

An Accidental Internet Architecture

Munich Internet Research Retreat
24-25 November 2016

Brian Trammell (ETH NSG)

~~An Accidental Internet Architecture~~

Post Sockets

a top-down architecture proposal,
and a way to think about the world after sockets
Munich Internet Research Retreat
24-25 November 2016

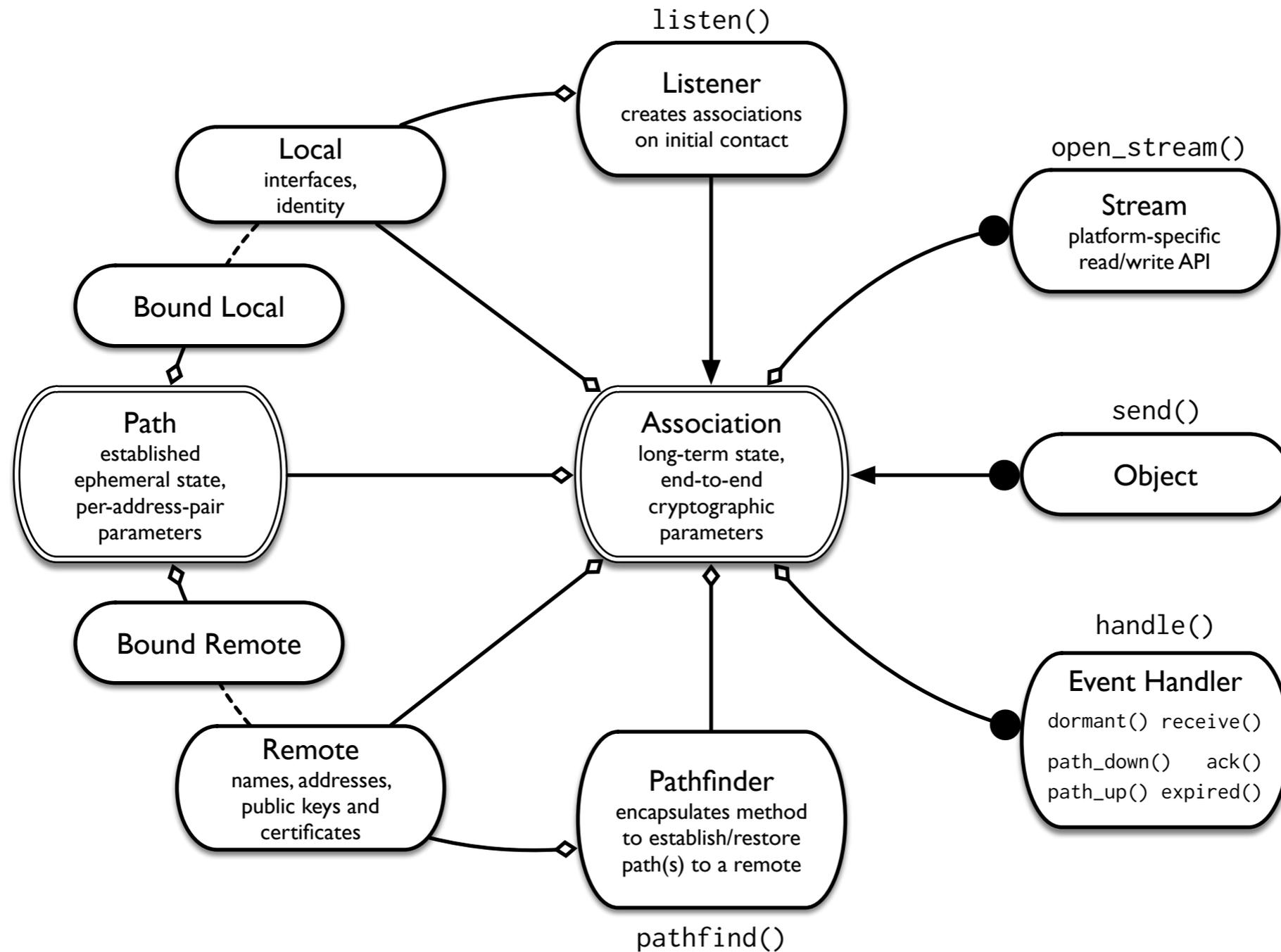
Brian Trammell (ETH NSG), Colin Perkins (U. Glasgow),
Tommy Pauly (Apple), and Mirja Kühlewind (ETH NSG)
(with thanks to Jason Lee and Laurent Chuat (ETH NetSec))

A few insights

- ***Applications deal in objects*** (messages) of arbitrary size
 - Files, assets, media frames, etc. may depend on each other, but usually don't require a strict ordering.
- The network of the future is ***explicitly multipath***.
- Future transports must ***guarantee security properties***.
- Message reception is ***inherently asynchronous***.
- There are ***two kinds of state*** in a transport connection: ephemeral per-path state, and durable per-identity-pair state.
 - Separating these makes new kinds of interaction possible.

Abstract Programming Interface

Classes and Entry Points



Abstract Programming Interface

Object and Stream properties

- Objects and streams have a **niceness**
 - Nicer `send()`s/`write()`s yield to less nice
- Objects have a **deadline**
 - An object will be cancelled if it cannot be realistically received before this deadline
 - Infinite-deadline objects are fully reliable
- Objects may have **antecedents**
 - Other objects which should be sent before
- All of these work **sender-side only**
 - Post needs no signaling: requires only an object framing primitive from the underlying transport protocol for full functionality.

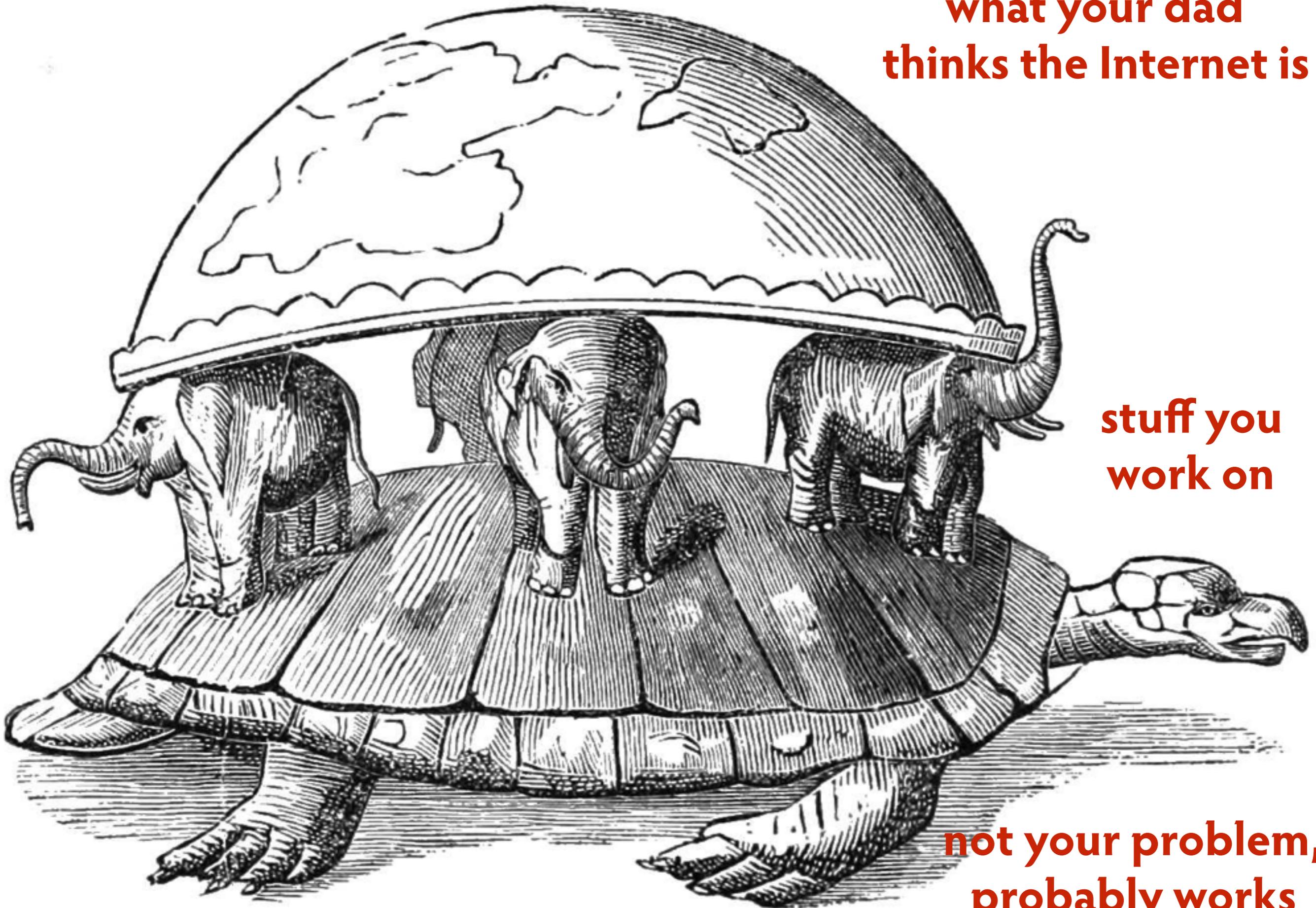
State Separation

- A connection between two endpoints is made up of two kinds of state with different lifetimes:
 - Association-scoped
 - identities of endpoints (names, certificates, etc.)
 - cacheable crypto state (resumption parameters, etc.)
 - Path-scoped
 - addresses of endpoints
 - ephemeral transport state (ports, tokens, sequence nr, etc.)
 - ephemeral crypto state (session keys, etc.)
- Transport layer to date has only dealt with ephemeral state.
 - With durable state the distinction between “connected” and “disconnected” disappears for the application.
 - Associations can migrate from one endpoint to another

an API does not an architecture make
(or does it?)

- *“The Internet” is defined by interfaces to the services and user agents it connects.*
- Thinking from a principle out results in a radical reimagining of the Internet, with a challenging deployment story and/or variable salvage value.
- Thinking from an interface down results in a more incremental (but easily deployable) redesign.

**what your dad
thinks the Internet is**



**stuff you
work on**

**not your problem,
probably works**