A Graph-Based Formal Semantics of Reactive Programming from First Principles

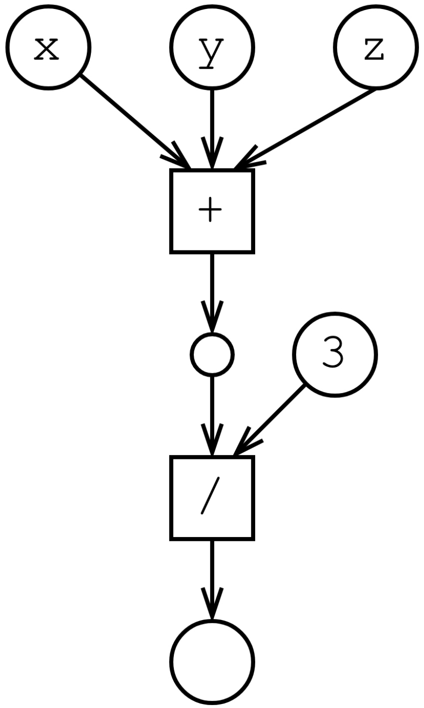
Bjarno Oeyen, Joeri De Koster, Wolfgang De Meuter

Software Languages Lab, Vrije Universiteit Brussel, Belgium

FTfJP Workshop @ Ecoop (07/06/2022)

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

<u>Higher</u>-Order

Sounds like...

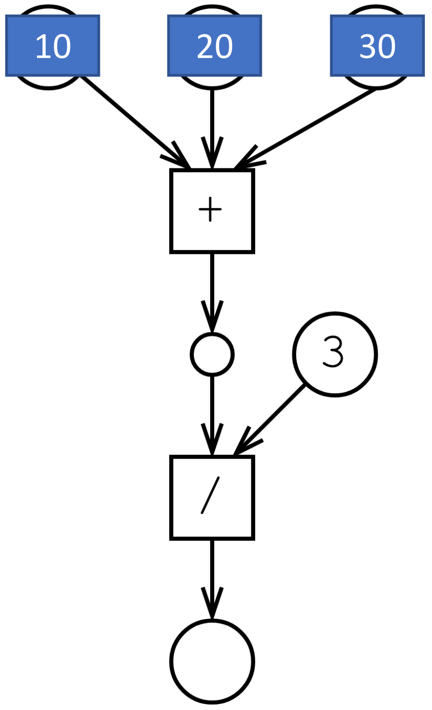
Haai

Is Dutch for...

Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

Higher-Order

Sounds like…

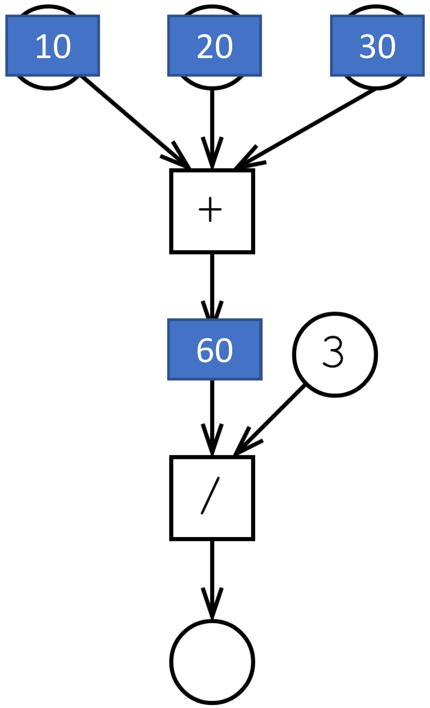
Haai

Is Dutch for…

Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

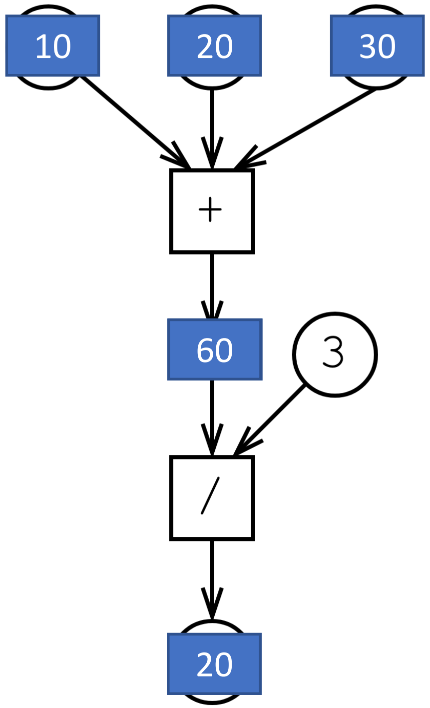


- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

Higher-Order
Sounds like...
Haai
Is Dutch for...
Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

Higher-Order

Sounds like...

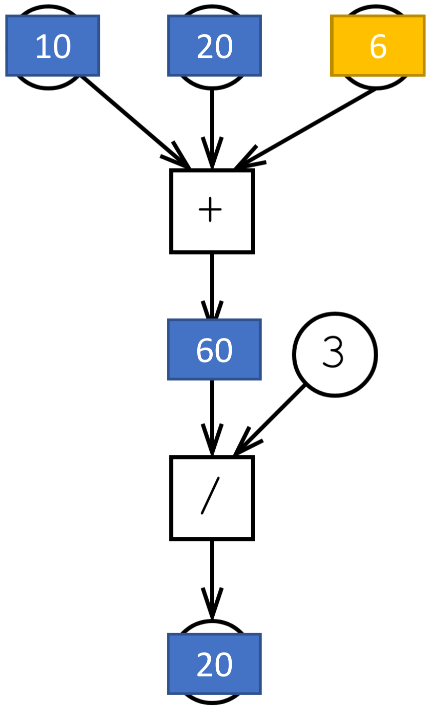
Haai

Is Dutch for...

Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

<u>Higher</u>-Order

Sounds like...

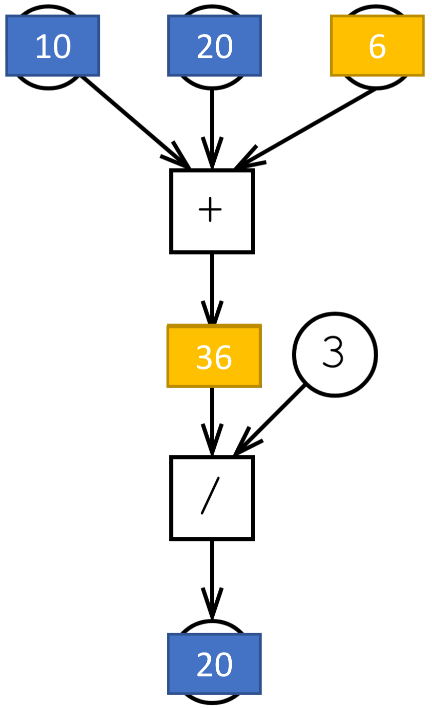
Haai

Is Dutch for...

Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

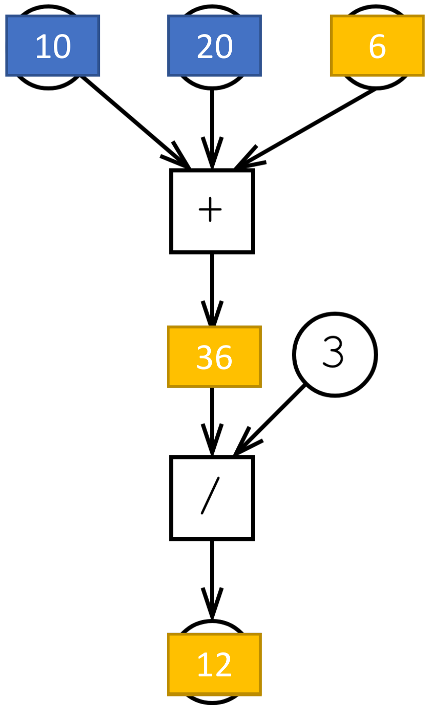


- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

Higher-Order
Sounds like...
Haai
Is Dutch for...
Shark

Reactive Programming in Haai

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



- Syntactically looks like Scheme code
  - Makes graphs
- Reactive semantics
  - Instead of *applying* functions, reactors are *deployed*.
  - Instances of reactors are called **deployments**
- **Push-based propagation**
  - Evaluation of RP programs in **turns**.
- Haai-specific characteristics
  - Everything is a reactor (even + and /)
  - Higher-Order Reactors

Higher-Order
Sounds like...
Haai
Is Dutch for...
Shark

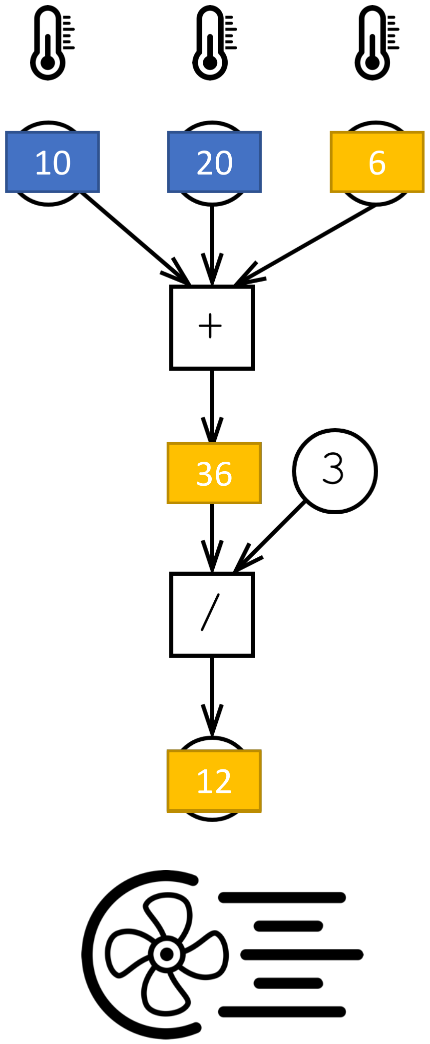
Reactive Programming in a nutshell

- **Automatic recomputation** of program state
  - By **declaring** constraints/dependencies between *signals*
  - **No callback hell** to keep data dependencies updated

- **Sources** bound to data producers
  - E.g., user input, sensors…

- **Sinks** bound to data consumers
  - E.g., actuators, user interface…

- Other "RP" languages:
  REScala, Frappé, FrTime, Elm, ReactiveX, Akka Streams…



- **Motivation**
- **Formalisation**
- **Lessons learned**

Two implementation styles of RP languages

- **Function-Based**
  Functions & Function Composition
  Examples: Fran, Yampa, SFRP, Dunai…
    - Usually implemented in Haskell…

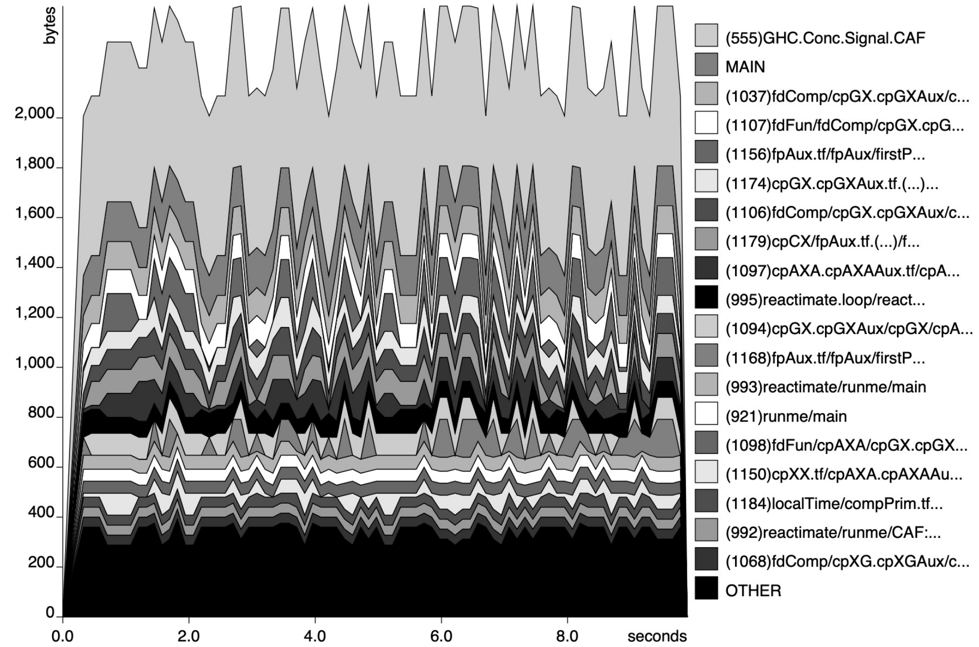
Well-studied
(∃ many formalisations of function-based RP)

- **Graph-Based**
  Graphs & Graph Composition
  Examples: FrTime, REScala, Frappé, EmFRP…

Not as well studied as function-based RP

So why bother formalising graph-based RP?

Memory Usage of a Yampa program



Memory usage of a simple Yampa program. Memory usage constantly fluctuates = GC needed.

- Haskell RP languages are, in general, unsuitable for embedded devices [*]
  - E.g., IoT, CPS, Real-Time Systems
- Risk of space leaks
- Need for a **garbage collector**
- …
- Which is usually also the case of their formalisations.

[*] Sawada, K., & Watanabe, T. (2016, March). Emfrp: a functional reactive programming language for small-scale embedded systems. In *Companion Proceedings of the 15th International Conference on Modularity* (pp. 36-44).

- Motivation
- **Formalisation**
- Lessons learned

Karcharias

$$
\begin{array}{rcl}
p \in \textbf{Program} & ::= & R \\
r \in R \subseteq \textbf{Reactor} & ::= & \mathcal{R}\langle x, N\rangle \\
n \in N \subseteq \textbf{Node} & ::= & (\bar{i}, nt, \bar{o}) \\
i \in \textbf{Input Port} & ::= & x \mid v \\
nt \in \textbf{Node Type} & ::= & \mathcal{RHO}\langle N\rangle \\
 & \mid & \mathcal{DEPLOY} \\
o \in \textbf{Output Port} & ::= & x \\
\multicolumn{3}{c}{x \in X \subseteq \textbf{Name}} \\
\multicolumn{3}{c}{\{in_{i,j}, out_{i,j} \mid \forall i \in \mathbb{N}^+, \forall j \in \mathbb{N}\} \subseteq X} \\
\multicolumn{3}{c}{v \in V \subseteq \textbf{Domain Value}}
\end{array}
$$

$$
\begin{array}{rcl}
k \in K \subseteq \textbf{Configuration} & ::= & \mathcal{K}\langle E, I_d, W, S, D\rangle \\
w \in W \subseteq \textbf{Deployment Wiring} & ::= & \mathcal{W}\langle \iota_d, N, \Sigma\rangle \\
s \in S \subseteq \textbf{Deployment Snapshot} & ::= & \mathcal{S}\langle \iota_d, \Sigma\rangle \\
c \in \textbf{Capture} & ::= & \mathcal{C}\langle \iota_c, N, \Sigma\rangle \\
\sigma \subseteq \textbf{Signal} & ::= & v \\
 & \mid & \mathcal{S}_{\mathcal{GLB}}\langle x\rangle \\
 & \mid & \mathcal{S}_{\mathcal{REF}}\langle \sigma, x\rangle \\
 & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \bar{\sigma}\rangle \\
i & ::= & \ldots \mid \sigma \\
v & ::= & \ldots \mid p \mid c \mid \iota_d \mid \bar{v} \\
E \subseteq \Sigma \subseteq \textbf{Value Environment} & ::= & \{x \mapsto v, \ldots\} \\
\Sigma \subseteq \textbf{Signal Environment} & ::= & \{x \mapsto \sigma, \ldots\} \\
D \subseteq \textbf{Toggle Environment} & ::= & \{(\iota_b, \iota_c) \mapsto \iota_d, \ldots\}
\end{array}
$$

$\iota_b \in I_b \subseteq$ **Branching Point Id**, $\iota_c \in I_c \subseteq$ **Capture Id**, $\iota_d \in I_d \subseteq$ **Deployment Id**

$p \in$ **Primitives**, $\delta_p : V* \to V*$ (for every p)

Local Wiring Rules ($\to_w$):

(W-REF)

$$\mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [x] \,\S\, \bar{i}_r, nt, \bar{o})\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [\Sigma(x)] \,\S\, \bar{i}_r, nt, \bar{o})\} \cup N, \Sigma\rangle$$

(W-DEPLOY)

$$\frac{x, \iota_b \text{ fresh} \qquad \Sigma' = \Sigma[x \mapsto \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \bar{\sigma}\rangle][o_i \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle, out_{i,0}\rangle \mid \forall i \in [1..|\bar{o}|]]}{\mathcal{W}\langle \iota_d, \{([\sigma] \,\S\, \bar{\sigma}, \mathcal{DEPLOY}, \bar{o})\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, N, \Sigma'\rangle}$$

(W-RHO)

$$\frac{\iota_c \text{ fresh} \qquad \Sigma_c = \text{shift\_io}(\Sigma) \qquad c = \mathcal{C}\langle \iota_c, N_{inner}, \Sigma_c\rangle}{\mathcal{W}\langle \iota_d, \{(\bar{\sigma}, \mathcal{RHO}\langle N_{inner}\rangle, [o])\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, N, \Sigma[o \mapsto c]\rangle}$$

Where $\text{shift\_io}(\Sigma) = \{x' \mapsto \sigma \mid \forall x \mapsto \sigma \in \Sigma, x' = \begin{cases} in_{i,j+1} & x = in_{i,j} \\ out_{i,j+1} & x = out_{i,j} \\ x & \text{otherwise} \end{cases}\}$

Global Rules ($\to_k$):

(W-CONGRUENCE)

$$\frac{w \to_w w'}{\mathcal{K}\langle E, I_d, \{w\} \uplus W, S, D\rangle \to_k \mathcal{K}\langle E, I_d, \{w'\} \cup W, S, D\rangle}$$

Evaluation Contexts:

$$
\begin{array}{rcl}
\mathcal{E} & ::= & \{x \mapsto \mathcal{E}_\sigma\} \uplus \Sigma \\
\mathcal{E}_\sigma & ::= & \square \\
 & \mid & \mathcal{S}_{\mathcal{REF}}\langle \mathcal{E}_\sigma, x\rangle \\
 & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{E}_\sigma, \bar{\sigma}\rangle \\
 & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \bar{\sigma} \,\S\, [\mathcal{E}_\sigma] \,\S\, \bar{\sigma}\rangle
\end{array}
$$

Local Snapshot Rules ($\to_s$):

(S-SELF-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle] \qquad v = \Sigma(x)}{\mathcal{S}\langle \iota_d, \Sigma\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[v]\rangle}$$

(S-DEPLOY-PRIMITIVE)

$$\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \bar{v}\rangle]\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[\delta_p(\bar{v})]\rangle$$

(S-TUPLE-REF)

$$\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \bar{v}, out_{(0,i)}\rangle]\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[v_i]\rangle$$

Global Rules ($\to_k$):

(S-CONGRUENCE)

$$\frac{s \to_s s'}{\mathcal{K}\langle E, I_d, W, \{s\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{s'\} \cup S, D\rangle}$$

(S-ACTIVATE)

$$\frac{\mathcal{W}\langle \iota_d, \varnothing, \Sigma\rangle \in W \qquad \iota_d \in I_d \qquad \forall s \in S : s = \mathcal{S}\langle \iota'_d, \Sigma'\rangle, \iota_d \neq \iota'_d}{\mathcal{K}\langle E, I_d, W, S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \cup S, D\rangle}$$

(S-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota'_d, x\rangle] \qquad v = \Sigma'(x) \qquad \Sigma'' = \mathcal{E}[v]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma''\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \cup S, D\rangle}$$

(S-GLOBAL)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{GLB}}\langle x\rangle] \qquad \Sigma' = \mathcal{E}[E(x)]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle}$$

(S-DEPLOY-NEW)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \bar{\sigma}\rangle] \qquad (\iota_b, \iota_c) \notin \text{dom}(D) \qquad \iota'_d \text{ fresh} \qquad w = \mathcal{W}\langle \iota'_d, N, \Sigma_c[in_{i,0} \mapsto \sigma_i \mid \forall i \in [1..|\bar{\sigma}|]]\rangle \qquad \Sigma' = \mathcal{E}[\iota'_d] \qquad D' = D[(\iota_b, \iota_c) \mapsto \iota'_d]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, \{w\} \cup W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D'\rangle}$$

(S-DEPLOY-EXISTING)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \bar{\sigma}\rangle] \qquad \iota'_d = D(\iota_b, \iota_c) \qquad \Sigma' = \mathcal{E}[\iota'_d]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle}$$

Given $p = \{\mathcal{R}\langle x_1, N_1\rangle, \ldots, \mathcal{R}\langle x_{|p|}, N_{|p|}\rangle\}$ ($|p|$ is the total number of user-defined reactor definitions):

$$
\begin{array}{rcl}
k_{init} & = & \mathcal{K}\langle E_{init}, \varnothing, W_{init}, \varnothing, \varnothing\rangle \\
E_{init} & = & \{x_i \mapsto \text{ntoc}(N_i) \mid \forall i \in [1..|p|]\} \cup \{time \mapsto 0, \ldots\} \\
W_{init} & = & \{\mathcal{W}\langle \iota_{d,main}, \{([main], \mathcal{DEPLOY}, [out_{(i,0)} \mid \forall i \in [1..|o|_{main}]])\}, \Sigma_{init}\rangle\} \\
\Sigma_{init} & = & \{+ \mapsto \ulcorner + \urcorner, \ldots\} \cup \{x_i \mapsto \mathcal{S}_{\mathcal{GLB}}\langle x_i\rangle \mid \forall i \in [1..|p|]\} \cup \{time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle time\rangle, \ldots\}
\end{array}
$$

Where $|o|_{main}$ is the number of outputs defined in the `main` reactor in p, and $\text{ntoc}(N) = \mathcal{C}\langle \iota_c, N, \Sigma_i nit\rangle$ (where ι_c fresh)

Intra-Turn Semantics

Semantic Entities:

$$
\begin{array}{rcl}
k^{inter} \in \textbf{Inter-Turn Configuration} & ::= & \mathcal{K}^{inter}\langle \tau, k\rangle \\
\tau \subseteq \textbf{Primitive Time-Varying Sources} & ::= & \{x \mapsto \bar{v}, \ldots\} \\
\widetilde{k} \in \widetilde{K} \subset K & \text{where} & \widetilde{k} \not\to_k
\end{array}
$$

Initial Configuration:

$$k^{inter}_{init} = \mathcal{K}^{inter}\langle \tau_{init}, k_{init}\rangle$$

$$\tau_{init} = \{time \mapsto [0, 1, 2, 3, \ldots], \ldots\}$$

Reduction Relation ($\rightsquigarrow$):

(INTRA-TURN)

$$\frac{k \to_k k'}{\mathcal{K}^{inter}\langle \tau, k\rangle \rightsquigarrow \mathcal{K}^{inter}\langle \tau, k'\rangle}$$

(NEXT-TURN)

$$\frac{k = \mathcal{K}\langle E, I_d, W, S, D\rangle \in \widetilde{K} \qquad \mathcal{S}\langle \iota_{main,d}, E_{main}\rangle \in S \qquad k' = \mathcal{K}\langle E[x_i \mapsto v_{i,now} \mid \forall i \in [1..|\tau|]], \{\iota_{main,d}\}, W, \varnothing, D\rangle}{\mathcal{K}^{inter}\langle \{x_i \mapsto [v_{i,now}] \,\S\, \bar{v}_i \mid \forall i \in [1..|\tau|]\}, k\rangle \rightsquigarrow \mathcal{K}^{inter}\langle \{x_i \mapsto \bar{v}_i \mid \forall i \in [1..|\tau|]\}, k'\rangle}$$

Inter-Turn Semantics

Higher-Order

Sounds like...

Haai

Is Dutch for...

Shark

In Greek...

Karcharias

Syntax

$$
\begin{array}{rcl}
p \in \textbf{Program} & ::= & R \\
r \in R \subseteq \textbf{Reactor} & ::= & \mathcal{R}\langle x, N\rangle \\
n \in N \subseteq \textbf{Node} & ::= & (\bar{i}, nt, \bar{o}) \\
i \in \textbf{Input Port} & ::= & x \mid v \\
nt \in \textbf{Node Type} & ::= & \mathcal{RHO}\langle N\rangle \\
 & \mid & \mathcal{DEPLOY} \\
o \in \textbf{Output Port} & ::= & x
\end{array}
$$

$$x \in X \subseteq \textbf{Name}$$

$$\{in_{i,j}, out_{i,j} \mid \forall i \in \mathbb{N}^{+}, \forall j \in \mathbb{N}\} \subseteq X$$

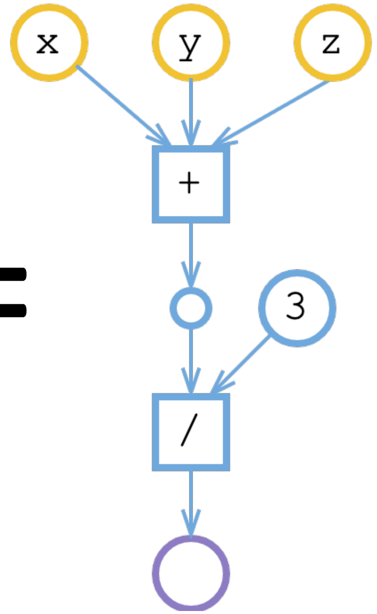
$$v \in V \subseteq \textbf{Domain Value}$$

Anonymous, nested, graph definitions with lexical scope.

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

=

Name: average



=

$$r_{avg} = \mathcal{R}\langle\ \text{average}, \{([+, in_{1,0}, in_{2,0}, in_{3,0}], \mathcal{DEPLOY}, [s_0]), ([/, s_0, 3], \mathcal{DEPLOY}, [out_{1,0}])\}\rangle$$

Special **input** names ($in_{i,j}$)

Special **output** names ($out_{i,j}$)

Haai Syntax **Graph Visualisation** **Karcharias Syntax**

Configurations

$$k ::= \mathcal{K}\langle E, I_d, W, S, D\rangle$$

Environment

Set of active deployment identifiers

Set of wirings

Set of snapshots

Toggle environment

Deployments

Wirings

$$w \in W \quad ::= \mathcal{W}\langle \iota_d, N, \Sigma\rangle$$

Deployment identifier

Node set

Signal environment

Snapshots

$$s \in S \quad ::= \mathcal{S}\langle \iota_d, \Sigma\rangle$$

Deployment identifier

Signal environment

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

```
(average sensor0 sensor1 sensor2)
```

Construct a *cascade* of reactor deployments (instances) to fully connect the average reactor.

Main Idea: Signal Environments

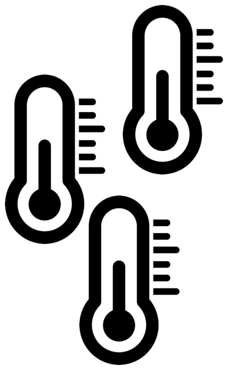
```
(defr (average x y z)
  (/ (+ x y z) 3))
```

```
(average sensor0 sensor1 sensor2)
```

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	
$in_{2,0}$	
$in_{3,0}$	
s_0	
$out_{1,0}$	

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

Main Idea: Signal Environments



```
(average sensor0 sensor1 sensor2)
```

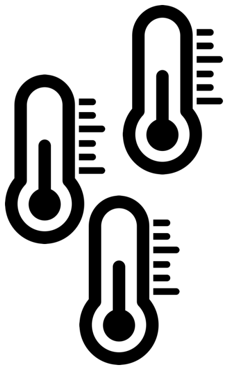
average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor2}\rangle$
s_0	
$out_{1,0}$	

Semantic entities that refer to global signals.

$$\sigma \subseteq \mathbf{Signal} \quad ::= \quad v \;\mid\; \mathcal{S}_{\mathcal{GLB}}\langle x\rangle \;\mid\; \mathcal{S}_{\mathcal{REF}}\langle \sigma, x\rangle \;\mid\; \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \overline{\sigma}\rangle$$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

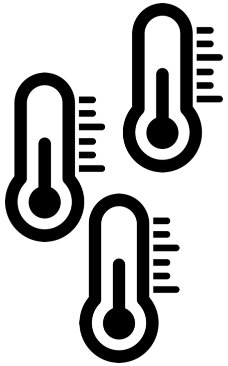


average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor2}\rangle$
s_0	
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	
$in_{2,0}$	
$in_{3,0}$	
$out_{1,0}$	

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

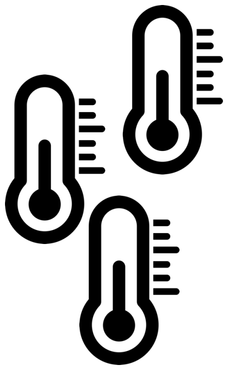


average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s_0	
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

Main Idea: Signal Environments

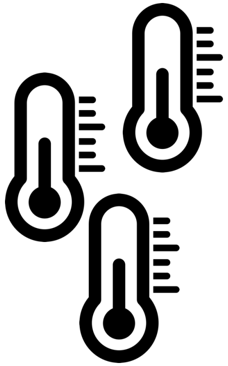


average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\mathrm{sensor2}\rangle$
s_0	
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{1,0}\rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle\iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

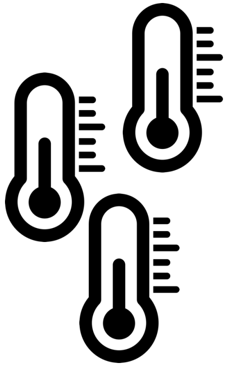


average ($\iota_{d,avg}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor0}\rangle$
$\text{in}_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor1}\rangle$
$\text{in}_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle\text{sensor2}\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
$\text{out}_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{1,0}\rangle$
$\text{in}_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{2,0}\rangle$
$\text{in}_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{3,0}\rangle$
$\text{out}_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle\iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



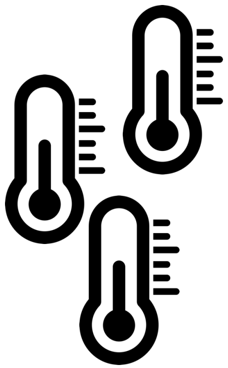
average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	
$in_{2,0}$	3
$out_{1,0}$	

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

Main Idea: Signal Environments



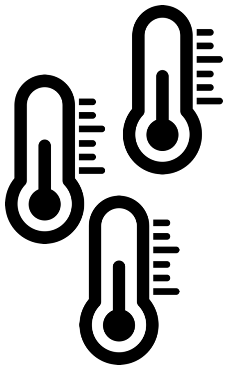
average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0 \rangle$
$in_{2,0}$	3
$out_{1,0}$	

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



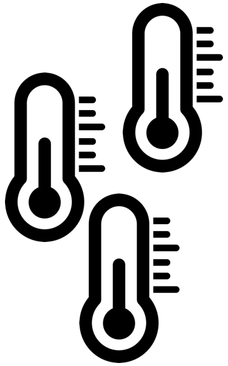
average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$out_{1,0}$	

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0 \rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots \rangle$

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

Main Idea: Signal Environments



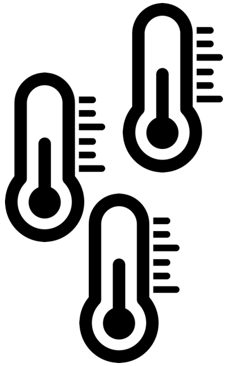
average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1 \rangle, out_{1,0} \rangle$

+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

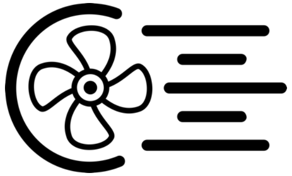
/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0 \rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots \rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1 \rangle, out_{1,0} \rangle$



+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0 \rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots \rangle$

Wiring (W-Rules)

- Connect signals between deployments
- Signal environment is populated step-by-step
  - Set of node

Local Wiring Rules ($\rightarrow_w$):

(W-REF)

$$\mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [x] \,\S\, \bar{i}_r, nt, \bar{o})\} \uplus N, \Sigma\rangle$$
$$\rightarrow_w \mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [\Sigma(x)] \,\S\, \bar{i}_r, nt, \bar{o})\} \cup N, \Sigma\rangle$$

(W-DEPLOY)

$$x, \iota_b \text{ fresh}$$
$$\Sigma' = \Sigma[x \mapsto \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \bar{o}\rangle][o_i \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle, out_{i,0}\rangle \mid \forall i \in [1..|\bar{o}|]]$$

$$N, \Sigma\rangle \rightarrow_w \mathcal{W}\langle \iota_d, N, \Sigma'\rangle$$

$$c = \mathcal{C}\langle \iota_c, N_{inner}, \Sigma_c\rangle$$

$$\Sigma\rangle \rightarrow_w \mathcal{W}\langle \iota_d, N, \Sigma[o \mapsto c]\rangle$$

$$\text{Where shift\_io}(\Sigma) = \{x' \mapsto \sigma \mid \forall x \mapsto \sigma \in \Sigma, x' = \begin{cases} in_{i,j+1} & x = in_{i,j} \\ out_{i,j+1} & x = out_{i,j} \\ x & \text{otherwise} \end{cases}\}$$

Global Rules ($\rightarrow_k$):

(W-CONGRUENCE)

$$\frac{w \rightarrow_w w'}{\mathcal{K}\langle E, I_d, \{w\} \uplus W, S, D\rangle \rightarrow_k \mathcal{K}\langle E, I_d, \{w'\} \cup W, S, D\rangle}$$

(W-REF)

$$\mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [x] \,\S\, \bar{i}_r, nt, \bar{o})\} \uplus N, \Sigma\rangle$$
$$\rightarrow_w \mathcal{W}\langle \iota_d, \{(\bar{i}_l \,\S\, [\Sigma(x)] \,\S\, \bar{i}_r, nt, \bar{o})\} \cup N, \Sigma\rangle$$

$$w \in W \ ::= \mathcal{W}\langle \iota_d, N, \Sigma\rangle$$

Deployment identifier

Node set

Signal environment

$$
\begin{aligned}
w_{avg} = \mathcal{W}\langle\ & \iota_{d,avg}, \\
& \{([+, in_{1,0}, in_{2,0}, in_{3,0}], \mathcal{DEPLOY}, [x]) \\
& \ ([/, x, 3], \mathcal{DEPLOY}, [out_{1,0}])\}, \\
& \{in_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \dots \rangle, \dots \\
& \ time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle \text{time} \rangle, \dots\}\rangle
\end{aligned}
$$

$$
\begin{aligned}
w'_{avg} = \mathcal{W}\langle\ & \iota_{d,avg}, \\
& \{([/, x, 3], \mathcal{DEPLOY}, [out_{1,0}])\}, \\
& \{x \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1 \rangle, out_{1,0} \rangle, \\
& \ in_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \dots \rangle, \dots \\
& \ time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle \text{time} \rangle, \dots\}\rangle
\end{aligned}
$$

$$
\begin{aligned}
w''_{avg} = \mathcal{W}\langle\ & \iota_{d,avg}, \\
& \{\}, \\
& \{out_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_2 \rangle, out_{1,0} \rangle, \\
& \ x \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1 \rangle, out_{1,0} \rangle, \\
& \ in_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \dots \rangle, \dots \\
& \ time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle \text{time} \rangle, \dots\}\rangle
\end{aligned}
$$

$$w \in W \ ::= \mathcal{W}\langle \iota_d, N, \Sigma \rangle$$

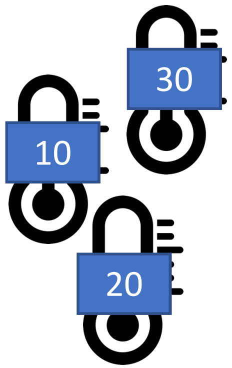
Deployment identifier

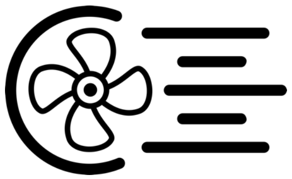
Node set

Signal environment

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```





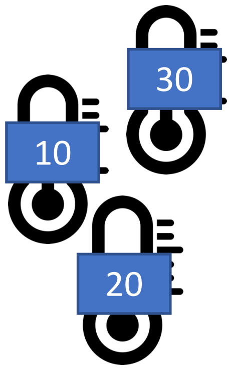
average ($\iota_{d,avg}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor0} \rangle$
$\mathrm{in}_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor1} \rangle$
$\mathrm{in}_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \mathrm{sensor2} \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1 \rangle, out_{1,0} \rangle$

+ ($\iota_{d,+}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$\mathrm{in}_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$\mathrm{in}_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

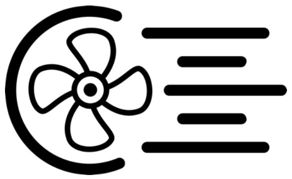
/ ($\iota_{d,/}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0 \rangle$
$\mathrm{in}_{2,0}$	3
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots \rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 ensor0$\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 ensor1$\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 ensor2$\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

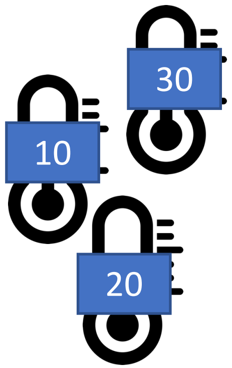


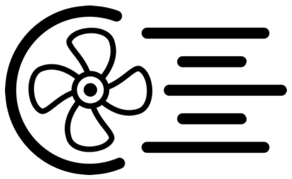
+ ($\iota_{d,+}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0}\rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0\rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots\rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```





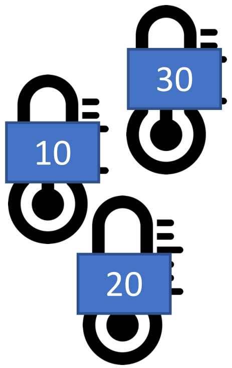
average ($\iota_{d,avg}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}$ 10 $\mathrm{ensor0}\rangle$
$\mathrm{in}_{2,0}$	$\mathcal{S}$ 20 $\mathrm{ensor1}\rangle$
$\mathrm{in}_{3,0}$	$\mathcal{S}$ 30 $\mathrm{ensor2}\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

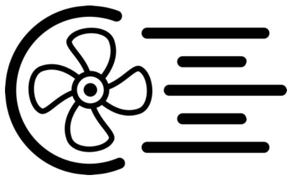
+ ($\iota_{d,+}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}$ 10 $avg, in_{1,0}\rangle$
$\mathrm{in}_{2,0}$	$\mathcal{S}$ 20 $avg, in_{2,0}\rangle$
$\mathrm{in}_{3,0}$	$\mathcal{S}$ 30 $avg, in_{3,0}\rangle$
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$\mathrm{in}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0\rangle$
$\mathrm{in}_{2,0}$	3
$\mathrm{out}_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots\rangle$

Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```





average ($\iota_{d,avg}$)

Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\text{ensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\text{ensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\text{ensor2}\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

+ ($\iota_{d,+}$)

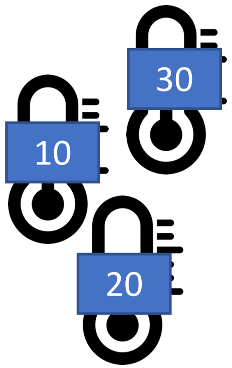
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $_{avg}, in_{1,0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $_{avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $_{avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}$ 60 $, \ulcorner + \urcorner, \ldots\rangle$

/ ($\iota_{d,/}$)

Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0\rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots\rangle$

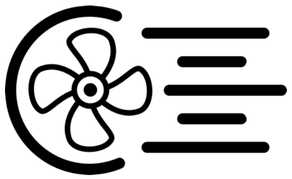
Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



average ($\iota_{d,avg}$)

Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\text{ensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\text{ensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\text{ensor2}\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}$ 60 $_{,avg}, d_0\rangle, out_{1,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$



+ ($\iota_{d,+}$)

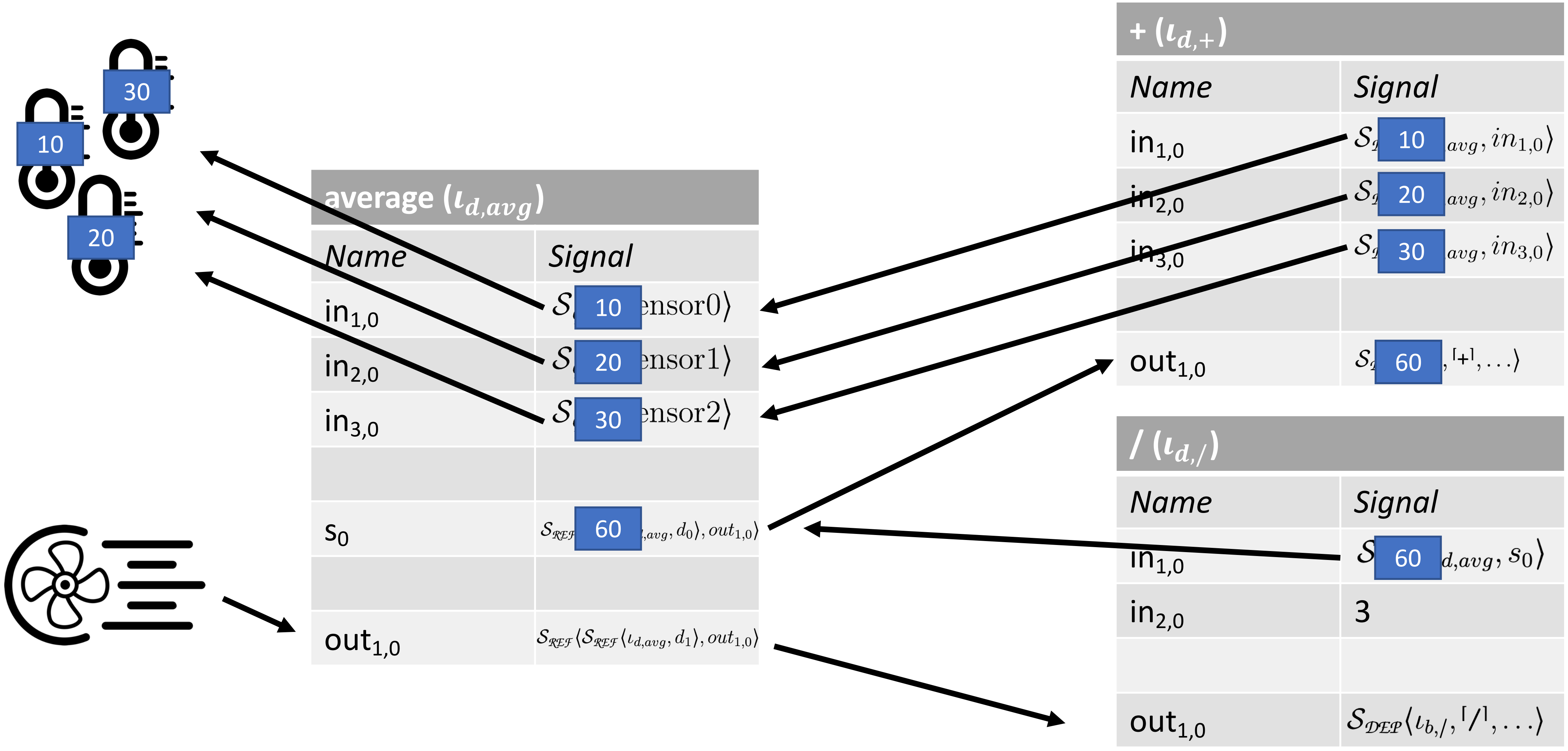
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $_{avg}, in_{1,0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $_{avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $_{avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}$ 60 $, \ulcorner + \urcorner, \ldots\rangle$

/ ($\iota_{d,/}$)

Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, s_0\rangle$
$in_{2,0}$	3
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,/}, \ulcorner / \urcorner, \ldots\rangle$

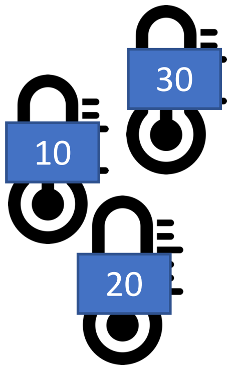
Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

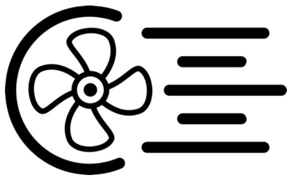


Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



average ($\iota_{d,avg}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 10 ensor0$\rangle$
$\text{in}_{2,0}$	$\mathcal{S}$ 20 ensor1$\rangle$
$\text{in}_{3,0}$	$\mathcal{S}$ 30 ensor2$\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}$ 60 $_{,avg}, d_0\rangle, out_{1,0}\rangle$
$\text{out}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

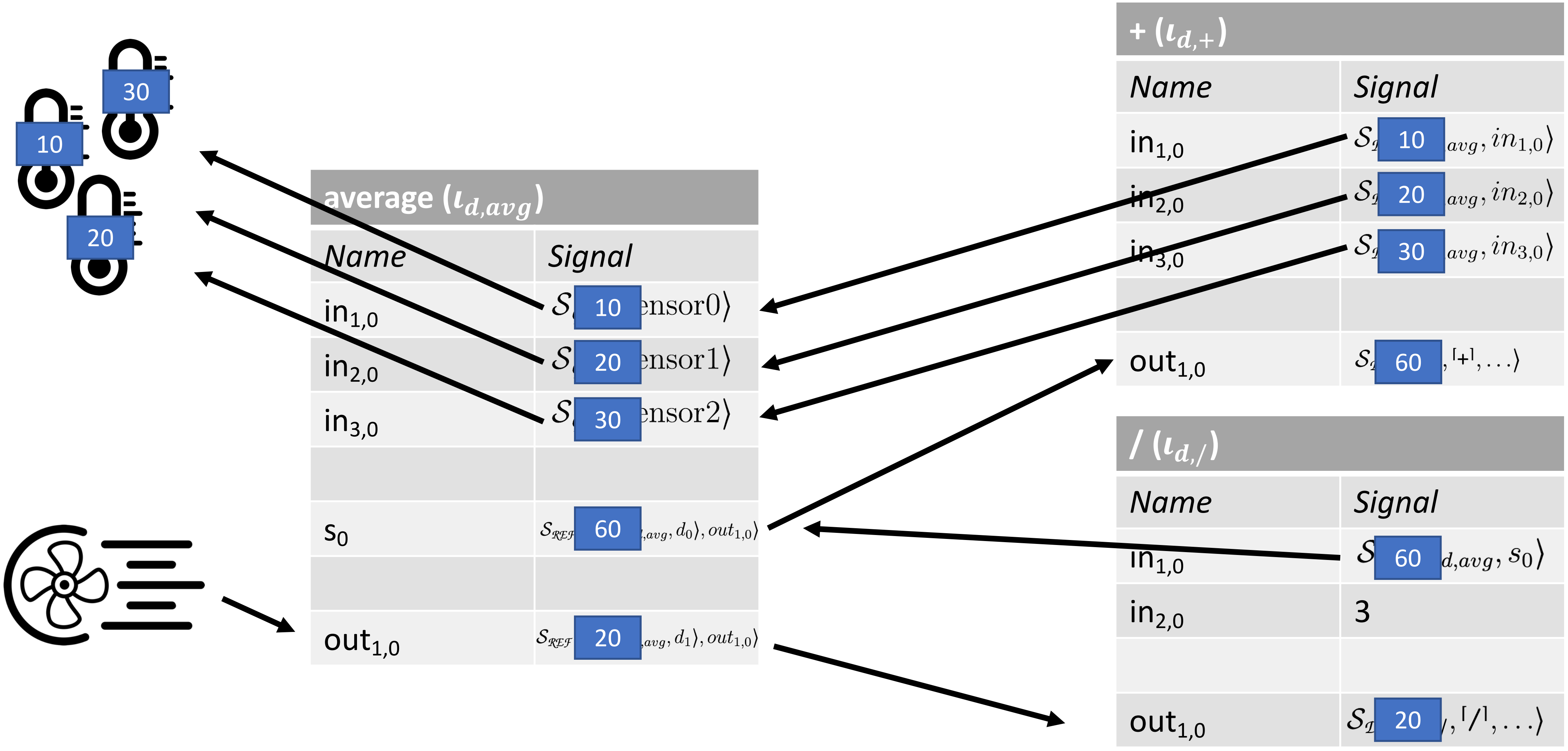


+ ($\iota_{d,+}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 10 $_{avg}, in_{1,0}\rangle$
$\text{in}_{2,0}$	$\mathcal{S}$ 20 $_{avg}, in_{2,0}\rangle$
$\text{in}_{3,0}$	$\mathcal{S}$ 30 $_{avg}, in_{3,0}\rangle$
$\text{out}_{1,0}$	$\mathcal{S}$ 60 $, \lceil + \rceil, \ldots\rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 60 $_{d,avg}, s_0\rangle$
$\text{in}_{2,0}$	3
$\text{out}_{1,0}$	$\mathcal{S}$ 20 $, \lceil / \rceil, \ldots\rangle$

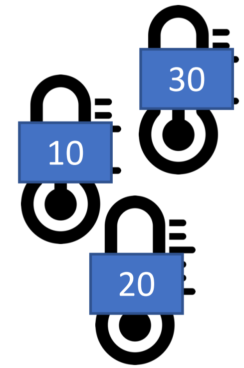
Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```

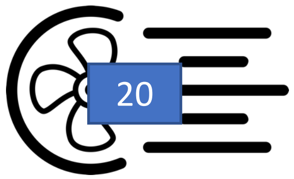


Main Idea: Signal Environments

```
(defr (average x y z)
  (/ (+ x y z) 3))
```



average ($\iota_{d,avg}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 10 $\text{ensor0}\rangle$
$\text{in}_{2,0}$	$\mathcal{S}$ 20 $\text{ensor1}\rangle$
$\text{in}_{3,0}$	$\mathcal{S}$ 30 $\text{ensor2}\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}$ 60 $_{,avg}, d_0\rangle, out_{1,0}\rangle$
$\text{out}_{1,0}$	$\mathcal{S}_{\mathcal{REF}}$ 20 $_{,avg}, d_1\rangle, out_{1,0}\rangle$



+ ($\iota_{d,+}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 10 $_{avg}, in_{1,0}\rangle$
$\text{in}_{2,0}$	$\mathcal{S}$ 20 $_{avg}, in_{2,0}\rangle$
$\text{in}_{3,0}$	$\mathcal{S}$ 30 $_{avg}, in_{3,0}\rangle$
$\text{out}_{1,0}$	$\mathcal{S}$ 60 $, \lceil + \rceil, \ldots\rangle$

/ ($\iota_{d,/}$)	
Name	*Signal*
$\text{in}_{1,0}$	$\mathcal{S}$ 60 $_{d,avg}, s_0\rangle$
$\text{in}_{2,0}$	3
$\text{out}_{1,0}$	$\mathcal{S}$ 20 $, \lceil / \rceil, \ldots\rangle$

Propagation (S-Rules)

- Reduce signal's into their current values.
- S-rules define the semantics of computing a **snapshot**.

$$s \in S \;::=\; \mathcal{S}\langle \iota_d, \Sigma \rangle$$

Deployment identifier

Signal environment

Evaluation Contexts:

$$\begin{array}{lcl} \mathcal{E} & ::= & \{x \mapsto \mathcal{E}_\sigma\} \uplus \Sigma \\ \mathcal{E}_\sigma & ::= & \square \\ & | & \mathcal{S}_{\mathcal{REF}}\langle \mathcal{E}_\sigma, x \rangle \\ & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{E}_\sigma, \overline{\sigma} \rangle \\ & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \overline{\sigma}\,\S\,[\mathcal{E}_\sigma]\,\S\,\overline{\sigma} \rangle \end{array}$$

Evaluation Contexts:

$$\begin{array}{lcl} \mathcal{E} & ::= & \{x \mapsto \mathcal{E}_\sigma\} \uplus \Sigma \\ \mathcal{E}_\sigma & ::= & \square \\ & | & \mathcal{S}_{\mathcal{REF}}\langle \mathcal{E}_\sigma, x \rangle \\ & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{E}_\sigma, \overline{\sigma} \rangle \\ & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \overline{\sigma}\,\S\,[\mathcal{E}_\sigma]\,\S\,\overline{\sigma} \rangle \end{array}$$

(S-GLOBAL)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{GLB}}\langle x \rangle] \qquad \Sigma' = \mathcal{E}[E(x)]}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma \rangle\} \uplus S, D \rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma' \rangle\} \cup S, D \rangle\end{array}}$$

$\mathcal{K}\langle E, I_d, W, \{s\} \uplus S, D\rangle \rightarrow_k \mathcal{K}\langle E, I_d, W, \{s'\} \cup S, D\rangle$

$\mathcal{K}\langle E, I_d, W, S, D\rangle \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \cup S, D\rangle$

(S-REF)

$\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \rightarrow_k \mathcal{K}\langle E, \{\iota_d'\} \cup I_d, \{w\} \cup W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D'\rangle$

(S-DEPLOY-EXISTING)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \overline{\sigma}\rangle] \qquad \iota_d' = D(\iota_b, \iota_c) \qquad \Sigma' = \mathcal{E}[\iota_d']}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, \{\iota_d'\} \cup I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle\end{array}}$$

$$
\begin{aligned}
s_{avg} = \mathcal{S}\langle\ & \iota_{d,avg}, \\
& \{out_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_2\rangle, out_{1,0}\rangle, \\
& \ x \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle, \\
& \ in_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \dots\rangle, \dots, \\
& \ time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle \text{time}\rangle, \dots\}\rangle
\end{aligned}
$$

$$
\begin{aligned}
s'_{avg} = \mathcal{S}\langle\ & \iota_{d,avg}, \\
& \{out_{1,0} \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_2\rangle, out_{1,0}\rangle, \\
& \ x \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle, \\
& \ in_{1,0} \mapsto 10, in_{2,0} \mapsto 20, in_{3,0} \mapsto 30, \\
& \ time \mapsto 1, \dots\}\rangle
\end{aligned}
$$

$$
\begin{aligned}
s''_{avg} = \mathcal{S}\langle\ & \iota_{d,avg}, \\
& \{out_{1,0} \mapsto 20, \\
& \ x \mapsto 60, \\
& \ in_{1,0} \mapsto 10, in_{2,0} \mapsto 20, in_{3,0} \mapsto 30, \\
& \ time \mapsto 1, \dots\}\rangle
\end{aligned}
$$

$$s \in S \ ::= \ \mathcal{S}\langle \underbrace{\iota_d}_{\text{Deployment identifier}}, \underbrace{\Sigma}_{\text{Signal environment}} \rangle$$

It's not that easy…

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

Sometimes +, sometimes *.
(e.g., `(if (even? time) + *)`)

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

Higher-Order Reactivity

average ($\iota_{d,avg}$)	
Name	*Signal*
in$_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
in$_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
in$_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
s$_0$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
…	…

Wiring decision is made at-propagation time, not at wiring time.

?

+ ($\iota_{d,+}$)	
Name	*Value*
in$_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
In$_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
in$_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
out$_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

* ($\iota_{d,*}$)	
Name	*Value*
in$_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
In$_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
in$_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
out$_{1,0}$	

Higher-Order Reactivity

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor0} \rangle$
$in_{2,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor1} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{GLB}}\langle \text{sensor2} \rangle$
d_0	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,d}, \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, \text{+-or-*} \rangle, \ldots \rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0 \rangle, out_{1,0} \rangle$
…	…

Meta signal with the deployment identifier (ι_d)

?

+ ($\iota_{d,+}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots \rangle$

* ($\iota_{d,*}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0} \rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0} \rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0} \rangle$
$out_{1,0}$	

Higher-Order Reactivity

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $sensor0\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $sensor1\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $sensor2\rangle$
d_0	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,d}, \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, \text{+-or-*}\rangle, \ldots\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
…	…

Meta signal with the deployment identifier (ι_d)

?

+ ($\iota_{d,+}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

* ($\iota_{d,*}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	

Higher-Order Reactivity

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\text{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\text{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\text{sensor2}\rangle$
d_0	$\mathcal{S}_{\mathcal{DE}}$ $\iota_{d,*}$ $\langle \iota_{d,avg}, \text{+-or-*}\rangle, \ldots\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
…	…

Meta signal with the deployment identifier (ι_d)

?

+ ($\iota_{d,+}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,+}, \ulcorner + \urcorner, \ldots\rangle$

* ($\iota_{d,*}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	

Higher-Order Reactivity

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

Deployment disabled (its output is not used in this turn)

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\mathrm{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\mathrm{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\mathrm{sensor2}\rangle$
d_0	$\mathcal{S}_{\mathcal{DE}}$ $\iota_{d,*}$ $\langle\iota_{d,avg}, \text{+-or-*}\rangle, \ldots\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
…	…

Meta signal with the deployment identifier (ι_d)

?

+ ($\iota_{d,+}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}_{\mathcal{REF}}\langle\iota_{d,avg}, in_{3,0}\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{DEP}}\langle\iota_{b,+}, `+`, \ldots\rangle$

* ($\iota_{d,*}$)	
Name	*Value*
$in_{1,0}$	$\mathcal{S}$ 10 $d,avg, in_{1,0}\rangle$
$In_{2,0}$	$\mathcal{S}$ 20 $d,avg, in_{2,0}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $d,avg, in_{3,0}\rangle$
$out_{1,0}$	

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

Higher-Order Reactivity

average ($\iota_{d,avg}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\mathrm{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\mathrm{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\mathrm{sensor2}\rangle$
d_0	$\mathcal{S}_{\mathcal{DEP}}\langle \iota_{b,d}, \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, \text{+-or-*}\rangle, \ldots\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
d_1	$\mathcal{S}_{\mathcal{DEP}}\langle \iota'_{b,d}, \mathcal{S}_{\mathcal{GLB}}\langle / \rangle, \ldots\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

What happens in the very first turn?

```
(defr (weird x y z)
  (def +or* …)
  (/ (+or* x y z) 3))
```

Higher-Order Reactivity

average ($\boldsymbol{\iota_{d,avg}}$)	
Name	*Signal*
$in_{1,0}$	$\mathcal{S}$ 10 $\mathrm{sensor0}\rangle$
$in_{2,0}$	$\mathcal{S}$ 20 $\mathrm{sensor1}\rangle$
$in_{3,0}$	$\mathcal{S}$ 30 $\mathrm{sensor2}\rangle$
d_0	$\mathcal{S}_{\mathcal{DE}}$ $\boldsymbol{\iota_{d,*}}$ $\langle \iota_{d,avg}, +\text{-or-}*\rangle, \ldots\rangle$
s_0	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_0\rangle, out_{1,0}\rangle$
d_1	$\mathcal{S}_{\mathcal{DEP}}\langle \iota'_{b,d}, \mathcal{S}_{\mathcal{GLB}}\langle / \rangle, \ldots\rangle$
$out_{1,0}$	$\mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_{d,avg}, d_1\rangle, out_{1,0}\rangle$

When “evaluating” the meta signal, the new deployment is created.

* ($\boldsymbol{\iota_{d,*}}$)	
Name	*Value*
$in_{1,0}$	
$In_{2,0}$	
$in_{3,0}$	
$out_{1,0}$	

W-Rules and S-Rules can be interleaved!

- Dynamically create new deployments for new captures (reactors) on the operator signals.

(W-DEPLOY)

$$\frac{x, \iota_b \text{ fresh} \qquad \Sigma' = \Sigma[x \mapsto \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \overline{\sigma}\rangle][o_i \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle, out_{i,0}\rangle \mid \forall i \in [1..|\overline{o}|]]}{\mathcal{W}\langle \iota_d, \{([\sigma] \S\, \overline{\sigma}, \mathcal{DEPLOY}, \overline{o})\} \uplus N, \Sigma\rangle \rightarrow_w \mathcal{W}\langle \iota_d, N, \Sigma'\rangle}$$

W-Rules and S-Rules can be interleaved!

- Dynamically create new deployments for new captures (reactors) on the operator signals.

Stored in the configuration.

Toggle Environment

$$D \;::=\; \{(\iota_b, \iota_c) \mapsto \iota_d, \ldots\}$$

Branching location identifier

Capture (reactor) identifier

Deployment identifier

Evaluation Contexts:

$$\begin{array}{lcl}
\mathcal{E} & ::= & \{x \mapsto \mathcal{E}_\sigma\} \uplus \Sigma \\
\mathcal{E}_\sigma & ::= & \square \\
 & | & \mathcal{S}_{\mathcal{REF}}\langle \mathcal{E}_\sigma, x\rangle \\
 & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{E}_\sigma, \overline{\sigma}\rangle \\
 & | & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \overline{\sigma} \,\S\, [\mathcal{E}_\sigma] \,\S\, \overline{\sigma}\rangle
\end{array}$$

(S-ACTIVATE)

$$\frac{\begin{array}{c}\mathcal{W}\langle \iota_d, \varnothing, \Sigma\rangle \in W \qquad \iota_d \in I_d \\ \forall s \in S : s = \mathcal{S}\langle \iota_d', \Sigma'\rangle, \iota_d \neq \iota_d'\end{array}}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \cup S, D\rangle\end{array}}$$

(S-DEPLOY-NEW)

$$\frac{\begin{array}{c}\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \overline{\sigma}\rangle] \qquad (\iota_b, \iota_c) \notin \mathrm{dom}(D) \qquad \iota_d' \text{ fresh} \\ w = \mathcal{W}\langle \iota_d', N, \Sigma_c[in_{i,0} \mapsto \sigma_i \mid \forall i \in [1..|\overline{\sigma}|]]\rangle \qquad \Sigma' = \mathcal{E}[\iota_d'] \\ D' = D[(\iota_b, \iota_c) \mapsto \iota_d']\end{array}}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, \{\iota_d'\} \cup I_d, \{w\} \cup W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D'\rangle\end{array}}$$

(S-DEPLOY-EXISTING)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \overline{\sigma}\rangle] \qquad \iota_d' = D(\iota_b, \iota_c) \qquad \Sigma' = \mathcal{E}[\iota_d']}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, \{\iota_d'\} \cup I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle\end{array}}$$

W-Rules and S-Rules can be interleaved!

- Dynamically create new deployments for new captures (reactors) on the operator signals.
- One reduction relation ($\rightarrow_k$)
  - Two helper relations ($\rightarrow_w$ and $\rightarrow_s$)
    - Both local and global wiring and snapshot rules.

Local Wiring Rules ($\rightarrow_w$):

(W-REF)

$$\begin{array}{l}\mathcal{W}\langle \iota_d, \{(\bar{i}_l \S [x] \S \bar{i}_r, nt, \overline{o})\} \uplus N, \Sigma\rangle \\ \rightarrow_w \mathcal{W}\langle \iota_d, \{(\bar{i}_l \S [\Sigma(x)] \S \bar{i}_r, nt, \overline{o})\} \cup N, \Sigma\rangle\end{array}$$

(W-DEPLOY)

$$\frac{\begin{array}{c}x, \iota_b \text{ fresh} \\ \Sigma' = \Sigma[x \mapsto \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \overline{\sigma}\rangle][o_i \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle, out_{i,0}\rangle \mid \forall i \in [1..|\overline{o}|]]\end{array}}{\mathcal{W}\langle \iota_d, \{([\sigma] \S \overline{\sigma}, \mathcal{DEPLOY}, \overline{o})\} \uplus N, \Sigma\rangle \rightarrow_w \mathcal{W}\langle \iota_d, N, \Sigma'\rangle}$$

(W-RHO)

$$\frac{\iota_c \text{ fresh} \qquad \Sigma_c = \text{shift\_io}(\Sigma) \qquad c = \mathcal{C}\langle \iota_c, N_{inner}, \Sigma_c\rangle}{\mathcal{W}\langle \iota_d, \{(\overline{\sigma}, \mathcal{RHO}\langle N_{inner}\rangle, [o])\} \uplus N, \Sigma\rangle \rightarrow_w \mathcal{W}\langle \iota_d, N, \Sigma[o \mapsto c]\rangle}$$

$$\text{Where shift\_io}(\Sigma) = \{x' \mapsto \sigma \mid \forall x \mapsto \sigma \in \Sigma, x' = \begin{cases} in_{i,j+1} & x = in_{i,j} \\ out_{i,j+1} & x = out_{i,j} \\ x & \text{otherwise} \end{cases}\}$$

Global Rules ($\rightarrow_k$):

(W-CONGRUENCE)

$$\frac{w \rightarrow_w w'}{\mathcal{K}\langle E, I_d, \{w\} \uplus W, S, D\rangle \rightarrow_k \mathcal{K}\langle E, I_d, \{w'\} \cup W, S, D\rangle}$$

Evaluation Contexts:

$$\begin{array}{lcl}\mathcal{E} & ::= & \{x \mapsto \mathcal{E}_\sigma\} \uplus \Sigma \\ \mathcal{E}_\sigma & ::= & \square \\ & \mid & \mathcal{S}_{\mathcal{REF}}\langle \mathcal{E}_\sigma, x\rangle \\ & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{E}_\sigma, \overline{\sigma}\rangle \\ & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \overline{\sigma} \S [\mathcal{E}_\sigma] \S \overline{\sigma}\rangle\end{array}$$

Local Snapshot Rules ($\rightarrow_s$):

(S-SELF-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle] \qquad v = \Sigma(x)}{\mathcal{S}\langle \iota_d, \Sigma\rangle \rightarrow_s \mathcal{S}\langle \iota_d, \mathcal{E}[v]\rangle}$$

(S-DEPLOY-PRIMITIVE)

$$\begin{array}{l}\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \overline{v}\rangle]\rangle \\ \rightarrow_s \mathcal{S}\langle \iota_d, \mathcal{E}[\delta_p(\overline{v})]\rangle\end{array}$$

(S-TUPLE-REF)

$$\begin{array}{l}\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \overline{v}, out_{(0,i)}\rangle]\rangle \\ \rightarrow_s \mathcal{S}\langle \iota_d, \mathcal{E}[v_i]\rangle\end{array}$$

Global Rules ($\rightarrow_k$):

(S-CONGRUENCE)

$$\frac{s \rightarrow_s s'}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{s\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{s'\} \cup S, D\rangle\end{array}}$$

(S-ACTIVATE)

$$\frac{\begin{array}{c}\mathcal{W}\langle \iota_d, \varnothing, \Sigma\rangle \in W \qquad \iota_d \in I_d \\ \forall s \in S : s = \mathcal{S}\langle \iota'_d, \Sigma'\rangle, \iota_d \neq \iota'_d\end{array}}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \cup S, D\rangle\end{array}}$$

(S-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota'_d, x\rangle] \qquad v = \Sigma'(x) \qquad \Sigma'' = \mathcal{E}[v]}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma''\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \cup S, D\rangle\end{array}}$$

(S-GLOBAL)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{GLB}}\langle x\rangle] \qquad \Sigma' = \mathcal{E}[E(x)]}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle\end{array}}$$

(S-DEPLOY-NEW)

$$\frac{\begin{array}{c}\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \overline{\sigma}\rangle] \qquad (\iota_b, \iota_c) \notin \text{dom}(D) \qquad \iota'_d \text{ fresh} \\ w = \mathcal{W}\langle \iota'_d, N, \Sigma_c[in_{i,0} \mapsto \sigma_i \mid \forall i \in [1..|\overline{\sigma}|]]\rangle \qquad \Sigma' = \mathcal{E}[\iota'_d] \\ D' = D[(\iota_b, \iota_c) \mapsto \iota'_d]\end{array}}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, \{w\} \cup W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D'\rangle\end{array}}$$

(S-DEPLOY-EXISTING)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \overline{\sigma}\rangle] \qquad \iota'_d = D(\iota_b, \iota_c) \qquad \Sigma' = \mathcal{E}[\iota'_d]}{\begin{array}{c}\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \\ \rightarrow_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle\end{array}}$$

Inter-Turn Semantics (Driver Loop)

Semantic Entities:

$k^{inter} \in$ **Inter-Turn Configuration** $::=$ $\mathcal{K}^{inter}\langle \tau, k \rangle$

$\tau \subseteq$ **Primitive Time-Varying Sources** $::=$ $\{x \mapsto \overline{v}, \ldots\}$

$\widetilde{k} \in \widetilde{K} \subset K$ where $\widetilde{k} \not\to_k$

Initial Co

$k^{inter}_{init} = \mathcal{K}^{inter}\langle$

$\tau_{init} = \{time \mapsto$

Reduction

(INTRA-TURN

$k \to$

$\mathcal{K}^{inter}\langle \tau, k \rangle \sim$

(NEXT-TURN)

$$\frac{\begin{array}{c} k = \mathcal{K}\langle E, I_d, W, S, D \rangle \in \widetilde{K} \\ \mathcal{S}\langle \iota_{main,d}, E_{main} \rangle \in S \\ k' = \mathcal{K}\langle E[x_i \mapsto v_{i,now} \mid \forall i \in [1..|\tau|]], \{\iota_{main,d}\}, W, \varnothing, D \rangle \end{array}}{\begin{array}{c} \mathcal{K}^{inter}\langle \{x_i \mapsto [v_{i,now}] \,\S\, \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k \rangle \\ \rightsquigarrow \mathcal{K}^{inter}\langle \{x_i \mapsto \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k' \rangle \end{array}}$$

(NEXT-TURN)

$$\frac{\begin{array}{c} k = \mathcal{K}\langle E, I_d, W, S, D \rangle \in \widetilde{K} \\ \mathcal{S}\langle \iota_{main,d}, E_{main} \rangle \in S \\ k' = \mathcal{K}\langle E[x_i \mapsto v_{i,now} \mid \forall i \in [1..|\tau|]], \{\iota_{main,d}\}, W, \varnothing, D \rangle \end{array}}{\begin{array}{c} \mathcal{K}^{inter}\langle \{x_i \mapsto [v_{i,now}] \,\S\, \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k \rangle \\ \rightsquigarrow \mathcal{K}^{inter}\langle \{x_i \mapsto \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k' \rangle \end{array}}$$

- Perform intra-turn steps “forever”

each turn

eter…

r loop of the

Karcharias

PLT Redex implementation of Karcharias can be found online:

https://gitlab.soft.vub.ac.be/boeyen/karcharias

$$
\begin{array}{rcl}
p \in \textbf{Program} & ::= & R \\
r \in R \subseteq \textbf{Reactor} & ::= & \mathcal{R}\langle x, N\rangle \\
n \in N \subseteq \textbf{Node} & ::= & (\bar{i}, nt, \bar{o}) \\
i \in \textbf{Input Port} & ::= & x \mid v \\
nt \in \textbf{Node Type} & ::= & \mathcal{RHO}\langle N\rangle \\
 & \mid & \mathcal{DEPLOY} \\
o \in \textbf{Output Port} & ::= & x
\end{array}
$$

$$x \in X \subseteq \textbf{Name}$$

$$\{in_{i,j}, out_{i,j} \mid \forall i \in \mathbb{N}^+, \forall j \in \mathbb{N}\} \subseteq X$$

$$v \in V \subseteq \textbf{Domain Value}$$

$$
\begin{array}{rcl}
k \in K \subseteq \textbf{Configuration} & ::= & \mathcal{K}\langle E, I_d, W, S, D\rangle \\
w \in W \subseteq \textbf{Deployment Wiring} & ::= & \mathcal{W}\langle \iota_d, N, \Sigma\rangle \\
s \in S \subseteq \textbf{Deployment Snapshot} & ::= & \mathcal{S}\langle \iota_d, \Sigma\rangle \\
c \in \textbf{Capture} & ::= & \mathcal{C}\langle \iota_c, N, \Sigma\rangle \\
\sigma \subseteq \textbf{Signal} & ::= & v \\
 & \mid & \mathcal{S}_{\mathcal{GLB}}\langle x\rangle \\
 & \mid & \mathcal{S}_{\mathcal{REF}}\langle \sigma, x\rangle \\
 & \mid & \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \bar{\sigma}\rangle \\
i & ::= & \ldots \mid \sigma \\
v & ::= & \ldots \mid p \mid c \mid \iota_d \mid \bar{v} \\
E \subseteq \Sigma \subseteq \textbf{Value Environment} & ::= & \{x \mapsto v, \ldots\} \\
\Sigma \subseteq \textbf{Signal Environment} & ::= & \{x \mapsto \sigma, \ldots\} \\
D \subseteq \textbf{Toggle Environment} & ::= & \{(\iota_b, \iota_c) \mapsto \iota_d, \ldots\}
\end{array}
$$

$\iota_b \in I_b \subseteq$ **Branching Point Id**, $\iota_c \in I_c \subseteq$ **Capture Id**, $\iota_d \in I_d \subseteq$ **Deployment Id**

$p \in$ **Primitives**, $\delta_p : V* \to V*$ (for every p)

Local Wiring Rules ($\to_w$):

(W-REF)

$$\mathcal{W}\langle \iota_d, \{(\bar{i}_l \S [x] \S \bar{i}_r, nt, \bar{o})\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, \{(\bar{i}_l \S [\Sigma(x)] \S \bar{i}_r, nt, \bar{o})\} \cup N, \Sigma\rangle$$

(W-DEPLOY)

$$\frac{x, \iota_b \text{ fresh} \quad \Sigma' = \Sigma[x \mapsto \mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \sigma, \bar{\sigma}\rangle][o_i \mapsto \mathcal{S}_{\mathcal{REF}}\langle \mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle, out_{i,0}\rangle \mid \forall i \in [1..|\bar{o}|]]}{\mathcal{W}\langle \iota_d, \{([\sigma] \S \bar{\sigma}, \mathcal{DEPLOY}, \bar{o})\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, N, \Sigma'\rangle}$$

(W-RHO)

$$\frac{\iota_c \text{ fresh} \qquad \Sigma_c = \text{shift\_io}(\Sigma) \qquad c = \mathcal{C}\langle \iota_c, N_{inner}, \Sigma_c\rangle}{\mathcal{W}\langle \iota_d, \{(\bar{\sigma}, \mathcal{RHO}\langle N_{inner}\rangle, [o])\} \uplus N, \Sigma\rangle \to_w \mathcal{W}\langle \iota_d, N, \Sigma[o \mapsto c]\rangle}$$

$$\text{Where shift\_io}(\Sigma) = \{x' \mapsto \sigma \mid \forall x \mapsto \sigma \in \Sigma, x' = \begin{cases} in_{i,j+1} & x = in_{i,j} \\ out_{i,j+1} & x = out_{i,j} \\ x & \text{otherwise} \end{cases}\}$$

Global Rules ($\to_k$):

(W-CONGRUENCE)

$$\frac{w \to_w w'}{\mathcal{K}\langle E, I_d, \{w\} \uplus W, S, D\rangle \to_k \mathcal{K}\langle E, I_d, \{w'\} \cup W, S, D\rangle}$$

Local Snapshot Rules ($\to_s$):

(S-SELF-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota_d, x\rangle] \qquad v = \Sigma(x)}{\mathcal{S}\langle \iota_d, \Sigma\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[v]\rangle}$$

(S-DEPLOY-PRIMITIVE)

$$\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, p, \bar{v}\rangle]\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[\delta_p(\bar{v})]\rangle$$

(S-TUPLE-REF)

$$\mathcal{S}\langle \iota_d, \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \bar{v}, out_{(0,i)}\rangle]\rangle \to_s \mathcal{S}\langle \iota_d, \mathcal{E}[v_i]\rangle$$

Global Rules ($\to_k$):

(S-CONGRUENCE)

$$\frac{s \to_s s'}{\mathcal{K}\langle E, I_d, W, \{s\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{s'\} \cup S, D\rangle}$$

(S-ACTIVATE)

$$\frac{\mathcal{W}\langle \iota_d, \varnothing, \Sigma\rangle \in W \qquad \iota_d \in I_d \qquad \forall s \in S : s = \mathcal{S}\langle \iota'_d, \Sigma'\rangle, \iota_d \neq \iota'_d}{\mathcal{K}\langle E, I_d, W, S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \cup S, D\rangle}$$

(S-REF)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{REF}}\langle \iota'_d, x\rangle] \qquad v = \Sigma'(x) \qquad \Sigma'' = \mathcal{E}[v]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma''\rangle, \mathcal{S}\langle \iota'_d, \Sigma'\rangle\} \cup S, D\rangle}$$

(S-GLOBAL)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{GLB}}\langle x\rangle] \qquad \Sigma' = \mathcal{E}[E(x)]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle}$$

(S-DEPLOY-NEW)

$$\frac{\begin{array}{c}\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \bar{\sigma}\rangle] \qquad (\iota_b, \iota_c) \notin \text{dom}(D) \qquad \iota'_d \text{ fresh} \\ w = \mathcal{W}\langle \iota'_d, N, \Sigma_c[in_{i,0} \mapsto \sigma_i \mid \forall i \in [1..|\bar{\sigma}|]]\rangle \qquad \Sigma' = \mathcal{E}[\iota'_d] \\ D' = D[(\iota_b, \iota_c) \mapsto \iota'_d]\end{array}}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, \{w\} \cup W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D'\rangle}$$

(S-DEPLOY-EXISTING)

$$\frac{\Sigma = \mathcal{E}[\mathcal{S}_{\mathcal{DEP}}\langle \iota_b, \mathcal{C}\langle \iota_c, N, \Sigma_c\rangle, \bar{\sigma}\rangle] \qquad \iota'_d = D(\iota_b, \iota_c) \qquad \Sigma' = \mathcal{E}[\iota'_d]}{\mathcal{K}\langle E, I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma\rangle\} \uplus S, D\rangle \to_k \mathcal{K}\langle E, \{\iota'_d\} \cup I_d, W, \{\mathcal{S}\langle \iota_d, \Sigma'\rangle\} \cup S, D\rangle}$$

Given $p = \{\mathcal{R}\langle x_1, N_1\rangle, \ldots, \mathcal{R}\langle x_{|p|}, N_{|p|}\rangle\}$ ($|p|$ is the total number of user-defined reactor definitions):

$$
\begin{array}{rcl}
k_{init} & = & \mathcal{K}\langle E_{init}, \varnothing, W_{init}, \varnothing, \varnothing\rangle \\
E_{init} & = & \{x_i \mapsto \text{ntoc}(N_i) \mid \forall i \in [1..|p|]\} \cup \{time \mapsto 0, \ldots\} \\
W_{init} & = & \{\mathcal{W}\langle \iota_{d,main}, \{([main], \mathcal{DEPLOY}, [out_{(i,0)} \mid \forall i \in [1..|o|_{main}]])\}, \Sigma_{init}\rangle\} \\
\Sigma_{init} & = & \{+ \mapsto \ulcorner + \urcorner, \ldots\} \cup \{x_i \mapsto \mathcal{S}_{\mathcal{GLB}}\langle x_i\rangle \mid \forall i \in [1..|p|]\} \cup \{time \mapsto \mathcal{S}_{\mathcal{GLB}}\langle time\rangle, \ldots\}
\end{array}
$$

Where $|o|_{main}$ is the number of outputs defined in the `main` reactor in p, and $\text{ntoc}(N) = \mathcal{C}\langle \iota_c, N, \Sigma_i nit\rangle$ (where ι_c fresh)

Semantic Entities:

$$
\begin{array}{rcl}
k^{inter} \in \textbf{Inter-Turn Configuration} & ::= & \mathcal{K}^{inter}\langle \tau, k\rangle \\
\tau \subseteq \textbf{Primitive Time-Varying Sources} & ::= & \{x \mapsto \bar{v}, \ldots\} \\
\widetilde{k} \in \widetilde{K} \subset K & \text{where} & \widetilde{k} \not\to_k
\end{array}
$$

Initial Configuration:

$$k^{inter}_{init} = \mathcal{K}^{inter}\langle \tau_{init}, k_{init}\rangle$$

$$\tau_{init} = \{time \mapsto [0, 1, 2, 3, \ldots], \ \ldots\}$$

Reduction Relation ($\rightsquigarrow$):

(INTRA-TURN)

$$\frac{k \to_k k'}{\mathcal{K}^{inter}\langle \tau, k\rangle \rightsquigarrow \mathcal{K}^{inter}\langle \tau, k'\rangle}$$

(NEXT-TURN)

$$\frac{\begin{array}{c}k = \mathcal{K}\langle E, I_d, W, S, D\rangle \in \widetilde{K} \\ \mathcal{S}\langle \iota_{main,d}, E_{main}\rangle \in S \\ k' = \mathcal{K}\langle E[x_i \mapsto v_{i,now} \mid \forall i \in [1..|\tau|]], \{\iota_{main,d}\}, W, \varnothing, D\rangle\end{array}}{\mathcal{K}^{inter}\langle \{x_i \mapsto [v_{i,now}] \S \bar{v}_i \mid \forall i \in [1..|\tau|]\}, k\rangle \rightsquigarrow \mathcal{K}^{inter}\langle \{x_i \mapsto \bar{v}_i \mid \forall i \in [1..|\tau|]\}, k'\rangle}$$

Intra-Turn Semantics

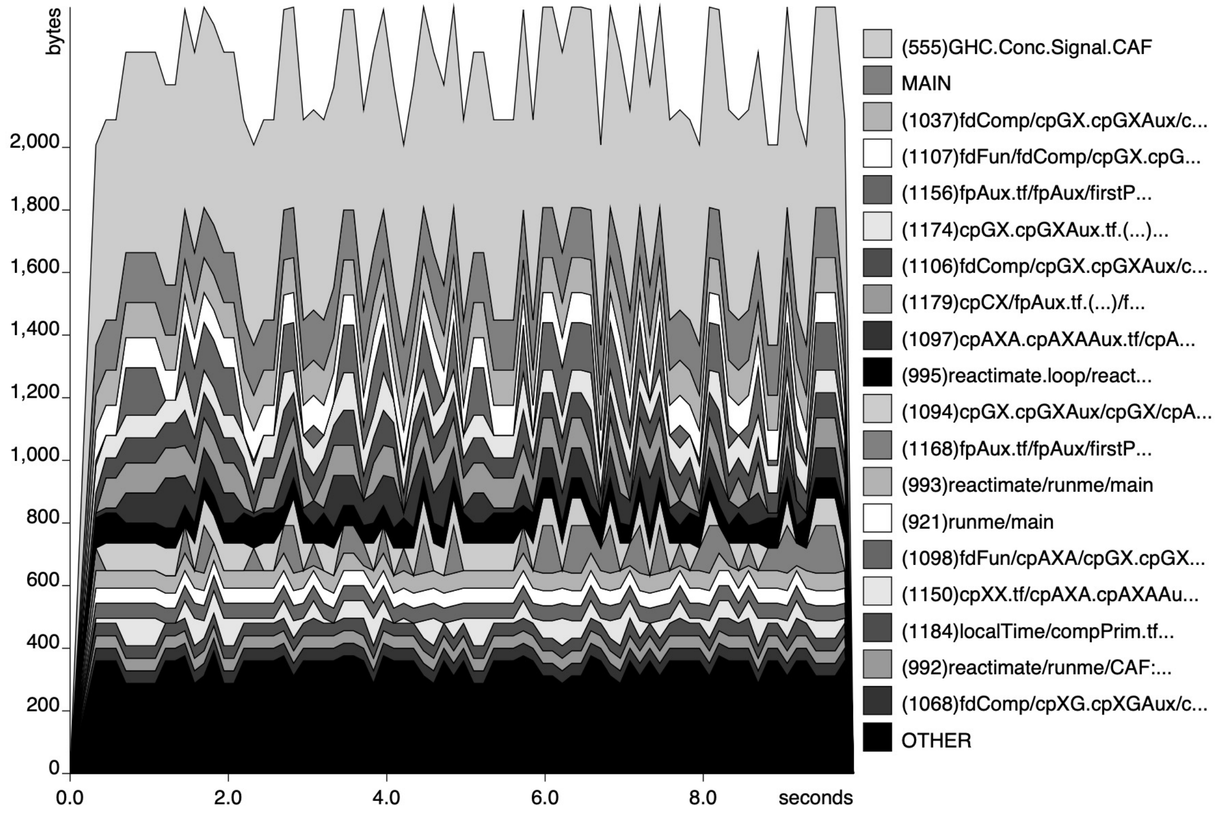
Inter-Turn Semantics

- Motivation
- Formalisation
- **Lessons learned**

Lessons Learned

- Graph-based RP needs wiring and propagation
  - w-rules and s-rules
  - Wiring and propagating can, in general, be interleaved!
    - Run-time wiring decisions due to higher-order reactivity
- Memory consumption …

Remember this graph?



Why does this happen?

(>>>): sequential composition of SFs

Signal Functions

```
(>>>) :: SF a b -> SF b c -> SF a c
(SF {sfTF = tf1}) >>> (SF {sfTF = tf2}) =
    SF {sfTF = tf}
  where
    tf dt a = (sf1' >>> sf2', c)
      where
        (sf2', c) = tf2 dt b
        (sf1', b) = tf1 dt a
```

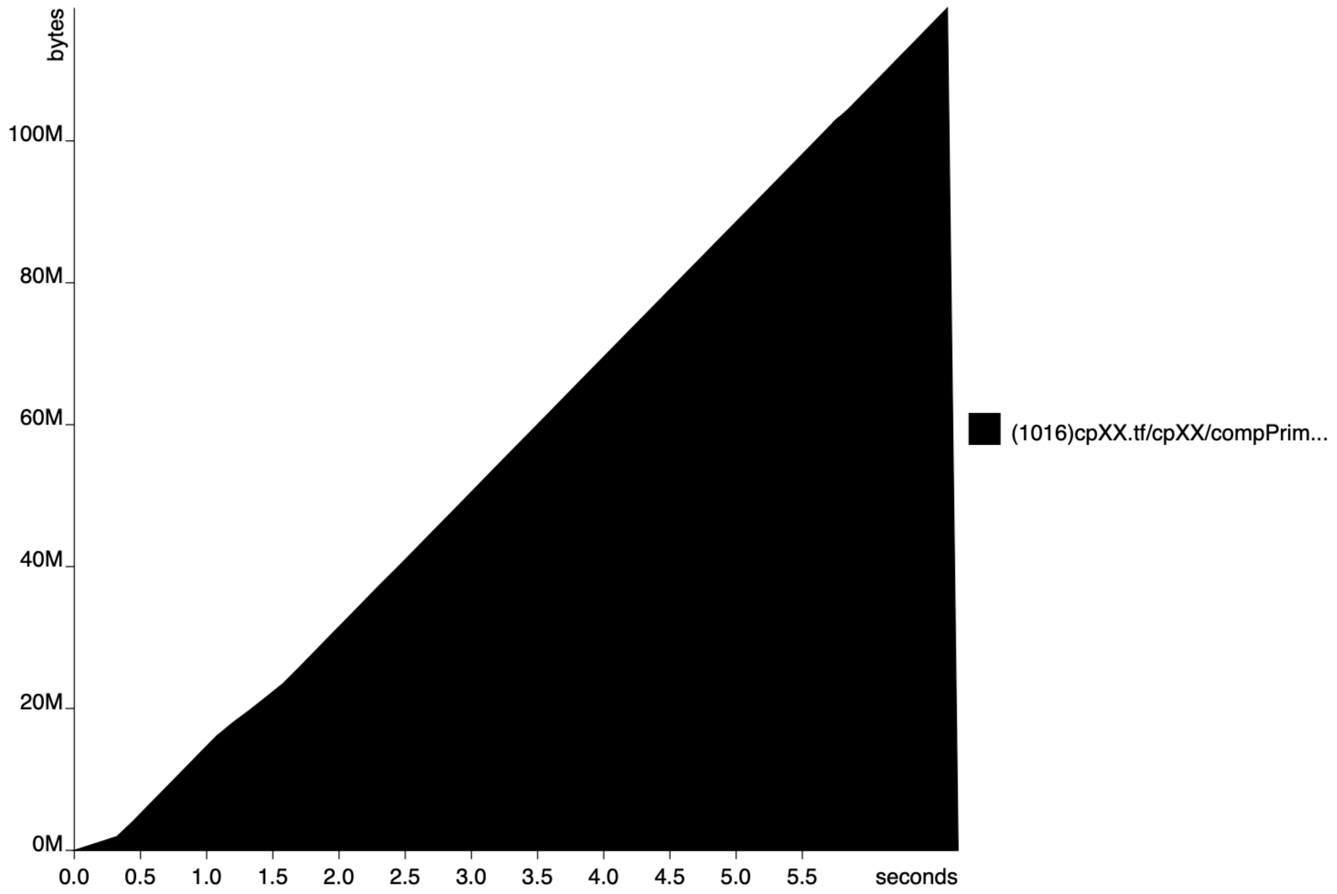
Recursion

Elegant, because it's all *just* composition of functions.

Actually, this is not what really happens due to optimisations…

Nilsson, H., Courtney, A., & Peterson, J. (2002, October). Functional reactive programming, continued. In *Proceedings of the 2002 ACM SIGPLAN workshop on Haskell* (pp. 51-64).

Without optimisations



Cooper, G. H., & Krishnamurthi, S. (2006, March). Embedding dynamic dataflow in a call-by-value language. In *European symposium on programming* (pp. 294-308). Springer, Berlin, Heidelberg.

GC isn't possible!

Each recursive application of (>>>) extends the function's lexical environment.
In addition to Haskell's lazy evaluation model, this introduces a space leak!

Which is what graph-based RPLs avoid altogether by treating RP programs as graphs [*].

In case of Karcharias…

(NEXT-TURN)

$$\frac{\begin{array}{c} k = \mathcal{K}\langle E, I_d, W, S, D\rangle \in \widetilde{K} \\ \mathcal{S}\langle \iota_{main,d}, E_{main}\rangle \in S \\ k' = \mathcal{K}\langle E[x_i \mapsto v_{i,now} \mid \forall i \in [1..|\tau|]], \{\iota_{main,d}\}, W, \varnothing, D\rangle \end{array}}{\begin{array}{c} \mathcal{K}^{inter}\langle \{x_i \mapsto [v_{i,now}] \S \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k\rangle \\ \rightsquigarrow \mathcal{K}^{inter}\langle \{x_i \mapsto \overline{v}_i \mid \forall i \in [1..|\tau|]\}, k'\rangle \end{array}}$$

Question now becomes: when do we have to wire?

- Easy/ier to determine memory consumption
  - Wirings remain static, after they have been constructed.
  - Snapshots are ”reduced” versions of wirings (with actual values of signals)

Future Work

- Currently lacking:
  - No error handling / detection of stuck states
  - Stateful computations aren’t formalised yet
  - Thorough comparison between function-based RP and graph-based RP

- Ultimate goal
  - Expressive, higher-order, RP language that works for very small computers
  - Statically classify RP programs w.r.t. their required memory allocation/consumption behaviour

Looking for ideas!

A Graph-Based Formal Semantics of Reactive Programming from First Principles

Bjarno Oeyen, Joeri De Koster, Wolfgang De Meuter

Software Languages Lab, Vrije Universiteit Brussel, Belgium

FTfJP Workshop @ Ecoop (07/06/2022)

SOFTWARE
LANGUAGES
LAB

Fonds Wetenschappelijk Onderzoek
Vlaanderen
Opening new horizons