MEZZO

http://protz.github.io/mezzo

Thibaut BALABONSKI
François POTTIER
Jonathan PROTZENKO
INRIA – Gallium

Well-typed programs cannot go wrong (Milner)

STATIC (types)

DYNAMIC (operational semantics)

INVARIANT

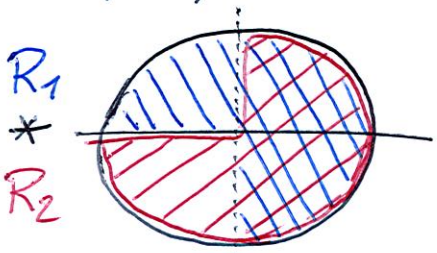
Subject reductio

$R_1 \vdash t : T \quad R_2 \vdash$ invariants of free locks $\equiv S$

$\vdash S / t$

SEPARATION LOGIC

turned into a type system



MODES

Excl.	Dupl.
R/W	R-only
ref	int
mutable structures	functions
	chan
	lock

Logical resource

R	Environment Heap + modes Lock invariants

Concrete state

S	Heap Lock statuses

APP

$$\frac{R_1 \vdash v : T \to U \quad R_2 \vdash t : T}{R_1 * R_2 \vdash v\,t : U} \qquad S / (\lambda x.t)\,u \longrightarrow S / t^{\{x := u\}}$$

ASSIGN

$$\frac{R_1 \vdash v : \mathrm{ref}\, T_1 \quad R_2 \vdash t : T_2}{R_1 * R_2 \vdash v := t : () \mid v @ \mathrm{ref}\, T_2} \qquad S / l := t \longrightarrow S[l \mapsto t] / ()$$

ACQUIRE

$$\frac{R \vdash v @ \mathrm{lock}\, P}{R \vdash \mathrm{acquire}\, v : () \mid P} \qquad S / \mathrm{acquire}\, k \longrightarrow S[k \mapsto \mathrm{taken}] / () \quad (S(k) = \mathrm{free})$$

WRONG

Stuck
↳ Progress

Data-race
↳ DR Freedom

Deadlock