

## 1. これは何をするものか

これは SAORI 規格の DLL です。ゴーストの位置を様々なロジックに従い、特定のウィンドウの矩形と同期させる機能を持ちます。ゴーストをウィンドウに「座らせる」「貼り付ける」「ぶら下げる」といった事を目的として作成されました。

Osuwari.dll は version1 が過去に公開されており、これは機能強化した version2 となります。

**尚、本 DLL は materia では動作しません。SSP もしくは CROW で利用してください。**

## 2. 利用法概要

ゴーストの位置の制御を開始する、あるいは制御パラメータを変更するには、START 構文を用います。START 構文はゴーストの存在する間、何度呼んでも構いません。

一般的には、HWND イベントの際に呼ぶことを推奨します。一度呼べばその際の制御パラメータに従い、DLL がアンロードされるか STOP 構文が呼ばれるまで位置の制御を行うため、呼び出し続ける必要はありません。

お座りを停止させる場合には STOP 構文を用いてください。一般的にはゴーストの終了と共に DLL がアンロードされるため、STOP 構文を利用しなくても問題ないでしょう。

この DLL はオプションが多彩なため構文が少々複雑です。単純に利用する場合は 4 . 記載例を先に見てください。

## 2. START 構文

### (1) 構文

START 構文は以下のパラメータを記載してください。

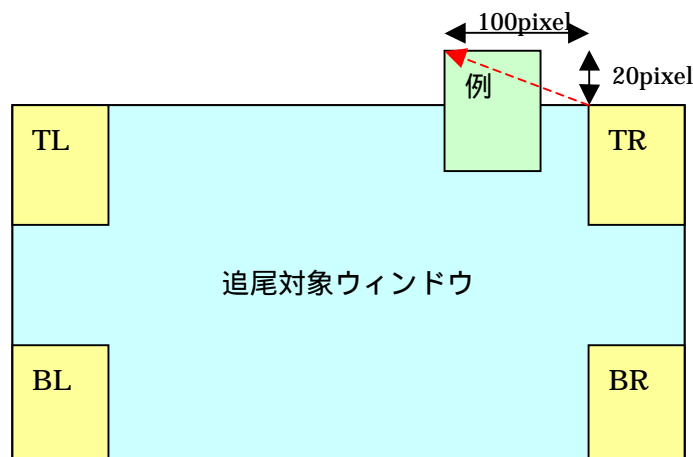
( START 等、この書体で記載されているものは、その通りの文字列を記載する事を示します )

Argument	記載内容	説明
Argument0	START	大文字文字列で記載
Argument1	制御対象の HWND	一般的には HWND イベントで渡されます。 ( 4 . 記載例も見てください )
Argument2	追尾対象	下記の何れかを記載してください。 ACTIVE : アクティブウィンドウを対象にします FIX : 画面全体もしくはワークエリア ( 画面からタスクバー等の領域を除いた部分、つまり壁紙が見えている部分 ) を対象にします。 @文字列 : ウィンドウタイトルに「文字列」を持つウィンドウを対象にします #数値 : ウィンドウハンドルが「数値」のウィンドウを対象にします
Argument3	位置原点	Argument2 で指定したウィンドウの、どこを原点とするかを、下記の何れかで指定してください。 TL : 左上隅を原点とします。 TR : 右上隅を原点とします。

		BL：左下隅を原点とします。 BR：右下隅を原点とします。
Argument4	x 座標差分	原点からの x 座標のずれを記載してください。
Argument5	y 座標差分	原点からの y 座標のずれを記載してください。
Argument6	更新間隔	位置のチェック間隔をミリ秒で記載してください。 (一般的には 50 ~ 100ms 程度が良いと思います。数値が小さい程「なめらかに」移動しますが、システムへの負荷も大きくなります)
Argument7 (optional)	オプション	下記のオプションをスペース区切りで指定してください。複数指定しても、まったく指定しなくてもかまいません。 XMOVE：横方向にゴーストを移動可能にします YMOVE：縦方向にゴーストを移動可能にします NOCLIP：見切れ判定を行いません
Argument8 (optional)	位置取得失敗時指定	Argument2 が FIX の場合、対象の矩形取得に失敗した場合、あるいは見切れが発生した場合の位置を指定します。 と をそれぞれスペース区切りで指定して下さい。指定しなかった場合は DESKTOP CENTER と指定されたものとして扱います。 位置取得失敗時の対象 DESKTOP：デスクトップを対象とします。 WORKAREA：ワークエリアを対象とします。 位置取得失敗時の仮想矩形 CENTER：デスクトップ又はワークエリアのウィンドウをそのまま用います。 TOP：対象の矩形の上側に仮想ウィンドウを想定します。 LEFT：対象の矩形の左側に仮想ウィンドウを想定します。 RIGHT：対象の矩形の右側に仮想ウィンドウを想定します。 BOTTOM：対象の矩形の下側に仮想ウィンドウを想定します。

## ( 2 ) 位置指定について

位置の指定、原点、差分について、下図を参考にしてください。黄色は x,y 差分が両方 0 の場合の位置になります。( 例：位置原点 = TR、x 差分 = -100、y 差分 = -20 とした場合 )



### ( 3 ) 位置取得失敗時・位置固定時の動作について

以下の場合にはデスクトップかワークエリアを追尾対象とし、Argument8 の記載が有効になります。

- ・ 追尾対象が `FIX` の場合
- ・ 追尾対象が `@文字列` あるいは `#数値` で、対象のウィンドウが存在しなかった場合
- ・ 追尾対象が `ACTIVE` で、アクティブウィンドウを追尾対象と出来ない場合（原因は色々あります）
- ・ オプションに `NOCLIP` が指定されておらず、「見切れ」（そのまま追尾すると位置が画面からはみ出すこと）が発生する場合

その際に「代用として」計算される位置と、位置取得失敗時指定の関係については下図を参考にしてください。以下の点線の矩形を「仮想的な追尾対象ウィンドウ」として、( 2 ) で示した位置計算を行います。（つまり、位置原点が TL もしくは TR の時に TOP を指定するのは 画面からはみ出して見えなくなってしまうため ナンセンスです。）



### 3 . STOP 構文

STOP 構文を用いると、DLL の動作を停止できます。以下のパラメータを記載してください。

Argument	記載内容	説明
Argument0	STOP	大文字文字列で記載

Argument1 以降は記載しても無視されます。

## 4 . 記載例

### ( 1 ) HWND 取得の記載例

本 DLL を利用するためには、位置制御対象の HWND を知る必要があります。一般的には¥0 もしくは¥1 の HWND を指定することになると思います。¥0、¥1 の HWND は本体から HWND イベントで通知されます。

以下に里々での HWND 取得コード例を記載します。このコードにより sakurahwnd 変数に¥0 の HWND が入ります。

```
* hwnd
$ temp      ( split , ( R 0 ) , ( sprintf , %c , 1 ) )
$ sakurahwnd ( S 0 )
```

### ( 2 ) 「お座り」

ゴーストをウインドウの右上に「お座り」させるための構文例を記載します。

Argument2:ACTIVE ( アクティブウインドウに「お座り」させる )

Argument3:TR ( ウインドウの右上を原点とする )

Argument4:-20 ( ウインドウの右上から 20 ピクセル左に寄った場所を初期位置とする )

Argument5:-90 ( ゴーストの「座高」が 90 ピクセルの場合 )

Argument6:100 ( 100[ms]毎に更新 )

Argument7:XMOVE ( 横方向に移動可能、見切れあり )

Argument8:BOTTOM WORKAREA ( 「お座り」失敗時にはワークエリアの下端に座る )

Argument2 を FIX とすると、つねにワークエリア下端に「お座り」します。

### ( 3 ) 「ぶら下がり」

ゴーストをウインドウの左下に「ぶら下がり」させるための構文例を記載します。

Argument2:ACTIVE ( アクティブウインドウに「ぶら下がり」させる )

Argument3:BL ( ウインドウの左下を原点とする )

Argument4:30 ( ウインドウの左下から 30 ピクセル右に寄った場所を初期位置とする )

Argument5:160 ( ゴーストの「手から下の高さ」が 160 ピクセルの場合 )

Argument6:100 ( 100[ms]毎に更新 )

Argument7: ( 移動できない、見切れあり )

Argument8:TOP WORKAREA ( 「ぶら下がり」失敗時にはワークエリアの上端にぶら下がる )

Argument2 を FIX とすると、つねにワークエリア上端に「ぶら下がり」ます。

#### ( 4 ) 総合例

例の ( 1 ) と ( 3 ) を総合し、里々でゴーストを「お座り」させるためのコード例を記載します。

##### 【dic02\_event.txt】

```
@お座り
( osuwari, ( A 0 ), ( A 1 ), ACTIVE, TR, -20, -90, 100, XMOVE, BOTTOM WORKAREA )

* OnSatoriLoad
$ お座り実行済 0
$ sakurahwnd 0

* OnSatoriUnload
> お座り解除 ( お座り実行済 )

* お座り解除
( call, お座り, STOP, ( sakurahwnd ) )

* hwnd
$ sakurahwndold ( sakurahwnd )
$ temp ( split, ( R 0 ), ( sprintf, %c, 1 ) )
$ sakurahwnd ( S 0 )
> hwnd 脱出 ( sakurahwnd ) == ( sakurahwndold )
> お座り一旦解除 ( sakurahwndold )
> お座り実行

* hwnd 脱出
# ここはなにも書かない

* お座り一旦解除
$ お座り実行済 0
( call, お座り, STOP, ( sakurahwndold ) )
> お座り実行

* お座り実行
$ お座り実行済 1
( call, お座り, START, ( sakurahwnd ) )
```

##### 【satori\_conf.txt】

```
@SAORI
osuwari,saori/osuwari.dll
```

## 5 . 注意事項・制限事項

- ・ 本 DLL を利用した事による一切の損害の責任を負いません。at your own risk をお願いします。
- ・ osuwari.dll を利用したゴーストを複数同時起動した場合の動作は未定です。
- ・ 現時点では制御対象とできるウィンドウは 1 つだけです。¥0 と ¥1 を同時に制御できません。  
「りむ・えくすとり~む」では、osuwari.dll を osuwari0.dll、osuwari1.dll とリネームし、¥0 と ¥1 でそれぞれを利用することによってこの制限を回避されています。思いつきませんでした。
- ・ その他異常な動作をする事があるかもしれません。その際はどんな事でも報告頂けると有難いです（それは仕様かもしれませんしバグかもしれません、直せるとも限りませんが）

## 6 . 更新履歴

2005/01/23 version2.0.0.0 初版作成

2006/02/15 version2.0.1.0 最前面に行くロジック変更

（「りむ・えくすとり~む」でのちらつき対策。他に最前面のウィンドウがあると負けやすくなりましたが）

2006/02/23 version2.0.2.0 正しく終了できないことがあるバグを修正

## 7 . 謝辞

開発にあたってはえびさわ様の gethwnd.dll ソースを大いに参考にさせて頂き、また勝手ながらだ  
いぶ流用させて頂いております。お礼申し上げます。

<http://www33.tok2.com/home/ebi/index.shtml> 「何かぬるめの…」 by えびさわ様

osuwari.dll の動作確認その他について、黒羽龍矢様及び TJ 様にご協力頂いております。お礼申し  
上げます。

<http://kurobanetatuya.hp.infoseek.co.jp/genkan.htm> 「龍屋」 by 黒羽龍矢様

<http://seriko.nanika.jp/sstpviewer/> 「SSTP-viewer's street」 by TJ 様

里々のコード例については殊海夕音様にご協力頂いております。お礼申し上げます。

<http://kotomishome.hp.infoseek.co.jp/> 「未設定航路-phase2-」 by 殊海夕音様

Chameleon Ponapalt 様には作成に当たって様々な助言を頂きました。お礼申し上げます。

<http://ssp.shillest.net/> 「SSP (en)BUGTRAQ」 by Chameleon Ponapalt 様

## 8 . 連絡先

本 DLL に関する連絡先は下記の通りです（2006 年 2 月現在）

浮子屋

<mailto:ukiya@saku2.com>

<http://ukiya.sakura.ne.jp/>

以上