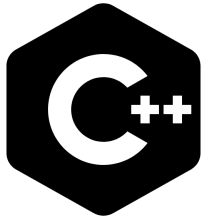
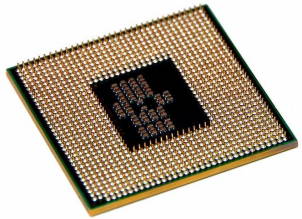
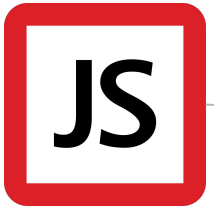
Hierarchy Builder

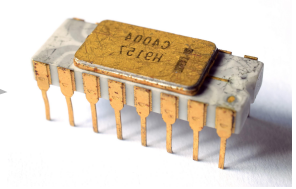
C.Cohen, K.Sakaguchi and E.Tassi

Interoperability, my take (½)

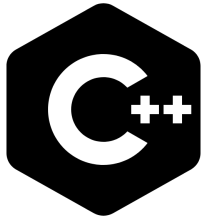


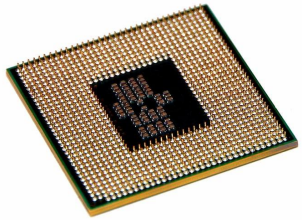


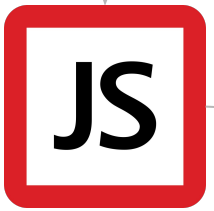


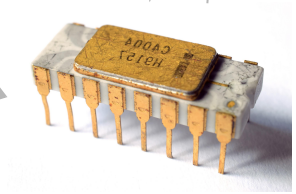


Interoperability, my take (½)





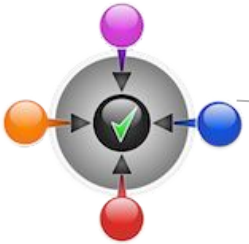




Spot the difference (2/2)

λCIC

Why not?



$\lambda\Pi$

Libraries and abstractions

In order to scale, library developers (in any language):

- Organize the contents around interfaces
- Organize the interfaces in a hierarchy

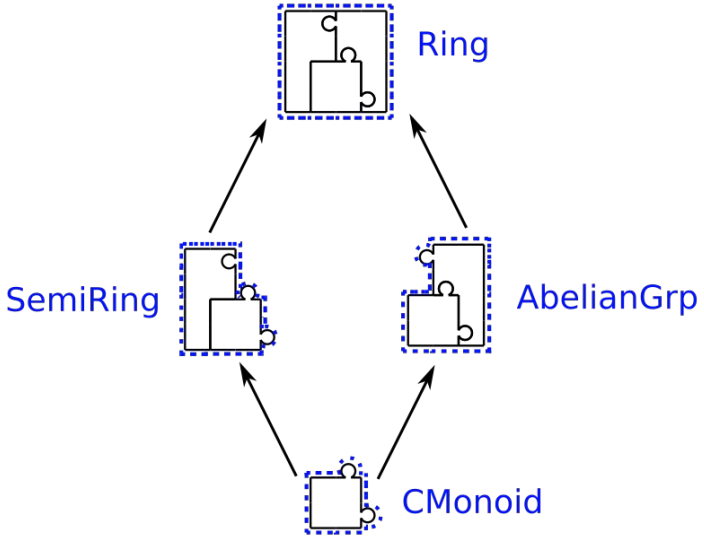
These ``linguistic tools" are in most programming languages

These ``linguistic tools" are **not** in our logics!

- I'm no logician, if you are please work to fix this
- Together with CC and KS we added these "tools" to the user facing language of Coq, this extension is called *HB (hierarchy builder)*

Demo [.v file]

The hierarchy of structures



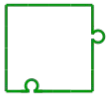
The puzzle pieces (mixins)


CMonoid_of_Type


AbelianGrp_of_CMonoid


SemiRing_of_CMonoid

The "virtual" puzzle pieces (factories)


Ring_of_Type

About WG4

- Crucial (to me) to “translate” the user facing languages, not just the assembly (some pun intended)
- Their implementation can be completely different
  - Even for the same system (HB may soon have two backends for Coq)
- Some success in the past with the SSR proof language
  - Easycrypt
  - HOL-light

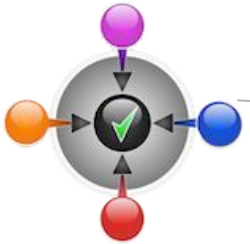
Thanks! Questions?

HB

λCIC

HB

why not?



$\lambda\Pi$