

BLOC5

An artful p5.js-inspired Bloc API for learning to code and making art with Pharo.

github.com/lucretiomsp/Bloc5

Domenico Cipriani · ESUG 2026

Inria

#PCD2021

CASEY
REAS



Thanks Casey, my pleasure

“I think Processing continuing to be used in other programming languages and environments [...], that’s an exciting future, is like breaking the way of thinking out of Processing and Java...”

<https://youtu.be/obzuaPSk3V8?si=kYp3cRIKDBi98QOU>

Inria

Processing p5.js

Processing is a programming language and environment for creating visual art, animations, and interactive graphics.

p5.js is a friendly reimplementation of Processing's ideas in JavaScript, built by an inclusive, supportive community.

Bare calls like: **fill**, **circle**, **square** are implicitly methods on a hidden global sketch object.

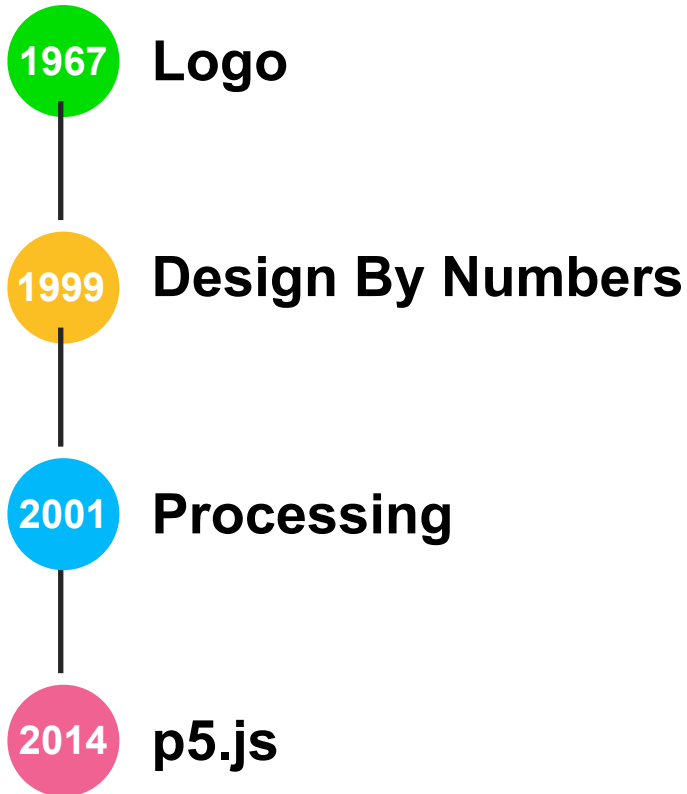
```
function setup() {  
  createCanvas(400, 400);  
}  
function draw() {  
  background(0);  
  circle(mouseX, mouseY, 50);  
}
```

Two main functions:

setup() runs once /

draw() runs on every frame.

A long tradition of teaching code through making



Seymour Papert's **Turtle graphics**. Code as a way for children to think.

John Maeda at M.I.T. Minimal language, visual-first.

Casey Reas & Ben Fry. Made coding art popular.

Lauren McCarthy. Radically accessible.

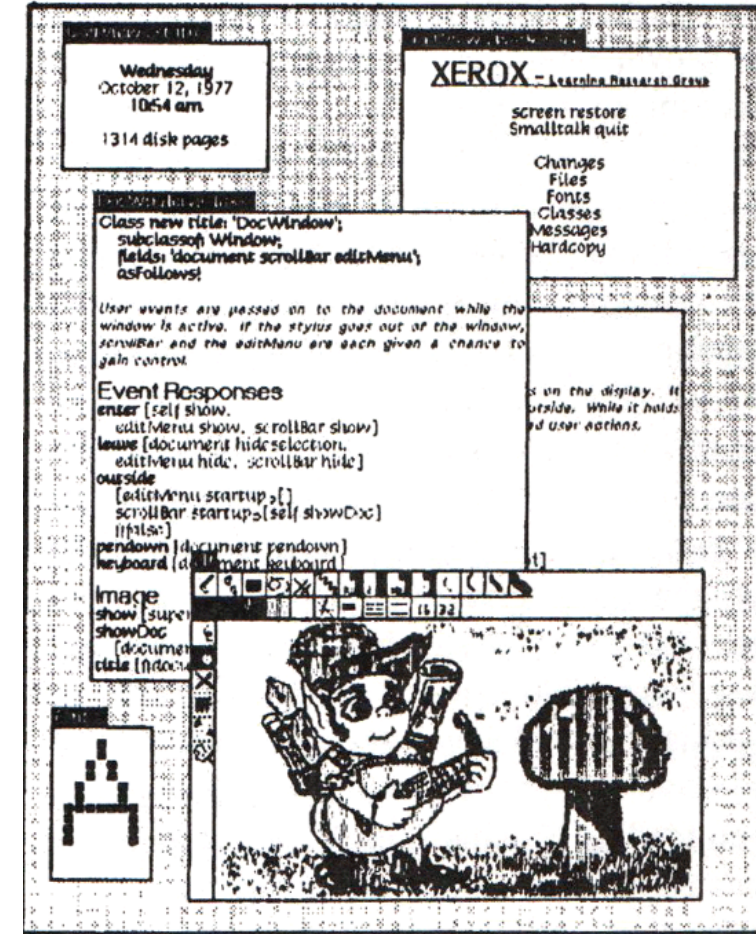
Smalltalk was born (also) for this

Smalltalk was designed at Xerox PARC as a language for **children** : a medium for creativity and learning, not just industry.

The Dynabook, painting interfaces, kids drawing on screens...

Then creativity quietly slipped off the agenda because...

Smalltalk grew up and got a day job...

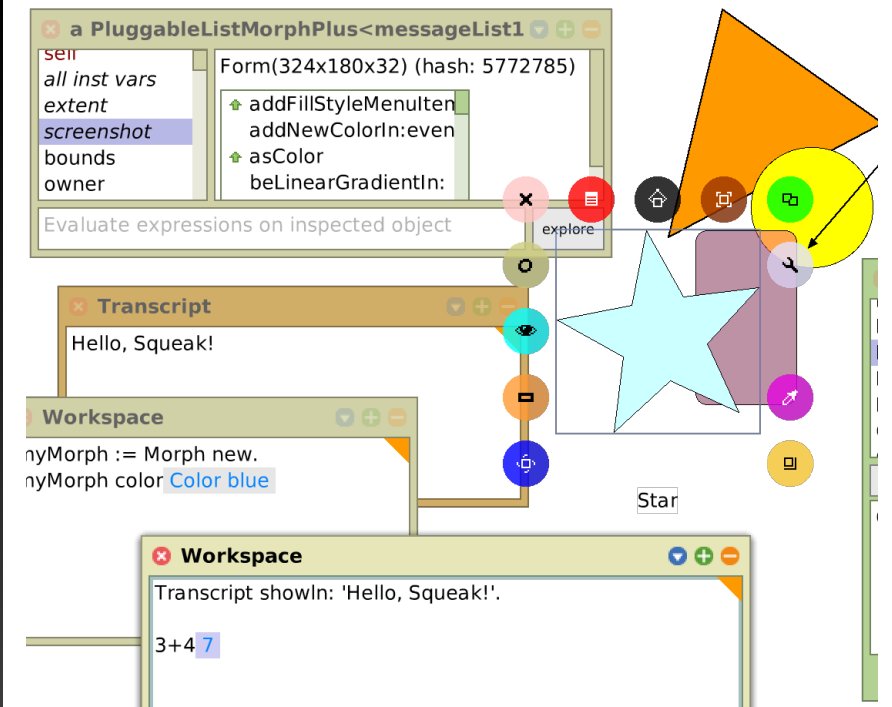


Inria

Morphic for visual art?

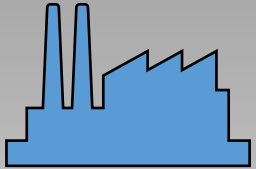
Morphic is **not** what can spark a generation of live coders in 2026

```
window := Morph new
  extent: 400@200;
  color: Color white;
  position: 100@100.
circle := (EllipseMorph new
  extent: 100@100; color: Color red; position: 1100@150).
square := (Morph new
  extent: 100@100; color: Color blue; position: 250@150).
triangle := (PolygonMorph vertices: { 50@0. 100@100. 0@100 }
  color: Color green
  borderWidth: 2
  borderColor: Color black).
window addMorph: circle; addMorph: square; addMorph: square.
window openInWorld.
window delete.
```



Inria

Pharo today



Industry

Widespread for business applications, trusted by many demanding commercial clients.



Research

A great platform for research and teaching good OOP.



Creativity?

Music arrived with me in 2022.

Inria

First, sound. Now, vision.



COYPU (2022) :
Programming music on-the-fly with Pharo.



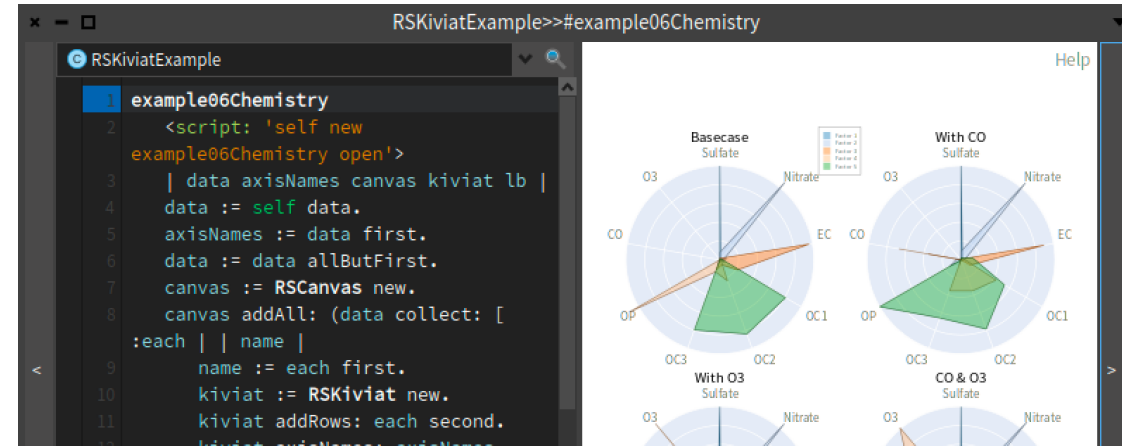
PHAUSTO (2024) :
sound synthesis (embedding FAUST) within Pharo.

Pharo can now make **sound**. It was missing **visual art**.

Inria

But doesn't Roassal already draw?

Roassal is excellent , for **data visualization**



Charts, graphs, trees, maps of data.

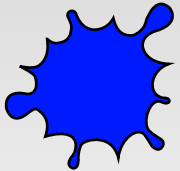
→ Bind data to: produce a visualization.

BLOC5 IS MADE FOR SKETCHING.

→ A Different goal: free creative expression, not insight from datasets.

BLOC5, a p5js inspired API for Bloc

An artful layer that turns Bloc into an immediate-mode sketching experience, while still taking advantage of many of its features.



Artists

Express ideas in code
without fighting the API.



Kids

A gentle, visual first
contact with programming.



Beginners

No Bloc literacy required.

What is Bloc?



Bloc is a low-level UI infrastructure & framework for Pharo.
State of the art library, but the API is verbose for sketching.

DAY-02
WED. JULY 8ND

Inria

ABSTRACTION! ABSTRACTION! ABSTRACTION! ABSTRACTION!



```
line := B5Line new from 20@20 to: 300 @ 300;  
stroke: Color green  
strokeWeight: 8
```

Írta

BLOC5's syntax fits on a Rizla

```
square := B5Square new
    extent: 100@100; fill: Color blue; position: 10@10 mode: #corner.
triangle := B5Triangle new v1: 50@0; v2: 100@100; v3 0@100; fill: Color green.
circle := B5Circle new diameter: 30; position: 300@300; fill: Color red.
line := B5Line new from 20@20 to: 300 @ 300; stroke: Color green; strokeWeight: 8.
txt := B5Text new text: 'ameega'; fontName: 'Helvetica'; fontSize: 44; position:
200@200.
cat := B5Image new fromFile: 'images/chuckCat.png'; extent: 200@200.
circle onKeyDown: [ :key | circle background: Color random].
square onMouseDown: [ :x :y | square position: x@y ].
triangle translate: 100@100.
cat rotate: 23.
space := B5Space new frameRate: 144 bpmAsFrameRate.
space drawAnimation: [ circle position: 300@(Random new * 400) ]
```

GOMME ARABIQUE NATURELLE

inria

BLOC

VS

BLOC5

```
square  BElement new
  size: 100@100;
  geometry: BIRectangleGeometry new;
  background: Color blue.

triangle := BElement new
  size: 100@100;
  geometry: (BIPolygonGeometry vertices: { 50@0. 100@100. 0@100 });
  background: Color green.

circle := BElement new
  size: 30@30;
  geometry: BICircleGeometry new;
  background: Color red.

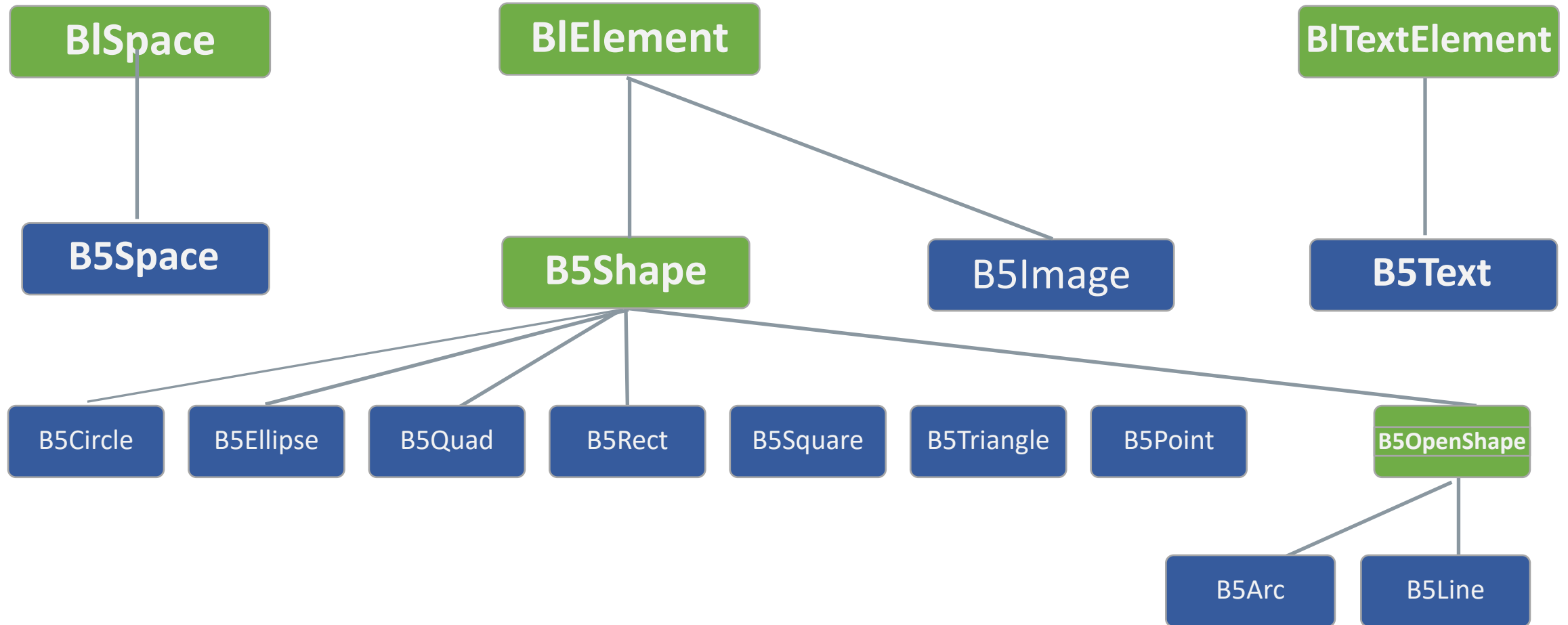
space := BISpace new.
space root addChild: square; addChild: triangle; addChild: circle
space show.
```

```
square := B5Square new
  extent: 100@100; fill: Color blue.
triangle := B5Triangle new
  v1: 50@0; v2: 100@100; v3: 0@100;
  fill: Color green.
circle := B5Circle new
  diameter: 30; fill: Color red.

space := B5Space new.
space root addChild: square; addChild: triangle; addChild: circle
space show.
```

Inria

BLOC5 class hierarchy



Animation in one message

drawAnimation: corresponds to p5's *draw()*:
the block runs every frame.

```
space := B5Space new frameRate: 30.  
c := B5Circle new position: 200@200; diameter: 80; fill: Color red.  
space addChild: c.  
t := 0.  
space drawAnimation: [  
    t := t + 0.05.  
    c position: 200 @ (200 + (50 * t sin)) ].  
space show.
```



```
drawAnimation: aBlock
```

```
task  
task := BlRepeatedTaskAction new  
    delay: (1000 / frameRate) milliseconds;  
    action: aBlock;  
    yourself.  
self enqueueTask: task
```

Interaction: event traits

A B5EventHandler trait gives every shape a p5-style event .

Mouse

```
onMouseClick:  onMouseDown:  onMouseUp:  
onMouseDrag:  onMouseEnter:  onMouseLeave:  
onMouseWheel
```

Keyboard

```
onKeyDown:  onKeyUp:
```

```
shape onClick: [ :x :y | shape fill: Color random ].  
shape onKeyDown: [ :key | shape position: shape position + (10@23) ].
```

Drawing Order

- p5.js: paint order = call order (immediate mode, irreversible)
- Bloc5: shapes are real children in Bloc's scene graph.
- Parent-child structure lets you transform whole "families" together rotate/translate/scale a group as one unit
- **addChild:** exposed deliberately -> no p5 equivalent

mouseX & mouseY

```
B5Space >> initialise
super initialize.
self frameRate: 60.
self root background: Color white.
mousePosition := 0 @ 0.
self root addEventHandlerOn:
BlMouseMoveEventdo: [ :event |
mousePosition := event localPosition ]
```

```
space drawAnimation: [
    circle position:
        space mouseX @ space mouseY].
```

```
space mouseX map: #(0 600 200 -300) .
space mouseY norm: #(0 600)
```

BLOC5 + Phausto = Audio/Visual Experience

Graphics from Bloc5. Sound from Phausto. One live image.

Pharo becomes a full creative-coding environment:
audio and visual, live, in one place.

The logo for BLOC5, featuring the letters 'B5' in a stylized, bold, red font.

BLOC5

+



Phausto

=



A/V

Inria

Future work

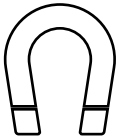


Full p5.js coverage

Bezier Curves and Splines, Color Effects, Full Typography

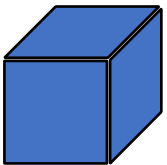


Full tutorials and examples



Fixed background

At the moment, user cant choose to redraw the background



3D rendering

With pharo-raylib already available thanks to [pharo-cig](#).

Conclusions

Bloc5 is good for Pharo

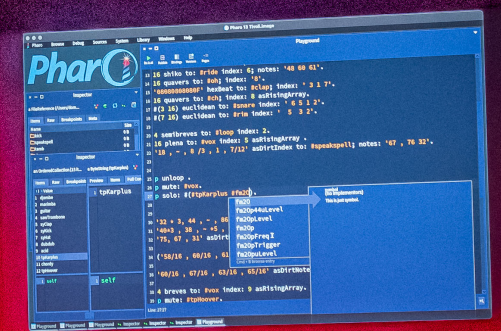
- Immediate visual feedback for kids and non-programmers
- Can bring creative coders to Pharo and the Smalltalk world
- Real-time stress-and-test for underlying Bloc infrastructure

What to improve

- Documentation, examples, tutorials are fundamental!
- Explicit choices on idiomatic choices
- Better handling for exceptions (debugger explosions on events)

The Pharo Music Image (beta)

A Pharo Music Image with Coypu, Phausto, CoypuIDE, Bloc5.
For artists, for kids, for punks, for greybeards. For everyone.



Thank you, have fun and contribute!



All cool but please never cut
my pretty nose again with that
GPSH*T never again

<https://github.com/lucretiomsp/Bloc5>

Invia