



# MDI Modes – Make Mine a Single, With a Twist

**P**rior to BBj® 9.0, when a BBj program used an MDI (Multiple Document Interface) window, all programs that were SCALed by that program (the MDI master process) appeared in the same MDI window and all the SCALed programs (the MDI child processes) terminated when the master process terminated. BBj 9.0 introduced new API methods and a new command line parameter that allows the program to control whether the child processes will display within the MDI window and whether the child processes will be terminated when the master process is terminated.

Now, the developer controls whether a BBj program that uses an MDI window displays all programs SCALed by that program in the same MDI window and terminates all the SCALed programs when the master process finishes. This article introduces this feature with how-to tips on making an MDI application into an SDI (Single Document Interface) anytime it is needed.



*By David Wallwork  
Senior Software  
Architect*

## Default MDI Mode Methods

The new BBj 9.0 method `BBjMDI::setDefaultMDIMode()`, allows the program to control the behavior of all MDI child processes that are started after calling `setDefaultMDIMode()`.

This method accepts three valid modes.

1. MDI – the traditional mode in which all MDI children appear within the master MDI window and are terminated when the MDI master is terminated.
2. SDI – displays each MDI child using a `BBjWindow` that is free to be outside the master MDI window but the children will be terminated when the MDI master is terminated.
3. DETACHED – causes the programs that are SCALed by the MDI master to behave as though they had been launched as separate programs from the command line. They do not appear within the MDI master window, they will not be terminated when the MDI master is terminated, and they will not share the `GroupNameSpace` of the MDI master; they are completely detached from the MDI master.

To call this new method, enter code such as:

```
mdi!=BBjAPI().getMDI()
mdi!.setDefaultMDIMode("SDI")
```

The corresponding method `BBjMDI.getDefaultMDIMode()` returns the current default MDI mode.

## The MDI Command Line Parameter

When starting a new program from the command line, enter the new `-MDI` command line parameter to set the `defaultMDIMode` of the program. To start a program in “SDI” mode, use a command like the following:

```
BBJ -MDISDI program.src
```

## Child Frame Mode Methods

To start a new `BBjSession` using `BBjCommandLineObject`, set the MDI mode for the new `BBjSession` by calling `setChildFrameMode()` with one of the string values MDI, SDI, or DETACHED. This setting will affect all `BBjSessions` started with `BBjCommandLineObject`; the value can be retrieved by calling `getChildFrameMode()` on the `BBjCommandLineObject`.

## Summary

BBj 9.0 provides methods on the `BBjAPI` object for setting the default MDI mode that will be used when creating new child MDI processes. Similarly, the command line setting and the methods of the `BBjCommandLineObject` provide control of the mode of a single BBj process. Combining these various methods, the user has complete control of the MDI behavior of all `BBjSessions`.

Today, you can truly develop your application, any way you want! ■