Polyphony:
Programming Interfaces and Interactions with the Entity-Component-System Model

Thibault Raffaillac, Stéphane Huot

Programming UIs with ECS

1. ECS, a composition model for video games

2. Polyphony, an experimental interaction toolkit

3. Designing UIs with *composition over inheritance*

4. Contributions, and future work

Entity-Component-System

An architectural pattern developed for video games
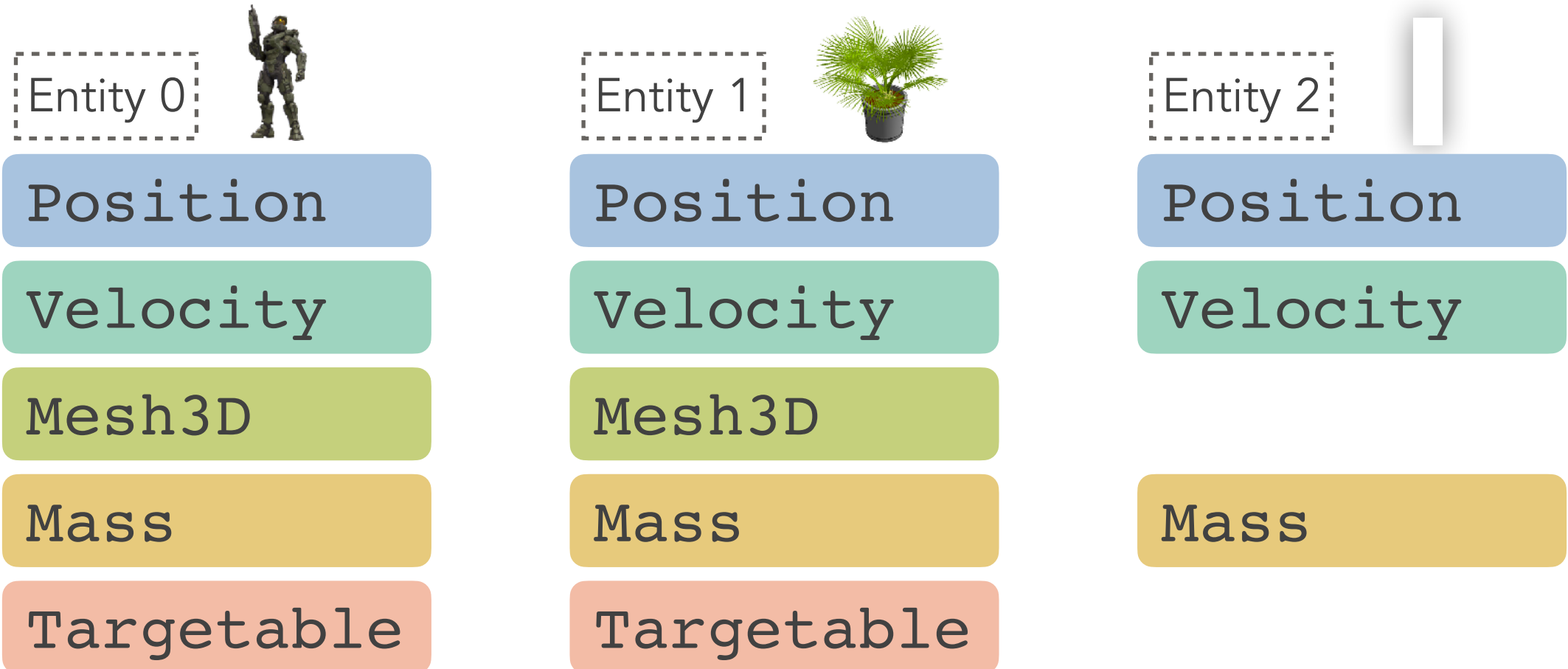
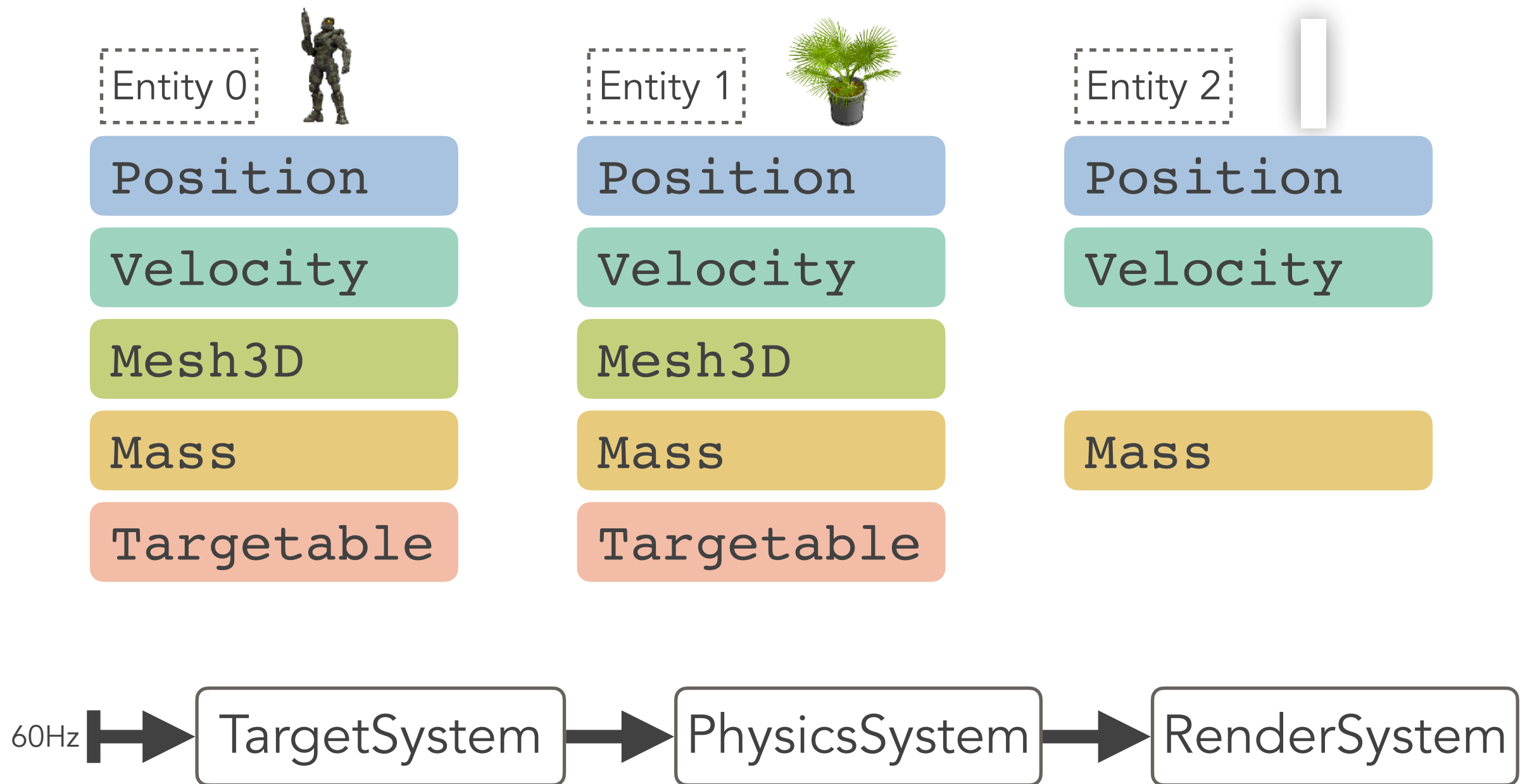Alternative to Object-Oriented Programming

Entity-Component-System

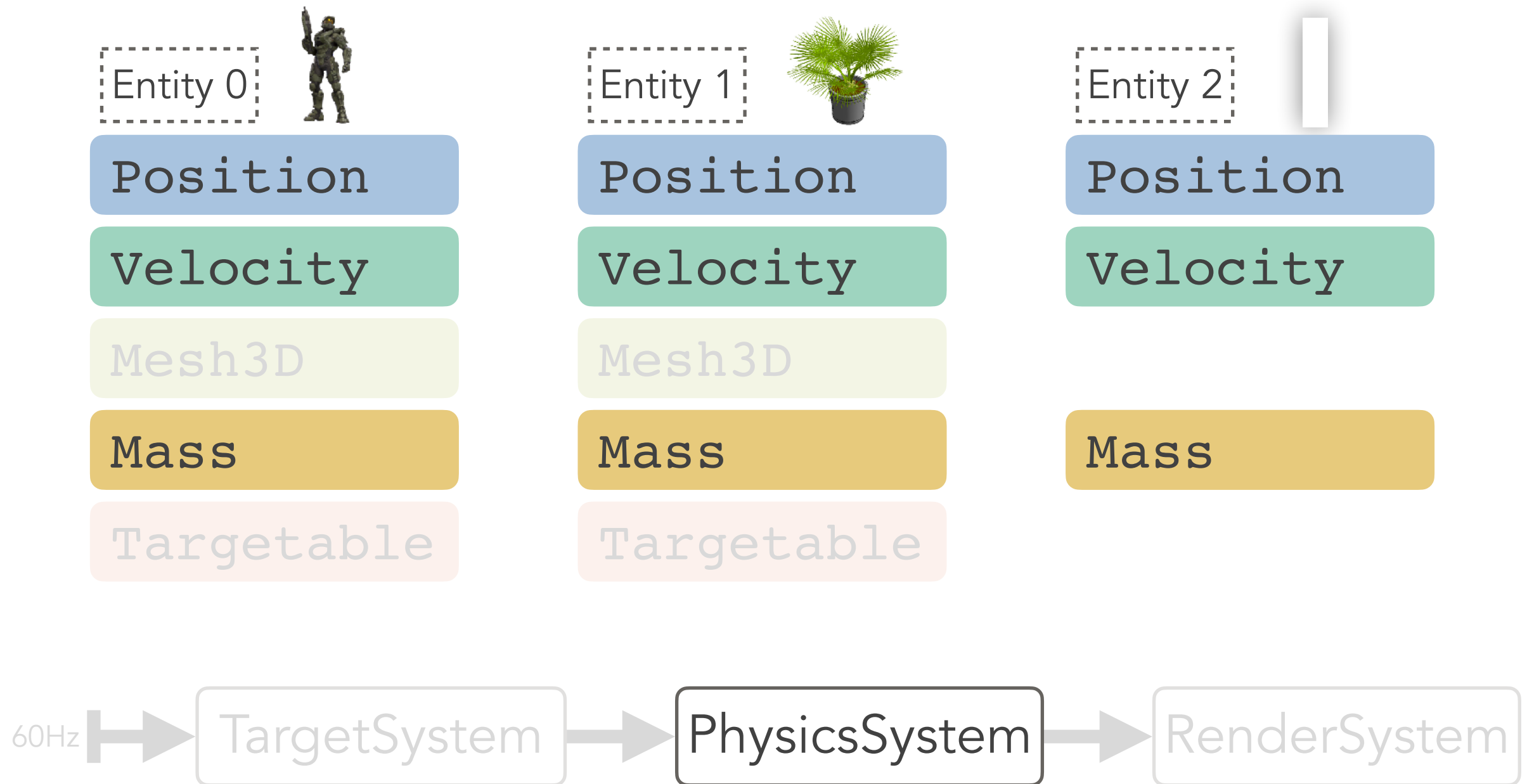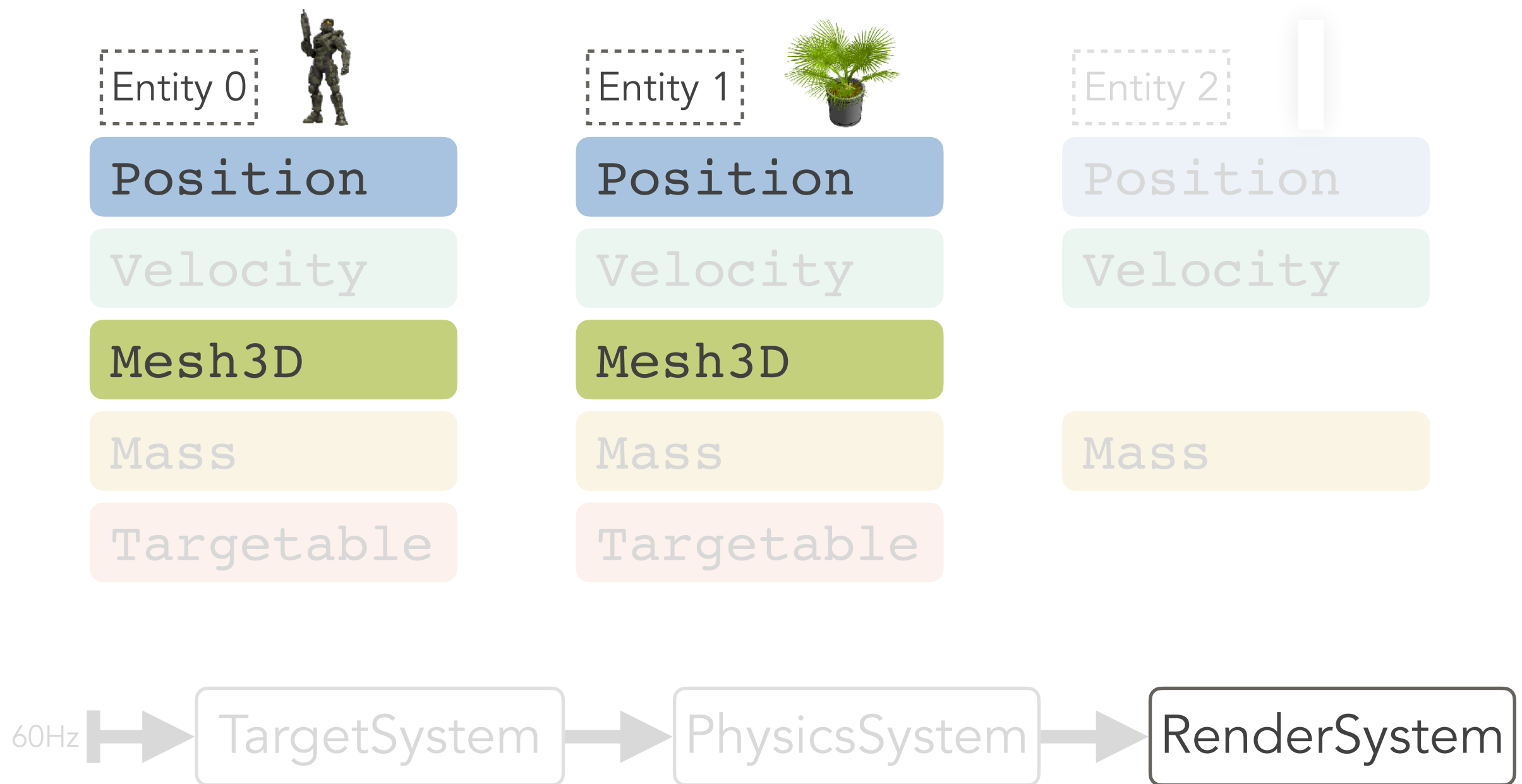Entity 0   Entity 1   Entity 2

Entity-Component-System

Entity 0

Position
Velocity
Mesh3D
Mass
Targetable

Entity 1

Position
Velocity
Mesh3D
Mass
Targetable

Entity 2

Position
Velocity
Mass

Entity-Component-System

Entity 0	Entity 1	Entity 2
Position	Position	Position
Velocity	Velocity	Velocity
Mesh3D	Mesh3D	
Mass	Mass	Mass
Targetable	Targetable	

60Hz ➡ **TargetSystem** ➡ **PhysicsSystem** ➡ **RenderSystem**

Entity-Component-System

Entity 0	Entity 1	Entity 2
Position	Position	Position
Velocity	Velocity	Velocity
Mesh3D	Mesh3D	
Mass	Mass	Mass
Targetable	Targetable	

60Hz → TargetSystem → PhysicsSystem → RenderSystem

Entity-Component-System

Entity 0	Entity 1	Entity 2
Position	Position	Position
Velocity	Velocity	Velocity
Mesh3D	Mesh3D	
Mass	Mass	Mass
Targetable	Targetable	

60Hz → **TargetSystem** → **PhysicsSystem** → **RenderSystem**

Why should we care?

Thief: The Dark Project (1998)

Operation Flashpoint: Dragon Rising (2007)

Unity (2019)

Dungeon Siege (2002)

Overwatch (2016)

Why should we care?

→ C ⓘ https://forum.unity.com/threads/gui-in-pure-ecs-projects.530578/ 🐾 ★ 👤

I would guess that most of us are just doing very bare bones "detecting clicks and touches on sprites" at the moment, but has anyone come across a true UI framework for ECS yet? Or have any ideas about where you intend to go with this? Are you writing your own input fields that handle mobile keyboards and the whole nine yards? Writing everything from square 1 seems a daunting and wasteful task, but is it unavoidable?

G gamedev.net 🔳 🔍 ☰

Paragon123 ⊕ 620
Posted July 26, 2016 ⮜

Does anyone have any tips, or articles about creating GUIs in an ECS system?

G gamedev.net 🔳 🔍 ☰

Tangletail ⊕ 2915
Posted July 26, 2016 ⮜

ECS is the LAST pattern you'd want to use for GUI. GUI gets really complicated real fast. If you've never programmed one, I'd suggest taking a look at some libraries to see just how involved it can actually get.

← → C 🔒 https://love2d.org/forums/viewtopi... ★ 👤 ⋮

ECS for GUI
📄 Sat Oct 10, 2015 10:21 am

So guys I have been thinking about GUIs recently. I have seen tutorials for Unity's GUI system from 4.6+ and it reminds me of an ECS. In reality it is probably done with classes and not entities but considering lua is better fit for ECS rather than OOP maybe an ECS based GUI would be a good thing? One benefit of an ECS based GUI would be that the output of a hypothetical GUI editor would be pretty easy to read and make. But what are yall opinions on a ECS based GUI?
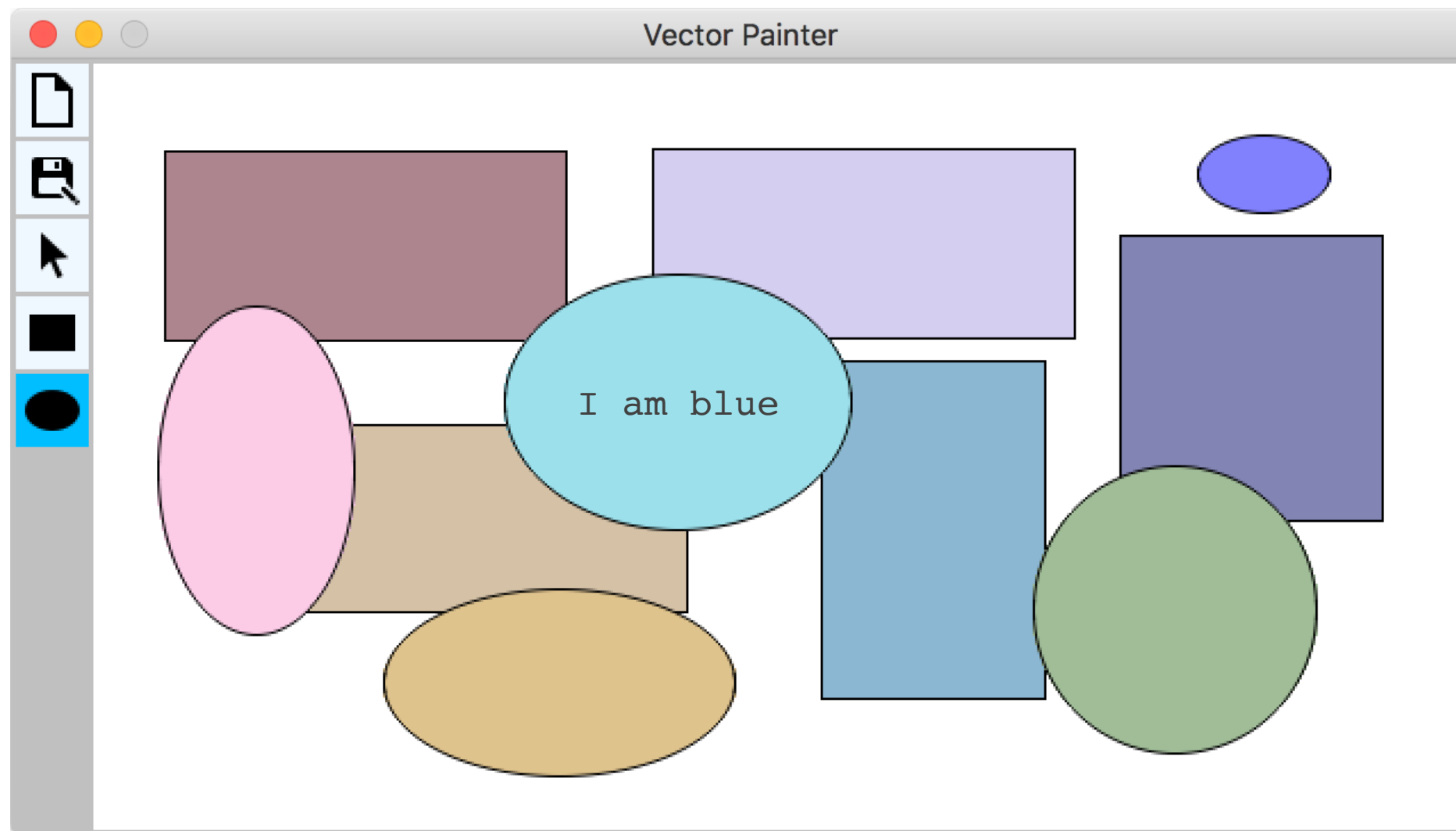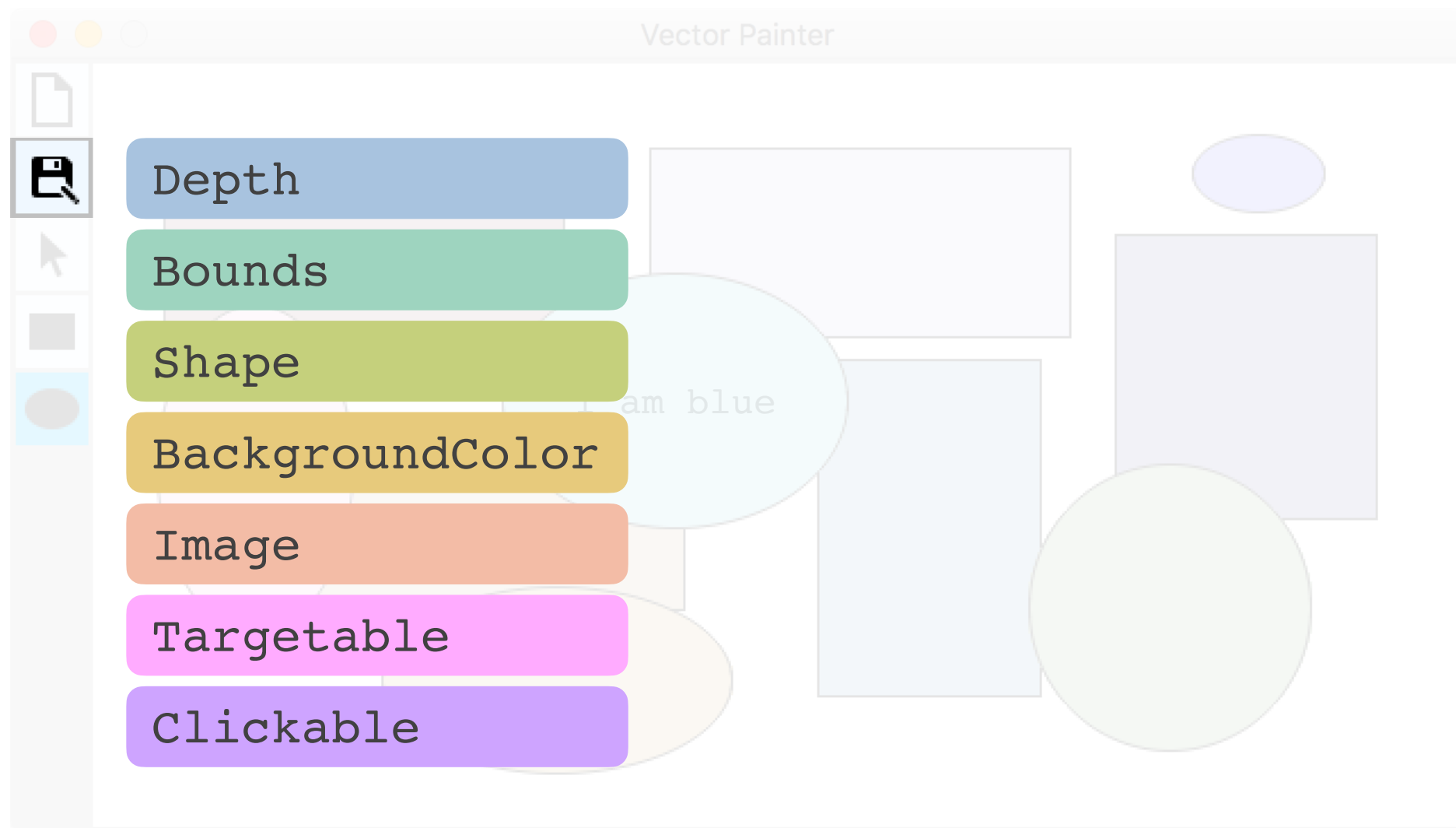
→ C ⓘ https://forum.unity.com/threads/gui-in-pure-ecs-projects.530578/

It is possible to make UI in ECS. Still I dont think anybody will do it anytime soon.

Apr 1, 2019

Programming UIs with ECS
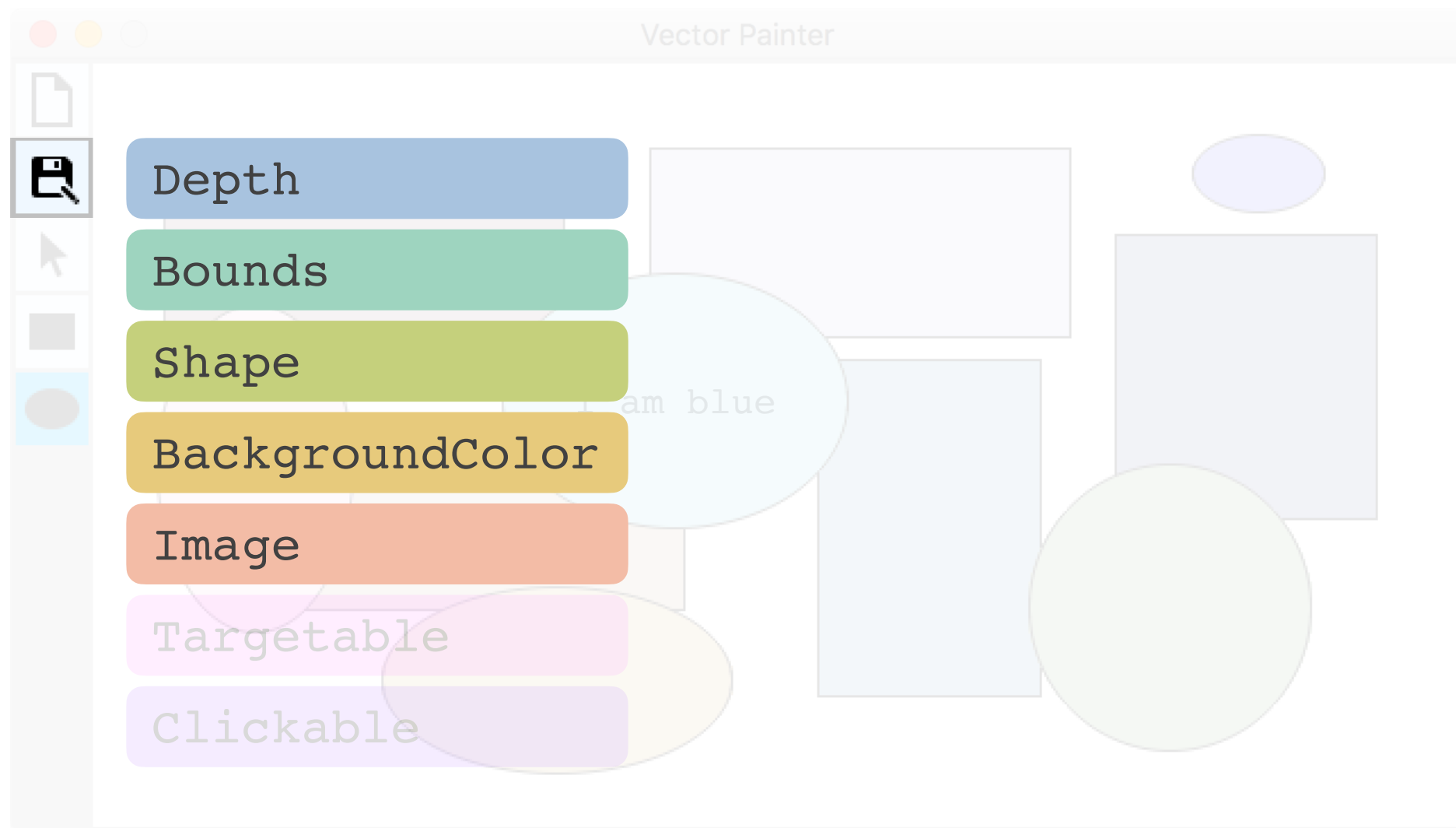
Illustrating Polyphony
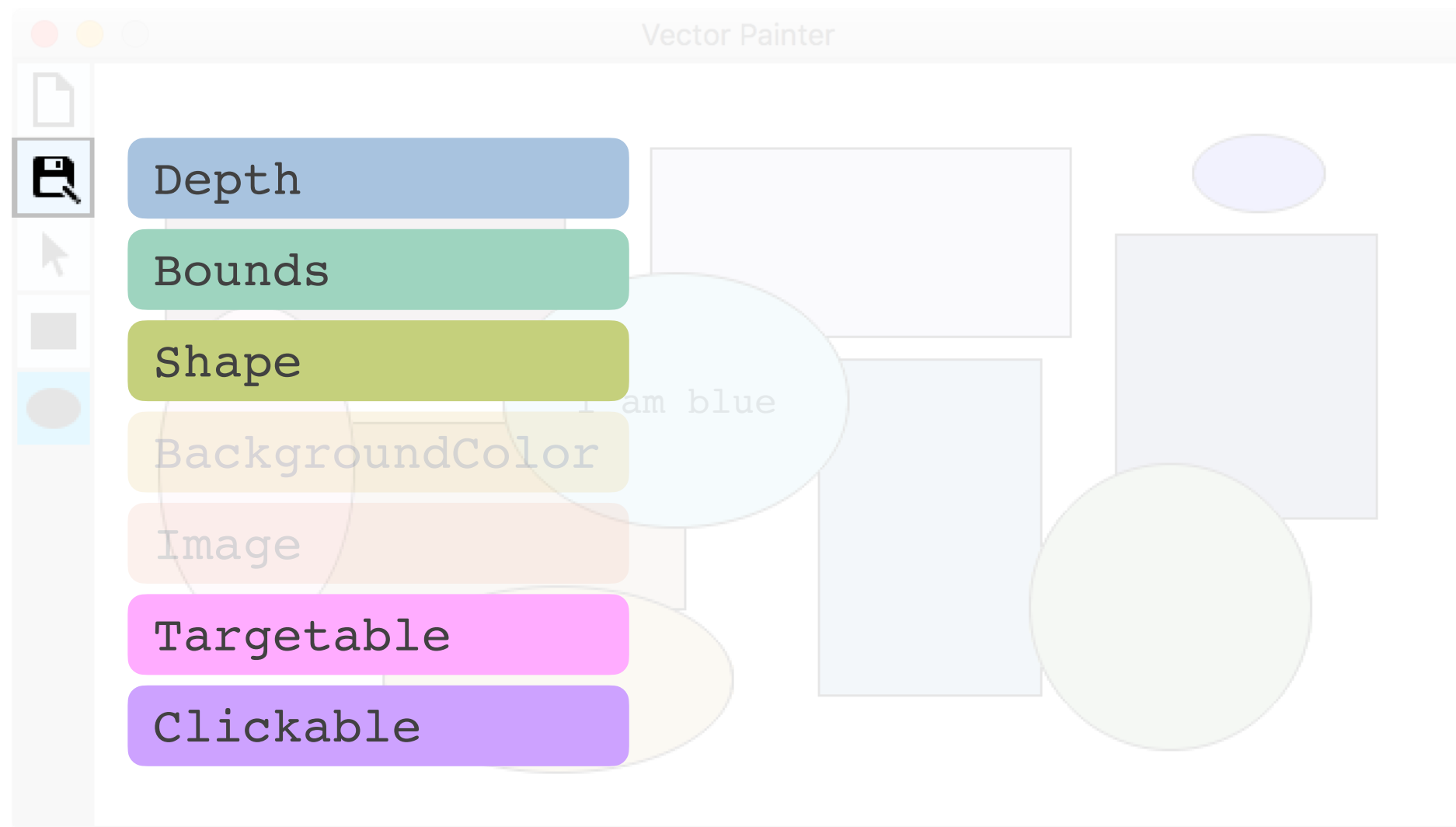
UI toolkit for Node.js, using SDL and GFX
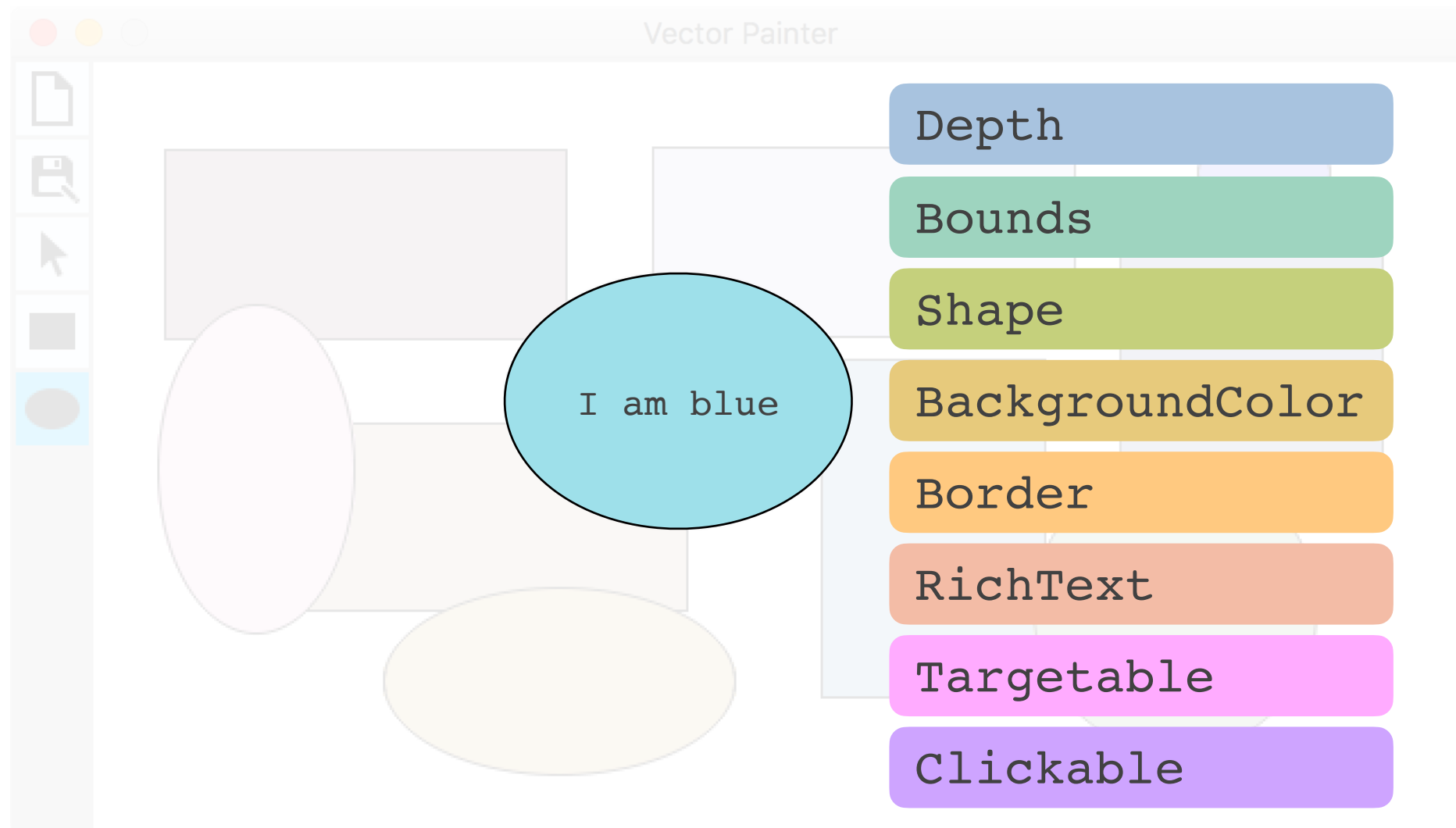
Illustrating Polyphony

Illustrating Polyphony

Illustrating Polyphony

Illustrating Polyphony

Device Entities

CursorPosition

Buttons

KeyStates

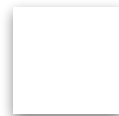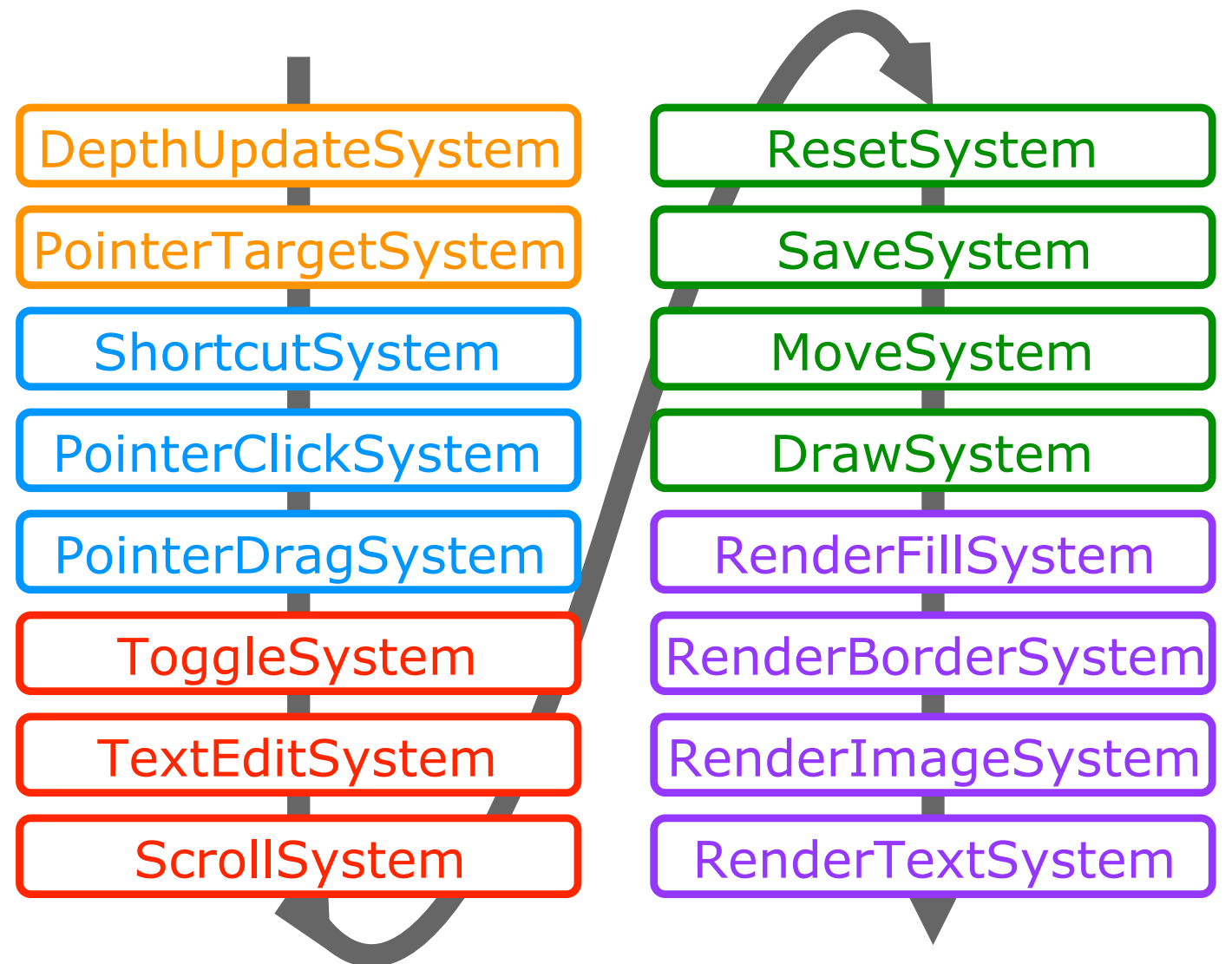Bounds

Origin

The chain of Systems

Input management

Interaction techniques

Widget-specific

Application-specific

Output rendering

DepthUpdateSystem

PointerTargetSystem

ShortcutSystem

PointerClickSystem

PointerDragSystem

ToggleSystem

TextEditSystem

ScrollSystem

ResetSystem

SaveSystem

MoveSystem

DrawSystem

RenderFillSystem

RenderBorderSystem

RenderImageSystem

RenderTextSystem

The chain of Systems

Entities with Targetable

Entities with CursorPosition

DepthUpdateSystem

PointerTargetSystem

ShortcutSystem

PointerClickSystem

PointerDragSystem

ToggleSystem

TextEditSystem

ScrollSystem

ResetSystem

SaveSystem

MoveSystem

DrawSystem

RenderFillSystem

RenderBorderSystem

RenderImageSystem

RenderTextSystem

The chain of Systems

CursorPosition

Buttons

TmpMotion

added by SDL binding

Target

TmpEnter

DepthUpdateSystem	ResetSystem
PointerTargetSystem	SaveSystem
ShortcutSystem	MoveSystem
PointerClickSystem	DrawSystem
PointerDragSystem	RenderFillSystem
ToggleSystem	RenderBorderSystem
TextEditSystem	RenderImageSystem
ScrollSystem	RenderTextSystem

The chain of Systems

CursorPosition

Buttons

TmpRelease

added by SDL binding

Target

TmpClick

DepthUpdateSystem	ResetSystem
PointerTargetSystem	SaveSystem
ShortcutSystem	MoveSystem
PointerClickSystem	DrawSystem
PointerDragSystem	RenderFillSystem
ToggleSystem	RenderBorderSystem
TextEditSystem	RenderImageSystem
ScrollSystem	RenderTextSystem

Programming UIs with ECS

1. ECS, a composition model for video games

2. Polyphony, an experimental interaction toolkit

3. Designing UIs with *composition over inheritance*
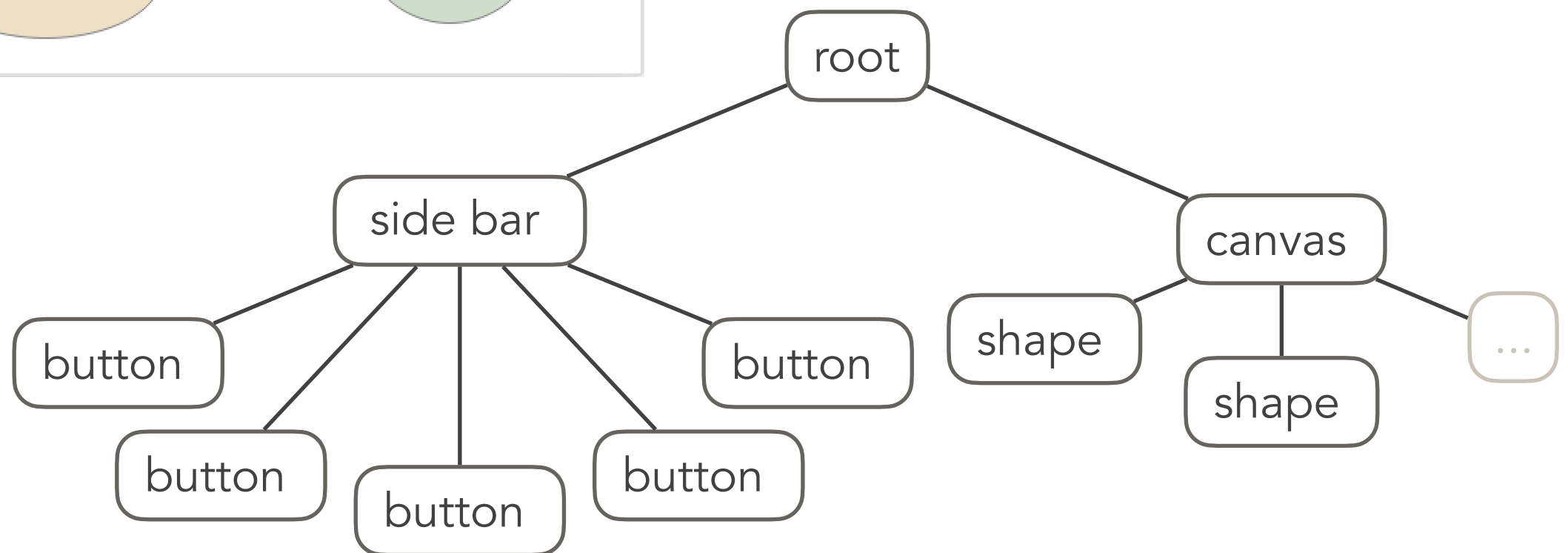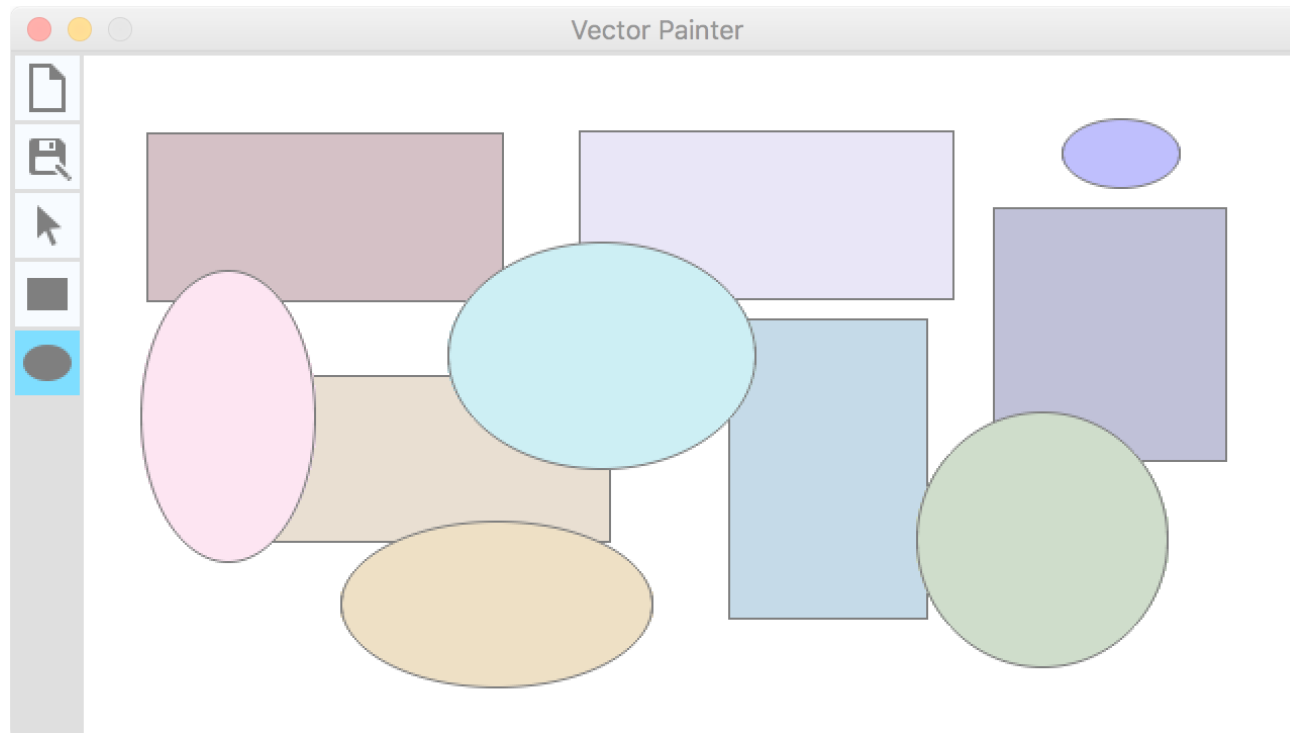
4. Contributions, and future work

Composition over inheritance

More than one "parent" per element → *Do we need it?*

3 hierarchies:

- scene tree

- type tree

- interaction graph

The scene tree
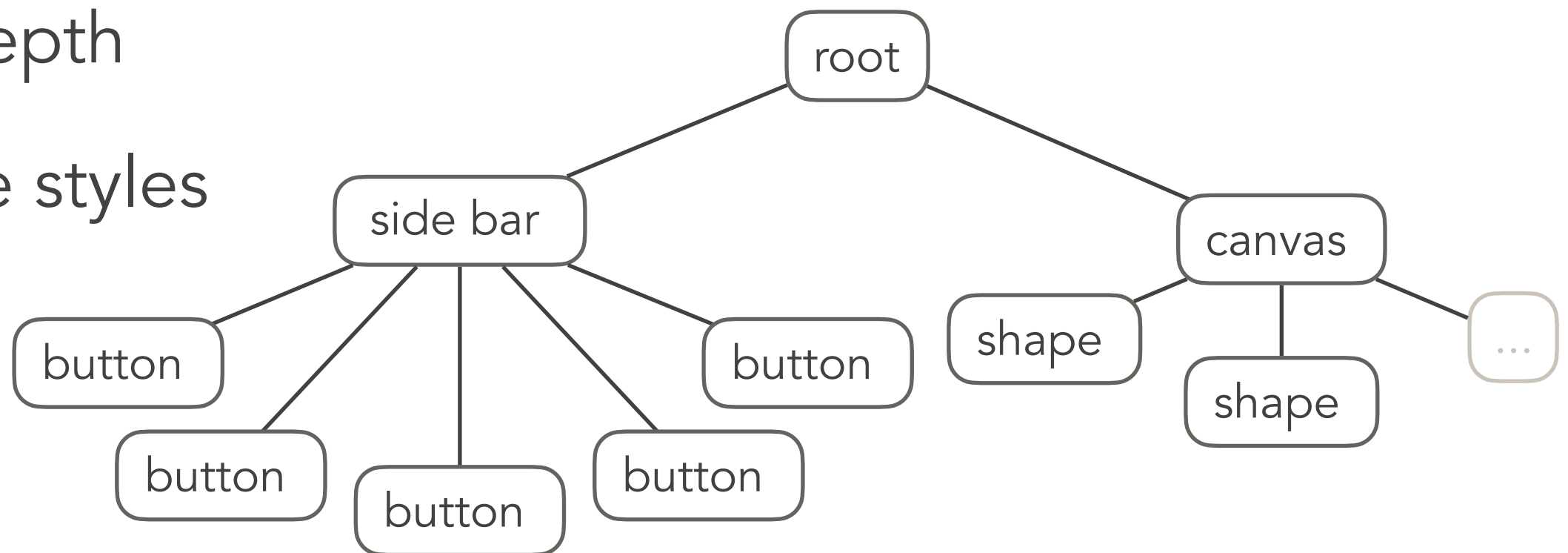
The scene tree

~~Iterate over nodes~~ **built-in with ECS**
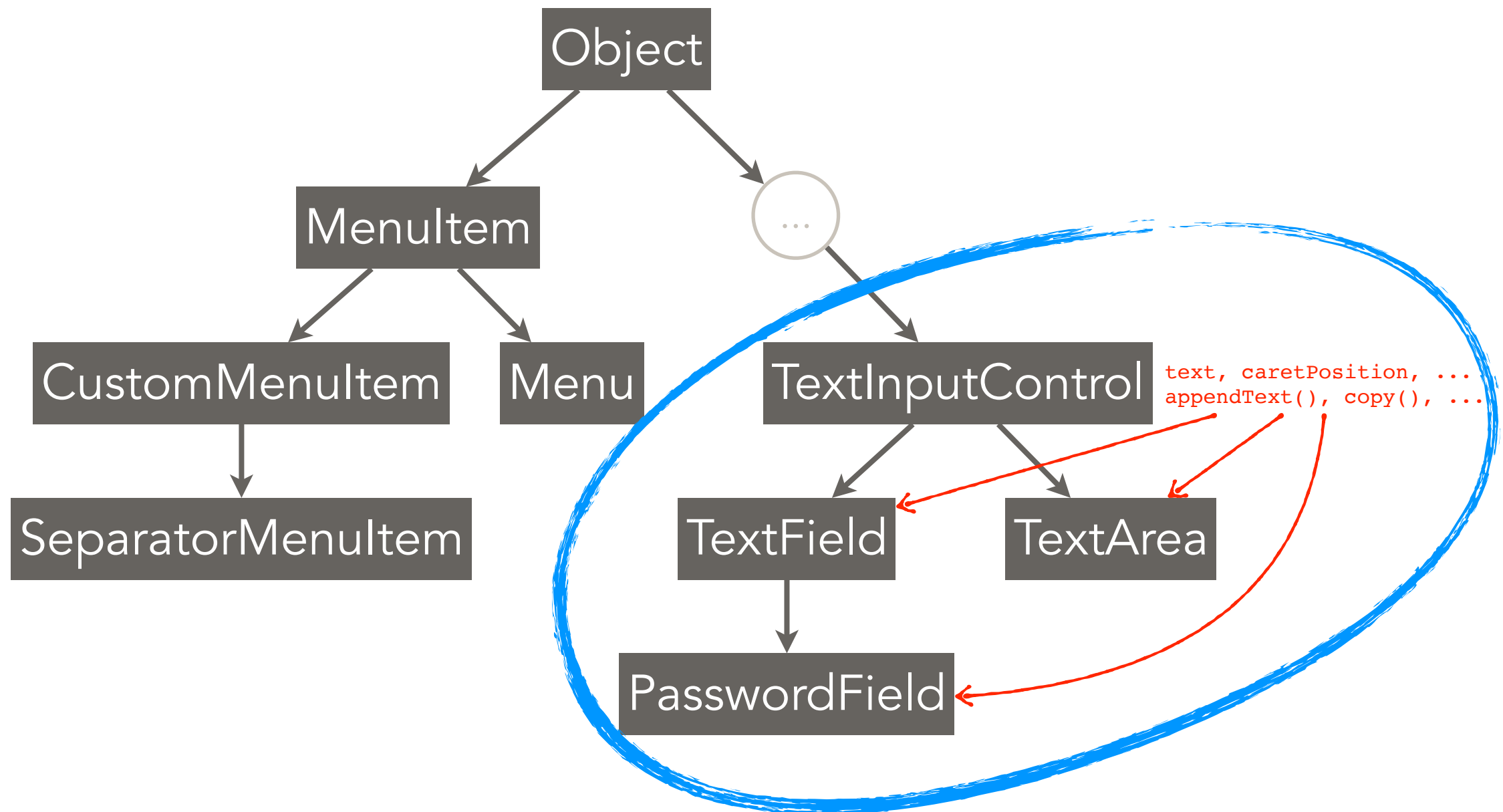
Explicit references

~~Relative layout~~ **global positioning (e.g. Cassowary)**
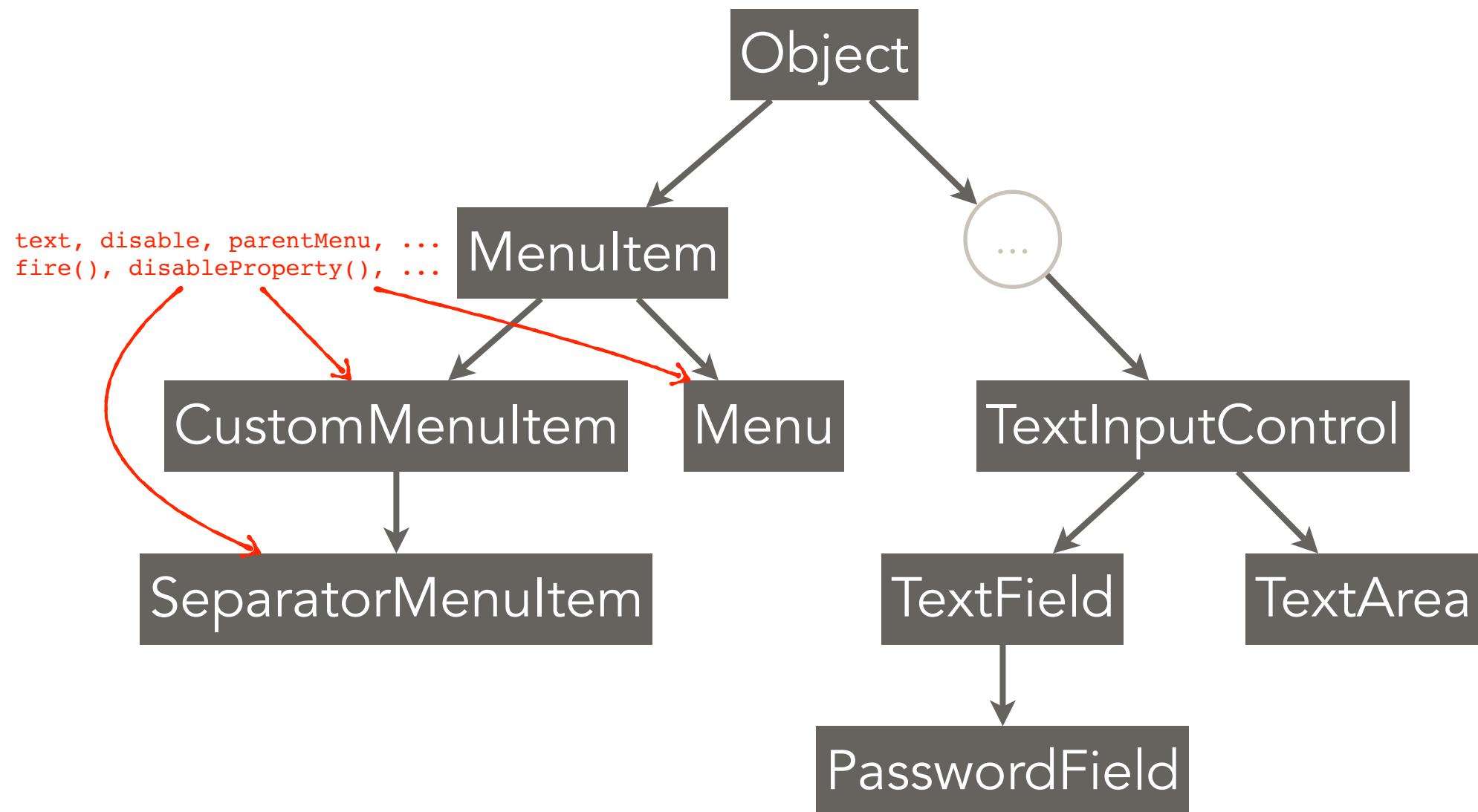
Display depth

Propagate styles

The type tree

Object

MenuItem

...

CustomMenuItem

Menu

TextInputControl

text, caretPosition, ...
appendText(), copy(), ...

SeparatorMenuItem

TextField

TextArea

PasswordField

The type tree

text, disable, parentMenu, ...
fire(), disableProperty(), ...

Object
MenuItem
...
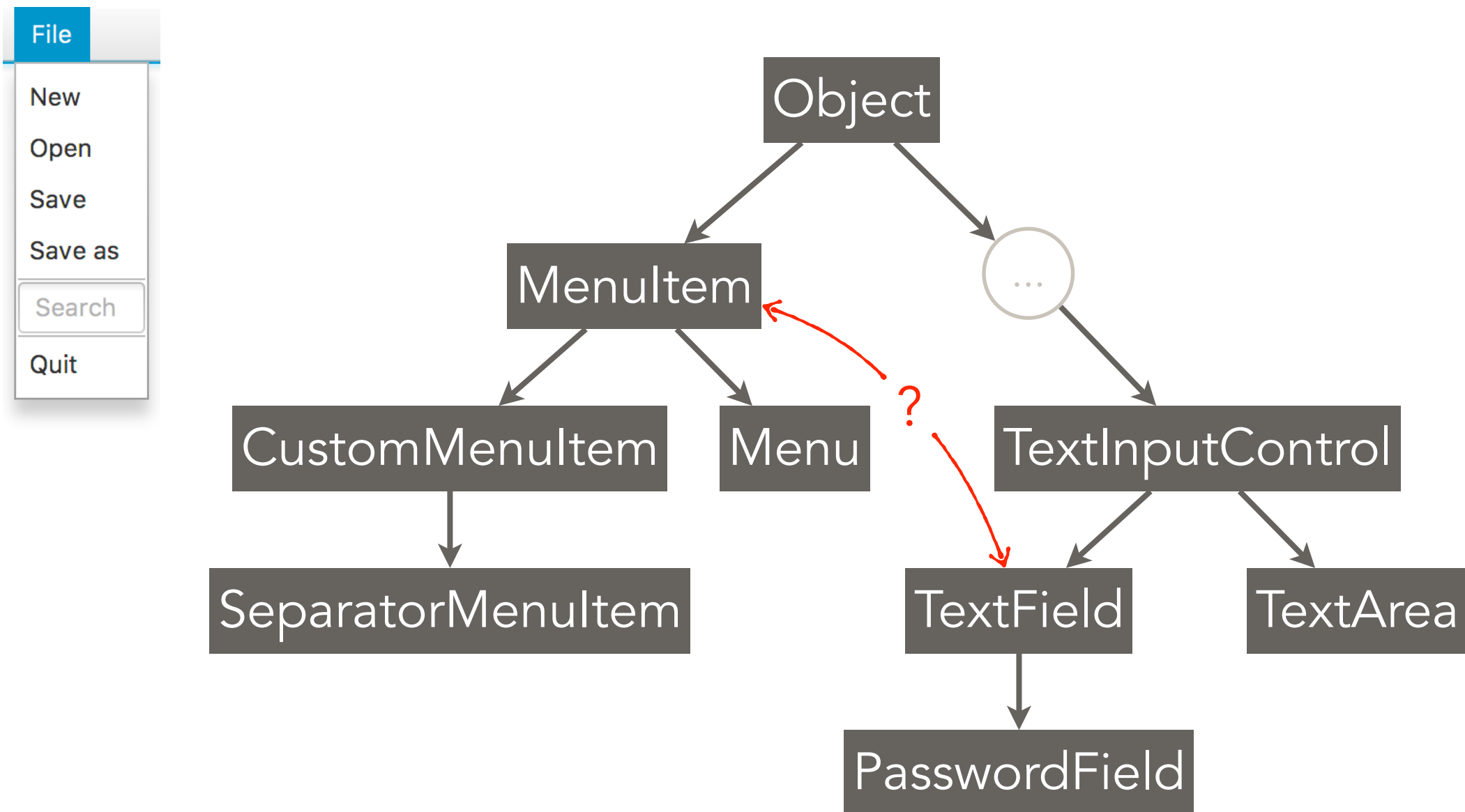CustomMenuItem
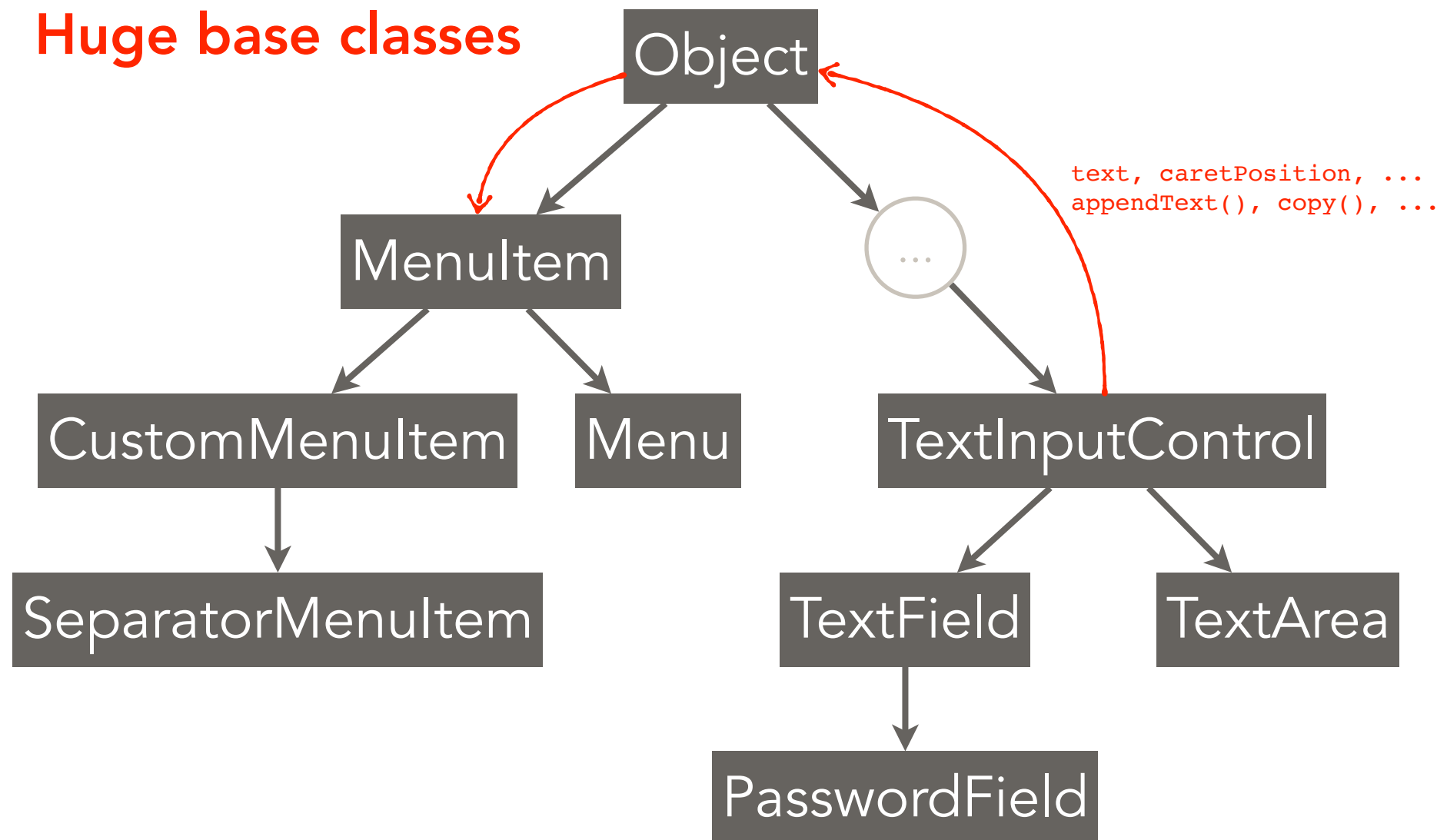Menu
TextInputControl
SeparatorMenuItem
TextField
TextArea
PasswordField

The type tree

The type tree

Huge base classes

Object

MenuItem

...

text, caretPosition, ...
appendText(), copy(), ...

CustomMenuItem

Menu

TextInputControl

SeparatorMenuItem

TextField

TextArea

PasswordField

The type tree

Poor interaction with inheritance

Object

TextEditable

MenuItem

...

`text, caretPosition, ...`
`appendText(), copy(), ...`

CustomMenuItem

Menu

TextInputControl

SeparatorMenuItem

TextField

TextArea

PasswordField

File
New
Open
Save
Save as
Search
Quit

The type tree

Object

MenuItem

...

CustomMenuItem   Menu   TextInputControl

SeparatorMenuItem   TextField   TextArea

PasswordField

menu item

text field

More complex scene tree

The type tree
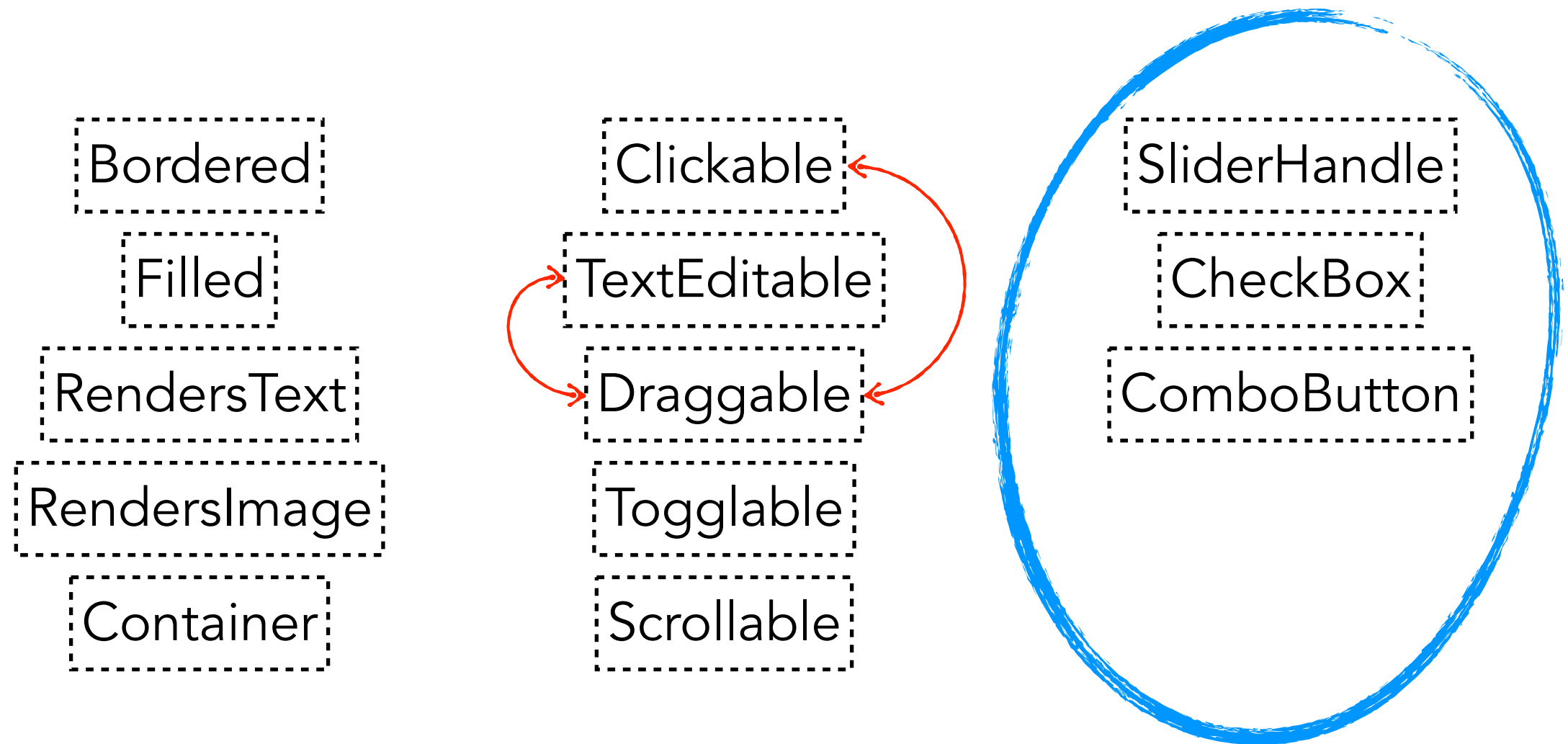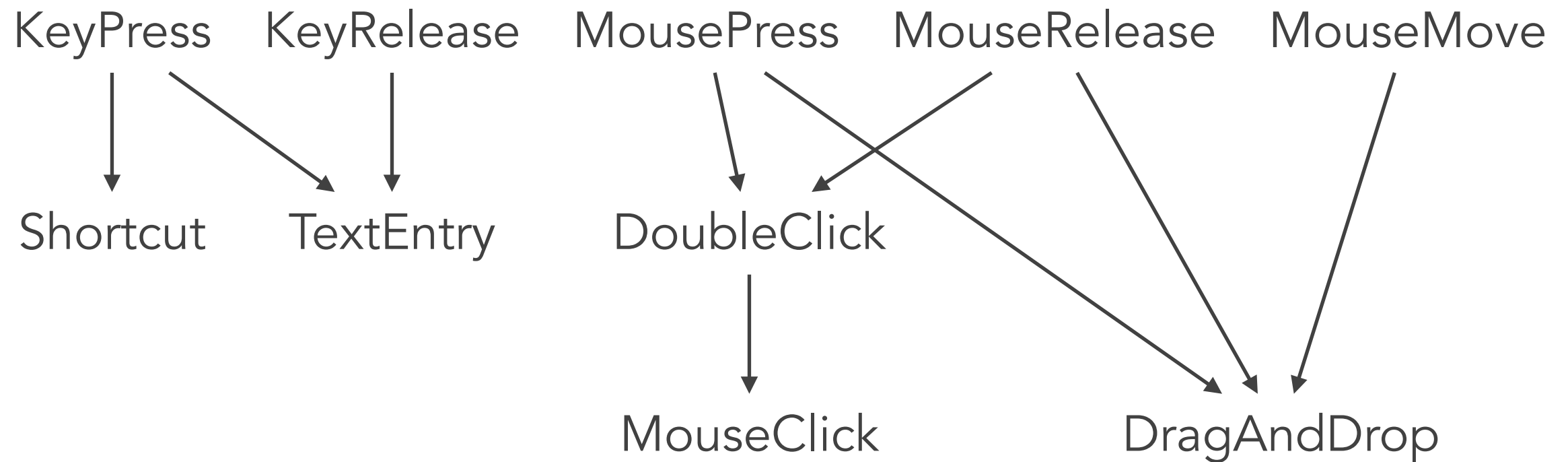
Bordered

Filled

RendersText

RendersImage

Container

Clickable

TextEditable

Draggable

Togglable

Scrollable

SliderHandle

CheckBox

ComboButton

The interaction graph

KeyPress   KeyRelease   MousePress   MouseRelease   MouseMove

Shortcut   TextEntry   DoubleClick
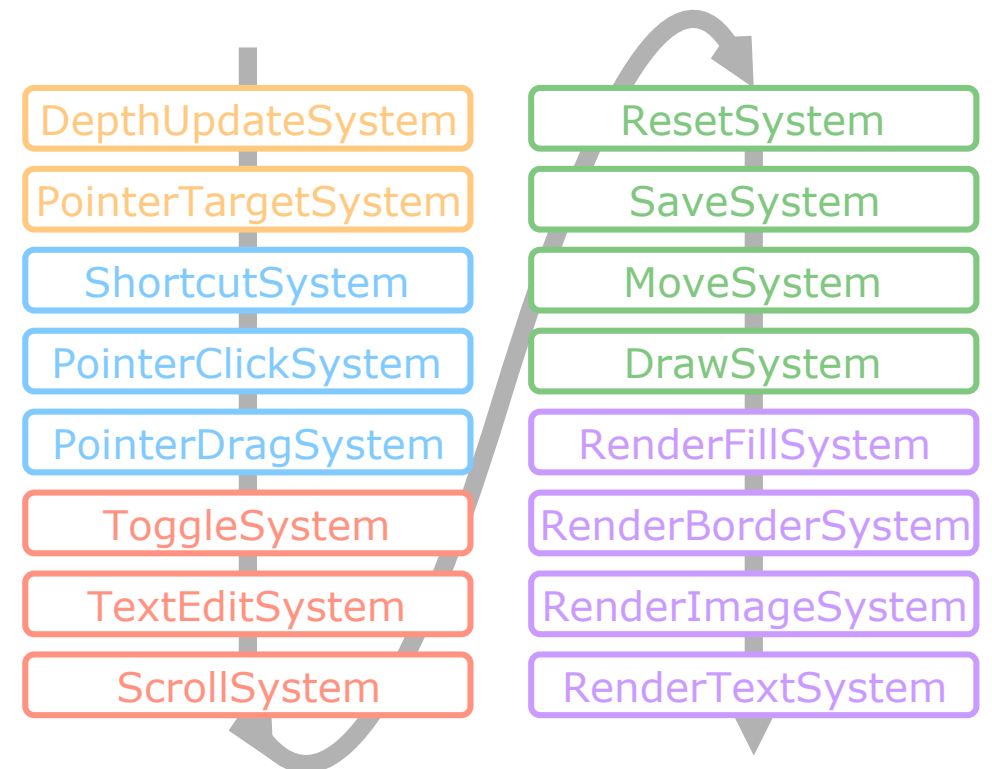
MouseClick   DragAndDrop

Programming UIs with ECS

1. ECS, a composition model for video games

2. Polyphony, an experimental interaction toolkit

3. Designing UIs with *composition over inheritance*

4. Contributions, and future work

No callbacks

All of the logic is contained inside Systems

Systems react to events through *temporary* Components

👍 No "spaghetti of callbacks" (Myers, 1991)

👎 Less control over the code of widgets (appearance)

🤔 Bigger functions

DepthUpdateSystem
PointerTargetSystem
ShortcutSystem
PointerClickSystem
PointerDragSystem
ToggleSystem
TextEditSystem
ScrollSystem

ResetSystem
SaveSystem
MoveSystem
DrawSystem
RenderFillSystem
RenderBorderSystem
RenderImageSystem
RenderTextSystem

No event structures

Event occurrence is signalled on device Entities

Event propagation with permanent and temporary Components
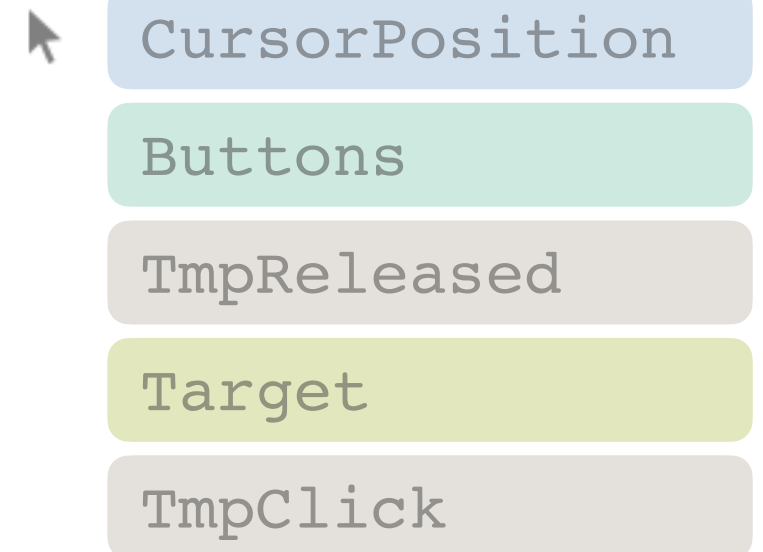
👍 No possibility of queuing/delaying events

👍 Preserve Components of specific devices (pressure touch)

👎 More work to pass events across applications/machines

`CursorPosition`

`Buttons`

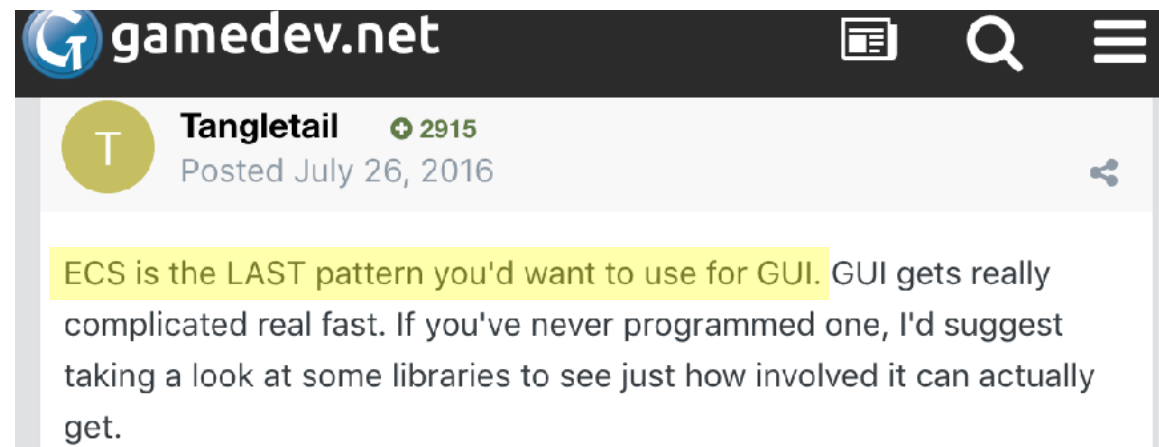`TmpReleased`

`Target`

`TmpClick`

Adapting ECS to UIs

Systems are Entities (modelling dependencies with Components)

Systems chain filtering (support for multimodality)

Device Entities (supporting multiple devices)

Temporary Components (reacting to events without callbacks)

Future works

gamedev.net

Tangletail ⊕ 2915
Posted July 26, 2016

ECS is the LAST pattern you'd want to use for GUI. GUI gets really complicated real fast. If you've never programmed one, I'd suggest taking a look at some libraries to see just how involved it can actually get.

Use article to promote support for interaction in ECS

Build more complex UIs

Contribute to future ECS language

Links

https://gitlab.inria.fr/Loki/PolyphonyECS/

https://www.gamasutra.com/view/feature/131762/postmortem_thief_the_dark_project.php

http://t-machine.org/index.php/category/entity-systems/

http://entity-systems.wikidot.com/

https://unity3d.com/learn/tutorials/topics/scripting/introduction-ecs

http://bit.ly/2ZzI6rc ← programming interaction study