UIDE

User Interface Design Environments
In the Beginning...

- There were punch cards
- and it was bad
and Then There was Graphics...
and it was good, but there were problems.

- View
  - View Creation
  - View Debugging
- Controller
  - Controller Creation
- Model
  - Event debugging
  - Visual Programming
Smalltalk

- No Direct Manipulation
- Separated Classes
Smalltalk

- No Direct Manipulation
- Separated Classes
Visualizing the View

- For IDE’s the first thing that was done to help program development
- Used widely for rapid prototyping
- Demo of .Net
- Demo of JSpy
1. Assuming SpyJ started correctly, click "Init Spy" button. Full SpyJ window will appear.
2. Resize the SpyJ and this demo window to not overlap each other
3. Press/release mouse button on "Point" button, move to this list box and press any key.

```
this = java.awt.List[1,0,46,551x64,selected=2. Resize the SpyJ and this
getClass() = class java.awt.List
getBounds(null) = java.awt.Rectangle[x=0,y=46,width=551,height=64]
getLocationOnScreen() = java.awt.Point[x=4,y=588]
getFont() = java.awt.Font[family=Dialog,name=Dialog,style=plain,size=12]
getName() = list1
getParent() = java.awt.Panel[panel26,0,551x110,layout=java.awt.GridBagLayout]
isEnabled() = true
isShowing() = true
isVisible() = true
getPreferredSize() = java.awt.Dimension[width=549,height=64]
```

- TabbedPane[breakpoint1,0,0,0x0,invalid,layout=java.awt.GridBagLayout]
- This component is not displayed by the trial version
- Panel[breakpoint,0,0,0x0,invalid,layout=java.awt.GridBagLayout]
- This component is not displayed by the trial version
- Panel[breakpoint,0,0,0x0,invalid,layout=java.awt.GridBagLayout]
Problems with using a Graphical View

- Keeping code and internal model of view synchronized
- Editing code that is machine generated
- Creating a prototype view can lead people to think the application is closer to completion than reality.
Visualizing Controller

- First Done with NeXT
- Demo with Interface Builder
- No known software for visual controller testing
Problems with using a Graphical Controller

- No code to edit
- Separation from program code
- Accessibility problems
Visualizing the Model

- Model usually seen as storage of memory
- Debugging tracks memory usage
- Demo of DDD (Data Display Debugger)
Problems with using a Graphical Model

- A lot of information to view at one time. Usually only viewing part of the model.
- A lot of algorithms do not have a understandable visual model yet.
Visual Programming

- Programming without syntax
- Used for people that don’t understand formal programming
- Demo of Automator
- Used in professional programming for when graphical visualization works better
- Demo of Quartz Composer
This VI continuously generates two signals: a pure sine wave of variable frequency and amplitude and a white noise signal of variable amplitude. The noise is then added to the pure sine. The sine wave with and without the noise are then shown in a time domain graph. Additionally an FFT is calculated for both signals and the results are then shown in the frequency domain graph. Note that the square shaped functions are subroutines in the form of subVIs.
Render in Image

This macro patch renders its subpatches into an image.

Render in Image basically creates a new rendering destination in the form of an image. The subpatches of this macro patch then render to that new destination instead of the original one. The resulting
Getting Started

To view the actions for an application, click the application.
To show all the actions, click the Applications folder.
To find an action, enter a word or phrase in the search field.

To add an action to the workflow, drag it into the workflow.
To open a saved workflow from the library, double-click it.
Problems with Visual Programming

- Hard to reproduce "code"
- Some problems hard to understand visually
Programs

- View
  - Creation: .Net (Visual Studio)
  - Debugging: jSpy
- Controller
  - Creation: Interface Builder
  - Debugging: Nothing Yet
- Model
  - Creation: Visual Programming
  - Debugging: DDD
Conclusion

- Problem of double learning visual path & code path
- Accessibility problems
- Problem of trying to keep machine and developer code synchronized
Questions?