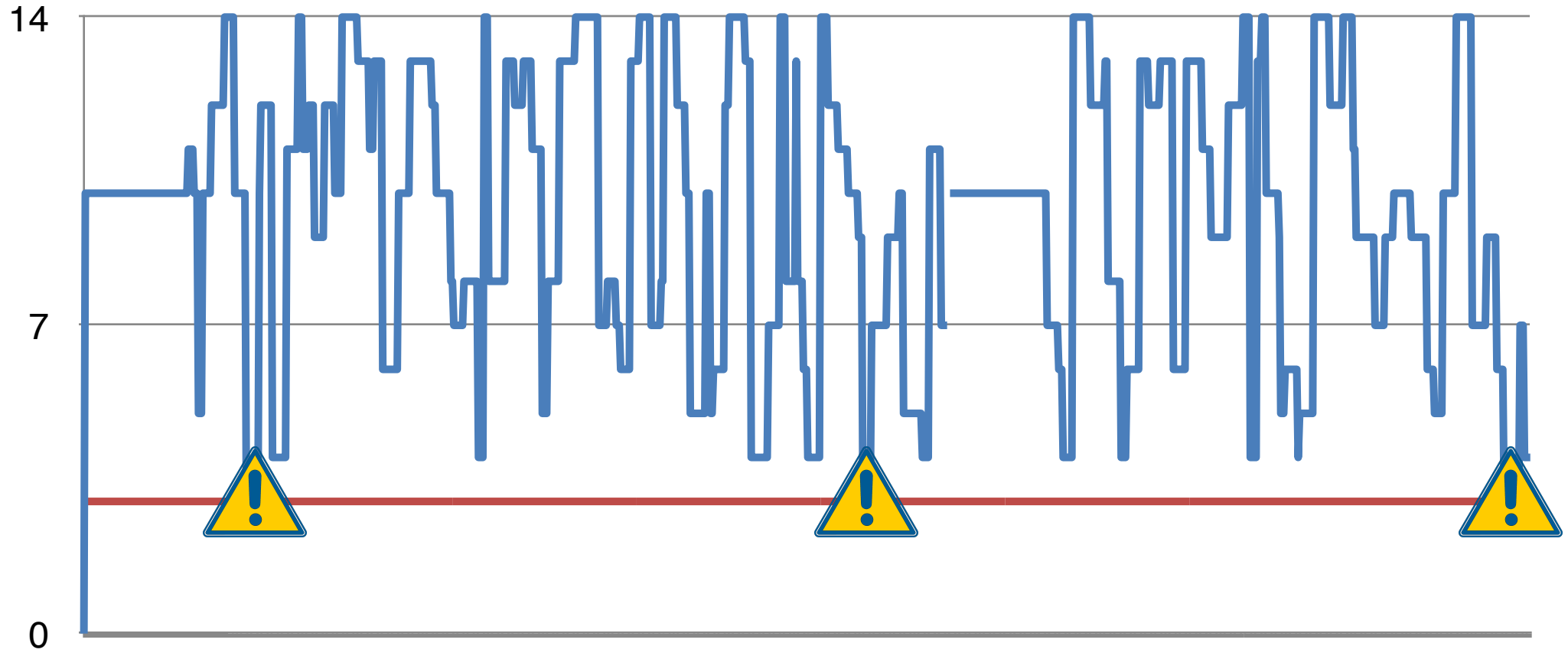
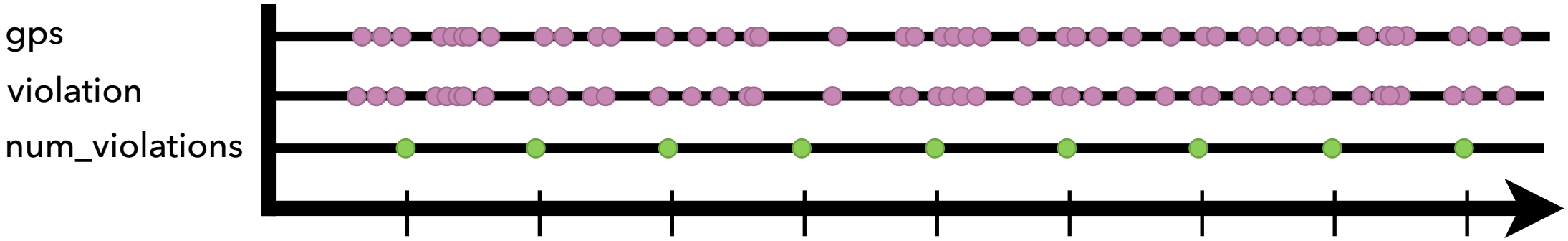
RTLola: Formal Monitoring for CPS

Jan Baumeister


14
7
0

Type System: Pacing Type + Value Type

```
input gps: Int64 @{gps}
output violation := gps < 9 @{gps}
output num_violations @1Hz := violation.aggregate(over: 10s, using: Σ) @1Hz
```



Type System ensures that the specification requires only constant memory.

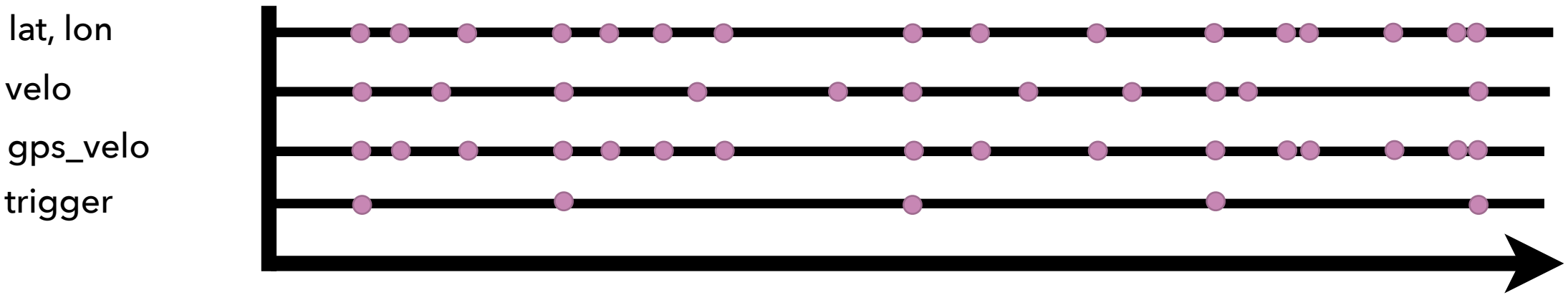
Asynchrony

```
input lat, lon: Float64

input velo: Float64

trigger abs(gps_velo - velo) > 0.1
```

Type System ensures that the lookups are infaillable.



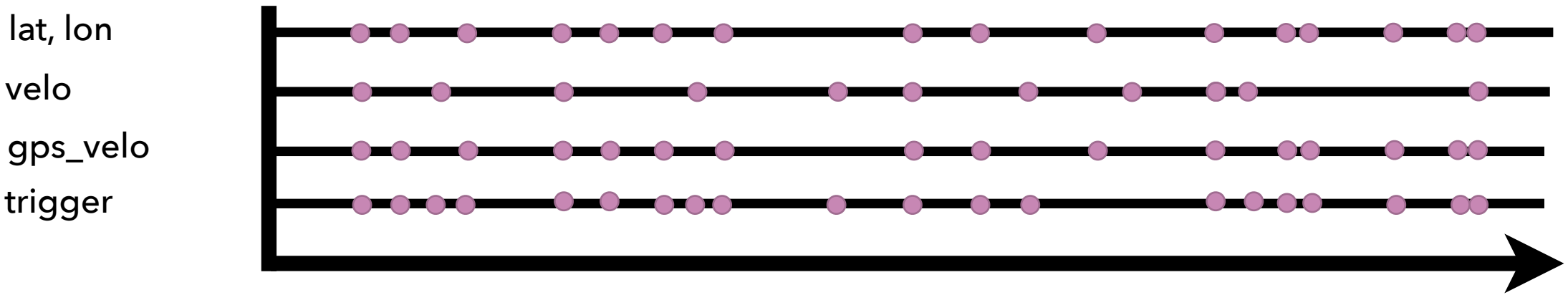
Asynchrony

```
input lat, lon: Float64
```

Type System ensures that the lookups are infallible:
Synchronous Lookups couple timing; Holds + Aggregations decouple timing

```
trigger
 abs(gps_velo.hold(or: 0.0) - velo.hold(or: 0.0)) > 0.1
```

@{(lat ∧ lon ∧ time) ∨ velo}



Dependency Graph

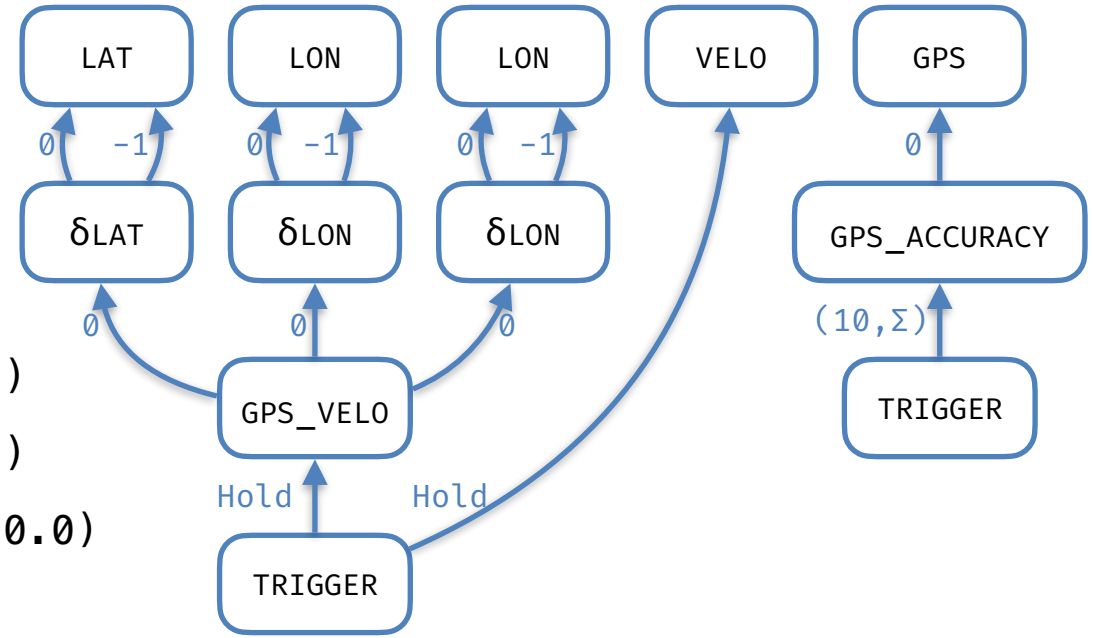
```
input gps: Int64
input lat, lon, velo, time: Float64

output gps_accuracy:= gps < 9
trigger @1Hz
  gps_accuracy.aggregate(over: 10s, using: Σ) > 5

output δlat:= lat−lat.offset(by:−1).defaults(to:lat)
output δlon:= lon−lon.offset(by:−1).defaults(to:lat)
output δtime:= time−time.offset(by:−1).defaults(to:0.0)
output gps_velo:= sqrt(δlat − δlon) / δtime

trigger @{(lat ∧ lon ∧ time) ∨ velo}
  abs(gps_velo.hold(or:0.0) − velo.hold(or:0.0)) > 0.1
```

Stream	#values	Size	Windows	Total	Layer
LAT	2	128		128	1
LON	2	128		128	1
TIME	2	128		128	1
VELO	1	64		64	1
GPS	1	64		64	1
δLAT	1	64		64	2
δLON	1	64		64	2
δTIME	1	64		64	2
GPS_ACCURACY	1	64	640	704	2
GPS_VELO	1	64		64	1



CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

www.rtlola.org