

PL@NES - Reading Club

Paper: **Deprecating the Observer
Pattern with Scala.React**

by Ingo Maier & Martin Odersky

Christophe.VanGinneken@cs.kuleuven.be

THE FOLLOWING **PRESENTATION** DOES **NOT** INCLUDE
MATERIAL ABOUT **SCALA** NOR **SCALA.REACT**.
IT WILL HIGHLIGHT THE GREAT PARTS OF THE ORIGINAL
PAPER IN A GENERIC WAY.



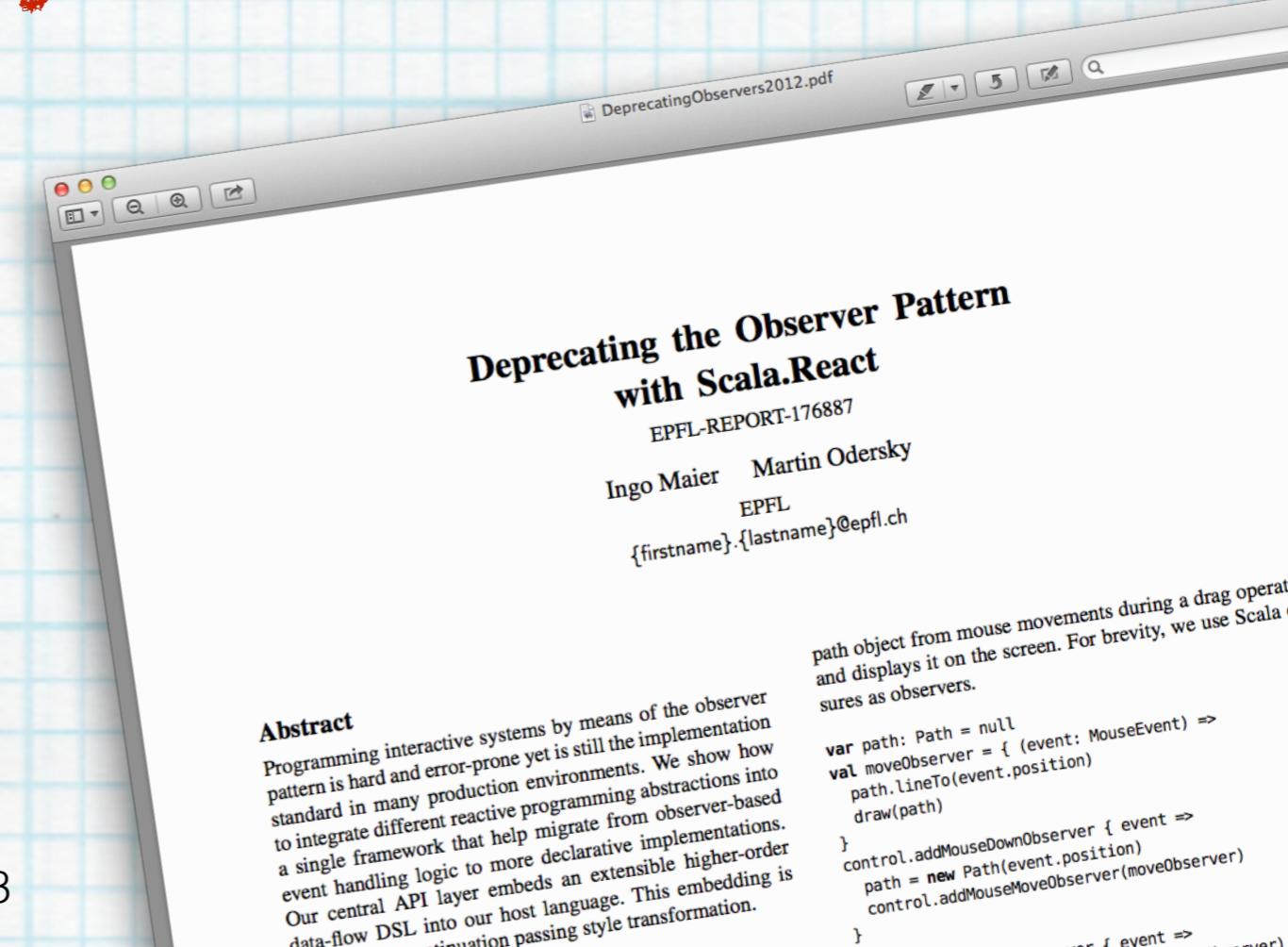
Contributions

- Observer Pattern
- Reactive Semantics
- Scala React
- Reactive Performance

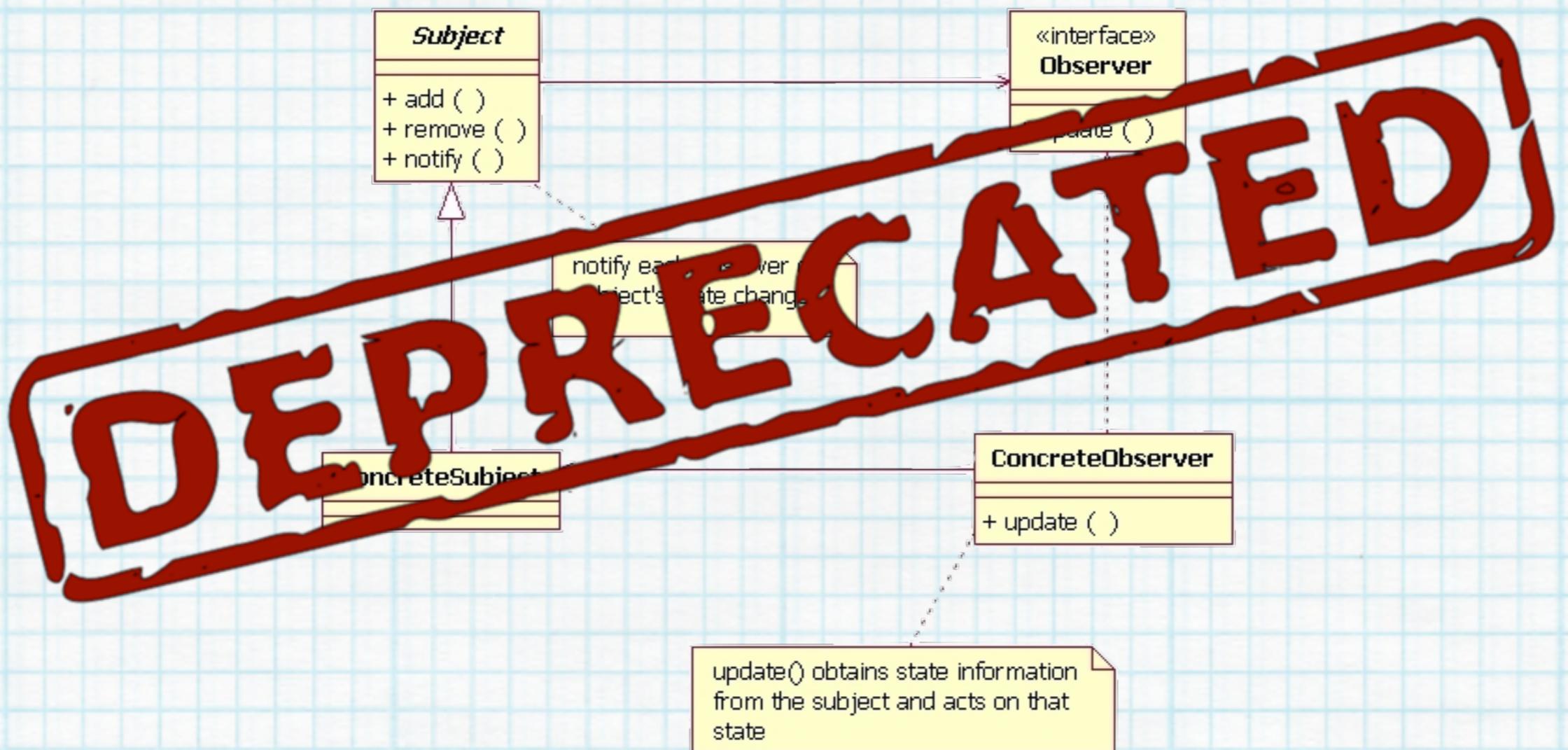
ok, maybe a little ;-)

Goals

- summary of RP
- applicability to NES



Observer Pattern





REALLY?

UPDATE WEEKENDUPDA

Observer Pattern

```
var path: Path = null
```

encapsulation

```
val moveObserver = { (event: MouseEvent) =>
    path.lineTo(event.position)
    draw(path)
}
```

abstraction

```
control.addMouseDownObserver { event =>
    path = new Path(event.position)
    control.addMouseMoveObserver(moveObserver)
}
```

uniformity

separation of
concerns

```
control.addMouseUpObserver { event =>
    control.removeMouseMoveObserver(moveObserver)
    path.closePath()
    draw(path)
}
```

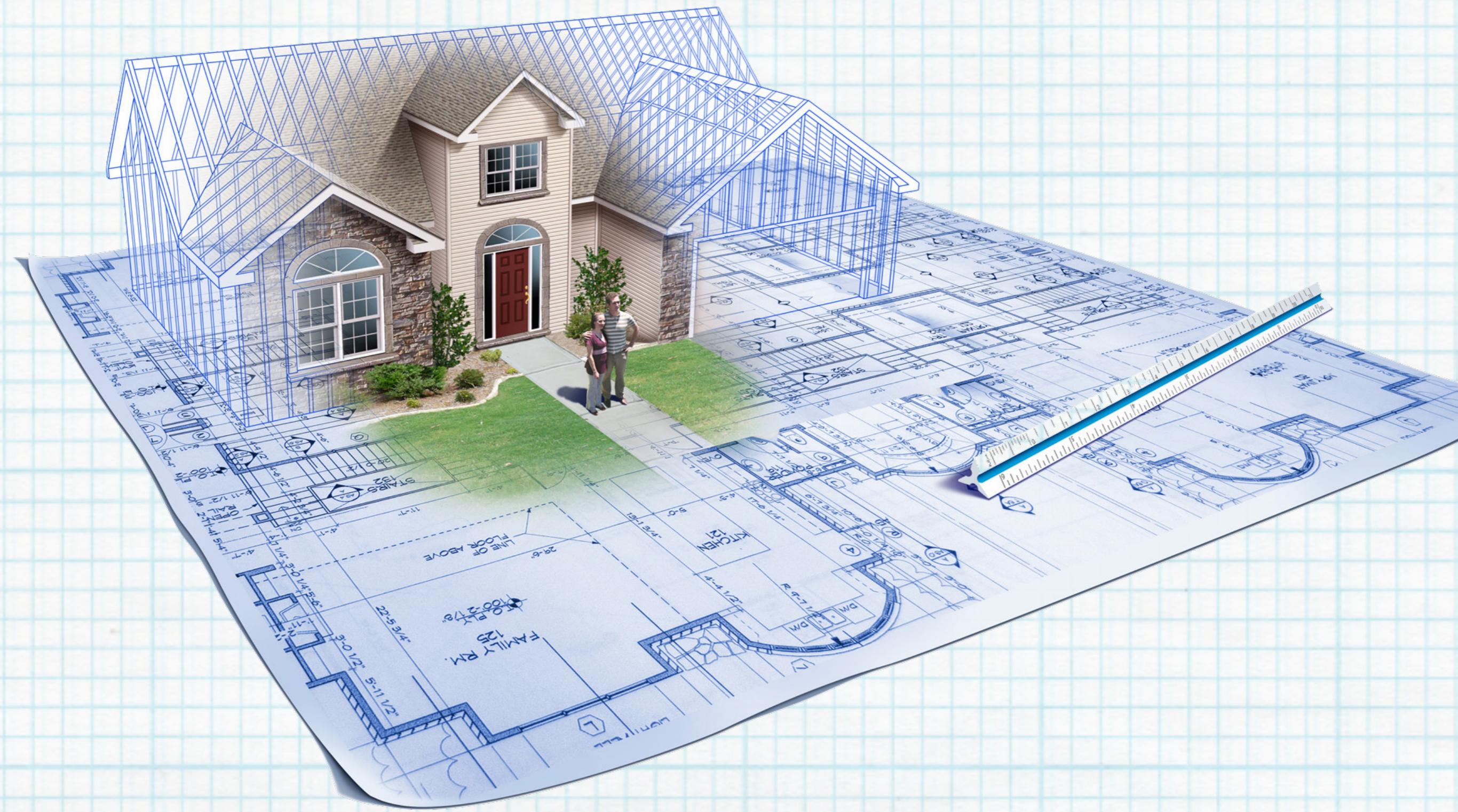
data
consistency

composability

semantic
distance

resource
management

side-effects



Reactive Semantics

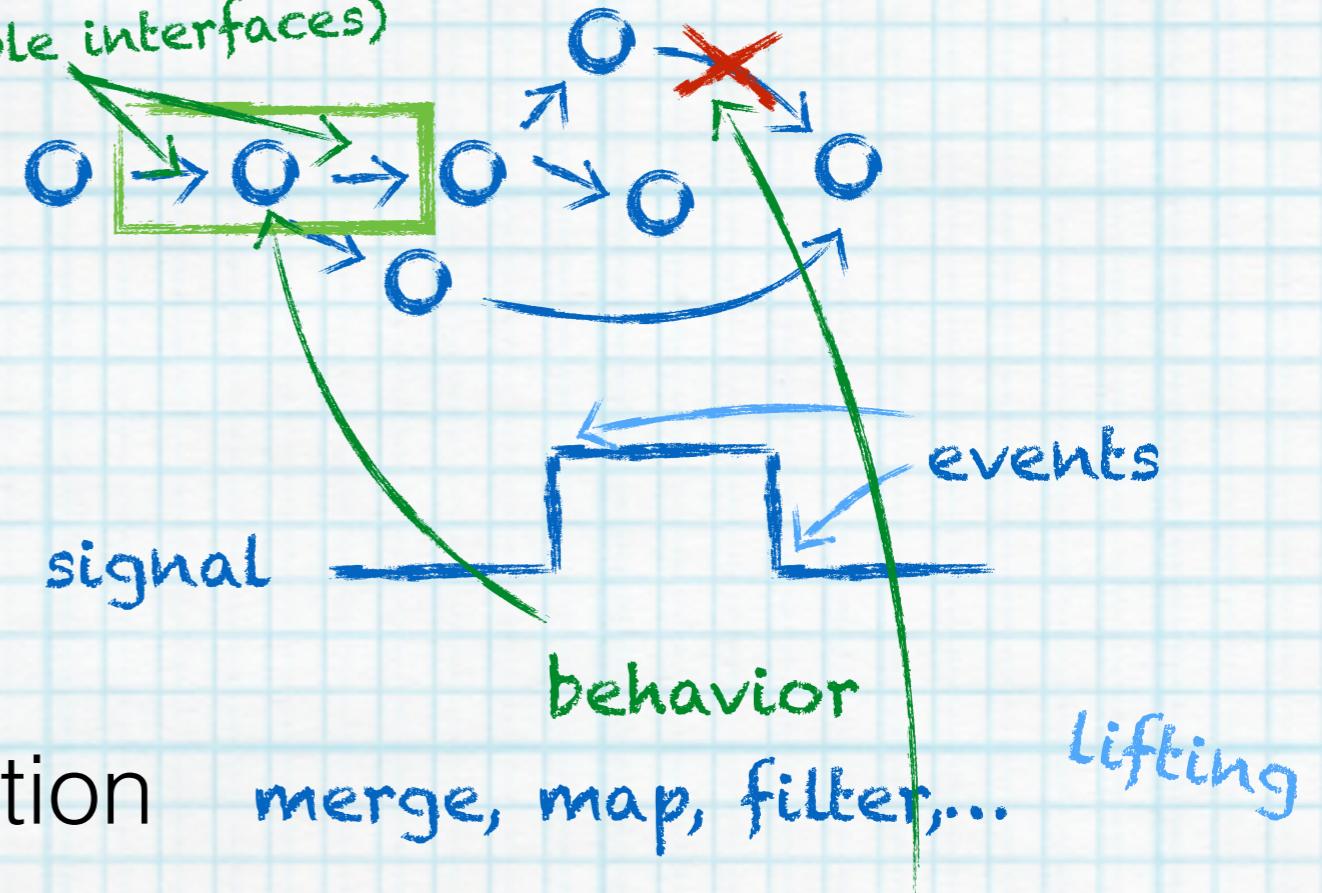
- Event Streams
- Events and Signals
- Composition/Transformation
- Reactors
- Data Flow Language

observable observers
(with simple interfaces)

dependency graph

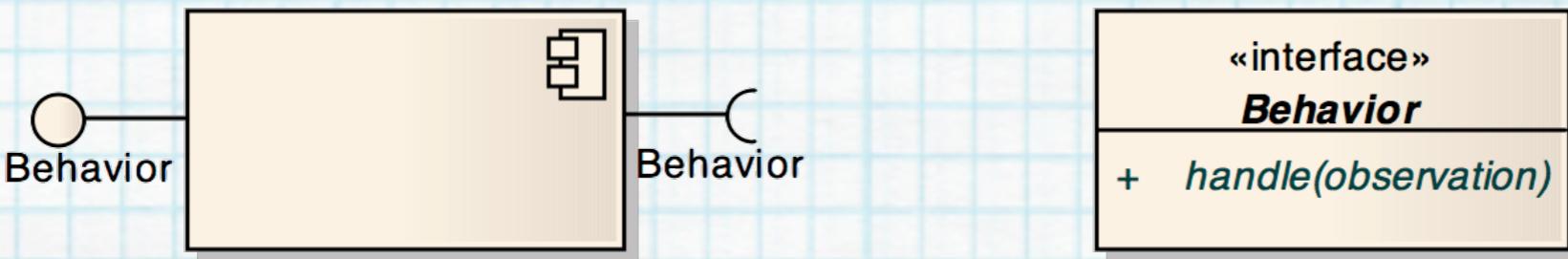
Resource Mgmt
Inversion of Control

pause, await, par, join, loop,...



scripting of
dynamic & structural
dependency graph
modifications

Reactive Semantics

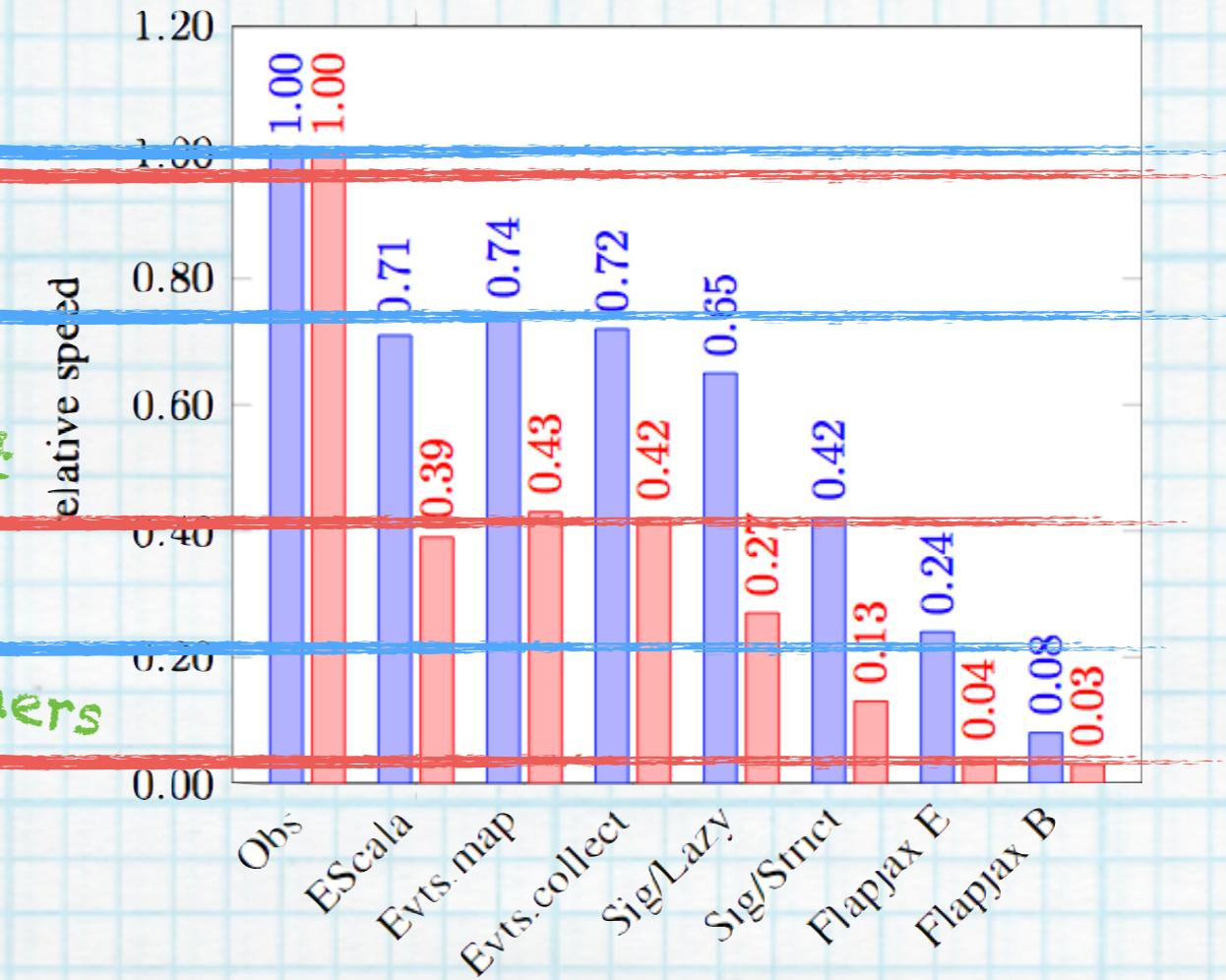
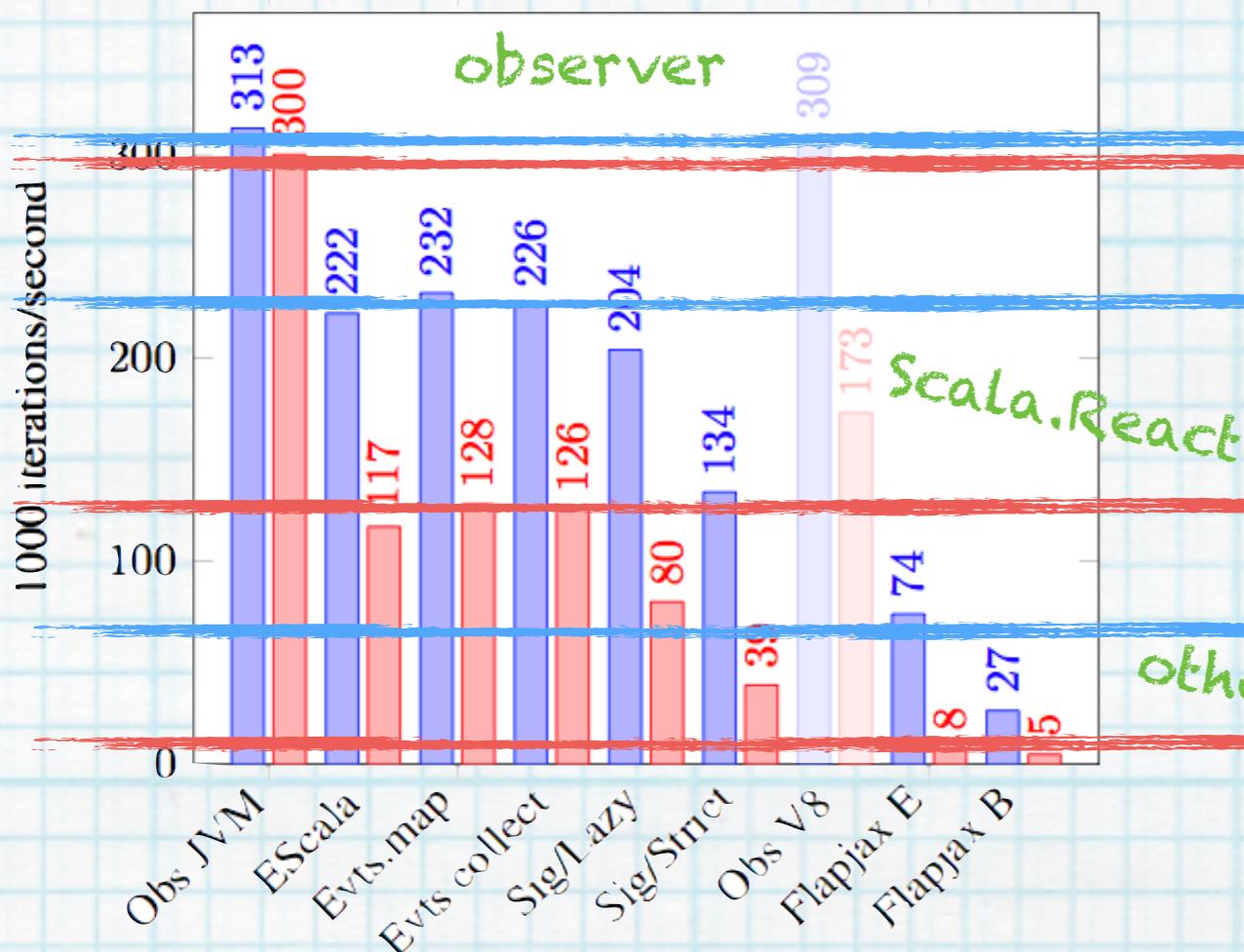
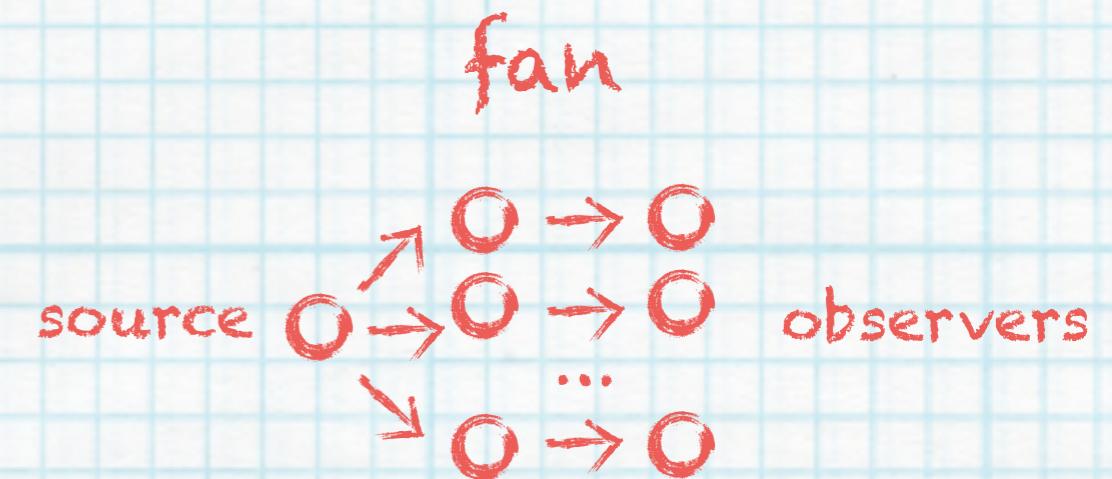


+ API for dynamic binding modifications

SLOW

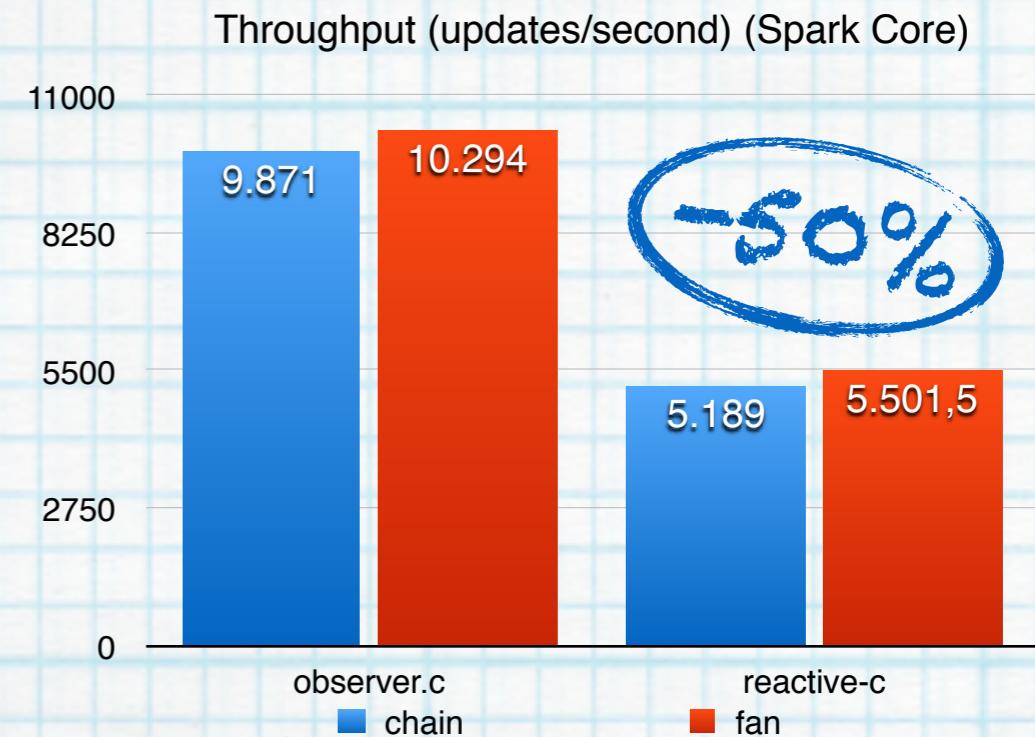
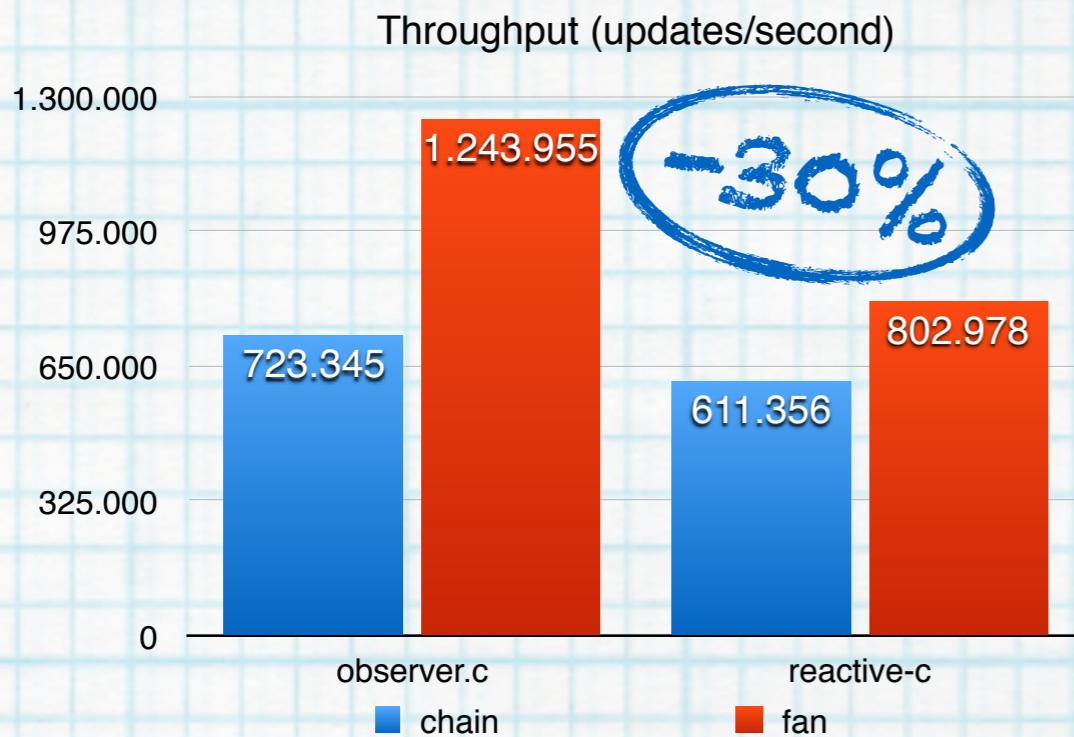


Reactive Performance



Does it Scale?

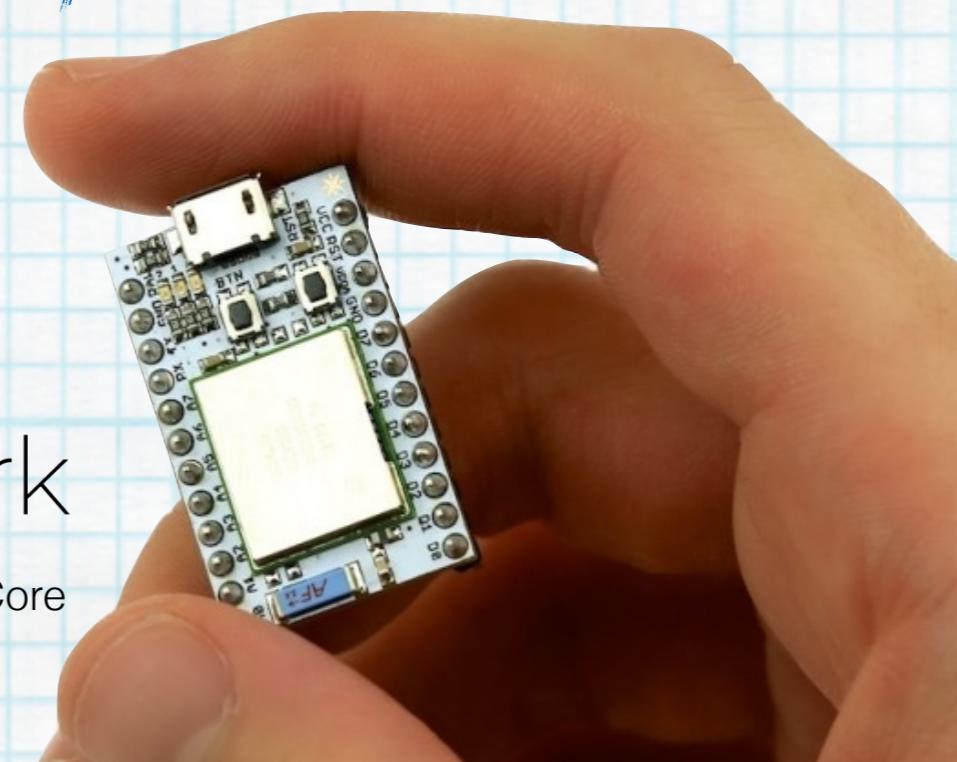
On embedded devices



Intel 2,4 GHz Intel Core 2 Duo
2530Mhz operating frequency

: 110

 spark
ARM 32-bit Cortex™-M3 CPU Core
72Mhz operating frequency





“Time moves in one direction,
memory in another.”

William Gibson

Shameless Plug

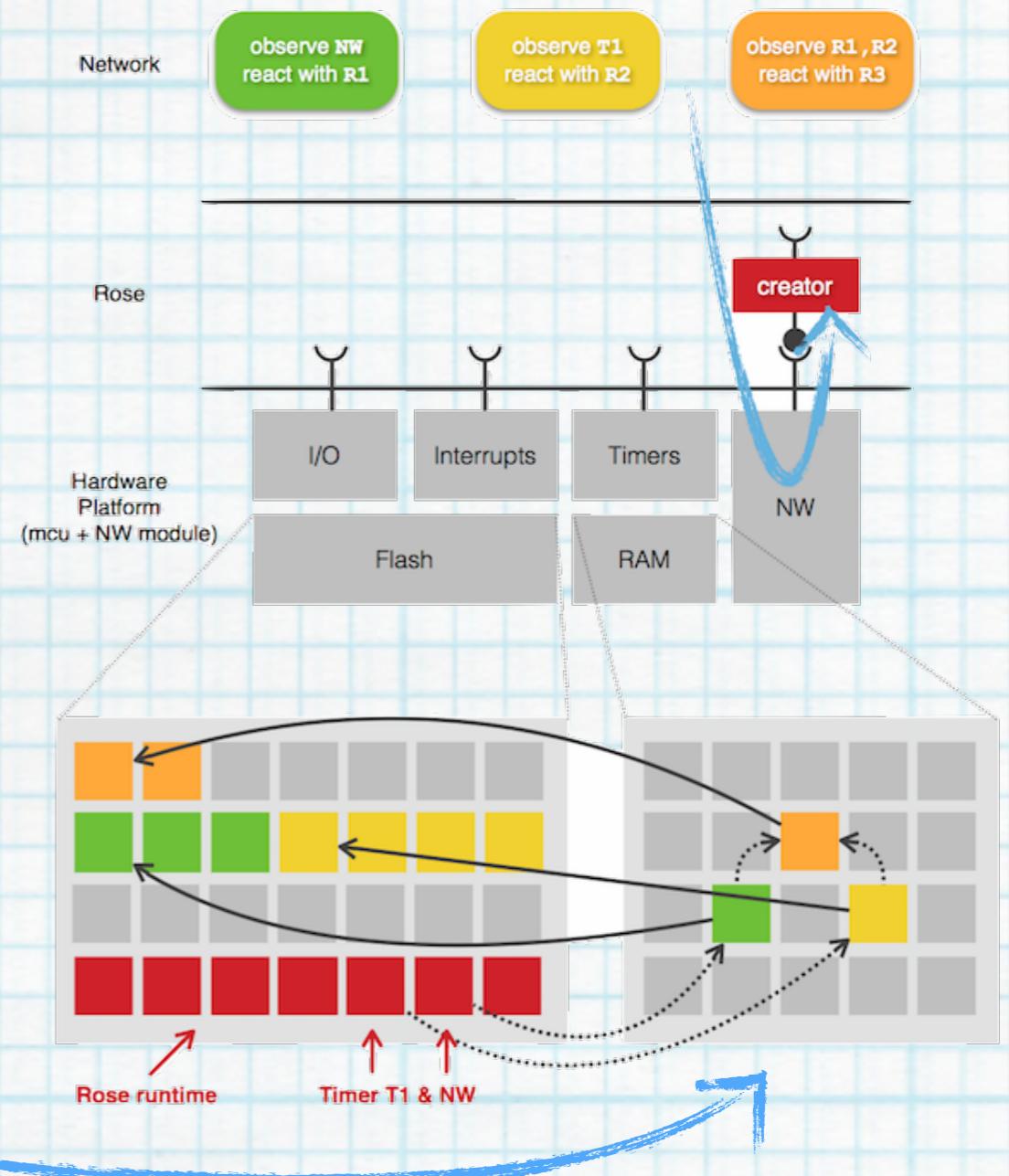
Reactive-C

Rose

untitled — reactive-c (git: master)

```

1 #include "reactive-c/api.h"
2
3 int main(void) {
4     int _a = 0, _b = 0, _c = 0;
5
6     observable_t a = observe(int, _a);
7     observable_t b = observe(int, _b);
8     observable_t c = observe(int, _c);
9
10    run(
11        script(
12            await(a),
13            await(b),
14            await(delayed(all(a, b, c))),
15            await(delayed(any(b, c))),
16            await(all(a, b, c))
17        )
18    );
19 }
20 
```



<https://github.com/christophevg/reactive-c>

<https://github.com/christophevg/rose>

Deprecating the Observer Pattern?

- Automate Dependency Graph Management
- Avoid Data Inconsistencies
- Offer a Functional API
- Allow for Imperative Encoding of Algorithms

Observer Pattern
on Steroids!



Reactions ?

PL@NES - Reading Club
Paper: **Deprecating the Observer
Pattern with Scala.React**
by Ingo Maier & Martin Odersky

Christophe.VanGinneken@cs.kuleuven.be

KU LEUVEN

DistriNet
Minds KU LEUVEN

Christophe.VanGinneken@cs.kuleuven.be

<http://www.slideshare.net/christophevg/deprecating-the-observer-pattern>