

# MiscDistributor

**Inherits From:** MiscTee : Object  
**Declared In:** <misckit/MiscDistributor.h>

## Class Description

A MiscDistributor is a subclass of MiscTee which allows forwarded messages to cross the boundaries of .nib files. This is done by creating named MiscTee instances in the global program space which are thought of as "distribution points". By connecting to a distribution point, the MiscDistributor is able to pass messages to all other MiscDistributors connected to that distribution point.

A choice is given to only send messages to or only receive messages from a given distribution point. Since more than one point may be connected to, a very complex interconnection network may be created. Messages coming from the distribution point to the MiscDistributor are "in" messages, messages sent to the distribution point are "out" messages. The constants MiscIn, MiscOut, and MiscInout reflect these possibilities.

Here is an example of how a MiscDistributor might be used to synchronize two sliders which are in completely different .nib files:

paste.eps ↵

In order to create the .nib files above, you would drag a slider onto a panel and drag a MiscDistributor into the objects area of the .nib. Next, drag the two connections between the Slider and the MiscDistributor, setting the actions as shown. Finally, in the MiscDistributor's inspector, add a connection and give it some name. Make sure to use the same name in both .nib files so that they can find each other. Also set the direction of the

connection to be both ways.

The easiest way to use the MiscDistributor is to take it off the accompanying InterfaceBuilder palette. In the inspector, add connections as needed, specifying the connection name and the direction messages should travel between the MiscDistributor and the named distribution point.

It is also possible to have non-.nib objects instantiate and connect to MiscDistributors. In this case, the MiscDistributor must be manually hooked up to the desired distribution points via the **±connectToDistributorNamed:direction:** method, specifying the name of the distribution point and the connection direction.

## Instance Variables

id **connections;**

connections

A list of connections (names and directions) set up in InterfaceBuilder which will be made upon unarchiving the object. A private class is used to store the information.

## Method Types

Connecting MiscDistributors

+ findDistributorNamed:  
- connectToDistributorNamed:direction:

## Class Methods

**findDistributorNamed:**

+ **findDistributorNamed:**(const char \*)*aString*

Finds the MiscTee instance which corresponds to the distribution point named *aString* and returns it, creating it if it doesn't yet exist.

**See also:**    $\pm$  `connectToDistributorNamed:direction:`

## Instance Methods

**`connectToDistributorNamed:direction:`**

- **`connectToDistributorNamed:`**(const char \*)*aString*  
    **`direction:`**(MiscDCDirection)*direction*

Connects the receiver to a distribution point named *aString*. The *direction* argument should be one of MiscIn, MiscOut, or MiscInout. Returns **self**.

**See also:**   **`+`** `findDistributorNamed:`