

ATHENS

Igor Stasenko

Smalltalks , 2012

ATHENS IS..

- a vector graphics framework for Pharo
- started around a year ago (April 2011)

GOALS

- modern graphics engine for Pharo
- make it simple
- make it object oriented (smalltalk oriented)
- one API, multiple backends

WHY WE NEED ATHENS

- Bitblt – lacks of abstraction
- We living in pixelated world
- Balloon engine is built “on top” of BitBlit, not-so-nice integration. As result – existing Canvas & Morphic code is cumbersome and overloaded
- hard to maintain & improve
- rendering quality

WHY WE NEED ATHENS

- new hardware: retina display
- resolution independent UIs

ATHENS DESIGN

User application

Athens API

Backends

Balloon

Cairo

SVG

OpenVG

OpenGL

INSPIRED BY

- BitBlt & Morphic
- OpenVG
- Cairo
- OpenGL
- SVG
- and of course SMALLTALK

API MAJOR POINT

- Implementation is based on ROLES and their protocols, NOT classes

~~canvas setColor: Color red.
canvas drawRectangle: (0@0
corner: 100@100).

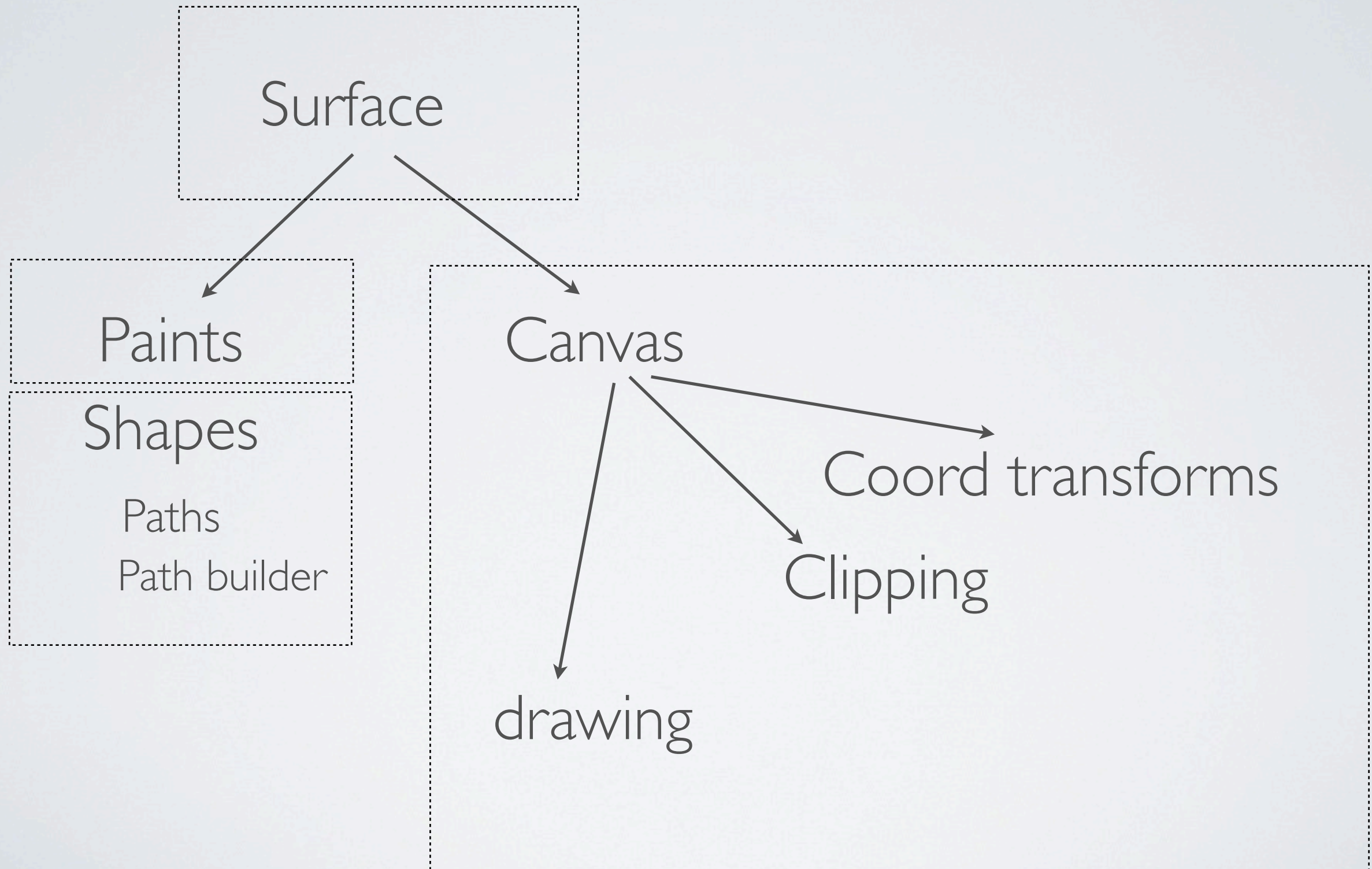
canvas setGradient: mygradient.
canvas drawLine: pt1 to pt2.~~

canvas setPaint: Color red.
canvas drawShape: (0@0
corner: 100@100).

canvas
 setPaint: mygradient;
 setShape: myLine;
 draw.

drawShape: anObject
 self setShape: anObject.
 self draw.

ROLES



SURFACE

- represents an output media & defines backend which you will be using.
- acts as a factory for creating paints, paths, canvas & other involved objects
- access to its canvas
- surfaces is ALWAYS backend-specific

CANVAS

- single drawing operation: **canvas draw**.
- maintains current coordinate system transformations:
 - canvas pathTransform
 - canvas paintTransform
- maintains currently selected shape:
 - canvas setShape: ...
- maintains currently selected paint:
 - canvas setPaint:
- paint modes (paint/over/clear/etc)

PAINT

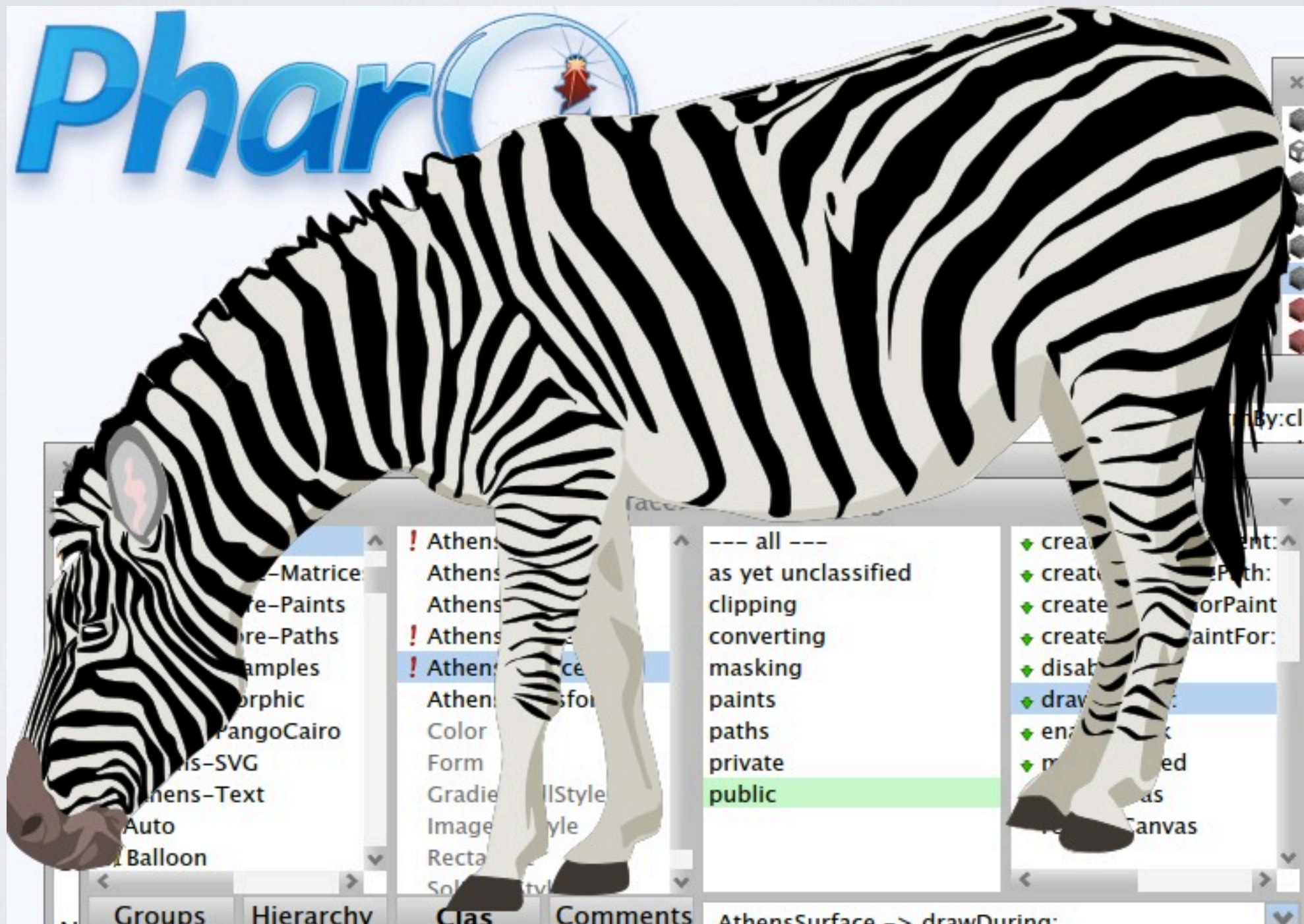
- a “paint” which is used to fill the shape
 - solid color
 - gradients
 - stroke paints
 - bitmap (form) paints
 - you can define own

SHAPE

- defines an enclosing region of drawing operation
 - rectangle
 - paths
 - a contours consisting of connected segments:
 - line
 - quadric Bezier curve
 - cubic Bezier curve
 - circular arcs
 - elliptical arcs
 - you can define own

DEMO

SVG



THINGS TO DO

- masking
- SVG export/import
- paint modes (overlay, screen , blend etc)
- OpenGL backend
- elliptical arcs
- DOCS



QUESTIONS?

THANKS

- Fernando Olivero for helping with implementation
- RMoD team for discussions & design decisions
- Pharo & its fans :)