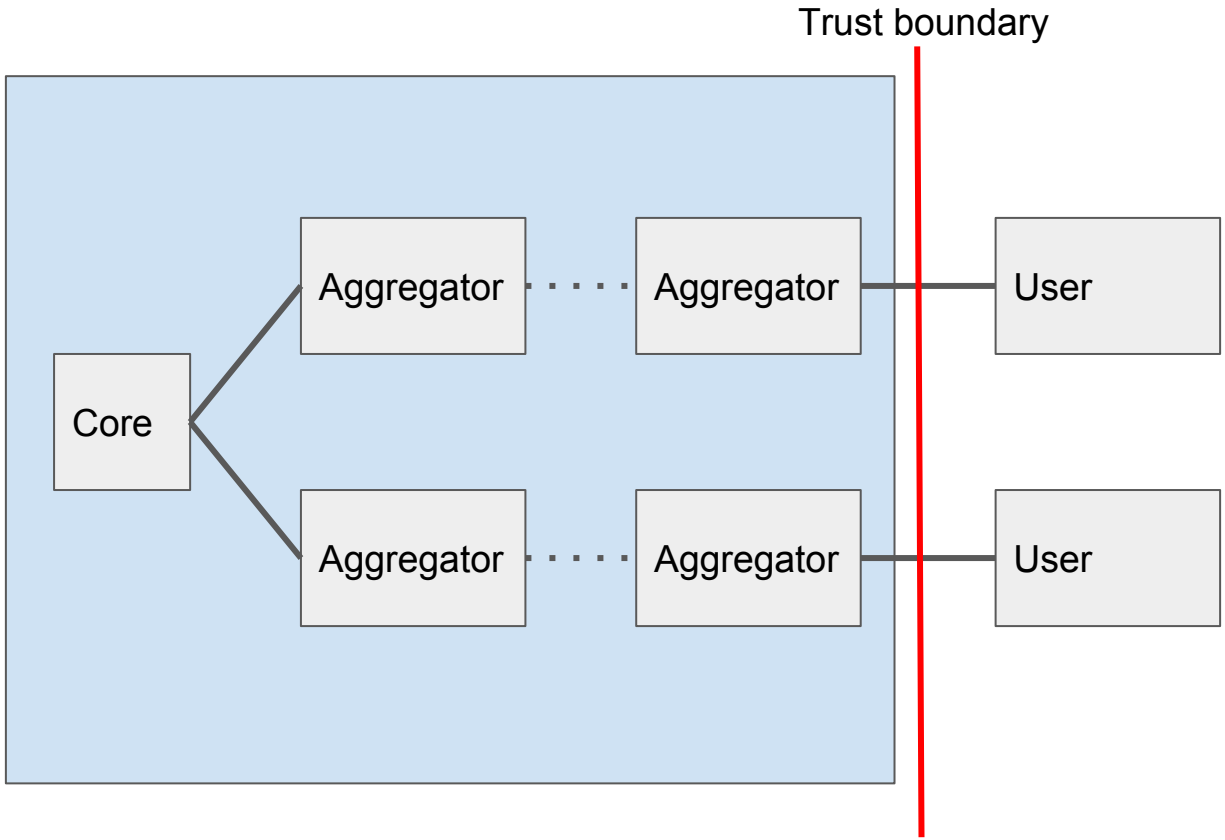
Languages at Galois

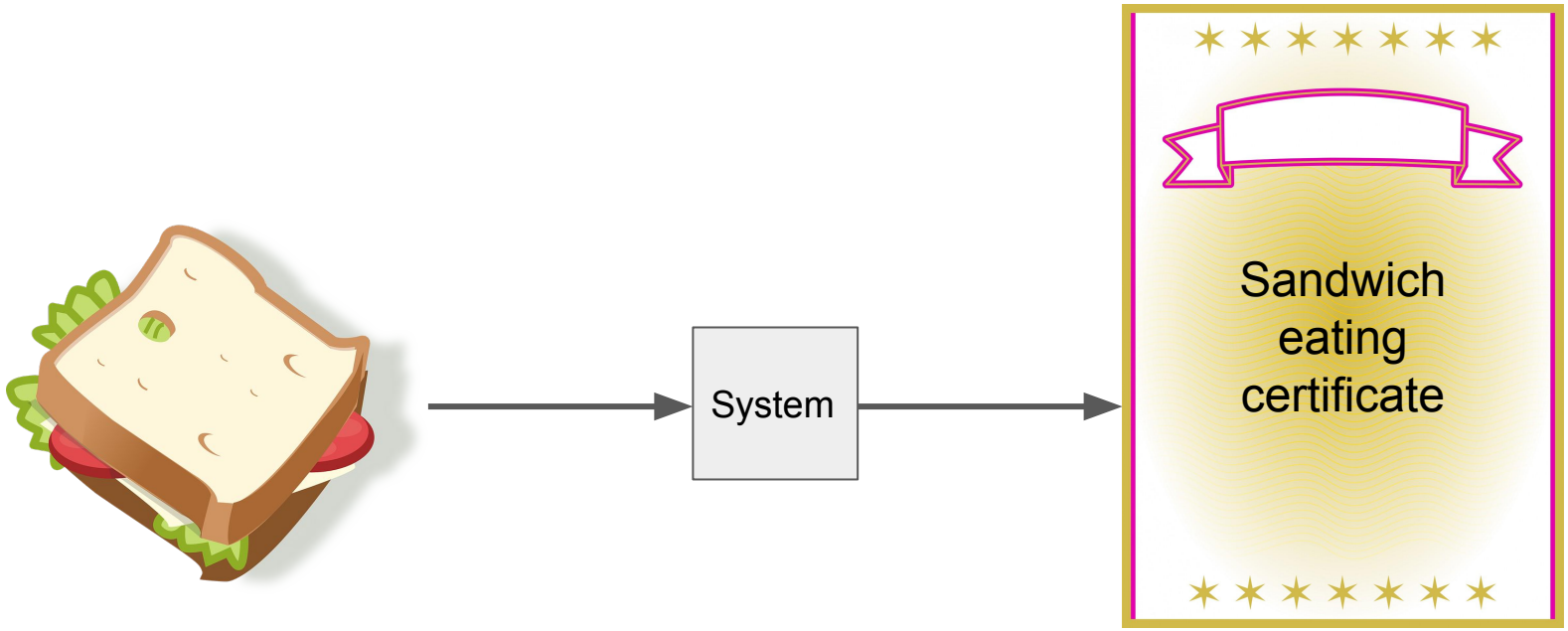
Joey Dodds and many others


PORTLAND
OREGON USA

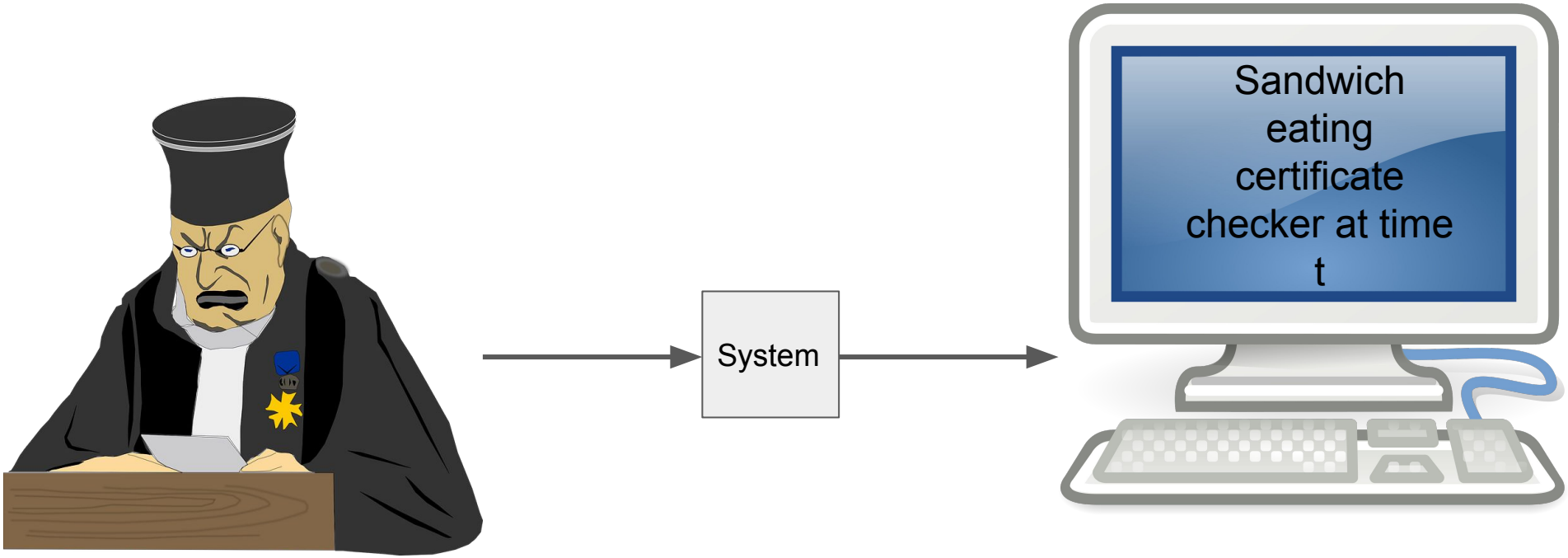

BUNK


Trust boundary
Core
Aggregator
Aggregator
Aggregator
Aggregator
User
User

What does a user want?



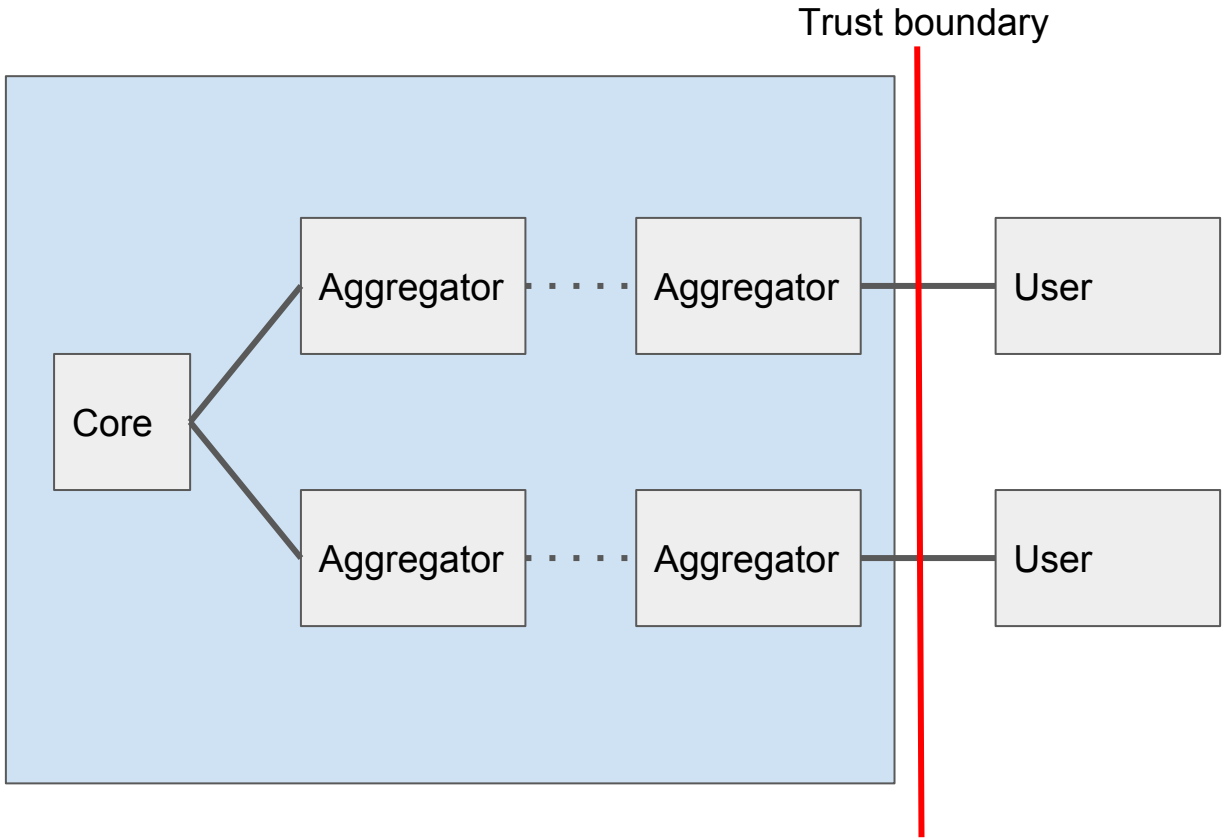
What does a user want?



How many languages?

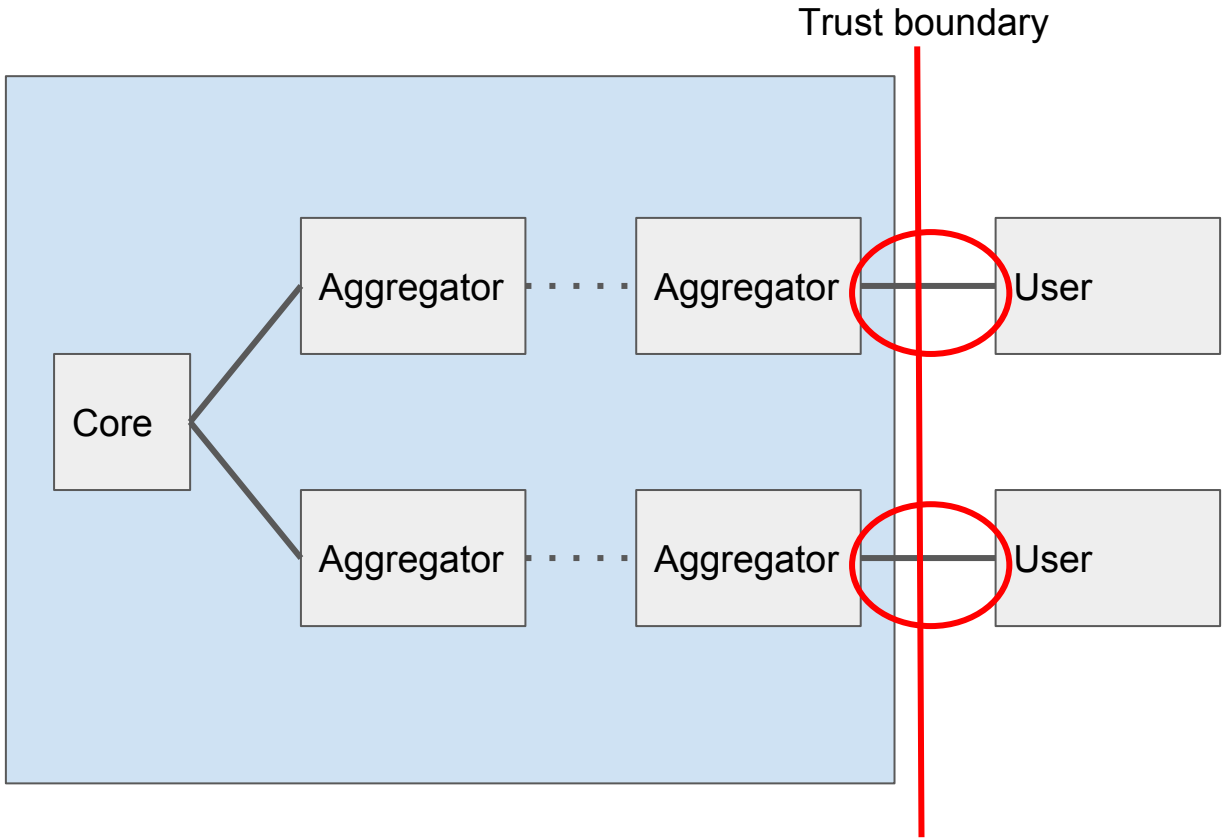
5*

*It’s way more than 5


Trust boundary
Core
Aggregator
Aggregator
Aggregator
Aggregator
User
User

Layered verification

- Code meets a low-level specification
- Low-level specification meets higher-level specification
- High level-specification has meaningful properties


Trust boundary
Core
Aggregator
Aggregator
User
Aggregator
Aggregator
User

Layered verification (TLS)

- Code meets a low-level specification

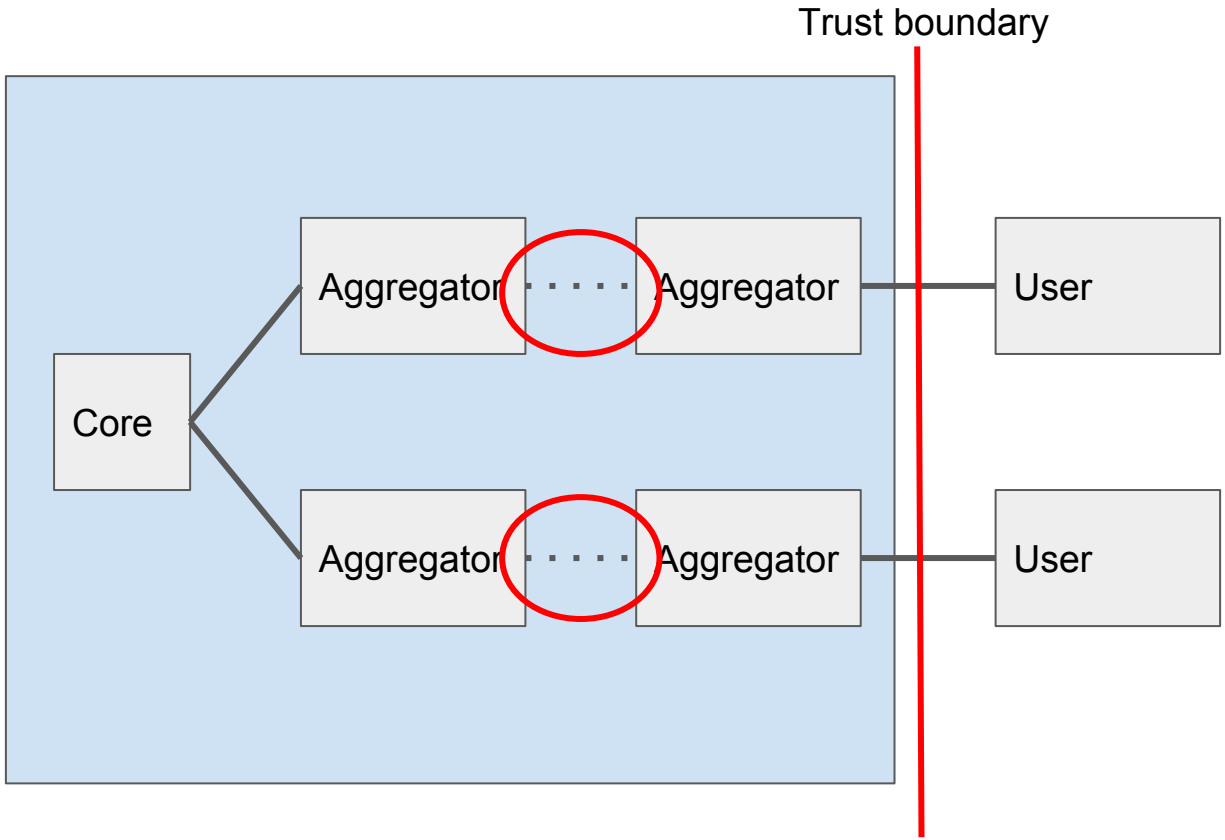
Proof that C code for transition function is equal to ours

- Low-level specification meets higher-level specification

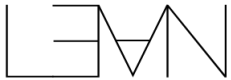
Proof that our transition fn is equal to a fn written from RFCs

- High level-specification has meaningful properties

Handshake always completes


Trust boundary
Core
Aggregator
Aggregator
Aggregator
Aggregator
User
User

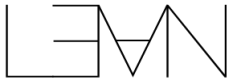
Proof for internal messages

In  we've defined *executable* serialization and parsing functions

We have proved:

∀ *msg*. parse (serialize *msg*) = *msg*

Proof for monolithic system

In  we've defined a linear temporal logic

This allows us to talk about things that □ always happen

And ♢ eventually happen

Proof for monolithic system

We prove that

if a user sends a message to the system now,

eventually they will get a certificate back

This was crazy hard