

# Preserve Host Virtualizability

Mark S. Miller, Agoric  
JF Paradis, Agoric  
Caridy Patiño, Salesforce  
Dan Finlay, MetaMask  
Alan Schmitt, Inria

tc39 February 2020, Oahu Hawaii

**User-mode  
instructions**

**standard EcmaScript  
language**

**Trapping**

**Global scope lookup**

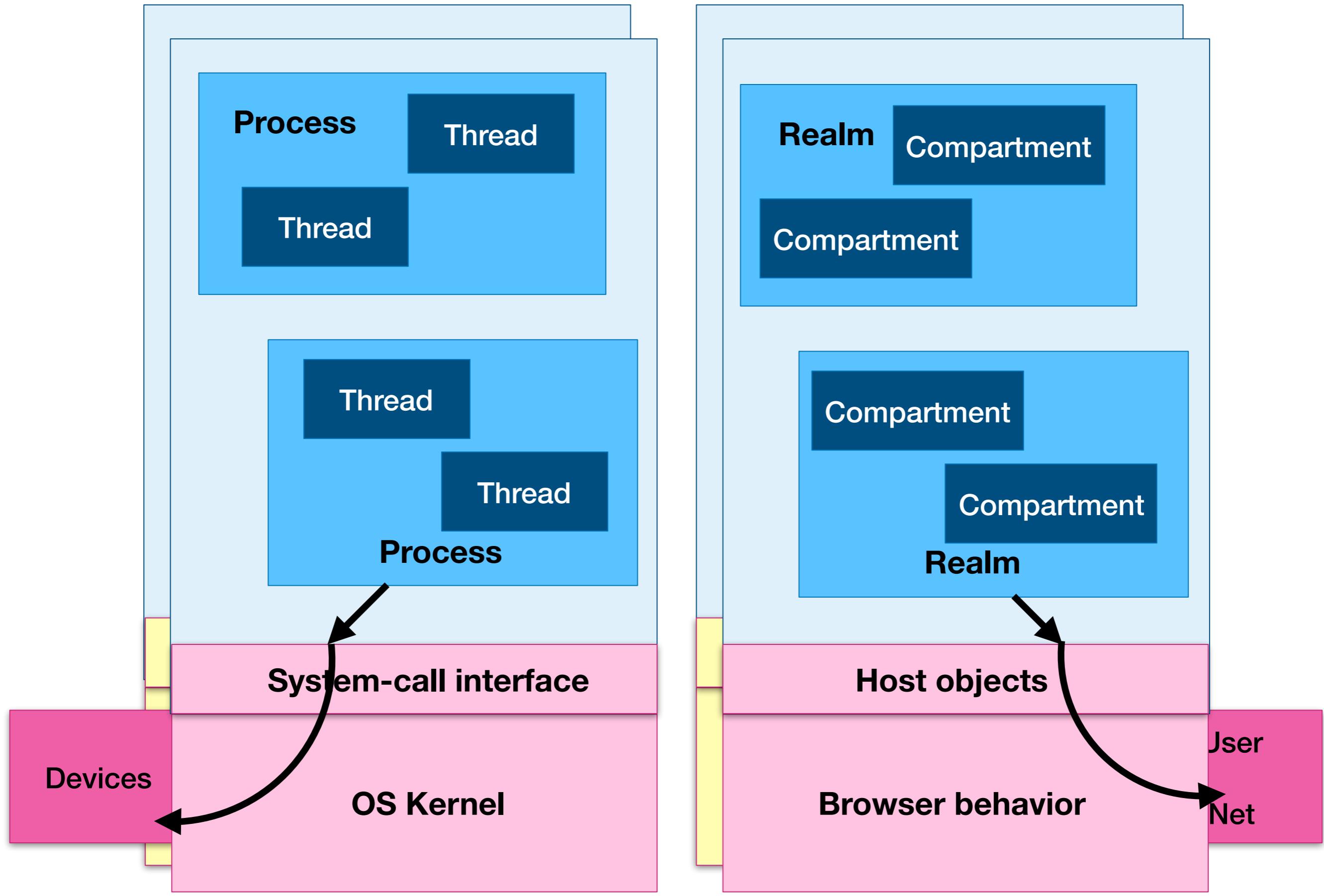
**Devices**

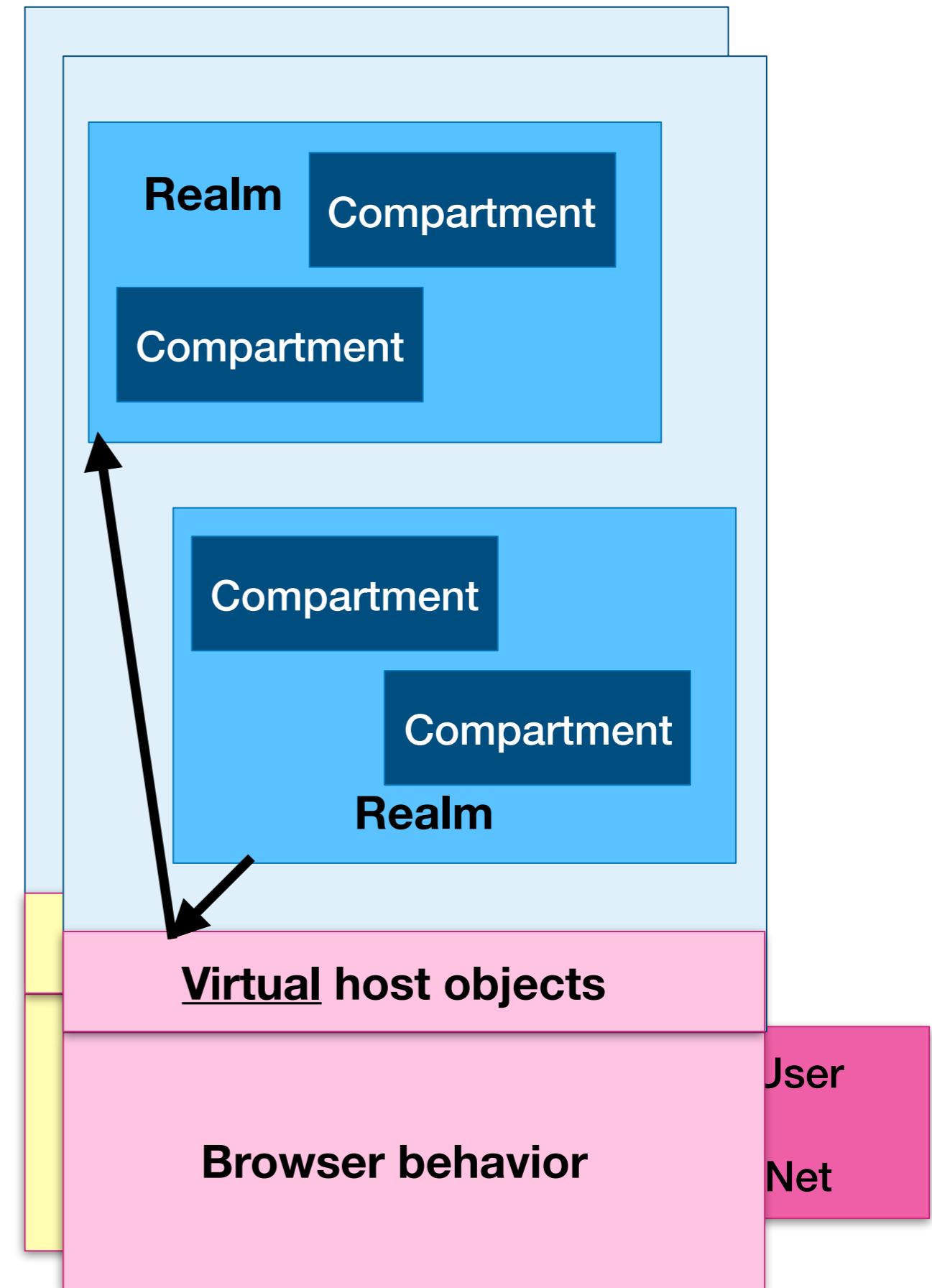
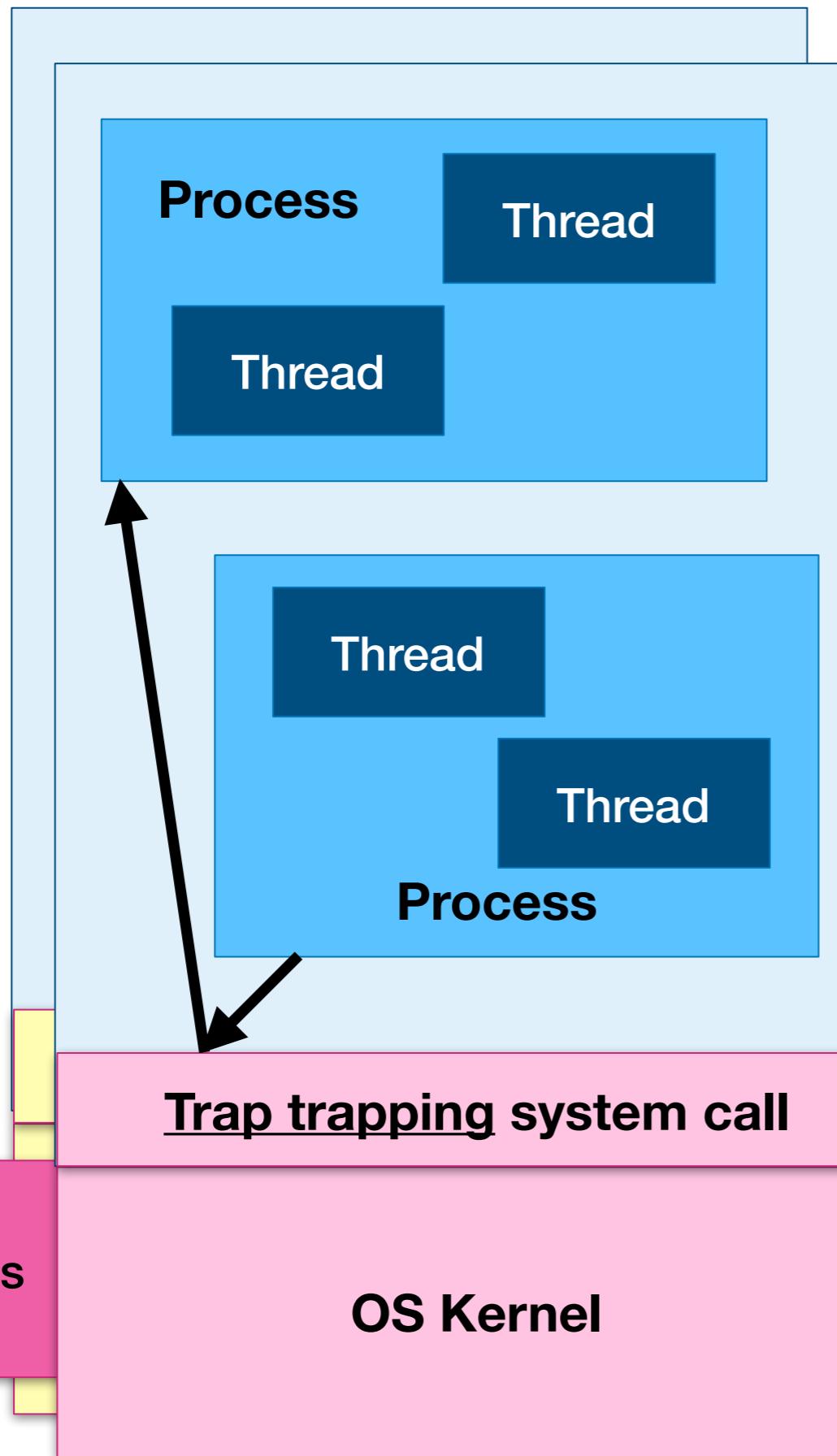
**System-mode  
instructions**

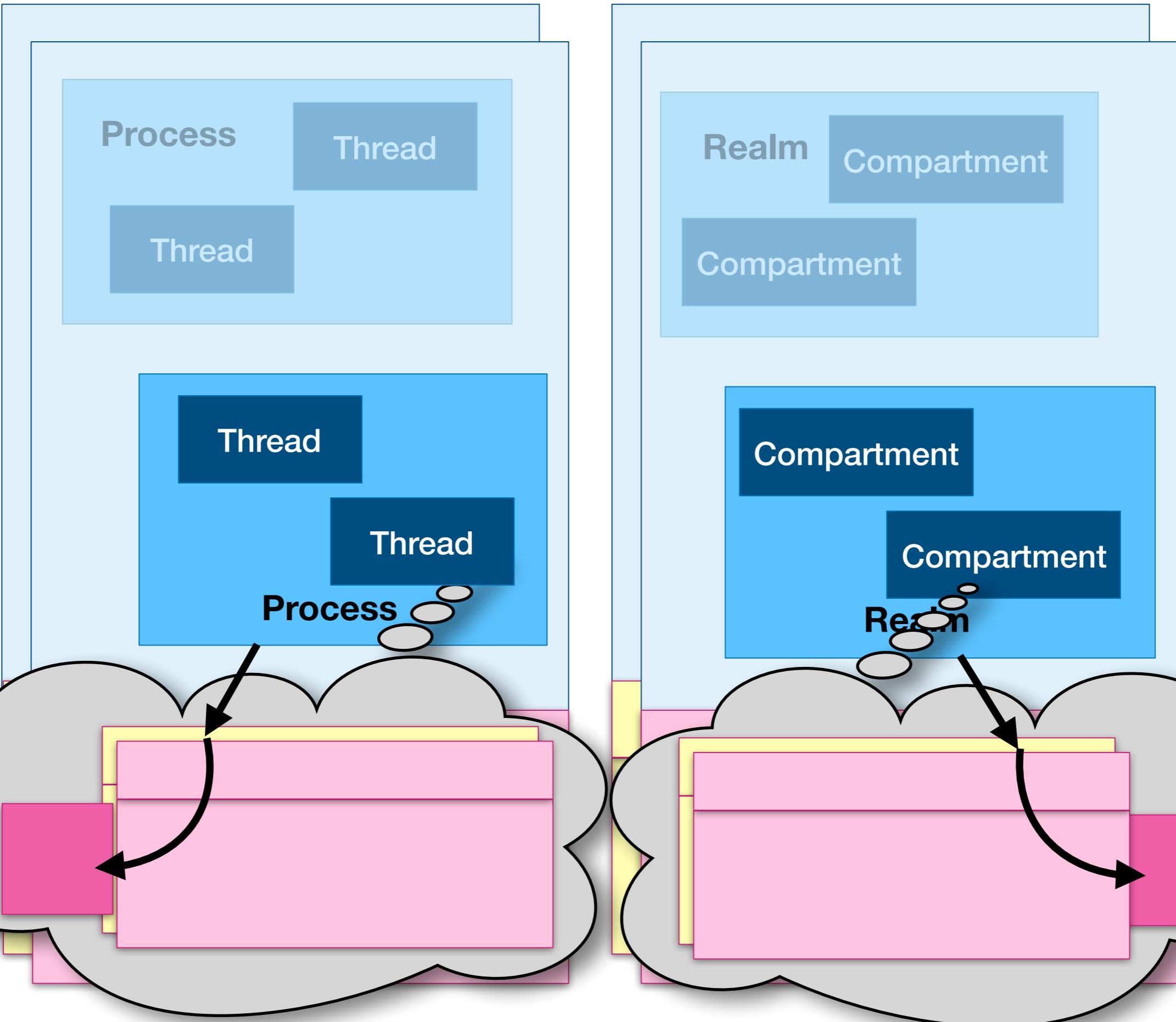
**Browser internals**

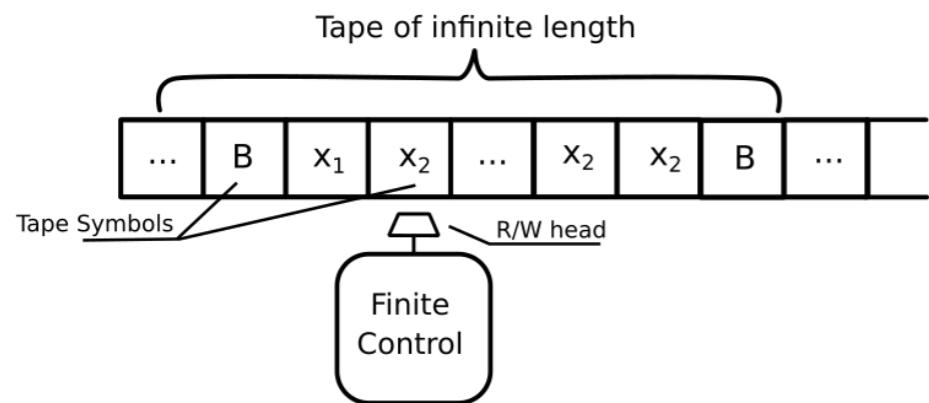
**User  
Net**











# Only Universal

# Virtualizable



# Threats to virtualizability

**Implementation adds:**

```
defineProperties(Array.prototype, {  
  peek: { value: peekFn, writable: false, configurable: false },  
  poke: { value: pokeFn, writable: false, configurable: false },  
});
```

**Shim tries:**

```
Array.prototype = ...;  
Array = ...;
```

**Doesn't matter**

```
[].poke(0x4f35bc40, 0x59cd97b0)
```



# Threats to virtualizability

**Better:**

```
defineProperties(Array.prototype, {  
  peek: { value: peekFn, writable: false, configurable: true },  
  poke: { value: pokeFn, writable: false, configurable: true },  
});
```

**Shim deletes whatever's not on whitelist:**

```
delete Array.prototype.peek;  
delete Array.prototype.poke;
```



# How hosts should influence behavior

Everything starts deletable, shimmable

Global host objects

Well behaved exotics

Intentional host hooks

Import namespace, loading behavior



# How hosts should influence behavior

Everything starts deletable, shimmable

Global host objects

Well behaved exotics

Intentional host hooks

Import namespace, loading behavior

# How host should not influence behavior

Prevent shimming

Extra properties, extra behavior, extra syntax

Badly behaved exotics (document.all)

Accidental host hooks

Hidden state, hidden I/O



Agent

Agent

Realm

SES Realm

Compartment

Compartment



Agent

Agent

Realm

SES Realm

Compartment

Compartment

obj

p

target

handler



# Questions?

