

KDラスターコンバート

DWF入力コンバータ

DWFXMLX



Kernel Computer System
カーネルコンピュータシステム株式会社

本社：パッケージ販売部
〒221-0056
横浜市神奈川区金港町 6-3 横浜金港町ビル
TEL：045-442-0500 FAX：045-442-0501
URL：<http://www.kernelcomputer.co.jp>

特長

- DWF, DWFx から各種ラスターデータ、ベクターデータへの変換を行います。
- コマンドラインから入出力ファイル名をキーインする事で実行されます。(標準入出力が利用可能)
- 実行時各入出力フォーマットの属性パラメータやオプションパラメータを設定できます。
- スケール、回転、原点移動、センタリングなどの処理が可能。
- 指定ファイル又はフォルダ毎の一括変換が可能。

対応 OS

Windows 2000 / Windows XP / Windows Server 2003

標準価格

製品名	製品仕様	標準価格
DWF2RS	A0 対応版 (A0/600DPI) モノクロ版 (JIS 第 1 水準漢字対応) (最大ドット数 32000)	35 万円 (税抜き)
DWF2RSC	A0 対応版 (A0/600DPI) モノクロ・カラー併用版 (JIS 第 1 水準漢字対応) (最大ドット数 32000)	50 万円 (税抜き)
DWFTODWG	DWG	20 万円 (税抜き)
DWFTODXF	DXF	20 万円 (税抜き)
DWFTOH2	HP-GL/2	20 万円 (税抜き)
DWFTOPST	PostScript PDF	20 万円 (税抜き)

操作方法

DWFXXX は、コマンドラインアプリケーションです。通常のコマンドと同じようにコマンドラインから実行したり、バッチファイルやシェルスクリプトに組み込む事ができます。Windows の場合は、「コマンドプロンプト」で DOS ウィンドウを開いて、その中で実行します。変換時にオプションを指定したり、属性データ変換パラメータファイルの内容を変更したりする事で、様々の変換処理が可能となります。

DWF からラスタデータに変換するには、次のような形式でコマンドラインに入力します。

```
実行モジュール 入力ファイル -0 出力ファイル -各オプション
```

[操作例]

DWFXXX を使用して DWF ファイル「sample.dwf」を出力ファイル「sample.dt」に変換する場合は以下のように入力して変換します。

```
A> DWFXXX sample.dwf -0sample.dt -Zout.atr  
ただいま、ファイル変換中です。
```

```
DWFXXX 変換終了
```

DWFXXX	: 実行モジュール名 (例: DWF2RS, DWFTOH2, DWFTOPST)
sample.dwf	: 入力ファイル名
sample.dt	: 出力ファイル名
out.atr	: 出力属性ファイル名 (例: rasl_out.atr, hp2_out.atr, psx_out.atr, pdf_out.atr)

-0 : 変換結果を出力するファイル名を指定するオプション。
マイナス記号とアルファベットの「オー」です。
-0 と変換後のファイル名の間には空白を入れないようにします。
変換前のファイル名と変換後のファイル名に同じものを指定する事はできません。

※ オプション指定は大文字でも小文字でもかまいませんが、全角文字は使えません。

※ オプションの詳細についてはオプションの項目を参照下さい。

実行時のオプション

- A : オートスケールの設定(用紙サイズ一杯に出力されます。)
- AJ : 指定した用紙サイズより大きい場合のみ、自動的に用紙サイズに合わせて縮小スケールリングします。
- AL : 無条件で A 系列用紙出力となり、入力データを用紙のほぼ中央に配置されます。(長尺サイズは除く)
- M : マルチページが可能なフォーマットの場合に前のファイル出力に重ねて(次のページ)に出力されます。
- C : 入力データの文字データを無視。(線分、円等のみ)
- P[paper] : 用紙サイズを番号で指定します。(0:A0 横 1:A1 横 20:A0 縦 21:A1 縦など)
- LS : 用紙サイズが A0 版を超える場合に指定します。(ラスター出力)
実際の指示は-P オプションによって行います。
- LD : 出力解像度が 600DPI を超える場合に指定します。(ラスター出力)
実際の指示は出力属性パラメータ (OUT_RESOLUTION) で行います。
- CMAP : フルカラーをインデックスカラーに変換します。(カラーラスター出力)
- MONO : カラー出力でなく、モノクロ(2 値)に変換します。(DWF2RSC のみ)
- R[rotation] : 出力時 90, 180, 270 度回転します。(反時計方向)
- S[scale] : 全体のスケール値を指定します。
- X[xoff] : X 方向のオフセット値設定。 [単位:0.01 mm]
- Y[yoff] : Y 方向のオフセット値設定。 [単位:0.01 mm]
- OPEN[pen_file] : ペン番号の情報をファイルに出力します。(HP-GL/2 出力)
HP-GL/2 データの SP コマンド(ペン番号)の設定パラメータを
取得し pen_file で指定されたファイルへ出力します。
Pen_file が指定されていない場合はこのオプションは無効になります。
- D : 実線と実線以外の層を区別して出力されます。(DWG, DXF 出力)
(属性データ変換パラメータファイルより優先されます(PAGE_LAYER))
レイヤー名 ALL :実線データ
レイヤー名 DASH:実線以外の線データ
初期状態は線分データ全てレイヤー名 0 にセットされます。

-BATCH[batch_file] : バッチファイル名

指定されたファイル又はディレクトリ内のファイルを一括処理したい時に利用します。

<使用方法>

DWFXMLX -BATCH バッチファイル名

<バッチファイルの指定方法>

バッチファイルの指定方法は下記の通り

入力ファイル -0 出力ファイル [オプションパラメータ]

<操作例>

- (1) 1つのファイルを違うパラメータで違うファイル名で変換します。
TEST.DATA -0TEST_1.dt
TEST.DATA -0TEST_2.dt -R90
- (2) 拡張子.dataのファイルを拡張子.dtのファイルに変換します。
(ファイル名は自動的に検索されます)
.data -0.dt
- (3) ディレクトリ(DIR_A)の中の拡張子.dataのファイルを違う
ディレクトリ(DIR_B)の中の拡張子.dtのファイルに変換します。
(ファイル名は自動的に検索されます)
DIR_A%*.data -0DIR_B%*.dt -R90
- (4) 複数ファイルを1つのマルチページ(test.dt)ファイルに
変換します。(マルチページ出力可能なフォーマットのみ有効)
*.data -0test.dt -M

<制限事項>

- ①バッチファイル内で指定できるファイル名(ディレクトリを含む)は最大255文字までです。
- ②ファイルでワイルドカードを使用する場合は、必ず拡張子を指定して下さい。
- ③出力コマンドは指定されたコマンドの後に出力ファイル名を付加して実行します。内容についてはシステムによって異なりますので弊社では責任を問いません。
- ④ワイルドカードで指定した場合の処理されるファイルの順番はアルファベット順で処理されます。
- ⑤漢字のファイル名はWindows版のみ有効になります。
- ⑥スペースを含むファイル名の指定はできません。

<バッチファイル作成上の注意>

- ①バッチファイル内の項目は半角英数字で作成して下さい。
- ②バッチファイルはテキストファイルで作成して下さい。

<作成例>

Microsoft Wordで作成する場合は『テキスト改行』でファイルを保存して下さい。

その他のテキストエディタ(メモ帳等)を使用して作成して下さい。

- ③ファイル名の文字数についての制限はシステムに依存します。

属性データ変換パラメータファイル

属性データ変換パラメータファイルには、初期状態では

DWF 入力データ用 dwf_in. atr
ラスター(モノクロ)出力データ用 rasl_out. atr
ラスター(カラー)出力データ用 rascl_out. atr
HP-GL/2 出力データ用 hp2_out. atr
DWG 出力データ用 dwg_out. atr
DXF 出力データ用 dxf_out. atr

があり、この内容を変更する事で種々な変換が可能となります。

■入力属性データ変換パラメータ

IN_SYMWD, INSYMHT : 入力データの文字高さ、文字幅を倍尺で指定
PENCOLOR : 入力 DWF の色番号を変更
PENLAYER : レイヤー番号を変更
WINDOWS_FONT_OUT : WINDOWS フォントで出力
WINDOWS_FONT : 任意の TrueType フォントを指定
WINDOWS_FONT_HT : WINDOWS フォントの高さを指定
WINDOWS_FONT_WD : WINDOWS フォントの文字幅を指定
WINDOWS_FONT_SP : WINDOWS フォントの文字と文字の間隔を指定

・ DWF 入力データ用データ変換属性パラメータファイルのサンプル (dwf_in. atr)

```
#####dwf_in. atr#####  
#pen & line_color table  
#PENCOLOR=(P1=1, P2=2, P3=3, P4=4, P5=5, P6=6, P7=7, P8=8)  
#symbol_height_scale  
IN_SYMHT = 1.0  
#symbol_width_scale  
IN_SYMWD = 1.0  
#pen & layer table (DXF_OUT, IGES_OUT)  
#PENLAYER=(P1=L1, P2=L2, P3=L3, P4=L4, P5=L5, P6=L6, P7=L7, P8=L8)  
##### Windows font information #####  
WINDOWS_FONT_OUT=Y  
#WINDOWS_FONT=MS ゴシック  
#WINDOWS_FONT_HT=1.0  
#WINDOWS_FONT_WD=1.0  
#WINDOWS_FONT_SP=1.0  
#####end dwf_in. atr#####
```



```

# output size (unit:mm) (0.01 --> 1189.0*6)
#OUTPUT_SIZE_X = 1000.0
#OUTPUT_SIZE_Y = 596.0
# output resolution (unit:DPI) (36 --> 600)
OUT_RESOLUTION = 400
# raster working memory (unit:Mbytes)
BUNDLE_MEMORY = 32.0
#BUNDLE_MEMORY = 10.0
# output device scale (0.9 --> 1.1)(default:1.0)
DEVICE_SCALE = 1.0
# auto offset & auto paper_size margin(unit:mm(default:0.0)) : used -g option
AUTO_OFFSET_X = 5.0
AUTO_OFFSET_Y = 5.0
# filling mode on(1)/off(0)
FILL_MODE = 1
# paper margin (unit:mm)(default:0.0) (0.0 --> 999.99)
PAPER_MARGIN_X = 5.0
PAPER_MARGIN_Y = 5.0
# dashline cap (0:cut , 1:circle(default) , 2:square)
#DASHLINE_CAP = 0
# TIFF byte order (default:MPU depend , motorola:M , intel:I)
#TIFF_BYTEORDER=I
#TIFF_BYTEORDER=M
# if -a(-aj) on , space_area (0:no cut(default) , 1:cut)
#CUT_SPACE_AREA = 1
# output paper_size overflow limits(unit:mm) (0-->100(default:5)) (-aj option on)
PAPER_X_LIMITS = 10.0
PAPER_Y_LIMITS = 10.0
# dashline pattern down/up ratio(10%-->900%) (default:100%)
#DASH_DOWN_RATIO = 100
#DASH_UP_RATIO = 100
# TIFF page tag output (Y/N)
#TIFF_PAGETAG = N
# output area size expand(only -g) (0.0-->100.0mm) (default:0.0)
#OUTAREA_UP_X = 0.5
#OUTAREA_ntrol(0:1_dot , 1:depend on line_width(default))
#SAME_PT_DOT = 0
# A_series paper out(-g option) (Y:yes,N:no(input_size)(Default))
#A_PAPER_OUT = Y
# width byte boundary(0:off(default),byte_bound(1,2,4))
BYTE_BOUNDARY_X = 1
# height byte boundary(0:off(default),byte_bound(1,2,4))
#BYTE_BOUNDARY_Y = 1
# output dot size(unit:dots)
#OUTPUT_DOT_X = 1728
#OUTPUT_DOT_Y = 2286
# DXFinput clip mode(1:std. clip , 2:DXFclip)
CLIP_MINMAX=2
#####end rasl_out.atr#####

```

・ 出カラスター(カラー)データ用データ変換属性パラメータファイルのサンプル (rascl_out. atr)

```
#####rascl_out. atr#####
# output version (TIFF, SUN_RASTER, XWD, BMP, JPEG, PNG, HP-RTL)
#OUT_VERSION=TIFF_NON
OUT_VERSION=TIFF_PACK
#OUT_VERSION=SUN_RASTER
#OUT_VERSION=XWD
#OUT_VERSION=BMP
#OUT_VERSION=PNG
#OUT_VERSION=JPEG
#OUT_VERSION=HPRTL_ROW
#OUT_VERSION=HPRTL_NON
#OUT_VERSION=HPRTL_PACK
#symbol height_scale (0.01 --> 99.99)
OUT_SYMHT=1.0
# symbol width_scale (0.01 --> 99.99)
OUT_SYMWD=1.0
# symbol space (unit:mm) (0.0 --> 99.99)
OUT_SYMSP=0.0
#line type pattern set(pattern_length * scale_factor) (unit:mm)
DOT_LINE = 1.0, 1.0
DASH_LINE = 2.0, 1.0
CENTER_LINE = 10.0, 1.0, 1.0
DIVIDE_LINE = 10.0, 1.0, 1.0
# linewidth available(Y:attribute(default),N:input data)
#ATR_LINEWIDTH = N
ATR_LINEWIDTH = Y
# color/width cross conversion table(max 256 tables)
COLORWIDTH=(C1=0.1, C2=0.2, C3=0.3, C4=0.4, C5=0.5, C6=0.6, C7=0.7, C8=0.8)
# output size (unit:mm) (0.01 --> 999.99)
#OUTPUT_SIZE_X = 100.0
#OUTPUT_SIZE_Y = 100.0
# output resolution (unit:DPI) (36 --> 600)
OUT_RESOLUTION = 100
# paper margin (unit:mm) (0.0 --> 999.99)
PAPER_MARGIN_X = 0.0
PAPER_MARGIN_Y = 0.0
# index color table (0 --> 254)
COLORRRGB=(C1=0, 0, 0, C2=1000, 0, 0, C3=0, 1000, 0, C4=1000, 1000, 0, C5=0, 0, 1000)
COLORRRGB=(C6=1000, 0, 1000, C7=0, 1000, 1000, C8=0, 0, 0, C9=1000, 0, 0, C10=0, 1000, 0)
COLORRRGB=(C11=1000, 1000, 0, C12=0, 0, 1000, C13=1000, 0, 1000, C14=0, 1000, 1000)
COLORRRGB=(C15=1000, 1000, 1000, C16=0, 0, 0, C17=1000, 0, 0, C18=0, 1000, 0)
COLORRRGB=(C19=1000, 1000, 0, C20=0, 0, 1000)
# color order (0:point , 1:line , 2:face) (only set NO_COMPRESS)
#COLOR_ORDER = 2
# color format (0:index , 1:RGB) (only set NO_COMPRESS)
#COLOR_FORMAT = 1
# gamma table file
#GAMMA_FILE=gamma.tbl
#symbol line_width (unit:mm) (0.0mm --> 20.0mm)
SYMBOL_LINEWIDTH = 0.1
# raster working memory (unit:Mbytes)
BUNDLE_MEMORY = 32.0
# output device scale (0.9 --> 1.1)
DEVICE_SCALE = 1.0
# auto offset & auto paper_size margin (unit:mm)
AUTO_OFFSET_X = 0.0
AUTO_OFFSET_Y = 0.0
# filling mode on(1)/off(0)
#FILL_MODE = 1
# dashline cap (0:cut , 1:circle(default) , 2:square)
#DASHLINE_CAP = 0
# if -a(-aj) on , space_area (0:no cut(default) , 1:cut)
#CUT_SPACE_AREA = 1
# output paper_size overflow limits(unit:mm) (0-->100(default:5)) (-aj option on)
PAPER_X_LIMITS = 10.0
PAPER_Y_LIMITS = 10.0
# dashline down/up ratio (unit:%, default:100%) (10 --> 900)
#DASH_DOWN_RATIO = 50
#DASH_UP_RATIO = 200
```

```
# TIFF page tag output (Y/N(default))
#TIFF_PAGETAG = N
# output area size expand(only -g) (0.0-->100.0mm) (default:0.0)
#OUTAREA_UP_X = 0.5
#OUTAREA_ntrol(0:1_dot ,1:depend on line_width(default))
#SAME_PT_DOT = 0
# A_series paper out(-g option) (Y:yes,N:no(input_size) (Default))
#A_PAPER_OUT = Y
# width byte boundary(0:off(default), byte_bound(1, 2, 4))
BYTE_BOUNDARY_X = 1
# height byte boundary(0:off(default), byte_bound(1, 2, 4))
#BYTE_BOUNDARY_Y = 1
# output dot size(unit:dot)
#OUTPUT_DOT_X = 1728
#OUTPUT_DOT_Y = 2286
#DXFinput clip mode(1:std. clip(default) , 2:DXFclip)
#CLIP_MINMAX = 1
#####end rascl_out.atr#####
```

■HP-GL/2 出力属性データ変換パラメータ (hp2_out.atr)

OUT_VERSION : 出力データフォーマットを変更
COLORWIDTH : 出力 HP-GL/2 データの線幅を変更
(注) DWF データのポリラインの線幅を有効にしたい時は COLORWIDTH パラメータを
コメント(先頭に#を付けます)として下さい。

```
#####hp2_out.atr#####  
# version  
OUT_VERSION = HPGL/2  
# symbol_height_scale  
OUT_SYMHT = 1.0  
# symbol_width_sclae  
OUT_SYMWD = 1.0  
# symbol_width_space(unit:mm)  
OUT_SYMSP = 0.0  
# color/pen cross conversion table  
COLORPEN=(C1=1, C2=2, C3=3, C4=4, C5=5, C6=6, C7=7, C8=8)  
# index color table (0 --> 254)  
COLORRGB=(C1=0, 0, 0, C2=1000, 0, 0, C3=0, 1000, 0, C4=1000, 1000, 0, C5=0, 0, 1000)  
COLORRGB=(C6=1000, 0, 1000, C7=0, 1000, 1000)  
COLORRGB=(C8=0, 0, 0, C9=1000, 0, 0, C10=0, 1000, 0, C11=1000, 1000, 0, C12=0, 0, 1000)  
COLORRGB=(C13=1000, 0, 1000, C14=0, 1000, 1000, C15=1000, 1000, 1000)  
COLORRGB=(C16=0, 0, 0, C17=1000, 0, 0, C18=0, 1000, 0, C19=1000, 1000, 0, C20=0, 0, 1000)  
COLORRGB=(C21=1000, 0, 1000, C22=0, 1000, 1000, C23=1000, 1000, 1000)  
COLORRGB=(C24=0, 0, 0, C25=1000, 0, 0, C26=0, 1000, 0, C27=1000, 1000, 0, C28=0, 0, 1000)  
COLORRGB=(C29=1000, 0, 1000, C30=0, 1000, 1000, C31=1000, 1000, 1000)  
# color/width cross conversion table  
COLORWIDTH=(C1=0.1, C2=0.2, C3=0.3, C4=0.4, C5=0.5, C6=0.6, C7=0.7, C8=0.8)  
# clip command out (Y:available , N:ignore(default))  
#CLIP_CMD_OUT = Y  
#####end hp2_out.atr#####
```

■DWG 出力属性データ変換パラメータ (dwg_out. atr)

OUT_VERSION : 出力データフォーマットを変更
KANJI_CODE : 漢字コードを変更
DASH_NAME : 線種パターンを変更
LEFT_LOW_X, LEFT_LOW_Y : 両端の空白域を制御
RIGHT_UP_X, RIGHT_UP_Y : 両端の空白域を制御
PEGE_LAYER : ページ単位にレイヤー別けを制御
KANJI_FONT_NAME : 文字のフォントを英語か日本語かを指定
LINE_CONNECT : 出力 DWG のデータ量を減らす

```
#####dwg_out. atr#####  
# dwg format version(R13J, R14J, AC2000)  
OUT_VERSION = R13J  
#OUT_VERSION = R14J  
#OUT_VERSION = AC2000  
# kanji code (SJIS, EUC)  
KANJI_CODE = SJIS  
#KANJI_CODE = EUC  
# kanji font name(only English version)  
#KANJI_FONT_NAME = NON  
#KANJI_FONT_NAME = D F 特太ゴシック体, @MS ゴシック, 1.1, 1.4  
#KANJI_FONT_NAME = *, HG 正楷書体-PRO, 2.0, 2.0  
# page layer (Y:yes , N:no(default))  
PAGE_LAYER = N  
# dash_line pattern name (STD:AutoCAD, KD:KDconvert(default))  
#DASH_NAME = STD  
# available line width(for POLYLINE) (N:no(default), Y:yes)  
#LINEWIDTH = Y  
#VPORT MARGIN(edge space(unit:mm))  
#LEFT_LOW_X = -10.0  
#LEFT_LOW_Y = -10.0  
#RIGHT_UP_X = 10.0  
#RIGHT_UP_Y = 10.0  
# connect mode(N:no(default), Y:yes)  
#LINE_CONNECT = Y  
#####end dwg_out. atr#####
```

■DXF 出力属性データ変換パラメータ (dxf_out. atr)

OUT_VERSION : 出力データフォーマットを変更
KANJI_CODE : 漢字コードを変更
DASH_NAME : 線種パターンを変更
PEGE_LAYER : ページ単位にレイヤー別けを制御
KANJI_FONT_NAME : 文字のフォントを英語か日本語かを指定
LINE_CONNECT : 出力 DXF のデータ量を減らす

```
#####dxf_out. atr#####  
# dxf format version(GX-5, R12J, R13J, R14J)  
#OUT_VERSION = R12J  
#OUT_VERSION = R13J  
OUT_VERSION = R14J  
# kanji code (SJIS, EUC)  
KANJI_CODE = SJIS  
#KANJI_CODE = EUC  
# kanji font name(only English version)  
#KANJI_FONT_NAME = NON  
# kanji font name(only English version)  
#KANJI_FONT_NAME = NON  
#KANJI_FONT_NAME = D F特太ゴシック体, @MS ゴシック, 1.1, 1.4  
#KANJI_FONT_NAME = *, HG 正楷書体-PRO, 2.0, 2.0  
# page layer (Y:yes , N:no(default))  
PAGE_LAYER = N  
# dash_line pattern name (STD:AutoCAD, KD:KDconvert(default))  
#DASH_NAME = STD  
#DASH_NAME = KD  
# color to line width(color no limit 0 - 4095, line width limit 0.0 - 30.0)  
#COLORWIDTH=(C1=0.1, C2=0.2, C3=0.3, C4=0.4, C5=0.5, C6=0.6, C7=0.7, C8=0.8)  
# available line width(for POLYLINE) (N:no(default), Y:yes)  
# COLORWIDTH が設定されている場合は無効になる  
#LINEWIDTH = Y  
#line kind scale set(1.0:default)  
#LTSCALE = 0.125  
# connect mode (Y:yes , N:no(default))  
#LINE_CONNECT = Y  
#####end dxf_out. atr#####
```

サンプル例

例： sample.dwf と同じ大きさになるように sample.tif へ変換
原点移動を行い入力データの最小値を第一象限の (0, 0) へ移動します。

```
C:¥>DWF2RS sample.dwf -0sample.tif -AG -I
Copyright(c) 2005-2009 by Kernel Computer System Co.,Ltd. All rights reserved.

DWF(sample.dwf) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

DWF2RS 変換終了
```

例： sample.dwf を A4 サイズへスケールして sample.tif へ変換

```
C:¥>DWF2RS sample.dwf -0sample.tif -A -P4 -I
Copyright(c) 2005-2009 by Kernel Computer System Co.,Ltd. All rights reserved.

DWF(sample.dwf) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

ベクター/ラスター変換処理 100%終了

DWF2RS 変換終了
```

例： sample.dwf を 1/10 のスケールを指定して sample.tif へ変換。-AG で第一象限に原点移動。

```
C:¥>DWF2RS sample.dwf -0sample.tif -S0.1 -AG -I
Copyright(c) 2005-2009 by Kernel Computer System Co.,Ltd. All rights reserved.

DWF(sample.dwf) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

DWF2RS 変換終了
```

制限事項

- ◇ DWF の読み込み処理は、Autodesk 社の DWF ToolKit(ver7.5) を使用しますので仕様については DWF ToolKit に依存します。
- ◇ 同一 PC 上での多重起動は行えません。