

Windows 用の Qfig を使う

岩熊哲夫：Qfig 作者

2003/ 某月 / 吉日 (Ver.1.0)

1 ダウンロード

ソフトそのものは Vector Design あるいは私のウェブ・ページから、ダウンロードできます。

2 起動と終了

2.1 最初のセッティング

配布のアーカイブを解凍し、すべてのファイルを同じディレクトリ（フォルダ）に置いてください。レジストリは触りません。Qfig.exe を実行してください。実行できず、何かダイナミック・リンク・ライブラリが無いというメッセージが表示された場合には、同じアーカイブ内のライブラリ群のアーカイブも解凍し、中身をすべて Qfig.exe と同じディレクトリに置いてください。

最初の最初には図1のような設定画面が表示されます。この設定は **optlon** ボタンを押すことによっていつでも変更できます。例えば図のように「ヘルプ」の **Japanese** をセットしておいてください。オプションは次のような意味です。

Warning: 警告の音設定で、何も設定しないとピープ音。Sound である種の音が出ます。さらに Voice にすると、警告文言が発せられます。

List & Help: Japanese をセットすると操作指示を日本語で表示します。

Character Pitch: これは、図化ソフト Ngraph のピッチ（これは私がいい加減に PostScript ピッチを設定したもの）にするか、通常の \TeX ピッチにするかのスイッチで、 \LaTeX を使う方は後者にしておいてください。

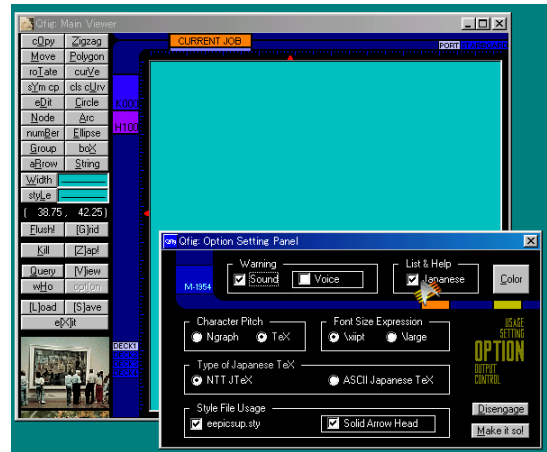


図 1: 一番最初の立ち上げ画面 — 各種設定

Font Size Expression: $\text{\LaTeX}209$ 用は `\xiipt` の方で構いませんが、 $\text{\LaTeX}2_{\epsilon}$ の場合は `\large` にしてください。

Type of Japanese TeX: アスキーの日本語 \TeX を使うか、NTTJ \TeX かの設定です。

Style File Usage: アーカイブ中の eepicup.sty を使うか否かの設定です。塗りつぶしや網掛けのためのコマンドについて eepic.sty を拡張したものです。Solid Arrow Head を設定すると、矢印の矢を塗りつぶします。

いずれの設定も、変更した場合には **Make it so!** を押した上で、右上の図が **Esc** キーを押して窓を閉じてください。変更を無視する場合は **Disengage** です。

設定画面中右上の **Color** ボタンを押すと図2の窓が出てきます。上の画面が描画領域の色と同じものです。この画面を見ながら、左下の **BACKGROUND** の RGB を設定すれば背景色の設定ができます。右下の **OBJECTS** の RGB で線の色の設定ができます。**ENERGIZE** ボタンで設定が有効になります。**ENERGIZE** ボタンを押さなければ設定は変更

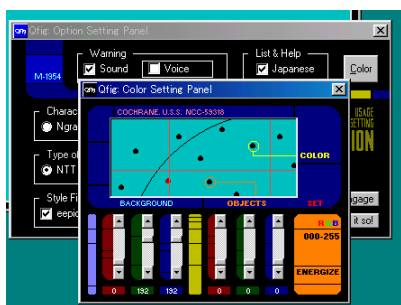


図 2: 色の設定

されません．この窓も，右上の図で閉じてください．

2.2 通常の起動

これは単に Qfig.exe を実行するだけです．

2.3 終了

図1のように，種々のコマンド・ボタンは窓の左側に並んでいます．ボタン群下の Seurat の絵のすぐ上にある **eXit** を押せば終了します．もちろん窓右上の図でも終了しますが，**eXit** を押して終了する場合には

- 描画途中だったときに，後述のような警告を発する．
- 最後にファイル・アクセスしたサブディレクトリ（フォルダ）の位置を次回まで保持する．
- Qfig そのものの窓サイズと位置を次回まで保持する．

ことになります．それに対し，図で終わるとこのような措置は取られず，強制終了になります．

描画途中で止める場合に **eXit** を押すと，図3のような確認のための窓が出てきます．たった二つの機能しかない窓にしては大きすぎますが，中央やや右の赤い小窓にメッセージが点滅表示されます．この描画途中での中止の場合は ‘Self-Destruction?’ と表示されます．もしそのまま終了してもいい場合には，**Engage** を押してください．終了する前に保存しておきたい場合には，**Cancel** を押した上で，適切な処理をしてから終了してください．描画したものが保存された状態で終了する場合には，確認窓は出てきません．

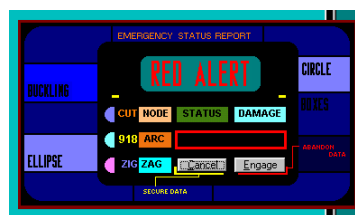


図 3: 終了時の確認画面

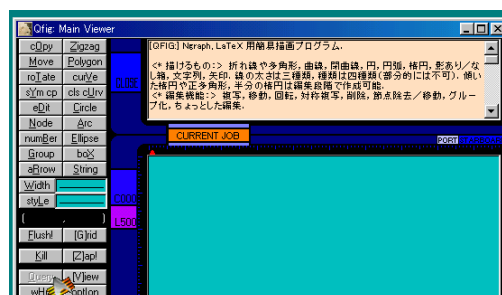


図 4: ヘルプ画面の表示

2.4 コマンド・ボタンとキーボード

optIon の I のように，コマンド機能標記の文字列の中に大文字で下線がついているものは，ボタンをマウスでクリックする代わりに，キーボードの **Alt** とその文字の同時押しでも同機能を実現できます．例えば **Alt**+**i** で「設定変更の窓が開き」ます．さらに，鉤括弧付きの **e[X]it** の X の場合には，キーボードの **Shift**+**Ctrl**+**Alt** と同時に該当するキーを押してください．例えば「終了」の場合には，この三つのキーと **x** を同時に押せばいいことになります．

窓によってはボタンらしいものが見えないものもあります．例えば前述の色設定窓の **ENERGIZE** ボタンがそれに当たります．こういったボタンは，マウスがその上を通過するときに枠線が発生しますから，もし分からなくなったら，マウスをそれらしい所に移動させてみてください．ちょっと遊びが過ぎたかもしれません．反省．

2.5 バージョンとヘルプの表示および座標位置

バージョン情報は **wHo** で表示されます．現れた窓は **ok** を押せば消えます．

起動してすぐに，**Query** のボタンを押してください．こうすると，図4のように，描画面面上部に一種のヘルプが表示されます．起動直後のヘルプ窓には，描画における一般的な情報が書いてあります．

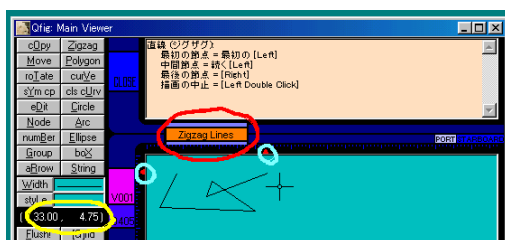


図 5: 各コマンドの操作マニュアル

一通り目を通してください。この窓は、ヘルプ表示部分の左側の **CLOSE** で閉じます。ヘルプ窓を表示したまま、左の Seurat の絵の写真部分をダブル・クリックしてみてください。ちょっと速すぎますかね。 **Esc** キーで止めることができます。

このヘルプ窓を開けたままであれば、その後例えば **Zigzag** ボタンを押すと、図5のように目的とする描画の操作マニュアルが表示されます。これを見ながら練習すれば、だいたいの描画の仕方は分かります。この同じ図の左側の黄色い楕円で囲った部分に、現在のマウス・カーソルのある位置が数値で表示されます。また描画面のすぐ上と左に水色丸で囲った赤いマーカーが、マウス・カーソルの位置を定規目盛上に（この図では微妙にずれてしまっていますが）示したものです。数値の座標値は mm 単位、定規の小さい目盛も mm です。

同じ図5のヘルプ窓と描画面の間の水平棒部分の、赤い楕円で囲んだところに小窓があり、この図の例では **Zigzag Lines** と表示されています。これは現在どのコマンド・ボタンが押されている状態なのかを示しています。この例では、直線描画コマンドの実行中であることを示しています。ときどき、その他の情報もこの小窓に現れることがあります。例えば何か線を描いた後に **Flush!** を押してみてください。再描画なので **Re-Drawn!** と表示されます。

3 基本的な描画

3.1 直線と曲線、閉じた図形

直線と曲線はそれぞれ **Zigzag** および **curVe** で描くことができます。このボタンを押した後、描画面中の連続する左クリックごとの位置を線で結んで行き、右クリックが最終点の決定になります。曲線の場合は、この最終点の決定で（極めて適当な）曲線が

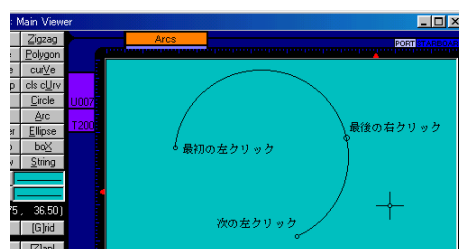


図 6: 円弧描画の三つのステップ

描画されます。閉じた図形の場合は、 **Polygon** あるいは **cls cUrv** を使います。この場合は最終点の右クリックで、始点と終点を結んでしまいます。

以上の四つのコマンドでたいの図形は描けますが、正確に直角を持つ四角形が必要な場合には、 **boX** を使います。最初の左クリックと次の右クリックを対角点とする四角形が描画されます。右クリックの前にゆっくり¹左クリックを続ければ、この四角形に三種類の濃度の網掛けを設定できます。4 回クリックすると網掛け無しに戻ります。

画面上で鉛直あるいは水平な線は、 **Shift** を押しながらマウスを動かせば引くことができます。しかしそういう正確な線を描きたい場合には後述の第9.1節で紹介する DOS 窓版の利用をお勧めします。また、マウスではなく、キーボード上の矢印キーでも位置を決めることができます。通常は 0.25 mm ピッチでしか動きませんが、 **Ctrl** を押しながらであれば 1 mm ピッチで動かすことができます。矢印キーを使っている場合には、左右クリックの代わりに、スペースキーとリターンキーを使う方が便利でしょう。ただし描画中のテンポラリな点が画面に残りますので、後述の **Flush!** を押して再描画させてください。

3.2 円・円弧と楕円

円は **Circle** を押した後、最初の左クリックが中心位置、次の右クリックが円弧上の点を決定します。また楕円は **Ellipse** を押した後、最初の左クリックと次の右クリックで囲まれた四角形の中にフィットする楕円を描きます。

円弧の場合には **Arc** を押してください。そして図6のように、円弧の弦の片方の位置が最初の左クリックです。次の左クリック位置に弦のもう片方が位

¹ 左クリックを急いしまうと、ダブル・クリックとみなされてしまいます。

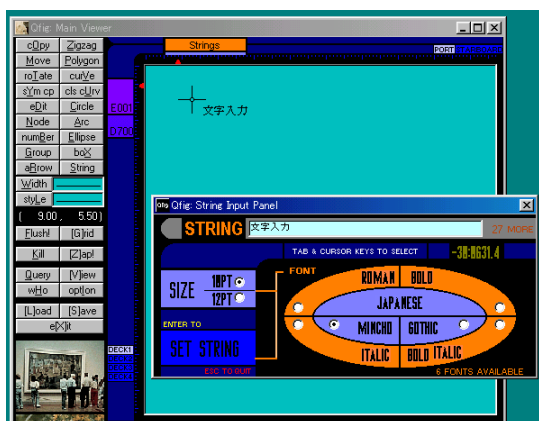


図 7: 文字列入力窓

置します。あとは、その二点を通る円弧が補助的に表示されますから、最後の右クリックで、欲しい円弧上の一点を決定します。

3.3 文字列

文字列を入力するには **String** を押してください。そうすることによって図7のような入力窓が現れます。フォントのサイズは 10 pt, 12 pt の二種類。フォント種の英数字は Roman, *Italic*, **Bold**, **Bold Italic** の四種類、漢字は明朝とゴシックの二種類が選べます。上部の **STRING** の右にある入力窓に文字列を入れ、左下の **SET STRING** を押せば、描画画面上的の位置決め状態になります。適当な位置で左クリックすれば、そこに文字列が確定入力されます。文字列入力窓は **図** を押して閉じてください。

ただし、パソコンにそれ相当のフォント²が存在していない場合には、必ずしも上述の六種類のフォントが選べるとは限りません。利用できるフォント種数は入力窓の右下に数字で表示されています。

上付き下付きの記号は、例えば 'A^p' と入力したい場合には 'A^p@'、また 'B_{min}' としたい場合には 'B_min@' と入力すれば可能です。しかし、こういった数学記号に類した文字列の場合は、Qfig 上で入力するのではなく、第6節で述べるように、一旦 T_EX ファイルに出力してから編集の方が綺麗に仕上がります。このように ^, _, @ は特殊記号となりますから、もしこの記号を実際に入力する必要がある場合は \ (日本語表示システム上での \ は ¥ マークです) で

回避してください。すなわち、以上の四種類の記号入力はそれぞれ、\^, _, \@, \\ となります。

3.4 線の属性

各種図形の線の太さは **Width** で三種類、実線なのか破線なのかは **style** で四種類選べます。ただし、線の種類については順に、実線・破線・長い破線・一点鎖線のように描画画面には表示されますが、出力した L^AT_EX ファイルによっては、これが異なります。例えば eepicsup.sty を用いた L^AT_EX ファイルの場合は、順に、実線・破線・一点鎖線・二点鎖線になります。

3.5 グループینگ

描画された図形を、例えばコピーする場合、いくつかの図形を同時に同じ配置でコピーしたい場合があるかもしれません。その場合には **Group** を押してください。そして、グループイングする図形を順に左クリックで設定していきます。グループイングする図形がすべて選択されたら、図形の無いところでダブル・クリックしてください。グループイング開始からダブル・クリックまでが一組の図形群グループ化になりますので、複数のグループごとにコピーや移動が行えます。ただし、グループイング情報はファイルには保存されませんので注意してください。

3.6 コピーと移動・回転

図形のコピーは **cOpy** で、移動は **Move** で行います。一つの図形だけの場合は、その該当図形の節点を一つ左クリックで選び、目的位置まで移動した後、右クリックでコピー・移動が完了します。複数の図形を同時に操作したい場合は、前述のグループイングをしておけば同様の操作でコピー・移動ができます。あるいは、移動する図形群を囲むように左クリックのままドラッグし、その対象とする図形群の領域をすべて囲んだところでマウスから一旦指を離します。そのままマウスを移動させると、対象とした図形群も移動しますので、そのまま目的位置まで動かして右クリックすることによっても、コピー・移動が行えます。この領域図形操作は、図形の一節点でもその領域内にあるものは対象図形とみなされることに注意してください。

²MS Sans Serif と MS 明朝・MS ゴシックを使います。

ちなみに、コピーと移動は連続してできるように、その機能は継続されます。もし必要な操作が終わった場合には、マウスを任意の位置でダブル・クリックしてください。これで選んだ操作機能はリセットされます。他のボタンを押したときも、実は同様のダブル・クリックで操作を中断できます。

図形の回転は `roTate` で行います。ただし、円・楕円（後述の斜めの楕円と半楕円は除く）・円弧・箱・文字列は回転できません。また回転する元の図形も保持されますから、この機能は回転コピーになります。`roTate` を押すと、描画領域の上横棒部分に `ANGLE=Pi/4` と現れます。これは、一回の左クリックで回転する角度の設定小窓です。デフォルトでは $\pi/4$ つまり 45 度になっています。この `4` の中の数値を適当に変更すれば、任意の角度での回転ができます。数値は実数でも構いません。また回転は反時計回りです。角度を設定した後、該当図形の節点にマウス・カーソルを置いて左クリックすると、その選んだ節点を中心にして設定した角度だけ回転します。ゆっくりと³左クリックを続ければ、さらに同じピッチで回転を続けます。欲しい位置に来たところで右クリックして決定します。

また対称コピーもできます。`sYm cp` を押した上で該当図形の一節点を選び、左クリックすればその節点を通る水平軸に対しての対称コピーになり、右クリックで鉛直軸に対しての対称コピーになります。

3.7 矢印の付加

直・曲線や円弧の端点には `aRow` で矢印を付加することができます。端点で左クリックすればその点に、右クリックすると両端に同時に矢印が付加されます。既に矢印がある節点で左クリックすると矢印を削除します。右クリックをすると、両端の矢印状態を反転させます。つまり、両方に矢印がある場合には両方が削除されます。反対側に矢印が無い場合には、そちらが矢印になって選んだ端の矢印が削除されます。

³左クリックを急いでしまうと、ダブル・クリックとみなされてしまいます。

3.8 直・曲線の節点の移動や切除

直・曲線の節点の変更には `Node` を使います。変更したい節点上で左クリックするとその節点を移動できます。移動先の位置に持っていった後、再度左クリックすれば位置が決定されます。これに対し、変更節点上で右クリックしてしまうと、その節点は削除されます。ただし、2点でできた直線の節は削除できません。また3点でできた曲線の端点を削除しても、節の数は3以下にはなりません。その瞬間は直線に見えますが、実は曲線のままです。ですから、再度 `Node` を押せば中間にもう一つの新しい節点が挿入されているのが見え、いずれかの節点を移動させると曲線になります。

4 編集による図形の変更について

4.1 編集における線の属性の変更

図形の形や線の属性や文字列の内容を変更したい場合には `eDit` を使います。このコマンドを用いれば、たいいていの図形の形態変更と文字列の変更ができます。線の太さや種類といった属性は、該当図形の節点をクリックする時点での `Width` と `styLe` の設定に変更されますので、注意してください。私自身もこれを忘れて、何度も `eDit` をし直すことがあります。

4.2 直線・曲線

直・曲線および多角形と閉じた曲線の場合には、`eDit` の左クリックで真っ直ぐか曲がっているかという属性が交換されます。つまり左クリックで、直線は曲線に、曲線は直線になります。ゆっくり左クリックを続ければ、これが繰り返されます。

4.3 正多角形や斜めの楕円・半楕円

ただし、多角形・曲線・閉じた曲線の場合に限っては、節点で右クリックをすると異なる結果になります。多角形を右クリックすると正多角形になります。閉じた曲線を右クリックすると斜めの楕円になります。開いた曲線を右クリックすると、半分の楕円になります。図8がその例です。この図は、上の3つの図形をその下にコピーした上で、下の図形だけ

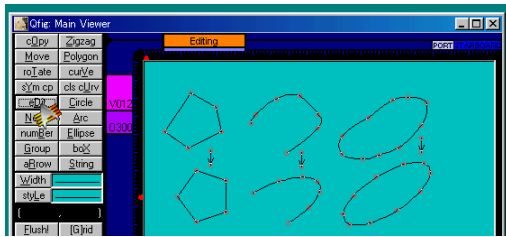


図 8: 正多角形と斜めの楕円・半楕円

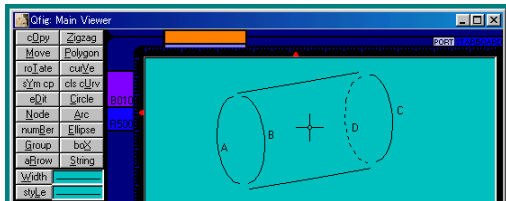


図 9: 円柱を描く：半楕円・対称コピー・グルーピング・属性変更

を **eDit** で右クリックして作りました。

この半楕円と斜めの楕円が、個人的には一番欲しかった機能です。このためにこのソフトを作ったと言ってもいいくらいです。というのも、図9のような円柱を描く場合に必要だからです。この図では分かり易いように、わざと部品の図形を少し離して置いてあります。まず、この図の半楕円図形 A を **curVe** と **eDit** の右クリックで作ります。次に、これを **sym cp** で水平軸回りに一回、鉛直軸回りにもう一回対称コピーし、途中の（水平軸対称の）図形は消去します。残った図形、つまり図形 A の点対称の図形を適切な位置まで移動させたのが図形 B です。この図形 A、B を一緒に（グルーピングしてもいいし、領域コピーでもいい）右の方にコピーします。そして、**styLe** で線の属性を破線にした後、図形 A のコピー版を **eDit** で左クリックを二度（一度は折れ線になるので、再度で元の曲線に戻す）行えば、図形 D ができます。あとは、線の属性を実線に戻した上で、上下の直線を描画すれば、見えない部分が破線になった円柱が描けるわけです。

4.4 その他の図形の編集

その他の図形を **eDit** すると

円: 左クリックで選んだ円の半径のみを、右クリックで変更できます。

楕円: 左クリックで選んだ楕円の半径比を、右クリッ



図 10: ファイルへの保存と変換

クで変更できます。

円弧: 左クリックで選んだ円弧の再設定です。最初の点が円弧描画の最初の左クリックに当たります。あとは円弧の描画手順に沿ってください。

箱: 左クリックで選んだ箱の節点を固定したまま、網掛けともう一つの対角点を設定し直せます。

文字列: 左クリックで文字列を選択すると、文字列入力窓が現れ、その文字列の内容を変更できます。

となります。

5 ファイル・アクセス

5.1 保存と変換

描いた図は **Save** で保存されます。画面には図10のような窓が現れます。上半分が保存機能部分です。保存する媒体に合わせて、適切なドライブとディレクトリを選びます。そしてファイル名を **Qfig original type [.qfg]:** の右窓に入力します。拡張子は入力しなくても構いません。そしてこの窓の左側の **SAVE** をクリックすれば保存されます。保存後 **DONE** を押せば、この窓は閉じます。グルーピング情報は保存されませんので注意してください。

保存時に拡大・縮小したい場合は、その下の **FAC-TOR** で % 指定をします。50% から 200% までの設定ができます。別々のファイルにある図形を合体したい場合のサイズ変更などに使えます。既にあるファイルに重ね書きする場合には、右上のファイル・リストの該当ファイルをダブル・クリックするだけです。こ

のときは確認の警告が出ますが、**OK**を押せば重ね書き、**CANCEL**を押せば中止します。

下半分が変換機能部分です。変換先のフォーマットは六種類

Ngraph: 図化ソフト Ngraph の旧フリーソフト版 Windows Classic Ver.5.45 に merge する場合があります。このソフトは Windows でも使えます。

P_TEX: いわゆる P_TEX のスタイル・ファイルを使って T_EX に利用する場合です。

ee_{pic}.sty: スタイル・ファイル **epic.sty** のコマンドを、TPIC special で置き換えて拡張した **ee_{pic}.sty** を用いて L^AT_EX に利用する場合に使います。網掛けなど Dviout for Windows で拡張した TPIC にも対応させるためには、アーカイブに入れておいた **ee_{pic}sup.sty** を利用します。いずれにしても、Dvi ドライバ依存の special コマンドで描画することになります。

ee_{pic}emu.sty: ドライバ依存が無いようにしたエミュレーション **ee_{pic}emu.sty** を使う場合です。ただし、この出力は見栄えがよくありませんので、代わりにもう一つのエミュレーション **ec_{le}epic.sty** を使うようにしてください。

curves.sty: ドライバ依存無しで図形を描くスタイル・ファイル **curves.sty** を用いる場合です。**ee_{pic}emu.sty** よりはましです。

Xbitmap: いわゆる X ビットマップ形式のファイルを出力します。T_EX 以外にウェブ・ページの図にも使えます。

という風になっています。拡張子を省いたファイル名を **Other types [.gra, .tex, .xbm]**: の右の窓に入力し、左の **EXPORT** を押せば変換・保存されます。既にあるファイルへの重ね書きは、ファイル・リストのファイル名をダブル・クリックです。各フォーマットごとの出力の違いは第9.4節に例示しました。

5.2 読み込み

保存してあったファイルからは **Load** で読み込みできます。図11のような窓が現れます。ファイ



図 11: ファイルへからの読み込み

ル・リストにあるファイル名を左クリックで選択した上で、窓左側の **LOAD** を押せば新規読み込み、**APPEND** であれば現在編集している図形群の上に重ね書きされます。入力が終了したら **DONE** で窓を閉じます。ファイル・リストにあるファイル名をダブル・クリックすると **LOAD** になります。

残念ながら、前述の「保存」すなわち Qfig original type のフォーマットで保存されたファイルしか読み込むことはできません。他のフォーマットに変換したものは読み込めませんので、もし何度も利用したり改訂する図形の場合には、前述の保存で必ず **SAVE** しておく必要があります。変換したものを逆変換する機能はありません。

5.3 データ・フォーマット

拡張子が **qfg** のデータも、エディタで直接編集できます。すなわち、座標値や属性を数値で記憶しているだけです。フォーマットさえ分かれば Qfig そのものを使わなくてもデータを作ることは（ま、誰もやらないけど）可能です。そういう意味で Draw 系の描画ソフトということになるのでしょうか。なお、データ・フォーマットはもう作者自身も覚えていませんが、古い PC98 用の Qfig のマニュアルには記載されています。

6 T_EX ファイルに変換してからの編集

6.1 図形番号の確認

網掛けや外形線の細かい設定、さらには、数学記号の入力などは Qfig 上ではなく、一旦 T_EX ファイルに出力したものを編集することで可能になります。どの図形を改訂するかは、T_EX ファイルのコメント行にある図形番号 '% object #' で判断します。各図形には、描画の順に番号が 0 から振られています。そ

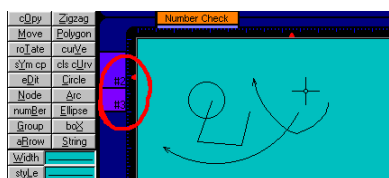


図 12: 図形番号の確認

の番号は `numBer` を押した後に、その該当図形のそば、具体的にはその図形の節点のできるだけそばで左クリックすることにより、図12に赤い楕円で囲った場所、つまり描画窓とコマンド・ボタン窓の間の小窓に表示されます。この例の場合は、現在のカーソルのそばにある図形は #2 であり、その次に近い図形が #3 であることを示しています。

6.2 数学用文字列

文字列に数学記号を入れたい場合には、Qfig 上ではなく、一旦 \TeX 出力してからの方が綺麗になります。 \TeX 出力ファイルの中の該当する文字列を探し出し、その文字列を $\$$ マークで挟んだ数学記号で置き換えます。長い文字列の場合には、出来上がりの位置が Qfig 上のそれと若干ずれますが、ほとんどの場合は気にならない程度でしょう。図13に π を表示してみました。ファイルの中身については次の節を参照してください。

6.3 網掛け・白塗りの重ね方と外形線消去

Qfig 上で網掛けができるのは箱だけでした。しかし、任意の閉じた図形に網掛けをしたり白で塗りつぶしたりするのは、`eepicsup.sty` のスタイル・ファイルで定義したコマンド (Dviout for Windows でサポートしている TPIC special) で可能になります。この改訂をする場合には、前節の `numBer` ボタンによって、改訂する図形の番号を調べておく と便利 です。例えば図13の場合は、左の図のような Qfig 上の入力を \TeX 出力し、そのファイルの

```
% object # 0 (line)
\path (55,100)(28,29).....
% object # 1 (line)
\path (96,83)(59,44)(108,26)(96,83)
% object # 2 (circle)
\put(153,77){\circle{44.272}}
% object # 3 (string)
\put(149,74){\rm p}
```

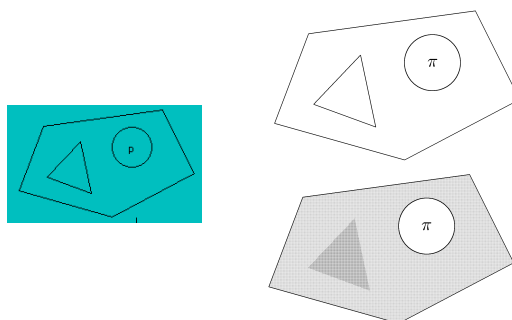


図 13: 網掛け・白塗り

となっていた部分を

```
% object # 0 (line)
\shade[0.2]\path (55,100)(28,29).....
% object # 1 (line)
\shadingwithoutoutline
\shade[0.7]\path (96,83)(59,44).....
\outlinedshading
% object # 2 (circle)
\whiten\put(153,77){\circle{44.272}}
% object # 3 (string)
\put(149,74){\rm $\pi$}}
```

としたものです。この例で分かるように、白で塗りつぶすような場合には、描画する順番も重要になります。予めそれを考えながら Qfig で作成しなければなりません。ちょっと面倒です。各コマンドの意味は

`\shade[0.2]`: 次の閉じた図形を 0.2 の濃度で網掛け。濃度は 0 から 1 までの数値。

`\shadingwithoutoutline`: 外形線無しにする。

`\outlinedshading`: 外形線有りに戻す。

`\whiten`: 白で塗りつぶす。

`\blacken`: 黒で塗りつぶす。`\shade[1]` と同じ。

となります。

7 描画補助

7.1 削除

図形ごとに削除するには `Kill` を用います。グループ化された図形を `Kill` すると、そのグループ全体の複数の図形が一度に削除されます。この削除は確認を求めませんので、注意して実行してください。元に戻す機能は (ユーザの多くから要望はありますが、面倒なので) 付いておりません。

すべての図形は `Zap!` で消去されます。確認画面が現れ、そのメッセージは 'Clear All Memory?' にな

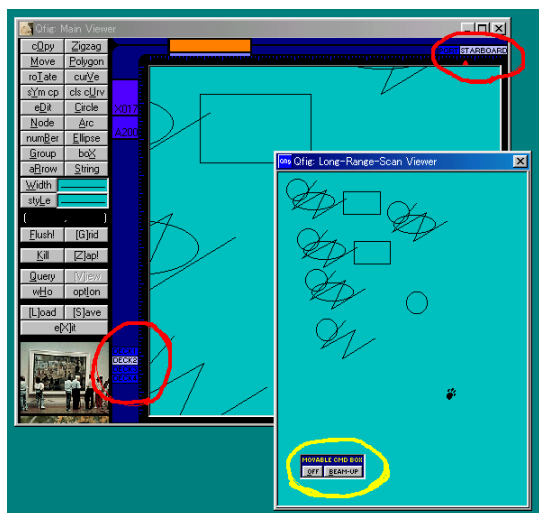


図 14: プレビューと描画窓の移動

ります．これへの対処は第2.3節で述べたのと同じです．

7.2 再描画

マウスではなくキーボードの矢印キーなどで描画した場合や、まだ残っているバグのせいで描画画面に黒い筋がたくさん残ってしまったような場合には、**Flush!**で再描画してください．多分、綺麗になるとと思います．

7.3 グリッド表示

マウス・カーソルを正確な位置に置くための補助として**Grid**を押せば、描画画面に等間隔の格子を表示することができます．この**Grid**を押すたびに格子のメッシュ幅が 2 mm, 5 mm, 1 cm と粗くなりますが、1 cm グリッド以上にはならず、グリッド無しに戻ります．

7.4 A4 サイズ・プレビューと描画窓の移動

描画画面は最大で A4 サイズ (800 × 600 のモニター画面でのフル表示で、描画画面が幅 21 cm、高さ 28.5 cm の領域にアクセスできます) で、このサイズでの図形プレビューは**View**によって別窓に表示されます (図14)．図中の黄色い楕円内の、プレビュー画面中にあるコマンド小窓は移動可能です．**MOVABLE CMD BOX** にマウスを合わせ、左クリックしたままドラッグしてもらえば、この窓内の任意

の位置に移動できます．この小窓の**OFF**を押すと、プレビュー画面は消えます．プレビューしたままで、元の描画画面に図形を加えても、そのままではプレビュー画面には反映されません．新しく加えた図形をプレビュー画面に表示するには、このコマンド小窓内の**BEAM-UP**を押します．

逆に言うと、描画画面の窓はこの A4 領域全体を一度にカバーすることができません．そのため、大きくなった図形を扱う場合には、A4 サイズの領域の中で描画画面の窓の方を上下左右に移動させなければなりません．図14中に赤い楕円で囲った二箇所にあるボタンでその移動が可能になります．描画画面左にある**DECK_n** (n 階: $n = 1, 2, 3, 4$) のボタンで上下移動ができます．1 階が最上部で、4 階が最下部です．また描画画面上部にある二つのボタン、**PORT side** (左舷) と **STARBOARD side** (右舷) が左右の移動に使われます．大きな図形を描いたり窓の外にコピーしたりして、このボタンの位置とプレビュー画面を比べてみれば、どうなっているかは分かると思います．

8 各種設定

8.1 オプション設定

オプション設定は**option**で行います．その中身の詳細については第2.1節に述べた通りです．

8.2 デフォルトの設定

各種設定で、この Qfig オリジナルのデフォルトに戻すには、ファイル 'Qfig.ini' を消去してしまうのが一番簡単です．

部分的な設定変更で**option**では行えないものがあるいくつかあり、それはファイル 'Qfig.ini' の直接編集で可能になります．各設定の意味は 'Qfig.ini' 内に記述されていますが、変更する可能性の高いものだけを補足しておきます．括弧内の数字はデフォルトです．

MaxObject (255): 図形の最大数 - 1 です．

MaxNode (31): 直・曲線の最大節点数 - 1 であり、一つの文字列の最長文字数です．

Arrow Head Proportion: 矢印の寸法補正関係です．

長さ: 矢そのものの長さを $0.25 \text{ mm} \times \{\text{arrowhead}(8) + \text{darrowhead}(1.5) \times \text{線の太さ}\}$ とする.

矢の傾き: 矢を付ける線に対する矢そのものの角度を $\pi/\text{narrowdirect}(9)$ とする.

矢の傾き補正 (円弧端): 円弧の場合の矢を若干内向きに $\pi/\text{narcarrowd}(30)$ だけ補正する.

9 その他

9.1 DOS 窓版 Qfig

マウス操作ではなかなか正確には位置決めができないので、例えばある程度正確に mm 単位で水平・鉛直の直線を引きたい場合には、この Windows 版は使いつらいでしょう。そんなときは IBM-PC 用 (DOS 窓版) のバージョン 1.00, 1.1a の二つがダウンロードして利用できます。私は v.1.00 の方をもっぱら使っています。文字列以外はデータに互換性がありますので、まずこれで大半を描画した上でファイル保存し、それを Windows 版に渡し、文字列を入れたり微調整をしたりしています。この DOS 窓版を使う場合、以下のことに注意してください。

- パソコンの `Num Lock` をはずす。これがロックされたままの状態では、ヘルプも表示できず、終了すらできません。
- ヘルプは `Ctrl` + `h` で表示されます。
- カーソルは矢印キーで動きます。ピッチは三種類変えられます。
- 使い方は Windows 版とほとんど同じです。マウスの時の左クリックをスペースキーに、右クリックをリターンキーに、ダブル・クリックを削除キーに置き換えるだけで、ほとんどの作業が同様にできるはずです。細かい点はヘルプを見てください。
- 描画などの機能選択はファンクション・キーで行います。v.1.00 では二つの機能が一つのキーに割り当てられており、`Esc` で切り替えます。
- 終了するには `Ctrl` + `q` を押します。

9.2 開発した経緯と環境

開発は、N88BASIC でお絵描きプログラムを作って遊んだのが最初です。その後、ワープロなるものがパソコンで使えるようにはなったものの、数式を入力するのが面倒だということが判明しました。幸い、東北大学の学内のネットワーク上での回りの環境にとっても影響を受け、 \LaTeX を使うようになりました。主に学内の電子掲示板のお世話になりました。

その頃はソフトが高価であったこともあり、 \LaTeX で書いた論文などへのグラフや絵をどうするかという問題にぶち当たりました。これまた幸いなことに、東北大学では Ngraph と Graph-P というものが広く流布されており、これを \LaTeX に組み込むことができるようになりました。あとは絵だけです。

そこで、先に作っていた N88BASIC のプログラムを Quick BASIC に移植し、 \LaTeX コマンドを含むファイルを出力するようにしました。その頃は某社の PC98 なるものがマジョリティであったこともあり、最初の Qfig は IBM 互換機では使えませんでした。幸い、某電気会社の外国人の方がこれを IBM 互換機用の Quick BASIC で作り直してくださいました。前節で紹介した v.1.00, 1.1a がそれです。特に v.1.1a は彼ら独自の改良が含まれており、 \LaTeX 2_ε への拡張もあり、ファンクション・キーの機能も一つあたり 3 機能になっています。

その後、私自身が Windows をようやく使うようになり、PC98 版のプログラムを Visual BASIC に移植したのが、現バージョンの Qfig です。

9.3 \LaTeX 2_ε を使っている場合の注意点

基本設定 `option` の **Font Size Expression** で、うっかり `\xiipt` を利用する設定をしたまま \TeX ファイル出力をすると、ファイルの文字列情報の部分に `\xpt`, `\xiipt` という \LaTeX 209 のコマンドが存在します。このまま \LaTeX 2_ε でコンパイルするとエラーになりますから、プリアンブル部に例えば

```
\let\xpt\null
\let\xiipt\null
```

という設定を入れてください。 \LaTeX 2_ε で直にフォント・サイズを指定する場合、どうしたらいいのかを私

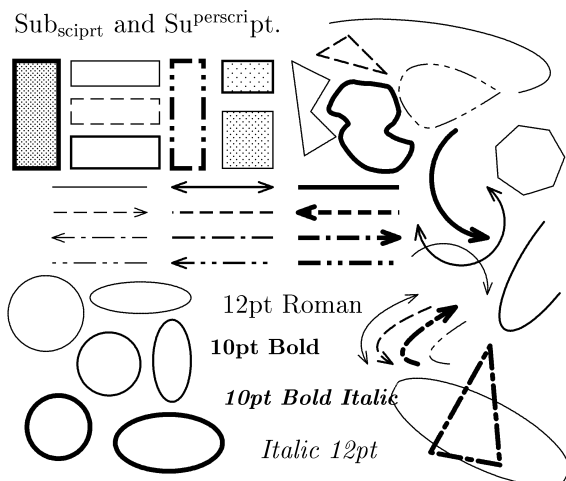


図 15: PCTEX フォーマット

が理解していないので、今は 10 pt なら何もせず、12 pt 文字の場合は `\large` が出力されるようになっていきます。このあたりについては、各自で工夫してください。

9.4 出力フォーマット間の相違

第5.1節で説明した六種類の変換の結果を、アーカイブに入れた例 `'qfig3ple.qfg'` のデータを用いて比較してみましょう。

図15: いわゆる PCTEX を用いた場合です。Dvi ドライバには依存しません。コンピュータの処理速度がかなり速くなったので、以前は重くて敬遠されていた PCTEX も短時間で処理ができるようになっており、Dvi ドライバ依存を嫌う場合には有益かもしれません。

図16: スタイル・ファイル `curves.sty` を用いた例です。一部の（凹な）曲線がスムーズではありませんが、これは常に生じるとは限りません。曲線を描くときに節点同士をあまり離さないようにすれば回避できると思います。これも Dvi ドライバ非依存です。

図17: スタイル・ファイル `eepicsup.sty` を利用したものです。これが一番のお勧めですが、Dvi ドライバが TPIC special を理解できる場合にのみ使えます。ドライバ依存です。回転や網掛けについての拡張 TPIC special が、現在よく

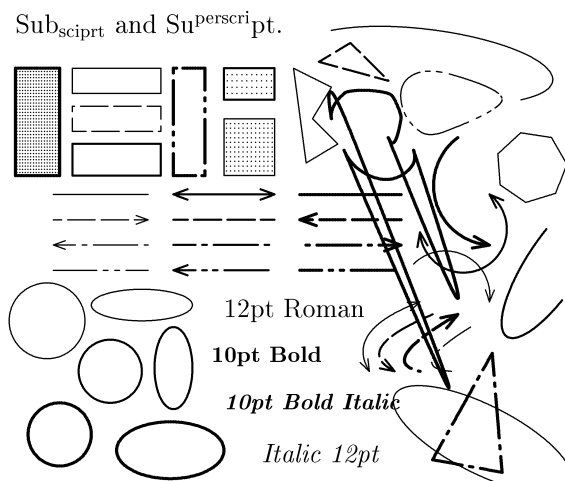


図 16: `curves.sty` を用いた出力

用いられているドライバの Dviout for Windows には組み込まれています。このような拡張を用いないのであれば `eepicsup.sty` は必要無く、`eepic.sty` で処理できます。

図18: `eepic.sty` のエミュレーションをするスタイル・ファイル `eepicemu.sty` を用いた場合の例で、矢印が少しずれていますが原因不明。また Qfig のバグで、網掛けのコマンドを出力してしまっています。eepicemu.sty を用いる場合には、プリアンブルに

```
\let\shade\null
```

という行を入れてください。でも、ずれは直らないので使わない方がいいでしょう。その代わりに次のスタイル・ファイルを使ってください。

図19: Dvi ウェアに依存しない出力をするためのもう一つのエミュレーションの `ecleepic.sty` を用いた場合です。こちらは `\shade` は無視されるだけですから、入れたままでもうまくいきます。もし Dvi ウェア非依存のファイルにしたい場合は、この `ecleepic.sty` がベストかもしれません。

図20: Ngraph の Gra ファイル形式で出力し、それを図化ソフト Ngraph に、フル・サイズで merge したものです。実験や数値解析の結果を Ngraph

Sub_{script} and Su^{perscri}pt.

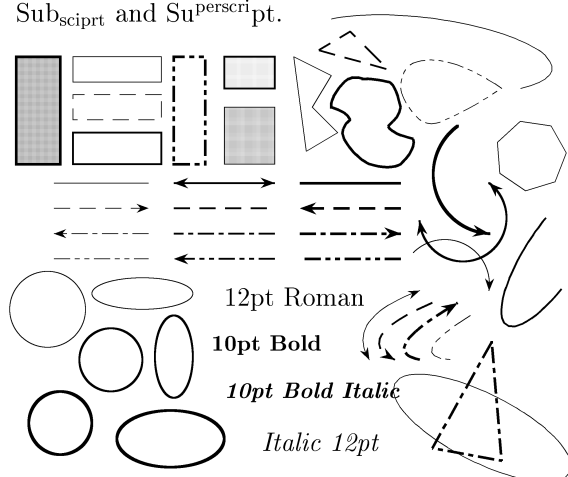


図 17: eepicsup.styを用いた出力 (推奨)

Sub_{script} and Su^{perscri}pt.

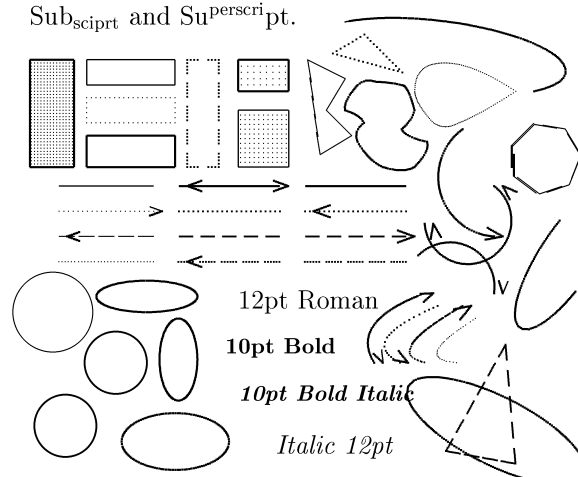


図 18: eepic.styのエミュレーション

Sub_{script} and Su^{perscri}pt.

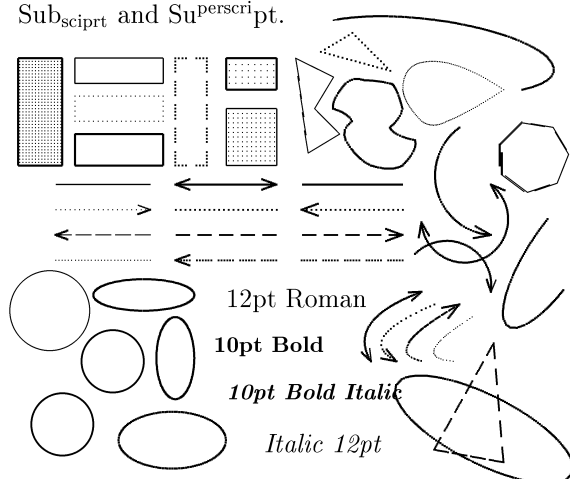


図 19: eepic.styのもう一つのエミュレーション : ecleepic.styの利用

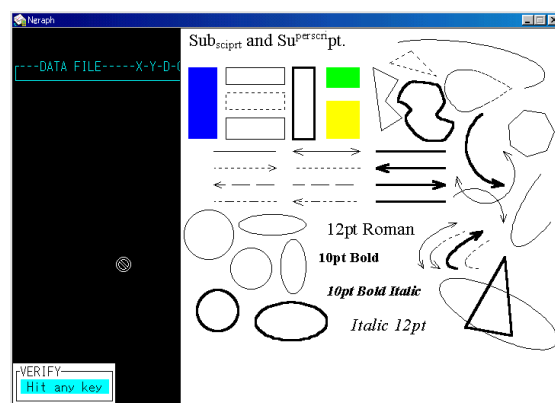


図 20: Ngraphに merge した場合の画面

LEAST SQUARE FIT

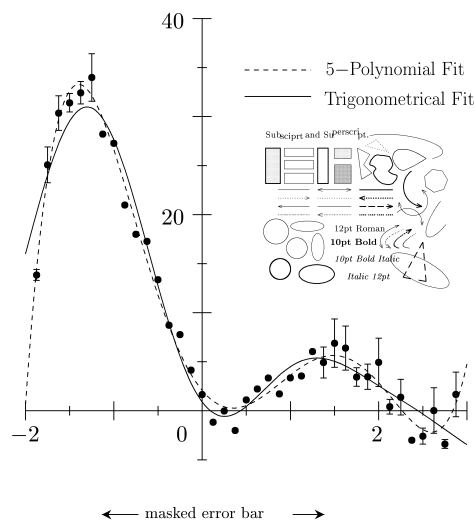
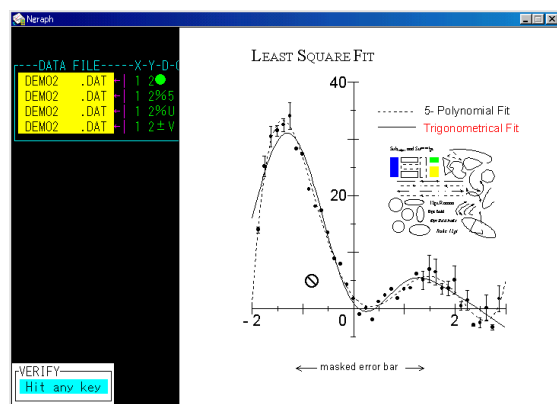
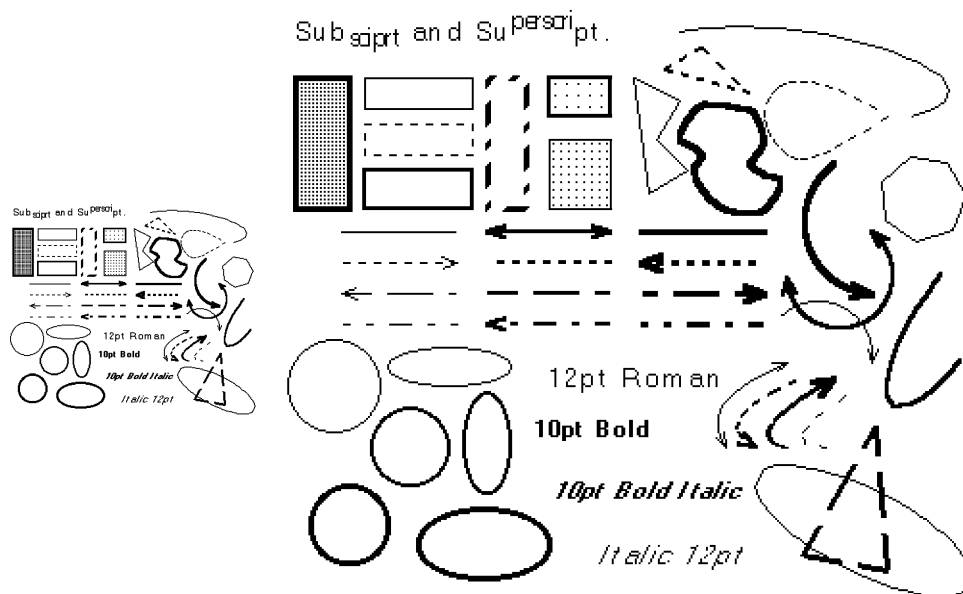


図 21: Qfigで作った絵をNgraphに merge し (左図) , それをさらに \LaTeX (eepicsup.sty) 化したもの (右図)



X ビットマップのデータ 3ple.xbm を読み込む.

`\bitmapinput{280}{3ple}` の場合 (左) と `\bitmapinput{100}{3ple}` (右)

図 22: Xbm ファイルを 'bitmap.sty' で利用する

でグラフにしたとき、その図中に簡単な解説図をこれで付加できます。ただし、merge する際に読み込み場所を適切に設定する必要があります。この例の図を Ngraph のデモ図に merge したものを図21に示しておきました。

図22: 変換リストの最後にある X ビットマップにしたファイルを、この形式の図ファイルを \LaTeX に組み入れるスタイル・ファイル `bitmap.sty` を用いて出力した例です。これに対し図23は、そのファイルを単純にブラウザで表示したものです。

以上の各フォーマットの出力は、アーカイブに含めたサンプルの \LaTeX ソース・ファイルを用いてコンパイルできます。ただし、サンプルは \LaTeX 209 用なので、 \LaTeX 2_ε でコンパイルする場合には、最初の行を `\documentclass` にするのはもちろん、オプションのスタイル・ファイルは `\usepackage` で読み込んでください。

9.5 Ngraph との連携

なお、図21のように、旧版 Ngraph のグラフを \LaTeX ファイル化するには私のウェブ・ページに

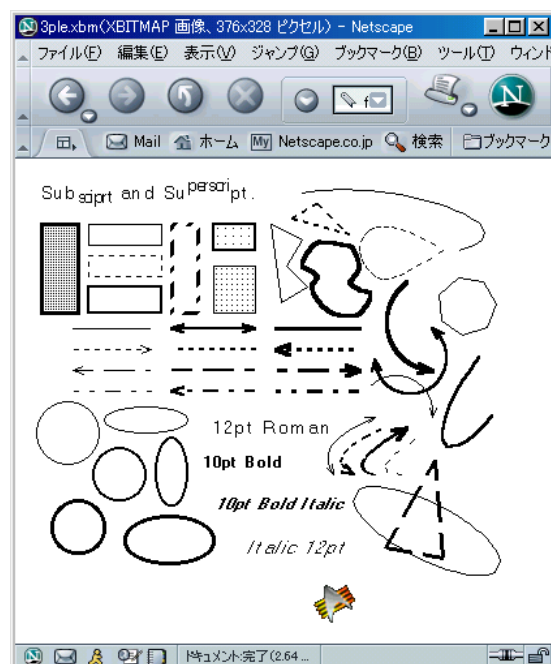


図 23: ブラウザで見た X ビットマップファイル

ある ibm-pc (in DOS window) 用の 'Ngraph.Gra to Eepic.TeX in us mode' を使ってください。Ngraph には、シェア・ウェアの Windows 版もあり、その出力を \TeX 化するソフトもあります。それについては Ngraph のページをご覧ください。

9.6 PDF ファイルへの変換

Dviout for Windows を用いるのであれば、図17に示したように、スタイル・ファイル eepicsup.sty を利用したものがお勧めです。例えば最終的に pdf ファイルにする場合であっても、文字回転の TPIC special 以外であれば、図も \TeX ファイルのまま dvipdfm で変換できます。

しかし国際論文集などへの投稿時に、この TPIC special の \TeX データをそのまま送付するのには不安があります。やはり PostScript ファイルにした方がいいかもしれません。そのときは、この Dviout for Windows の画像出力を利用すれば、他の画像変換ソフトとの連携で PostScript 画像ファイルに変換でき、dvipdfm で pdf ファイルに組み込むことが可能になります。実際、このマニュアルそのものがそういう風にして作られています。私の場合には具体的に

1. 普通に eepicsup.sty を用いて絵を dvi ファイルにする。例えばアーカイブにあるサンプル・ソースを使う。
2. Dviout for Windows で表示し、**Option Setup Parameters Display** で、**Start** の解像度を 1 に設定し、表示した絵を、ファイル出力の **Save as image** で例えば bmp ファイルに出力する。Dviout for Windows を画像変換に使ってしまっています。ごめんなさい。
3. 適当な画像処理ソフト、例えば Infan View で不要な空白をトリムし、それを例えば Gimp などを用いて PostScript に変換する。

といったステップを用いています。こうすれば、図中の数学文字などの文字を本文の \TeX の文字と同じフォントで表示することができます。第2ステップの解像度を 1 にしなかったり、第3ステップで bmp ファイルを縮小したりすると、最後の PostScript ファイルの解像度が悪くなることがありますので、注意してください。昨今は、ハード・ディスク容量も十分にあり、また保存も CD や DVD を用いることが普通になってきていますから、あまりファイル・サイズには気にせず、高解像度のまま保存しておく方がいいと思います。

10 この文書について

10.1 文書の絵が不鮮明

この pdf ファイルをカラー・プリンタに出力した場合には、文章脇に置いた図中の小さいボタンの文字などが目を凝らせばどうにか読めるかもしれませんが、しかし、白黒でプリンタ出力したものは、ほとんど読めないかもしれません。図中の細かい文字を確認したい場合には、印刷せずに Acrobat Reader の画面表示のまま拡大するといい (300% がベスト) でしょう。

10.2 Web リンク

この pdf ファイルには、インターネットの URL が埋め込まれています。例えば次の括弧書きの文字列、「私が集めたスタイル・ファイルのページ」のように、水色の四角で囲まれた部分をクリックすると、ブラウザにそのページが表示されます。確実に表示させるためには、Acrobat Reader の「環境設定」「Web リンク」「Web ブラウザ・アプリケーション」に、普段使っているブラウザを正しく設定しておいた上で、このリンク部分をクリックする前にブラウザを立ち上げておいてください。そうすれば、ページが確実に表示されます。