

HandyView

- ・ 図面ファイルの代わりにイメージファイルが読込めるようになりました。イメージデータの上にマーカーデータを上書きできます。
- ・ 図面ファイルを読込みイメージデータとして保存できます。
- ・ マークアップでフリーハンドの落書きが描けるようになりました。
- ・ AutoCAD2004 図面ファイルの読込みに対応しました。

HandyView は、インターネットエクスプローラで動作する2次元CAD図面ビューワです。インターネットエクスプローラに、Plug-InできるOCX版と、単体動作するEXE版とが、標準装備されています。遠方のサーバーにある図面ファイルも、特別な操作をすることなしにビューイングすることが可能です。要素の長さ/半径、要素間/2点間距離と座標値がナビゲーションでき、周長/面積の計測が行えます。

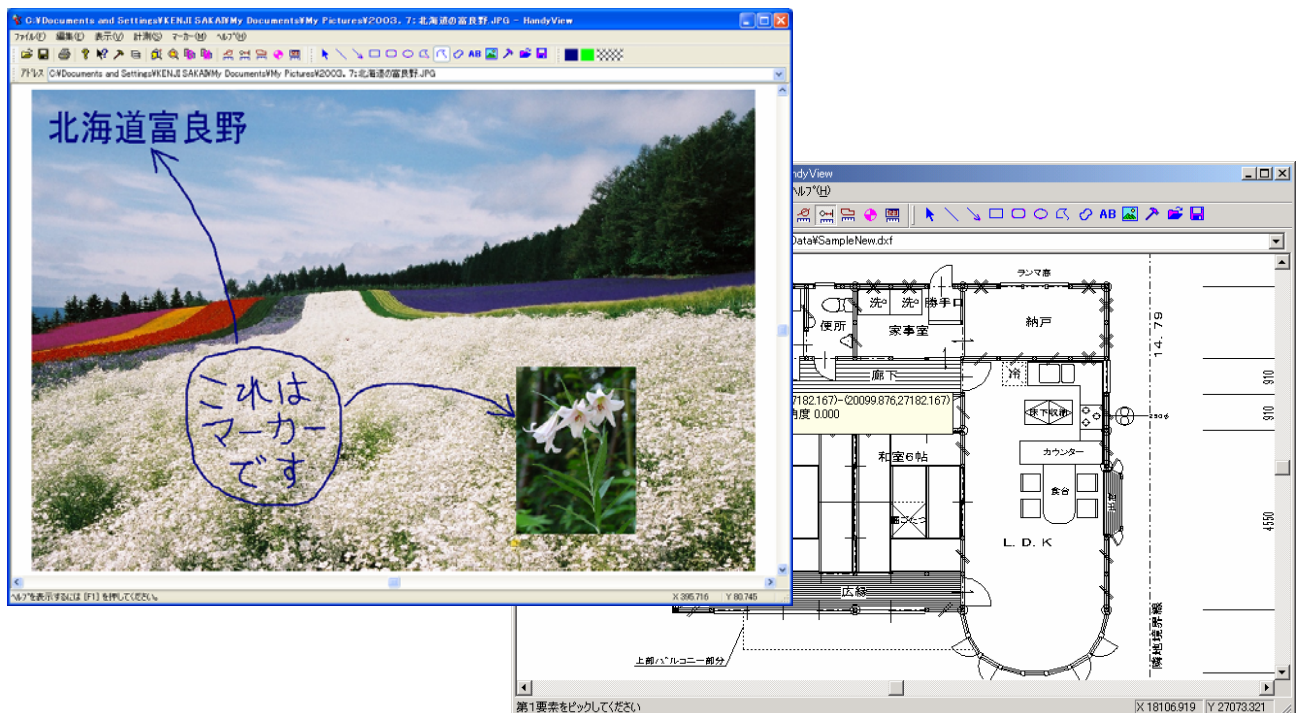
ActiveX Control に対応していますので、HandyViewをHTMLの中に埋め込んだり、VB, VC, Delphiやオフィスシリーズへ組み込むことができ、拡大/縮小/要素計測が可能です。また、VBScript/VBAにより、図面から情報を取り出すことも可能です。これらの開発ツールを利用し、カスタマイズすることによって、全く新しいシステムを作成したり、PDMなどの図面管理システムを構築することができます。

対応図面ファイル : AutoCAD(DWG, DXF), SXF(P21, SFC), JW_CAD(JWC, JWW), IGES(2D), HP-GL, CADSUPER, EASYDRAW, HandyCAD

対応イメージファイル : BMP, JPEG, TIFF-G4

動作環境 : 日本語 Windows 98(SE)/Me/2000/XP(32bit)

英語リソースに対応(日本語Windows以外では、英語で動作します)



株式会社システムアイ

<<体験版をダウンロードして試用できます。>>

〒532-0011 大阪市淀川区西中島 3-10-12-3F

TEL:06-6390-8808

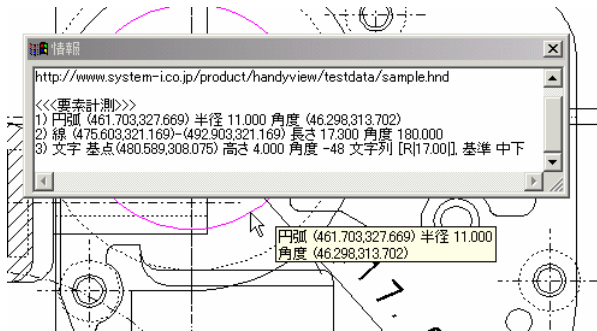
FAX:06-6390-8809

<http://www.system-i.co.jp>

E-mail: system-i@system-i.co.jp

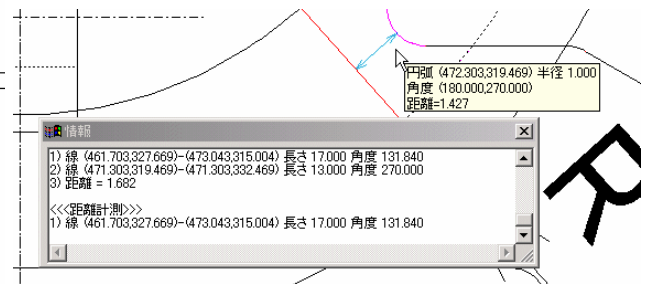
●要素計測

計測したい要素にマウスを近づけると、認識されている要素が紫色表示され、ナビゲーションチップ内に詳細な情報が表示されます。
マウスの移動時に、コントロール(CTRL)キーを押していると、要素ではなく、要素端点/交点の選択となります。マウスの左ボタンをクリックすると、要素情報は「情報ウィンドウ」にも書き出されます。



●距離計測

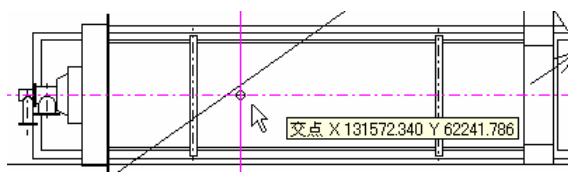
距離計測には、2つの要素または端点/交点が必要です。最初の要素が確定された後、マウスを動かすとマウス近傍要素が仮選択され(紫色で表示)、最初の要素との最短点となる位置に水色で矢線が表示されます。
また、ナビゲーションチップには要素情報に加え、最短距離が表示されます。



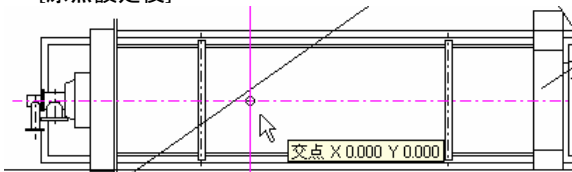
●計測原点

要素端点/交点で小さな丸印が表示されます。この丸印が表示されているときに、マウスの左ボタンをクリックすると、その座標が新しい原点として登録され、以降の「要素計測」「距離計測」等の基点として解釈されます。

[原点設定前]

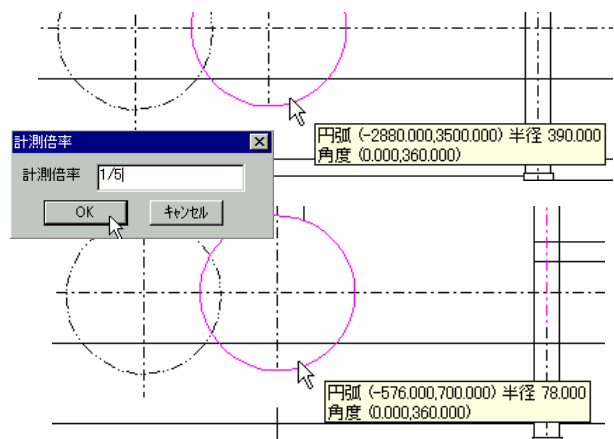


[原点設定後]



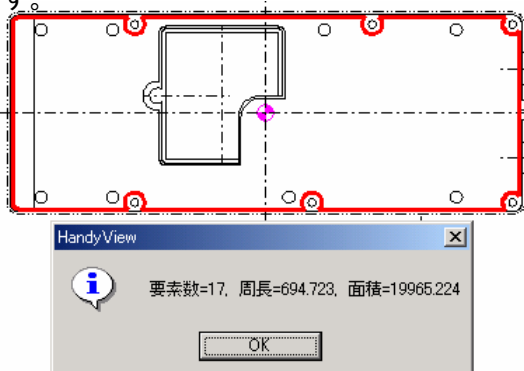
●計測倍率

図面を読み込んだ直後は、図面は原寸で描かれていると解釈します。縮尺付きの図面で、ナビゲーションの値と実寸が合わない場合に計測倍率を指定してください。



●周長/面積計測

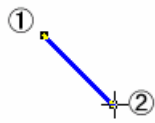
周長/面積の計測が行えます。単要素/連続要素の指定ができるので、連続していない要素や交差している要素の周長/面積も計測できます。



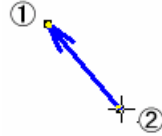
●マークアップ機能

マークアップ機能により、表示したCADデータ上に、コメントや引き出し線、長方形、楕円、角丸矩形、イメージデータ(JPEG)等を記入することができます。図面の修正個所の指示等に利用すると便利です。

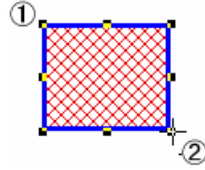
「線分作画」



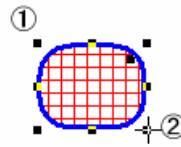
「矢印線分」



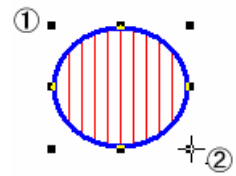
「矩形作画」



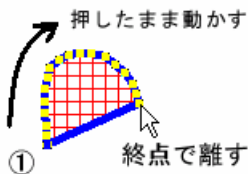
「角丸矩形」



「楕円作画」



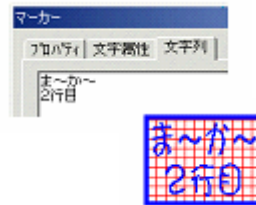
「ポリゴン作画」



「ポリライン作画」

マーカー

「文字列作画」

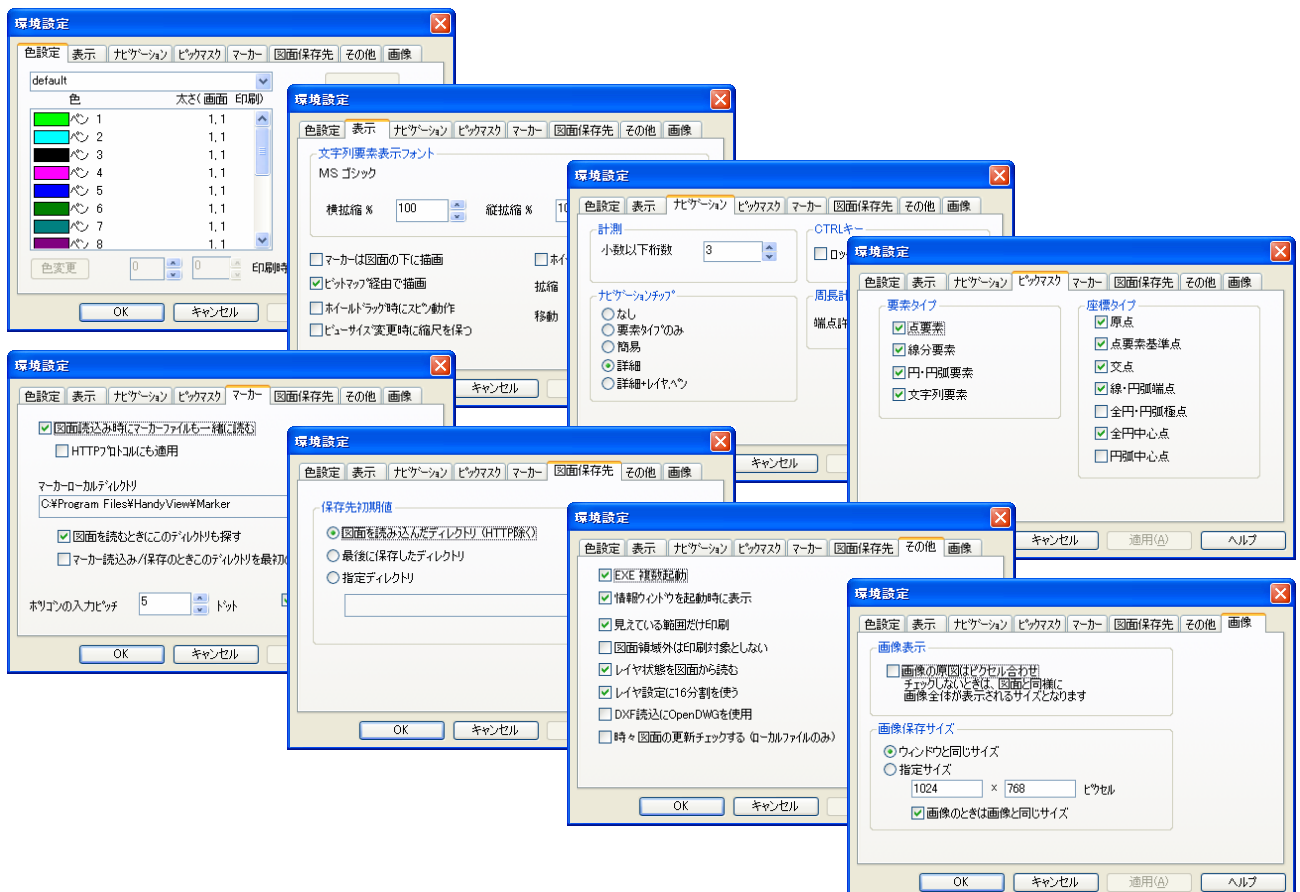


「イメージデータ」



●環境設定

色設定ページでは、ペン番号に対応する色の設定、要素認識色、背景色の設定が行えます。また、色の組み合わせに対して名前を付けて保存することができます。表示ページでは、文字フォントや拡大/移動の倍率などの指定ができます。ナビゲーションページでは、「要素計測」等の際のナビゲーションに関する設定を行います。ピクマスクページでは、要素認識時の要素タイプ/座標タイプの設定を行います。マーカーページでは、マーカーファイルのディレクトリ等の設定を行います。



●HandyView ActiveX Control の特徴

ワードやエクセルのファイルに埋め込んで、CAD図面を部品表や文章の一部として管理出来ます。
VC++やVBでダイアログボックスを作成するとき、コントロールとして埋め込むことによって、図形の入り
わかり易いプログラムを作成できます。
メタファイル貼り付けやOLE貼り付けと違い、貼り付いたままで、拡大/縮小、要素計測などが行えます。
メソッド/イベントを使用して、貼り付けたコントロールを制御できます。開発言語としては、IE組込みのVBScript、
Visual C++, VisualBasic, マイクロソフト社オフィスファミリーのVisualBasic for Applicationなどを始め、
Delphiやその他の言語でも、COMオブジェクトが扱えるものなら全て可能です。

●HTML埋め込みサンプル

HandyViewActiveX Control のメソッド/イベント処理のサンプルです。
VBScriptを使用して、HTMLページに貼り付けた、HandyViewコントロールを制御出来ます。
このActiveX を正常に動作させる為には、HandyViewのOCX版がインストールされていて
ブラウザ上でVBScriptが動作することが必要です。

```
<head>
<title>HTML 埋め込み サンプル</title>
</head>
<Body bgcolor="#000000" text="#FFFFFF" link="#00FFFF"
link="#FF00FF">
<CENTER>
<H1>HTML 埋め込み サンプル</H1>
<TABLE width="300" border="1" bgcolor="#ff
<TBODY>
<TR>
<TD align="center">
<OBJECT classid="CLSID:1839A2B6-A0
height="384" width="512" ic
<PARAM name="URL" value="../sample
</OBJECT>
</TD>
</TR>
</TBODY>
</TABLE>
</CENTER>

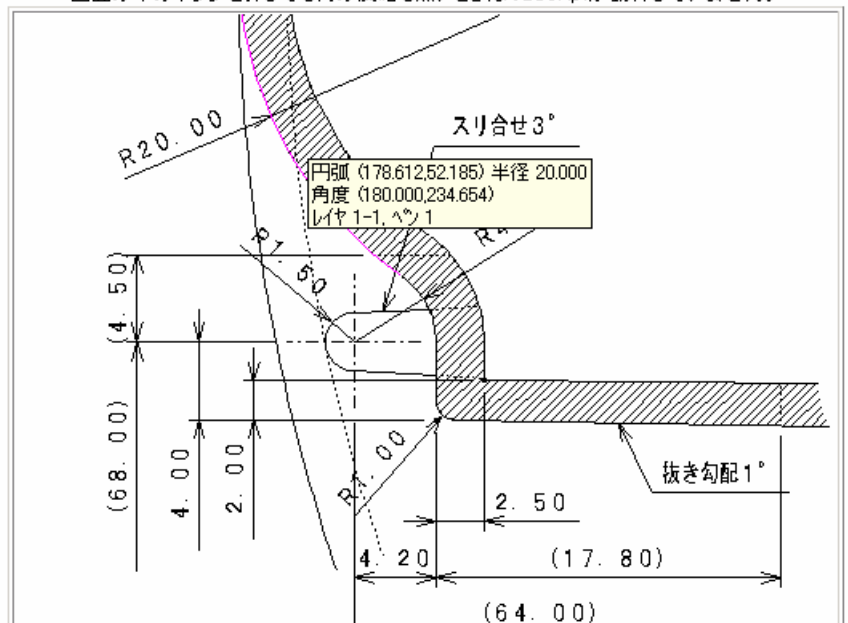
<FORM>
作画 →
<input type="button" value="点" onClick=
<input type="button" value="線分" onClic
<input type="button" value="円弧" onClic
<input type="button" value="文字列" onClic
</FORM>

<SCRIPT language="VBScript">
' ===== グローバル変数 =====
Dim nCommand, nCommandSeq
Dim oldX, oldY

' ===== 線分コマンド =====
Sub OnLButtonDown_Line
nSts = HandyView.vGetNaviPoint( x, y )
If 0<nCommandSeq then
HandyView.CurrentPen = 1
HandyView.SetCurrentLStyle 3, 0, 1
HandyView.AddLine oldX, oldY, x, y
Else
nCommandSeq = nCommandSeq + 1
End If
oldX = x: oldY = y
End Sub
```

HTML埋め込み サンプル

HandyView Active-X コントロールのメソッド/イベント処理のサンプルです。
VBScriptを使用して、HTMLページに貼り付けたHandyView コントロールを制御出来ます。
下のウィンドウに図面が表示されているときは、実際に動作しますので
マウスの右クリックなど、いろいろ行ってください。
図面の下のボタンを押しても何の反応も無いときはVBScriptが動作していません。



図面選択 →

色設定 →

作画 →

計測他 →

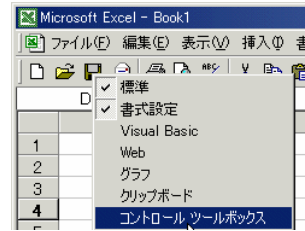
組み込みコマンドの実行 →

このページを正常に動作させる為には HandyViewのOCX版(Ver 1.10以降)がインストールされていて
ブラウザ上で VBScriptが動作することが必要です。

●エクセルのシートへHandyViewコントロールを貼り付ける手順

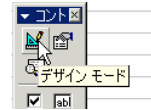
1. 「コントロールツールボックス」を出す。

エクセルのツールバーの上で右クリックし、ポップアップメニューから「コントロールツールボックス」を選択してツールボックスを出します。



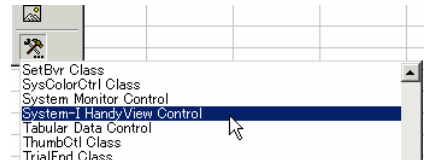
2. デザインモードに移行します。

三角定規アイコンをクリックし、デザインモードに入ります。アイコンがへこんだ状態になります。



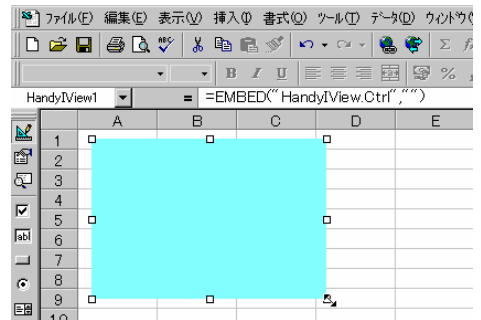
3. 「コントロールの選択」で、「HandyViewControl」を選びます。

挿入するコントロールを選択します。



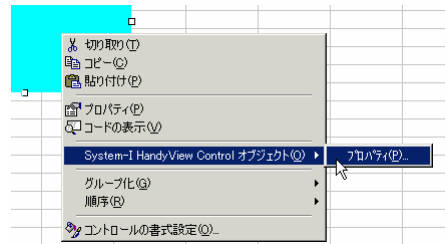
4. エクセルのシート内で、配置する対角点を指定して、コントロールを配置します。

シート内で、配置する左上点をクリックし、マウスボタンを離さないまま右下点まで移動して、マウスのボタンを離します。



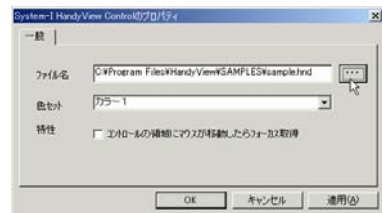
5. HandyViewControlのプロパティを設定する。

配置したコントロール上で、右クリックし、ポップアップメニューから「HandyViewControl」 - 「プロパティ」を選択します。



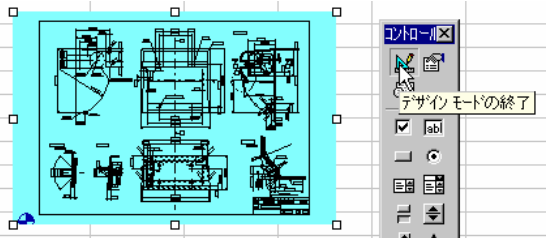
6. 「ファイル名」の右の方にあるボタンを押して、図面ファイルを選択してください。

ファイルブラウザダイアログが現れます。



7. 図面が表示されたら、デザインモードを終了します。

もう一度、三角定規アイコンをクリックし、デザインモードから抜けます。アイコンがへこんだ状態が解除されます。



8. 配置位置や大きさを変更するには、もう一度デザインモードに戻り、コントロールをクリックしてください。

デザインモードを終了すると、コントロールはアクティブになり、図面の拡大／縮小、要素の参照などが、エクセルのシートに貼り付いたままで行えます。

OLEの埋め込みとは違い、エクセルのシートのセルと図面の間が自由に行き来できます。

エクセルのファイルを保存すると、コントロールが貼り付いたまま保存されます。

次回読み込み時には、コントロールの状態も一緒に読み込まれます。

● エクセルのマクロと組み合わせた例（図面から文字列取り込み）

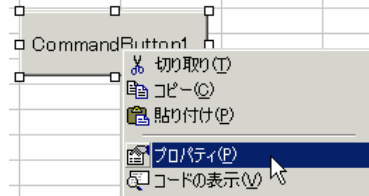
1. シート上にボタンを貼り付ける。

エクセルのシート上に、マクロを起動する為のボタンを貼り付けます。
デザインモードで「コマンドボタン」を選択し、配置する対角を指定します。



2. ボタンに書いてある文字列を変更する(1)。

貼り付けたボタンの上で右クリックし、「プロパティ」を選択します。



3. ボタンに書いてある文字列を変更する(2)。

「Caption」の値を書き換えます。



4. ボタンを押したときの動作を定義する。

ボタンをダブルクリックすると、Visual Basic Editor が立ち上がります。



5. マクロの記述。

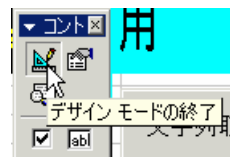
右のように、マクロの内容を記述します

※HandyView ActiveX Controlのオブジェクト名は通常、HandyIView1になりますが、何回も試行していると数字の部分が変わります

```
Dim nRow, nCol 'セルの位置
'----- 「文字列取り込み」ボタン押下時 -----
Private Sub CommandButton1_Click()
    HandyIView1.SetUserCommandMode 'イベント受け取りモード
    HandyIView1.SetNaviMode 1 '要素ナビ
    nRow = 2: nCol = 6 'セルの位置
End Sub
'----- マウス左ボタン押下時 -----
Private Sub HandyIView1_OnLButtonDown()
    nElemNo = HandyIView1.GetNaviElem() 'ナビで求めた要素
    If nElemNo <= 0 Then Exit Sub
    nElemType = HandyIView1.GetElemType(nElemNo) '要素タイプ
    If nElemType <> 8 Then Exit Sub '文字列以外は無視
    bSts = HandyIView1.vGetTextData(nElemNo, h, strText)
    If False = bSts Then Exit Sub 'データ取り込み失敗
    Set Cell = Cells(nRow, nCol) '書き込み先セルを指定
    Cell.Value = strText '文字列書き込み
    nRow = nRow + 1
End Sub
```

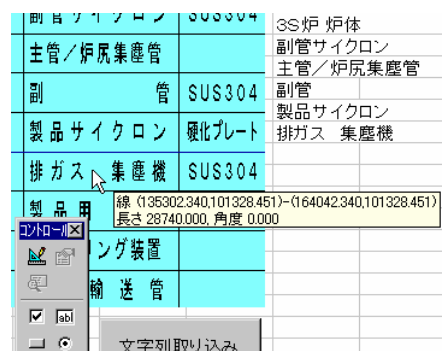
6. デザインモードを終了します。

三角定規アイコンをクリックして、デザインモードを終了します



7. 「文字列取り込み」を実行します。

「文字列取り込み」ボタンを押して実行します。
文字列要素をピックする度に、エクセルのセルに取り込まれていきます。



●VisualBasicのフォームでの利用手順

1. プロジェクトの種類として標準EXEを選びます。

VB6を立ち上げると「新しいプロジェクト」ダイアログが現れます。「標準EXE」を選択します。

2. 標準のコントロールを貼り付けます。

(5. に配置例があります)

コントロール一覧から、“DriveListBox”を選択し、フォーム上に配置します。

コントロール一覧から、“DirListBox”を選択し、フォーム上に配置します。

コントロール一覧から、“FileListBox”を選択し、フォーム上に配置します。

3. HandyViewをコンポーネントリストから選択します。

コントロール一覧のバーを右クリックし、ポップアップメニューから「コンポーネント」を選択します。

ダイアログボックスが現れたら、「参照(B)...」ボタンを押し、HandyViewをインストールしたディレクトリから、HandyView.ocxファイルを選択します。

4. HandyViewコントロールを配置します。

3. の手順により、コントロール一覧にHandyViewのアイコンが現れます。

このアイコンを押し、フォーム上に配置してください。

5. 配置終了状態はこんな感じです。

標準コントロールから3つと、HandyViewコントロールを配置します。

6. コードのスケルトンを作成します。

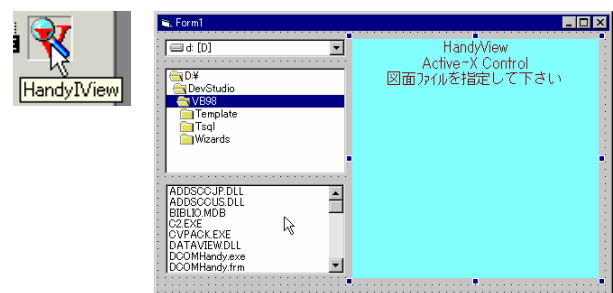
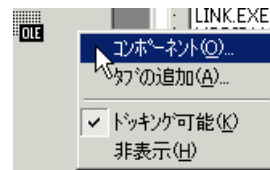
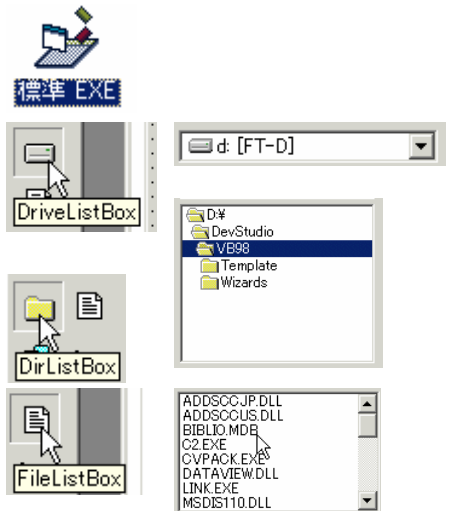
「ドライブ選択」「ディレクトリ選択」「ファイル選択」それぞれをダブルクリックしてください。右のコードが現れます。

7. コードを完成させます。

コードを右のように書き換えます。下から2行目は初期ディレクトリの指定です。ご利用の環境に合わせて修正してください。

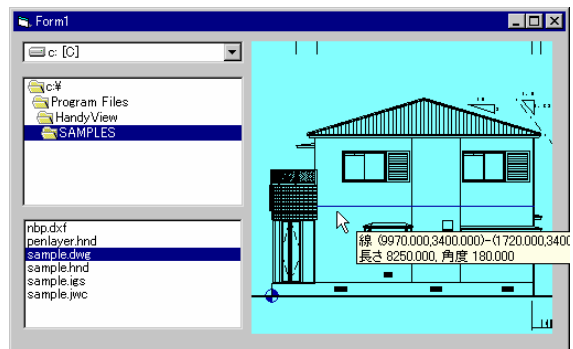
8. 実行

これで、完成です。開始ボタンを押して実行してください。左下のファイルを選択することで、図面が表示されます。



```
Private Sub Drive1_Change()
End Sub
Private Sub Dir1_Change()
End Sub
Private Sub File1_Click()
End Sub
```

```
Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub
Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub
Private Sub File1_Click()
    strPath = File1.Path & "¥" & File1.FileName
    HandyView1.URL = strPath
End Sub
Private Sub Form_Load()
    Dir1.Path="C:¥ProgramFiles¥HandyView¥Samples"
End Sub
```



●HandyView ActiveX Control (API) の説明

1. プロパティ一覧: プロパティとは一般の言語でいう変数に相当します。

変数型	変数名	説明
BSTR	URL	図面ファイルの URL または パス
long	CurrentPen	図形要素を書き込むときのペン番号
BSTR	ColorSet	色の組み合わせ名称
BOOL	AutoFocus	自動フォーカス取得

2. メソッド一覧: メソッドとは一般の言語でいう手続き/関数に相当します。

戻り型	メソッド名	説明
void	AbortRunningCommands	今、実行しているコマンドをキャンセルする
long	AddArc	円・円弧要素を書き込む
long	AddLine	線分要素を書き込む
long	AddPoint	点要素を書き込む
BOOL	AddPopupMenu	メニューの追加
long	AddText	文字列要素を書き込む
long	DeleteTemporaryEntities	追加した要素を削除して図面の初期状態に戻す
BOOL	EnableEvents	イベントの許可/禁止
BOOL	EnableInternalCommand	内部コマンドの許可/禁止
BOOL	GetArcData	円・円弧要素の座標データを取り出す
BOOL	GetElemLayer	図形要素のレイヤ番号を返す
BOOL	GetElemLStyle	線・円／円弧要素の線種を取り出す
BOOL	GetElemPenno	図形要素のペン番号を返す
long	GetElemType	図形要素の要素タイプを返す
long	GetEntityCount	レイヤに書かれている要素数を返す
long	GetFireCoord	マウスイベント発生時の座標を返す
BSTR	GetLayerName	レイヤ名称を返す
BOOL	GetLineData	線分要素の座標データを取り出す
long	GetNaviElem	ナビゲーションで求めた図形要素を返す
int	GetNaviPoint	ナビゲーションで求めた座標を返す
BOOL	GetPointData	点要素から座標データを取り出す
BOOL	GetTextData	文字列要素から文字列を取り出す
long	GetUserMenuID OnUserMenu	発生時のメニュー番号を返す
long	GetVersionNo	コントロールのバージョン番号 x 100 を返す
BOOL	GetViewMinMax	現在設定されている表示域を返す
void	Invalidate	コントロール表示の無効化
void	RunInternalCommand	内部組み込みコマンドの実行
long	SaveAs	指定のファイル形式で保存
void	SetCurrentCharAttr	文字要素を書き込むときの属性を指定
long	SetCurrentLayer	要素を追加するときのレイヤ番号を指定
void	SetCurrentLStyle	線分/円弧要素を書き込むときの線種を指定
long	SetEmpty	図面を空にする
void	SetInfoText	“情報ウィンドウ” に指定の文字列を書く
long	SetLayerStatus	レイヤの表示状態設定
BOOL	SetMeasureScale	計測縮尺を指定
long	SetNaviDigits	ナビゲーションチップで座標値を表示するときの小数以下桁数を指定
void	SetNaviMode	ナビゲーションモードの設定
long	SetNaviTipLevel	ナビゲーションチップ表示の詳細度を指定
long	SetPenColor	ペン番号に対応する色を設定
void	SetUserCommandMode	コンテナにイベントを送り込むモードにする
BOOL	SetViewMinMax	表示域を設定
void	WindowFit	表示領域のフィット

3. イベント一覧: イベントとは、ある事象が起こったことの通知です。

イベント名	通知	説明
OnLButtonDown	常に	マウスの左ボタンが押されたとき
OnMouseMove	許可時	マウスの移動
OnNaviElem	許可時	ナビゲーションで要素が求めたとき
OnNaviPoint	許可時	ナビゲーションで点が求めたとき
OnRButtonDown	許可時	マウスの右ボタンが押されたとき
OnTerminateUserCommand	常に	コンテナ側のコマンドが中断されるというのを通知
OnUserMenu	常に	40000 ~ 40009 の ID のメニューが選択されたとき