

**in**

COLLABORATORS

	TITLE :  in		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		July 31, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>in</b>	<b>1</b>
1.1	externalclass.guide . . . . .	1
1.2	externalclass/--background-- . . . . .	1
1.3	externalclass/EXT_Class . . . . .	1
1.4	externalclass/EXT_ClassID . . . . .	2
1.5	externalclass/EXT_MinHeight . . . . .	2
1.6	externalclass/EXT_MinWidth . . . . .	3
1.7	externalclass/EXT_NoRebuild . . . . .	3
1.8	externalclass/EXT_Object . . . . .	4
1.9	externalclass/EXT_TrackAttr . . . . .	4

# Chapter 1

## in

### 1.1 externalclass.guide

Search  
TABLE OF CONTENTS

externalclass/--background--  
externalclass/EXT\_Class  
externalclass/EXT\_ClassID  
externalclass/EXT\_MinHeight  
externalclass/EXT\_MinWidth  
externalclass/EXT\_NoRebuild  
externalclass/EXT\_Object  
externalclass/EXT\_TrackAttr

### 1.2 externalclass/--background--

NAME  
Class: externalclass  
Superclass: baseclass  
Include File: <libraries/bgui.h>

FUNCTION  
To provide an interface class which can be used to include third-party gadget classes like the colorwheel.gadget in a BGUI user interface. Notification is currently only possible via the WM\_ADDUPDATE method. See the windowclass documentation for more information on this subject.

### 1.3 externalclass/EXT\_Class

NAME  
EXT\_Class -- ( Class \* )

FUNCTION  
Set the class from which this class needs to create an object.

---

This attribute expects a pointer to an already initialized class.

#### NOTE

Please note that `_you_` are responsible for opening and closing the class yourself.

#### DEFAULT

NULL.

#### APPLICABILITY

(I).

#### SEE ALSO

EXT\_ClassID

## 1.4 externalclass/EXT\_ClassID

#### NAME

EXT\_ClassID -- ( STRPTR )

#### FUNCTION

Set the class from which this class needs to create an object. This attribute expects a pointer to the name of the public class like for instance "colorwheel.gadget".

#### NOTE

Please note that `_you_` are responsible for opening and closing the class library yourself.

#### DEFAULT

NULL.

#### APPLICABILITY

(I).

#### SEE ALSO

EXT\_Class

## 1.5 externalclass/EXT\_MinHeight

#### NAME

EXT\_MinHeight -- ( ULONG )

#### FUNCTION

As external classes normally do not understand the layout engine methods used by BGUI it has to be helped a little. With this attributes you set the minimum height of the external object. It is very important to set reasonable values here because no checks are made.

#### DEFAULT

0 (stupid size).

---

APPLICABILITY  
(I).

SEE ALSO  
EXT\_MinWidth

## 1.6 externalclass/EXT\_MinWidth

NAME  
EXT\_MinWidth -- ( ULONG )

FUNCTION  
As external classes normally do not understand the layout engine methods used by BGUI it has to be helped a little. With this attributes you set the minimum width of the external object. It is very important to set reasonable values here because no checks are made.

DEFAULT  
0 (stupid size).

APPLICABILITY  
(I).

SEE ALSO  
EXT\_MinHeight

## 1.7 externalclass/EXT\_NoRebuild

NAME  
EXT\_NoRebuild -- ( BOOL )

FUNCTION  
To tell the external class that the external object does not have to be rebuilt after a re-size. Most classes are smart enough to handle a resize of the object themselves but there are classes like the colorwheel.gadget that requires a rebuild upon a size change.

When this attribute is set to TRUE the class will not rebuild the external object and you do not need to use the EXT\_TrackAttr attribute to handle the object settings.

DEFAULT  
FALSE.

APPLICABILITY  
(I).

SEE ALSO  
EXT\_TrackAttr, EXT\_Object

---

## 1.8 externalclass/EXT\_Object

NAME

EXT\_Object -- ( Object \* )

FUNCTION

Get a pointer to the "real" external object. Please note that this pointer changes at every size change unless the EXT\_NoRebuild attribute is FALSE.

APPLICABILITY

(G).

SEE ALSO

EXT\_NoRebuild

## 1.9 externalclass/EXT\_TrackAttr

NAME

EXT\_TrackAttr -- ( Tag )

FUNCTION

To tell which attributes from the external object need to be tracked. Because some external classes, like the colorwheel, cannot change size once created it is necessary that the externalclass recreates the object at each size change. As this usually means that the current external object settings are lost you can tell which attributes need to be tracked.

The tracked attributes are obtained by sending the external object a OM\_GET method for each of the attributes. This means that the only attributes that can be tracked are the ones that are gettable on the external object. There is no limit as to the number of attributes that are trackable.

Please note however that internally the tracked attributes are stored in a 32bit integer which means that tracking attributes like the colorwheel.gadget it's WHEEL\_RGB will not work.

You can also pass attributes that are meant for the external object at initialization time. These attributes are remembered by this class and reused at each recreation of the external object.

EXAMPLE

```
Object      *wheel;
struct Screen *screen;

/*
** Create a "colorwheel.gadget" external object.
**/
wheel = ExternalObject,
    EXT_MinWidth,      30,
    EXT_MinHeight,     30,
    EXT_ClassID,       "colorwheel.gadget",
```

```
WHEEL_Saturation,    0,  
WHEEL_Screen,        screen,  
EXT_TrackAttr,        WHEEL_Saturation,  
EXT_TrackAttr,        WHEEL_Hue,  
EndObject;
```

All tags defined above are saved (including the tags passed automatically by the ExternalObject macro). Now a separate copy of the attributes to track is created.

Once the object needs to be recreated the first thing what is done is getting the tracked attribute values from the old object.

Now the old object is disposed of and a new one is created with exactly the same attributes that were passed at initialization time.

Once this is accomplished the tracked attributes are set to the new object.

Please note that tracking attributes is only necessary with classes that require a rebuild of the object when it is resized.

APPLICABILITY  
(I).

BUGS  
The EXT\_xxx attributes from the initialization tags are not filtered out of the saved tag list.

Only attributes which fit in a 32bit integer can be tracked.

SEE ALSO  
EXT\_NoRebuild?@endnode