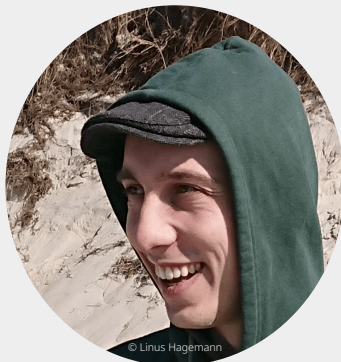


Live Programming and Designing of Dynamic Web Applications

Towards a totally RAD Development Experience

Linus Hagemann at FrOSCon 2023



Linus Hagemann





Robin Schreiber

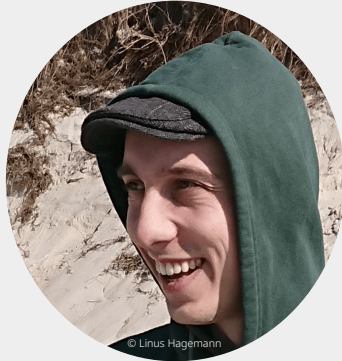


Linus Hagemann

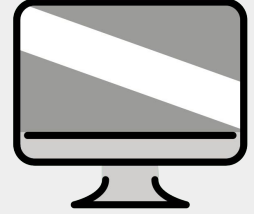
 /linusha


 @linushagemann@digitalcourage.social


 linushagemann.de

Linus Hagemann



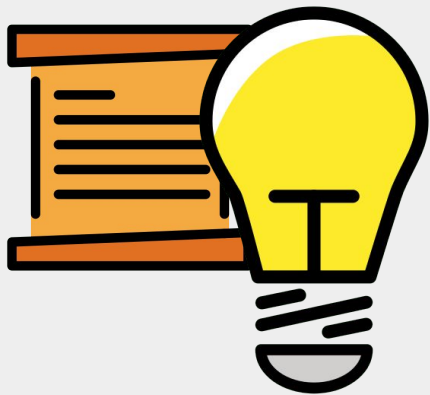
@linushagemann@digitalcourage.social



linushagemann.de



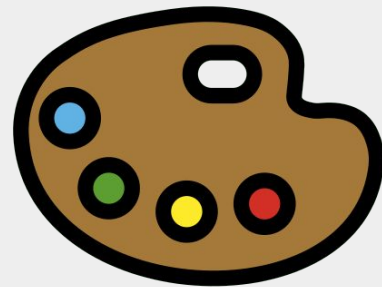
I



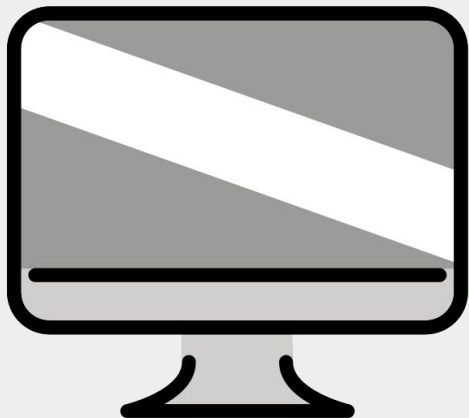
II

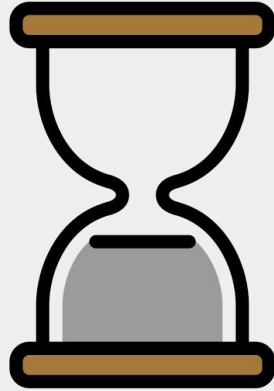


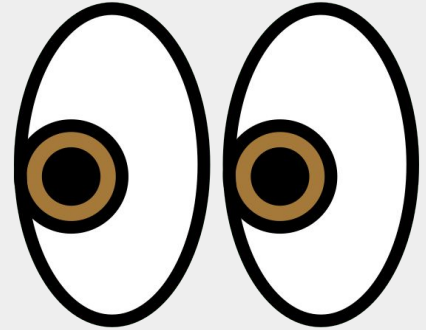
III

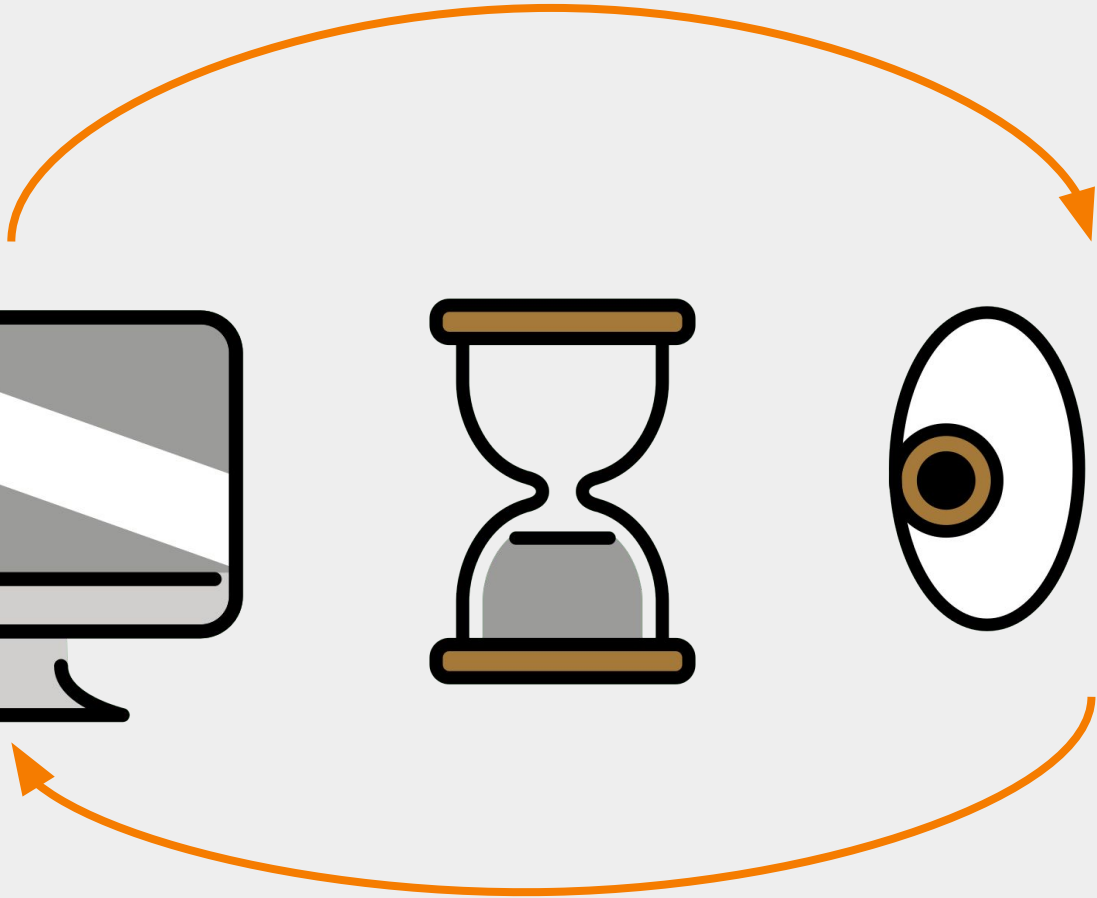
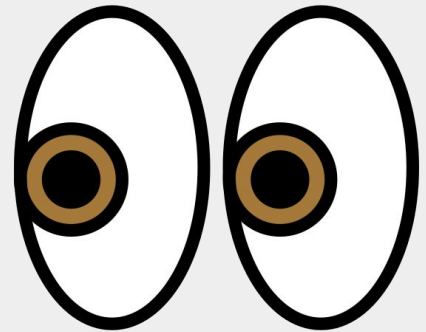
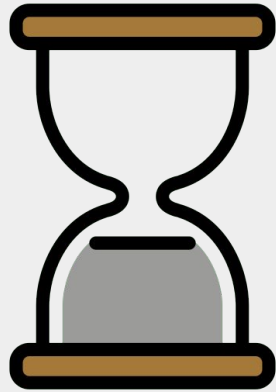
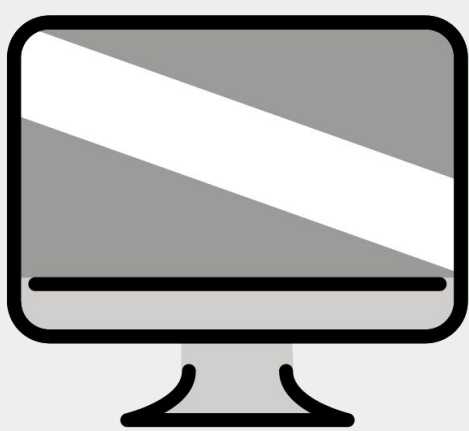


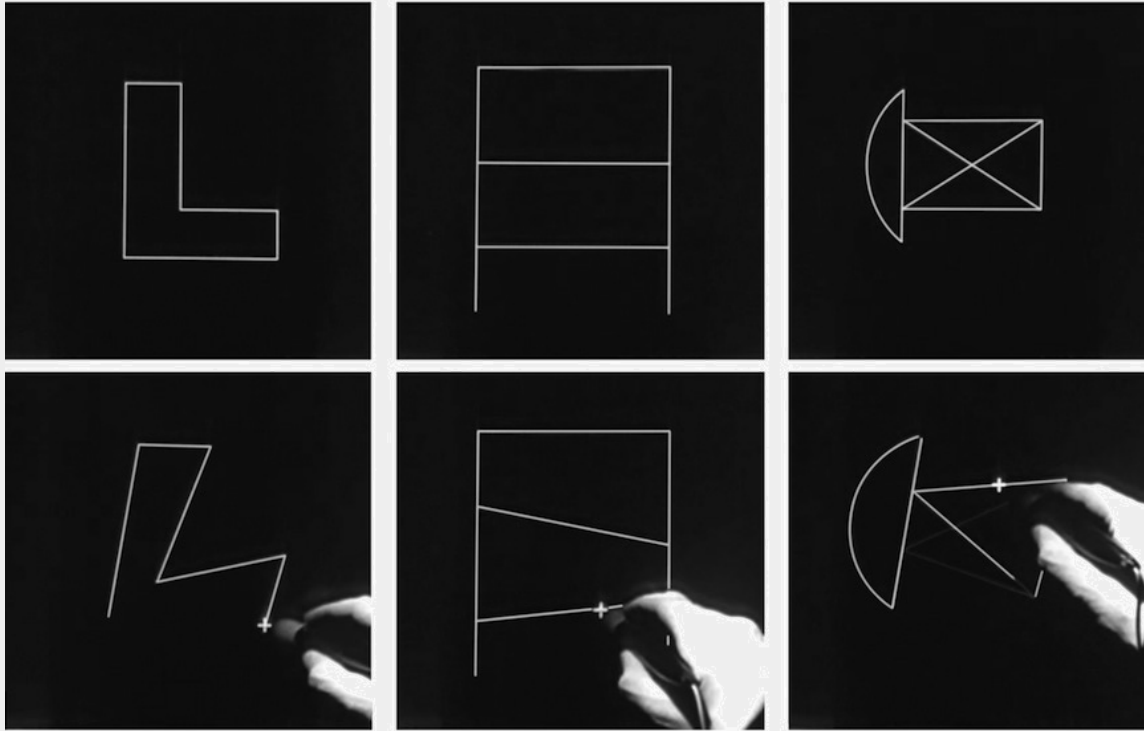
How do we write Software?











Sketchpad (1963) by Ivan Sutherland



direct manipulation

direct feedback

© 1964 MIT - Lincoln Laboratory, edited via rhizome.org



Sketchpad (1963)
by Ivan Sutherland



direct manipulation

direct feedback



Smalltalk (1972)

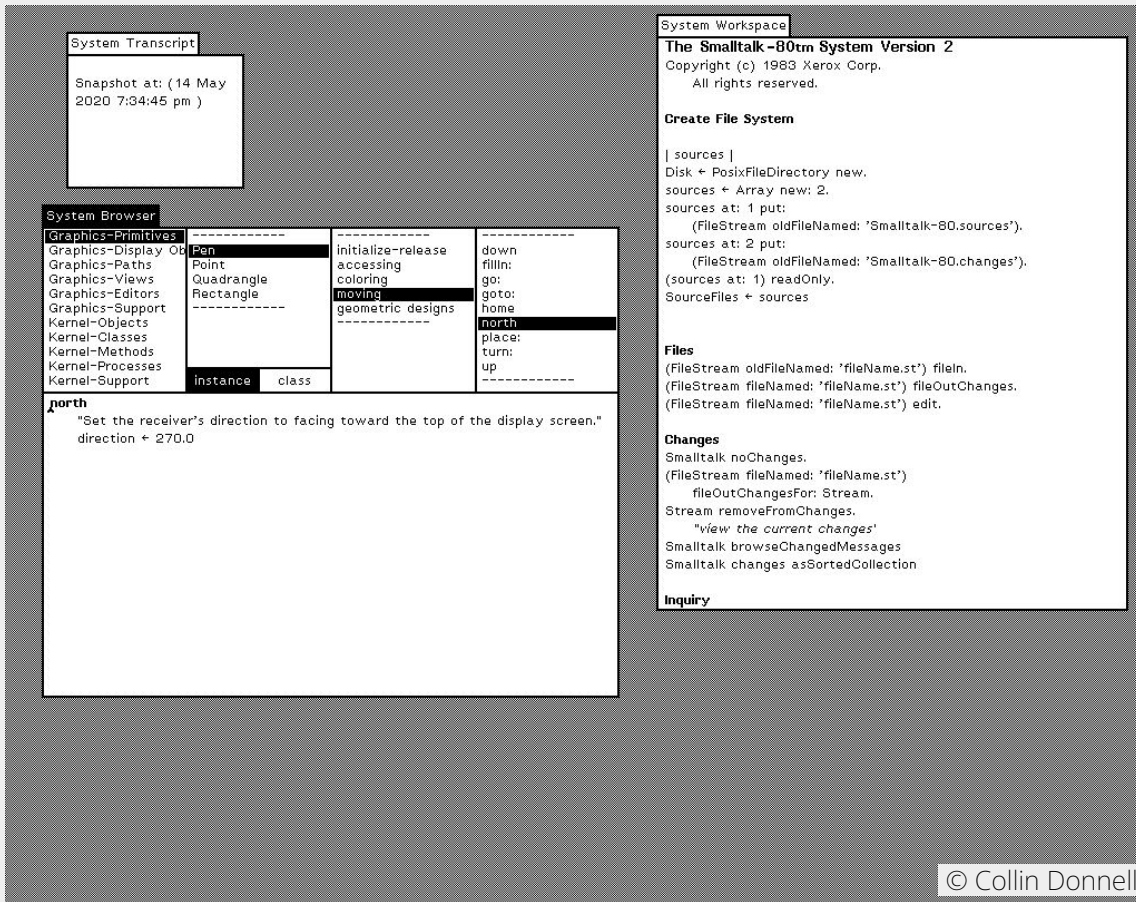
Xerox PARC

(OOP)

GUIs

self contained

fully live



The screenshot displays the Smalltalk-80 System Workspace and System Browser. The System Workspace window shows the title "The Smalltalk-80 System Version 2" and copyright information for Xerox Corp. It contains sections for "Create File System", "Files", "Changes", and "Inquiry". The System Browser window shows a hierarchical tree of classes and methods, with "north" selected. The "north" class is described as "Set the receiver's direction to facing toward the top of the display screen." with a "direction" slot value of 270.0.

System Transcript

Snapshot at: (14 May 2020 7:34:45 pm)

System Workspace

The Smalltalk-80 System Version 2
Copyright (c) 1983 Xerox Corp.
All rights reserved.

Create File System

| sources |
Disk + PosixFileDirectory new.
sources + Array new: 2.
sources at: 1 put:
 (FileStream oldFileNamed: 'Smalltalk-80.sources').
sources at: 2 put:
 (FileStream oldFileNamed: 'Smalltalk-80.changes').
(sources at: 1) readOnly.
SourceFiles + sources

Files

(FileStream oldFileNamed: 'fileName.st') fileIn.
(FileStream fileName: 'fileName.st') fileOutChanges.
(FileStream fileName: 'fileName.st') edit.

Changes

Smalltalk noChanges.
(FileStream fileName: 'fileName.st')
 fileOutChangesFor: Stream.
Stream removeFromChanges.
 "view the current changes"
Smalltalk browseChangedMessages
Smalltalk changes asSortedCollection

Inquiry

System Browser

Graphics-Primitives	Pen	initialize-release	down
Graphics-Display Obj	Point	accessing	fillin:
Graphics-Paths	Quadrangle	coloring	go:
Graphics-Views	Rectangle	moving	goto:
Graphics-Editors		geometric designs	home
Graphics-Support			north
Kernel-Objects			place:
Kernel-Classes			turn:
Kernel-Methods			up
Kernel-Processes			
Kernel-Support	instance	class	

north

"Set the receiver's direction to facing toward the top of the display screen."
direction ← 270.0

© Collin Donnell

Smalltalk (1972)

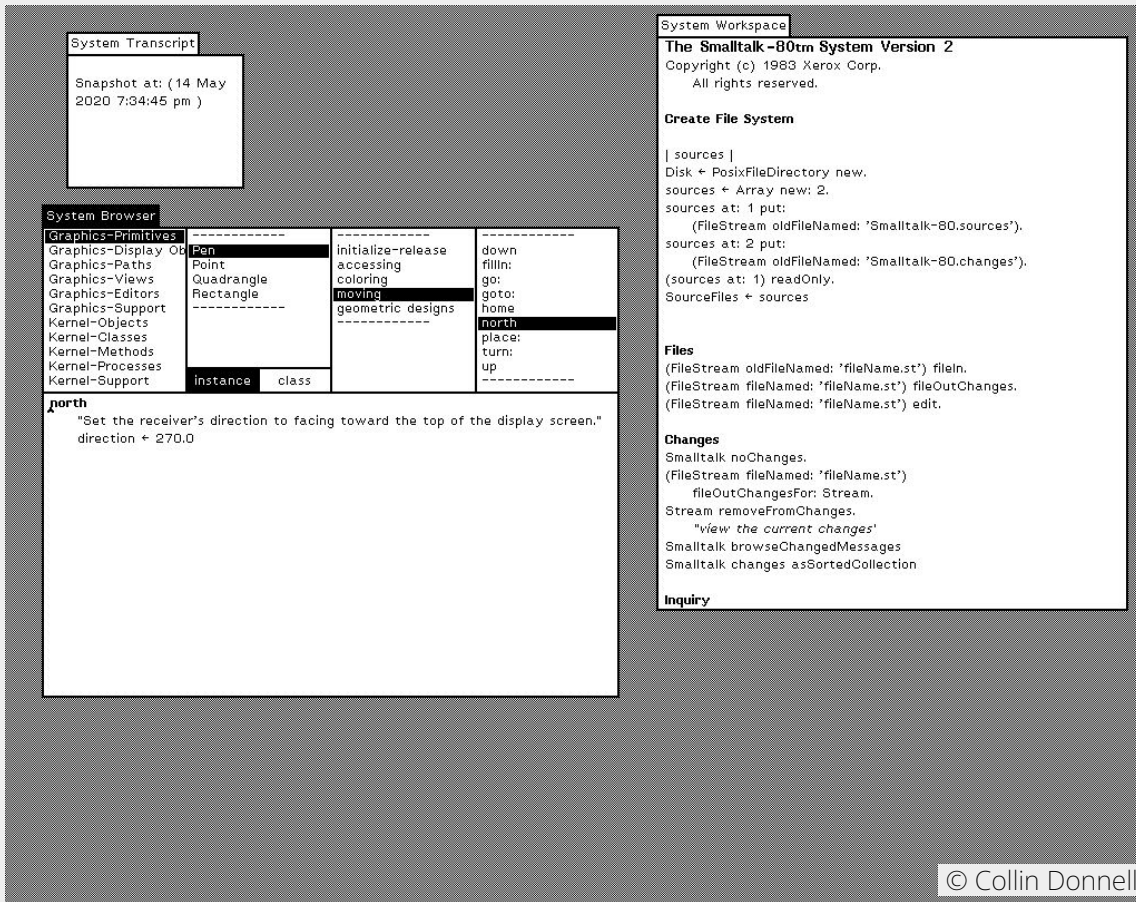
Xerox PARC

(OOP)

GUIs

self contained

fully live



System Transcript

Snapshot at: (14 May 2020 7:34:45 pm)

System Browser

Graphics-Primitives	Pen	initialize-release	down
Graphics-Display Ob	Point	accessing	fillin:
Graphics-Paths	Quadrangle	coloring	go:
Graphics-Views	Rectangle	moving	goto:
Graphics-Editors		geometric designs	home
Graphics-Support			north
Kernel-Objects			place:
Kernel-Classes			turn:
Kernel-Methods			up
Kernel-Processes			
Kernel-Support	instance	class	

north

"Set the receiver's direction to facing toward the top of the display screen."
direction ← 270.0

System Workspace

The Smalltalk-80m System Version 2
Copyright (c) 1983 Xerox Corp.
All rights reserved.

Create File System

| sources |
Disk ← PosixFileDirectory new.
sources ← Array new: 2.
sources at: 1 put:
 (FileStream oldFileName: 'Smalltalk-80.sources').
sources at: 2 put:
 (FileStream oldFileName: 'Smalltalk-80.changes').
(sources at: 1) readOnly.
SourceFiles ← sources

Files

(FileStream oldFileName: 'fileName.st') fileIn.
(FileStream fileName: 'fileName.st') fileOutChanges.
(FileStream fileName: 'fileName.st') edit.

Changes

Smalltalk noChanges.
(FileStream fileName: 'fileName.st')
 fileOutChangesFor: Stream.
Stream removeFromChanges.
 "view the current changes"
Smalltalk browseChangedMessages
Smalltalk changes asSortedCollection

Inquiry

© Collin Donnell

Squeak/Smalltalk

The screenshot displays the Squeak/Smalltalk IDE interface with several windows and objects:

- Object Inspector (top left):** Shows the object `a PluggableListMorphPlus(1713338)` with variables `destForm`, `index`, `maxX`, `minX`, and `minX`. A `browse` button is highlighted.
- Object Inspector (top right):** Shows the object `a GraphMorph<Graph>(3320986)` with variables `data`, `dataColor`, `cursor`, `cursorColor`, `cursorColorAtZeroCrossings`, and `startIndex`. A `self cursor: 150` message is shown.
- System Browser (bottom right):** Shows the class `GradientFillStyle` under the `Balloon-Fills` category. It lists methods like `addFillStyle`, `addNewColor`, `asColor`, `beLinearGradient`, `beRadialGradient`, `changeColor`, and `changeFillColor`.
- Workspace (bottom center):** Contains the code `myMorph := Morph new.` and `myMorph color Color blue.`
- Transcript (middle left):** Shows the message `Hello, Squeak!` and a timestamp `{18 August 2016. 12:30:19 pm}`.
- Workspace (bottom left):** Shows the result of the transcript: `Transcript shown: 'Hello, Squeak!'.` and the calculation `3+4 7`.
- Visual Objects:** A central workspace contains a yellow circle, an orange triangle, a purple rectangle, and a cyan star. A yellow circle is highlighted with a black arrow.
- Graph (top right):** A small window showing a sine wave graph with a red vertical line at the zero-crossing.

A gradient fill style is a fill which interpolates smoothly between

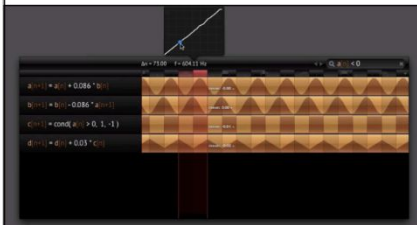
Marcel Taeumel, CC BY 4.0 via Wikimedia Commons

> Creators need an immediate connection to what they create... So much of creation is discovery, and you can't discover anything if you can't see what you are doing.

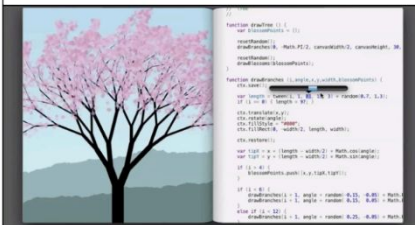
 **Bret Victor**
 Inventing on Principle

SOFTWARE-BASED TOOLS ARE TRAPPED IN TINY RECTANGLES.

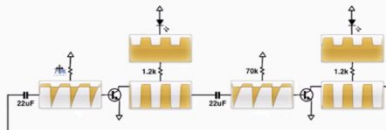
FOR YEARS, I'VE BEEN DESIGNING TOOLS.



TOOLS FOR PEOPLE MAKING SOFTWARE...



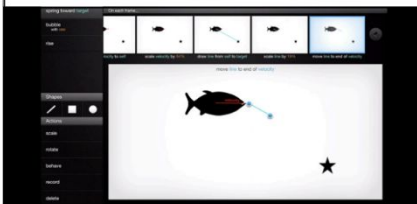
... ELECTRONICS, MUSIC, ANIMATION,
MATHEMATICAL SYSTEMS...



THINGS WITH **COMPLEX BEHAVIOR**.



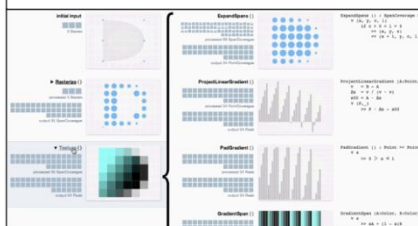
MY FOCUS HAS ALWAYS BEEN —
HOW CAN CREATORS **SEE** THAT BEHAVIOR?



HOW CAN THEY SEE WHAT THE THING
THEY'RE BUILDING IS **ACTUALLY DOING**?



AND WHAT ARE **POWERFUL WAYS OF SEEING**



SO THEY CAN **UNDERSTAND** WHAT IT'S DOING?



THESE HAVE GENERALLY BEEN SOFTWARE-BASED TOOLS.

WHAT "SOFTWARE-BASED" MEANS TODAY IS THAT THESE TOOLS ARE TRAPPED INSIDE A **TINY RECTANGLE** THAT SITS ON YOUR DESK.



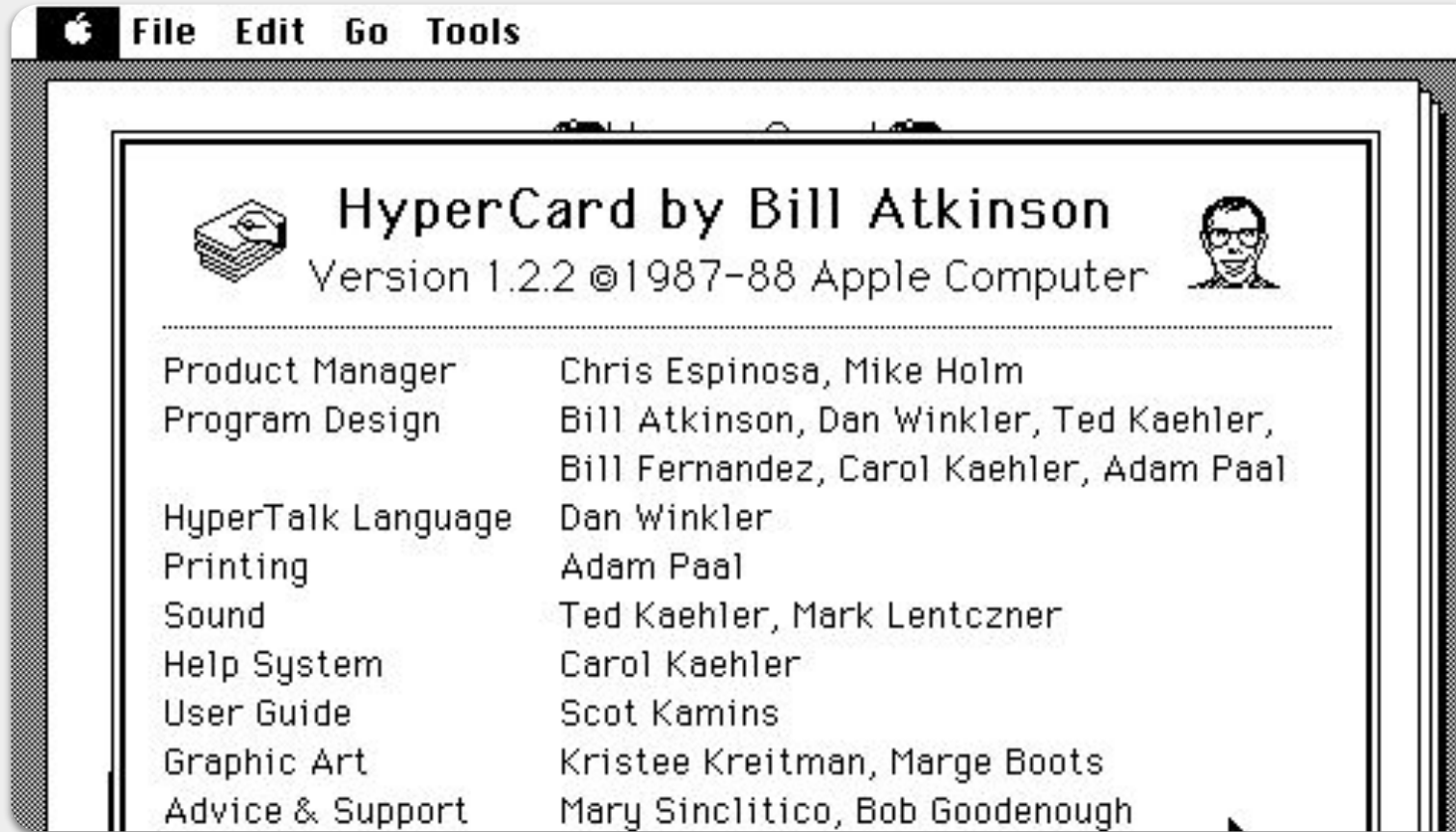
TO DO YOUR WORK, YOU SIT AT YOUR DESK
AND YOU STARE AT THIS **TINY RECTANGLE**.

AND THIS FRUSTRATES ME.



Dynamicland

HyperCard



© <https://winworldpc.com>

The image shows the Scratch programming environment. At the top, there is a blue header with the Scratch logo, a globe icon, and menu options: File, Edit, Tutorials, Join Scratch, and Sign in. Below the header, there are tabs for Code, Costumes, and Sounds. The left sidebar contains a vertical menu with colored circles representing different block categories: Motion (blue), Looks (purple), Sound (pink), Events (yellow), Control (orange), Sensing (teal), Operators (green), Variables (orange), and Cycles (red). The main workspace is a grid with a small Scratch cat sprite in the top right corner. The right side of the workspace is a larger stage area with a larger Scratch cat sprite in the center. At the bottom, there is a control panel for the selected sprite, showing its name (Sprite1), x and y coordinates (0, 0), size (100), and direction (90). There are also buttons for play, stop, and a list of sprites.

Scratch

Scratch

nev123456789nev, CC BY-SA 4.0, via Wikimedia Commons

Scratch

File Edit Tutorials Join Scratch Sign in

Code Costumes Sounds

Motion

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer

Scratch

Sprite Sprite1 x 0 y 0

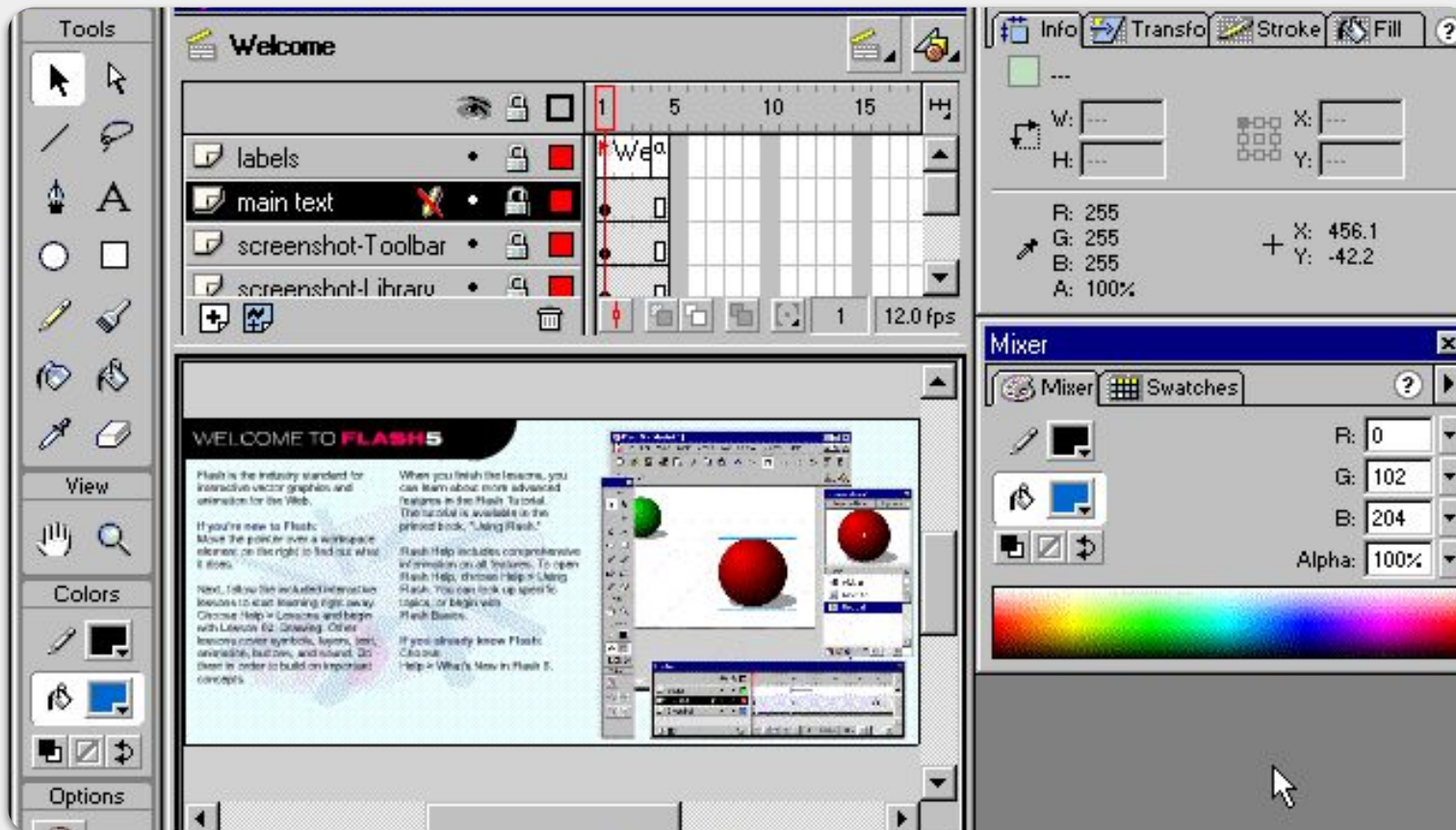
Size 100 Direction 90

Stage

Backdrops 1

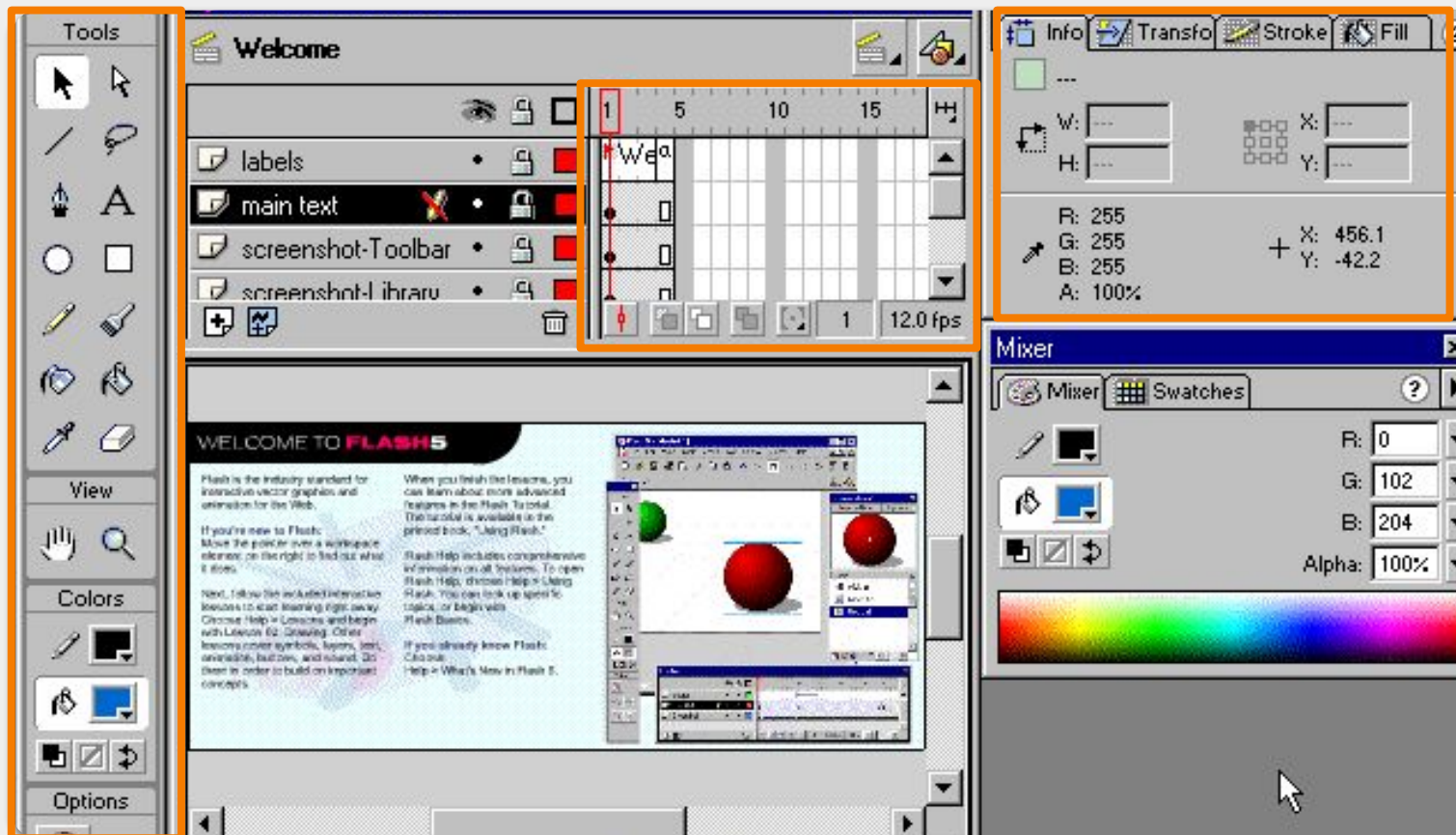
nev123456789nev, CC BY-SA 4.0, via Wikimedia Commons

Flash 

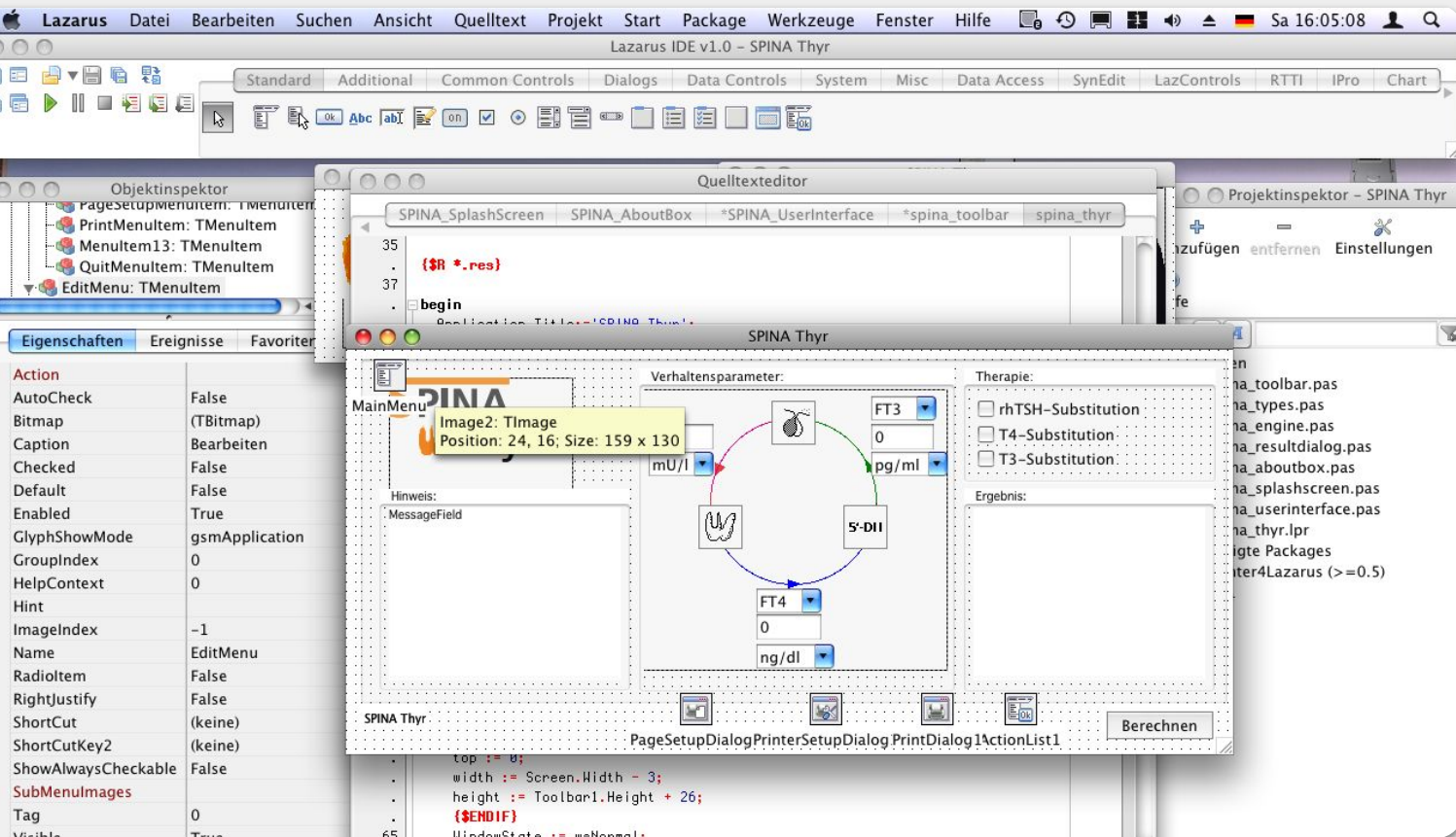



© <https://winworldpc.com>

Flash

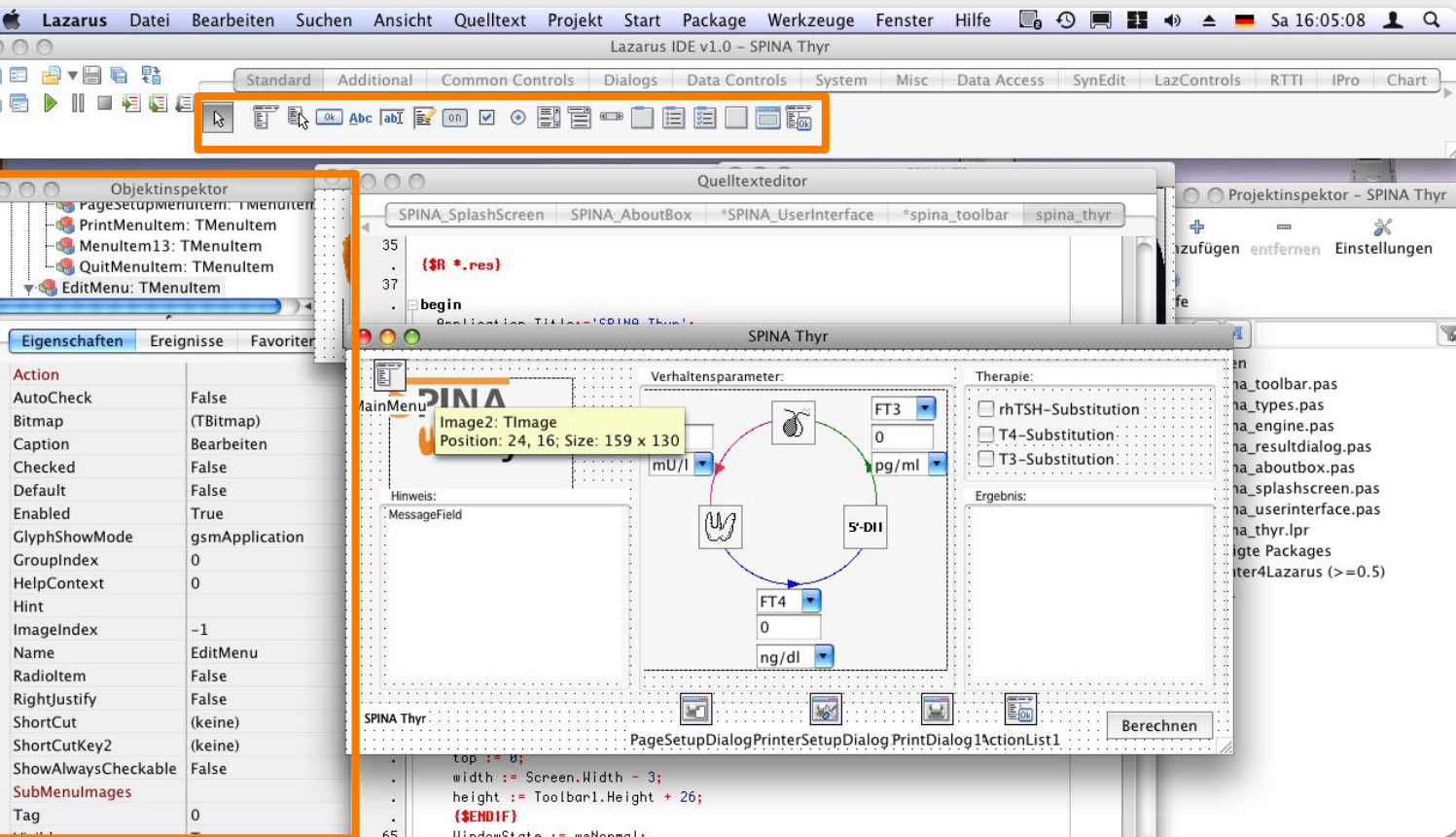



© <https://winworldpc.com>



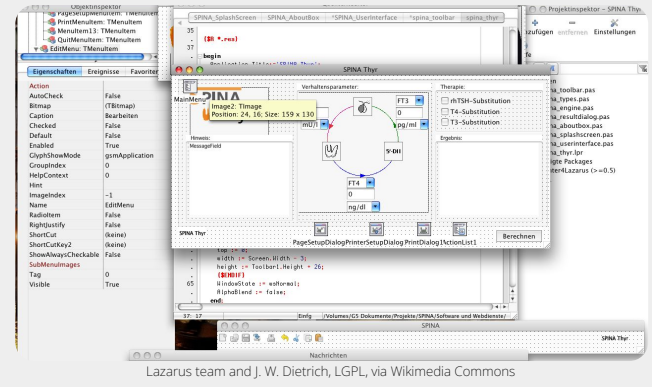
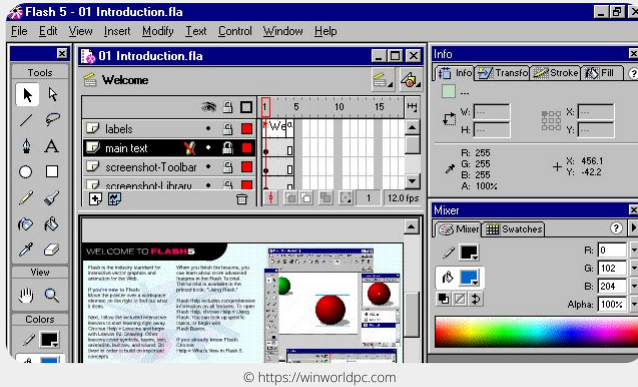
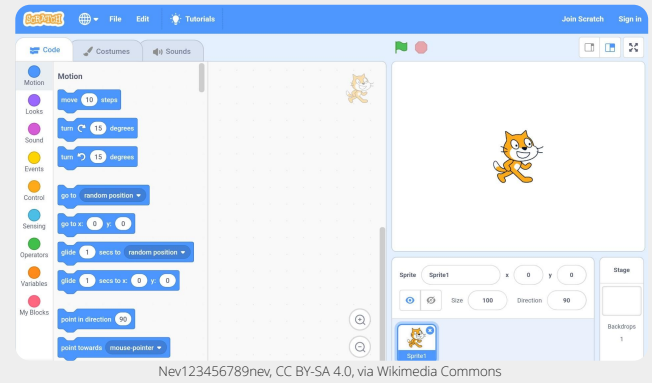
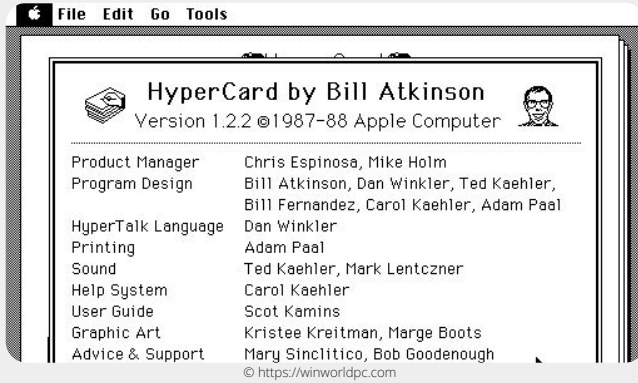
Lazarus 
(Delphi)

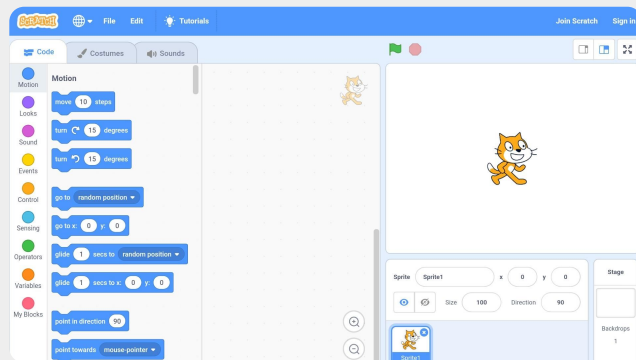
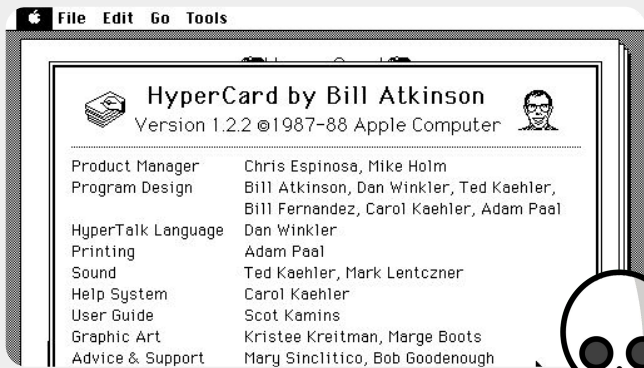
Lazarus team and J. W. Dietrich, LGPL, via Wikimedia Commons



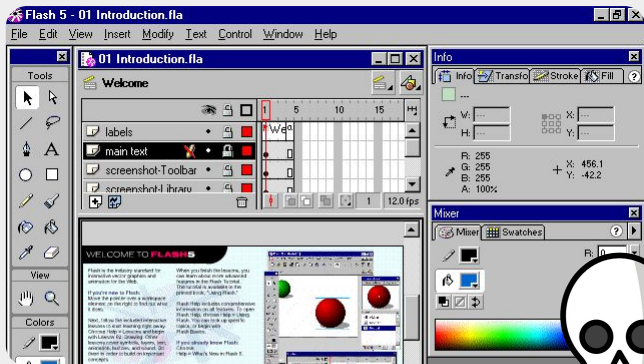
Lazarus 
(Delphi)

Lazarus team and J. W. Dietrich, LGPL, via Wikimedia Commons

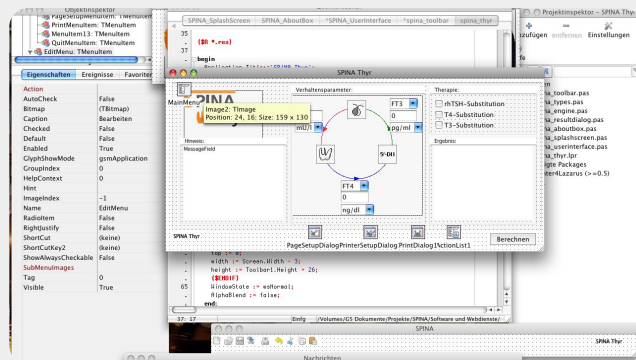




Nev123456789nev, CC BY-SA 4.0, via Wikimedia Commons



https://winworldpc.com



Lazarus team and J. W. Dietrich, LGPL, via Wikimedia Commons

HyperCard by Bill Atkinson
Version 1.2.2 ©1987-88 Apple Computer

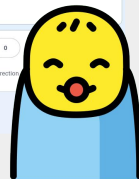
Product Manager: Chris Espinosa, Mike Holm
 Program Design: Bill Atkinson, Dan Winkler, Ted Kæhler, Bill Fernandez, Carol Kæhler, Adam Paal
 HyperTalk Language: Dan Winkler
 Printing: Adam Paal
 Sound: Ted Kæhler, Mark Lentczner
 Help System: Carol Kæhler
 User Guide: Scot Kamins
 Graphic Art: Kristee Kreitman, Marge Boots
 Advice & Support: Mary Sinclitico, Bob Goodenough

© <https://winworldpc.com>



Scratch interface showing a cat sprite on a stage. The left sidebar contains the 'Motion' tab with various code blocks like 'move 10 steps', 'turn 15 degrees', 'go to random position', and 'point towards mouse pointer'. The bottom right shows the 'Sprite' panel with a cat sprite selected.

Nev123456789nev, CC BY-SA 4.0, via Wikimedia Commons



Flash 5 - 01 Introduction fla

File Edit View Insert Modify Text Control Window Help

Tools: Welcome, labels, main text, screenshot-Toolbar, screenshot-Header

Info: W: 255, H: 255, X: 456.1, Y: -42.2

Mixer: Swatches

WELCOME TO FLASH 5



© <https://winworldpc.com>

Project Inspector - SPINA Thyr

Verhaltensparameter: SPINA_AboutBox, SPINA_UserInterface, spina_toolbar, spina_thyr

Therapie: mTSH-Substitution, T4-Substitution, T3-Substitution

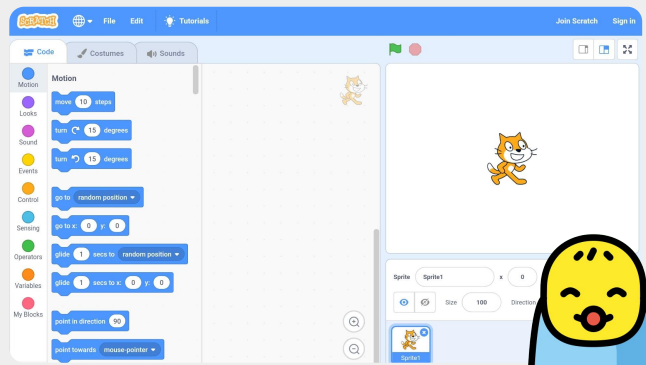
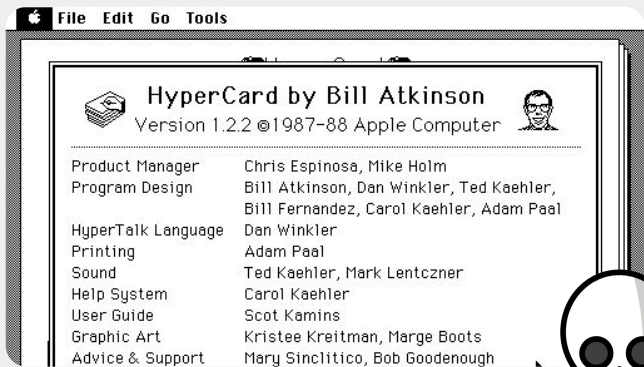
SPINA Thyr: Image2: Yellow, Position: 24, 16, Size: 159 x 130

Code:

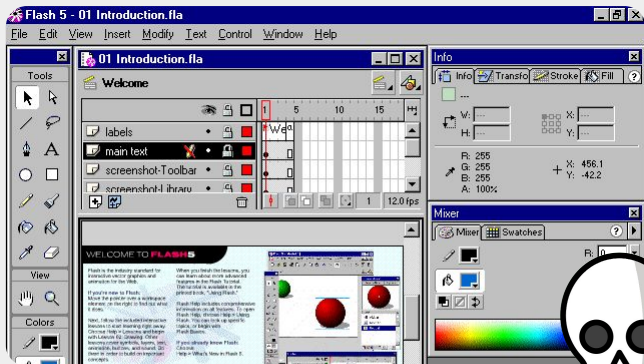
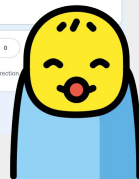
```

    goto = Screen.W/2 - 2;
    goto = TextStart; goto = 24;
    [Bilder]
    #labelindex := 1;
    #labelindex := 1;
  
```

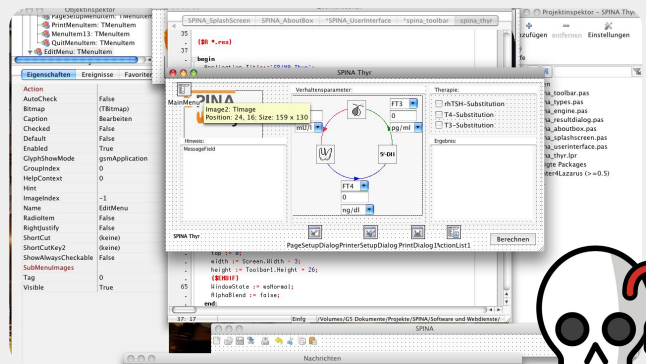
Lazarus team and J. W. Dietrich, LGPL, via Wikimedia Commons



Nev123456789nev, CC BY-SA 4.0, via Wikimedia Commons




https://winworldpc.com



Lazarus team and J. W. Dietrich, LGPL, via Wikimedia Commons





 Dan Ingalls

Lively Kernel (2006)



File Edit View History Bookmarks Window Help
http://research.sun.com/projects/lively/index.xhtml
Apple Amazon eBay Yahoo! News (231) ▾

A Lively Engine
Sun 3D Logo
Score: 0
Stock Widget
BigCharts
A service of MarketWatch
DOW JONES
NASDAQ
NYSE
S&P INX

JavaScript Code Browser

- initialize
- makeNewFace
- reshape
- startSteppingScripts
- setHands**

```
ClockMorph.prototype.setHands = function ()  
{  
  var currentDate = new Date();  
  var center = this.shape.bounds().center();  
  var second = currentDate.getSeconds();  
  var minute = currentDate.getMinutes()+second/60;  
  var hour = currentDate.getHours()+minute/60;  
  this.getNamedMorph("hours").setRotation(hour/24*360);  
}
```

- self contained
- fully live
- direct manipulation driven

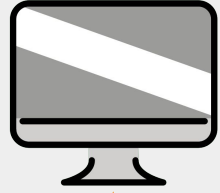


DEMO

TIME

- morphic
- live evaluation & inspection
- all time is runtime

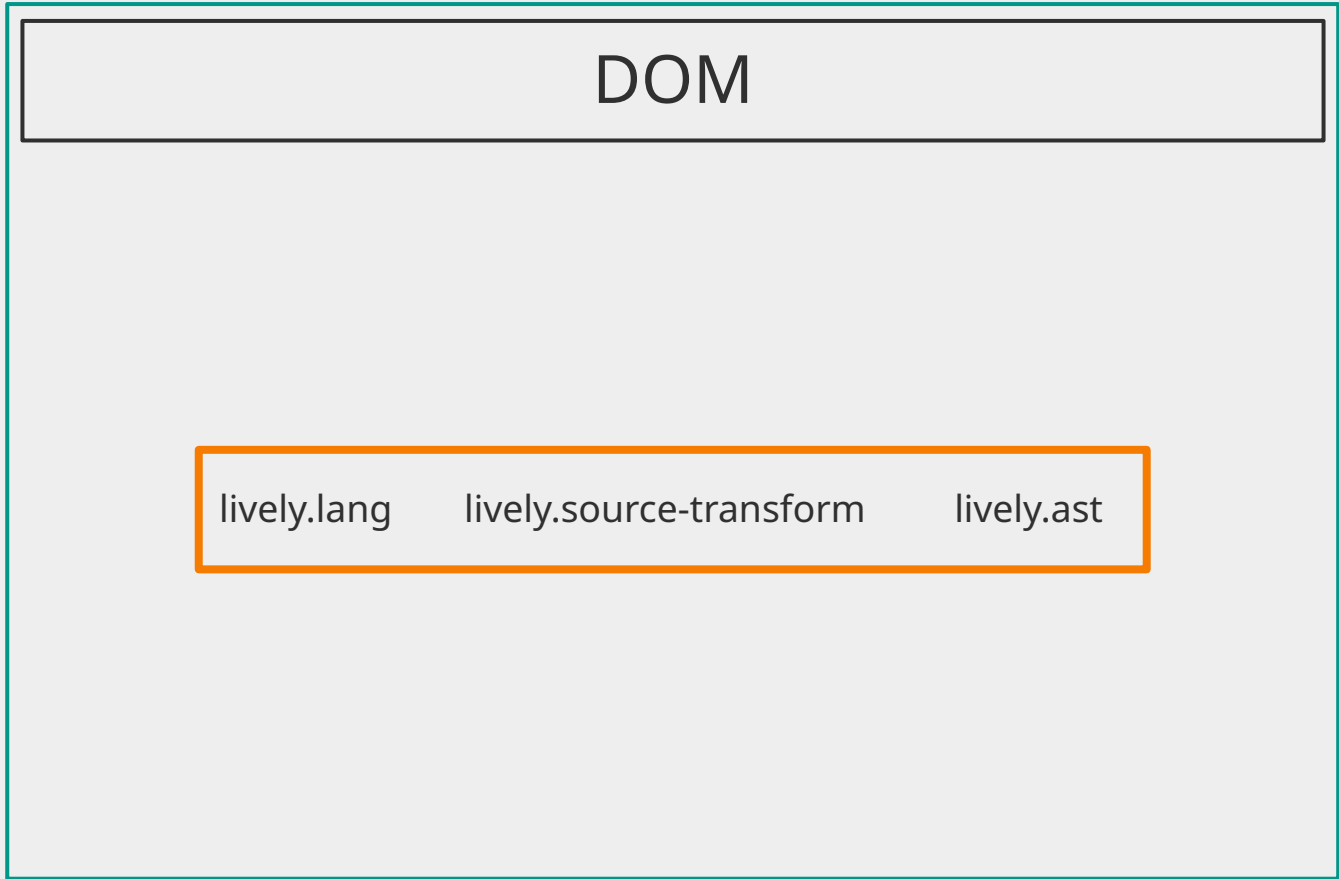
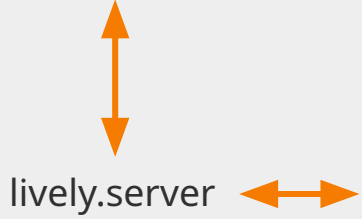
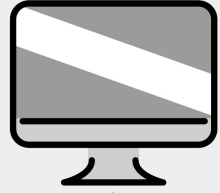
How does that work?



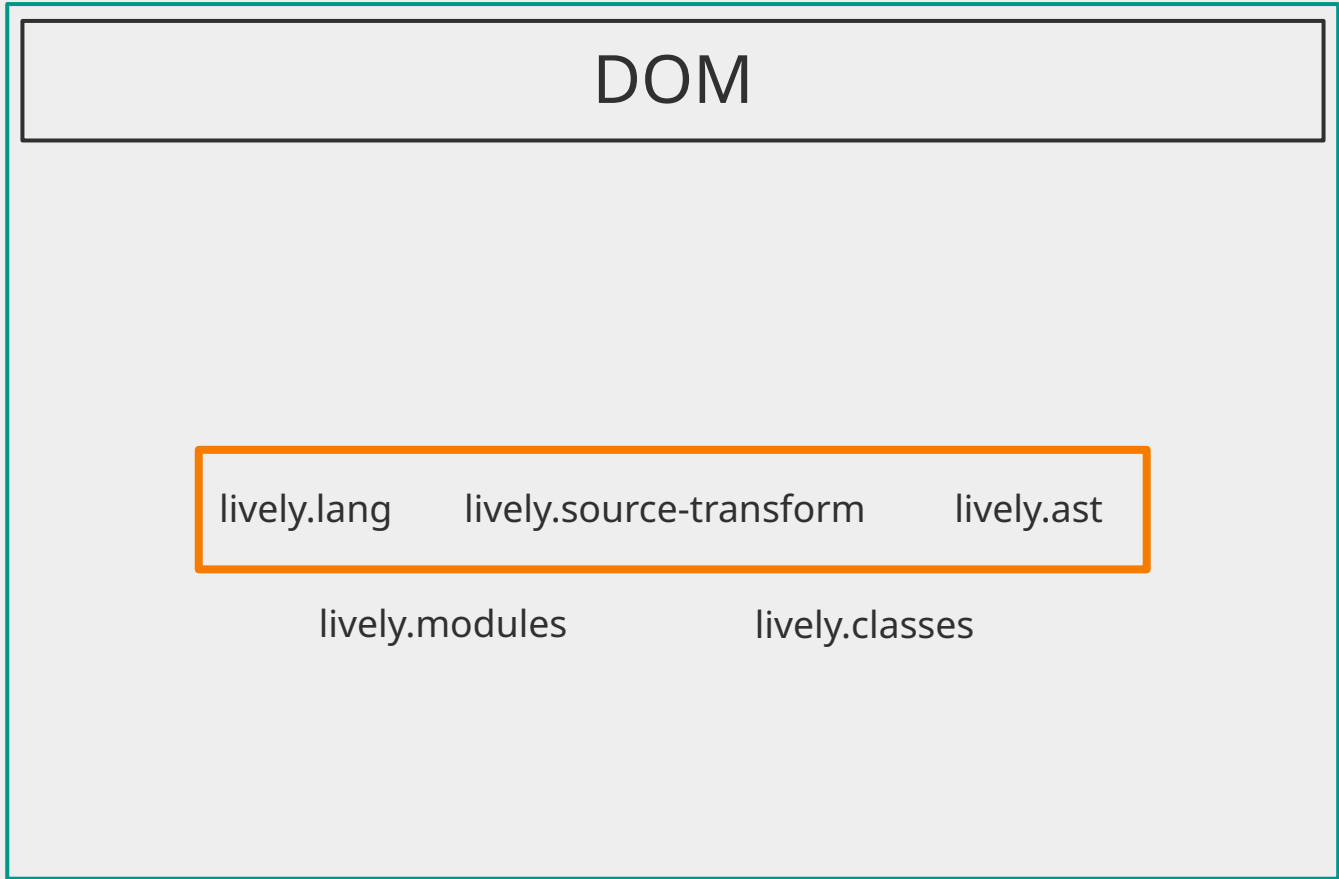
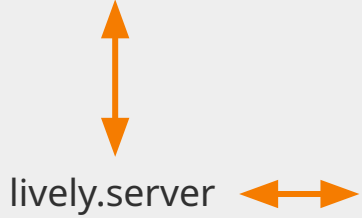
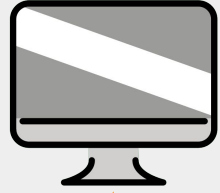
lively.server



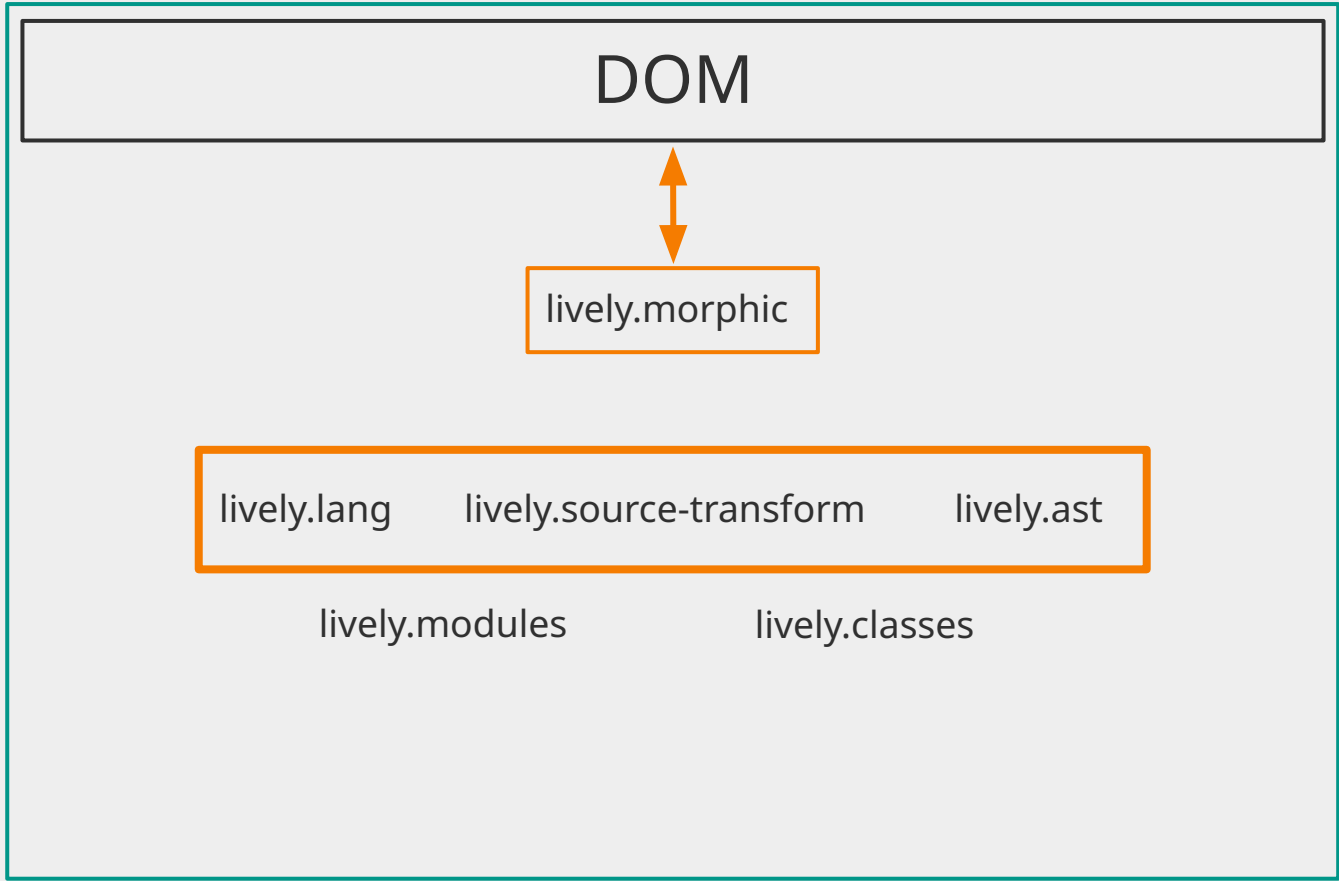
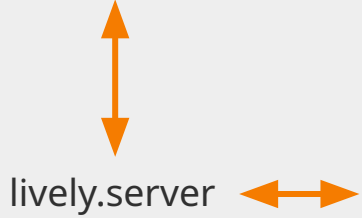
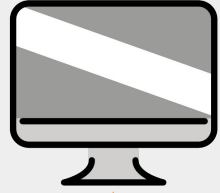
Browser



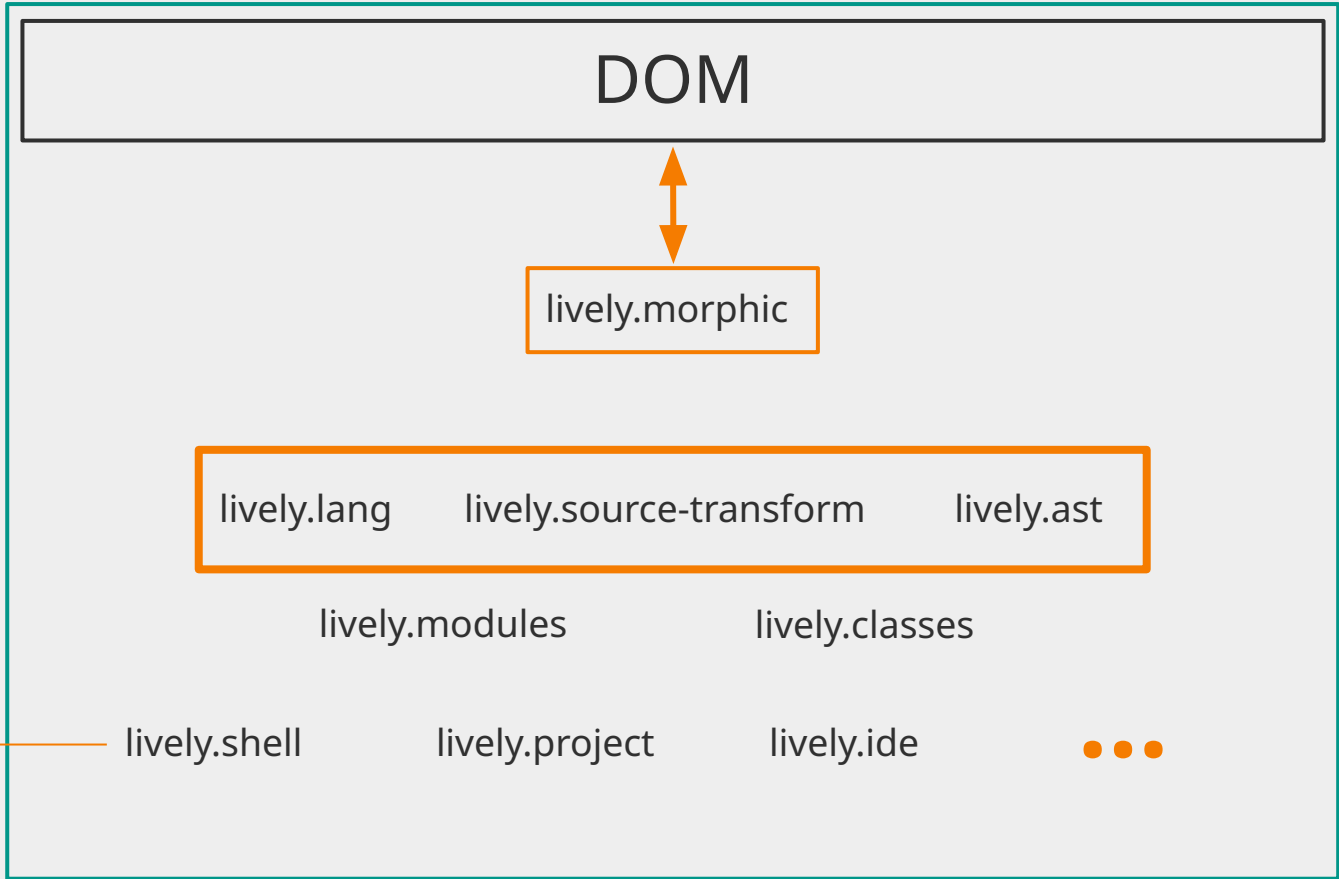
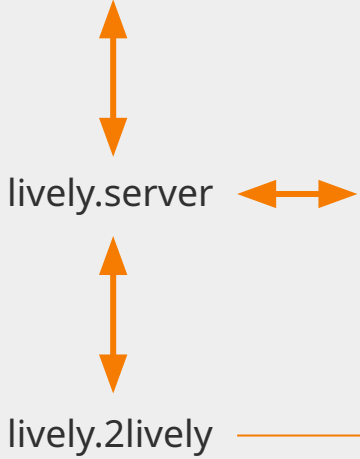
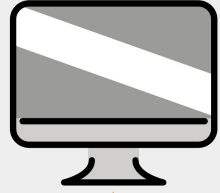
Browser



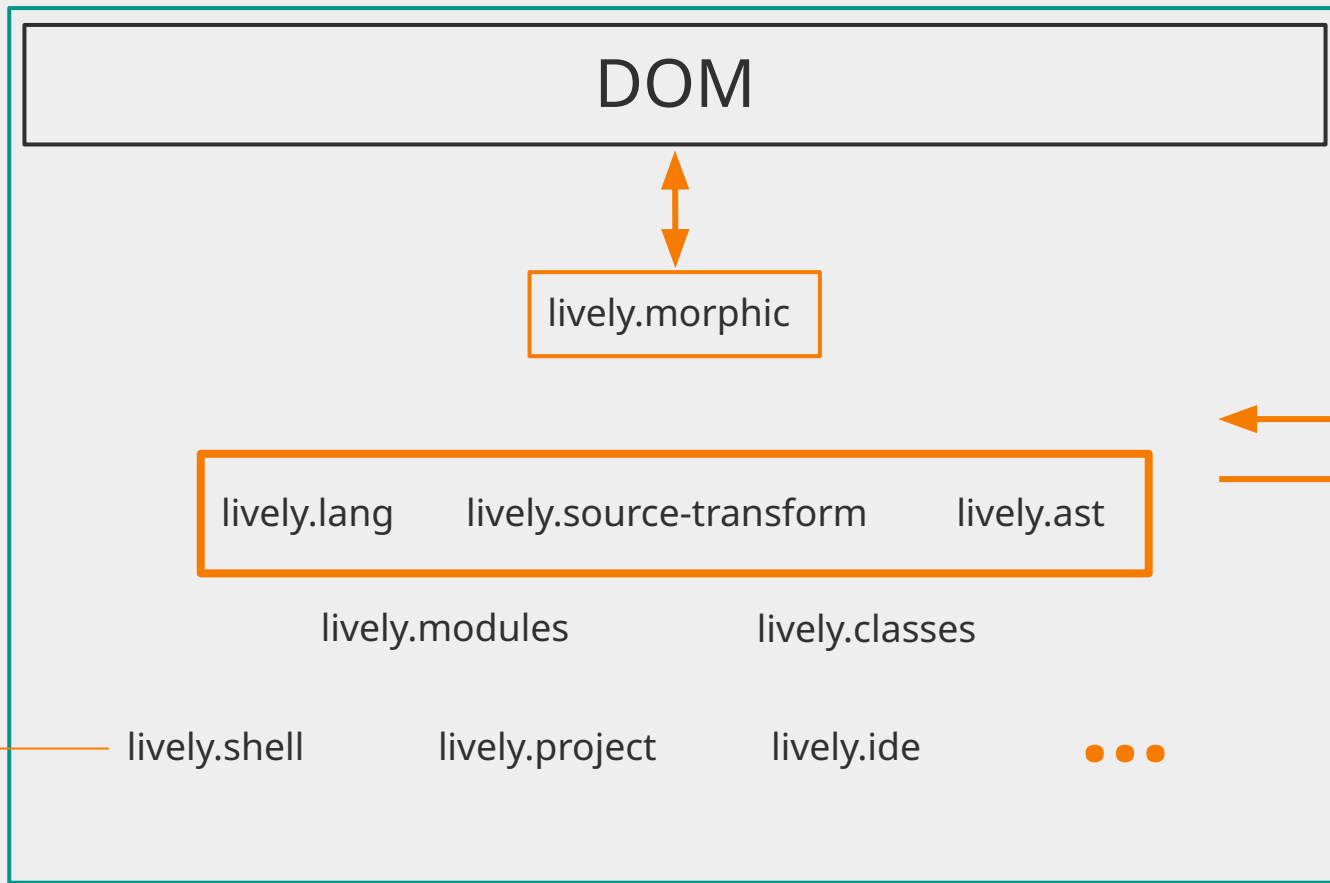
Browser



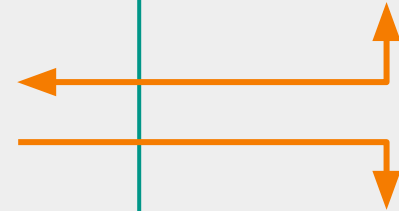
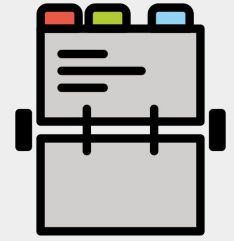
Browser



Browser

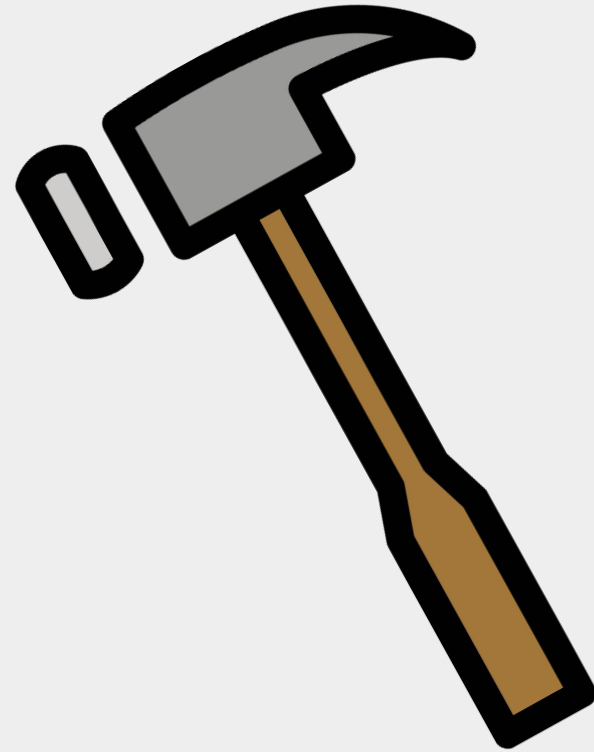


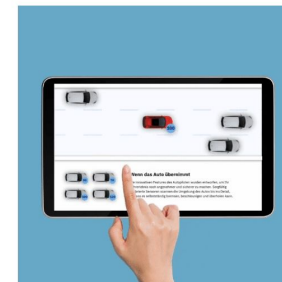
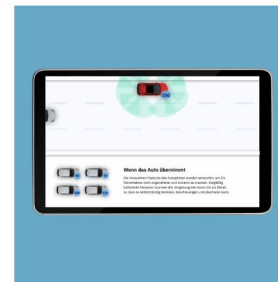
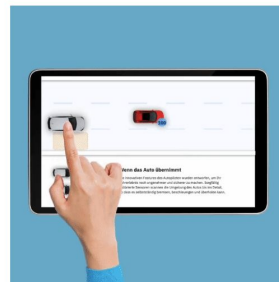
lively.serializer



lively.freezer

Browser





← ZURÜCK

WENN DAS AUTO ÜBERNIMMT

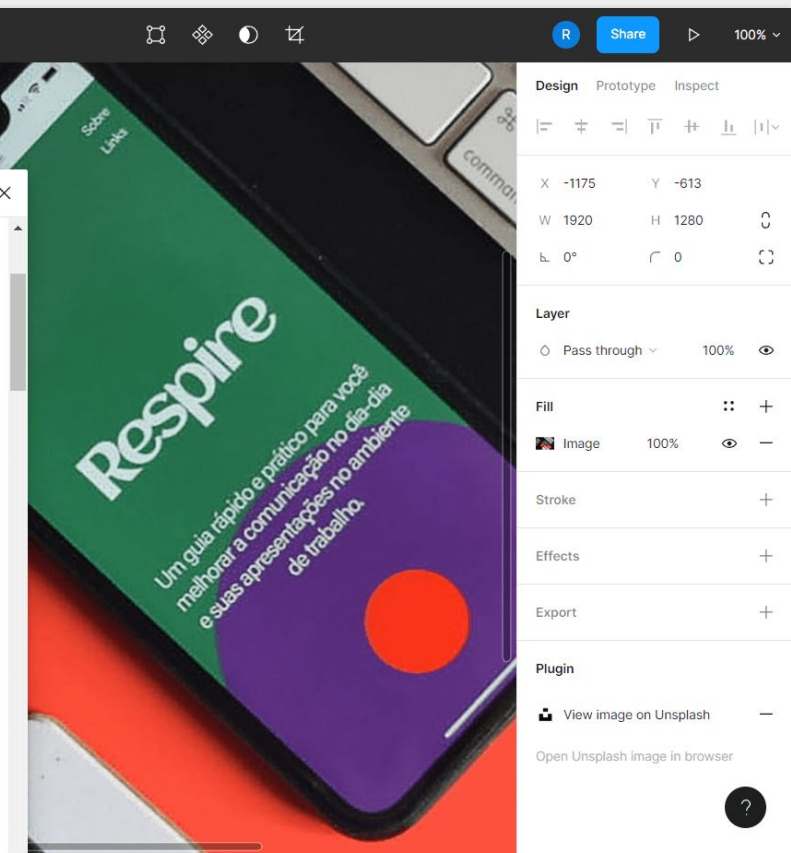
Autonomes Fahren wird durch technologische Neuerungen immer interessanter für Autohersteller und natürlich auch ihre Kunden. Dabei geht es stets auch um die Frage, wie der Autopilot in verschiedenen Situationen reagiert. In dieser kleinen Demo simulieren wir, wie sich ein autonom gesteuertes Auto verhält, wenn es auf langsamere oder schnellere Fahrzeuge trifft, wann es überholt und bremst.

CATEGORY
Simulation

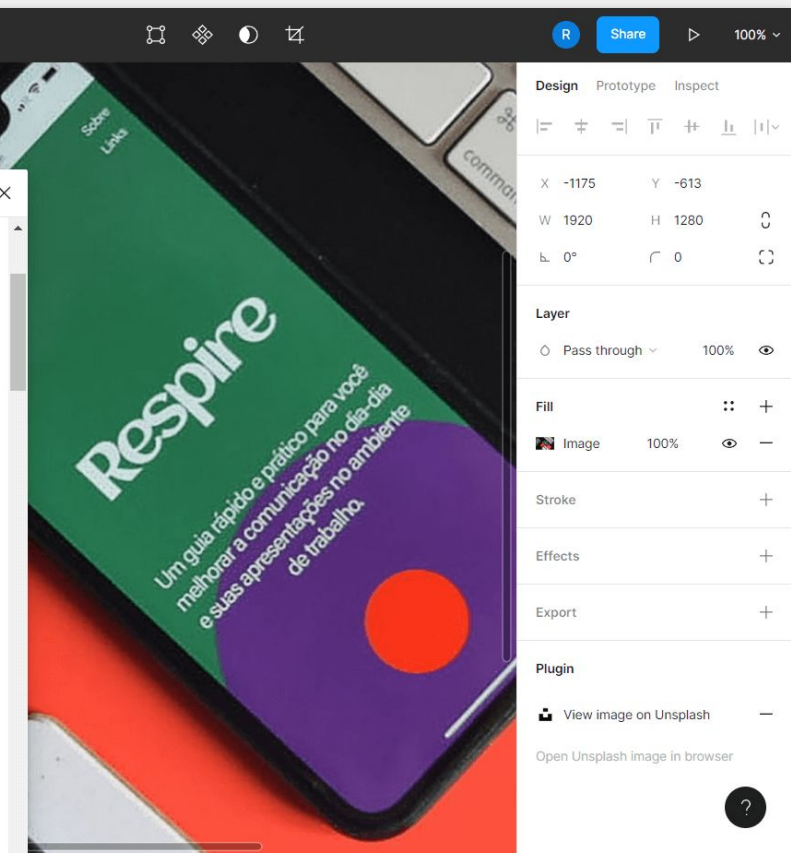
DATE
08.2020

[Ausprobieren](#)

© typeshift



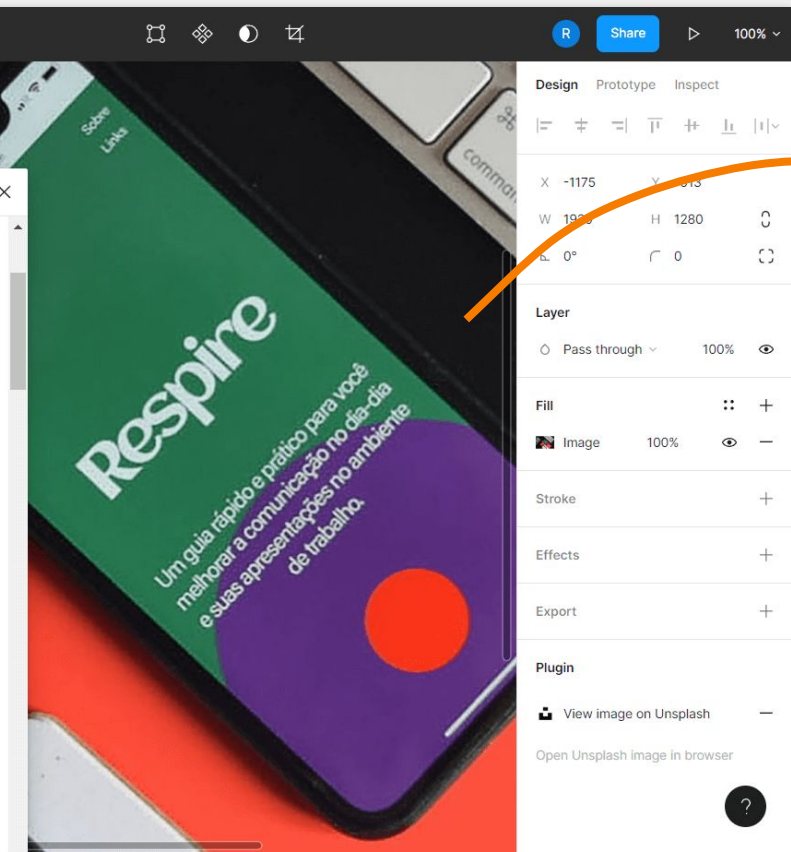
© <https://www.toushenne.de>



© <https://www.toushenne.de>

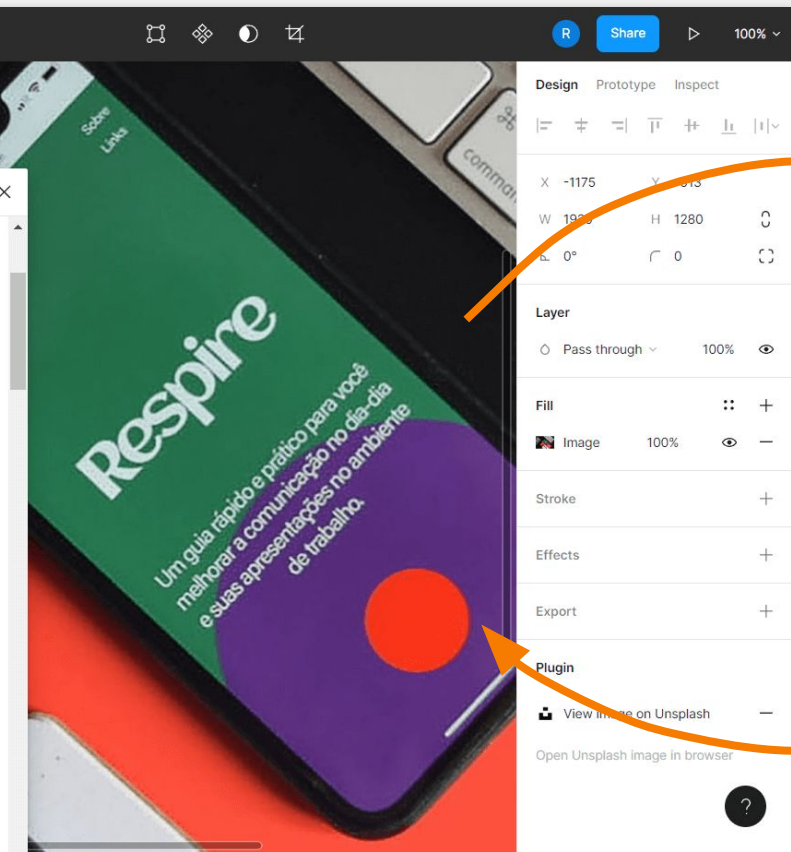


© <https://www.pngfind.com/>



© <https://www.pngfind.com/>

© <https://www.toushenne.de>



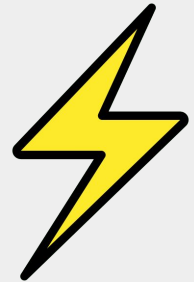
© <https://www.toushenne.de>



© <https://www.pngfind.com/>

Rapid Application Development

Rapid Application Development



1. Workflows for Designers and Coders with the same Artifact
2. Collaboration (git)

1. Workflows for Designers and Coders with the same Artifact
2. Collaboration (git)

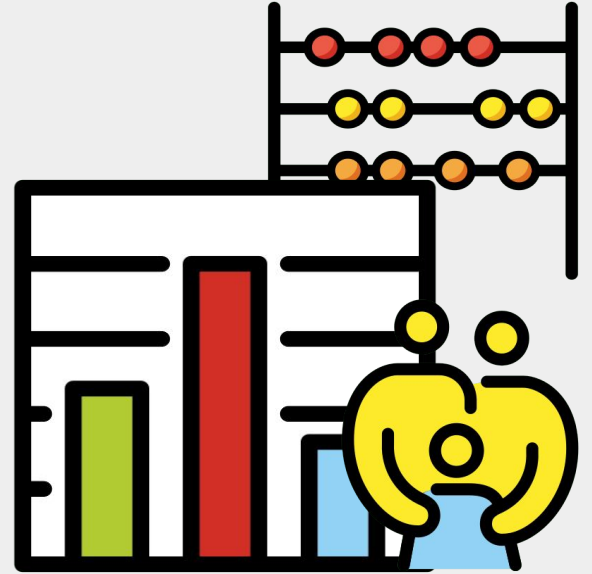
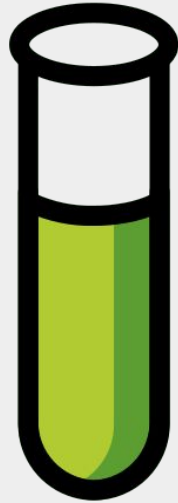
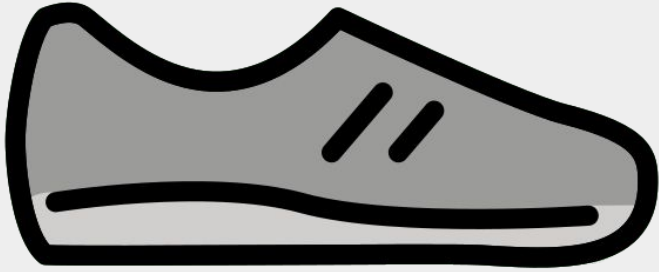
- generate declarative component definitions from direct manipulation operations
- provide means to attach behavior

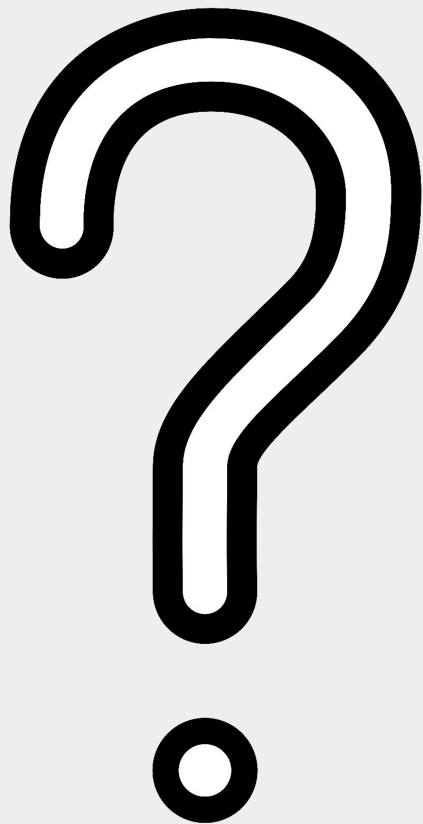
DEMO

TIME

- reconciliation
- components & parts
- ViewModels & bindings

What next?





[/LivelyKernel/lively.next](#)



[#lively.next:matrix.org](#)



[lively-next.org](#)



Rights for used images lie with the attributed sources.

All emojis designed by [OpenMoji](#) – the open-source emoji and icon project. License: [CC BY-SA 4.0](#).

Otherwise this slide deck is licensed under CC BY 4.0. © Linus Hagemann, 2023.