

GadgetClassD

COLLABORATORS

	TITLE : GadgetClassD		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		March 29, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	GadgetClassD	1
1.1	GadgetClassD.doc	1
1.2	gadget.library/gadget.class()	1
1.3	gadget.library/buttongad.class()	8
1.4	gadget.library/strgad.class()	9
1.5	gadget.library/propgad.class()	10
1.6	gadget.library/lvgad.class()	11
1.7	gadget.library/fvgad.class()	13
1.8	gadget.library/pathgad.class()	15
1.9	gadget.library/txtboxgad.class()	16
1.10	gadget.library/checkboxgad.class()	17
1.11	gadget.library/mxgad.class()	17
1.12	gadget.library/bargad.class()	18
1.13	gadget.library/switchgad.class()	18
1.14	gadget.library/popupgad.class()	19
1.15	gadget.library/grpgad.class()	20
1.16	gadget.library/gfxgad.class()	21

Chapter 1

GadgetClassD

1.1 GadgetClassD.doc

gadget.library

```
bargad.class()  buttongad.class()  checkboxgad.class()  fvgad.class()
gadget.class()  gfxgad.class()    grpgad.class()       lvgad.class()
mxgad.class()   pathgad.class()   popupgad.class()     propgad.class()
strgad.class()  switchgad.class()  txtboxgad.class()
```

1.2 gadget.library/gadget.class()

NAME

gadget.class -- Gadget-Root-Class

FUNCTION

Stellt das Grundverhalten von Gadgets dar.

TAGS

ICLTAG_Left (SLONG) [ISG]

Linke absolute Gadget-Ecke. Die Position wird in gad_LeftEdge vermerkt.

ICLTAG_Top (SLONG) [ISG]

Obere absolute Gadget-Ecke. Die Position wird in gad_TopEdge vermerkt.

ICLTAG_Width (ULONG) [ISG]

Gadget Pixel-Breite. Der Wert wird in gad_Width vermerkt.

ICLTAG_Height (ULONG) [ISG]

Gadget Pixel-Höhe. Der Wert wird in gad_Height vermerkt.

ICLTAG_ID (ULONG) [ISG]

Gadget-ID. Der Wert wird in gad_ID vermerkt.

ACHTUNG: Unter pOS wird die GadgetID nicht als Standard-Message-Code (pOS_IntuiMessage->im_Code) verwendet.

ICLTAG_RelBottom (SLONG) [IS.]
Die Gadgetposition in Y wird relativ zur Window-Höhe gerechnet.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_RelBottom)
Der System-Layouter kann nur absolute Koordinaten berechnen.

ICLTAG_RelRight (SLONG) [IS.]
Die Gadgetposition in X wird relativ zur Window-Breite gerechnet.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_RelRight)
Der System-Layouter kann nur absolute Koordinaten berechnen.

ICLTAG_RelWidth (SLONG) [IS.]
Die Gadget-Breite paßt sich an die Window-Breite an.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_RelWidth)
Der System-Layouter kann nur absolute Koordinaten berechnen.

ICLTAG_RelHeight (SLONG) [IS.]
Die Gadget-Höhe paßt sich an die Window-Höhe an.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_RelHeight)
Der System-Layouter kann nur absolute Koordinaten berechnen.

ICLTAG_Immediate (BOOL) [IS.]
Das Gadget sendet beim GCLMTH_GoActive (Mausklick ins Gadget) eine IDCMP_GadgetDown-Message ans Window.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_Immediate)

ICLTAG_RelVerify (BOOL) [IS.]
Wird die Gadget-Eingabe beendet GCLMTH_HandleInput + GCLMTHR_InputDone, so sendet das System ans Window eine IDCMP_GadgetUp-Message.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_RelVerify)

ICLTAG_UpdateGadget (BOOL) [IS.]
Eine Veränderung im Gadget, während das Gadget aktiv ist, kann als IDCMP_UpdateGadget-Message ermittelt werden.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_UpdateGadget)
In der Empfangenen IDCMP-Message kann aus dem im_Code ermittelt werden, was im Gadget verändert wurde. (enum pOS_InputEventGdUpCode)

ICLTAG_RightBorder (BOOL) [IS.]
Das Gadget befindet sich im Window-Border (rechts).
Dem System muß dieser Sachverhalt mitgeteilt werden, damit aus dem Window-Border eine Fläche herausgeschnitten werden kann.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_RightBorder)

ICLTAG_LeftBorder (BOOL) [IS.]
Das Gadget befindet sich im Window-Border (links).
Dem System muß dieser Sachverhalt mitgeteilt werden, damit aus dem

Window-Border eine Fläche herausgeschnitten werden kann.
 Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
 (gad_Activation & GACT_LeftBorder)

ICLTAG_TopBorder (BOOL) [IS.]

Das Gadget befindet sich im Window-Border (oben).
 Dem System muß dieser Sachverhalt mitgeteilt werden, damit aus dem Window-Border eine Fläche herausgeschnitten werden kann.
 Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
 (gad_Activation & GACT_TopBorder)

ICLTAG_BottomBorder (BOOL) [IS.]

Das Gadget befindet sich im Window-Border (unten).
 Dem System muß dieser Sachverhalt mitgeteilt werden, damit aus dem Window-Border eine Fläche herausgeschnitten werden kann.
 Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
 (gad_Activation & GACT_BottomBorder)

ICLTAG_AnimateGadget (BOOL) [IS.]

Ist das Gadget selektiert (GFLG_Selected) und dieser Tag==TRUE, dann wird das IObjekt als Animation behandelt. (pIntui-Ticks)
 Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
 (gad_Activation & GACT_AnimateGadget)

ICLTAG_SysGadgetType (ULONG) [IS.]

Der Gadget-Type wird gesetzt. (gad->gad_Type)

ICLTAG_GadBorderType (ULONG) [ISG] (enum pOS_GadgetBorderTypes)

Um das Gadget wird ein Border (Rahmen) gezeichnet.
 Beim Setzen vom Type wird gad_BorLeft, gad_BorTop, gad_BorRight, gad_BorBottom berechnet. Die Berechnung erfolgt mittels pOS_CalcBorderBox().
 Das Gadget kann seine eigene Größe innerhalb durch pOS_CalcGadgetInnerBox() berechnen. Die innere Größe ist die Fläche, in der das Gadget sich zeichnen darf. pOS_CalcGadgetBox() berechnet die gesamte Gadget-Fläche (mit Border).
 pOS_CalcGadget...() berechnet immer absolute Window/Layer - Koors.

Border-Typen:

GADBORTYP_None	- keinen Border
GADBORTYP_BlkJBox	- einfache schwarze Umrandung
GADBORTYP_BevelUpBox	- 3D Aussehen, hochgestellt
GADBORTYP_BevelDownBox	- 3D Aussehen, tiefgestellt
GADBORTYP_RBevelUpBox	- 3D Aussehen, hochgestellt, rund
GADBORTYP_RBevelDownBox	- 3D Aussehen, tiefgestellt, rund
GADBORTYP_BevelBox	- 3D Aussehen, hoch-/tiefgestellt
GADBORTYP_HfBevelUpBox	- 3D Aussehen, hochgestellt (halbhell)
GADBORTYP_HfBevelDownBox	- 3D Aussehen, tiefgestellt (halbhell)
GADBORTYP_HfRBevelUpBox	- 3D Aussehen, hochgestellt/rund (halbhell)
GADBORTYP_HfRBevelDownBox	- 3D Aussehen, tiefgestellt/rund (halbhell)
GADBORTYP_XBox	- verzahnte Box für Gruppen-Rahmen
GADBORTYP_ZBox	- B-W-B für Boxen mit weißem Inhalt
GADBORTYP_BevelUpBox	- wie GADBORTYP_BevelUpBox, aber mit 1 Pixel 'Luft'

ICLTAG_ToggleSelect (BOOL) [ISG]
Gadget wechselt bei jeder Aktivierung den Selektzustand.
Das Gadget-Verhalten ist weiterhin abhängig von ICLTAG_Immediate und ICLTAG_RelVerify.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_ToggleSelect)

ICLTAG_Selected (BOOL) [ISG]
Gadget ist selektiert (ausgewählt).
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_Selected)

ICLTAG_Disabled (BOOL) [ISG]
Das Gadget ist gesperrt und ist an einem Gitter zu erkennen.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_Disabled)

ICLTAG_TabCycle (BOOL) [IS.]
Die TAB-Taste aktiviert das nächste TAB-Gadget.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_TabCycle)

ICLTAG_Layout (BOOL) [IS.]
Der Gadget-Layerrouter wird angewiesen, dem Gadget die Methode GCLMTH_Layout zu senden.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_UseMthLayout)

ICLTAG_NoSelect (BOOL) [IS.]
Jede Aktivierung vom Gadget wird nicht als Selektierung interpretiert, dennoch wird eine IDCMP_GadgetDown-Message erzeugt.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_NoSelect)

ICLTAG_FixSize (BOOL) [IS.]
Der Gadget-Layerrouter darf diesem Gadget nur seine Nominal-Größe zuteilen. Jede dynamische Vergrößerung ist untersagt.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_FixSize)
Siehe ICLTAG_Gwk

ICLTAG_StandardDrop (BOOL) [IS.]
Beim Drop ins Gadget wird standardmäßig verfahren.
- Drop-Test
- Highlight Border
- Event-Handling
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_StandardDrop)

ICLTAG_StandardDrag (BOOL) [IS.]
Beim Drag vom Gadget wird standardmäßig verfahren.
- Drag-Test
- Create Drag-Object
- Event-Handling
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Activation & GACT_StandardDrag)

ICLTAG_HighlightDropBox (BOOL) [IS.]

Beim Drop ins Gadget wird der Border hervorgehoben gezeichnet.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad_Flags & GFLG_HighlightDropBox)

ICLTAG_FixFocus (BOOL) [IS.]

Der Eingabefokus darf nicht mit einem Funktionsaufruf vom Gadget abgezogen werden. Der User muß selbständig durch Mausklick den Fokus ändern.
Dieses Flag macht Sinn bei String-Gadget, damit während der Text-eingabe der User nicht unterbrochen wird.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Activation & GACT_FixFocus)
Mit pos_InactivateGadget() kann der 'FixFocus' durchbrochen werden.

ICLTAG_IgnoreIE (BOOL) [IS.]

Das System unterdrückt alle InputEvents und somit reagiert das Gadget nicht. Dieser Modus ist für schnelle synchrone Prozesse oder zum gezielten unterdrücken von Nachrichten, bei denen das Gadget nicht 'Disabled' werden muß, gedacht.
Eine Gadget-Gruppe reicht den Tag an alle Members weiter.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_IgnoreIE)

ICLTAG_PrivateDisable (BOOL) [IS.] Default=FALSE

Immer wenn dieses Flag gesetzt ist, wird vom System kein Gitter über das Gadget gezeichnet. Es bleibt dem Gadget überlassen, wie es sich als 'disabled' zeichnet.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_PrivateDisable)

ICLTAG_StdTargetMap (BOOL) [IS.] Default=FALSE

Die im Gadget vermerkte Map-Liste darf vom System autom. ausgelesen werden. (gad_Map)
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_StdTargetMap)

ICLTAG_Layer (pos_Layer*) [IS.]

Layer in dem das Gadget gezeichnet werden soll. (gad_Layer)
Eine Gadget-Gruppe reicht den Tag an alle Members weiter.

ICLTAG_Gwk (UBYTE) [IS.] Default=0

Layout-Gewichtung in einer Gadget-Gruppe. (gad_Gwk)
Je höher das Gewicht, desto mehr Fläche nimmt das Gadget (falls das Gadget dies kann) ein.
Ein Gewicht von '0' bedeutet, daß das Gadget nur seine nominale Größe einnimmt, nur in Sonderfällen wird dies unterlaufen.
Siehe ICLTAG_FixSize

ICLTAG_ShortKey (CHAR) [IS.]

Wird der ShortKey von der Root-Class (gadget.class) entdeckt:
(gad_ShortKey)
- Gadget ist ein String-Gadget (gad_Type) => Gadget wird aktiviert
- Gadget ist ToggleSelect => Selektierung wird geändert
- Gadget ist RelVerify => Gad wird kurz als selektiert gezeichnet und die Message wird als IDCMP_GadgetUp umgewandelt.

- Gadget ist Immediate => Message wird als IDCMP_GadgetDown umgewandelt

ICLTAG_Target (APTR) [IS.]
Falls ICLTAG_StdTargetMap==TRUE ist wird das System die Map-Liste an 'Target' senden. (gad_Target)

ICLTAG_Map (const pOS_TagItem) [IS.]
Map-Liste setzen. (gad->gad_Map)

ICLTAG_HelpID (const CHAR*) [IS.]
Help-String-ID des Gadgets. Das System wird bei jedem Help-Event diesen String als Pfadanteil interpretieren und im Help-File danach suchen.

ICLTAG_LableLayMode (UBYTE) [IS.] (enum pOS_IObjectClassLayout)
Layout-Vorschrift für das Gadget-Lable. (gad_LableLay)

ICLTAG_RenderLayMode (UBYTE) [IS.] (enum pOS_IObjectClassLayout)
Layout-Vorschrift für den Gadget-Innhalt. (gad_RenderLay)

enum pOS_IObjectClassLayout:

- IOBLAY_HVCenter - Horizontal und Vertikal zentiert
- IOBLAY_Left - linksbündig
- IOBLAY_Right - rechtsbündig
- IOBLAY_Top - oben-bündig
- IOBLAY_Bottom - unten-bündig
- IOBLAY_DynWidth - gesamte Breite ausfüllen
(schließt IOBLAY_Left, IOBLAY_Right aus)
- IOBLAY_DynHeight - gesamte Höhe ausfüllen
(schließt IOBLAY_Top, IOBLAY_Bottom aus)

ICLTAG_MutualExclude (UBYTE) [IS.]
Ist das Gadget eine Gruppe, dann wird gad_MutualExclude als Bitmaske interpretiert (gad_MutualExclude==0 => ohne Auswirkung)
Das System arbeitet nach folgender Regel:
Ein gesetzten Bit in der Bitmaske de-selektiert das Member-Gadget.
Z.B Maske 0x05 %00000101 => Bit 0 und Bit 2 sind gesetzt
Die Member-Gadget an den Positionen 0, 2 werden de-selektiert.

ICLTAG_Flags (ULONG) [IS.] (enum pOS_GadgetFlags)
Gadget-Flags (gad_Flags)
ACHTUNG: Die Flags werden überschrieben.

ICLTAG_UserData (ULONG) [IS.]
Gadget-Userdaten. Das System verändert bzw. liest diese Daten NIE. (gad_UserData[0])

ICLTAG_SpecialInfo (APTR) [IS.]
Gadget eigener Zeiger auf private bzw. öffentliche Daten.
(gad_SpecialInfo)

ICLTAG_Render (pOS_IntuiObj*) [IS.]
Darzustellendes Objekt. (gad_Render)
Ist gad_Select==NULL, so wird gad_Render ebenfalls zum Zeichen des selektierten Gadgets verwendet.

ICLTAG_SelectRender (pOS_IntuiObj*) [IS.]
Selektierte Gadget-Darstellung. (gad_Select)
Siehe ICLTAG_Render

ICLTAG_Callback (pOS_Callback*) [IS.]
Gadget-Callback (gad_Callback)

ICLTAG_StdCallback (BOOL) [IS.]
Markiert gad_Callback als Standard-Callback.
Das System bearbeitet ausschließlich Standard-Callbacks.
Private Gadgets können gad_Callback für ihre eigenen
Zwecke einsetzen.
Im Gadget wird das Verhalten durch ein Flag gekennzeichnet.
(gad->gad_Flags & GFLG_StdCallback)

ICLTAG_AutoDelete (BOOL) [IS.] Default=FALSE, außer in Gruppen
Das Gadget wird bei pOS_CloseWindow() autom. gelöscht.
Ist ein Gadget Member einer Gruppe (Mitglied), so ist es immer
'auto-delete'. Die Gadget-Gruppe löscht immer und alles.

ICLTAG_ShortKeyUSStr (const CHAR*) [IS.]
Aus dem Textstring wird der ShortKey mittels '_' gesucht und
im Gadget vermerkt. (gad_ShortKey)
Z.B "H_allo" => gad_ShortKey='a'

ICLTAG_LableTags (const pOS_TagItem*) [IS.]
Der Gadget-Lable wird über Tags erzeugt. (gad_Lable)
=pOS_NewIObjectA(
 ICLTAG_LableNClass (pOS_NClass*)
 ICLTAG_LableClassName (const CHAR*)
 ICLTAG_LableNCVer (ULONG)
 Tags von ICLTAG_LableTags
);

ICLTAG_RenderTags (const pOS_TagItem*) [IS.]
Der Gadget-Render wird über Tags erzeugt. (gad_Render)
=pOS_NewIObjectA(
 ICLTAG_RenderNClass (pOS_NClass*)
 ICLTAG_RenderClassName (const CHAR*)
 ICLTAG_RenderNCVer (ULONG)
 Tags von ICLTAG_RenderTags
);

ICLTAG_SelectTags (const pOS_TagItem*) [IS.]
Der Gadget-Select wird über Tags erzeugt. (gad_Select)
=pOS_NewIObjectA(
 ICLTAG_SelectNClass (pOS_NClass*)
 ICLTAG_SelectClassName (const CHAR*)
 ICLTAG_SelectNCVer (ULONG)
 Tags von ICLTAG_SelectTags
);

ICLTAG_StdLableName (const CHAR*) [IS.]
Es wird ein Gadget-Lable als Text-String erzeugt.
Beinhaltet der String ein '_', so wird das nächste
Zeichen als ShortKey interpretiert.

Liegt noch keine Lable-Layout-Vorschrift vor, so wird immer IOBLAY_Left verwendet.

ICLTAG_AddDropAttName (const CHAR*) [IS.]
 Definiert die Eigenschaften, auf die das Gadget beim Drop reagiert.
 Es dürfen beliebig viele ICLTAG_AddDropAttName-Tags angegeben werden.
 Der String wird in einen privaten Puffer kopiert.

ICLTAG_AddDragAttName (const CHAR*) [IS.]
 Definiert die Eigenschaften, die das Gadget beim Drag weitergibt.
 Es dürfen beliebig viele ICLTAG_AddDragAttName-Tags angegeben werden.
 Der String wird in einen privaten Puffer kopiert.

SEE ALSO

pGadget/Gadget.h, pIntui/OClass.h, pIntui/Tags.h

1.3 gadget.library/buttongad.class()

NAME

buttongad.class -- Text-Bool-Gadget

Basis-class [gadget.class]

DrawInfo [SCRNAM_GButtonClass]

FUNCTION

Bool-Gadget mit Text. Der Border ist bereits gesetzt.
 Normalerweise muß ICLTAG_RelVerify zusätzlich angegeben werden.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
 Referenzdaten zum Zeichnen.

ICLTAG_AddWidth (SLONG) [IS.] Default=8
 Optionale Gadget-Vergrößerung

ICLTAG_AddHeight (SLONG) [IS.] Default=8
 Optionale Gadget-Vergrößerung

IOBJTAG_FGPen (UBYTE) [IS.]
 Zeichen-Pen im de-selektierten Zustand.
 Default => wird aus ICLTAG_DrawInfo ermittelt.

IOBJTAG_FGSelPen (UBYTE) [IS.]
 Zeichen-Pen im selektierten Zustand.
 Default => wird aus ICLTAG_DrawInfo ermittelt.

IOBJTAG_TextAttr (const pOS_TextAttr*) [IS.]
 Font-Daten für den Text.
 Default => Window-Font

IOBJTAG_Underscore (CHAR) [IS.] Default='_'
 Definiert, welches Zeichen als Einleiter für markierte

Zeichen dient.

IOBJTAG_StaticString (const CHAR*) [IS.]
 Darzustellender Text-String, der NICHT kopiert wird.
 Der String MUSS über die Lebensdauer des Gadgets bestehen.

IOBJTAG_String (const CHAR*) [IS.]
 Darzustellender Text-String, der kopiert wird.
 Der String wird nur beim Setzen benötigt.

SEE ALSO
 gadget.class

1.4 gadget.library/strgad.class()

NAME

strgad.class -- String-Gadget

Basis-class [gadget.class]
 DrawInfo [SCRNAM_GStrClass]

FUNCTION

String-Gadget zur Eingabe einer Zeile.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
 Referenzdaten zum Zeichnen.

STRGADTAG_String (const CHAR*) [ISG]
 Text-String im Gadget.

STRGADTAG_MaxChars (UWORD) [ISG] Default=64
 Anzahl der zu editierenden Zeichen.

STRGADTAG_DisPos (UWORD) [ISG]
 Startposition des Strings.

STRGADTAG_StdDragAtt (BOOL) [IS.]
 Erlaubt Drag in der Standard-Form. (ascii)

STRGADTAG_StdDropAtt (BOOL) [IS.]
 Erlaubt Drop in der Standard-Form. (ascii)

STRGADTAG_CharFilter (const CHAR*) [IS.]
 Nur die Zeichen, die im CharFilter-Array vorkommen
 werden als Eingabe zugelassen. Das Array muß mit einem
 NULL-Byte enden.

STRGADTAG_Integer (SLONG) [ISG]
 Interpretiert den String als Integer-Wert.

SEE ALSO

gadget.class

1.5 gadget.library/propgad.class()

NAME

propgad.class -- Proportional-Gadget

Basis-class [gadget.class]

DrawInfo [SCRNAM_GPropClass]

FUNCTION

Schieber für beliebige Richtungen.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]

Referenzdaten zum Zeichnen.

PRGADTAG_Flags (ULONG) [IS.] (enum pOS_PropGadFlags)

PRGADF_FreeHoriz - Schieber kann sich horizontal bewegen

PRGADF_FreeVert - Schieber kann sich vertikal bewegen

PRGADTAG_HorizPot (UWORD) [IS.]

Horizontale Position in 16Bit.

0x0000 => links

0xffff => rechts

PRGADTAG_HorizBody (UWORD) [IS.]

Horizontale Größe des Schiebers in 16Bit.

0x0000 => minimale Größe

0xffff => füllt die gesamte Fläche aus

PRGADTAG_VertPot (UWORD) [IS.]

Vertikale Position in 16Bit.

0x0000 => links

0xffff => rechts

PRGADTAG_VertBody (UWORD) [IS.]

Vertikale Größe des Schiebers in 16Bit.

0x0000 => minimale Größe

0xffff => füllt die gesamte Fläche aus

PRGADTAG_Total (ULONG) [IS.]

Anzahl aller Elemente, die verwaltet werden.

Siehe PRGADTAG_Visible, PRGADTAG_Top

PRGADTAG_Visible (ULONG) [IS.]

Anzahl der Elemente, die sichtbar sind.

Siehe PRGADTAG_Total, PRGADTAG_Top

PRGADTAG_Top (ULONG) [ISG]

Erstes sichtbares Element [0,1,2,3...].

PRGADTAG_Fmt (const CHAR*) [IS.]

RawDoFmt-Format-Schablone.
 Zusätzlich muß PRGADTAG_CrtTxtObj=TRUE angegeben werden.
 Z.B.: "Size %ld"
 Im Schieber wird der aktuelle Wert angezeigt.

PRGADTAG_IntArray (const SLONG*) [IS.]
 Der aktuelle Wert im Gadget wird als Index ins SLONG-Array verwendet.

PRGADTAG_TxtArray (const CHAR**) [IS.]
 Mit z.B. PRGADTAG_Fmt="%s" wird der aktuelle Index im TxtArray angezeigt.

PRGADTAG_CrtTxtObj (BOOL) [IS.]
 Erzeugt ein anzeigbares Text-Objekt für PRGADTAG_Fmt.

SEE ALSO
 gadget.class

1.6 gadget.library/lvgad.class()

NAME

lvgad.class -- ListView-Gadget

Basis-class [gadget.class]
 DrawInfo [SCRNAM_GLVClass]

FUNCTION

Gadget zum Darstellen von Listen mit beliebigen IObjekten,

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
 Referenzdaten zum Zeichnen.

IOBJTAG_BGPen (UBYTE) [IS.]
 Hintergrund-Pen vom Gadget setzen.

LVGADTAG_List (struct pOS_List*) aus pOS_GadgetItem [ISG]
 Zeiger auf die darzustellende Liste.
 NULL nimmt die Liste aus der Gadget-Verwaltung, jedoch wird der grafische Inhalt nicht gelöscht.
 Die Liste MUSS zu jederzeit gültig sein. Leere Listen sind erlaubt.

LVGADTAG_TopItem (pOS_GadgetItem*) [ISG]
 Definiert den ersten anzuzeigenden Item.
 Ist zu diesem Zeitpunkt die Liste (LVGADTAG_List) NULL, so wird der Tag ignoriert. LVGADTAG_TopItem==NULL wird ignoriert.

LVGADTAG_MultiSelect (BOOL) [IS.] default=FALSE
 Erlaubt die Auswahl mehrerer Items.

LVGADTAG_ReadOnly (BOOL) [IS.] default=FALSE
Das Gadget reagiert nicht auf Eingaben.

LVGADTAG_AlwaysOne (BOOL) [IS.] default=FALSE
Es wird versucht, immer einen Item aktiv zu halten.

LVGADTAG_Space (SLONG) [IS.] default=1
Leerraum zwischen den Items in vertikaler Richtung.

LVGADTAG_CursorScroll (BOOL) [IS.] default=FALSE
Mit den Cursor-Tasten kann die Liste in alle Richtungen verschoben werden. (Gadget muß aktiv sein)

LVGADTAG_CursorSelect (BOOL) [IS.] default=FALSE
Mit den Cursor-Tasten (Up/Down) wird ein Item angewählt. (Gadget muß aktiv sein)

LVGADTAG_SelectItem (pOS_GadgetItem*) [ISG]
Aktiver Item setzen bzw. ermitteln.
Ist zu diesem Zeitpunkt die Liste (LVGADTAG_List) NULL, so wird der Tag ignoriert.
Wird LVGADTAG_SelectItem=NULL gesetzt, wird kein Item aktiviert.

LVGADTAG_VSliderGad (pOS_Gadget*) [IS.]
Dem Gadget wird ein Zeiger auf ein PropGadget mitgeteilt, das als Vertikal-Schieber dienen soll.
Das PropGadget wird beim Entfernen ebenfalls gelöscht.

LVGADTAG_HSliderGad (pOS_Gadget*) [IS.]
Dem Gadget wird ein Zeiger auf ein PropGadget mitgeteilt, das als Horizontal-Schieber dienen soll.
Das PropGadget wird beim Entfernen ebenfalls gelöscht.

LVGADTAG_Key1 (BOOL) default=FALSE [IS.]
Das Gadget werten weitere Events aus.
Doppelklick => IDCMP_GadgetUp + IEGUCODE_Ok

LVGADTAG_FirstDrawItem (const pOS_GadgetItem*) [IS.]
Nächster Gadget-Refresh zeichnet erst ab diesem Item.

LVGADTAG_LastDrawItem (const pOS_GadgetItem*) [IS.]
Nächster Gadget-Refresh zeichnet nur bis zu diesem Item.

LVGADTAG_HasNewNodes (BOOL) [IS.]
Wird die Liste erweitert bzw. verkleinert, so müssen die Schieber und andere Daten neu berechnet werden.
TRUE => Daten neu berechnen.

LVGADTAG_VAutoGad (ULONG) [I..]
Das Gadget erzeugt selbständig einen Vertikal-Schieber.

LVGADTAG_CalcSlider (BOOL) [I..]
Nur die Schieber neu berechnen.

LVGADTAG_HAutoGad (ULONG) [IS.]
Das Gadget erzeugt selbständig einen Horizontal-Schieber.

```

LVGADTAG_HAutoKt (ULONG) [IS.]
    Layout von verketteten IObjekten berechnen.
    LVGADTAG_HAutoKt=0 => ausgeschaltet
    LVGADTAG_HAutoKt=1 => eingeschaltet

    pOS_IntuiObj* IObj;
    for(IObj=pOS_GadgetItem->gdt_Render; IObj; IObj=IObj->iobj_Next) ;

LVGADTAG_HCursorStep (ULONG) [IS.] Default=4
    Pixelverschiebung beim Cursor-Scroll in horizontaler Richtung

LVGADTAG_DeselectItem (pOS_GadgetItem*) [IS.]
    De-Selektiert einen Item. Ist der Item nicht selektiert, wird der
    Tag ignoriert.

LVGADTAG_InvalidItem (pOS_GadgetItem*) [IS.]
    Item aus der internen Verwaltung entfernen.
    Wird ein Item aus der Liste entfernt und gelöscht, darf kein
    Zeiger im Gadget auf den Item zurückbleiben.

LVGADTAG_RedrawItem (const pOS_GadgetItem*) [IS.]
    Gezieltes zeichnen eines Items.

LVGADTAG_ResetKetArray (BOOL) [IS.]
    Interne Berechnung von verketteten IObjekten zurücksetzen.
    (LVGADTAG_HAutoKt)

LVGADTAG_DragItem (const pOS_GadgetItem*) [..G]
    Nach dem erfolgreichen Drag eines Items aus dem Gadget,
    kann der betreffende Item ermittelt werden.

LVGADTAG_ViewItem (const pOS_GadgetItem*) [IS.]
    Liste so verschieben, daß der betreffende Item sichtbar wird.

LVGADTAG_CR (BOOL) [IS.]
    Das Gadget werten weitere Events aus.
    RETURN-Taste => IDCMP_GadgetUp + IEGUCODE_Ok

LVGADTAG_ESC (BOOL) [IS.]
    Das Gadget werten weitere Events aus.
    ESC-Taste => IDCMP_GadgetUp + IEGUCODE_Cancel

```

SEE ALSO
gadget.class

1.7 gadget.library/fvgad.class()

NAME
fvgad.class -- FileView-Gadget

Basis-class [lvgad.class]
DrawInfo [SCRNAM_GFVClass]

FUNCTION

Gadget zum Darstellen von File-Listen.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]

Referenzdaten zum Zeichnen.

FVGADTAG_Lock (pOS_FileLock*) [ISG]

Lock auf das zu lesende Verzeichnis.

NULL - Aktuell laufender Scan-Process wird abgebrochen und
alle Objekte werden gelöscht.

~0 - Aktuell laufender Scan-Process wird abgebrochen und
alle Objekte werden gelöscht.
Das Löschvorgang wird optisch nicht angezeigt.

[G] - Es wird ein Lock-Duplikat zurückgegeben. Dieser neue Lock MUSS
von der Applikation mit pOS_UnlockObject() freigegeben werden.

FVGADTAG_TargetAddPathGad (pOS_Gadget*) [ISG]

Das fv-Gadget kann den Pfad an ein weiteres Gadget weitergeben.
(pathgad.class)

FVGADTAG_TargetAddFileGad (pOS_Gadget*) [ISG]

Das fv-Gadget kann ein File an ein weiteres Gadget weitergeben.
(strgad.class)

FVGADTAG_ScanStatus (ULONG*) [..G]

Ermittelt den aktuellen internen Scan-Status.

Bit 0 set - der Scan-Process läuft im Hintergrund.

FVGADTAG_RemItem (pOS_FVGadItem*) [IS.]

Entfernt und löscht ein Item.

Das Item MUSS Member der aktuellen Liste (LVGADTAG_List)
sein. FVGADTAG_RemItem==NULL - keine Reaktion

FVGADTAG_AddItem (const pOS_FVGadItem*) [IS.]

Siehe FVGADTAG_QuickAddItem

FVGADTAG_QuickAddItem (const pOS_FVGadItem*) [IS.]

Erzeugt eine Kopie des Items und fügt diesen in die Liste ein.
Bei FVGADTAG_QuickAddItem wird der neue Node nicht sofort in die
Anzeigeliste übernommen. Mittels ti_Data==~0 wird nur die
Anzeigeliste erstellt.
fgt_Name => enthält den Namen

FVGADTAG_QuickAddItem = NULL - keine Reaktion

FVGADTAG_QuickAddItem = ~0 - alle Quick-Items bearbeiten

FVGADTAG_Pattern (const dosname_t*) [IS.] default = "#?"

Definiert den Scan-Filter mittels Patternmatching.

FVGADTAG_Pattern = NULL - entspricht "#?"

FVGADTAG_MultiAssign (BOOL) [IS.] default = FALSE

Multi-Assigns werden autom. eingelesen.

FVGADTAG_ReScan (BOOL) [IS.]

Das aktuelle Verzeichnis wird neu eingelesen.

```

FVGADTAG_MatchFiles (BOOL) default=TRUE [IS.]
    Patternmatching auf Dateien anwenden.

FVGADTAG_MatchDirs (BOOL) [IS.] default=FALSE
    Patternmatching auf Verzeichnisse anwenden.

FVGADTAG_Files (BOOL) [IS.] default=TRUE
    Dateien werden angezeigt und bearbeitet.

FVGADTAG_Dirs (BOOL) [IS.] default=TRUE
    Verzeichnisse werden angezeigt und bearbeitet.

FVGADTAG_FilterInfos (BOOL) [IS.] default=FALSE
    Info-Dateien werden herausgefiltert.

FVGADTAG_AutoAddDir (BOOL) [IS.] default=FALSE
    Verzeichnisse werden durch Maus-Klick eingelesen.

FVGADTAG_DevList (ULONG) [ISG]
    Anzeigemodus im Gadget.
    FVGADTAG_DevList=0 - normale Datei-Liste
    FVGADTAG_DevList=1 - Liste der Geräte und Assigns
    FVGADTAG_DevList=2 - wechselt zwischen 0 und 1
    FVGADTAG_DevList=3 - wie 0, jedoch ohne re-scan

FVGADTAG_ViewFileItem (const dosname_t*) [IS.]
    Sucht den nächsten passenden File-Item und verschiebt die
    Liste so, daß dieser sichtbar wird.
    FVGADTAG_ViewFileItem = "a" - sucht den ersten Item, der mit "a"
    beginnt.

FVGADTAG_SelectFileItem (const dosname_t*) [IS.]
    Selektiert den File-Item mit passenden Namen.

FVGADTAG_DbCkFile (BOOL) [IS.] default=FALSE
    Ein Doppelklick auf 'File' bestätigt die Eingabe. (IDCMP_GadgetUp)

FVGADTAG_DbCkDir (BOOL) [IS.] default=FALSE
    Ein Doppelklick auf 'Dir' bestätigt die Eingabe. (IDCMP_GadgetUp)

```

SEE ALSO
 gadget.class

1.8 gadget.library/pathgad.class()

```

NAME
    pathgad.class -- FileView-Gadget

Basis-class [strgad.class]
DrawInfo [ SCRNAM_GPathClass ]

```

FUNCTION
 Gadget zum Darstellen von Dos-Pfaden.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

STRGADTAG_String (const dosname_t*) [ISG]
Gadget-String setzen bzw. auslesen.
Wird ein neuer String (Path) gesetzt, erzeugt das Gadget einen FileLock.

PATHGADTAG_TargetFVGad (pOS_Gadget*) [IS.]
Verbindet das Gadget mit einem (fvgad.class) - Gadget.

PATHGADTAG_DoParent (BOOL) [IS.]
Setzt den Pfad um ein Verzeichnis zurück.

PATHGADTAG_AddPath (const dosname_t*) [IS.]
Addiert den neuen Pfad zum Gadget-Pfad.
Der Vorgang erfolgt mittels. pOS_AddPart().

PATHGADTAG_CurrLock (const pOS_FileLock*) [IS.]
Definiert den Current-Dir im Gadget.
Default = Boot-Lock

PATHGADTAG_SetPath (const dosname_t*) [IS.]
Siehe STRGADTAG_String

PATHGADTAG_TargetFileGad (pOS_Gadget*) [IS.]
Der File-Anteil wird an diesen Gadget weitergereicht.

PATHGADTAG_AutoReScan (BOOL) [IS.] default=TRUE
Ein GCLMTHR_InputDone bzw. IDCMP_GadgetUp (RETURN, Drop)
löst re-scan aus.

SEE ALSO

gadget.class

1.9 gadget.library/txtboxgad.class()

NAME

txtboxgad.class -- Textanzeige-Gadget

Basis-class [buttongad.class]
DrawInfo [SCRNAM_GTxtBoxClass]

FUNCTION

Read-Only - Textanzeige

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

IOBJTAG_StaticString (const CHAR*) [IS.]
Darzustellender Text-String, der NICHT kopiert wird.
Der String MUSS über die Lebensdauer des Gadgets bestehen.

IOBJTAG_String (const CHAR*) [IS.]
Darzustellender Text-String, der kopiert wird.
Der String wird nur beim Setzen benötigt.

SEE ALSO
gadget.class

1.10 gadget.library/checkboxgad.class()

NAME
checkboxgad.class -- Auswahl-Gadget

Basis-class [gadget.class]
DrawInfo [SCRNAM_GCheckBoxClass]

FUNCTION
Ein/Aus - Wahlgadget.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

ICLTAG_Selected (BOOL) [ISG]
Aktueller Status.

IOBJTAG_StaticString (const CHAR*) [IS.]
Darzustellender Text-String, der NICHT kopiert wird.
Der String MUSS über die Lebensdauer des Gadgets bestehen.

IOBJTAG_String (const CHAR*) [IS.]
Darzustellender Text-String, der kopiert wird.
Der String wird nur beim Setzen benötigt.

SEE ALSO
gadget.class

1.11 gadget.library/mxgad.class()

NAME
mxgad.class -- Mutual-Exclude-Gadget

Basis-class [gadget.class]
DrawInfo [SCRNAM_GMXClass]

FUNCTION

Ein/Aus - Wahlgadget.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

MXGADTAG_StrLabels (const CHAR**) [I..]
String-Array zum Erzeugen der Menu-Punkte.
Das Array wird mit einem NULL-Pointer terminiert.
Es können maximal 8-Menüpunkte verwaltet werden.

MXGADTAG_Active (ULONG) [ISG]
Aktiver Punkt. (0,1,2,..)

SEE ALSO

gadget.class

1.12 gadget.library/bargad.class()

NAME

bargad.class -- Trennleiste

Basis-class [gadget.class]
DrawInfo [SCRNAM_GBarClass]

FUNCTION

Optische Trennleiste im Gadget-Layout.
Je nach Gruppenausrichtung wird die Leiste
horizontal bzw. vertikal gezeichnet.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

SEE ALSO

gadget.class

1.13 gadget.library/switchgad.class()

NAME

switchgad.class -- Gadget-Layers

Basis-class [gadget.class]
DrawInfo [SCRNAM_GSwitchClass]

FUNCTION

Mehrere Gadget-Seiten(Gruppen) liegen an der gleichen

Window-Position und können aktiviert werden.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

SWTGADTAG_AddGadget (pOS_Gadget*) [IS.]
Gadget-Seite(Gruppe) hinzufügen.
Das Switch-Gadget wird beim Dispose() alle Member-Gadget-Seiten ebenfalls löschen.

SWTGADTAG_RemGadget (pOS_Gadget*) [IS.]
Gadget-Seite entfernen.

SWTGADTAG_ActivateGadget (pOS_Gadget*) [IS.]
Legt fest, welche Gadget-Seite angezeigt und bearbeitet werden soll.

SEE ALSO

gadget.class

1.14 gadget.library/popupgad.class()

NAME

popupgad.class -- Entfalltbares Gadget

Basis-class [gadget.class]
DrawInfo [SCRNAM_GPopupClass]

FUNCTION

Es wird eine Gadgetseite im Layer angezeigt.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

POPUPGADTAG_PopGad (pOS_Gadget*) [IS.]
Gadget-Seite (Gruppe), die beim Aktivieren angezeigt werden soll.
Dieser Tag MUSS immer angegeben werden.

POPUPGADTAG_MinStdWidth (ULONG) [IS.] (default=2)
Minimale Breite in Standard-Width Einheiten.

POPUPGADTAG_MinStdHeight (ULONG) [IS.] (default=1)
Minimale Höhe in Standard-Height Einheiten.

POPUPGADTAG_MainGadWidth (BOOL) [IS.]
Die Popup-Breite wird vom Haupt-Gadget übernommen.

POPUPGADTAG_OkCode (BOOL) [IS.] Default=FALSE
IEGUCODE_Ok beendet den PopUp

POPUPGADTAG_CancelCode (BOOL) [IS.] Default=FALSE
IEGUCODE_Cancel beendet den PopUp

SEE ALSO
gadget.class

1.15 gadget.library/grpgad.class()

NAME

grpgad.class -- Layout-Gruppe

Basis-class [gadget.class]
DrawInfo [SCRNAM_GGroupClass]

FUNCTION

Logische Gadget-Gruppe für den Layouter.
Alle Member-Gadgets liegen immer innerhalb dieser Gruppe.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]
Referenzdaten zum Zeichnen.

GRPGADTAG_AddGadget (pOS_Gadget*) [IS.]
Weist der Gruppe ein neues Gadget bzw. Gadget-Gruppe als
Member zu.
Die Gadget-Gruppe löscht alle Member-Gadgets beim Dispose().

GRPGADTAG_RemGadget (pOS_Gadget*) [IS.]
Entfernt ein Member-Gadget aus der Gruppe.

GRPGADTAG_Horiz (BOOL) [IS.]
Die Gruppe erzeugt ein horizontales Layout.

GRPGADTAG_Vert (BOOL) [IS.] default
Die Gruppe erzeugt ein vertikales Layout.

GRPGADTAG_SpaceHoriz (ULONG) [IS.] default=2
Zwischenraum in horizontaler Richtung.

GRPGADTAG_SpaceVert (ULONG) [IS.] default=2
Zwischenraum in vertikaler Richtung.

GRPGADTAG_CenterHoriz (BOOL) [IS.] default=FALSE
Gadgets werden zentriert ausgerichtet.

GRPGADTAG_CenterVert (BOOL) [IS.] default=FALSE
Gadgets werden zentriert ausgerichtet.

GRPGADTAG_BorLeft (ULONG) [IS.]
GRPGADTAG_BorTop (ULONG) [IS.]
GRPGADTAG_BorRight (ULONG) [IS.]
GRPGADTAG_BorBottom (ULONG) [IS.]
Definiert einen Rahmen um die Gruppe, der beim Layout nicht

beschrieben wird.

GRPGADTAG_PrivateDisable (BOOL) [IS.]

Falls ein neues Gadget von dem grpgad.class - Gadget abgeleitet wird, kann das Disable-Handling unterdrückt werden.

GRPGADTAG_List (const pOS_List*) [..G]

Zeiger auf die interne Member-Liste. Die Liste darf NUR ausgelesen werden.

GRPGADTAG_LDMode (ULONG) [IS.] default=0

Layout-Mode:

GRPGADTAG_LDMode = 0 - langsam, jeder Member wird einzeln berechnet

GRPGADTAG_LDMode = 1 - der Gadget-Untergrund (Background) wird immer vollständig gelöscht.
=> Es wird keine aufwendige Region-Berechnung benötigt.

GRPGADTAG_LDMode = 2 - wie 1, jedoch wird der Gadget-Untergrund nicht verändert. Für Member-Gadgets, die die gesamte Fläche bedecken geeignet.

GRPGADTAG_FreeLayout (BOOL) [IS.]

Kein Layout berechnen.

Folgende Tags werden an alle Member-Gadgets weitergeleitet:

ICLTAG_Disabled

ICLTAG_Layer

ICLTAG_IgnoreIE

SEE ALSO

gadget.class

1.16 gadget.library/gfxgad.class()

NAME

gfxgad.class -- Grafik-Gadget

Basis-class [gadget.class]

DrawInfo [SCRNAM_GGfxClass]

FUNCTION

Der Gadget-Inhalt wird über den Picture-DataType nachgeladen.

TAGS

ICLTAG_DrawInfo (const pOS_DrawInfo*) [IS.]

Referenzdaten zum Zeichnen.

IOGFXTAG_FileName (const dosname_t*) [I..]

Dateipfad auf das anzuzeigende Grafikobjekt.

Der Pfad kann relativ zum CurrentDir der Applikation sein.

IOGFXTAG_EnvFile (const dosname_t*) [I..]
Relativer Dateinamen, der über ENV: bzw. ENVARC: gesucht wird.

IOGFXTAG_LowMem (BOOL) [I..]
Das Objekt wird angehalten, so wenig wie möglich
Speicherplatz zu belegen. (Sparmodus)
IOGFXTAG_LowMem = TRUE - belegt möglichst wenig Speicherplatz
Hieraus resultiert: Die Qualität und die Geschwindigkeit kann sinken.

IOGFXTAG_RefFrameNum (ULONG) [I..] Default=1
Legt das Referenz-Frame fest, das im Normalzustand gezeichnet wird.
(1,2,3..)

GFXGADTAG_IntuiTicksAnim (BOOL) [IS.]
Eine Animation wird bei IntuiTicks und selektiertem
Gadget angezeigt.

SEE ALSO
gadget.class