

View & Window Controllers (Mac)

Lecture 11

These slides are licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Window Controllers

- Technically a View Controller
- Typically for document oriented applications.
- Many try to *not* use them to be “more” portable to iOS.

Window Controllers

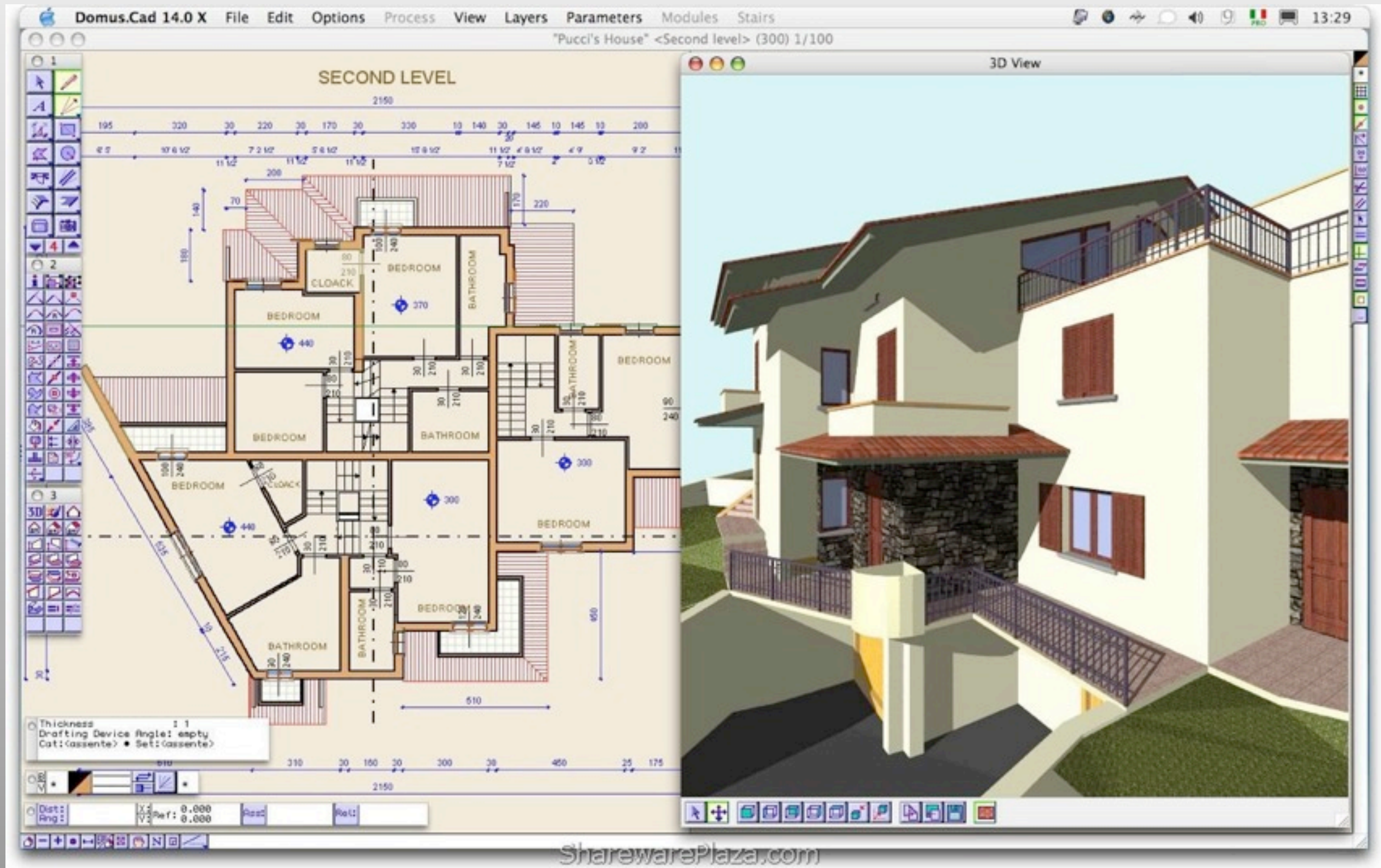
- Loading and displaying the window
- Closing the window when appropriate
- Customizing the window's title
- Storing the window's frame (size and location) in the defaults database
- Cascading the window in relation to other document windows of the application

Window Controllers

“If your document requires or allows multiple windows for a single document, that is a good reason to subclass `NSWindowController`.”

- Apple

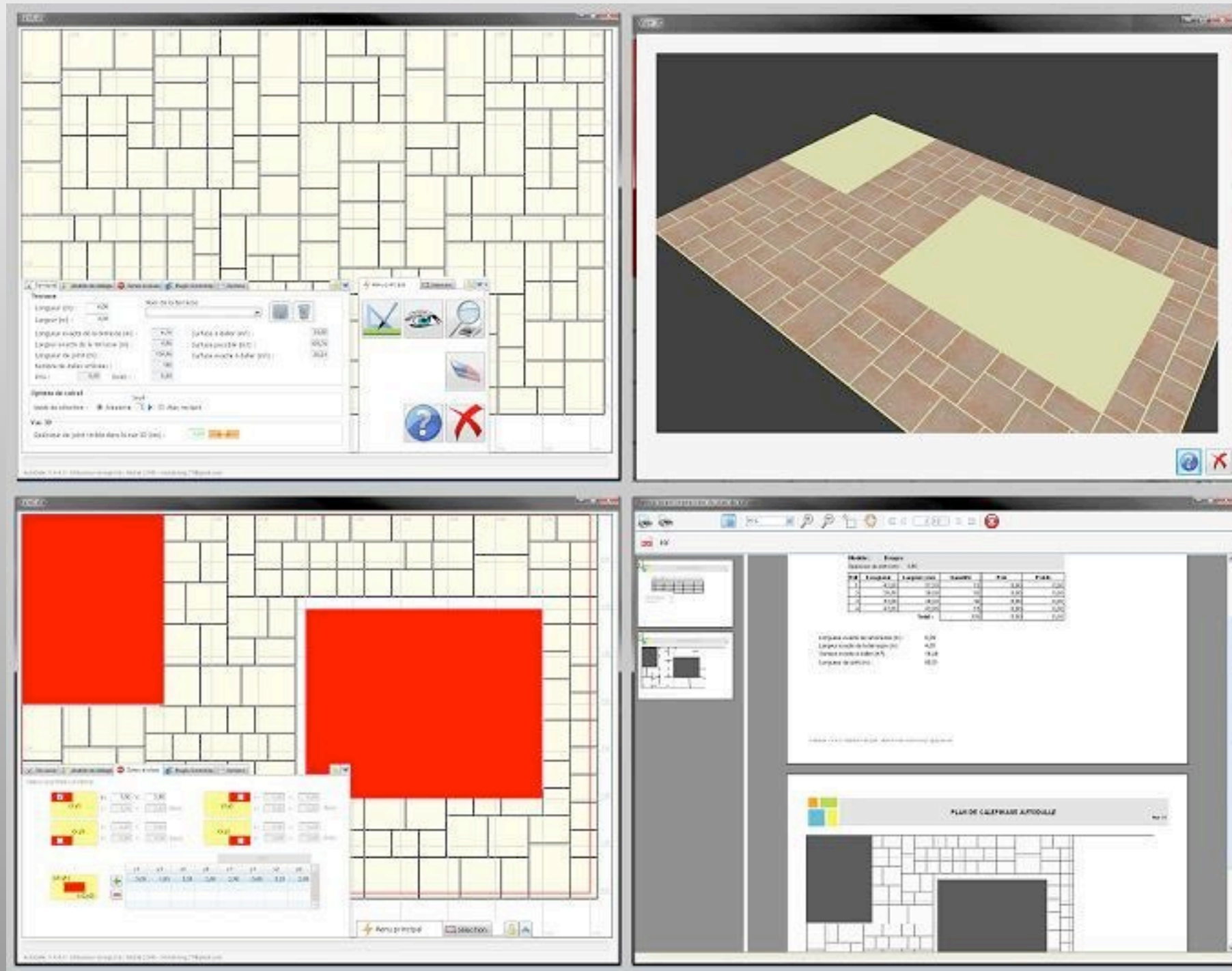
Window Controllers



These slides are licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Window Controllers



These slides are licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Window Controllers

Typically Overridden

`windowWillLoad`

Override to perform tasks before the window nib file is loaded.

`windowDidLoad`

Override to perform tasks after the window nib file is loaded.

`windowTitleForDocumentDisplayName:`

Override to customize the window title.

These slides are licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

NSWindowController

- Expects `initWithWindowNib:...` method to be called (generic NSWC behavior).
- Subclasses are usually designed to load and control interfaces within a different nib file.
- Error-prone for instances of these subclasses to tell exactly which Nib file to load.

NSWindowController

- The solution is to override the `init` method, and call `[super initWithWindowNibName:]` with the correct Nib name.

NSWindowController Nib File

File's owner

- When a window controller loads a nib file, it sets itself as the owner of that nib file. To enable connections from the window controller to other objects in the nib file, you need to set the file's owner of the nib file that will be managed by a custom window controller to the class of that custom window controller.

• **window outlet**

- Window controllers keep track of their window using their **window** outlet. The **window** outlet of your window controller (set as the file's owner in your nib file) should be connected to the window your window controller is responsible for.

• **delegate outlet (optional)**

- While not required, it's often convenient to set up your window controller as the delegate of the window it manages. In your nib file, connect the **delegate** outlet of the window your window controller is managing to the object that represents your window controller—**specifically, the file's owner object.**