

# HView Reference

## (1) Viewを獲得と配置

- HViewGetRoot

ウィンドウのルートViewを得る (Mac OS X 10.2以降)

```
HViewRef HViewGetRoot (  
    WindowRef inWindow  
);
```

inWindow

- ルートViewを得たいウィンドウ (WindowRefが正しくない場合はNULLが返る)

- HViewFindByID

ID値を指定してViewを得る (Mac OS X 10.2以降)

```
OSStatus HViewFindByID (  
    HViewRef inStartView,  
    HViewID inID,  
    HViewRef * outControl  
);
```

inStartView

- 検索を開始するView

inID

- 検索するViewのID値 (HViewIDはControlIDと同じ)

outControl

- ・ 発見されたView (HViewRefはコントロールのControlRefと同じ)

- ・ HViewGetSuperview

指定したViewの親View (Superview) を得る (Mac OS X 10.2以降)

```
HViewRef HViewGetSuperview (  
    HViewRef inView  
);
```

inView

- ・ 親Viewを得たいView (親が無い場合はNULLが返る)

- ・ HViewPlaceInSuperviewAt

Viewを親Viewの絶対位置に配置する (Mac OS X 10.2以降)

```
OSStatus HViewPlaceInSuperviewAt (  
    HViewRef inView,  
    float inX,  
    float inY  
);
```

inView

- ・ 配置したいView

inX

- ・ Viewを配置するためのX座標

inY

- ・ Viewを配置するためのY座標

- ・ HViewGetNextView

指定されたViewの次のViewを返す (Mac OS X 10.2以降)

```
HViewRef HViewGetNextView (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (次にViewが無ければNULLが返る)

- ・ HViewGetPreviousView

指定されたViewの前のViewを返す (Mac OS X 10.2以降)

```
HViewRef HViewGetPreviousView (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (前にViewが無ければNULLが返る)

## (2) Subview関連

- ・ HViewAddSubview

指定Viewを親Viewに追加する (Mac OS X 10.2以降)

```
OSStatus HViewAddSubview (  
    HViewRef inParent,  
    HViewRef inNewChild  
);
```

inParent

- ・ Viewを追加するための親View

inNewChild

- ・ 親Viewに追加したいView

## ・ HViewRemoveFromSuperview

指定Viewを親Viewから削除する (Mac OS X 10.2以降)

```
OSStatus HViewRemoveFromSuperview (  
    HViewRef inView  
);
```

inView

- ・ 削除したいView

## ・ HViewGetFirstSubview

親Viewに埋め込まれている最初のViewを得る (Mac OS X 10.2以降)

```
HViewRef HViewGetFirstSubview (  
    HViewRef inView  
);
```

inView

- ・ 対象となる親View (Viewが無ければNULLが返る)

## ・ HViewGetLastSubview

親Viewに属埋め込まれている最後のViewを得る (Mac OS X 10.2以降)

```
HViewRef HViewGetLastSubview (  
    HViewRef inView  
);
```

inView

- ・ 対象となる親View (Viewが無ければNULLが返る)

## ・ HViewCountSubviews

指定された親Viewに埋め込まれているViewの個数を得る (Mac OS X 10.4以降)

```
CFIndex HViewCountSubviews (  
    HViewRef inView  
);
```

inView

- ・ 対象となる親View

## ・ HViewGetIndexedSubview

親Viewに埋め込まれたViewをインデックス値により得る (Mac OS X 10.4以降)

```
CFIndex HViewGetIndexedSubview (  
    HViewRef inView,  
    CFIndex inSubviewIndex,  
    HViewRef * outSubview  
);
```

inView

- ・ 対象となる親View

inSubviewIndex

- ・ インデックス値

outSubview

- ・ 得られたView

## (3) Viewの操作

- `HIViewSetVisible`

Viewの表示／非表示を切り替える (Mac OS X 10.2以降)

```
OSStatus HIViewSetVisible (  
    HIViewRef inView,  
    Boolean inVisible  
);
```

`inView`

- 対象となるView

`inVisible`

- `true`なら表示に`false`なら非表示に切り替える

- `HIViewIsVisible`

Viewが表示されているかどうかチェックする (Mac OS X 10.2以降)

```
Boolean HIViewIsVisible (  
    HIViewRef inView  
);
```

`inView`

- チェック対象となるView (戻り値が`true`なら表示`false`なら非表示)

- `HIViewIsLatentlyVisible`

親Viewが非表示の場合にViewが表示されているかどうかチェックする (Mac OS X 10.4以降)

```
Boolean HViewIsLatentlyVisible (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (戻り値がtrueなら表示されているfalseなら非表示)

## ・ HViewSetHilite

Viewの一部をハイライト表示にする (Mac OS X 10.4以降)

```
OSStatus HViewSetHilite (  
    HViewRef inView,  
    HViewPartCode inHilitePart  
);
```

inView

- ・ 対象となるView

inHilitePart

- ・ ハイライト表示にする部分 (HViewPartCode)

## ・ HViewIsActive

Viewがアクティブかどうかをチェックする (Mac OS X 10.4以降)

```
Boolean HViewIsActive (  
    HViewRef inView,  
    Boolean * outIsLatentActive  
);
```

inView

- ・ 対象となるView (戻り値がtrueならアクティブfalseならインアクティブ)

outIsLatentActive

- ・ trueなら隠れた状態でアクティブfalseならそうではない（必要なければNULLを渡す）

## ・ HViewSetActive

Viewをアクティブに設定する（Mac OS X 10.4以降）

```
OSStatus HViewSetActive (  
    HViewRef inView,  
    Boolean inSetActive  
);
```

inView

- ・ 対象となるView

inSetActive

- ・ trueならアクティブfalseならインアクティブ

## ・ HViewIsEnabled

Viewが使用可能なか不可なのかをチェックする（Mac OS X 10.4以降）

```
Boolean HViewIsEnabled (  
    HViewRef inView,  
    Boolean * outIsLatentEnabled  
);
```

inView

- ・ 対象となるView（戻り値がtrueなら使用可falseなら使用不可）

outIsLatentEnabled

- ・ trueなら隠れた状態で使用可falseならそうではない（必要なければNULLを渡す）



## ・ HViewSetEnabled

Viewを使用可または不可に設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetEnabled (  
    HViewRef inView,  
    Boolean inSetEnabled  
);
```

inView

- ・ 対象となるView

inSetEnabled

- ・ trueなら使用可にfalseなら使用不可に

## ・ HViewIsCompositingEnabled

Viewがコンポジットモードかどうかをチェックする (Mac OS X 10.4以降)

```
Boolean HViewIsCompositingEnabled (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

## ・ HViewSetText

Viewにテキストを設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetText (  
    HViewRef inView,  
    CFStringRef inText  
);
```

inView

- ・ 対象となるView

outIsLatentEnabled

- ・ 設定するテキスト (CFStringRef)

- ・ **HViewCopyText**

Viewからテキストを得る (Mac OS X 10.4以降)

```
CFStringRef HViewCopyText (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (CFStringRefが不要になったらリリースする必要あり)

- ・ **HViewGetValue**

Viewからコントロール値を得る (Mac OS X 10.4以降)

```
SInt32 HViewGetValue (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

- ・ **HViewSetValue**

Viewにコントロール値を設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetValue (  
    HViewRef inView,
```

```
    SInt32 inValue  
);
```

inView

- ・対象となるView

inValue

- ・設定するコントロール値

## ・ HViewGetMinimum

Viewから最小値を得る (Mac OS X 10.4以降)

```
SInt32 HViewGetMinimum (  
    HViewRef inView  
);
```

inView

- ・対象となるView

## ・ HViewSetMinimum

Viewに最小値を設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetMinimum (  
    HViewRef inView,  
    SInt32 inMinimum  
);
```

inView

- ・対象となるView

inMinimum

- ・ 設定する最小値

## ・ HViewGetMaximum

Viewから最大値を得る (Mac OS X 10.4以降)

```
SInt32 HViewGetMaximum (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

## ・ HViewSetMaximum

Viewに最大値を設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetMaximum (  
    HViewRef inView,  
    SInt32 inMaximum  
);
```

inView

- ・ 対象となるView

inValue

- ・ 設定する最大値

## ・ HViewGetViewSize

ViewのViewサイズ値を得る (Mac OS X 10.4以降)

```
SInt32 HViewGetViewSize (  
    HViewRef inView
```

);

inView

- ・ 対象となるView

- ・ **HViewSetViewSize**

ViewにViewサイズ値を設定する (Mac OS X 10.4以降)

OSStatus HViewSetViewSize (

    HViewRef inView,

    SInt32 inViewSize

);

inView

- ・ 対象となるView

inValue

- ・ 設定するViewサイズ値

- ・ **HViewsValid**

そのViewがHIToolboxで認識可能かどうかチェックする (Mac OS X 10.4以降)

Boolean HViewsValid (

    HViewRef inView

);

inView

- ・ 対象となるView (戻り値がtrueなら認識されているfalseならそうではない)

- ・ **HViewGetID**

ViewのID (HViewID) を得る (Mac OS X 10.4以降)

```
OSStatus HViewGetID (  
    HViewRef inView,  
    HViewID * outID  
);
```

inView

- ・ 対象となるView

outID

- ・ 得られたID (HViewID)
- ・ HViewSetID

ViewにID (HViewID) を設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetID (  
    HViewRef inView,  
    HViewID inID  
);
```

inView

- ・ 対象となるView

inID

- ・ 設定するID (HViewID)
- ・ HViewGetCommandID

ViewのコマンドIDを得る (Mac OS X 10.4以降)

```
OSStatus HViewGetCommandID (  
    HViewRef inView,  
    UInt32 * outCommandID
```

);

inView

- ・ 対象となるView

outCommandID

- ・ 得られたコマンドID

## ・ HViewSetCommandID

ViewにコマンドIDを設定する (Mac OS X 10.4以降)

```
OSStatus HViewSetCommandID (  
    HViewRef inView,  
    UInt32 inCommandID  
);
```

inView

- ・ 対象となるView

inCommandID

- ・ 設定するコマンドID

## ・ HViewGetKind

Viewの種類を得る (Mac OS X 10.4以降)

```
OSStatus HViewGetKind (  
    HViewRef inView,  
    HViewKind * outViewKind  
);
```

inView

- ・ 対象となるView

outViewKind

- ・ 得られた種類

## ・ HViewGetAttributes

Viewの属性を得る (Mac OS X 10.2以降)

```
OSStatus HViewGetAttributes (  
    HViewRef inView,  
    OptionBits* outAttrs  
);
```

inView

- ・ 対象となるView

outAttrs

- ・ 得られた属性

## ・ HViewChangeAttributes

Viewの属性を変更する (Mac OS X 10.2以降)

```
OSStatus HViewChangeAttributes (  
    HViewRef inView,  
    OptionBits inAttrsToSet,  
    OptionBits inAttrsToClear  
);
```

inView

- ・ 対象となるView

inAttrsToSet



- ・セットするアトリビュート

inAttrsToClear

- ・クリアするアトリビュート

## ・ HViewGetNeedsDisplay

指定Viewの再描画が必要かどうかを決定する (Mac OS X 10.2以降)

```
Boolean HViewGetNeedsDisplay (  
    HViewRef inView  
);
```

inView

- ・対象となるView

## ・ HViewSetNeedsDisplay

指定Viewの完全な再描画が行われるように指示する (Mac OS X 10.2以降)

```
Boolean HViewSetNeedsDisplay (  
    HViewRef inView  
);
```

inView

- ・対象となるView

## ・ HViewSetNeedsDisplayInRect

指定Viewの一部 (矩形領域内) を再描画するように指示する (Mac OS X 10.4以降)

```
OSStatus HViewSetNeedsDisplayInRect (  
    HViewRef inView,  
    const HIRect * inRect,
```

```
Boolean inNeedsDisplay  
);
```

inView

- ・ 部分再描画の対象となるView

inRect

- ・ 部分描画の領域を指定するための矩形領域 (HIRect)

inNeedsDisplay

- ・ inRectを部分描画領域に追加するのかそこから削除するのかを指定する

- ・ **HIViewSetNeedsDisplayInShape**

指定Viewの一部 (Shape内) を再描画するように指示する (Mac OS X 10.4以降)

```
OSStatus HIViewSetNeedsDisplayInShape (  
    HIViewRef inView,  
    HIShapeRef inArea,  
    Boolean inNeedsDisplay  
);
```

inView

- ・ 部分再描画の対象となるView

inArea

- ・ 部分描画の領域を指定するための領域 (HIShapeRef)

inNeedsDisplay

- ・ inAreaを部分描画領域に追加するのかそこから削除するのかを指定する

- ・ **HIViewSetNeedsDisplayInRegion**

指定Viewの一部（Region内）を再描画するように指示する（Mac OS X 10.2以降）

```
OSStatus HViewSetNeedsDisplayInRegion (  
    HViewRef inView,  
    RgnHandle inRgn,  
    Boolean inNeedsDisplay  
);
```

inView

- ・ 部分再描画の対象となるView

inRgn

- ・ 部分描画の領域を指定するためのRegion

inNeedsDisplay

- ・ inRgnを部分描画領域に追加するのかそこから削除するのかを指定する

- ・ HViewRender

Viewの再描画が必要な部分を再描画する（Mac OS X 10.3以降）

```
OSStatus HViewRender (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

- ・ HViewGetSizeConstraints

Viewの最小サイズと最大サイズ（サイズ制限の範囲）を得る（Mac OS X 10.2以降）

```
OSStatus HViewGetSizeConstraints (  
    HViewRef inView,  
    Rect *pRect,  
    Rect *pRect2  
);
```

```
HViewRef inView,  
HSize* outMinSize,  
HSize* outMaxSize  
);
```

inView

- ・対象となるView

outMinSize

- ・得られた最小サイズ

outMaxSize

- ・得られた最大サイズ

## ・ HViewIsDrawingEnabled

指定Viewが描画可能かどうかをチェックする (Mac OS X 10.2以降)

```
Boolean HViewIsDrawingEnabled (  
    HViewRef inView  
);
```

inView

- ・対象となるView (可能ならtrue不可能ならfalseが返る)

## ・ HViewSetDrawingEnabled

指定Viewの描画可／不可を切り替える (Mac OS X 10.2以降)

```
OSStatus HViewSetDrawingEnabled (  
    HViewRef inView,  
    Boolean inEnabled  
);
```

inView

- ・ 対象となるView

isEnabled

- ・ trueなら描画可にfalseなら描画不可に切り替える

## ・ HViewScrollRect

Viewの内容をスクロールさせる (Mac OS X 10.2以降)

```
OSStatus HViewScrollRect (  
    HViewRef inView,  
    const HRect* inRect<null>,  
    float inDX,  
    float inDY  
);
```

inView

- ・ 対象となるScroll View

nRect<null>

- ・ スクロールさせる矩形枠 (Viewすべてが対象であればNULLを渡す)

inDX

- ・ X方向へのスクロール距離 (プラスは右へマイナスは左へ)

inDY

- ・ Y方向へのスクロール距離 (プラスは下へマイナスは上へ)

## ・ HViewSetZOrder

Viewの順番を変更する (Mac OS X 10.2以降)

```
OSStatus HViewSetZOrder (  
    HViewRef inView,  
    HViewZOrderOp inOp,  
    HViewRef inOther<null>  
);
```

inView

- ・ 順番を変更したいView

inOp

- ・ inOtherの前方なのか後方なのかを決定するオプション

nOther<null>

- ・ 順番変更の基点となるView (NULLであれば全体の最初や最後へ移動)

## ・ HViewReshapeStructure

指定ViewのStructure Regionの変更をシステムに通知する (Mac OS X 10.2以降)

```
OSStatus HViewReshapeStructure (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

## ・ HViewRegionChanged

指定ViewのsRegionの変更をシステムに通知する (Mac OS X 10.3以降)

```
OSStatus HViewRegionChanged (  
    HViewRef inView,  
    HViewPartCode inRegionCode
```

);

inView

- ・ 対象となるView

inRegionCode

- ・ 変更対象となったViewの一部分 (HUIViewPartCode)

## ・ HUIViewCopyShape

指定ViewのsRegionの変更をシステムに通知する (Mac OS X 10.4以降)

```
OSStatus HUIViewCopyShape (
```

```
    HUIViewRef inView,
```

```
    HUIViewPartCode inPart,
```

```
    HShapeRef * outShape
```

```
);
```

inView

- ・ 対象となるView

inPart

- ・ 対象となるViewの一部分 (HUIViewPartCode)

outShape

- ・ 得られたViewの一部分を含んだHShapeRef

## ・ HUIViewGetOptimalBounds

指定Viewの最適サイズと最適テキスト位置を得る (Mac OS X 10.4以降)

```
OSStatus HUIViewGetOptimalBounds (
```

```
    HUIViewRef inView,
```

```
    HIRect * outBounds,  
    float * outBaseLineOffset  
);
```

inView

- ・対象となるView

outBounds

- ・Viewの最適矩形領域サイズ（必要なければNULLを渡す）

outBaseLineOffset

- ・テキストの最適位置（必要なければNULLを渡す）

## ・ HViewFlashDirtyArea

ウィンドウのDirty Regionをフラッシュする（Mac OS X 10.2以降）

```
OSStatus HViewFlashDirtyArea (  
    WindowRef inWindow  
);
```

inWindow

- ・対象となるウィンドウ

## ・ HViewGetWindow

指定されたViewを含むウィンドウ（WindowRef）を得る（Mac OS X 10.3以降）

```
WindowRef HViewGetWindow (  
    HViewRef inView  
);
```

inView



- ・ 対象となるView

## ・ HViewGetFeatures

指定されたViewのFeaturesを得る (Mac OS X 10.3以降)

```
OSStatus HViewGetFeatures (  
    HViewRef inView,  
    HViewFeatures * outFeatures  
);
```

inView

- ・ 対象となるView

outFeatures

- ・ 得られたFeatures (HView.hを参照)

## ・ HViewChangeFeatures

指定されたViewのFeaturesを設定する (Mac OS X 10.3以降)

```
OSStatus HViewChangeFeatures (  
    HViewRef inView,  
    HViewFeatures * inFeaturesToSet,  
    HViewFeatures inFeaturesToClear  
);
```

inView

- ・ 対象となるView

inFeaturesToSet

- ・ セットするFeatures (HView.hを参照)

inFeaturesToClear

- ・ クリアするFeatures (HView.hを参照)

- ・ HViewGetEventTarget

Viewのイベントターゲット (EventTargetRef) を得る (Mac OS X 10.4以降)

```
EventTargetRef HViewGetEventTarget (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

## (4) フォーカスの操作

- ・ HViewGetFocusPart

現在フォーカスされている部分のパートコードを得る (Mac OS X 10.2以降)

```
OSStatus HViewGetFocusPart (  
    HViewRef inView,  
    HViewPartCode* outFocusPart  
);
```

inView

- ・ 対象となるView

outFocusPart

- ・ フォーカスされているViewの部分 (HViewPartCode)

- ・ HViewSetNextFocus

次にキーボードフォーカスを実行するViewを設定する (Mac OS X 10.2以降)

```
OSStatus HViewSetNextFocus (  
    HViewRef inView,  
    HViewRef inNextFocus<null>  
);
```

inView

- ・ 対象となるView

inNextFocus<null>

- ・ 次にフォーカスするViewを指定 (NULLだとデフォルトとなる)

## ・ HViewAdvanceFocus

キーボードフォーカスを次のViewへ移動させる (Mac OS X 10.2以降)

```
OSStatus HViewAdvanceFocus (  
    HViewRef inRootForFocus,  
    EventModifiers inModifiers  
);
```

inRootForFocus

- ・ フォーカス対象のViewが配置されているルートView

inModifiers

- ・ 修飾キーのEventModifiersパラメータ (移動方向を逆転できる)

## ・ HViewSubtreeContainsFocus

同階層のViewにキーボードフォーカスがあるかチェックする (Mac OS X 10.2以降)

```
Boolean HViewSubtreeContainsFocus (  
    HViewRef inSubtreeStart
```

);

inSubtreeStart

- ・ 検索を開始するView
- ・ `HViewSetFirstSubViewFocus`

キーボードフォーカスを最初に受けるViewを設定する (Mac OS X 10.2以降)

```
OSStatus HViewSetFirstSubViewFocus (  
    HViewRef inParent,  
    HViewRef inSubView<null>  
);
```

inParent

- ・ 対象となる親View

inSubView<null>

- ・ 最初に受けるべきView (NULLを指定した場合はデフォルトとなる)

## (5) Viewのイベント処理とヒットテスト

- ・ `HViewClick`

マウスダウンイベントをViewへ送る (Mac OS X 10.2以降)

```
OSStatus HViewClick (  
    HViewRef inView,  
    EventRef inEvent  
);
```

inView

- ・ 対象となるView

inEvent

- ・ 受けた取ったイベント

## ・ HViewSimulateClick

指定View上でマウスクリックの試験を行う (Mac OS X 10.2以降)

```
OSStatus HViewSimulateClick (  
    HViewRef inView,  
    HViewPartCode inPartToClick,  
    UInt32 inModifiers,  
    ControlPartCode* outPartClicked  
);
```

inView

- ・ 対象となるView (これに複数のCarbon Eventを送る)

inPartToClick

- ・ クリックされたViewの部分 (HViewPartCode)

inModifiers

- ・ 修飾キーのEventModifiersパラメータ

outPartClicked

- ・ ヒットした部分のパートコード (返す必要が無い場合はNULLを渡す)

## ・ HViewGetPartHit

指定座標上のViewのパートコードを返す (Mac OS X 10.2以降)

```
OSStatus HViewGetPartHit (  

```

```
HViewRef inView,  
const HIPoint* inPoint,  
HViewPartCode* outPart  
);
```

inView

- ・対象となるView

inPoint

- ・Viewに対する相対座標

outPart

- ・その座標位置のViewの部分 (HViewPartCode)

## ・ HViewGetViewForMouseEvent

マウスイベントを受け取るViewを返す (Mac OS X 10.2以降)

```
OSStatus HViewGetViewForMouseEvent (  
    HViewRef inView,  
    EventRef inEvent,  
    HViewRef* outView  
);
```

inView

- ・検索する階層の最上位View (通常はウィンドウのルートView)

inEvent

- ・対象となるマウスイベント

outView

- ・マウスイベントを受け取ることになるView

## ・ HViewGetSubviewHit

座標指定でViewの子View (Subview) を捜査して返す (Mac OS X 10.2以降)

```
OSStatus HViewGetSubviewHit (  
    HViewRef inView,  
    const HPoint* inPoint,  
    Boolean inDeep,  
    HViewRef* outView  
);
```

inView

- ・ 捜査の対象となる親View

inPoint

- ・ マウス座標 (Viewに対するローカル座標)

inDeep

- ・ trueなら最深層まで捜査しfalseなら階層1レベルの深さで止める

outView

- ・ 捜査された子View (見つからなかったらNULLが返る)

## (6) Viewの座標操作

### ・ HViewGetBounds

Viewの矩形枠を得る (Mac OS X 10.2以降)

```
OSStatus HViewGetBounds (  
    HViewRef inView,  
    HRect* outRect
```

);

inView

- ・対象となるView

outRect

- ・得られたViewの矩形枠（対象Viewのローカル座標となる）

## ・ HViewSetBoundsOrigin

Viewの原点を移動させる（Mac OS X 10.2以降）

```
OSStatus HViewSetBoundsOrigin (  
    HViewRef inView,  
    float inX,  
    float inY  
);
```

inView

- ・対象となるView

inX

- ・移動後のX座標

inY

- ・移動後のY座標

## ・ HViewGetFrame

Viewのフレームを得る（Mac OS X 10.2以降）

```
OSStatus HViewGetFrame (  

```



```
HViewRef inView,  
HRect* outRect  
);
```

inView

- ・対象となるView

outRect

- ・得られたViewのフレーム（親Viewのローカル座標となる）

## ・ HViewSetFrame

Viewのフレームを設定する（Mac OS X 10.2以降）

```
OSStatus HViewSetFrame (  
    HViewRef inView,  
    const HRect* inRect  
);
```

inView

- ・対象となるView

inRect

- ・設定するViewのフレーム（親Viewのローカル座標となる）

## ・ HViewMoveBy

Viewを相対移動させる（Mac OS X 10.2以降）

```
OSStatus HViewMoveBy (  
    HViewRef inView,  
    float inDX,  
    float inDY  
);
```

inView

- ・ 移動の対象となるView

inDX

- ・ X方向移動値（プラスは右へマイナスは左へ）

inDY

- ・ Y方向移動値（プラスは下へマイナスは上へ）

- ・ **HViewConvertPoint**

あるViewのポイントを別のViewのポイントへと変換する（Mac OS X 10.2以降）

```
OSStatus HViewConvertPoint (  
    HIPoint* ioPoint,  
    HViewRef inSourceView,  
    HViewRef inDestView  
);
```

ioPoint

- ・ 座標値を変換するポイント

inSourceView

- ・ 変換前のView（NULLを代入するとウィンドウのローカル座標）

inDestView

- ・ 変換後のView（NULLを代入するとウィンドウのローカル座標）

- ・ **HViewConvertRect**

あるViewの矩形を別のViewの矩形へと変換する（Mac OS X 10.2以降）

```
OSStatus HViewConvertRect (  
    HRect* ioRect,  
    HViewRef inSourceView,  
    HViewRef inDestView  
);
```

ioRect

- ・ 座標値を変換する矩形

inSourceView

- ・ 変換前のView (NULLを代入するとウィンドウのローカル座標)

inDestView

- ・ 変換後のView (NULLを代入するとウィンドウのローカル座標)

## ・ HViewConvertRegion

あるViewのRegionを別のViewのRegionへと変換する (Mac OS X 10.2以降)

```
OSStatus HViewConvertRegion (  
    RgnHandle ioRgn,  
    HViewRef inSourceView,  
    HViewRef inDestView  
);
```

ioRgn

- ・ 座標値を変換するRegion

inSourceView

- ・ 変換前のView (NULLを代入するとウィンドウのローカル座標)

inDestView

- ・変換後のView (NULLを代入するとウィンドウのローカル座標)

## (7) Combo Boxの作成と操作

- ・HIComboBoxCreate

Combo Boxを作成する (Mac OS X 10.2以降)

```
OSStatus HIComboBoxCreate (  
    const HIRect*boundsRect,  
    CFStringRef    text<null>,  
    const ControlFontStyleRec*style<null>,  
    CFArrayRef     list<null>,  
    OptionBits     inAttributes,  
    HViewRef*      outComboBox  
);
```

boundsRect

- ・Combo Boxの矩形枠

text<null>

- ・デフォルト表示テキスト (表示しない場合はNULLを渡す)

style<null>

- ・編集カラムと一覧表示のフォント情報 (NULLならデフォルト)

list<null>

- ・一覧 (Disclosure List) に表示するテキスト (表示しない場合はNULLを渡す)

inAttributes

- ・Comb Boxの初期アトリビュート

outComboBox

- ・作成されたCombo BoxのHViewRef

## ・ HIComboBoxGetAttributes

Combo Boxの属性を得る (Mac OS X 10.2以降)

```
OSStatus HIComboBoxGetAttributes (  
    HViewRef inComboBox,  
    OptionBits* outAttributes  
);
```

inComboBox

- ・対象となるCombo Box

outAttributes

- ・得られた属性

## ・ HIComboBoxChangeAttributes

Combo Boxの属性を変更する (Mac OS X 10.2以降)

```
OSStatus HIComboBoxChangeAttributes (  
    HViewRef inComboBox,  
    OptionBits inAttributesToSet,  
    OptionBits inAttributesToClear  
);
```

inComboBox

- ・対象となるCombo Box

inAttributesToSet

- ・セットする属性

inAttributesToClear

- ・ クリアするアトリビュート

## ・ HIComboBoxAppendTextItem

Combo Boxの一覧 (Disclosure List) にテキストを追加する (Mac OS X 10.2以降)

```
OSStatus HIComboBoxAppendTextItem (  
    HViewRef inComboBox,  
    CFStringRef inText,  
    CFIndex* outIndex<null>  
);
```

inComboBox

- ・ 対象となるCombo Box

RefinText

- ・ 追加する新規項目のテキスト

outIndex<null>

- ・ 追加項目のインデックス値が返る (必要ない場合はNULLを代入する)

## ・ HIComboBoxCopyTextItemAtIndex

Combo Boxの一覧 (Disclosure List) 表示のテキストを得る (Mac OS X 10.2以降)

```
OSStatus HIComboBoxCopyTextItemAtIndex (  
    HViewRef inComboBox,  
    CFIndex inIndex,  
    CFStringRef* outString  
);
```

inComboBox

- ・対象となるCombo Box

inIndex

- ・テキストを指示するためのインデックス値（範囲外だとエラー）

outString

- ・得られたテキスト（コピーされているので不用になったらreleaseの必要あり）

## ・ HIComboBoxGetItemCount

Combo Boxの一覧（Disclosure List）の項目数を返す（Mac OS X 10.2以降）

```
ItemCount HIComboBoxGetItemCount (  
    HViewRef inComboBox  
);
```

inComboBox

- ・対象となるCombo Box

## ・ HIComboBoxInsertTextItemAtIndex

Combo Boxの一覧（Disclosure List）に項目を追加する（Mac OS X 10.2以降）

```
OSStatus HIComboBoxInsertTextItemAtIndex (  
    HViewRef inComboBox,  
    CFIndex inIndex,  
    CFStringRef inText  
);
```

inComboBox

- ・対象となるCombo Box

inIndex

- ・ 項目の追加位置を示すインデックス値（範囲外だと最後に追加される）

inText

- ・ 追加する項目のテキスト

## ・ HIComboBoxRemoveItemAtIndex

Combo Boxの一覧 (Disclosure List) から項目を削除する (Mac OS X 10.2以降)

```
OSStatus HIComboBoxRemoveItemAtIndex (  
    HViewRef inComboBox,  
    CFIndex inIndex  
);
```

inComboBox

- ・ 対象となるCombo Box

inIndex

- ・ 項目の削除位置を示すインデックス値

## ・ HIComboBoxIsListVisible

Combo Boxの一覧 (Disclosure List) の表示をチェックする (Mac OS X 10.4以降)

```
Boolean HIComboBoxIsListVisible (  
    HViewRef inComboBox  
);
```

inComboBox

- ・ 対象となるCombo Box (表示されていればtrueそうでなければfalse)

## ・ HIComboBoxSetListVisible



Combo Boxの一覧 (Disclosure List) の表示をON/OFFする (Mac OS X 10.4以降)

```
OSStatus HIComboBoxIsListVisible (  
    HViewRef inComboBox,  
    Boolean inVisible  
);
```

inComboBox

- ・対象となるCombo Box

inVisible

- ・trueで表示するfalseで非表示にする

## (8) Image Viewの作成と操作

- ・HIImageViewCreate

Image Viewを作成する (Mac OS X 10.2以降)

```
OSStatus HIImageViewCreate (  
    CGImageRef inImage<null>,  
    ControlRef * outControl  
);
```

inImage<null>

- ・表示するCore Graphics Image (NULLを渡すと後からセットすることになる)

outControl

- ・作成されたImage ViewのControlRef

- ・HIImageViewCopyImage

Image ViewからCore Graphics Imageを得る (Mac OS X 10.2以降)

```
CGImageRef HImageViewCopyImage (  
    HViewRef inView  
);
```

inView

- ・ 対象となるImage View
- ・ HImageViewSetImage

Image ViewにCore Graphics Imageをセットする (Mac OS X 10.2以降)

```
OSStatus HImageViewSetImage (  
    HViewRef inView,  
    CGImageRef inImage<null>  
);
```

inView

- ・ 対象となるImage View

inImage<null>

- ・ セットするCore Graphics Image (APIはImageをretainする)

- ・ HImageViewGetAlpha

Image Viewに設定されているアルファ値 (透明度) を得る (Mac OS X 10.2以降)

```
float HImageViewGetAlpha (  
    HViewRef inView  
);
```

inView

- ・ 対象となるImage View

## ・ HllmageViewSetAlpha

Image Viewにアルファ値（透明度）を設定する（Mac OS X 10.2以降）

```
OSStatus HllmageViewSetAlpha (  
    HViewRef inView,  
    float inAlpha  
);
```

inView

- ・ 対象となるImage View

inAlpha

- ・ 設定するアルファ値（透明度 0.0から1.0の範囲）

## ・ HllmageViewSetScaleToFit

Image Viewの矩形枠に合わせてImageが拡大縮小表示されているかどうかチェックする（Mac OS X 10.2以降）

```
Boolean HllmageViewSetScaleToFit (  
    HViewRef inView  
);
```

inView

- ・ 対象となるImage View（返値がtrueならフィット表示でfalseならクリップ表示）

## ・ HllmageViewSetScaleToFit

Image Viewの矩形枠に合わせてImageが拡大縮小表示するように設定する（Mac OS X 10.2以降）

```
OSStatus HllmageViewSetScaleToFit (  
    HViewRef inView,
```

```
Boolean inScaleToFit  
);
```

inView

- ・対象となるImage View

inScaleToFit

- ・trueならフィット表示でfalseならクロップ表示

## ・ HllmageViewIsOpaque

Image Viewが不透明（透明度が1.0）かどうかを調べる（Mac OS X 10.2以降）

```
Boolean HllmageViewIsOpaque (  
    HViewRef inView  
);
```

inView

- ・対象となるImage View（trueが返れば不透明）

## ・ HllmageViewSetOpaque

Image Viewを不透明（透明度が1.0）に設定する（Mac OS X 10.2以降）

```
OSStatus HllmageViewSetOpaque (  
    HViewRef inView,  
    Boolean inOpaque  
);
```

inView

- ・対象となるImage View

inOpaque

- ・ trueを渡せば不透明となる（透明度を変更したい場合はfalseを渡す）

## (9) Scroll Viewの作成と操作

- ・ HIScrollViewCreate

Scroll Viewを作成する（Mac OS X 10.2以降）

```
OSStatus HIScrollViewCreate (  
    OptionBits inOptions,  
    HViewRef* outView  
);
```

inOptions

- ・ オプション（左右と上下、どのスクロールバーを用いるか選択する）

outView

- ・ 作成されたScroll ViewのHViewRef

- ・ HIScrollViewGetScrollBarAutoHide

Scroll Viewにスクロールバーを自動で隠す（Auto-Hide）機能が設定されているかチェックする（Mac OS X 10.2以降）

```
Boolean HIScrollViewGetScrollBarAutoHide (  
    HViewRef inView  
);
```

inView

- ・ 対象となるScroll View（返り値がtureならAuto-Hide機能が設定されている）

- ・ HIScrollViewSetScrollBarAutoHide

Scroll Viewにスクロールバーを自動で隠す (Auto-Hide) 機能を設定する (Mac OS X 10.2以降)

```
OSStatus HIScrollViewSetScrollBarAutoHide (  
    HViewRef inView,  
    Boolean inAutoHide  
);
```

inView

- ・対象となるScroll View

inAutoHide

- ・tureならAuto-Hide機能を設定しfalseならそれを解除する

## ・ HIScrollViewCanNavigate

Scroll Viewのナビゲーションが可能かどうかをチェックする (Mac OS X 10.3以降)

```
Boolean HIScrollViewCanNavigate (  
    HViewRef inView,  
    HIScrollViewAction inAction  
);
```

inView

- ・対象となるScroll View (tureならナビゲーション可能falseならそうではない)

inAction

- ・ナビゲーションの対象となるHIScrollViewAction (詳しくはHView.h参照)

## ・ HIScrollViewNavigate

Scroll Viewの現在の表示位置を変更する (Mac OS X 10.3以降)

```
Boolean HIScrollViewCanNavigate (  
    HViewRef inView,  
    HIScrollViewAction inAction,  
    CGFloat scrollY,  
    CGFloat scrollX
```

```
HViewRef inView,  
HScrollViewAction inAction  
);
```

inView

- ・対象となるScroll View

inAction

- ・表示位置を変更するためのHScrollViewAction（詳しくはHView.h参照）

## (10) Layoutの作成と操作

- ・HViewGetLayoutInfo

指定したViewからレイアウト情報を得る（Mac OS X 10.3以降）

```
OSStatus HViewGetLayoutInfo (  
    HViewRef inView,  
    HLayoutInfo * outLayoutInfo  
);
```

inView

- ・対象となるView

outLayoutInfo

- ・得られたレイアウト情報（HLayoutInfo）

- ・HViewSetLayoutInfo

指定したViewにレイアウト情報を設定する（Mac OS X 10.3以降）

```
OSStatus HViewSetLayoutInfo (  
    HViewRef inView,  
    HLayoutInfo * inLayoutInfo  
);
```

inView

- ・ 対象となるView

inLayoutInfo

- ・ 設定するレイアウト情報 (HLayoutInfo)
- ・ HViewApplyLayout

指定したViewのレイアウト情報を有効にする (Mac OS X 10.3以降)

Applies the current layout to the specified view.

```
OSStatus HViewApplyLayout (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

- ・ HViewSuspendLayout

指定したViewのレイアウト情報の効果を一時停止にする (Mac OS X 10.3以降)

```
OSStatus HViewSuspendLayout (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (このViewに埋め込まれているSubviewも対象となる)



## ・ HViewResumeLayout

指定したViewのレイアウト情報の効果を復帰させる (Mac OS X 10.3以降)

```
OSStatus HViewSuspendLayout (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView

## ・ HViewsLayoutActive

指定したViewのレイアウト情報が有効になっているかどうかをチェックする (Mac OS X 10.3以降)

```
Boolean HViewsLayoutActive (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (tureなら有効であるfalseならそうではない)

## ・ HViewsLayoutLatentlyActive

親Viewがアクティブでない場合に指定したViewのレイアウト情報が有効になっているかどうかをチェックする (Mac OS X 10.4以降)

```
Boolean HViewsLayoutActive (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (tureなら有効であるfalseならそうではない)

## (11) トラッキングエリアの操作

- ・ `HViewNewTrackingArea`

指定したViewに新しいトラッキングエリアを作成する (Mac OS X 10.4以降)

```
OSStatus HViewNewTrackingArea (  
    HViewRef inView,  
    HShapeRef inShape,  
    HViewTrackingAreaID inID,  
    HViewTrackingAreaRef * outRef  
);
```

`inView`

- ・ 対象となるView

`inShape`

- ・ トラッキングエリアの領域 (NULLを指定でStructure Region全体の指定となる)

`inID`

- ・ 新規トラッキングエリアのID (識別子)

`outRef`

- ・ 作成されたトラッキングエリア (Viewの削除と同時に削除される)

- ・ `HViewChangeTrackingArea`

指定されたトラッキングエリアの領域を変更する (Mac OS X 10.4以降)

```
OSStatus HViewChangeTrackingArea (  
    HViewTrackingAreaRef inArea,  
    HShapeRef inShape  
);
```

inArea

- ・対象となるトラッキングエリア

inShape

- ・変更するトラッキングエリアの領域 (NULLを渡すとすべての領域になる)
- ・ HViewGetTrackingAreaID

指定されたトラッキングエリアのIDを得る (Mac OS X 10.4以降)

```
OSStatus HViewGetTrackingAreaID (  
    HViewTrackingAreaRef inArea,  
    HViewTrackingAreaID* outID  
);
```

inArea

- ・対象となるトラッキングエリア

outID

- ・トラッキングエリアのID
- ・ HViewDisposeTrackingArea

指定されたトラッキングエリアを削除する (Mac OS X 10.4以降)

```
OSStatus HViewDisposeTrackingArea (  
    HViewTrackingAreaRef inArea  
);
```

inArea

- ・対象となるトラッキングエリア

## (12) Search Fieldの作成と操作

- ・HISearchFieldCreate

Search Fieldコントロールの作成 (Mac OS X 10.3以降)

```
OSStatus HISearchFieldCreate (  
    const HIRect * inBounds,  
    OptionBits inAttributes,  
    MenuRef inSearchMenu,  
    CFStringRef inDescriptiveText,  
    HViewRef * outRef  
);
```

inBounds

- ・初期矩形枠 (NULLを渡すと空の矩形を指定)

inAttributes

- ・初期アトリビュート

inSearchMenu

- ・サーチ項目用のメニュー (NULLを渡すとメニューは表示しない)

inDescriptiveText

- ・テキスト編集カラムに表示する文字列 (NULLを渡すと表示はなし)

outRef

- ・作成されたSearch FieldコントロールのHViewRef

## ・ HISearchFieldSetSearchMenu

指定されたSearch Fieldにサーチ項目用のメニューを設定する (Mac OS X 10.3以降)

```
OSStatus HISearchFieldSetSearchMenu (  
    HViewRef inSearchField,  
    MenuRef inSearchMenu  
);
```

inSearchField

- ・対象となるSearch Field

inSearchMenu

- ・サーチ項目用のメニュー (NULLを渡すとメニューは表示しない)

## ・ HISearchFieldGetSearchMenu

指定されたSearch Fieldのサーチ項目用のメニューを得る (Mac OS X 10.3以降)

```
OSStatus HISearchFieldGetSearchMenu (  
    HViewRef inSearchField,  
    MenuRef * outSearchMenu  
);
```

inSearchField

- ・対象となるSearch Field

outSearchMenu

- ・得られたサーチ項目用のメニュー (存在しなければNULLが返る)

## ・ HISearchFieldChangeAttributes

指定されたSearch Fieldのアトリビュートを変更する (Mac OS X 10.3以降)

```
OSStatus HISearchFieldChangeAttributes (  
    HViewRef inSearchField,  
    OptionBits inAttributesToSet,  
    OptionBits inAttributesToClear  
);
```

inSearchField

- ・ 対象となるSearch Field

inAttributesToSet

- ・ セットするアトリビュート (詳しくはHView.hを参照)

inAttributesToClear

- ・ クリアするアトリビュート (詳しくはHView.hを参照)

## ・ HISearchFieldGetAttributes

指定されたSearch Fieldのアトリビュートを得る (Mac OS X 10.3以降)

```
OSStatus HISearchFieldGetAttributes (  
    HViewRef inSearchField,  
    OptionBits* outAttributes  
);
```

inSearchField

- ・ 対象となるSearch Field

outAttributes

- ・得られたアトリビュート（詳しくはHView.hを参照）

## ・ HISearchFieldSetDescriptiveText

Search Fieldのテキスト編集カラムに文字列を設定する（Mac OS X 10.3以降）

```
OSStatus HISearchFieldSetDescriptiveText (  
    HViewRef inSearchField,  
    CFStringRef inDescription  
);
```

inSearchField

- ・対象となるSearch Field

inDescription

- ・テキスト編集カラムに文字列を設定する（NULLを渡すと削除）

## ・ HISearchFieldCopyDescriptiveText

Search Fieldのテキスト編集カラムから文字列を得る（Mac OS X 10.3以降）

```
OSStatus HISearchFieldCopyDescriptiveText (  
    HViewRef inSearchField,  
    CFStringRef * outDescription  
);
```

inSearchField

- ・対象となるSearch Field

outDescription

- ・得られたテキスト編集カラムの文字列（無ければNULLが返る）

## (13) メニューの操作

### ・ HIMenuViewGetMenu

指定されたViewに対応するメニュー (MenuRef) を返す (Mac OS X 10.3以降)

```
MenuRef HIMenuViewGetMenu (  
    HViewRef inView  
);
```

inView

- ・ 対象となるView (対象メニューが無い場合はNULLが返る)

### ・ HIMenuGetContentView

描画対象となるMenuのViewを得る (Mac OS X 10.3以降)

```
OSStatus HIMenuGetContentView (  
    MenuRef inMenu,  
    ThemeMenuItemType inMenuItemType,  
    HViewRef * outView  
);
```

inMenu

- ・ 対象となるメニュー (MenuRef)

inMenuItemType

- ・ 対象となるメニューのタイプ (Appearance.hを参照)

outView

- ・ 描画対象となるHViewRef (メニューがHViewモデルでない場合はNULLが返る)



## (14) Segmented Viewの操作

### ・ HISegmentedViewCreate

Segmented Viewを作成する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewCreate (  
    const HIRect * inBounds,  
    HViewRef * outRef)  
);
```

inBounds

- ・ 初期矩形枠 (NULLを渡すと空の矩形を指定)

outRef

- ・ 作成されたSegmented View (HViewRef)

### ・ HISegmentedViewSetSegmentCount

指定されたSegmented Viewのセグメント個数を設定する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentCount (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentCount)  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentCount

- ・ Segmented Viewのセグメント個数

## ・ HISegmentedViewGetSegmentCount

指定されたSegmented Viewのセグメント個数を得る (Mac OS X 10.3以降)

```
UInt32 HISegmentedViewGetSegmentCount (  
    HViewRef inSegmentedView  
);
```

inSegmentedView

- ・ 対象となるSegmented View (Segmented Viewのセグメント個数が返る)

## ・ HISegmentedViewSetSegmentBehavior

Segmented Viewの指定セグメントの挙動を設定する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentBehavior (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    HSegmentBehavior inBehavior  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

inBehavior

- ・ 設定するセグメントの挙動 (HView.hを参照)

## ・ HISegmentedViewGetSegmentBehavior

Segmented Viewの指定セグメントの挙動を得る ( (Mac OS X 10.3以降)

```
HISegmentBehavior HISegmentedViewGetSegmentBehavior (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased  
);
```

inSegmentedView

- ・対象となるSegmented View (セグメントの挙動が返る)

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

## ・ HISegmentedViewChangeSegmentAttributes

Segmented Viewの指定セグメントのアトリビュートを変更 (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewChangeSegmentAttributes (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    OptionBits inAttributesToSet,  
    OptionBits inAttributesToClear  
);
```

inSegmentedView

- ・対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

inAttributesToSet

- ・セットするアトリビュート

inAttributesToClear

- ・ クリアするアトリビュート
- ・ `HISegmentedViewGetSegmentAttributes`

Segmented Viewの指定セグメントのアトリビュートを得る (Mac OS X 10.3以降)

```
OptionBits HISegmentedViewGetSegmentAttributes (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
);
```

inSegmentedView

- ・ 対象となるSegmented View (アトリビュートが返る)

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

- ・ `HISegmentedViewSetSegmentValue`

Segmented Viewの指定セグメントの値を設定する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentValue (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    SInt32 inValue  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

inValue

- ・ 設定するコントロール値

## ・ HISegmentedViewGetSegmentValue

Segmented Viewの指定セグメントの値を得る (Mac OS X 10.3以降)

```
SInt32 HISegmentedViewGetSegmentValue (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased  
);
```

inSegmentedView

- ・ 対象となるSegmented View (コントロール値が返る)

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

## ・ HISegmentedViewSetSegmentEnabled

Segmented Viewの指定セグメントの利用可/不可を変更する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentEnabled (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    Boolean inEnabled  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

isEnabled

- ・ trueであれば利用可となりfalseであれば利用不可

## ・ HISegmentedViewsSegmentEnabled

Segmented Viewの指定セグメントの利用可／不可をチェック (Mac OS X 10.3以降)

```
Boolean HISegmentedViewsSegmentEnabled (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased  
);
```

inSegmentedView

- ・ 対象となるSegmented View (返値がtrueなら使用可でありfalseなら不可)

inSegmentIndexOneBased

- ・ インデックス値 (値は1ペース)

## ・ HISegmentedViewSetSegmentCommand

Segmented Viewの指定セグメントのコマンドIDを設定する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentCommand (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    UInt32 inCommand  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値（値は1ペース）

inCommand

- ・設定するコマンドID
- ・ `HISegmentedViewGetSegmentCommand`

Segmented Viewの指定セグメントのコマンドIDを得る（Mac OS X 10.3以降）

```
UInt32 HISegmentedViewGetSegmentCommand (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased  
);
```

inSegmentedView

- ・対象となるSegmented View（設定済みのコマンドIDが返る）

inSegmentIndexOneBased

- ・インデックス値（値は1ペース）
- ・ `HISegmentedViewSetSegmentLabel`

Segmented Viewの指定セグメントのラベル文字列を設定する（Mac OS X 10.3以降）

```
OSStatus HISegmentedViewSetSegmentLabel (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    CFStringRef inLabel  
);
```

inSegmentedView

- ・対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

inLabel

- ・ラベルとして表示する文文字列 (CFStringRef)

- ・ **HISegmentedViewCopySegmentLabel**

Segmented Viewの指定セグメントのラベル文字列を得る (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewCopySegmentLabel (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    CFStringRef * outLabel  
);
```

inSegmentedView

- ・対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

outLabel

- ・ラベルとして表示されている文字列が返る (CFStringRef)

- ・ **HISegmentedViewSetSegmentContentWidth**

Segmented Viewの指定セグメントの幅を設定する (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentContentWidth (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,
```



```
    Boolean inAutoCalculateWidth,  
    float inWidth  
);
```

inSegmentedView

- ・対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

inAutoCalculateWidth

- ・trueを渡すと自動で計算されfalseの場合にはinWidthの値が使われる

inWidth

- ・セグメント幅 (ピクセル数)
- ・HISegmentedViewGetSegmentContentWidth

Segmented Viewの指定セグメントの幅を得る (Mac OS X 10.3以降)

```
float HISegmentedViewGetSegmentContentWidth (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    Boolean * outAutoCalculated  
);
```

inSegmentedView

- ・対象となるSegmented View (セグメント幅がピクセル値で返る)

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

outAutoCalculated

- ・ trueなら自動で計算されているfalseの場合には設定値が使われている

## ・ HISegmentedViewSetSegmentImage

Segmented Viewの指定セグメントに画像を表示させる (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewSetSegmentImage (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    const HViewImageContentInfo * inImage  
);
```

inSegmentedView

- ・ 対象となるSegmented View

inSegmentIndexOneBased

- ・ インデックス値 (値は1ベース)

inImage

- ・ HViewImageContentInfo構造体にセットされた画像情報

## ・ HISegmentedViewGetSegmentImageContentType

Segmented Viewの指定セグメントに表示された画像の種類を得る (Mac OS X 10.3以降)

```
HViewImageContentType HISegmentedViewGetSegmentImageContentType (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased  
);
```

inSegmentedView

- ・ 対象となるSegmented View (画像の種類が返る)

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

## ・ HISegmentedViewCopySegmentImage

Segmented Viewの指定セグメントに表示された画像を得る (Mac OS X 10.3以降)

```
OSStatus HISegmentedViewCopySegmentImage (  
    HViewRef inSegmentedView,  
    UInt32 inSegmentIndexOneBased,  
    HViewImageContentInfo * ioimage  
);;
```

inSegmentedView

- ・対象となるSegmented View

inSegmentIndexOneBased

- ・インデックス値 (値は1ペース)

ioimage

- ・得られたHViewImageContentInfo構造体 (画像情報)

## (15) Core Graphics Image関連

### ・ HICreateTransformedCGImage

オリジナルから新規にCore Graphics Imageを作成する (Mac OS X 10.4以降)

```
OSStatus HICreateTransformedCGImage (  
    CGImageRef inImage,
```

```
OptionBits inTransform,  
CGImageRef * outImage  
);
```

inImage

- ・オリジナルのCore Graphics Context

inTransform

- ・オプションでDisableやSelectedが選べる（詳しくはHView.hを参照）

outImage

- ・作成されたCore Graphics Image
- ・ **HViewCreateOffscreenImage**

HObjectのサブクラスをレジストする（Mac OS X 10.2以降）

```
OSStatus HViewCreateOffscreenImage (  
    HViewRef inView,  
    OptionBits inOptions,  
    HIRect* outFrame<null>,  
    CGImageRef* outImage  
);
```

inView

- ・Core Graphics Imageを作成したいView

inOptions

- ・オプションビット（現状はゼロのみ）

outFrame<null>

- ・作成されたCore Graphics Imageの矩形枠（必要なければNULLを渡す）

outImage

- ・ 作成されたCore Graphics Image
- ・ `HViewDrawCGImage`

Core Graphics ImageをView上において反転せず正しい向きで描画する (Mac OS X 10.2以降)

```
OSStatus HViewDrawCGImage (  
    CGContextRef inContext,  
    const HIRect* inBounds,  
    CGImageRef inImage  
);
```

inContext

- ・ 描画の対象となるCore Graphics Context

inBounds

- ・ 描画ターゲットの矩形枠

inImage

- ・ 描画するCore Graphics Image

## (16) Grow Box関連

- ・ `HIGrowBoxViewIsTransparent`

Grow Boxが透明表示されているかどうか調べる (Mac OS X 10.2以降)

```
Boolean HIGrowBoxViewIsTransparent (  
    HViewRef inGrowBoxView  
);
```

inGrowBoxView

- ・ 対象となるGrow Box
- ・ HIGrowBoxViewSetTransparent

Grow Boxを透明表示にする (Mac OS X 10.2以降)

```
OSStatus HIGrowBoxViewSetTransparent (  
    HViewRef inGrowBoxView,  
    Boolean inTransparent  
);
```

inGrowBoxView

- ・ 対象となるGrow Box

inTransparent

- ・ trueなら透明表示を設定しfalseならそれを解除する

---

本ドキュメントの履歴

オリジナル2005年9月8日 要約2005年10月15日 v1.00