

# Making A Slick Max Standalone

Dan Nigrin  
Defective Records Software

Expo '74  
New York  
October 15, 2011

# About Me

- Late 1980's: club DJ in Baltimore, MD - industrial / alternative / techno
- Early 1990's: “Glitch” – releases on multiple labels: R&S, Fax, Industrial Strength, Radikal, Antler-Subway... Collaborations with Richie Hawtin, Lenny Dee, and others.  
<http://discogs.com/artist/Glitch>
- Mid 1990's: Created our own label, Defective Records
- Moved to Boston

Firefox ▾

Defective Records +

http://defectiverecords.com/ Google

**defective**

R E C O R D S

[home](#)

[label](#)

[history](#)

[discography](#)

[forum](#) NEW

[ordering info](#)

[contact info](#)

[submit a demo](#)

[links](#)

**Fungus of Terror**  
New green vinyl LP!  
Featuring remixes by  
[Tipsy](#), [Solar X](#) &  
[Snakeoiler](#)

Watch classic videos from  
Defective's own  
**GLITCH**  
*"Closed Mind", "Synaptic Breakdown" & "Surge"*  
(1993, 1993 & 1992)

Get our releases digitally!

**Cambodian Psych-Out**  
14 track Cambodian rock compilation

<http://defectiverecords.com>

# About Me

- Defective Records now home to my software creations, almost all of which are Max creations
  - **VSTi Host** - lightweight VST instrument (and effect) host
  - **MC-202 Hack** - program the Roland MC-202's sequencer
  - **Major Malfunction** - Pluggo-based “glitch” plugin
  - **Klee Step Sequencer**
  - **M-185 Step Sequencer**
  - **MC-4 Hack** – program the Roland MC-4's sequencer

## defective records software

### M-185

software step sequencer



**NEW** Roland System 100m step sequencer -  
now in software!



a collaboration with



### MC-202 Hack



**MC-202 Hack v2.0**  
a better way to program  
your Roland MC-202



### Klee

software step sequencer



the most interesting step sequencer around -  
now in software! **v2.1**



a collaboration with  
[electro-music.com](http://electro-music.com)



**pc1600 user**  
the unofficial Peavey PC1600™  
& PC1600X™ pages

### VSTI HOST

Use your VST instruments  
& effects without a bulky host  
environment

### PARTY FAVOR



A software utility for the [WTPA](#)  
8-bit bendable sampler



### OMS Convert

Convert your OMS patch name  
files to Cubase 5 & SX formats

### major malfunction



mangle, glitch and FSU plugin for  
[Ableton Live](#)  
version 2.0 with Multi Mode!



# M-185

software  
step  
sequencer

## who's using it:

(Daniel Miller, Mute Records founder)



## cEvin Key

(Skinny Puppy, Cyberaktif, Tear Garden, Doubting Thomas, etc...)



# Klee

software  
step  
sequencer

major  
malfunction  
.mangle, glitch &  
fsu in ableton live.  
.version 2.0.

## who's using it:

Vernon Reid  
(Living Colour)



BT



Joey Beltram



Chris Carter  
(Chris & Cosey,  
Throbbing Gristle)



Daniel Miller  
(Mute  
Records)



Richard Devine



Taylor  
Deupree  
(12k.com)



Mark Bell  
(LFO)



Kid Beyond



Blank & Jones



Mijk van Dijk  
(Microglobe)



Chris Fortier

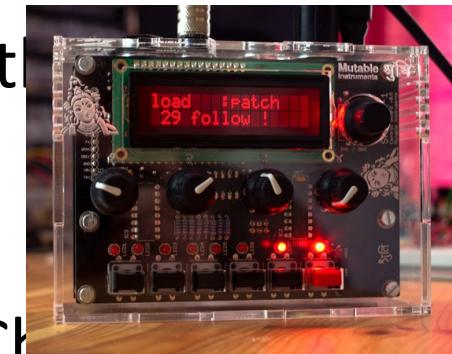


Pascal F.E.O.S.



# About Me

- Contributor to Jack OS X (Mac port of J Connection Kit, <http://jackosx.com> )
- DIY synth builder (x0xb0x/x0xi0, Shruti, etc...)
- Day job: Pediatric Endocrinologist & Clinical Information Officer, Children's Hospital



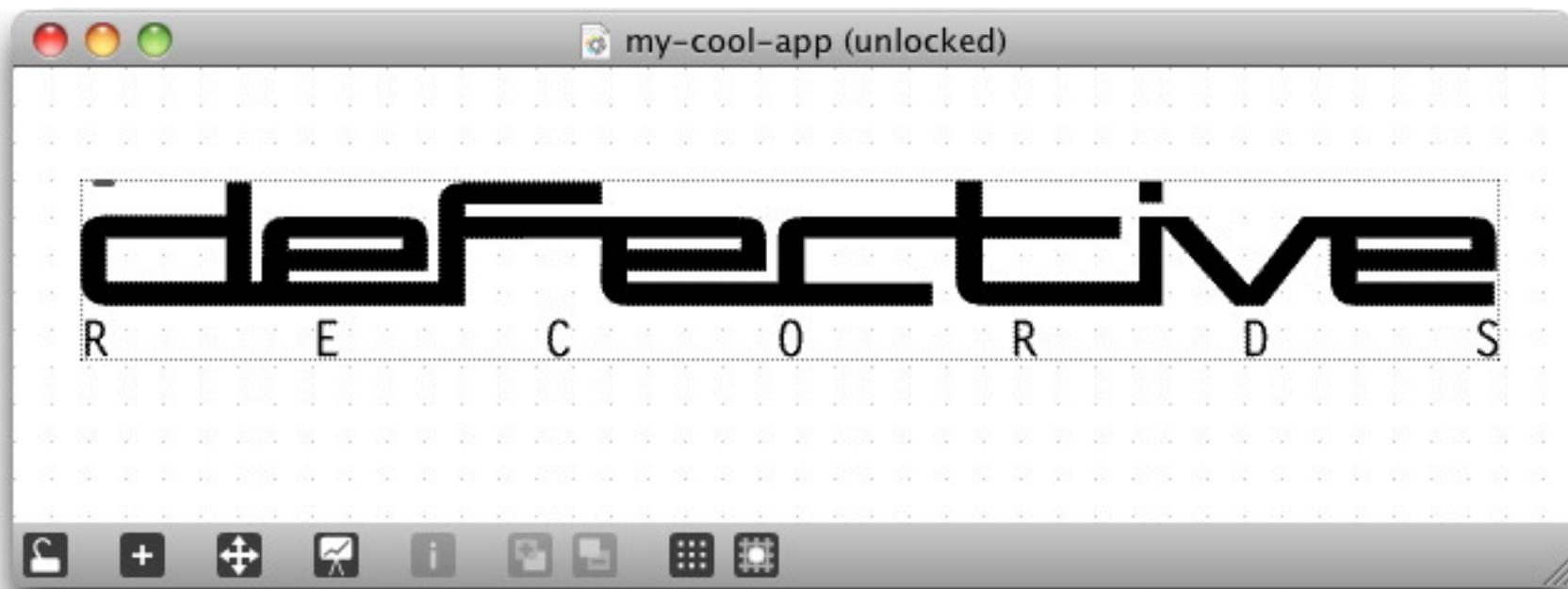
Presentation available at:

<http://defectiverecords.com/expo74/standalones.pdf>

# What I Will Cover

- Work our way through slickify-ing an app – from very basic to slightly less basic
- NOT going to talk about how to make fancy graphics – if I knew how to make them, I would! ☺
- Will try to cover both Mac and Windows

# The original patcher



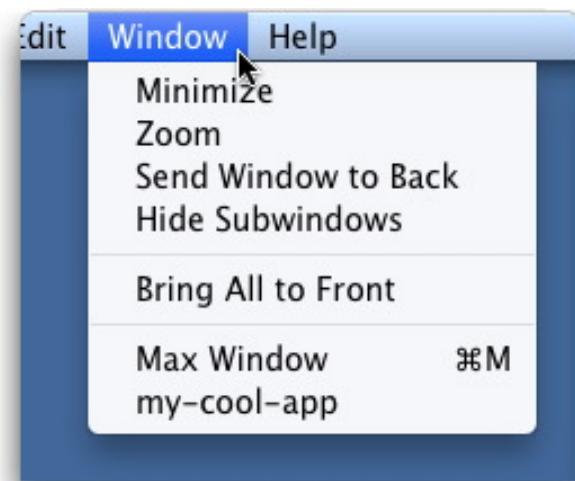
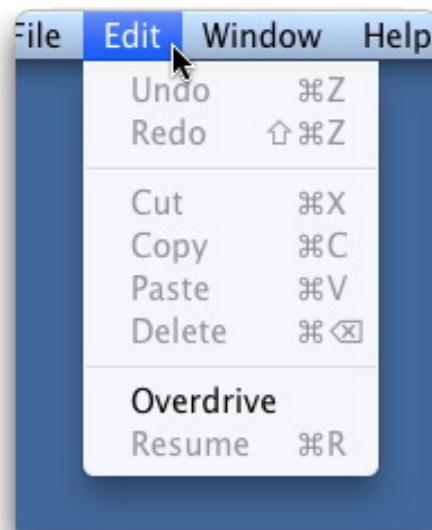
# The resulting app



5

my-cool-app

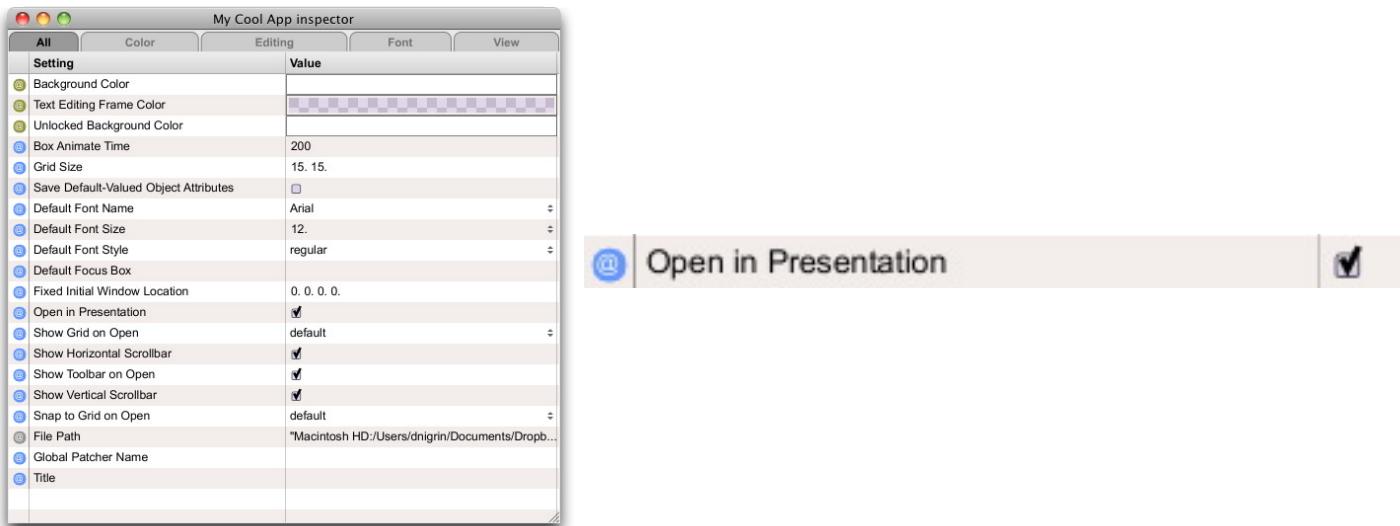
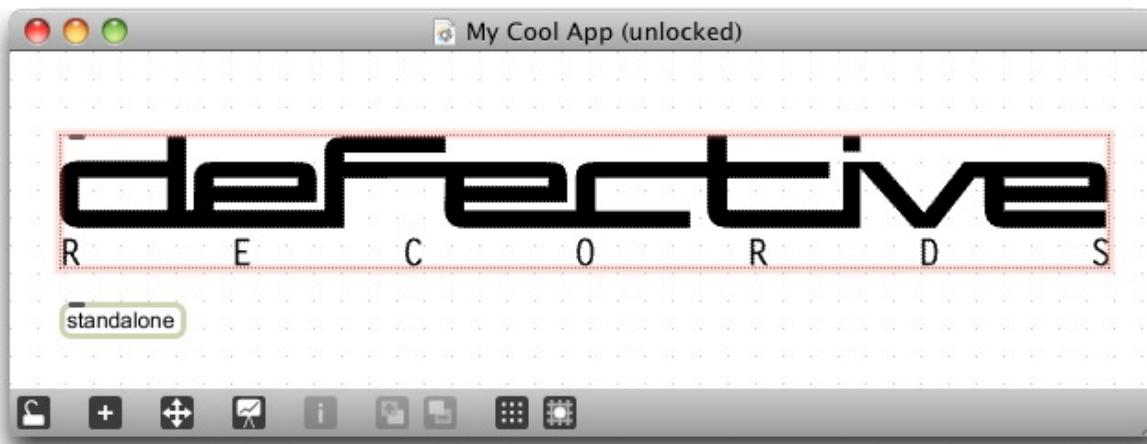
# The resulting app's menus



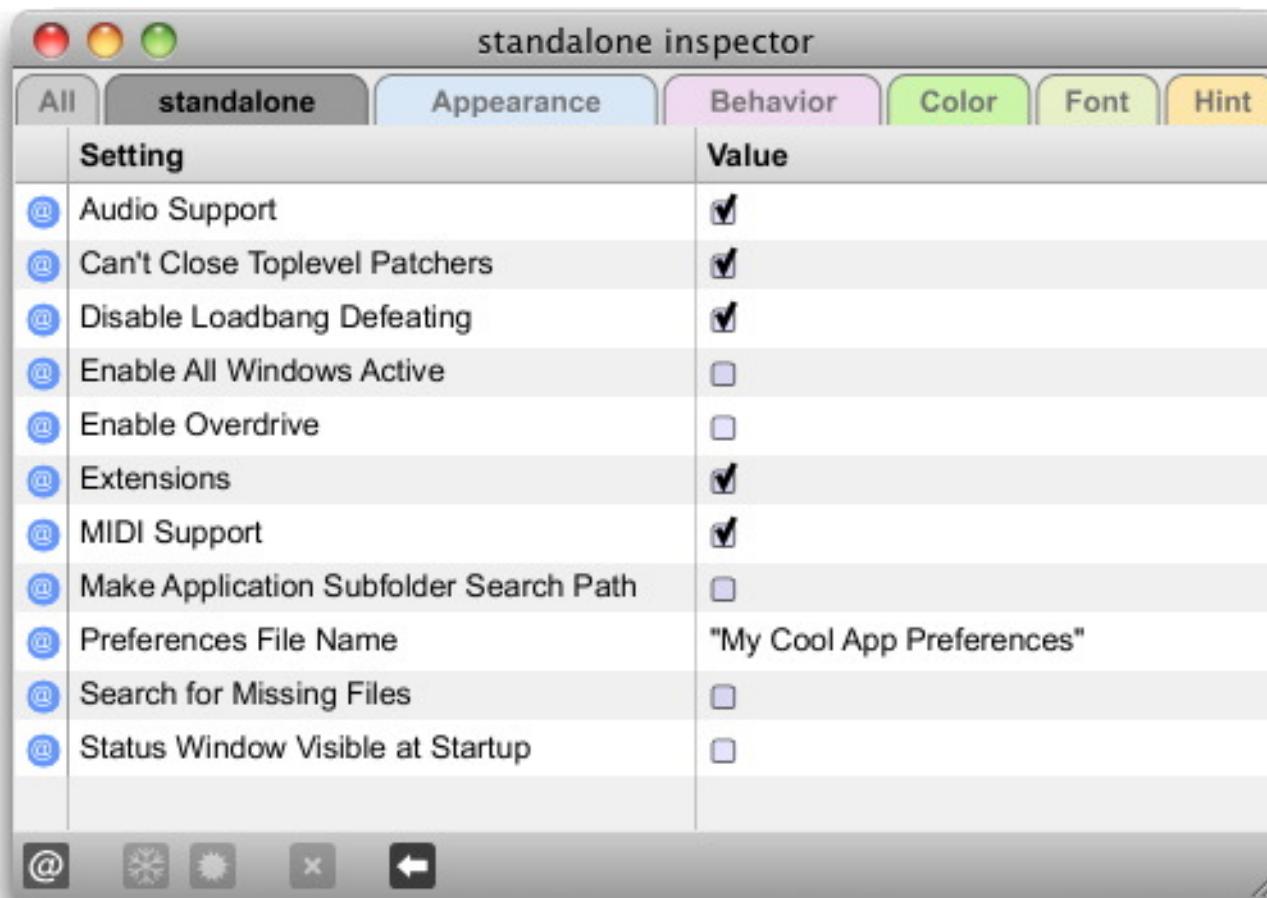
# Fix the app name



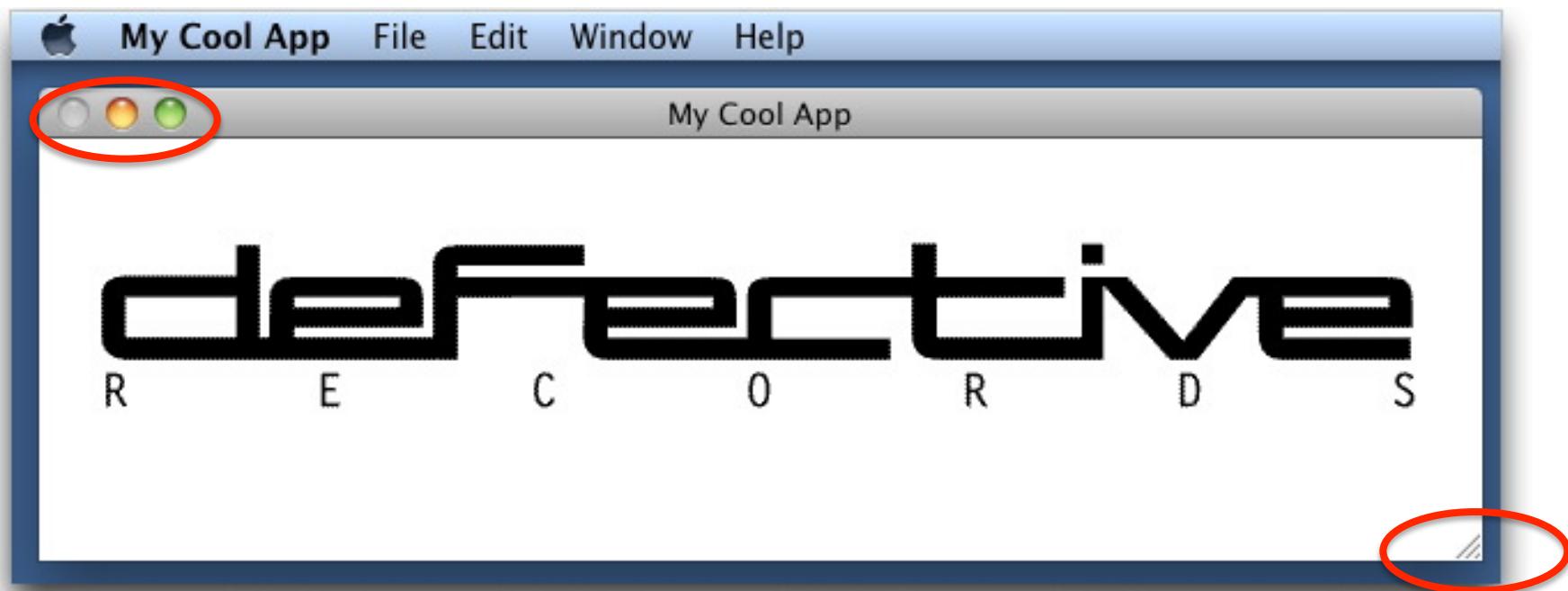
# Standalone object, presentation mode



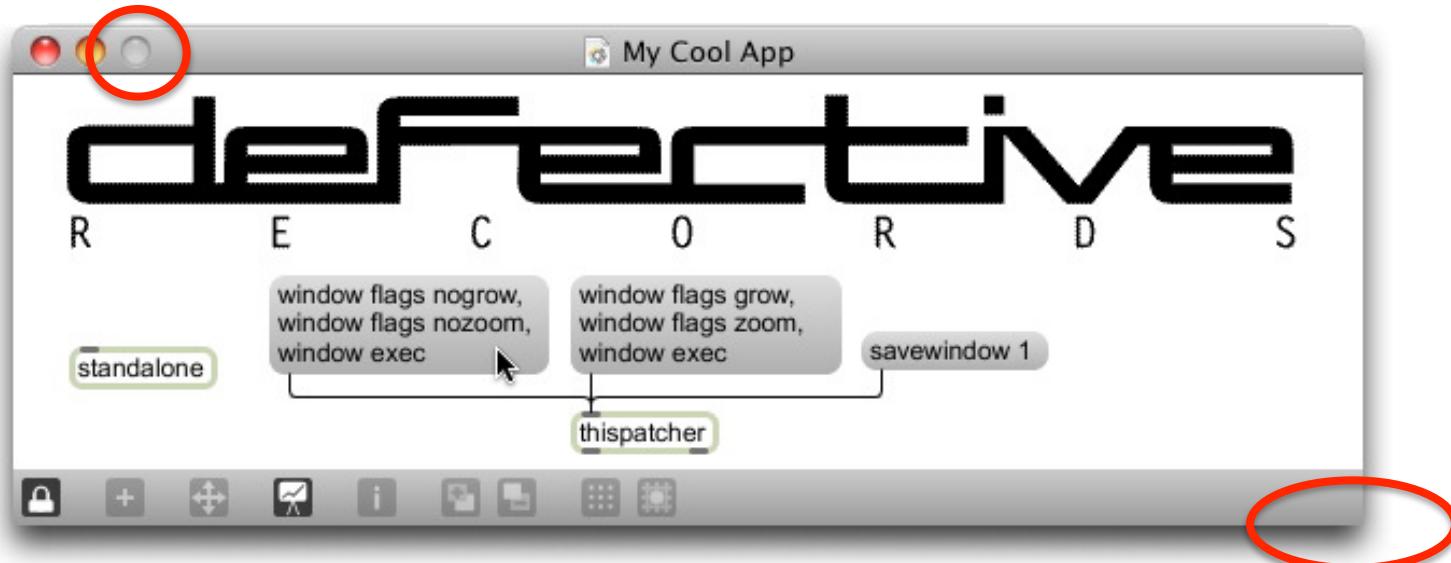
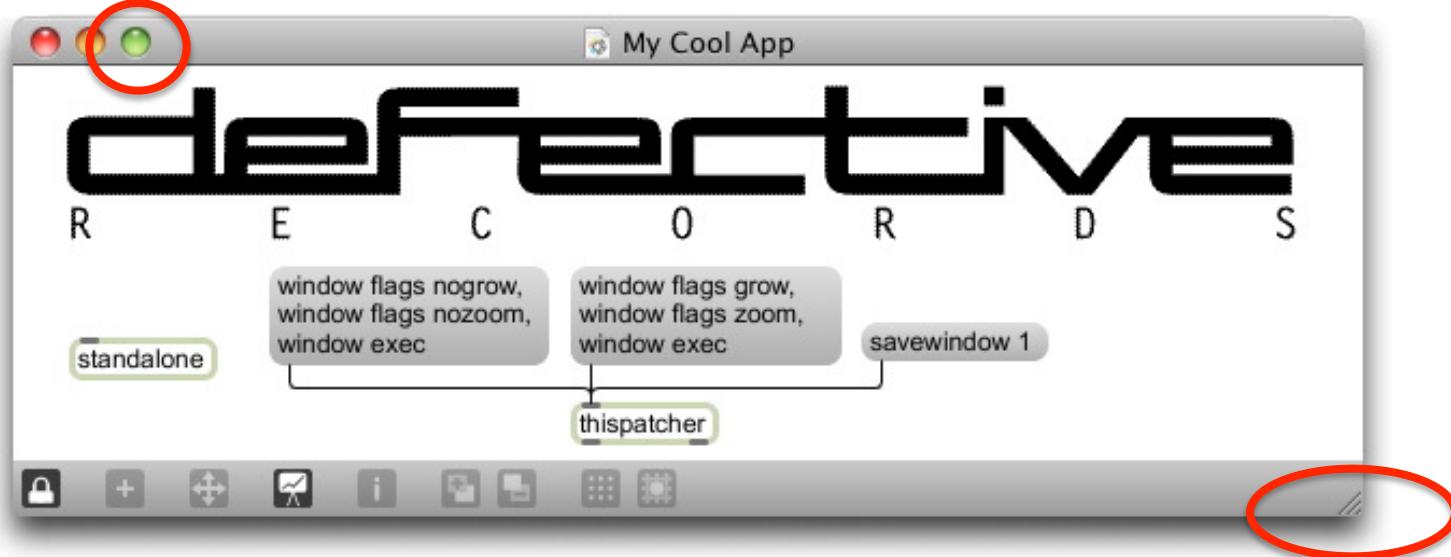
# Standalone object Inspector, Preferences, Max window, Loadbang defeating, etc...



# Current status



# Window settings



# Current status



# Custom app icon



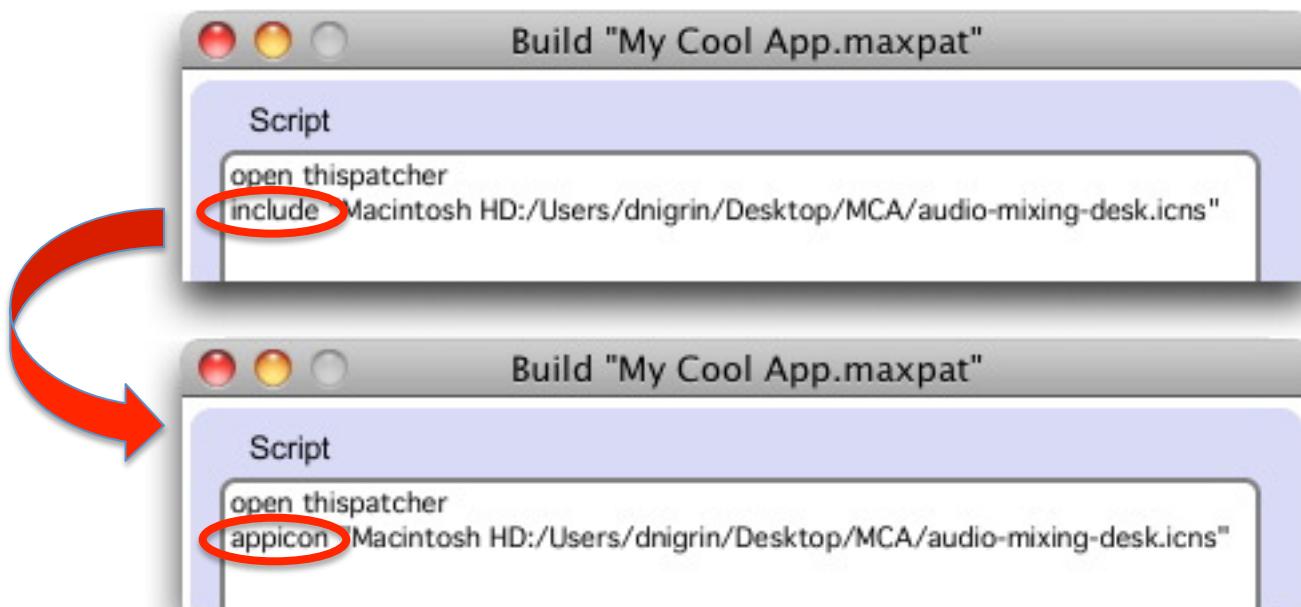
**Mac:** use.icns file (audio-mixing-desk.icns)

**Win:** use.ico file (audio-mixing-desk.ico)

icon from iconshock.com

# Build app to include custom icon

- “Include File...” in Build window, navigate to your icon file and select it
- Change “include” to “appicon”
- Now Build your app



# custom app icon result



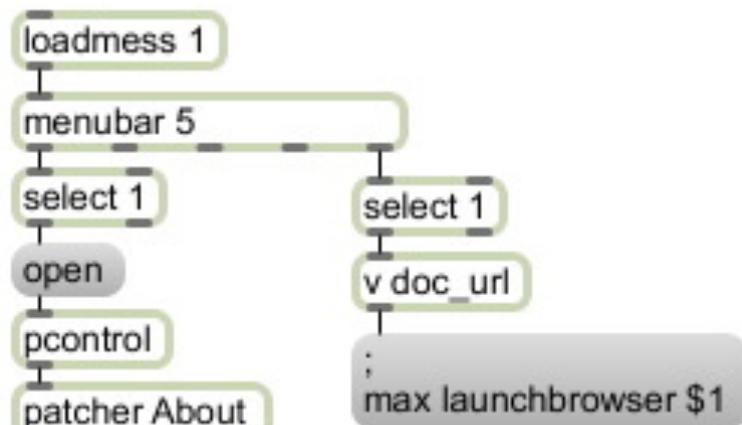
My Cool App



**Tip:** Include high resolution (256 x 256) versions within icon files

# Menu tweaking

- menubar object
- Make an About My Cool App entry, and one for Documentation



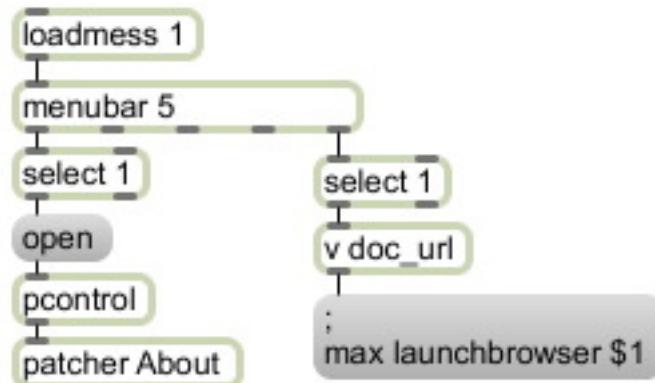
menubar script

```
#X about About My Cool App; ←
#X closeitem;
#X menutitle 5 Documentation;
#X item 5 1 My Cool App Documentation; } ←
#X end;
```

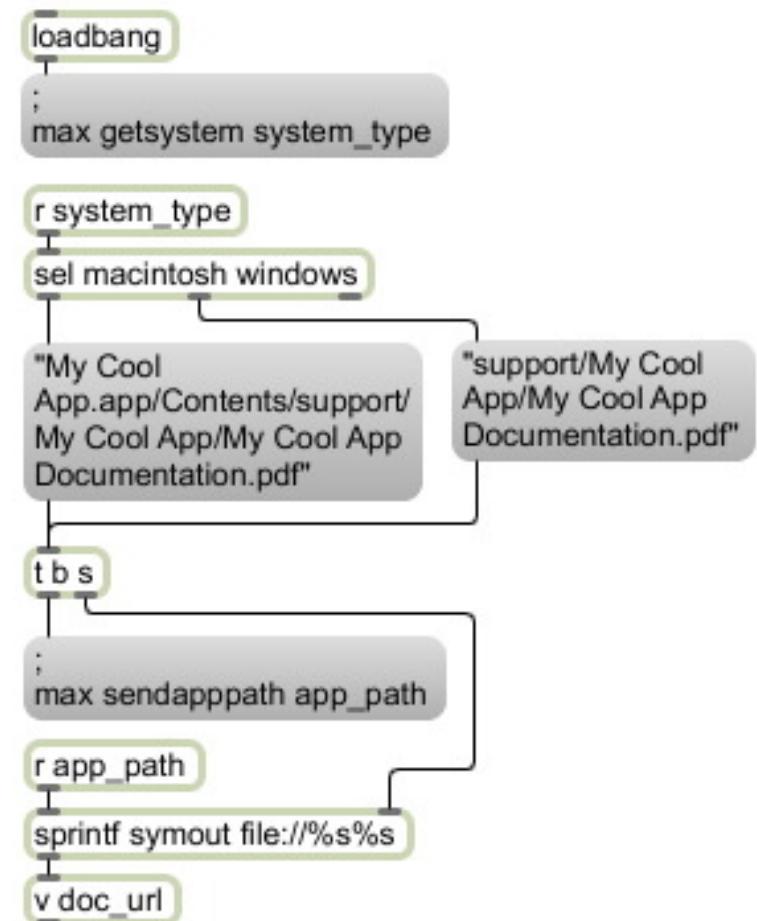
Cancel

OK

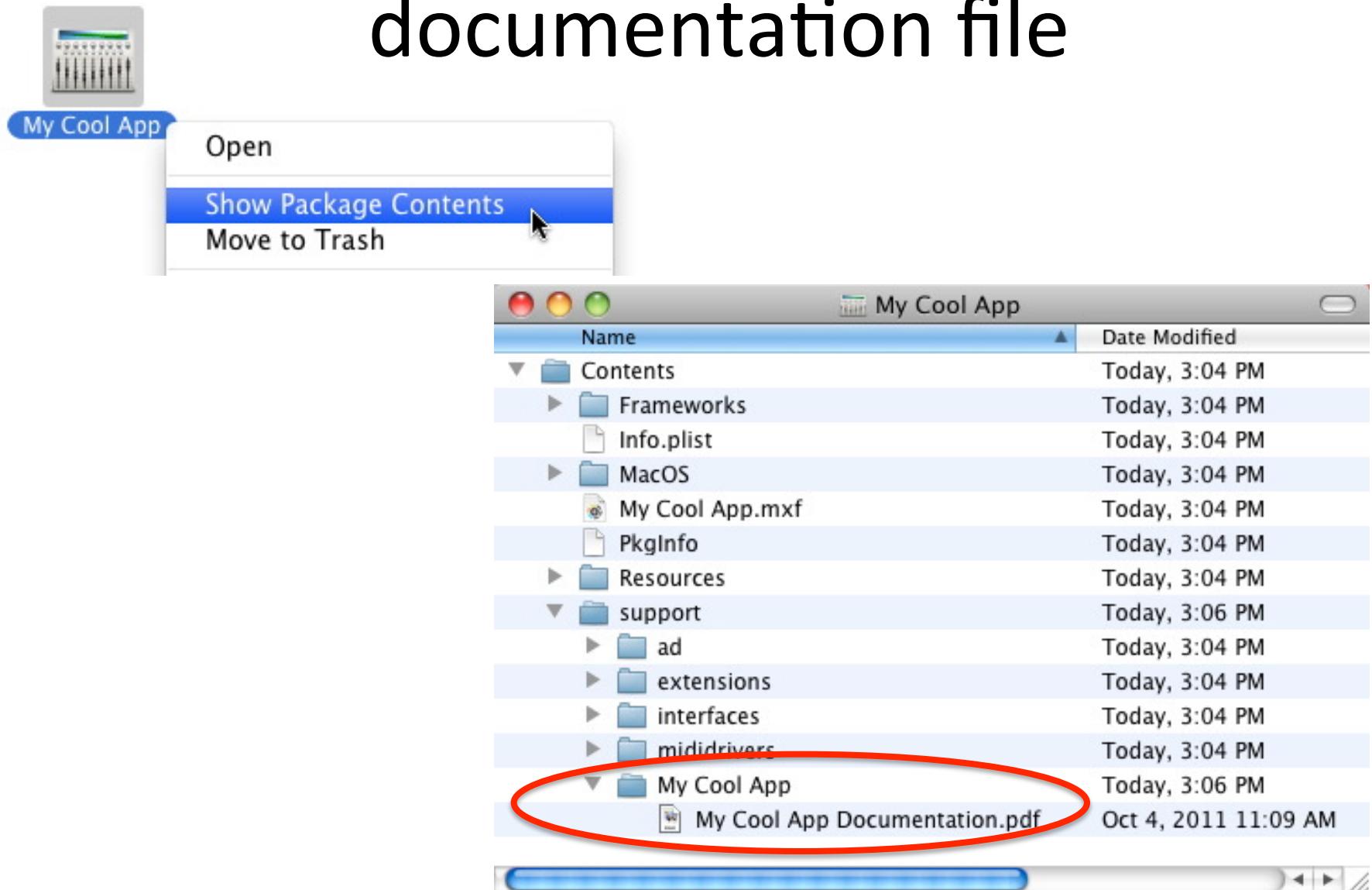
# Defining location of documentation file



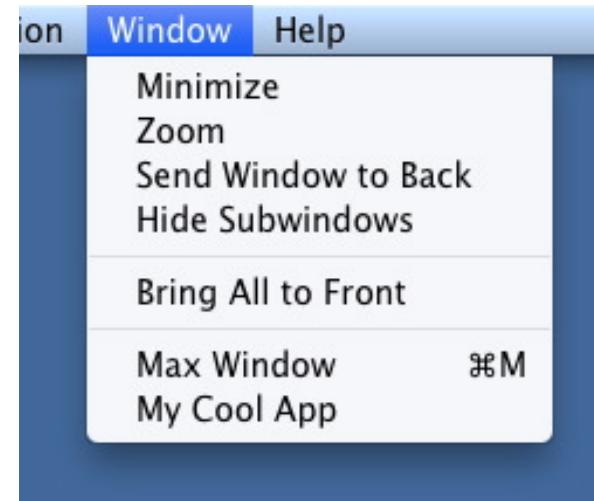
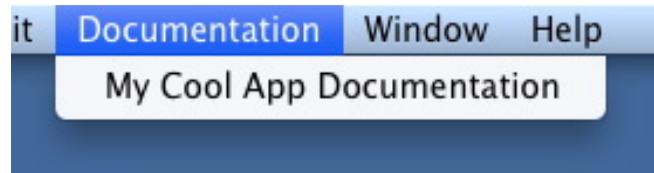
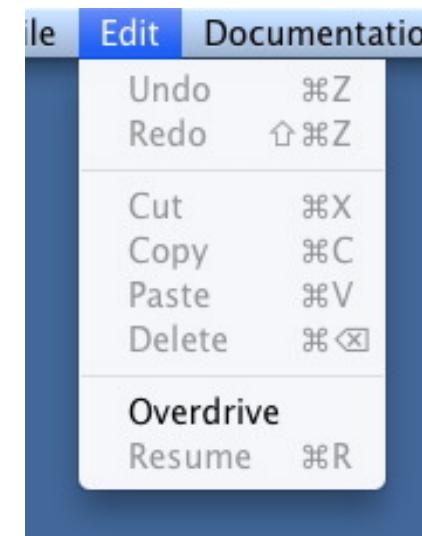
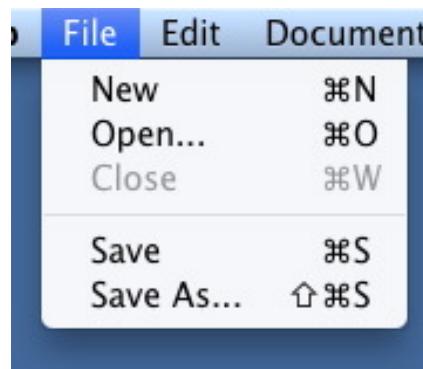
**Tip:** Open any file by sending max message  
[; max launchbrowser file://Path\_To\_File]



# Defining location of documentation file



# Current status of menus



# More menu tweaking

- maxinterface.json file
- **Mac:** within application bundle
  - Contents/support/interfaces/
- **Windows:** next to application executable
  - support/interfaces
- Save a copy of the original file, in case you make mistakes!

# maxinterface.json

- <http://jsonviewer.stack.hu/>
- Paste contents of maxinterface.json into “Text”

The screenshot shows a JSON viewer application window. At the top, there are two tabs: "Viewer" (which is selected) and "Text". Below the tabs, the left pane displays a hierarchical tree structure of the JSON object. The root node is "JSON", which contains a single child node "interface". The "interface" node has four children: "commands", "menus", "menubar", and "toolbars". The right pane shows a table with two columns: "Name" and "Value". There is one row in the table, corresponding to the "interface" node, with "interface" in the Name column and "..." in the Value column. At the bottom of the left pane, there is a search bar labeled "Search:".

Name	Value
interface	...

# maxinterface.json

Viewer    Text

JSON

interface

commands

menus

menubar

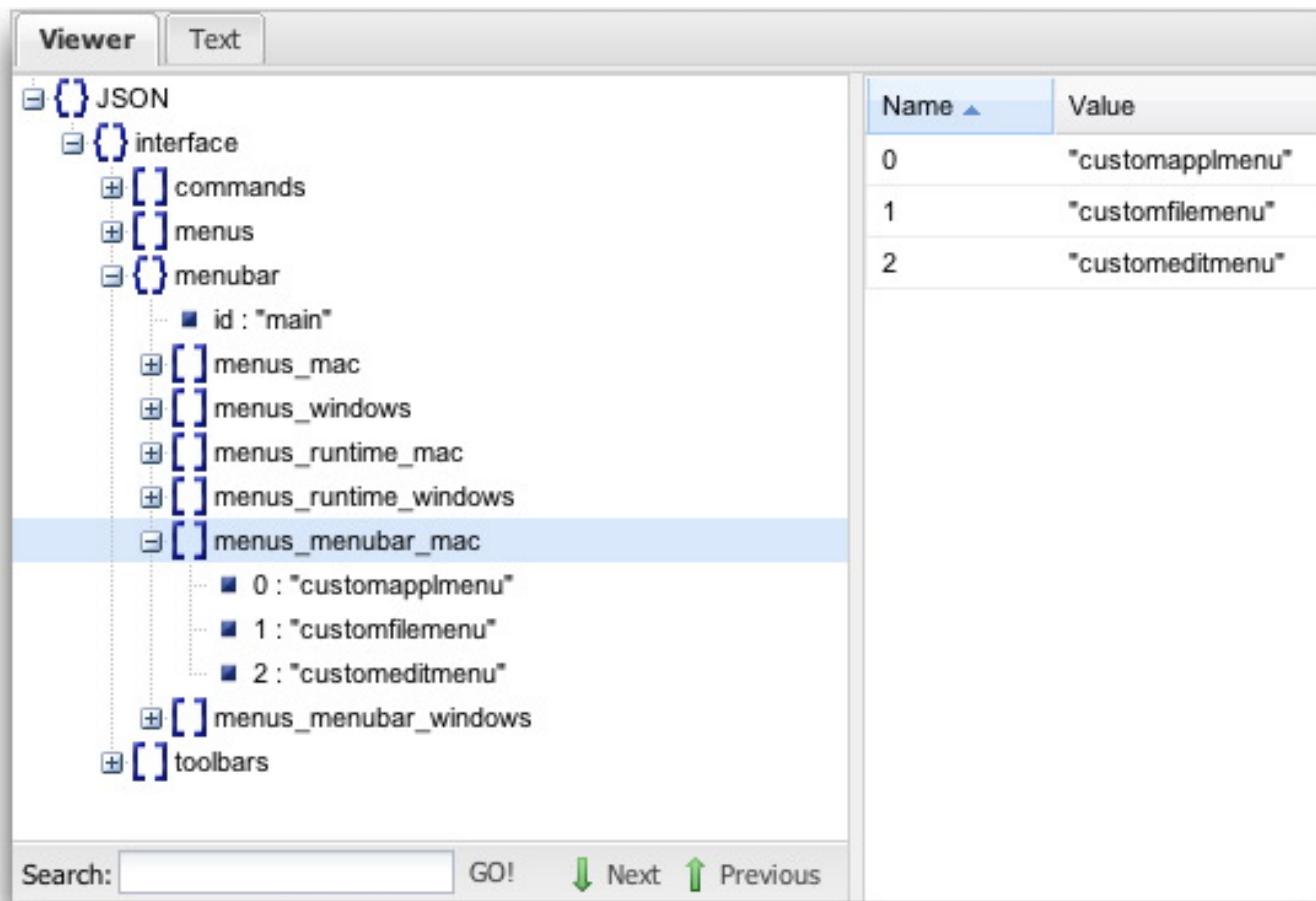
- id : "main"
- menus\_mac
- menus\_windows
- menus\_runtime\_mac
- menus\_runtime\_windows
- menus\_menuBar\_mac
  - 0 : "customapplmenu"
  - 1 : "customfilemenu"
  - 2 : "customeditmenu"
- menus\_menuBar\_windows

toolbars

Name ▲ Value

Name	Value
0	"customapplmenu"
1	"customfilemenu"
2	"customeditmenu"

Search:  GO!  



# maxinterface.json

The screenshot shows a software interface with a menu bar and a central panel for managing file menu configurations.

**File Menu:**

- New ⌘N
- Open... ⌘O
- Close ⌘W
- Save ⌘S
- Save As... ⌘⌘S

**Viewer Panel:**

The panel displays the structure of the `maxinterface.json` file under the `menus` key. The tree view shows the following structure:

- menus
  - 0
  - 1
  - 2
  - 3
    - id : "customfilemenu"
    - label : "File"
    - navkey : "F"
    - commands
    - commands\_runtime\_mac
      - 0 : "customnew"
      - 1 : "customopen"
      - 2 : "close"
      - 3 : "-"
      - 4 : "customsave"
      - 5 : "customsaveas"
    - commands\_runtime\_windows

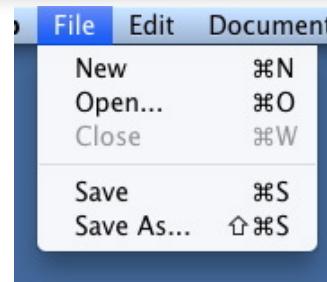
A search bar at the bottom left contains the text `customfilemenu`.

**Table View:**

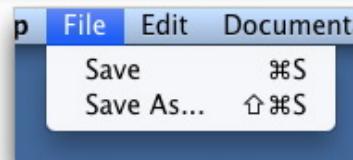
Name	Value
0	"customnew"
1	"customopen"
2	"close"
3	"-"
4	"customsave"
5	"customsaveas"

# maxinterface.json editing

```
{  
    "id" : "customfilemenu",  
    "label" : "File",  
    "navkey" : "F",  
    "commands" : [ "customnew", "customopen", "close", "-", "customsave", "customsaveas", "-", "backtomax" ],  
    "commands_runtime_mac" : [ "customnew", "customopen", "close", "-", "customsave", "customsaveas" ], ←  
    "commands_runtime_windows" : [ "customnew", "customopen", "close", "-", "customsave", "customsaveas", "-", "quit" ]  
},
```

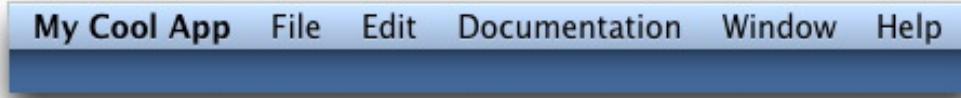


```
{  
    "id" : "customfilemenu",  
    "label" : "File",  
    "navkey" : "F",  
    "commands" : [ "customnew", "customopen", "close", "-", "customsave", "customsaveas", "-", "backtomax" ],  
    "commands_runtime_mac" : [ "customsave", "customsaveas" ], ←  
    "commands_runtime_windows" : [ "customnew", "customopen", "close", "-", "customsave", "customsaveas", "-", "quit" ]  
},
```

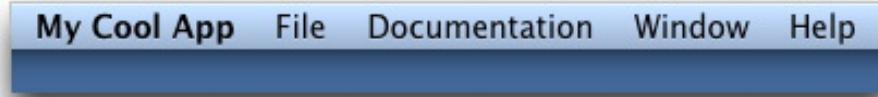


# maxinterface.json editing

```
"menubar" : {  
    "id" : "main",  
    "menus_mac" : [ "applmenu", "filemenu", "editmenu", "viewmenu", "objectmenu", "arrangemen  
    "menus_windows" : [ "filemenu", "editmenu", "viewmenu", "objectmenu", "arrangemen  
    "menus_runtime_mac" : [ "applmenu", "filemenu", "editmenu", "windowmenu", "helpmenu" ],  
    "menus_runtime_windows" : [ "filemenu", "editmenu", "windowmenu", "helpmenu" ],  
    "menus_menubar_mac" : [ "customapplmenu", "customfilemenu", "customeditmenu" ],  
    "menus_menubar_windows" : [ "customfilemenu", "customeditmenu" ]  
},
```

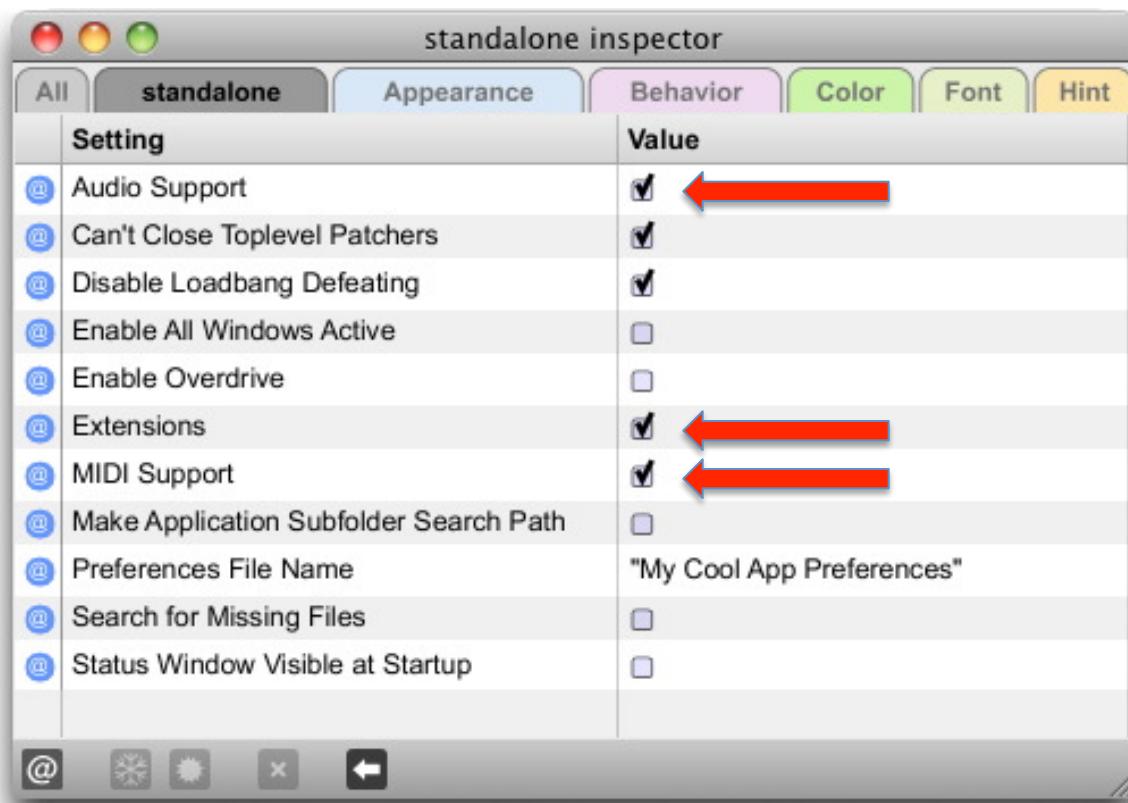


```
"menubar" : {  
    "id" : "main",  
    "menus_mac" : [ "applmenu", "filemenu", "editmenu", "viewmenu", "objectmenu", "arrangemen  
    "menus_windows" : [ "filemenu", "editmenu", "viewmenu", "objectmenu", "arrangemen  
    "menus_runtime_mac" : [ "applmenu", "filemenu", "editmenu", "windowmenu", "helpmenu" ],  
    "menus_runtime_windows" : [ "filemenu", "editmenu", "windowmenu", "helpmenu" ],  
    "menus_menubar_mac" : [ "customapplmenu", "customfilemenu" ],  
    "menus_menubar_windows" : [ "customfilemenu", "customeditmenu" ]  
},
```



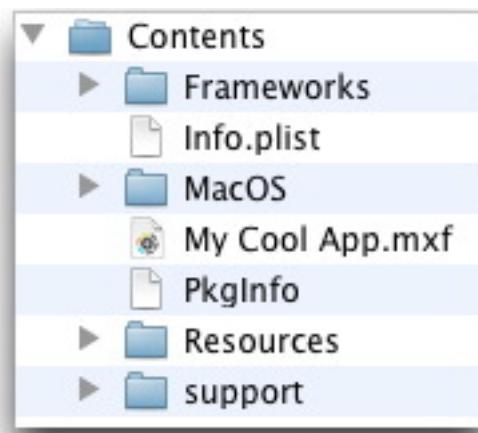
# Remove unnecessary files

- Use standalone object to exclude if not needed
  - Audio Support
  - MIDI Support
  - Extensions



# Remove unnecessary files

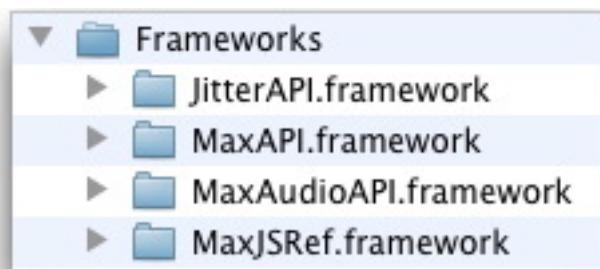
## Mac – Top Level



- Leave alone
  - PkgInfo
  - My Cool App.mxf
  - MacOS folder
- Info.plist we will work on later

# Remove unnecessary files

## Mac – Frameworks folder

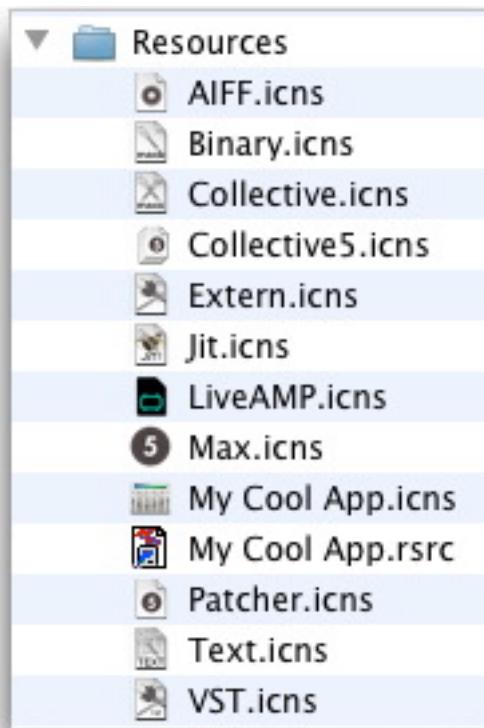


Delete what you don't use!

- No Jitter – delete JitterAPI.framework
- No Javascript – delete MaxJSRef.framework
- No audio – delete MaxAudioAPI.framework

# Remove unnecessary files

## Mac – Resources folder

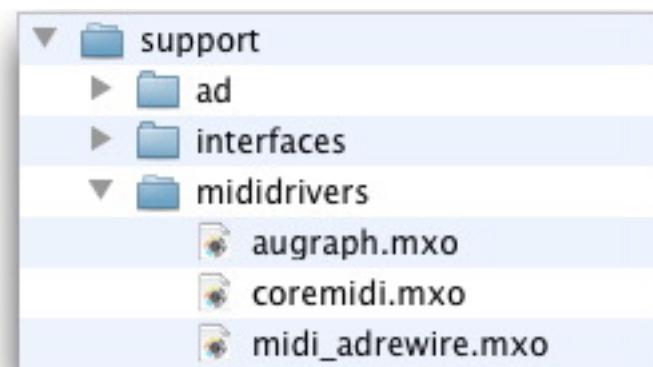


Delete everything except

- My Cool App.icns
- My Cool App.rsrc

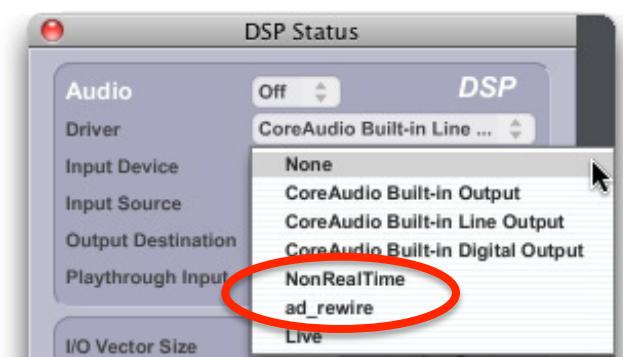
# Remove unnecessary files

## Mac – support folder



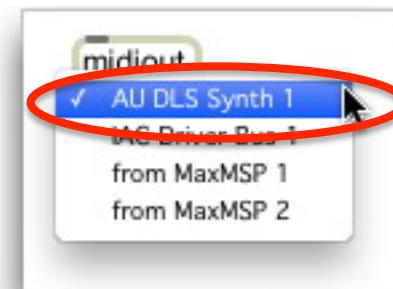
### ad folder

- No need for Rewire? Delete
  - ad\_rewire.mxo
  - MSPReWireDevice.bundle
- No need for Non Realtime? Delete
  - ad\_nonreal.mxo



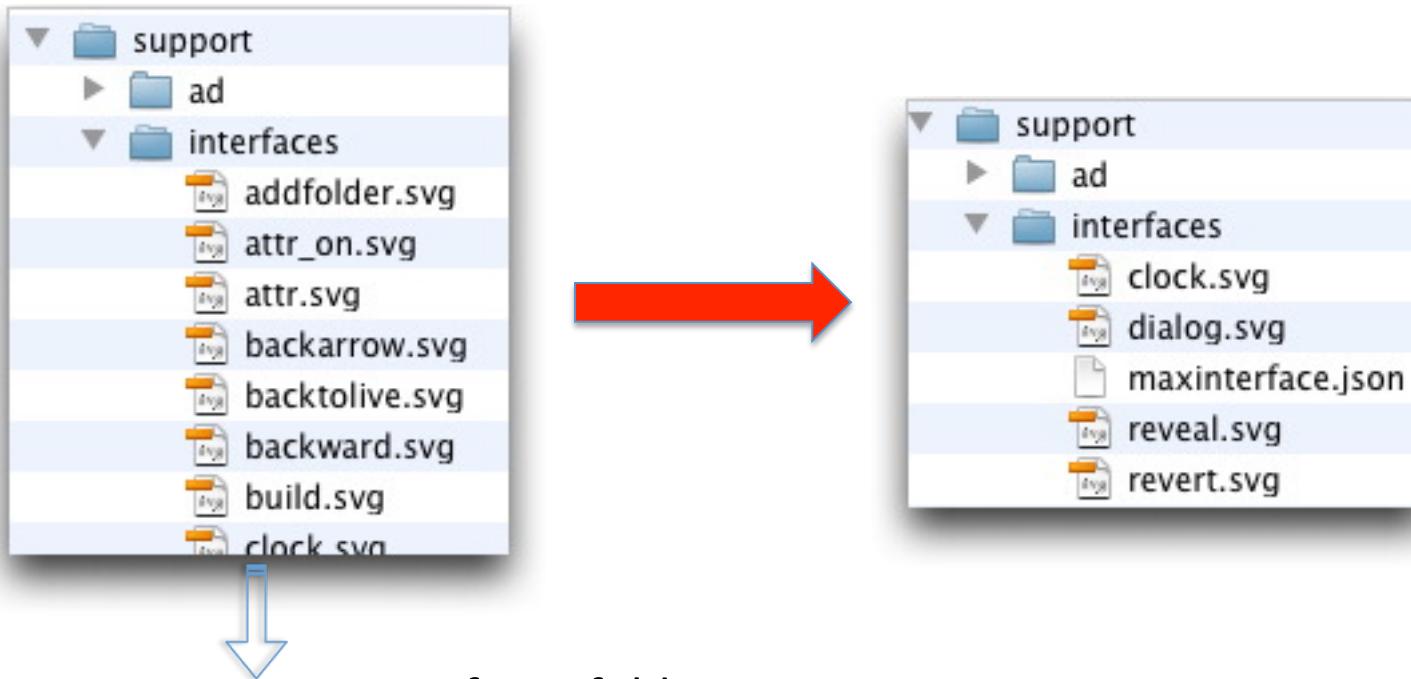
### mididrivers folder

- Not using General MIDI? Delete
  - augraph.mxo
- Not using ReWire? Delete
  - Midi\_adrewire.mxo



# Remove unnecessary files

## Mac – support folder

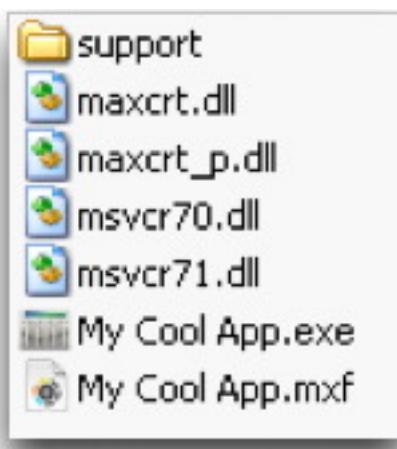


### Interfaces folder

- Be sure to keep maxinterface.json (your custom one!)
- I also keep those files necessary for Max window – leave only what you need (experiment!)

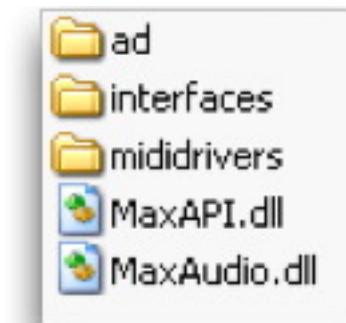
# Remove unnecessary files

## Windows



Top level - leave alone

- maxcrt.dll
- maxcrt\_p.dll
- msvcr70.dll
- msvcr71.dll
- My Cool App.mxf



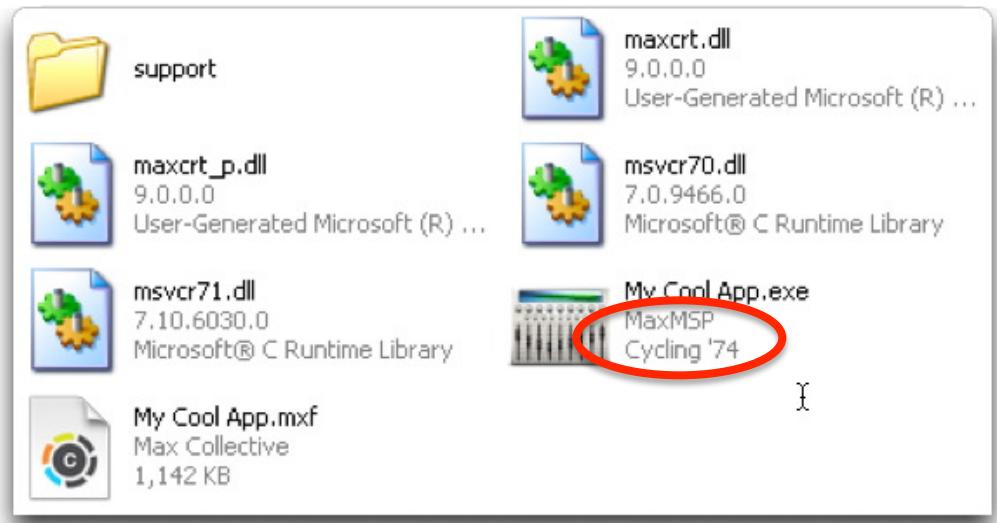
support folder - leave alone

- MaxAPI.dll
- MaxAudio.dll

ad, interfaces and mididrivers folders

- Same as Mac

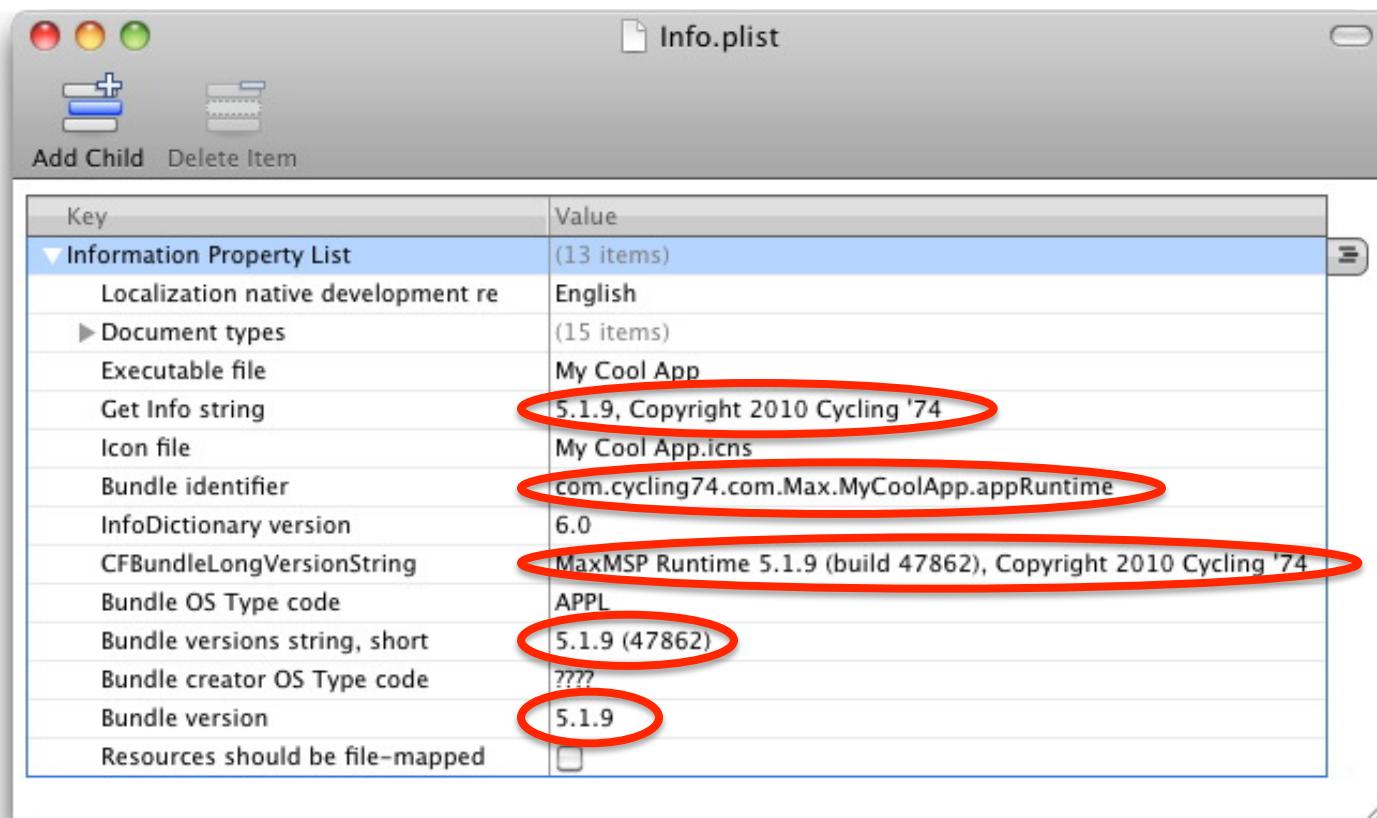
# Application meta information



# Application meta information

## Mac

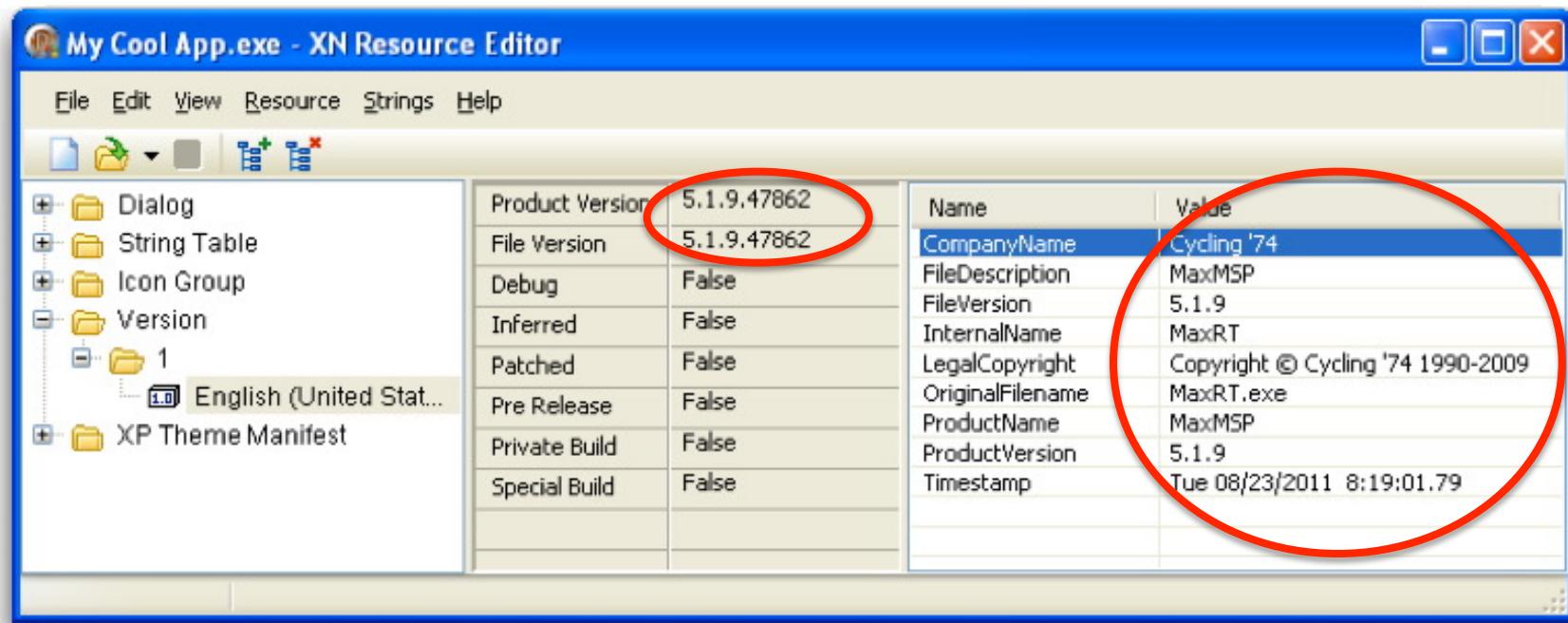
- Modify Info.plist file
- Can use Property List Editor (installed with Apple Developer Tools)



# Application meta information

## Windows

- Must use resource editor
- E.g. “XN Resource Editor” (free)
- Can also delete unneeded icons here



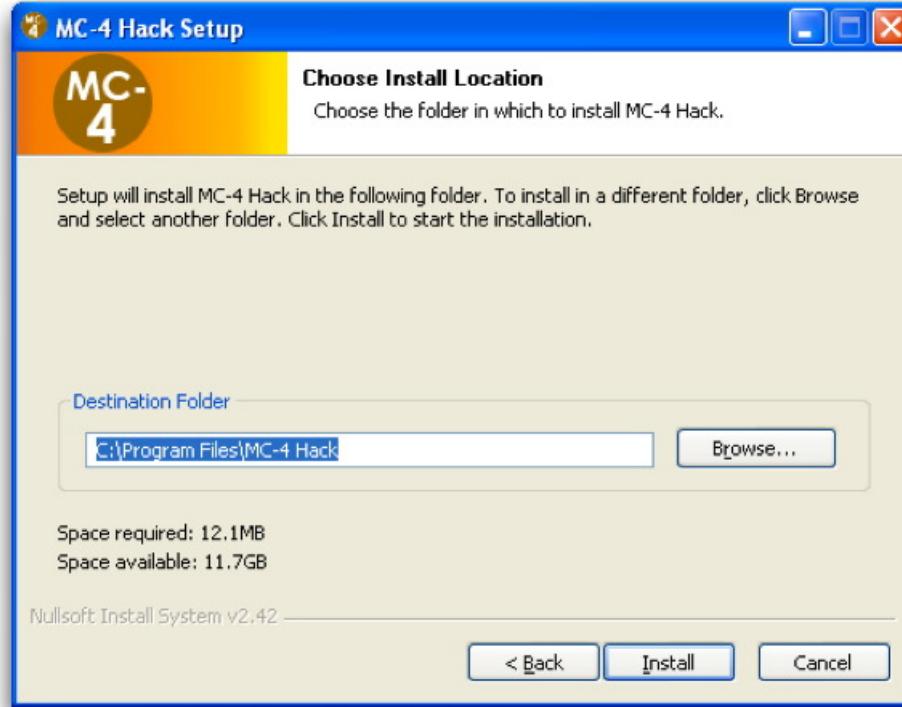
# Installer/Uninstaller Mac

- Distribute as .dmg file, double click to mount, drag to Applications folder
- I use “DropDMG” (\$24 USD) – plenty of other (free) ways to do it
- Uninstallation – just delete application



# Installer/Uninstaller Windows

- Can just distribute as .zip, but slicker to build installer that places files in right places, Start menu entries, Registry entries, uninstaller, etc...
- I use Nullsoft Scriptable Install System (NSIS) (free and open source)
- Can get fancy – e.g. test to see if Java installed, if not, divert user to install it first.



# Application Protection

- Commercial option now available – but \$\$
  - <http://www.excelsoftware.com>
- My approach
  - Lots of small, relatively easily defeat-able, but in aggregate, “good enough” protections
  - Username/password combo
  - Username is based on user’s \*real\* name or other credentials
  - Tie authorization to specific machine attribute
  - If providing demo version, don’t have full version functionality within it
  - Don’t make the full version publicly downloadable
  - Obfuscate within your Max patches themselves (in case people open up your collectives)
  - Make software relatively cheap, and make it up on volume!

# Miscellaneous

- Why does double-clicking a Max patcher launch my standalone instead of Max (on a Mac)?

The screenshot shows two versions of the Info.plist file in a Mac OS X-style plist editor. A red arrow points from the top window down to the bottom window, indicating a change.

**Top Window (Original State):**

Key	Value
Information Property List	(13 items)
Localization native development re	English
► Document types	(15 items)

**Bottom Window (Modified State):**

Key	Value
Information Property List	(13 items)
Localization native development re	English
► Document types	(2 items)
► Item 0 (Max collective)	(4 items)
► Item 1 (Application)	(4 items)

# Thank You...

All at Cycling '74, the Cycling '74 forums,  
and the (defunct) Max mailing list!

Presentation available at:

<http://defectiverecords.com/expo74/standalones.pdf>