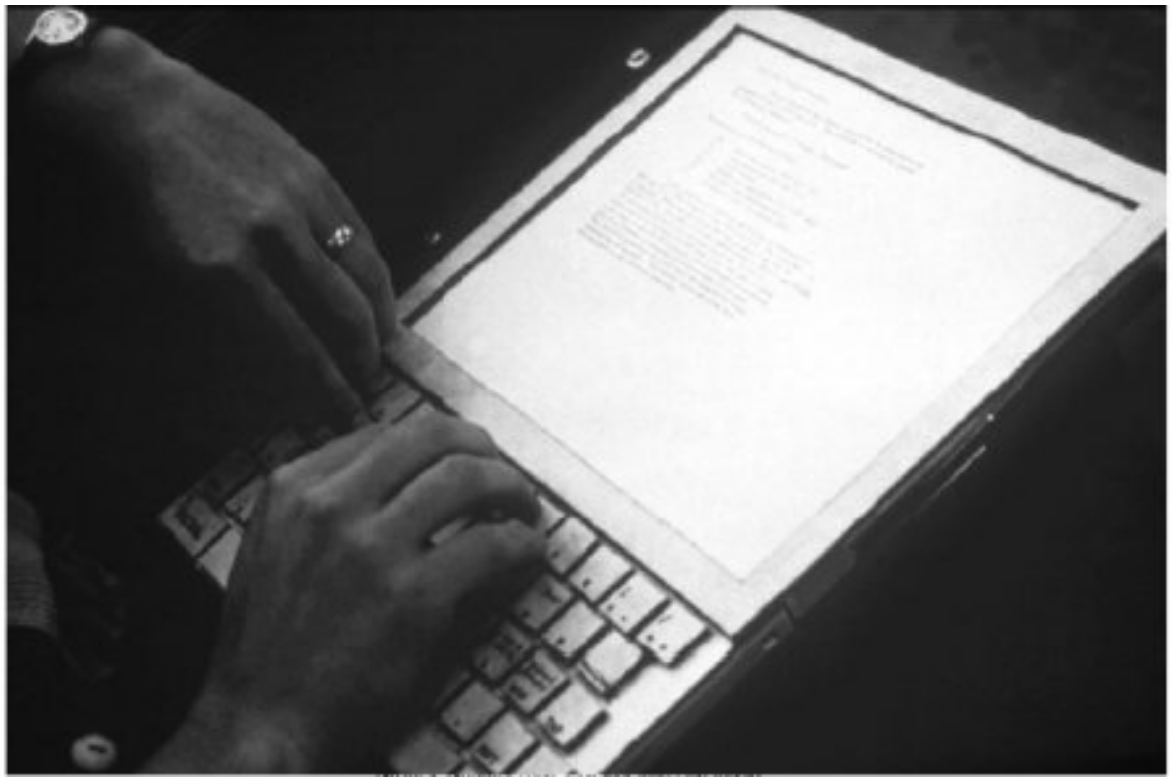
Squeak on Android

Presented by Dmitry Golubovsky

2011

40 Years Ago

Dynabook (Alan Kay, 1972)

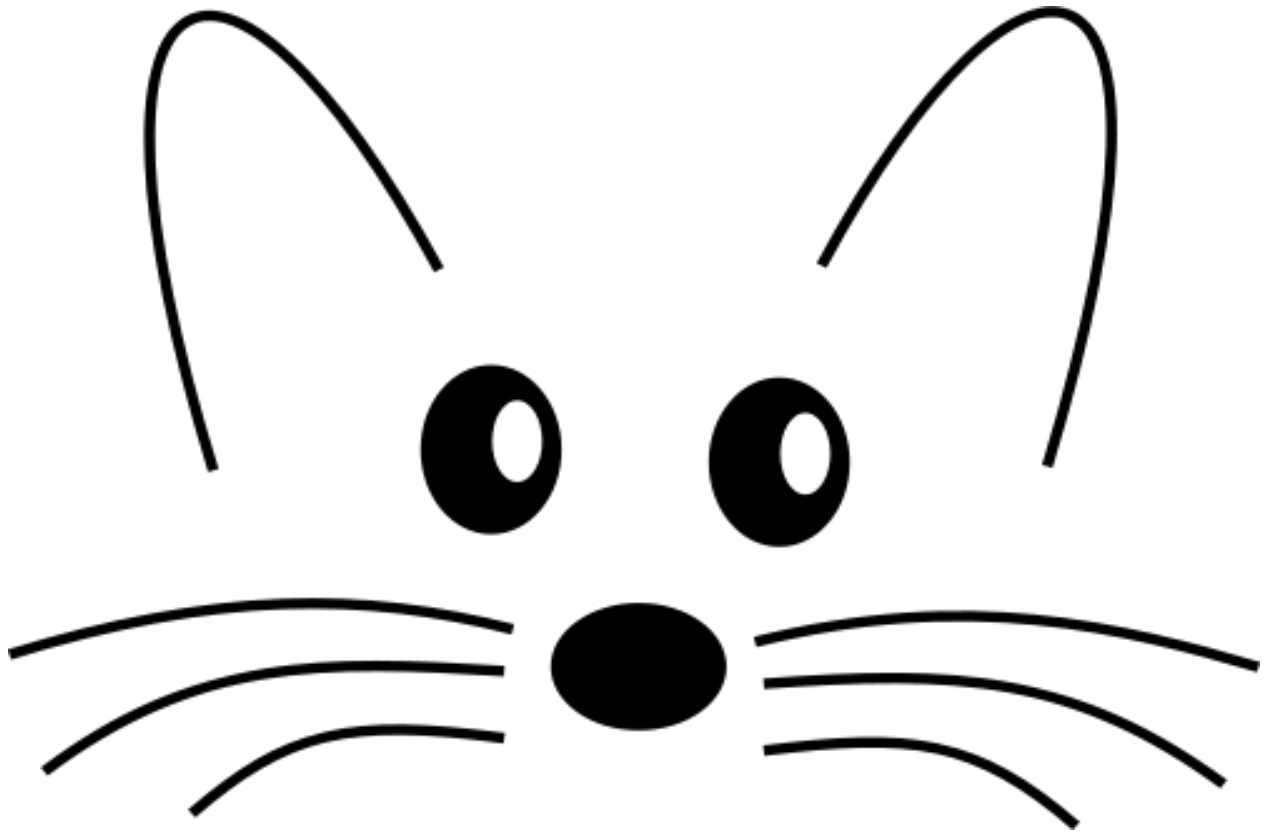


These Days

Android Tablets

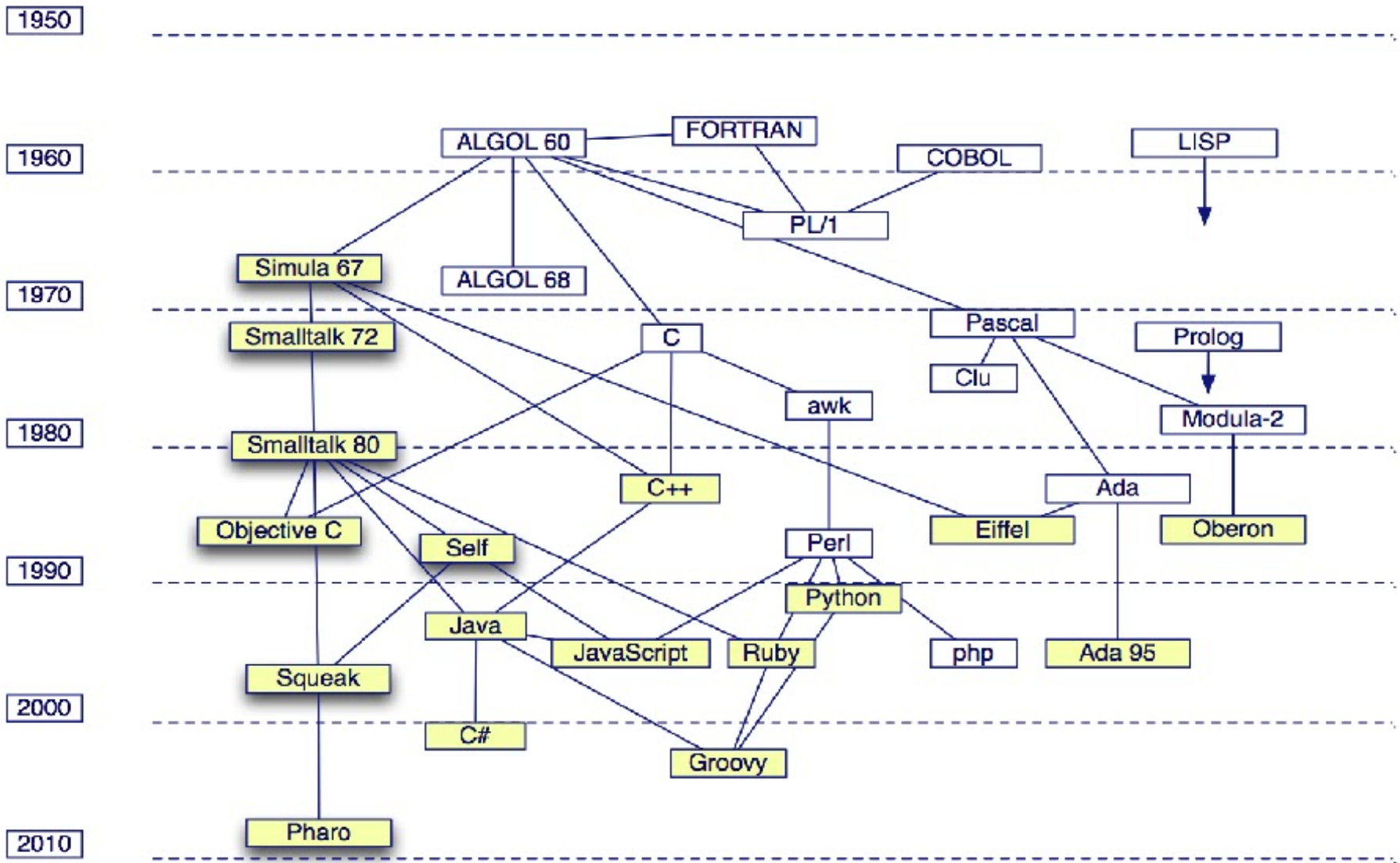
Squeak

Squeak is a highly portable, open-source Smalltalk with powerful multimedia facilities. Squeak is the vehicle for a wide range of projects from educational platforms to commercial web application development.(source: squeak.org)



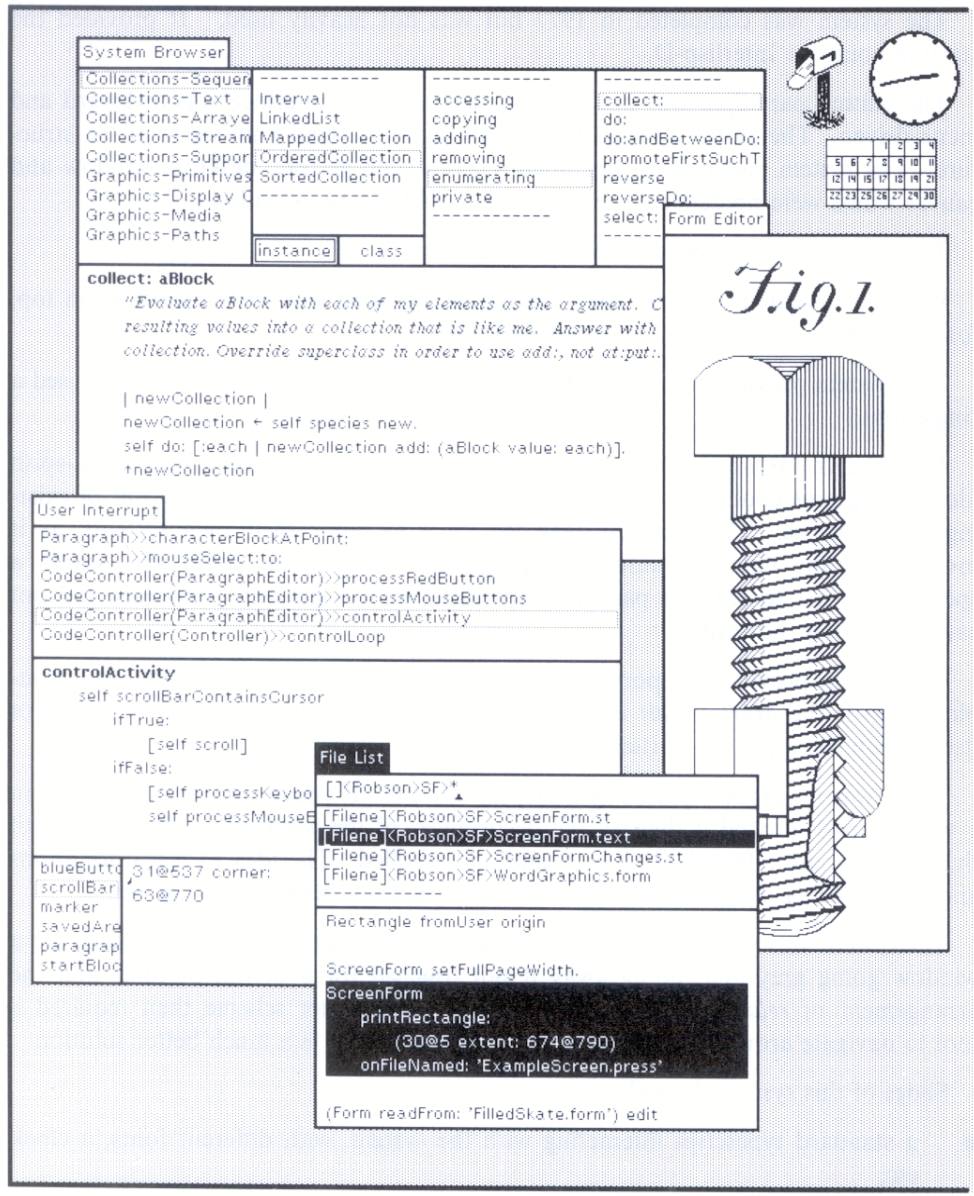
History of Smalltalk

(source unknown)



Early Smalltalk Graphics (ca.1980)

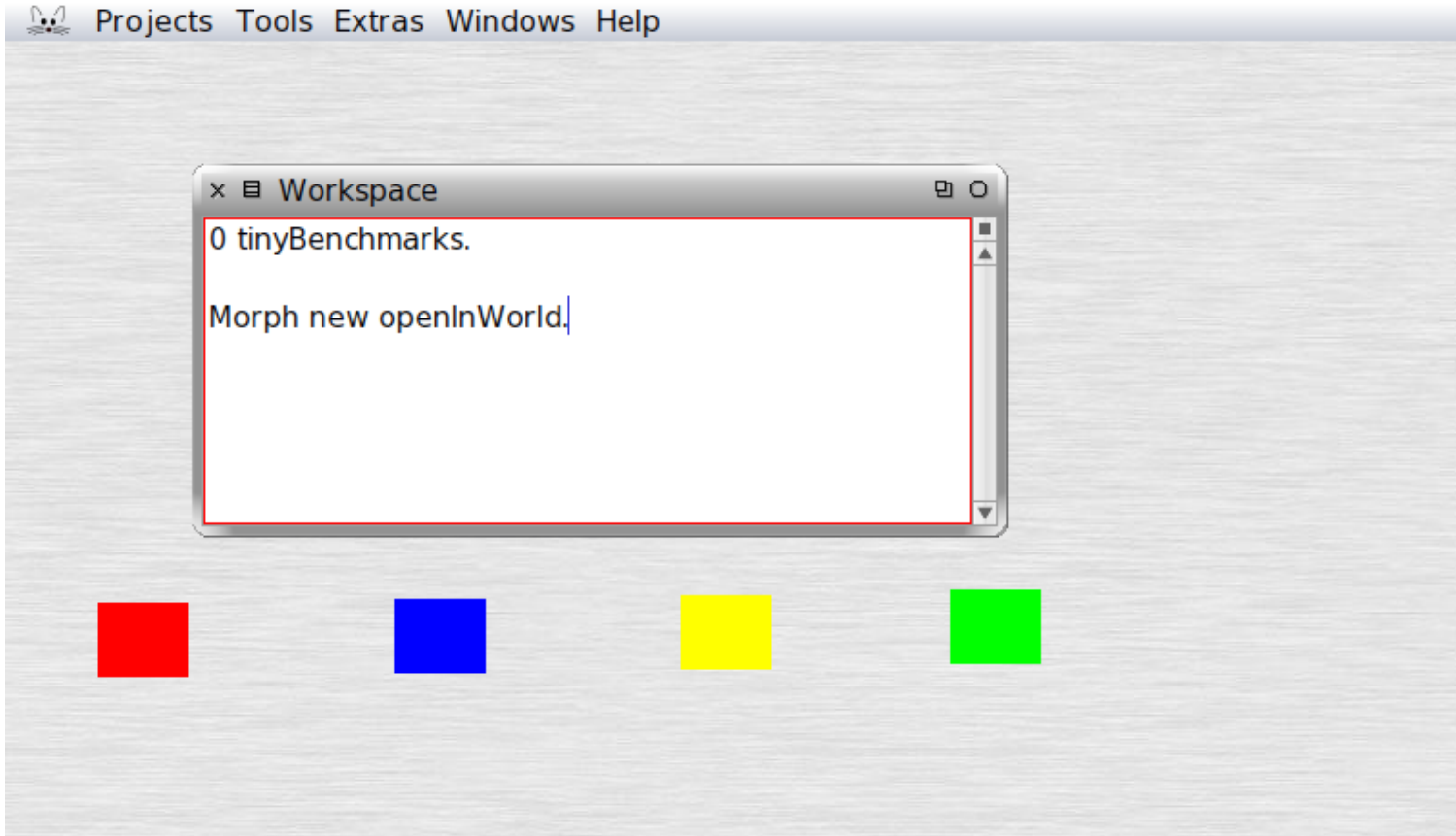
(source: "Methodology of Window Management")



In Smalltalk:

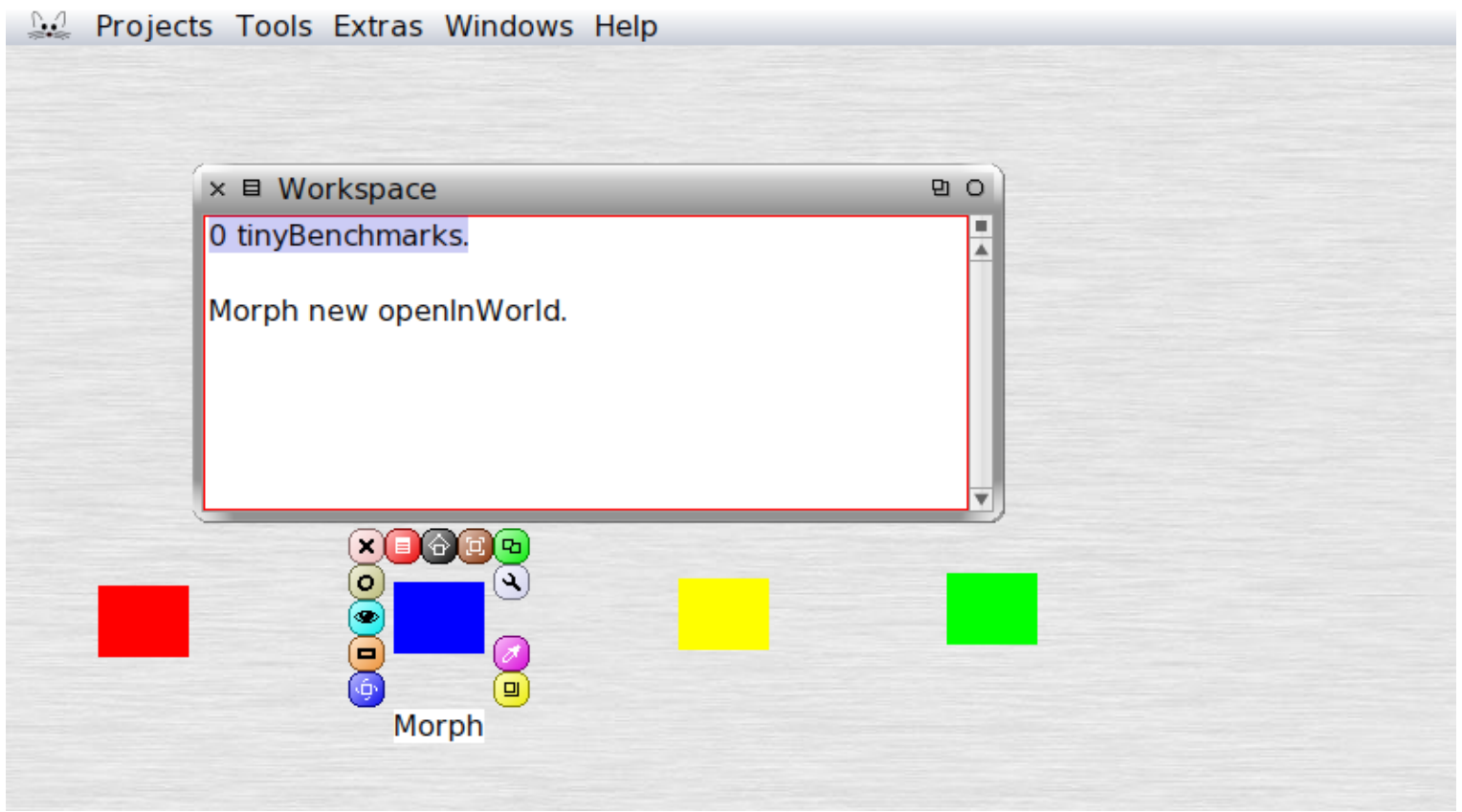
- Everything is an object
- Objects are controlled by sending messages
- Classes are objects, too
- Full introspection (the whole program structure is observable and can be modified at run time)
- New instances of classes created at run time, programmatically or interactively
- Classes may be modified at run time, reflects in all instances' behavior immediately
- GUI elements are instances of classes that can be drawn on screen surface
- Creation/deletion/change of behavior of any GUI elements at run time, interactively.

Change Color of a GUI Element



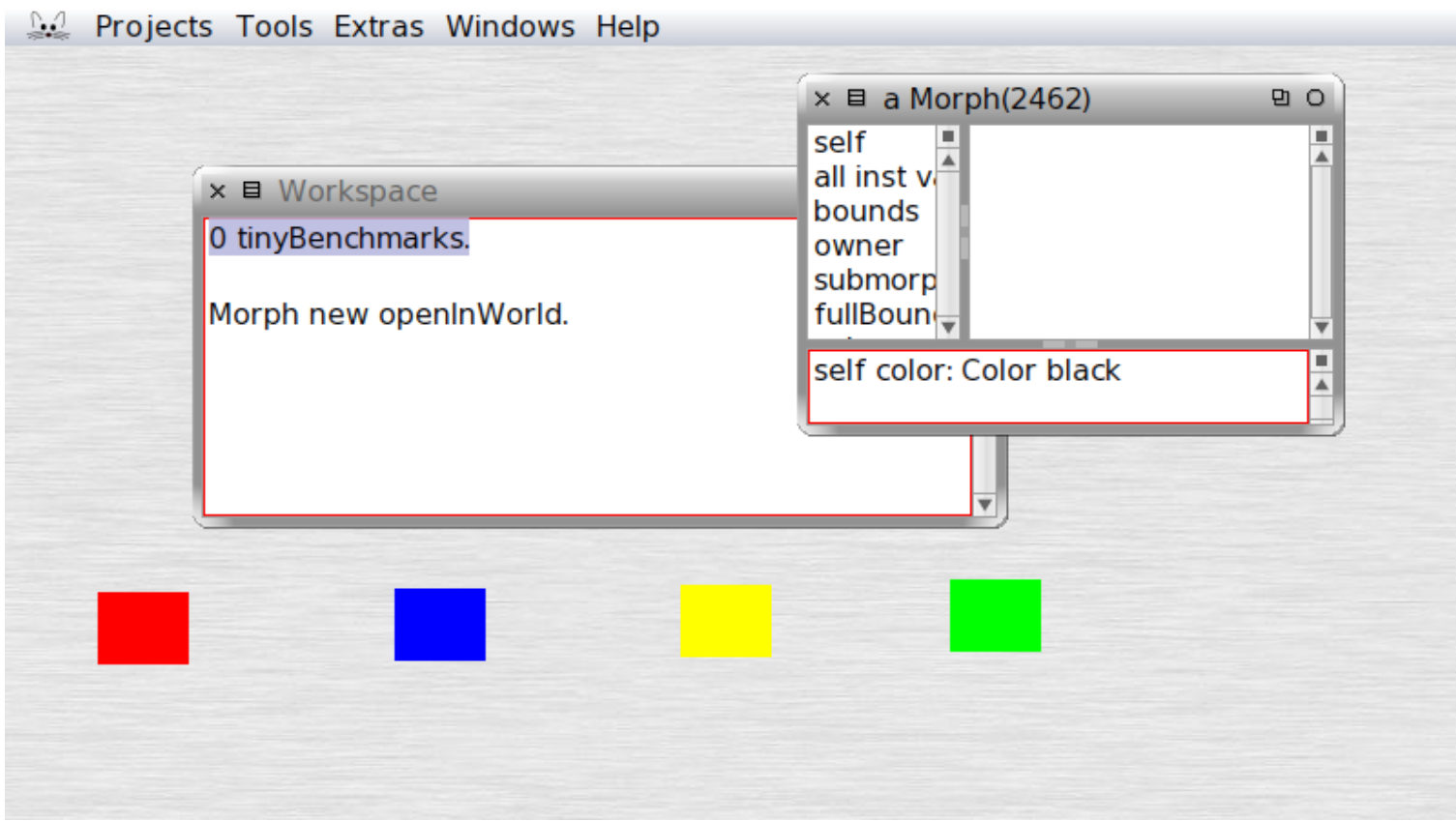
A Morph is Blue

Change Color of a GUI Element
(cont'd)



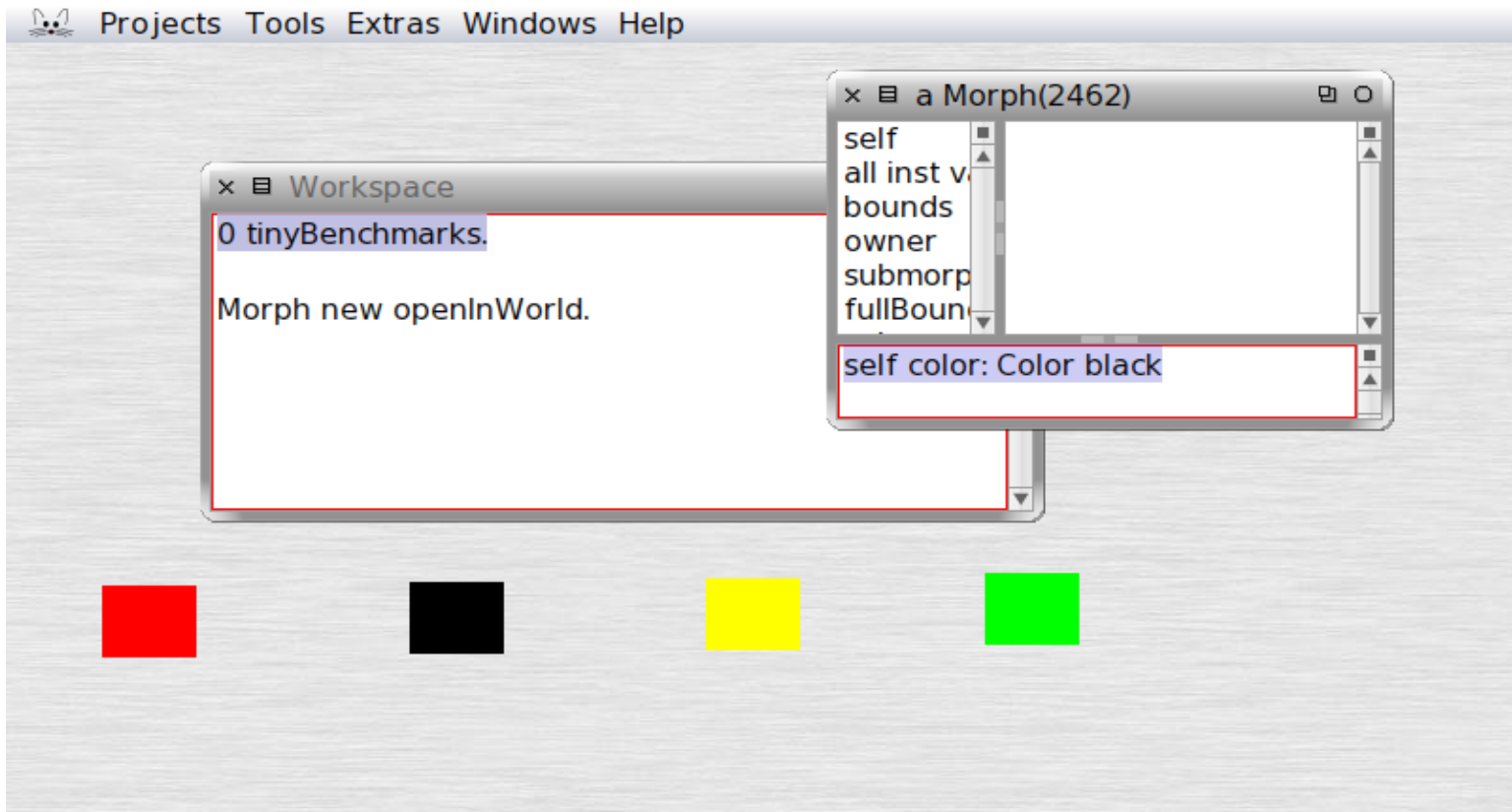
Bring up Halo

Change Color of a GUI Element
(cont'd)



Bring up Inspection Window

Change Color of a GUI Element
(cont'd)



Execute the Code
The Morph is Black Now

Main Components of Squeak

Virtual Machine

Squeak Virtual Machine is an executable program for the host operating system where Squeak is running

Image File

Image File is a snapshot of the Virtual Machine's heap (object memory)

Changes File

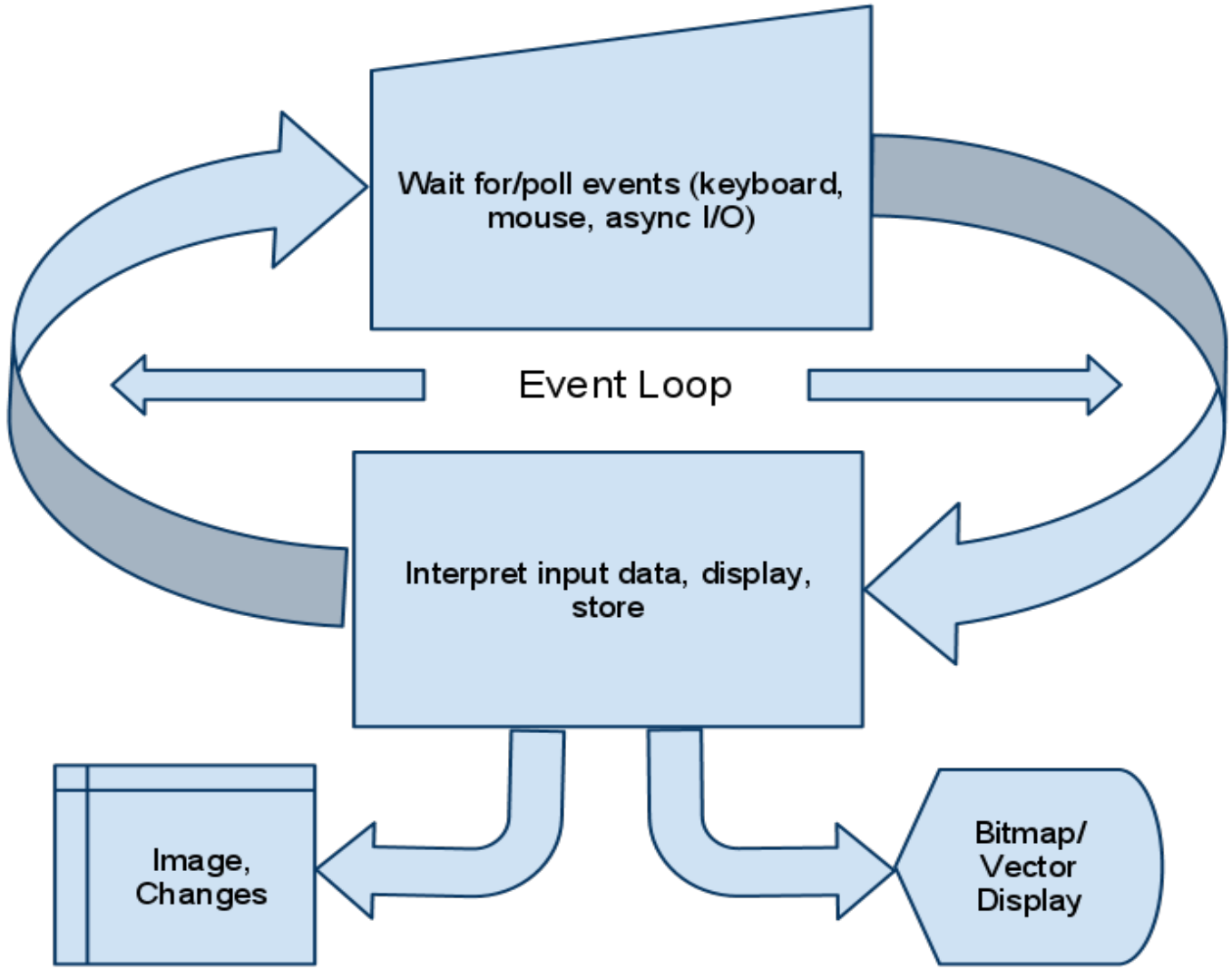
Changes File reflects history and evolution of the Image File

Sources File

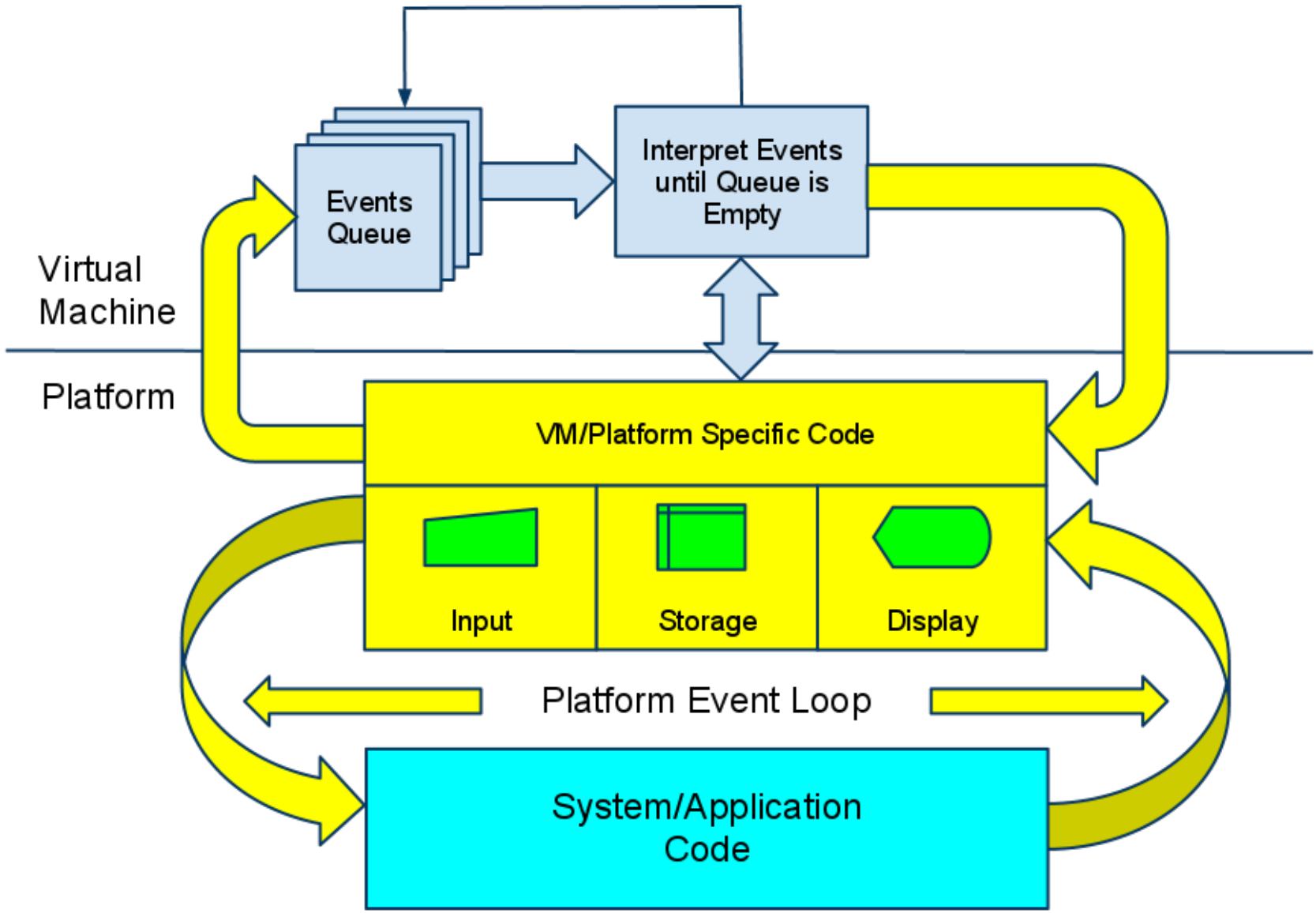
Sources File helps restore names in source code when decompiling

Icons by Bert Freudenberg

Traditional VM with Event Loop



Event-Driven VM



Squeak Event-Driven VM

- Implemented by Andreas Raab (2009)
- Platform provides an event queue (e. g. ring buffer)
- Platform pushes one or more events into the event queue and enters the Squeak bytecode interpreter
- The interpreter dispatches queued events to appropriate objects, thus driving execution of methods
- Once the event queue is empty, interpreter exits
- Absence of event loop makes embedding of such VM easier
- The Android port is an example of embedding the event-driven Squeak VM into the Dalvik VM stack

Source: vm-dev mailing list

Squeak on Android: Evolution of the Project

- The "Parent" project:

  http://code.google.com/p/squeak-android-vm/

  - Initial porting of the Event VM to Android
  - Maintained through 2010

- The "Current" project:

  http://code.google.com/p/squeakvm-tablet/

  - Mainly left the VM untouched
  - Fixed/implemented some facilities missing in the parent project

Current Status: Ready for Experimental Use

Improved integration with Android Input Methods (on-screen keyboard). Emulation of three-button mouse (colored clicks) using tablet's hardware buttons

Squeak image may now be loaded from the tablet's local filesystem. Entire image cannot be saved, but Squeak project loading and saving is supported.

Ad-hoc interface to Android TTS (Text to Speech Synthesis)

Conclusions and Future Work

- Derive an event-driven version of VM from a modern revision of Squeak VM (traditional) or Cog
- Improve the VM Platform Code to implement features still missing such as network sockets support
- Create a lighweight Squeak/Pharo/Cuis image file themed for mobile applications

Try It out

- Download and install the pre-packaged Squeak on Android Application
- Download and unzip the pre-packaged Squeak Image for Android
- Take the Test Drive
- Report Issues

Questions?