

in

COLLABORATORS

	<i>TITLE :</i> in		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 31, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	in	1
1.1	groupclass.guide	1
1.2	groupclass/--background--	1
1.3	groupclass/GRM_ADDMEMBER	2
1.4	groupclass/GRM_ADDSPACEMEMBER	3
1.5	groupclass/GRM_DIMENSIONS	4
1.6	groupclass/GRM_INSERTMEMBER	4
1.7	groupclass/GRM_REMMEMBER	5
1.8	groupclass/GRM_REPLACEMEMBER	5
1.9	groupclass/GRM_WHICHOBJECT	7
1.10	groupclass/OM_DISPOSE	7
1.11	groupclass/GROUP_BackFill	7
1.12	groupclass/GROUP_EqualHeight	8
1.13	groupclass/GROUP_EqualWidth	8
1.14	groupclass/GROUP_Inverted	9
1.15	groupclass/GROUP_Member	9
1.16	groupclass/GROUP_SpaceMember	10
1.17	groupclass/GROUP_Spacing	10
1.18	groupclass/GROUP_Style	10
1.19	groupclass/GROUP_[xxx]Offset	11
1.20	groupclass/GROUP_[xxx]Pen	11

Chapter 1

in

1.1 groupclass.guide

Search

TABLE OF CONTENTS

```
groupclass/--background--
groupclass/GRM_ADDMEMBER
groupclass/GRM_ADDSPACEMEMBER
groupclass/GRM_DIMENSIONS
groupclass/GRM_INSERTMEMBER
groupclass/GRM_REMEMBER
groupclass/GRM_REPLACEMEMBER
groupclass/GRM_WHICHOBJECT
groupclass/OM_DISPOSE
groupclass/GROUP_BackFill
groupclass/GROUP_EqualHeight
groupclass/GROUP_EqualWidth
groupclass/GROUP_Inverted
groupclass/GROUP_Member
groupclass/GROUP_SpaceMember
groupclass/GROUP_Spacing
groupclass/GROUP_Style
groupclass/GROUP_[xxx]Offset
groupclass/GROUP_[xxx]Pen
```

1.2 groupclass/--background--

NAME

Class: groupclass

Superclass: baseclass

Include File: <libraries/bgui.h>

FUNCTION

To layout it's members in a specific area. A groupclass object may have any number of members which are layouted according to a set of layout attributes.

The number of members a group can have is only limited by the amount of available memory. This will allow for simple and extremely complex layouts.

1.3 groupclass/GRM_ADDMEMBER

NAME

GRM_ADDMEMBER -- Add a object to a group.

SYNOPSIS

```
err = DoMethod( obj, GRM_ADDMEMBER, member, tag, data, ... );
```

```
ULONG    err;
Object    *member;
Tag       tag;
ULONG     data;
```

FUNCTION

This method can be used to add a member to an existing group. Layout specific attribute can be passed to suit your layout needs.

INPUTS

member - A pointer to the object which you want to add to the group.
tag,data - This is simply a tagitem array terminated by a TAG_DONE.

The following attributes are possible:

LGO_FixWidth -- ti_Data contains the width at which the object is fixed.

LGO_FixHeight -- ti_Data contains the height at which the object is fixed.

LGO_Weight -- Scaling objects is weight controlled. The default weight an object get's is 50. This attribute can be used to change that setting.

LGO_FixMinWidth -- The object is fixed at it's minimum width.

LGO_FixMinHeight -- The object is fixed at it's minimum height.

LGO_Align -- Normally only objects in a vertical group are aligned when they have a label on the left side of their hitbox. When this attribute is TRUE the object will get aligned wherever the label is placed.

LGO_NoAlign ** V38 ** -- Normally all objects in a vertical group with a label on the left of the hitbox are all aligned. This tag prevents this.

LGO_Relayout ** V41.10 ** -- Normally the group is layed-out again if adding the new member to the group succeeds. Setting this tag to FALSE prevents this so you may do that later calling GRM_RELAYOUT explicitly.

EXAMPLE

```
Object      *group, *cycle;
```

```
DoMethod( group, GRM_ADDMEMBER, cycle, LGO_FixMinHeight, TRUE,  
          LGO_Weight,      DEFAULT_WEIGHT,  
          TAG_END );
```

This adds a cycle object to a group fixing the height to it's minimum size and the weight to 50.

NOTES

Please note that the weight of an object affects to width of the object in a horizontal group and the height of the object in a vertical group.

Since V40 of the library it is allowed to use this method to change the group contents even while the window in which the group is located is open.

RESULT

err - TRUE uppon success, FALSE uppon failure.

If this method was successfull you do not need to dispose of the added member anymore. This will be disposed of as soon as the group to which the member was added is disposed of.

SEE ALSO

GRM_INSERTMEMBER, GRM_REMMEMBER, GRM_REPLACEMEMBER

1.4 groupclass/GRM_ADDSPACEMEMBER

NAME

GRM_ADDSPACEMEMBER -- Add a spacing member.

SYNOPSIS

```
err = DoMethod( obj, GRM_ADDSPACEMEMBER, weight );
```

```
ULONG    weight;
```

FUNCTION

This method can be used to add a special kind of object to the group. The member will be a weight controlled spacing in the group.

INPUTS

weight - The weight of the spacing object. Please refer to the GRM_ADDMEMBER section for more information about weights.

NOTE

This method may only be used when the group is not active (I.E. Displayed in the window).

RESULT

err - TRUE uppon success, FALSE uppon failure.

SEE ALSO

GRM_ADDMEMBER

1.5 groupclass/GRM_DIMENSIONS

NAME

GRM_DIMENSIONS -- Find out a group it's minimum size.

SEE ALSO

This method is sent to all members to inquire about their minimum sizes. Please refer to the "methods.doc" file for more information on this method.

1.6 groupclass/GRM_INSERTMEMBER

NAME

GRM_INSERTMEMBER -- Insert a member behind an existing member.

SYNOPSIS

```
err = DoMethod( obj, GRM_INSERTMEMBER, member, pred, tag, data, ... );
```

```
ULONG      err;
Object      *member;
Object      *pred;
Tag         tag;
ULONG      data;
```

FUNCTION

This method is basically the same as GRM_ADDMEMBER with the exception that you can determine the object after which the object is added.

INPUTS

member - A pointer to the object to insert.
pred - This must point to the object after which the new member is inserted. You may set this to NULL in which case the new member is inserted at the start of the list.
tag,data - Here you can pass a set of tagitems in which layout specific attributes can be defined. Please refer to the GRM_ADDMEMBER section for more information on the available attributes.

EXAMPLE

```
Object      *group, *cycle, *button;
```

```
DoMethod( group, GRM_INSERTMEMBER, cycle, button,
          LGO_FixMinHeight, TRUE,
          LGO_Weight, DEFAULT_WEIGHT,
          TAG_END );
```

This adds a cycle object to a group fixing the height to it's minimum size and the weight to 50. The cycle object is inserted after the button object.

NOTES

Please note that the weight of an object affects to width of the object in a horizontal group and the height of the object in a vertical group.

Since V40 of the library it is allowed to use this method to change the group contents even while the window in which the group is located is open.

RESULT

err - TRUE uppon success, FALSE uppon failure.

If this method was successfull you do not need to dispose of the added member anymore. This will be disposed of as soon as the group to which the member was added is disposed of.

SEE ALSO

GRM_ADDMEMBER, GRM_REMMEMBER

1.7 groupclass/GRM_REMMEMBER

NAME

GRM_REMMEMBER -- Remove an object from the group.

SYNOPSIS

```
DoMethod( obj, GRM_REMMEMBER, member );
```

```
Object      *member;
```

FUNCTION

To remove an object previously added at create time or with the GRM_ADDMEMBER or GRM_INSERTMEMBER methods.

INPUTS

member - A pointer to the object which is to be removed from the group.

RESULT

No return code specified.

NOTE

Since V40 of the library it is allowed to use this method to change the group contents even while the window in which the group is located is open.

After the object has been removed you are responsible for disposing of the removed object.

SEE ALSO

GRM_ADDMEMBER, GRM_INSERTMEMBER, intuition.library/DisposeObject()

1.8 groupclass/GRM_REPLACEMEMBER

NAME

GRM_REPLACEMEMBER -- ReTplace an object with another.

SYNOPSIS

```
rep = DoMethod( obj, GRM_REPLACEMEMBER, rem, add, tag, data, ... )
```

```
Object    *rep;  
Object    *rem;  
Object    *add;  
Tag       tag;  
ULONG     data;
```

FUNCTION

This method can be used to replace a member of a group with another member. It is basically a GRM_REMMEMBER and a GRM_INSERTMEMBER roled into one.

INPUTS

rem - This should point to the object which is to be replaced.
add - This should point to the object which is to replace the object described in "rem".
tag,data - This is an array of attributes which control the way the new object is layouted. The attributes allowed here are exactly the same as the attributes you can use with the GRM_ADDMEMBER method.

EXAMPLE

```
Object    *group, *cycle, *button;  
  
DoMethod( group, GRM_REPLACEMEMBER, cycle, button,  
          LGO_FixMinHeight, TRUE,  
          LGO_Weight,      DEFAULT_WEIGHT,  
          TAG_END );
```

This replaces the cycle object with the button object in the group fixing the height to it's minimum size and setting the weight to 50.

NOTES

Please note that the weight of an object affects to width of the object in a horizontal group and the height of the object in a vertical group.

Since v40 of the library it is allowed to use this method to change the group contents even while the window in which the group is located is open.

RESULT

rep - A pointer to the replaced object uppon success, and NULL uppon failure.

If this method was successful you do not need to dispose of the member which replaces the other member anymore. This will be disposed of as soon as the group to which the member was added is disposed of.

You are however responsible to dispose of the object which you have replaced.

SEE ALSO
GRM_ADDMEMBER

1.9 groupclass/GRM_WHICHOBJECT

NAME
GRM_WHICHOBJECT -- Find the object at specific coordinates.

SYNOPSIS
obj = DoMethod(obj, GRM_WHICHOBJECT, xy);

Object *obj;
ULONG xy;

FUNCTION
This method should be used to find out which object is located under the given X/Y coordinates.

INPUTS
xy - A 32bit unsigned integer which holds the horizontal coordinate in the upper 16 bits and the vertical coordinate in the lower 16 bits. The coordinates should be relative to the upper-left corner of the window.

NOTE
Please note that this method will only work properly when the window in which the group is located is open.

This method is primarily used for the tool tips but may also be useful for the application programmer.

RESULT
Returns the object under the coordinates or NULL if there is no object under the given coordinates.

1.10 groupclass/OM_DISPOSE

NAME
OM_DISPOSE -- Dispose of the group.

CHANGE
Before passing to the superclass the group will automatically dispose of all it's members.

1.11 groupclass/GROUP_BackFill

NAME
GROUP_BackFill -- (ULONG)

FUNCTION

To provide a backfill possibility the same as the frameclass supplies. The data passed with this tag should be the same as defined and documented for the FRM_BackFill attribute of the frameclass documentation. Please note that this tag is overided when the group has a frame attached to it.

DEFAULT

0.

APPLICABILITY

(I).

SEE ALSO

frameclass.doc/FRM_BackFill, GROUP_BackPen, GROUP_BackDriPen

1.12 groupclass/GROUP_EqualHeight

NAME

GROUP_EqualHeight -- (BOOL)

FUNCTION

To force all members of the group to get the same minimum height. This makes it easy to create a row of objects which all are equally high.

DEFAULT

FALSE.

APPLICABILITY

(I).

SEE ALSO

GROUP_EqualWidth

1.13 groupclass/GROUP_EqualWidth

NAME

GROUP_EqualWidth -- (BOOL)

FUNCTION

To force all members of the group to get the same minimum width. This makes it easy to create a row of objects which all are equally wide.

DEFAULT

FALSE.

APPLICABILITY

(I).

SEE ALSO

GROUP_EqualHeight

1.14 groupclass/GROUP_Inverted

```

NAME
GROUP_Inverted -- ( BOOL )

FUNCTION
To force the members added at create time to be AddHead()'ed to the
member list instead of AddTail()'ed. This is necessary for the
assembly macros of the BGUI package.

DEFAULT
FALSE.

APPLICABILITY
(I).

```

1.15 groupclass/GROUP_Member

```

NAME
GROUP_Member -- ( Object * )

FUNCTION
To add a member to a group at initialization time. This tag is read a
little different than the other tags. Following the object a set of
layout attributes can be passed.

EXAMPLE
UBYTE  *labels[] = { "LAB1", "LAB2", NULL };
Object  *group, *cycle;

/* With macros */
group = HGroupObject,
    StartMember,
        cycle = Cycle( "Cycle", labels, 0, 0 ),
        FixMinHeight, Weight( DEFAULT_WEIGHT ),
    EndMember,
EndObject;

/* Without macros */
group = BGUI_NewObject( BGUI_GROUP_GADGET,
    GROUP_Member,
        cycle = BGUI_NewObject( BGUI_CYCLE_GADGET,
            LAB_Label, "Cycle",
            CYC_Labels, labels,
            CYC_Active, 0,
            GA_ID, 0,
            TAG_END ), LGO_FixMinHeight, TRUE,
            LGO_Weight, DEFAULT_WEIGHT,
            TAG_END, 0,
        TAG_END );

```

As you can see the layout attributes follow the object pointer in the `ti_Data` field directly. Therefore it is also important that the layout attributes are terminated with a `TAG_END, 0` rather than a single

TAG_END. The macros will automatically do this for you.

If by any chance the object you add is NULL. The creation of the group will fail. All objects that were successfully added to the group are disposed of in this case.

DEFAULT
NULL.

APPLICABILITY
(I).

SEE ALSO
Methods/GRM_ADDMEMBER

1.16 groupclass/GROUP_SpaceMember

NAME
GROUP_SpaceObject -- (ULONG)

FUNCTION
To add a weight controlled spacing member at create time. The data of this tag should contain the weight of the spacing.

DEFAULT
50.

APPLICABILITY
(I).

SEE ALSO
GRM_ADDSPACEMEMBER

1.17 groupclass/GROUP_Spacing

NAME
GROUP_Spacing -- (ULONG)

FUNCTION
Set the number of pixels the group uses as a space between the group members.

DEFAULT
0.

APPLICABILITY
(I).

1.18 groupclass/GROUP_Style

NAME
GROUP_Style -- (ULONG)

FUNCTION
Set the style of the group to create. The following styles are possible:

GRSTYLE_HORIZONTAL -- A horizontal group is created.

GRSTYLE_VERTICAL -- A vertical group is created.

DEFAULT
GRSTYLE_HORIZONTAL.

APPLICABILITY
(I).

1.19 groupclass/GROUP_[xxx]Offset

NAME
GROUP_HorizOffset, GROUP_VertOffset, GROUP_LeftOffset,
GROUP_RightOffset, GROUP_TopOffset, GROUP_BottomOffset -- (ULONG)

FUNCTION
Set the offset in pixels from the group borders at which the layouting is started. Please note that frames are not taken into consideration.

DEFAULTS
0.

APPLICABILITY
(I).

1.20 groupclass/GROUP_[xxx]Pen

NAME
GROUP_BackPen, GROUP_BackDriPen -- (ULONG) ** V40 **

FUNCTION
To determine the pen or DrawInfo pen which is used to back fill the group. The data passed here should be the pen number of the color you want to be used for the GROUP_BackPen attribute.

The GROUP_BackDriPen attribute expects the DrawInfo index number to be used to backfill the group.

Specifying ~0 (-1) for the pens will deactivate the backfill color and the class will use the BACKGROUNDPEN for the backfill.

DEFAULTS
~0.

APPLICABILITY
(I) .

SEE ALSO
GROUP_BackFill