

目次

第 6 章	ドローイング : Tgif	121
6.1	Tgif の起動	122
6.2	機能の呼出し : メニュー	123
6.2.1	モードメニュー	126
6.2.2	線の属性 : Dash, Type, Style, Width	127
6.2.3	ペン種と塗りつぶし : Pen, Fill	128
6.2.4	テキスト : Text	129
6.2.5	配置 : Arrange	130
6.2.6	レイアウト : Layout	133
6.2.7	ファイル : File	136
6.2.8	編集 : Edit	138
6.3	日本語の入力	139
6.4	キー入力コマンド	140
6.5	他のツールとの連携	141
6.5.1	xpm 形式で画面のダンプ : xgrab	141
6.5.2	Gnuplot からの obj 形式出力	143
6.5.3	PS 形式から obj 形式への変換 : pstoeedit	144
6.5.4	写実的に見せる工夫	145
6.6	テクニックの上達	146
6.6.1	基本要素を描く	146
6.6.2	四角や円などの単純な組合せ	147
6.6.3	Fill-White の応用	149
6.6.4	折れ線で描けるもの	149
6.6.5	ベジエ曲線の応用	151
6.6.6	変形	152
6.7	図面集	154

6

ドローイング : Tgif

実験装置などの図面も論文には必要です。図面は大きくペイント系（ビットマップとして一点ごとの色で保存）と，ドロー系（オブジェクトとして形と位置を保存）に分かれます。ペイント系では任意の絵が描けますが，拡大縮小に耐えません。一方，ドロー系では出力先の大きさに合わせて綺麗に拡大縮小が可能ですから，あまり複雑でない装置の概念図などはドロー系のツールで書くのが得策でしょう。

ここでは William Chia-Wei Chang (william@cs.ucla.edu) 氏によって開発・保守されている tgif の使い方を詳しく紹介します。操作にやや不統一な点がありますが，図を綺麗に描くという目的をもって基本的な図の作成方法を学ぶには十分な機能があります。また，バージョン 3.0 では，オブジェクトの任意回転や pixmap の加工が行えるようになりました。それ以外にも非常に多くの機能を備えており，例えばオブジェクトをクリックすると外部コマンドが起動される機能を用いればオーサリングツールとしても使えますし，URL にも対応していて，他のホストにある Tgif ファイルを閲覧する機能もあります。このように，総合的なグラフィックツールを目指して開発が精力的に行われています。



図 6.1: 配布ファイルに含まれている Tgif のアイコン

6.1 Tgif の起動

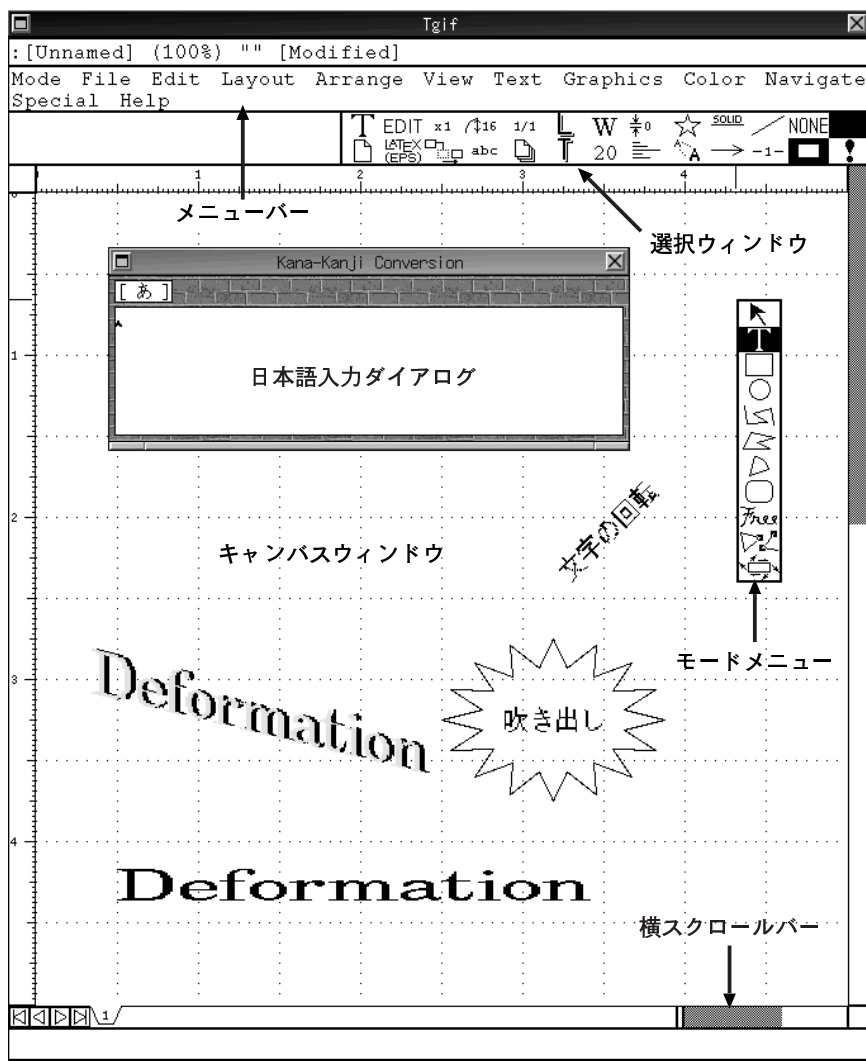


図 6.2: tgif の実行画面

図 6.2 は tgif を起動して色々なオブジェクトを描いたものです .

tgif &

画面右上に選択ウィンドウがあります . 図形や機能を選択するには , 選択ウィンドウの上段左端のアイコンにカーソルを合わせて , マウスの左右ボタンどちらかを押します . するとアイコンが変化して , 図形や機能が選択されます . 図では "T" が現れていますが , これは文字列が選択されていることとなります . このメニューはモードメニューと呼ばれます . キャンバス上でマウスの右ボタンを押すと , 上記のモードが一覧になったメニューが現れます . 図のキャンバス右上にあるのがそれで , マウスドラッグによりその中から選択することができます .

6.2 機能の呼出し : メニュー

tgif では , 以下に示すように機能の選択を複数の方法で行うことが可能です .

- 選択ウィンドウのアイコン (=機能) を変化させる
- 選択ウィンドウでメニュー (機能の一覧) を呼び出す
- キャンバスでメニューを呼び出す
- キャンバスでキー入力により機能を直接指定する

これはユーザーの習熟度に合わせて最適な機能選択が行えるように配慮されたものですが , 逆に操作が複雑であるかの印象を与えます . ここでは , 選択ウィンドウのアイコンをマウスの中ボタンでクリックしてメニューを呼び出す方法を中心に説明します . まず , 呼び出されるメニューを図 6.3 と図 6.4 に示しておきます . 個々のメニューを見ながら , 図の描き方について説明していきます .

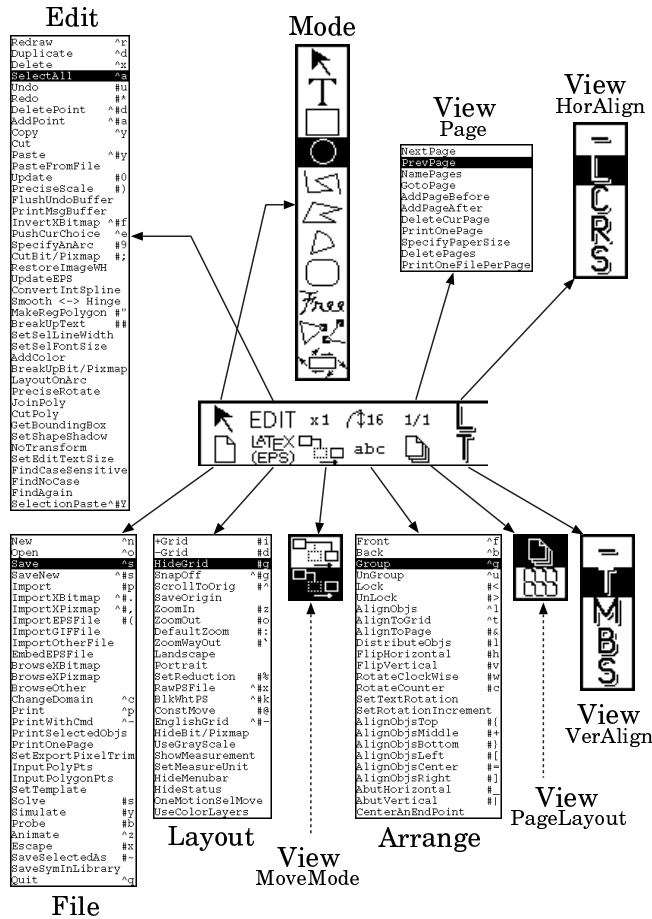


図 6.3: 選択ウィンドウのアイコンを中ボタンでクリックして現れるメニュー (左半分)

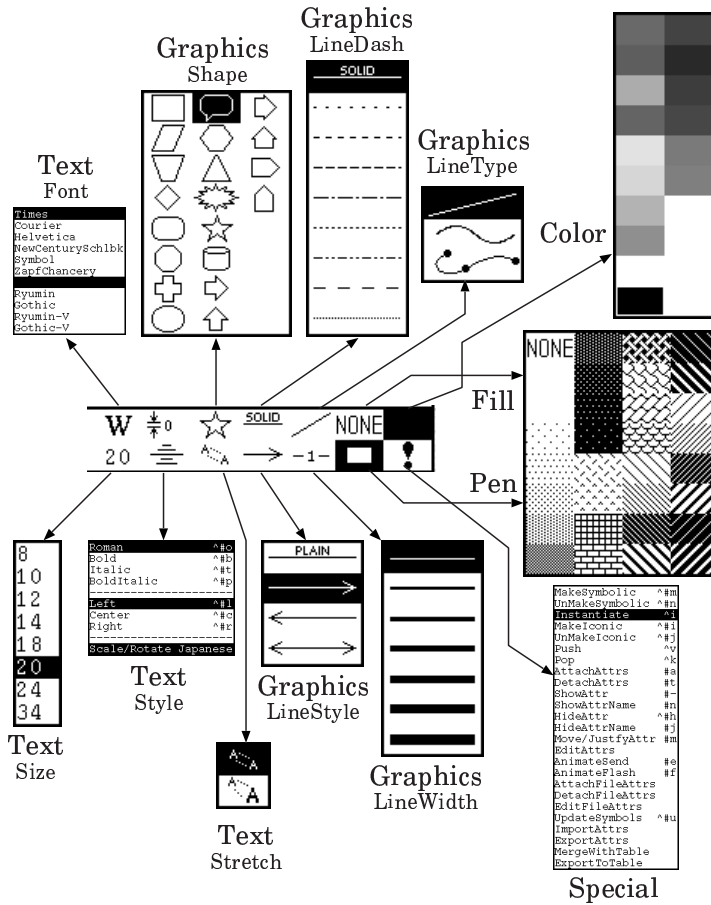


図 6.4: 選択ウィンドウのアイコンを中ボタンでクリックして現れるメニュー (右半分)

6.2.1 モードメニュー

最初は、最も頻繁に使うモードです。アイコンを見るとおおよそ想像できますが、描く図形を選びます（図 6.5）。ただ、一番上の”矢印”は矢印を描くのではなく、図形選択：移動や複写や拡大縮小の対象として選択状態にすることの機能呼び出すものです。また、一番下から2番目は、ちょっとわかりにくいですが、頂点選択：整列や削除の対象として折れ線および多角形の頂点を選択するためのモードです。他は、見たとおり、上から順に文字入力・長方形・楕円・折れ線・多角形（折れ線で閉じられた図形）・扇形・丸四角・フリーハンド・ずりと回転です。



図 6.5: モードメニューの各項目

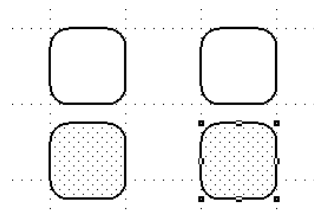


図 6.6: 図形選択：選択されると右図下の丸四角のようにマークが付けられる。

図の描き方ですが、例えば四角の場合は対角の2点（始点と終点）を指定します。始点位置で左ボタンを押してマウスカーソルをドラッグすると、カーソル形状が指に変化し、始点位置と現在のカーソル位置を対角とした四角の外形がラバーラインでトレースされますから、自分の望む大きさになったら、左ボタンを離して確定します。簡単ですね。楕円と丸四角（図 6.6）とフリーハンドが四角と同じように、左ボタンのみの操作（押してドラッグ離して確定）で描けます。文字も、描きたい位置（左端）を指定するだけなので、左ボタンのみの操作です。

線分を描くときには、右ボタンも使います。左ボタンを押して線分の端（折点）を指定し、右ボタンで終了です。したがって、ドラッグ（ボタンを押し下げたままカーソルを移動させる）は意味を持ちません。多角形では、最初の頂点と最後の頂点を結んで図形が完成します（すなわち、閉じるつもりで、最初の頂点にカーソルを重ねてはいけません）。

ずりと回転

バージョン 3.0 からは、図形のずりと回転がモードメニューから選択できるようになりました。図形を選択してから、ツールバーのモードメニューを中ボタンでクリックしてずりを呼び出してください。すると、カーソルが十字形に変化します。長方形にずりと回転を実行している様子を図 6.7 に示します。辺の中点を左ボタンでクリックしてドラッグすると斜めにずれて変形します。一方、頂点をドラッグすると回転します。

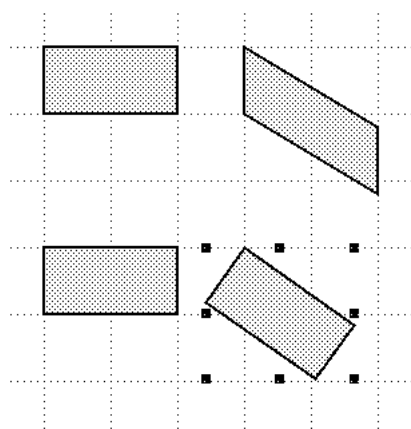


図 6.7: 長方形のずりと回転：上図は辺の中点をドラッグして斜めにずらした場合。下図は頂点をドラッグして回転させた場合を示す。

6.2.2 線の属性 : Dash, Type, Style, Width

図形を描く線の種類（実線，破線，一点鎖線など）・タイプ（直線，ベジエ曲線，スプライン曲線）・両端形状（矢印）・太さを変えることができます（図 6.8）。これらは、選択ウィンドウの右から 3,4 番目にあり，Graphics メニューに属しています。デフォルトは実線（Solid），細線（Thin），直線，終端矢印となっています。

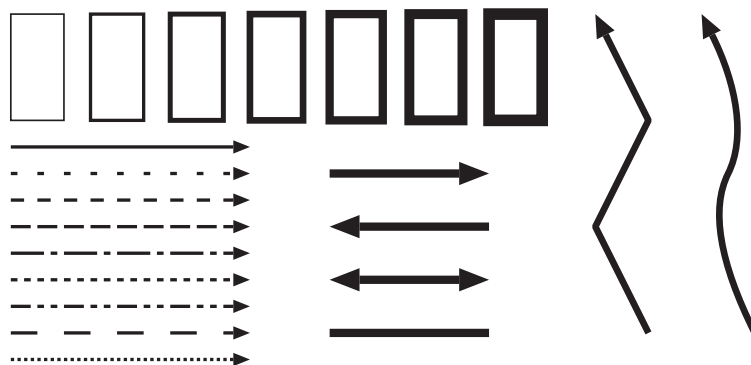


図 6.8: 線の色々な属性

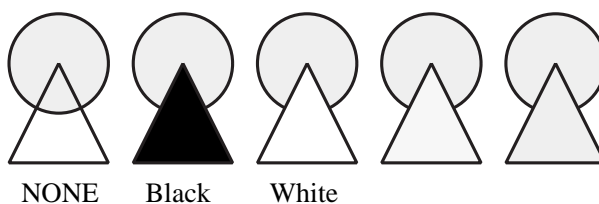


図 6.9: 色々な塗りつぶし（ペン種は黒）

6.2.3 ペン種と塗りつぶし：Pen, Fill

選択ウィンドウの右から 2 番目にあるのが塗りつぶし（上段：Fill）とペン種（下段：Pen）の指定ですが、まったく同じメニューが現れます。すなわち、指定できる属性が同じということです。図形の外形を描くのに使う線の属性が Pen で、囲まれた図形内部の塗りつぶし属性が Fill です。デフォルトでは、Pen は黒（枠）ですからわかるのですが、Fill は None となっています。None とは何も色を塗らない=透明ということです。したがって、図形を重ねた場合、下の図形が透けて見えることになります。図 6.9 に色々な塗りつぶしを示しました。


10pt Times	Helvetica	αβγδεφ	Times-Bold
12pt Times	Helvetica	αβγδεφ	Times-Roman
14pt Times	Helvetica	αβγδεφ	<i>Times-Italic</i>
18pt Times	Helvetica	αβγδεφ	<i>Times-BoldItalic</i>
24pt Times	Helvetica	αβγδεφ	

図 6.10: 色々な英数フォント

24pt 日本語フォント: Ryumin
 24pt 日本語フォント: Gothic
 24pt □長罫ハキハム .. Ryumin-V
 24pt □長罫ハキハム .. Gothic-V

図 6.11: 色々な日本語フォント

6.2.4 テキスト : Text

選択ウィンドウの左から 7,8 番目および 9 番目下はテキストに関する設定で、表示する際のフォント (Font)・大きさ (TextSize)・スタイル (TextStyle)・行間 (TextVspace)・伸縮可能 (StretchText) を指定します (図 6.10, 6.11)。スタイルには書体 (太字やイタリック) の他にそろえ (左, 中央, 右) が含まれており、それがアイコンとなっているため戸惑います。文字の伸縮はバージョン 3.0 から盛り込まれたものです。アイコンが  となっている場合に ON です。ビットマップフォントを元に伸縮しますから、画面上の仕上りはあまり美しくはないのですが、PS プリンターであれば出力は綺麗です。

設定は簡単なのですが、あまり色々な種類のテキストを使うと煩わしいので、按配を考えて使うとよいでしょう。また、実際に印刷される文字はプリンターに依存しており X 上の画面出力とは異なっている場合もありますから、プリンターの印字出力結果をよく確かめるよう心掛けてください。英数フォントは、Times, Courier, Helvetica, NewCentury, Symbol から選べますが、プリンターによっては印字できないものもあるようです。

6.2.5 配置：Arrange

Front	^f
Back	^b
Group	^g
UnGroup	^u
Lock	#<
UnLock	#>
AlignObjs	^l
AlignToGrid	^t
AlignToPage	#&
DistributeObjs	#l
FlipHorizontal	#h
FlipVertical	#v
RotateClockWise	#w
RotateCounter	#c
SetTextRotation	
SetRotationIncrement	
AlignObjsTop	#{
AlignObjsMiddle	#+
AlignObjsBottom	#}
AlignObjsLeft	#[
AlignObjsCenter	#=
AlignObjsRight	#]
AbutHorizontal	#_
AbutVertical	#
CenterAnEndPoint	

図 6.12: アレンジメニュー

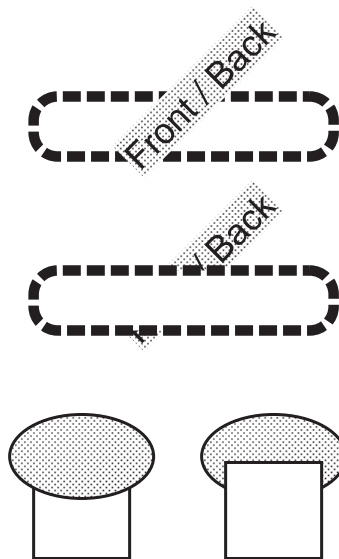


図 6.13: 前後関係を変える。

図面が正確であることはもちろん、見た感じを美しく仕上げるためには個々オブジェクトを描いた後に配置を調整する必要があります。そのためのメニューが用意されています。図 6.12 がそれです。沢山あって全部は説明できませんから、頻繁に使う主なものについて簡単に説明します。

フロント/バック

オブジェクトは時間的に後で描いたものが図面上は前面にきます。図面上でのオブジェクトの前後関係を変化させる場合にこれらの命令を使います。図 6.13 に前後関係を変化させた例を示します。

グループとアングループ

オブジェクトをまとめて一つのオブジェクトにする機能がグループです。図 6.14 に実行例を示します。グループ化すると全体をまとめて移動できますし、拡大・縮小も可能となります。むろん線分の属性変更などはグループ全体に及びます。したがって、グループ内のある要素を変更したい場合は、グループを解除してその要素のみを選択しなければなりません。グループ化のためには複数のオブジェクトを選択状態にできなければなりません。それには、**[Shift]** キーを押し続けながら左ボタンをクリックして次々に選択状態にします。

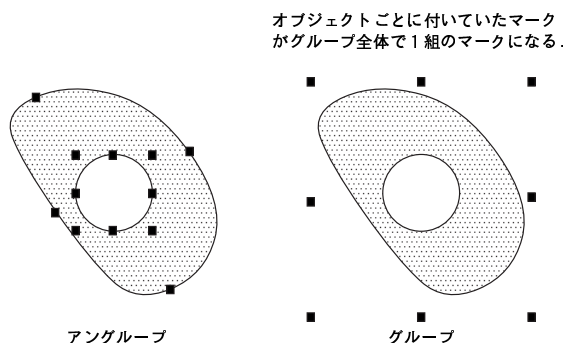


図 6.14: グループ化の実行

整列 : Align

オブジェクトを整列させます (図 6.15)。鉛直方向の位置関係は上端 (Top)、下端 (Bottom)、中央 (Middle) のいずれかでそろえるように指定します。水平方向は左端 (Left)、中央 (Center)、右端 (Right) の中から指定します。単に `AlignObjs` を行うと `View/HorAlign` と `View/VertAlign` の設定に従って整列します。既定は `Left` と `Top` になっています。

反転 : Flip

図 6.16 に示すように、水平方向 (`FlipHorizontal`) あるいは鉛直方向 (`FlipVertical`) にオブジェクトを反転します。




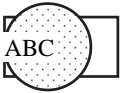





	Left	Center	Right
Top			
Middle			
Bottom			

図 6.15: 整列の様子：垂直方向 3 通り，水平方向 3 通りの組合せがある．

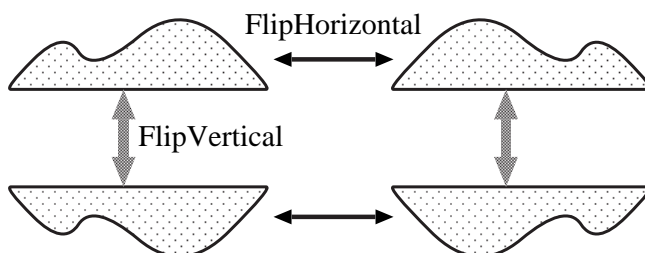


図 6.16: 水平方向，鉛直方向への反転

回転：Rotate

オブジェクトを時計回り (RotateClockwise) あるいは反時計回り (RotateCounter) に回転させます．バージョン 3.0 からは 1 回のステップで 45 度回転するよう設定されています．もちろんこの値は `SetRotationIncrement` を呼び出して変更することができます (バージョン 2.* では 90 度の整数倍の回

転しかできなかったので、大幅な機能強化といえます)。また、このように段階的に変化させるのではなく、値を直接指定して一度に回転させる方法として、Edit メニューの PreciseRotate があります。値入力ダイアログが表示され(図 6.17)、正負どちらの値も受け付けますが、時計回りを正に取っているので、ちょっと戸惑います。

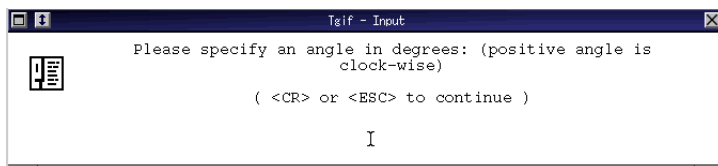


図 6.17: 任意回転 PreciseRotation の値入力ダイアログ

6.2.6 レイアウト : Layout

配置メニューとの区別がはっきりしませんが、配置がオブジェクト同士の位置関係の調整であるのに対し、レイアウトは頁全体に対する位置関係などを調節するものようです。

グリッド・グリッドスナップ

図形を整然と並べるためには、まったく任意の位置におけるよりは、ある決まった位置にしか置けないようにすべきです。そのために正方のグリッドが用意されており、グリッド位置にしか置けないモード：グリッドスナップ ON 時には、図形(枠の左上隅)はグリッドの位置にのみ置かれます。

グリッドの細かさは設定できますが、長さの単位に、英米式の inch を用いる場合 (EnglishGrid) と (仏式の?) cm を用いる場合 (MetricGrid) とでは、目盛の刻みが違います。EnglishGrid では 1/2, 1/4, 1/8, 1/16, 1/32 inch の 5 段階に設定できるのに対し、MetricGrid では、10, 5, 2, 1 mm の 4 段階しか設定できません。すると MetricGrid では細かい作図ができないと思われるかもしれませんが、グリッドスナップ OFF 時には 1/128inch あるいは 0.2mm になり、ほぼ等しい細かさです。図 6.18 は、馴染みの薄い英米式のグリッドで、左から 1/8 (デフォルト)、1/16, 1/32 inch の場合のルーラーの

様子を示します。

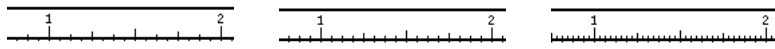



図 6.18: ルーラーはグリッド単位大きさを常に表示している

ズーム：Zoom

選択ウィンドウの上段左から 3 番目にある $\times 1$ は拡大率を表示しています。左ボタンクリックでズームイン (ZoomIn) して表示拡大率を大きくすると細かい作業ができます。また、逆に右ボタンクリックでズームアウト (ZoomOut) して拡大率を小さくすれば全体が見わたせるようになります。うまく使い分けて作業の効率をあげましょう。バージョン 3.0 からは、ズームインを実行するとマウスのカーソルが虫眼鏡  に変化し、拡大の中心位置を指定できるようになりました。

New	^n
Open	^o
Save	^s
SaveNew	^#s
Import	#p
ImportXBitmap	^#.
ImportXPixmap	^#,
ImportEPSFile	#(
ImportGIFFile	
ImportOtherFile	
EmbedEPSFile	
BrowseXBitmap	
BrowseXPixmap	
BrowseOther	
ChangeDomain	^c
Print	^p
PrintWithCmd	^~
PrintSelectedObjs	
PrintOnePage	
SetExportPixelTrim	
InputPolyPts	
InputPolygonPts	
SetTemplate	
Solve	#s
Simulate	#y
Probe	#b
Animate	^z
Escape	#x
SaveSelectedAs	#~
SaveSymInLibrary	
Quit	^q

Redraw	^r
Duplicate	^d
Delete	^x
SelectAll	^a
Undo	#u
Redo	#*
DeletePoint	^#d
AddPoint	^#a
Copy	^y
Cut	
Paste	^#y
PasteFromFile	
Update	#0
PreciseScale	#)
FlushUndoBuffer	
PrintMsgBuffer	
InvertXBitmap	^#f
PushCurChoice	^e
SpecifyAnArc	#9
CutBit/Pixmap	#;
RestoreImageWH	
UpdateEPS	
ConvertIntSpline	
Smooth <-> Hinge	
MakeRegPolygon	#"
BreakUpText	##
SetSelLineWidth	
SetSelFontSize	
AddColor	
BreakUpBit/Pixmap	
LayoutOnArc	
PreciseRotate	
JoinPoly	
CutPoly	
GetBoundingBox	
SetShapeShadow	
NoTransform	
SetEditTextSize	
FindCaseSensitive	
FindNoCase	
FindAgain	
SelectionPaste	^#Y

図 6.19: ファイルメニュー (左図) と編集メニュー (右図)

6.2.7 ファイル：File

図 6.19の左図がファイルメニューです。Open・Save は独自の*.obj形式ファイルに対する読み込み・保存です。この他に、Import***命令によって、eps, xbm, xpm, gif, tiff, jpeg 形式のビットマップファイルを取り込みます。実際には pbmplus という一連の図形変換ツール群と djpeg を裏で起動しているたん xpm 形式に変換しますから、これらのツールがインストールされている必要があります。

Save：保存

保存について少し詳しく説明しましょう。完成した図面を保存するにはファイルメニューで Save を選択します。前のファイルは取っておいて、ちょっと内容を変更した図を新しく名前を変えて保存したい場合があります。その場合には SaveNew を選択しますと、新しいファイル名の入力ダイアログが現れますから、それに答えます。その際に、拡張子 .obj は入力してもしなくてもかまいません。なければ補ってくれるからです。

Save と SaveNew で保存されるファイルは tgif 独自の obj 形式です。ところで、 \LaTeX で使うためには EPS 形式のファイルが必要で、そのために”Print”を実行します。ツールバー下段左から 2 番目のアイコンが Print 時の出力形式です。デフォルトは” \LaTeX (EPS)”となっています。他にも、PS ファイル、XBM ファイル、PS プリンターへの直接印刷が選択可能ですから、EPS ファイルとなっていることを確認してください。もちろん本来の obj 形式での保存を忘れないでください。後で tgif で読み込んで編集可能なファイルは obj 形式だけです。もう一つ方法があります。それは、作成された obj ファイルを EPS ファイルに変換する方法です。

```
tgif -print -eps filename.obj
```

と tgif をコマンドラインで使うことにより可能で、*filename.eps* が得られます。

読み込み : Open , Import

保存のついでに、読み込みについて話します。tgif はビットマップ画像を読み込むことができます。ファイルメニューの 6 番目に ImportXBitmap , 7 番目に ImportXPixmap , 9 番目に ImportGifFile があります。これらを選択するとファイル選択画面が現れますから読み込みたいファイル名を入力します。そして、読んだ画像に手を加えて、EPS で保存すれば L^AT_EX で利用できるようになります。8 番目には ImportEPSFile があり、確かに直接 EPS 画像が読み込めますが、残念なことに枠のみの表示で内容は表示されません。10 番目の ImportOtherFile は他のフォーマットの画像を読み込みます。インストール標準では jpeg と tiff が読み込める設定になっています。もちろん、適切なツールを登録すれば、これら例外のフォーマットにも対応できます。

ところで eps 形式を読み込むと外枠(大きさ)しか表示されませんが、xbm, xpm 形式では画像が表示されます。したがって、画像の部分を矢印で指して説明を付け加える等の場合には、xbm, xpm 形式が扱いやすいでしょう(図 6.20)。

練習 6.1 ファイルメニューから ImportXPixmap を選択し、xpm 画像を読み込んでみなさい。ファイル選択ダイアログで、/usr/include/X11/pixmaps にディレクトリを移すとよいでしょう。そこから任意の *.xpm を選びなさい(このコマンドの呼出しショートカットキーは **Ctrl**-**Alt**-**G** です)。

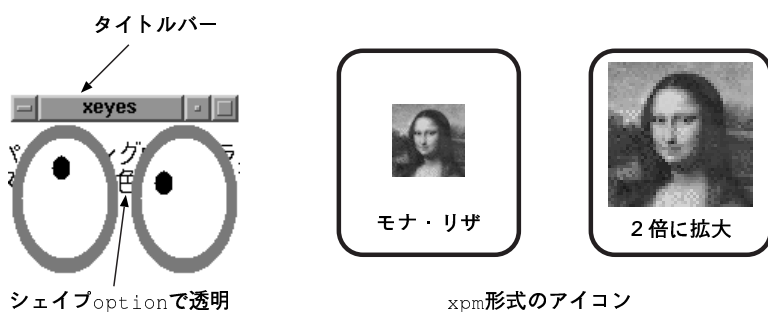


図 6.20: xpm 画像を取り込んで説明を付け加えた例

6.2.8 編集：Edit

図 6.19の右図は編集メニューです。最も項目の多いメニューです。項目はおおよそ理解できると思いますが、まぎらわしいものについて説明します。

複写：Duplicate, Copy, Paste

Duplicate (複製) と Copy (複写) はどちらもまったく同じオブジェクトを二つ以上作成する場合に使います。その違いは、Duplicate では複製されたオブジェクトが画面上に現れるので、位置を確認しながらどこかに移動し、他のコマンドを実行した時点で内容が失われます。一方、Copy はバッファを持っており、そのバッファにオブジェクトを複写するだけです (したがって、画面上にはオブジェクトが現れません)。ただし、バッファの内容を任意の時点で何回でも Paste (張付け) できます (このときにはオブジェクトの外形が表示されます)。Paste といえば、一番下に SelectionPaste (X セレクションペースト) があります。これは X クライアント同士でデータをやりとりする機構を利用するものです。したがって、例えば Mule や ktrem などの tgif 以外の X クライアント上でマウスの左ボタンを押しながらドラッグした領域の内容を、tgif 上に直接 Paste します。また、Paste コマンドのすぐ下の PasteFromFile はテキストファイルの内容を (Text オブジェクトとして) 張り付けます。

取消し：Undo

Undo：取消しはどこまでさかのぼれるかマニュアルにないのですが、かなり前まで戻ることが可能です。ペイント系のツール Xpaint や ImageMagick (の display) や Gimp では undo はたいてい 1 回だけですから、ビットマップの加工をする場合にはかなり嬉しい仕様です。

頂点：DeletePoint, AddPoint

折れ線や多角形を描いた後に頂点を削除・追加して形を変更できます (図 6.21)。削除は直観的にも明らかで、削除したい頂点をクリックします。追加する場合には、近くの頂点をいったんクリックしてからドラッグし、位置を確

定します．多角形は比較的面倒な図形をベジエ曲線を使ってトレースする場合に使いますから，この機能は絶対に必要です．なぜなら，もしこの機能がなかったら，頂点の数を間違ってしまった場合，最初から描き直しとなるからです．

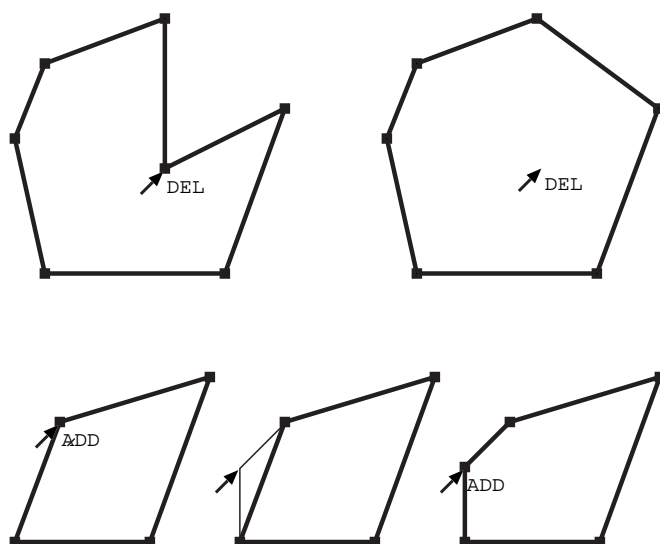


図 6.21: 多角形への頂点の削除と追加

6.3 日本語の入力

日本語化されている `tgif` では当然のことながら日本語が描けます．フォントメニューの下半分にある `Ryumin`, `Gothic`, `Ryumin-V`, `Gothic-V` が日本語のフォントです．まずこれら日本語フォントを選択してください．それだけでは日本語は入力できませんので，日本語入力システムが必要です．変換エンジンは `Canna`, `Wnn` どちらでもよいですが，X 上の入力クライアントとして `kinput2` を使います．具体的な手順は，

1. `mode` を文字列描画にする．
2. 文字を描きたい場所で左ボタンをクリックする．

3. **Ctrl**–**SPC** により kinput2 を呼び出す .

です . 日本語入力ダイアログが現れれば成功です (前出の図 6.2 参照) . ただし , kinput2 は .xinitrc に次の 1 行を追加して起動させておかなければなりません .

```
kinput2 -canna &
```

6.4 キー入力コマンド

メニューをたどってコマンドを実行するのがわずらわしくなってきたら , キー入力にコマンドが割り当てられていますから , それを使いましょう . 基本的には **Ctrl** and/or **Alt** (tgif のメニューでは # と表記されている) キーと 1 文字を入力します .

表 6.1: Tgif のキー入力コマンド

キー入力	機能	キー入力	機能
C-n	新規ファイル	C-f	オブジェクト前面
C-o	ファイル開く	C-b	オブジェクト後面
C-s	ファイル保存	Alt-z	拡大
C-p	出力 (プリンター , EPSF)	Alt-o	縮小
C-q	終了	Alt-:	デフォルトの拡大率
C-d	オブジェクト複製	Alt-p	挿入
C-y	バッファへの複写	C-Alt-y	張付け
C-x	オブジェクト削除	Alt-(EPSF ファイルの挿入
C-a	全オブジェクト選択	C-Alt-g	グリッドスナップ (トグル)
C-u	操作取消し	C-Alt-s	別名でファイル保存
Alt-i	グリッドを粗くする	Alt-d	グリッドを細かくする

6.5 他のツールとの連携

6.5.1 xpm 形式で画面のダンプ : xgrab

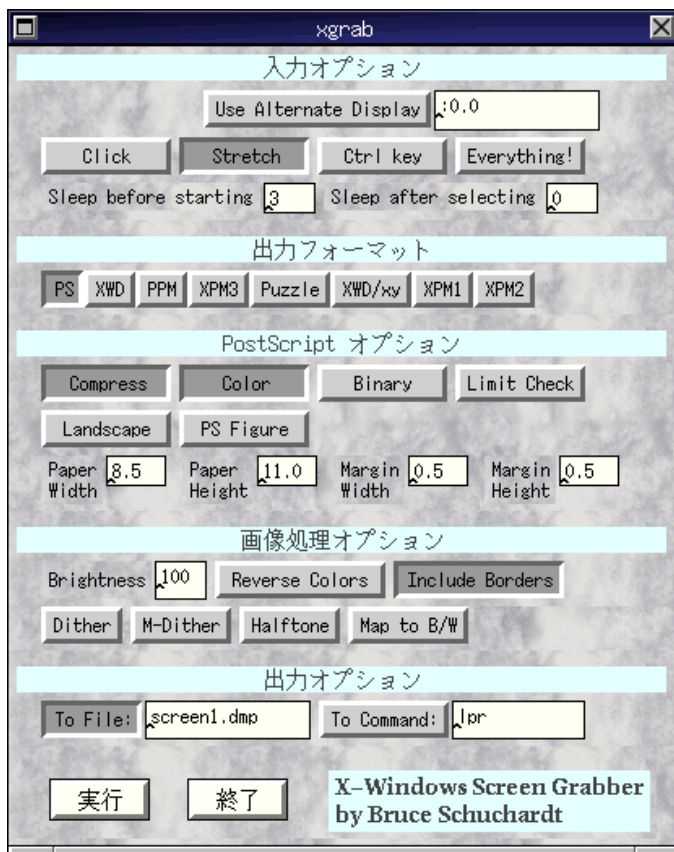


図 6.22: xgrab の実行画面

xpm 形式の画面ダンプを取るツールとしては, xgrab/xgrabsc が使えます. xgrab は xgrabsc の GUI メニューとなっていて, option を覚えなくてもよいので使いやすいでしょう. 図 6.22 を見てください. この画面はリソースを変更して日本語メニューに変えたものです. したがって, インストールしただけのものとはまったく違った画面となっていますが, 項目は同じです. 三つの項

目を選択し、ファイル名を入力します。Input Option (入力オプション)には , Output Format (出力フォーマット)には , Output Options (出力オプション)には を選択して出力ファイル名を入力しましょう。 (実行)をクリックすると、メニューが消えてカーソルが変化しますから画面ダンプしたい Window 上でクリックします。ダンプした後、再びメニューが現れますから、 (終了)で終了します。



X の画面ダンプを行うツールは上記 xgrabsc の他に、xwd, xwpick, import (ImageMagick に含まれる) などがあり、それぞれ一長一短があります。保存できる画像形式・対応する色数・重要なオプションを下表に整理しました。オプションの項で、border とはフレームの外にわずかにある境界線を含める意味です。afterstep ではこのオプションを指定しないとダンプ画像の横の境界がなくなってしまいます。delay は実行の前にウィンドウのフォーカスやメニューのポップアップを行うゆとりを与えるものです。xwd はこのオプションがないので、“sleep 5; xwd -out *.xwd” などとして時間を稼ぎましょう。pause はマウスでダンプする対象を指定した後、キー入力があるまで (マウスで他の操作ができる)、ダンプの実行を待ちます。local や screen はポップアップメニューなどのサブウィンドウをダンプする場合に必要な仕掛けです。これらの指定をしないと、例えば Mule のサブメニューなどはダンプできません。

ツール名	保存画像形式	色数	オプション
xgrabsc	xwd, xpm, ps, ppm, puzzle	8bit	GUI あり (xgrab), border, compress, delay, pause, local
xwd	xwd	24bit	border, screen
xwpick	eps, ppm, gif	8bit	compress, pause, local
import	xwd, xpm, eps, ppm, jpeg, gif, png その他多数	24bit	delay, screen

LaTeX で使うために最も便利な eps 形式について、作成されるファイル

の大きさを比べてみると xwpick が圧倒的 (1/3 以下になる) に優れています。ただし, gzip で圧縮した場合にはどれもほぼ同じ大きさになりますから, 情報量は同じです (当たり前ですね)。xwd ファイルは ImageMagick に含まれる convert で eps に変換すればよいです。import は万能ですが, 対象に含まれているボタンやウィンドウの数が多いとスキャンに時間がかかります。

6.5.2 Gnuplot からの obj 形式出力

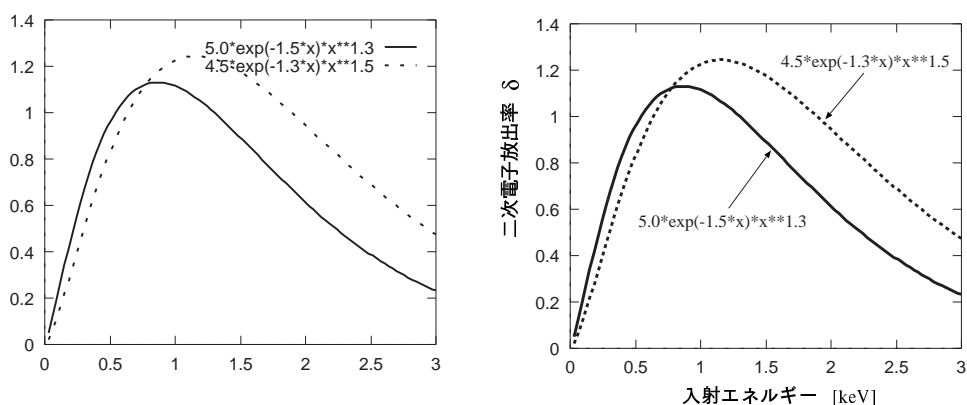


図 6.23: Gnuplot からの出力直後の図と Tgif で編集した結果の図

Gnuplot は Tgif の独自記述言語を出力しますから, とても相性がよいです。Gnuplot で日本語の説明や微妙な位置調整は難しいので, いったん Tgif に図を渡して, 修正をその場で見ながら編集しましょう。以下の Gnuplot スクリプトの出力直後の図と Tgif で編集して日本語などを加えた結果を図 6.23 に示します。

```
set output 'dionne.obj'
set term tgif
set size 0.7,0.5
set xrange[0:3]
plot 5.0*exp(-1.5*x)*x**1.3, 4.5*exp(-1.3*x)*x**1.5
```


6.5.3 PS 形式から obj 形式への変換 : pstoeedit

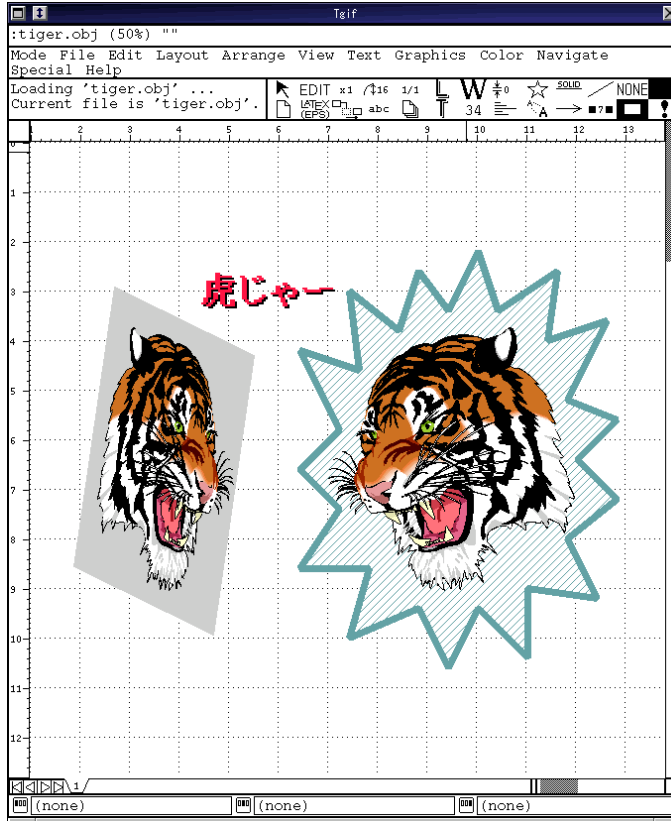


図 6.24: tiger.ps を pstoeedit で obj に変換し, Tgif 上で編集している例

UNIX 系で図形情報の記述言語は PostScript がほぼ標準といってよい地位を占めています。この PS ファイルの中身(図)を表示しながら, 編集していくツールがあればよいのですが, 残念なことに今のところ開発されていないようです。Tgif は PS 形式で保存はできますから, PS ファイルを直接編集できなくてもいったん obj 形式に変換して編集できれば実用上問題はないこととなります。そのためのツールが Wolfgang Glunz (Wolfgang.Glunz@zfe.siemens.de) 氏が開発した pstoeedit です (pstoeedit は tgif の他に xfig, FrameMaker ファイルにも変換できます)。例えば, Ghostscript の examples に含まれる, かの有名な『虎』の PS 画像(すべて多角形で描いている労作)を

```
pstoedit -f tgif tiger.ps
```

と obj 形式に変換します。そうしてできあがった tiger.ps.obj を Tgif で編集している例を 図 6.24 に示します。

なお、pstoedit は簡単な PS 命令 (Flat PostScript) しか理解しないので、変換できない場合も多いようです。

6.5.4 写実的に見せる工夫

Tgif は 2D のドローイングツールですから、3D あるいは写実的・立体的に描くための機能は持っていません。どうしても写実的な表現をしたい場合には、影を付けるなどの工夫をしてみましょう。図 6.25 などはその例です。

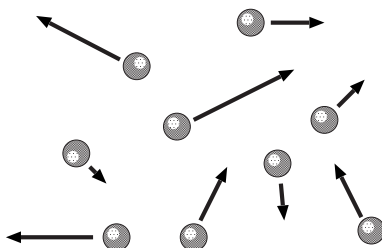


図 6.25: 影付けにより立体化を試みた

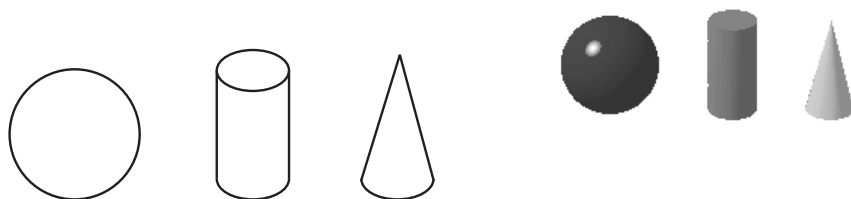


図 6.26: Tgif で影付けができるか?

図 6.27: POV-Ray で作成した立体画像

しかし、球以外の図形に関してはどうしたらよいのでしょうか。例えば、図 6.26 の図形に影を付けて、立体的にしてみてください。グラデーションがないと、ドロー系のツールで影に濃淡をつけるのは、非常に大変ですね。こういう場合はむしろペイント系のツール (Xpaint や GIMP) のほうがずっと

描きやすいはずですが、Tgif ではその中で描くのはあきらめて、ペイント系のツールやレイトレーシングツール POV-Ray で描いた立体的な画像を読み込んで活用しましょう (図 6.27)。

6.6 テクニックの上達

写真とは異なり、図面は概念を的確に伝えるための強力な手段です。したがって、細かすぎず粗すぎず適当な模型化が必要です。そのためには、複雑な図形を単純な図形の組合せに分解する能力が必要です。そこで、まず基本的と思われる図形の描き方を十分に練習しましょう。

6.6.1 基本要素を描く

メニューにある基本図形には、文字、四角、円 (楕円)、折れ線 (線分)、閉じた折れ線、扇形 (円弧)、丸四角、フリーハンド が用意されています。Ver.3 からは吹出しがメニューバーに載っています。

練習 6.2 基本要素を描く練習をしてみてください(図 6.28)。

1. 種類を変えて適当な大きさのものを描いてみましょう。
2. 線の太さや線の種類を変えてみましょう。
3. 内部や外枠を変えてみましょう。

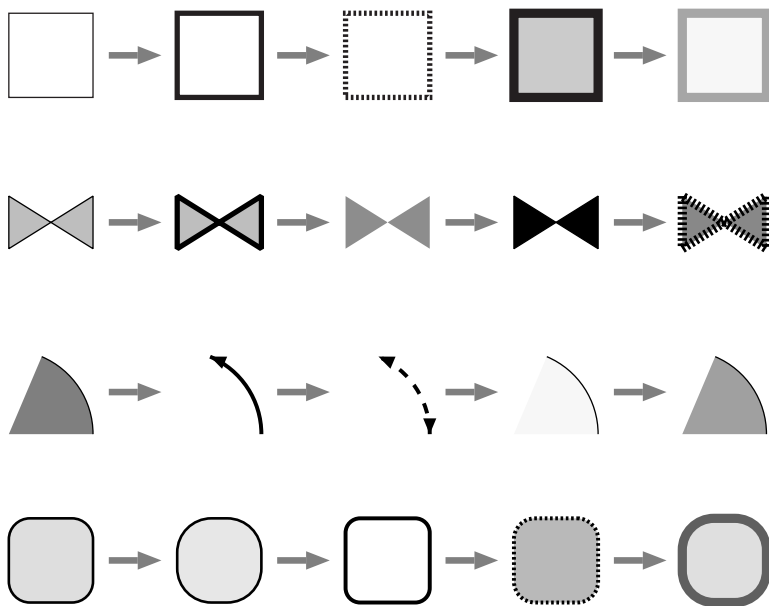


図 6.28: 基本要素の描画実行例

6.6.2 四角や円などの単純な組合せ

四角や円(楕円)はメニューでも上位にあることからわかる通り、非常に多用される要素です。例えば、開いた箱を描くには、直観では折れ線ですが、四角を二つ(一つは辺を消すための見えない四角)組み合わせたほうが後で修正が楽という場合があります。四角や円を組み合わせるブロックダイアグラム・影付きのカード・グラフ用紙などを書いてみましょう(図 6.29)。複製: **Ctrl**-**D** をうまく活用してください。

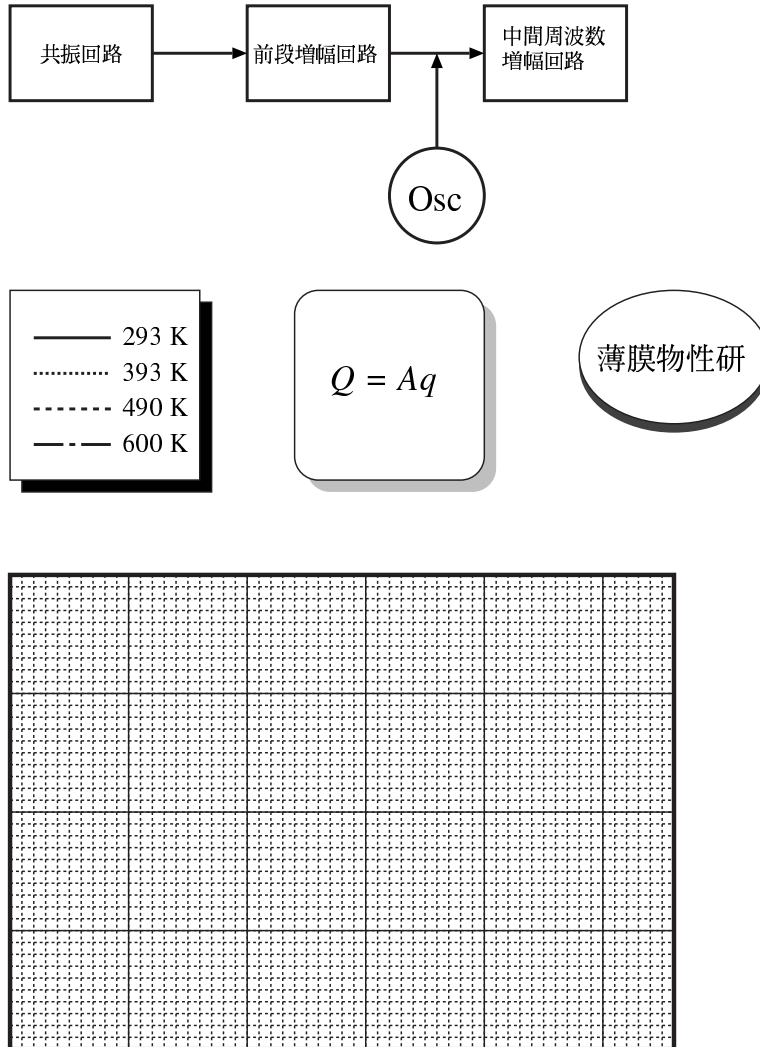


図 6.29: 基本要素の単純な組合せで実現できる図

6.6.3 Fill-White の応用

薄膜を作製する基板を描く場合を考えましょう。これは、四角でよいように思えますが、図 6.30 の二つの図を比較してください。右側は基板内部を表す

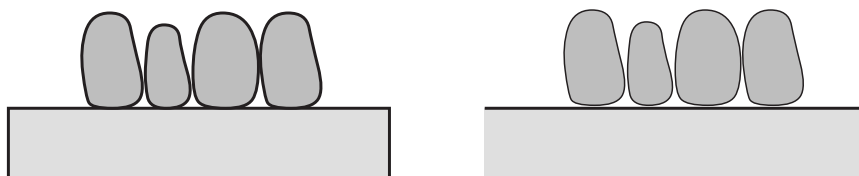


図 6.30: 外枠を内部と同じ塗りつぶしにする。

ための外枠が透明で内部を塗りつぶした四角と、表面を表すための直線との組合せとなっています。外枠は黒という概念を捨てると応用がみえてきます。特に、外枠も内部も白のものは、下の図形を覆い隠すことができます。

練習 6.3 二つの真空槽を管で接続した図を描くとしましょう(図 6.31)。真空槽を四角や円で描くと管との接続部分が閉じられてしまいます。この部分に穴をあけてください。すべてを折れ線で描くといった暴挙はいけません。解決方法は塗りつぶしです。

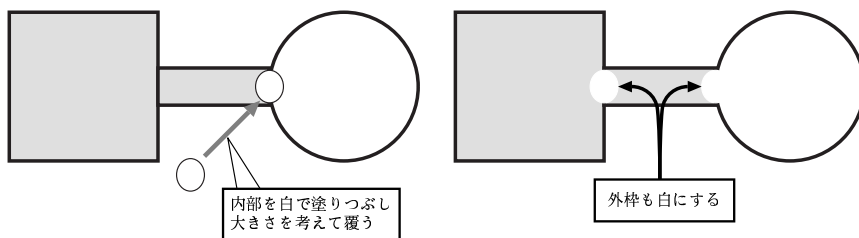


図 6.31: 見えない覆い(塗りつぶし)の効用

6.6.4 折れ線で描けるもの

折れ線は基本中の基本要素です。四角も折れ線で描くことが可能ですし、究極的には曲線も折れ線で近似されます(フリーハンドはそうになっている)。しかし、四角は四角にまかせて、折れ線ならではの簡単な図形を描きましょう。

練習 6.4 折れ線を使って電子部品の図記号を描いてみなさい(図 6.32)。電池, 抵抗, コンデンサー, アース, スイッチ, コイル(インダクタンス), 可変抵抗, トランジスター, オペアンプなど。円と組み合わせなければならないものもありますが, 比較的簡単に描けるはずです。

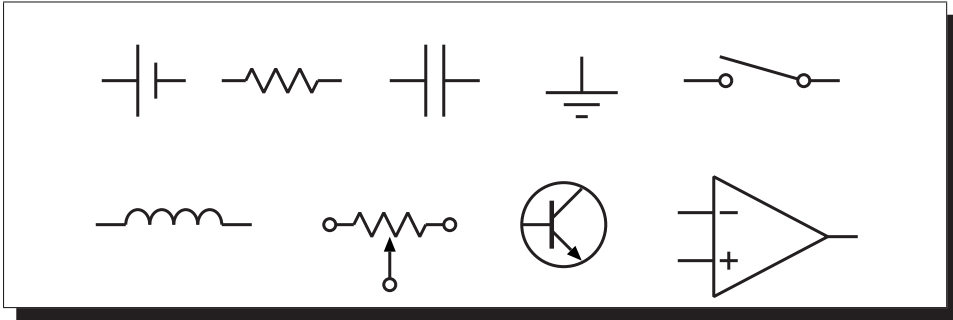
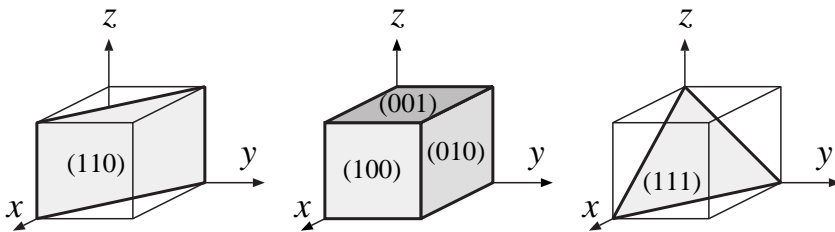


図 6.32: 折れ線による電子部品の図記号

練習 6.5 ミラー指数の概念を表す次の図を描いてみなさい。



6.6.5 ベジエ曲線の応用

ばねや正弦波などの曲線をフリーハンドで書くのは、マウスでは非常に難しいです。フリーハンドでは短い線分（直線）で近似することになり、見苦しい図ができあがります。多数の短い線分を描くのではなく、代表する数点を制御して、その間は計算によりある種の曲線で補間して全体を構成する方法を練習しましょう。直線上にない3点を滑らかに繋ぐ（補間する）曲線には、2種類用意されています。3点を通過するように三次曲線で補間するスプライン（spline）と、端の2点は通過するが、中間へ向けての接線が滑らかになるように補間する（当然中間の点は通過しない）ベジエ（bezier）です。一見スプラインのほうが使いやすそうですが、実はスプラインは制御点を無理に通過させる宿命のため滑らかでなくなる場合が多々生じます。ベジエ曲線に慣れることをすすめます。

練習 6.6 長い小括弧や中括弧、正弦波、ばねなどをベジエ曲線を使って描いてみなさい（図 6.33）。複写や線対称移動をうまく使うこと。また、曲率半径を小さくしたい場合（鋭い角がほしい場合）には、制御点を近づけてください。

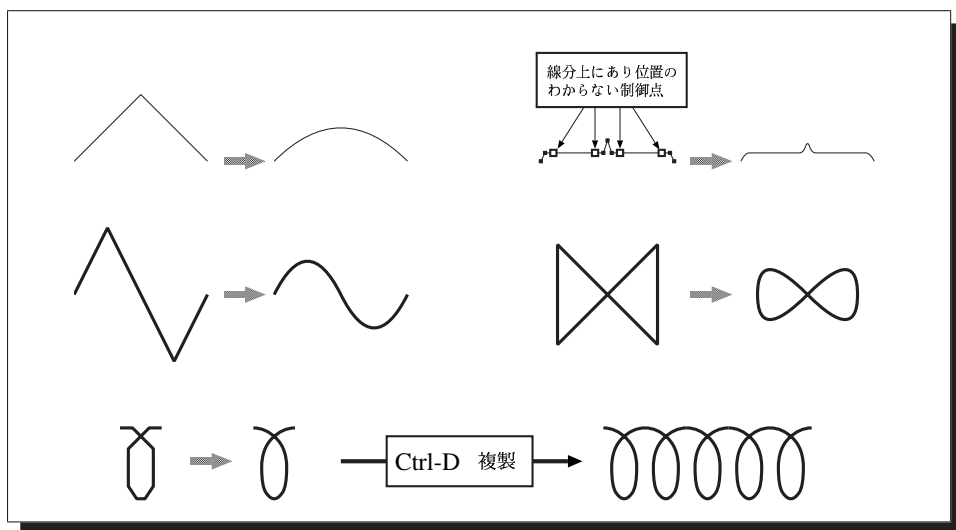


図 6.33: ベジエ曲線で描いた、括弧類・正弦波・ばね

練習 6.7 ベジエ曲線はとても便利です．スパッタリング中のプラズマの様子であるとか，薄膜の島構造の模式などを描いてみましょう（図 6.34）．

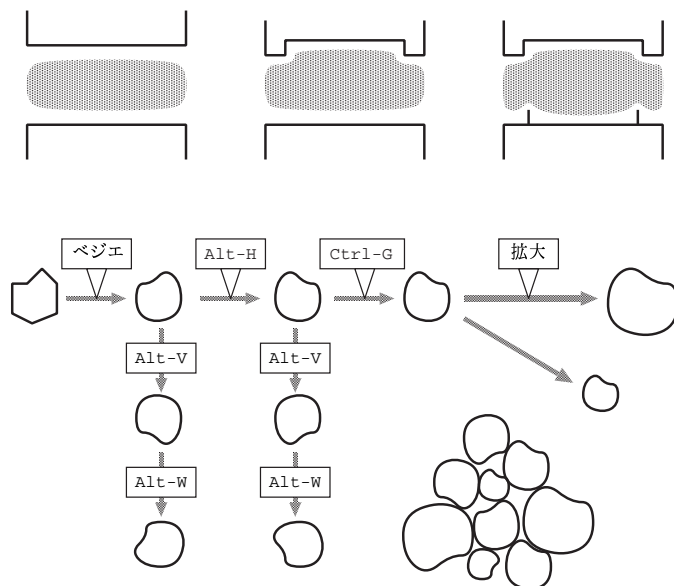


図 6.34: ベジエ曲線で描いた，プラズマ領域，薄膜の構造

6.6.6 変形

描いた図形はいつでも変形（拡大・縮小・回転）が可能です（図 6.35）．図形を選択すると制御点が現れますから，制御点をマウスの左ボタンで選択してドラッグし，望みの位置で（ドラッグにつれて図形が変形しますから確認して）リリースします．変形はグループ化された図形に対してはグループ全体に作用します．ただし，文字サイズだけは（位置は動く）変更になりませんから，後で調整が必要でしょう．

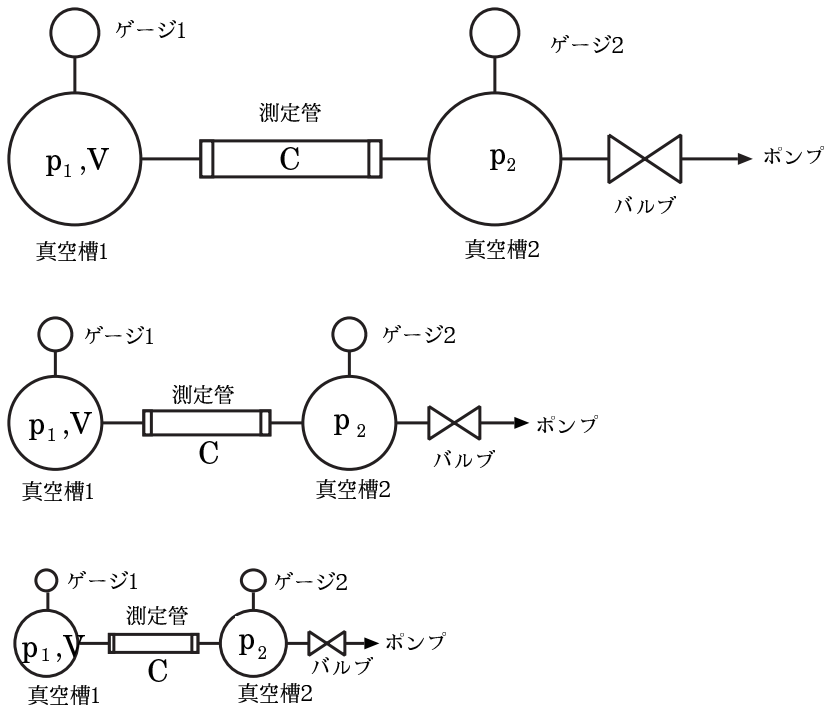
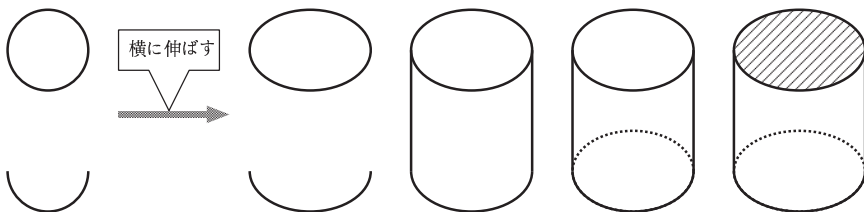


図 6.35: 変形の実行例 : グループ化して複写・縮小・文字位置の修正

練習 6.8 円筒を斜めから見た図を描いてみなさい . 問題は半分の楕円をどう作るかですが...

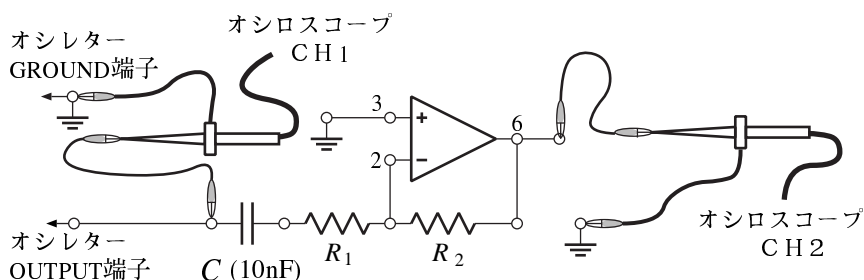


半円を変形 (横に伸ばす) すればよいことに気づきましたか?

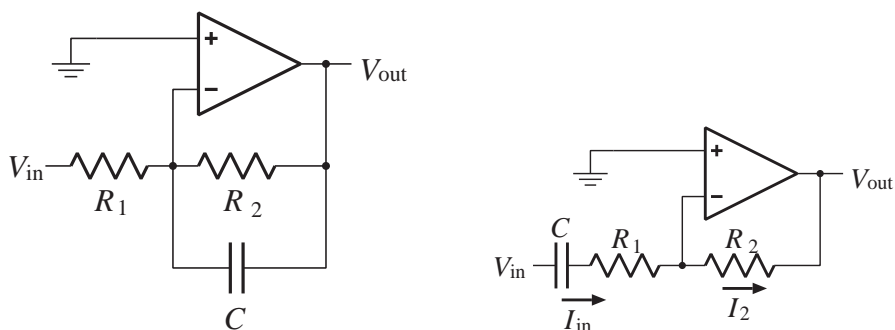
6.7 図面集

物理関連の実験用テキストを $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}+\text{Tgif}$ で作成しています．その中から面白そうなものを選んでみました．

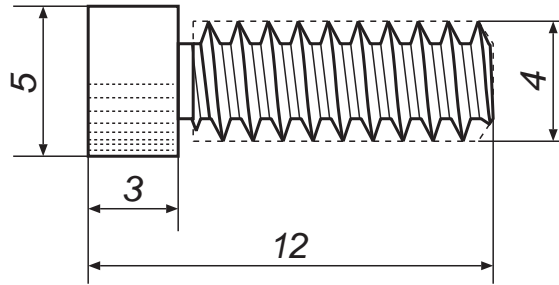
練習 6.9 次の図を描いてみなさい．実体配線の部分はベジエ曲線で描きましょう．ミノ虫クリップがなかなか面倒です．簡単に描くにはどうしたらよいでしょうか？



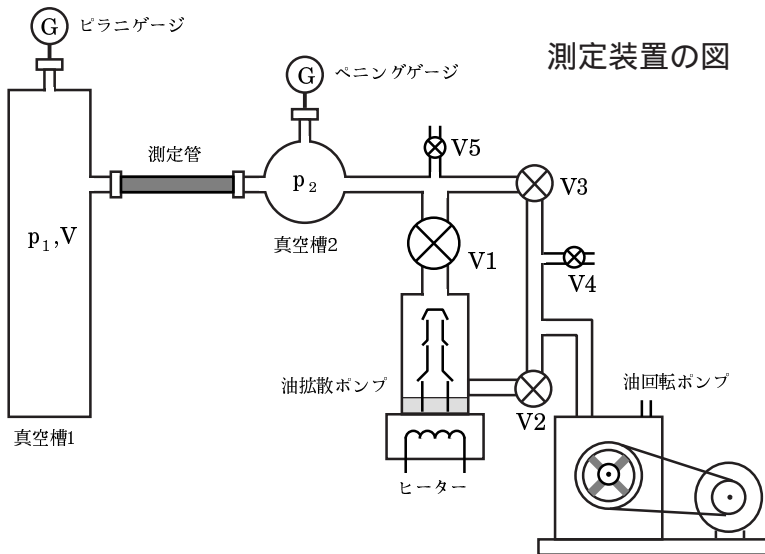
練習 6.10 次の回路図を描いてみなさい．コンデンサーや抵抗や OP アンプなどの図記号をあらかじめ作成しておき，ライブラリとして使いましょう．

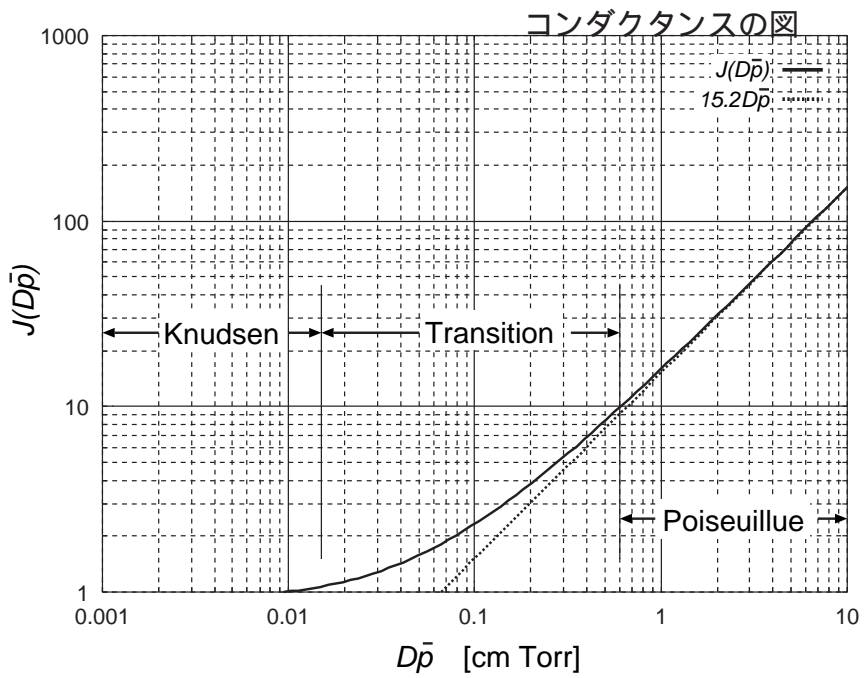


練習6.11 次の図を描いてみなさい．これはかなり面倒です．特にねじの部分の立体感を出すためにベジエ曲線を使っています．これを直線で簡潔に表現してみてください．



練習6.12 真空技術の実験テキストから抜粋しました．コンダクタンスの図はGnuplot を併用します．





練習6.13 tgif で図面を作成し, \LaTeX に組み込んでください.

出力結果 : EPS 図面の張付け

長さ l の片持ち梁の自由端に集中荷重 F を加えた場合の棒の位置 x と変位 y が満たす微分方程式は

$$\frac{d^2y}{dx^2} = -\frac{1}{R} = \frac{(l-x)F}{EI_z} \quad (6.1)$$

である, これを x について解き (積分する), 境界条件を考慮すると, 棒の自由端の降下量 δ は

$$\delta = -\frac{F}{6EI_z}(3lx^2 - x^3)\Big|_{x=l} = -\frac{Fl^3}{3EI_z} \quad (6.2)$$

と求められる.

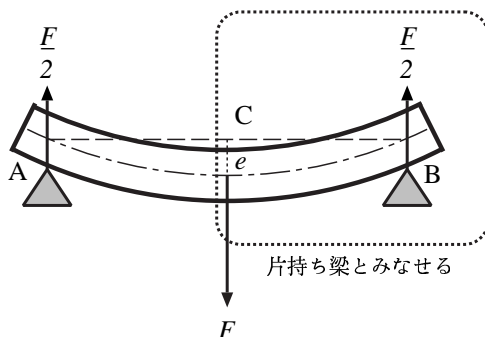


図 6.36: 両端支持棒の中央の集中荷重によるたわみ

ところで図 6.36 に示すように, 実験では棒の両端を支えて棒の中央に集中荷重が加わっており, 中央から半分の部分は, 長さ $l/2$, 荷重 $F/2$ の片持ち梁とみなすことができる. よって, 式 (6.2) にこれらの値を代入して, 次の結果を得る.

$$e = \frac{Fl^3}{48EI_z} \quad (6.3)$$



関連サイト情報

- <http://bourbon.cs.umd.edu:8001/> , 作者 W. CW. Cheng 氏の Web サイト .tgif の help で表示されるアドレスは接続できませんでした . 注意してください .
- <http://bourbon.cs.umd.edu:8001/index.obj> , 作者 W. CW. Cheng 氏の Web サイトですが .~/Tgif_hotlist に記述しておく と tgif の Navigate 機能でアクセスできます . 前項同様 help で表示されるアドレスは無効のようです .

索引

- align, 131
- arrange, 130

- back, 130
- border, 142

- convert, 143
- copy, 138

- delay, 142
- duplicate, 138

- eps, 142

- fill, 128
- flip, 131
- front, 130

- GIMP, 146
- gnuplot, 143
- grid, 133
- gridsnap, 133
- group, 131

- import
 - ImageMagick, 142
 - Tgif, 137

- layout, 133
- line, 127

- obj, 136
- open, 137

- paste, 138
- pause, 142
- pen, 128
- point
 - AddPoint, 138
 - DeletePoint, 138
- POV-Ray, 146
- print, 136
- pstoedit, 144

- rotate
 - PriceRotate, 133
 - Rotate, 132

- save, 136
- shear, 127

- undo, 138
- ungroup, 131

- xgrab, 141
- Xpaint, 146
- xwd, 142
- xwpick, 142

- zoom, 134

折れ線, 150

画面のダンプ, 141

キー入力コマンド

 Tgif, 140

キャンバス, 123

スプライン曲線, 151

透明, 128

ドロ-系, 121

日本語入力

 Tgifの場合, 139

ファイルメニュー, 136

ペイント系, 121

ベジエ曲線, 151

編集メニュー, 138

モードメニュー, 126