

Making your class a delegate; ↵ Making your class a delegate

- 1 **Connect your instance to an object that has delegates.**
- 2 **Select the delegate outlet in the Connections inspector.**
- 3 **Click Connect.**
- 4 **Implement the delegate methods.**

Several OpenStep classes allow you to register an object as their delegate. As certain events occur, the objects send messages to their delegates, giving them the opportunity to participate in processing. In Interface Builder, you can easily designate your class's instance as a delegate.

[_MakingClassDelegate.eps ↵](#)

Next, implement the delegate methods you want your class to respond to. In this example, the object acting as delegate archives itself before the application terminates.

```
- (void)applicationWillTerminate:(NSNotification *)note
{
    [NSArchiver archiveRootObject:self toFile:@"TravelData"];
}
```

Tip: You can programmatically set an object's delegate by sending it the **setDelegate:** method.

Messages to delegates sometimes notify them of impending or just-transpired events, and sometimes request them to complete some work. Major classes with delegate methods are `NSApplication`, `NSWindow`, `NSText`, and `NSBrowser`. See the *Application Kit Reference* for details on delegate methods.