

Navigating using the stack; ↩ Navigating using the stack

- 1 Click the inspector button on the Launch panel to bring up the Task Inspector.
- 2 Choose Stack from the pop-up list.
- 3 Click a stack frame to have the debugger jump to that stack frame.

Each time your program invokes a C function, a C++ member function, or an Objective-C method, the information about where in the program the call was made is saved in a block of data called a stack frame. The frame also contains the arguments of the call and the local variables of the function that was called.

The Task Inspector displays the stack on the right side of the window. Each row in the stack display represents one stack frame. The current stack frame is numbered 0, the frame that called it is 1, and so on.

_NavigatingUsingStack.eps ↩

At any given time, one of the stack frames is selected by **gdb**; many **gdb** commands refer implicitly to this selected frame. In particular, whenever you ask **gdb** for the value of a variable in the program, the value is found in the selected frame. You can select any frame by clicking it. You can then examine the values of variables pertaining to that stack frame. As you navigate to a different stack frame, the Project Builder main window shows you the currently executing line of code at that frame.

Tip: You can return from the current level by Shift-clicking the program counter.