.6/
  data pza/0.2,0.1,0.1,0.2/
  data pdy/4.0,-1.0/,pdx/1.0,-4.0/,pdtz/0.0,1.0/
  data pinv/0.15,0.15/,pvar/0.15,0.20/
C prologue
  call popph(99,0)
  call popwk(0,1,1)
  call popst(strct)
C definition of axis
  call ppl3(2,pvar,pinv,pinv)
  call ppl3(2,pinv,pvar,pinv)
  call ppl3(2,pinv,pinv,pvar)

```

```

C set the plane
    call psisi(-20)
    call pfa3(4,pxa,pya,pza)
C set the string 'PHIG'
    call pschh(0.08)
    call ptx3(0.1,0.1,0.1,pdtx,pdty,pdtz,'PHIG')
C set the circle at the text point
    call psmk(4)
    call ppm3(1, pxa(2),pya(2),pza(2))
C set annotation text
    call patr3(0.1,0.1,0.1,0.01,-0.01,0.01,'TEXT POINT')
    call patr3(0.2,0.5,0.1,0.01,-0.01,0.01,'DIR:VEC.')
    call patr3(-0.3,0.2,0.2,0.01,-0.01,0.01,'DIR:VEC.')
C set direction vectors
    call pslwsc(3.0)
    call ppl3(3,pxa,pya,pza)
    call pclst(strct)
C
C
C   この部分、座標変換
C
C
C   call ppost(0,strct,0.0)
C epilogue
    call pclwk(0)
    call pclph
    stop
    end

```

5.7 Cell Array

セル配列。詳細は省略する。

5.8 Generalized Drawing Primitive(GDP)

システムの提供する、PHIGSの規格外の出力要素。MSPのシステムでは折れ線セットと円、円弧が提供されている。詳細は省略。

終盤になってだいぶはしょってしまった。具体例は次回以降にまわすことにしよう。

参考文献

- [1] OS IV/PHIGS89 解説書,70SG-6401-1, 富士通.
- [2] OS IV PHIGS89 文法書 FORTRAN77 編 V10L11 用, 70SP-6411-1, 富士通.
- [3] OS IV/MSP PHIGS89 使用手引書 V10L11 用,79SP-4430-1, 富士通.
- [4] 「MARC/MENTAT 利用の手引」, 山成實, 九州大学大型計算機センター広報, Vol.25, 1992, pp. 1-31.
- [5] 「PHIGS89 の公開について」, 九州大学大型計算機センターニュース, No. 459, 1992(also 九州大学大型計算機センター広報 Vol. 25, No. 3, pp. 233-234, 1992).