

Spritely Goblins: A Distributed Journey

Christopher Lemmer Webber

<https://dustycloud.org/>

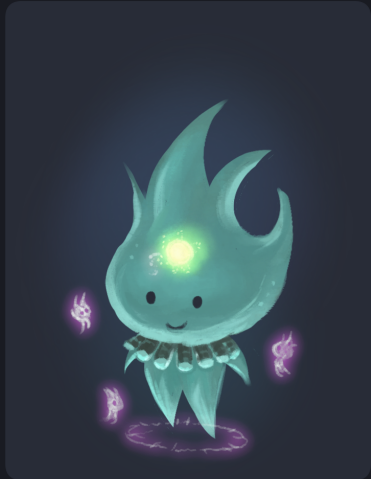
Fediverse: <https://octodon.social/@cwebber>

Birdsite: <https://twitter.com/dustyweb>

Spritlely

✿ Social Worlds Await ✿

<https://spritlelyproject.org/>







Distributed Programming Environment



Quasi-Functional State Management



Easy Transactions

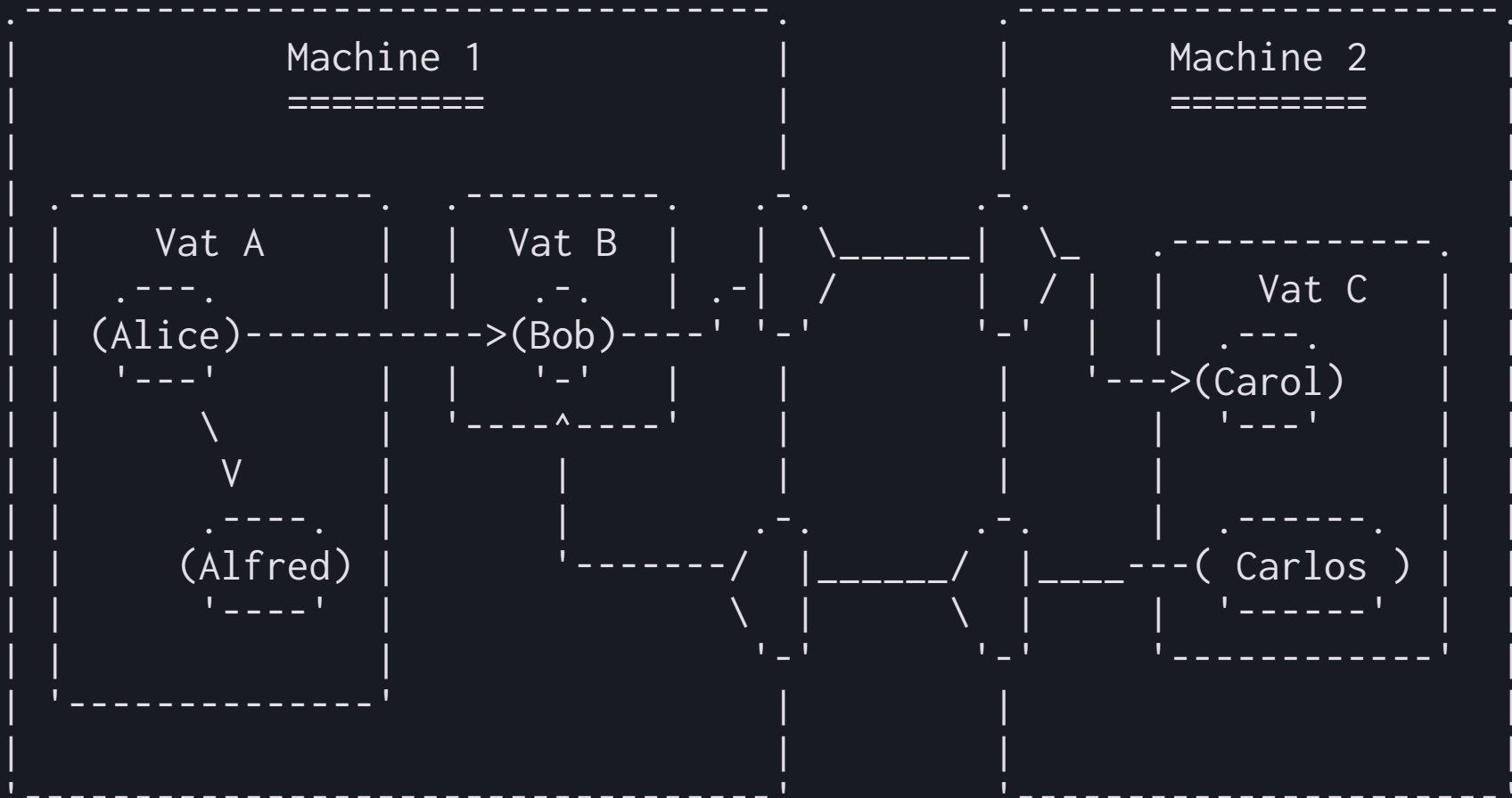


Time Travel Included



Object Capability ("ocap") Security

* 4th wall breakage... *



```
(machine (vat (actormap {refr: (mactor object)})))
```

(Condensed) Goblins' Heritage

- ~193X: λ
- ~1958: (λ () ...)
- ~1972: Smalltalk
- ~1973: Actors
- ~1975: Scheme
- ~1995: Joule
- ~1997: E

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	(<code><- ...</code>) or (<code>\$...</code>)

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	(<code><- ...</code>) or (<code>\$...</code>)
Create actors	

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	(<code><- ...</code>) or (<code>\$...</code>)
Create actors	(<code>spawn ...</code>)

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	(<code><- ...</code>) or (<code>\$...</code>)
Create actors	(<code>spawn ...</code>)
Designate next behavior	

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

Actors	Goblins
Send messages / invoke	(<- ...) or (\$...)
Create actors	(spawn ...)
Designate next behavior	(bcom ...)

Carl Hewitt & co: the (classic) actor model

In response to a message, can:

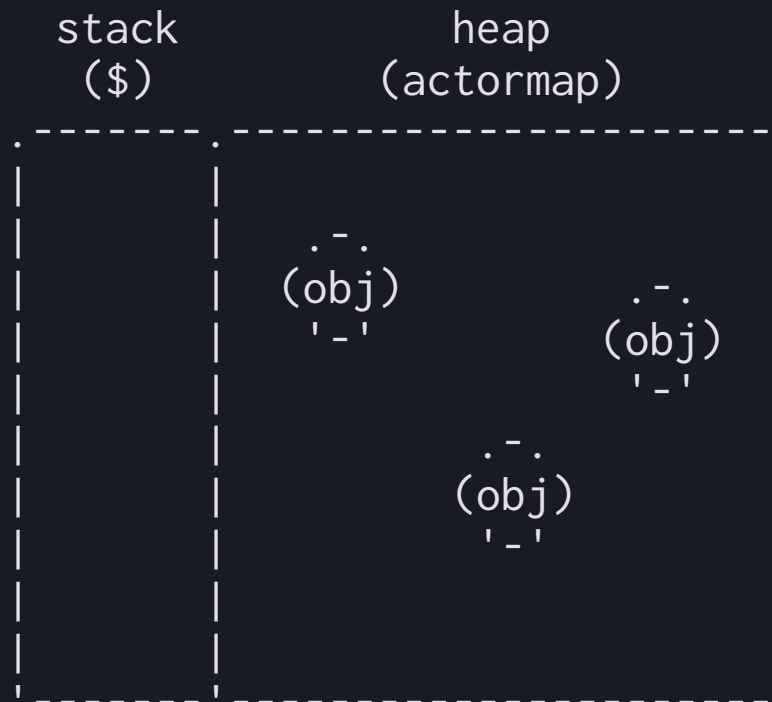
Actors	Goblins
Send messages / invoke	(<- ...) or (\$...)
Create actors	(spawn ...)
Designate next behavior	(bcom ...)

WTF, why two?!

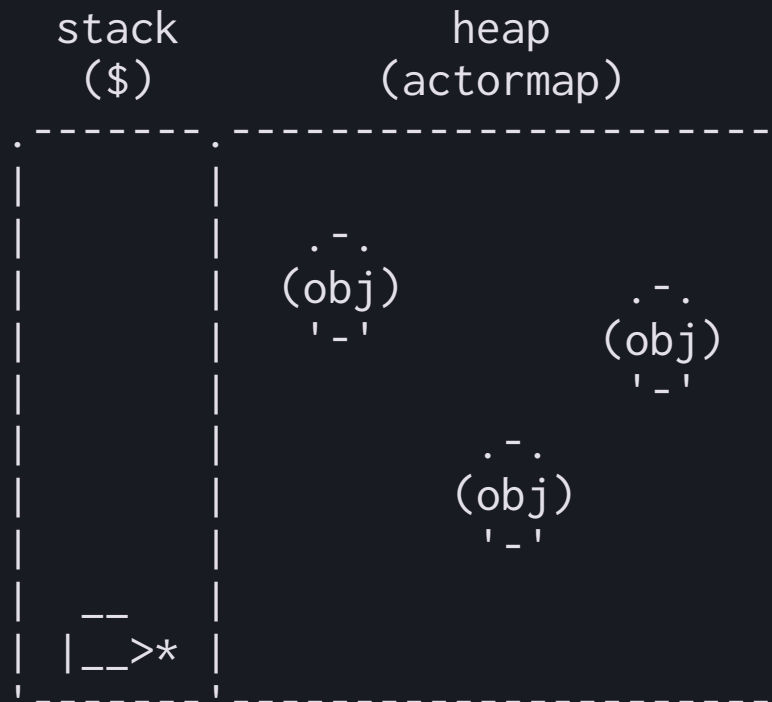
The Synchronous Call-Return Worldview

- ~193X: λ
- ~1958: $(\lambda () \dots)$
- ~1972: Smalltalk
- ~1973: Actors
- ~1975: Scheme
- ~1995: Joule
- ~1997: E (... kinda)

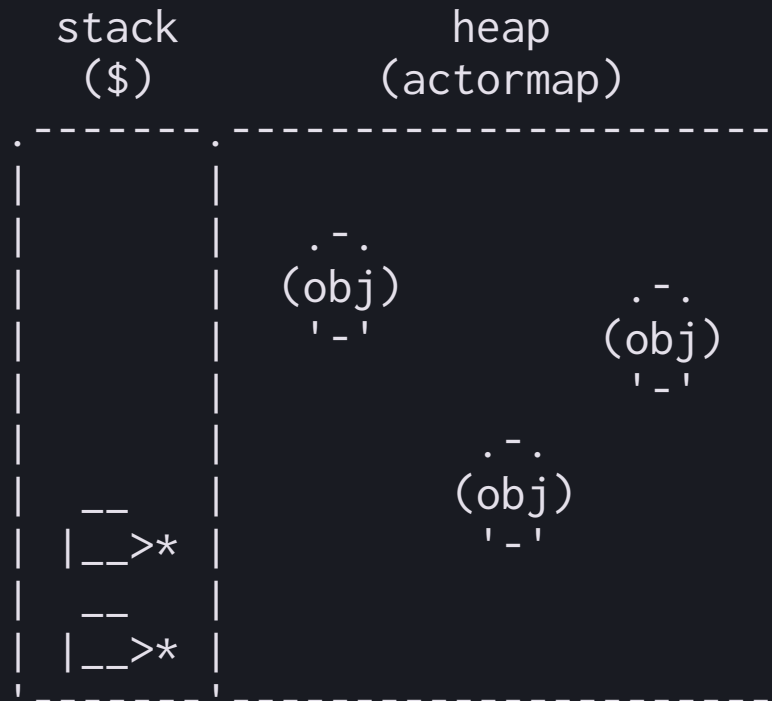
The Synchronous Call-Return Worldview



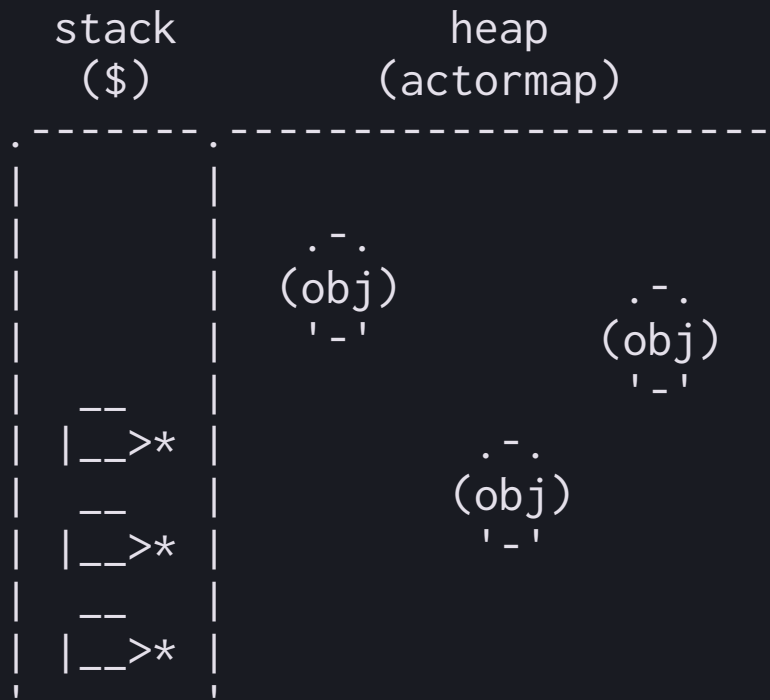
The Synchronous Call-Return Worldview



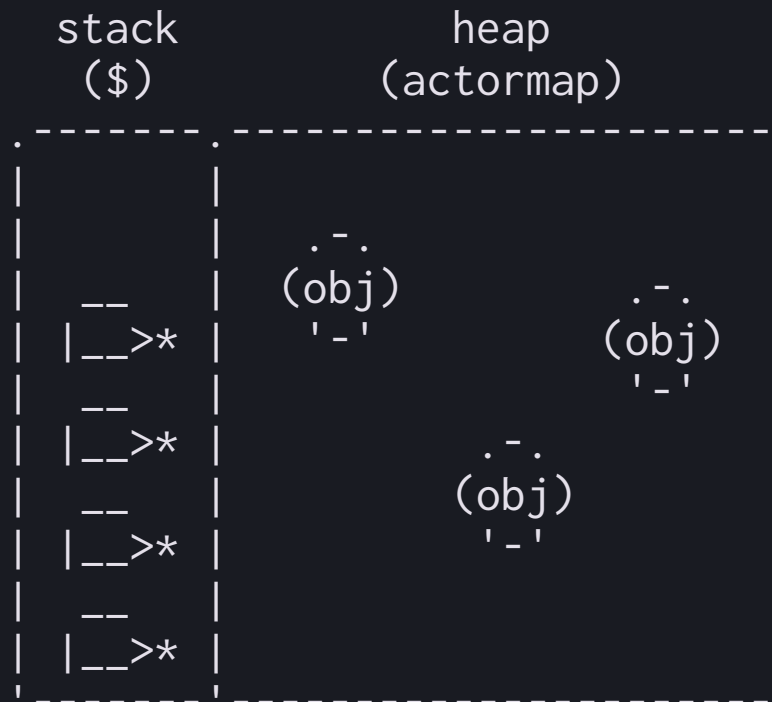
The Synchronous Call-Return Worldview



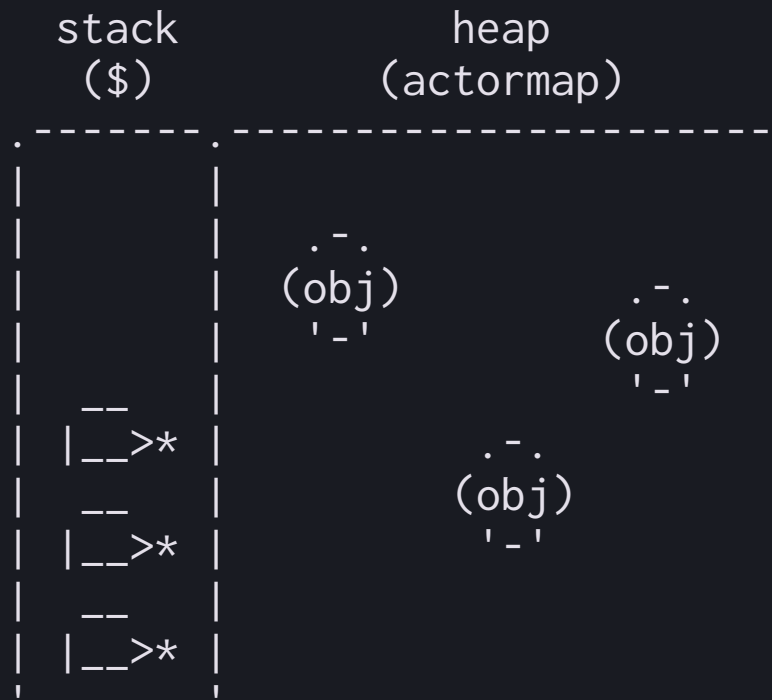
The Synchronous Call-Return Worldview



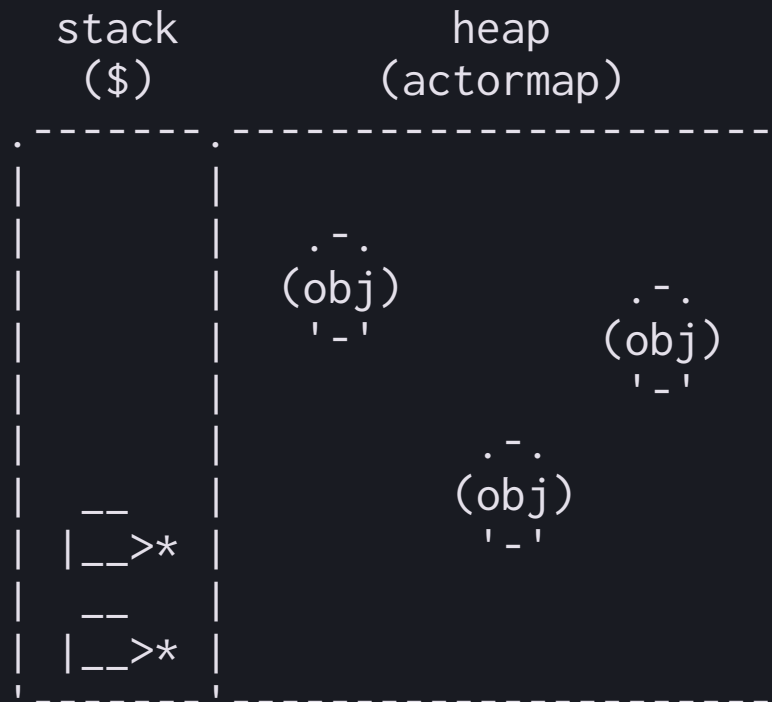
The Synchronous Call-Return Worldview



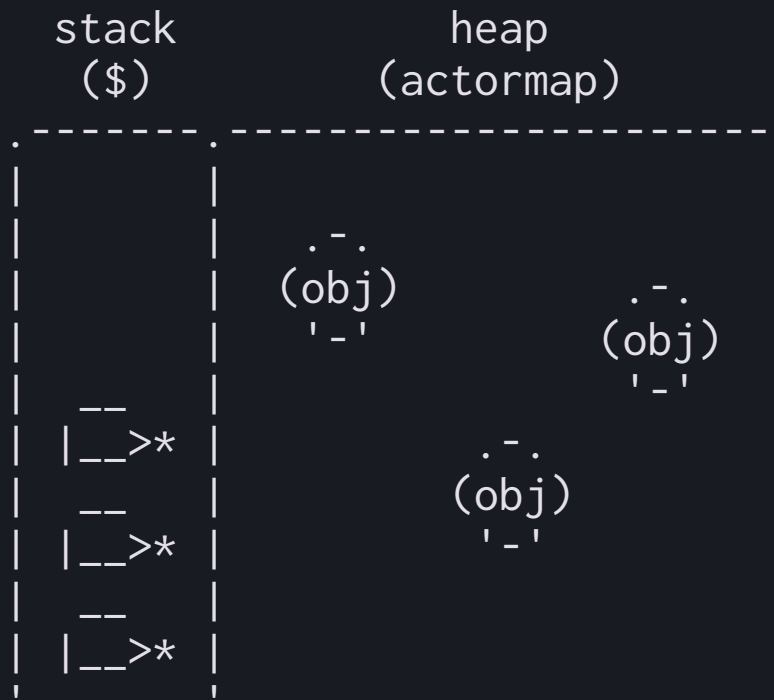
The Synchronous Call-Return Worldview



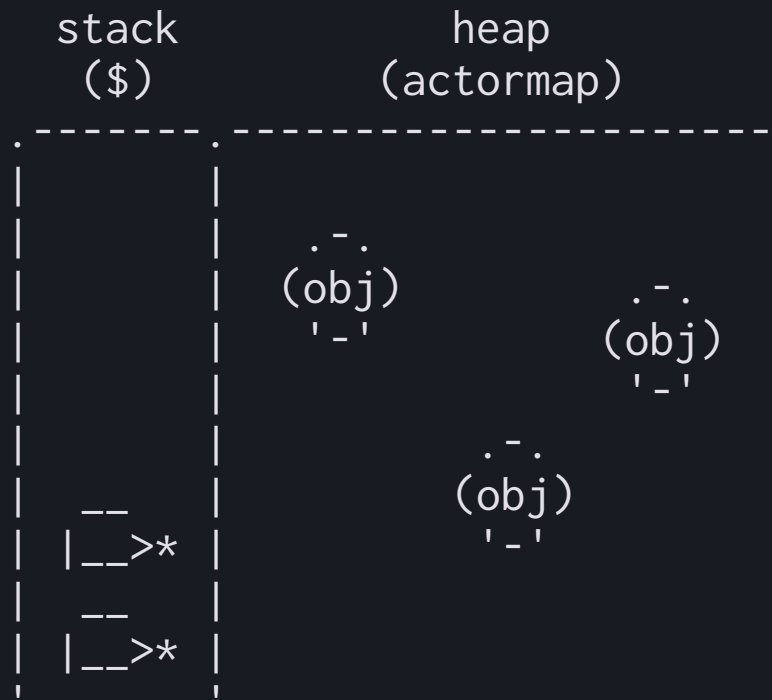
The Synchronous Call-Return Worldview



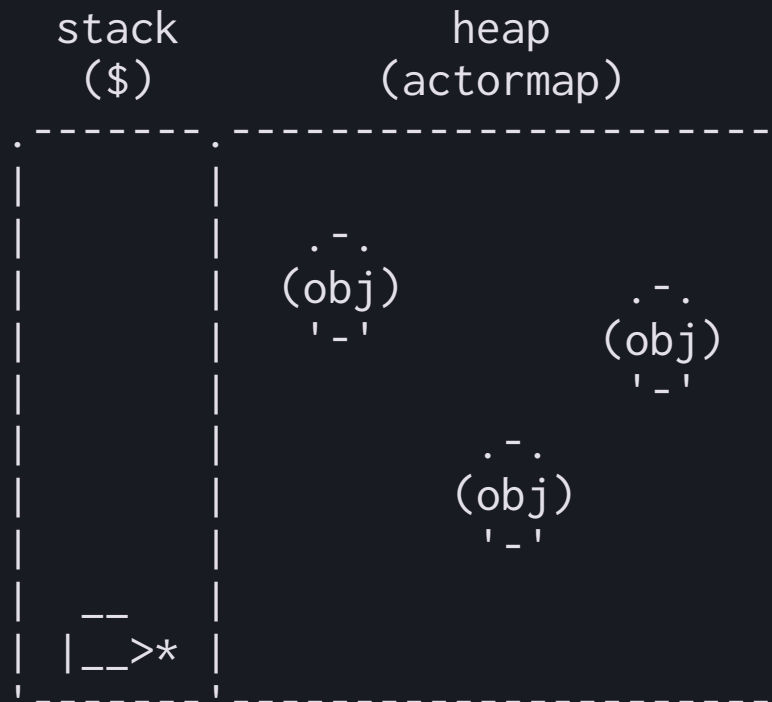
The Synchronous Call-Return Worldview



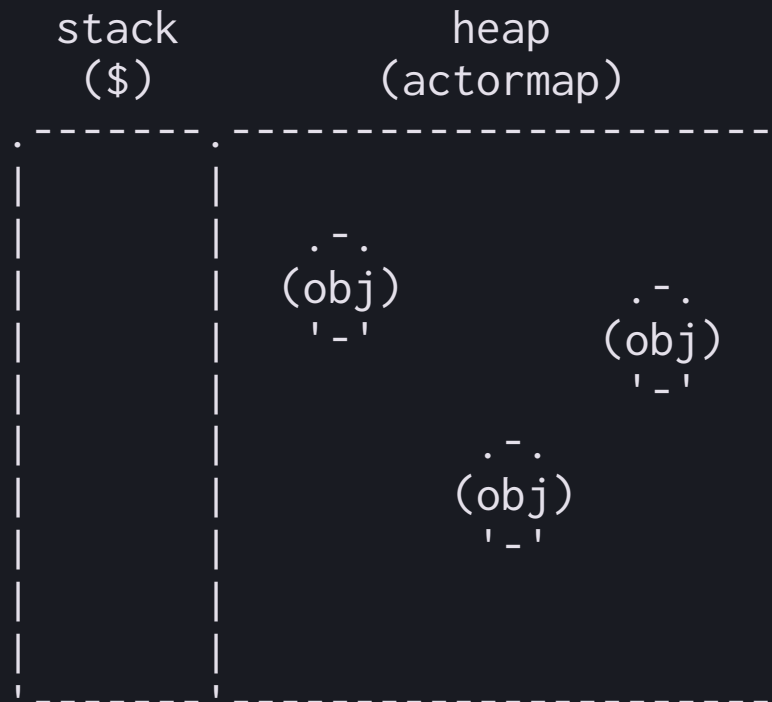
The Synchronous Call-Return Worldview



The Synchronous Call-Return Worldview



The Synchronous Call-Return Worldview



The Eventual Send Worldview

- ~193X: λ
- ~1958: $(\lambda () \dots)$
- ~1972: Smalltalk
- ~1973: Actors
- ~1975: Scheme
- ~1995: Joule
- ~1997: E (... kinda)

The Eventual Send Worldview



The Eventual Send Worldview



The Eventual Send Worldview



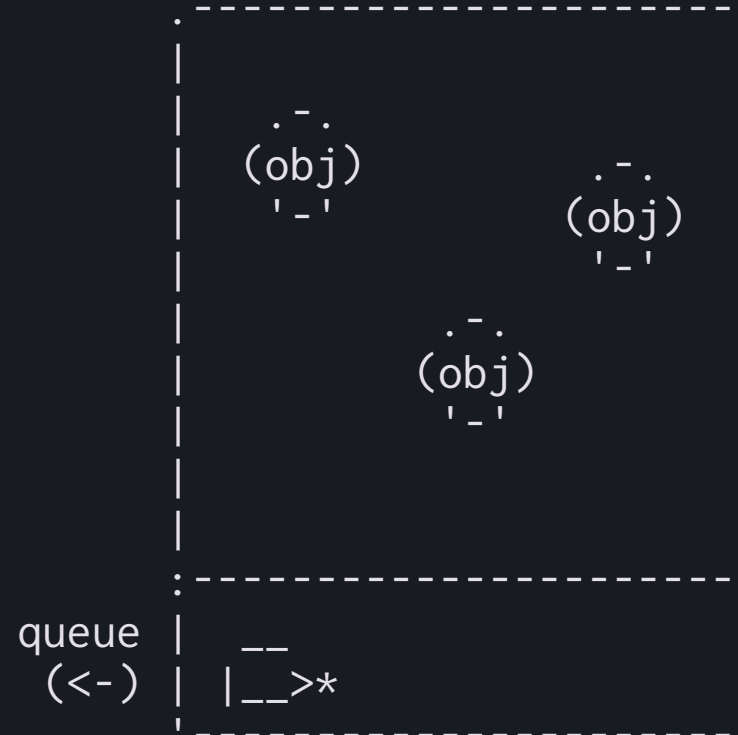
The Eventual Send Worldview



The Eventual Send Worldview



The Eventual Send Worldview



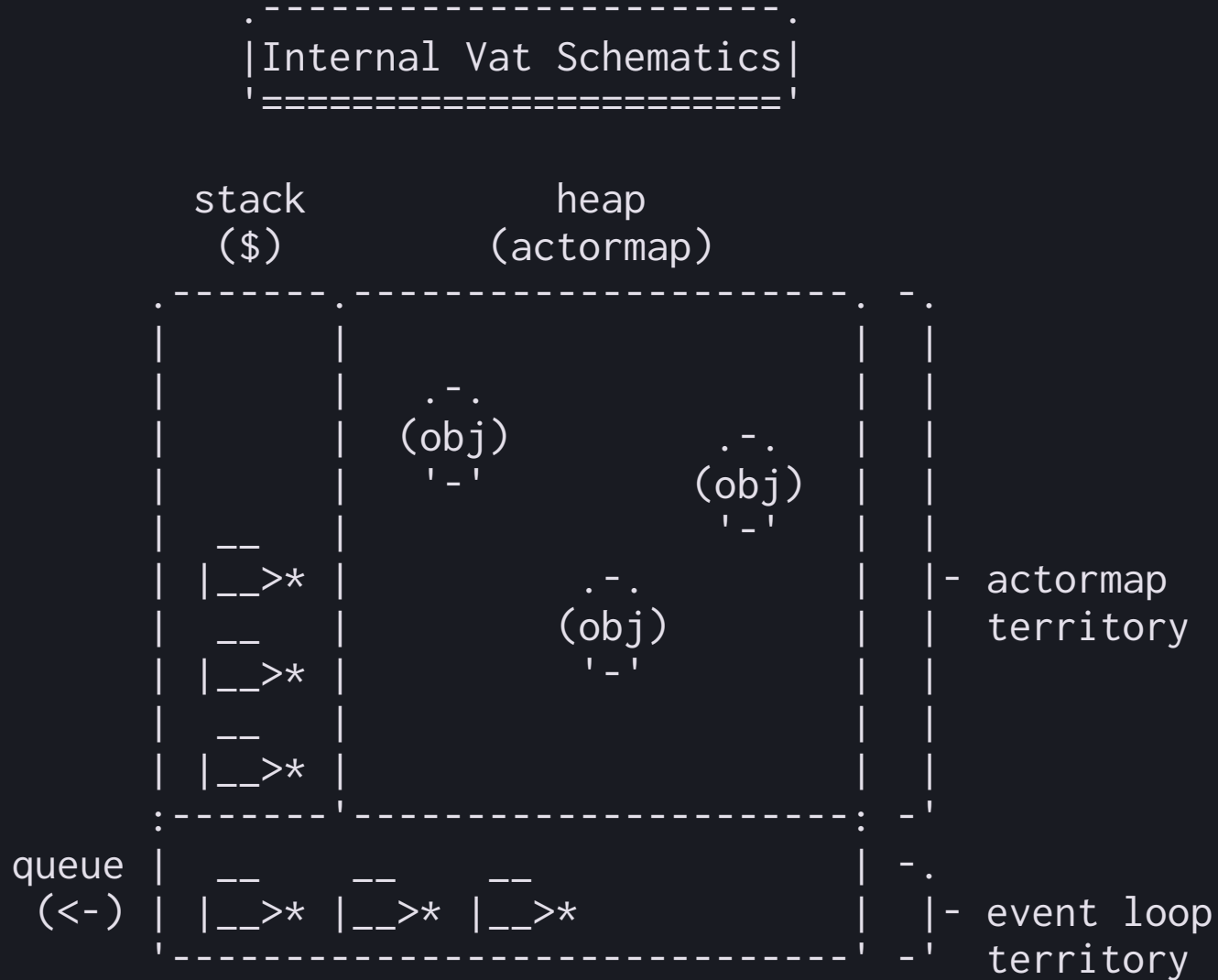
The Eventual Send Worldview



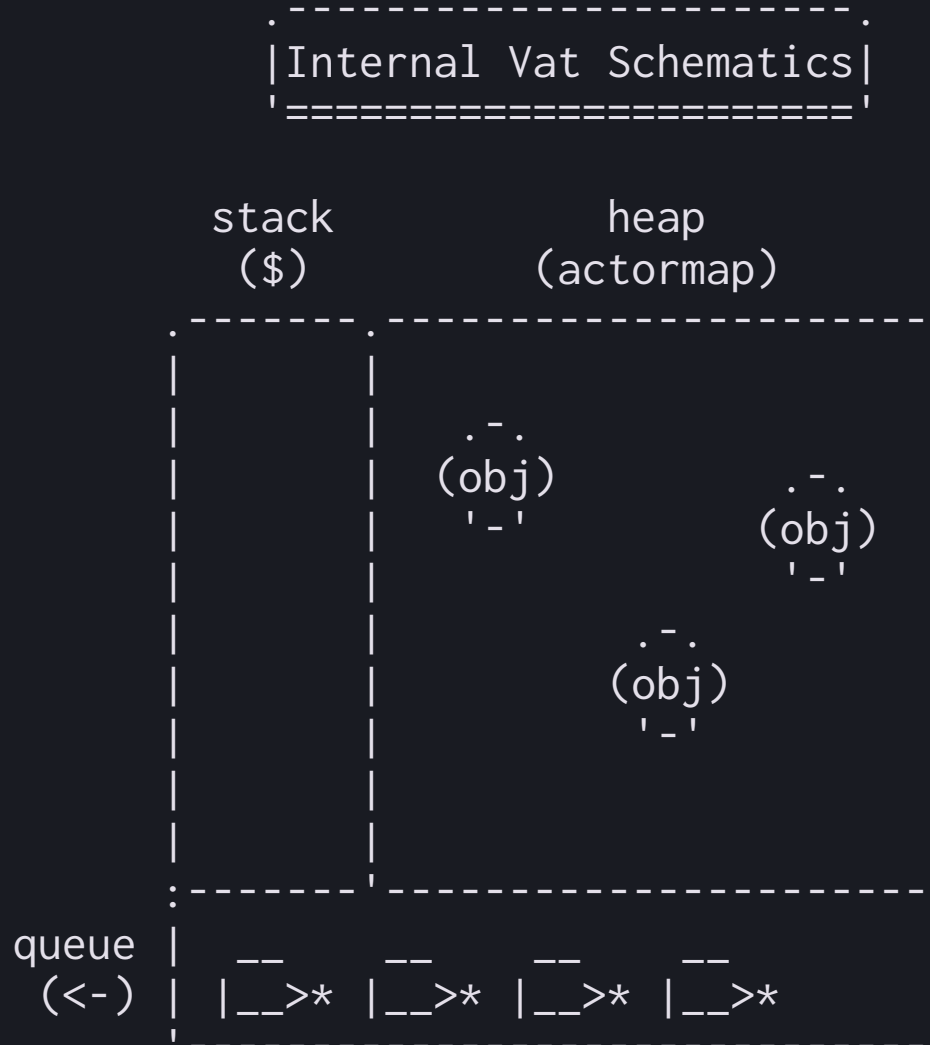
The Hybrid Worldview ("The Vat Model")

- ~193X: λ
- ~1958: (λ () ...)
- ~1972: Smalltalk
- ~1973: Actors
- ~1975: Scheme
- ~1995: Joule
- ~1997: E
- ~2020: Goblins

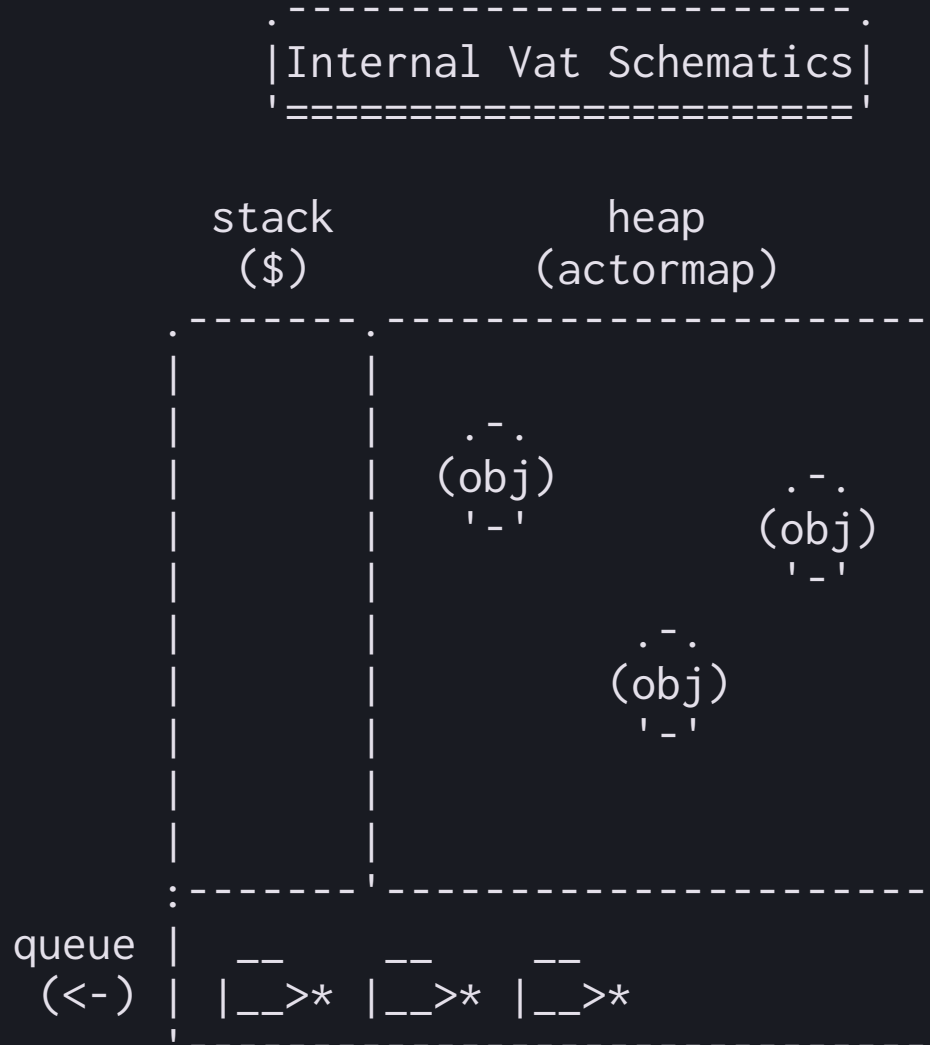
The Hybrid Worldview ("The Vat Model")



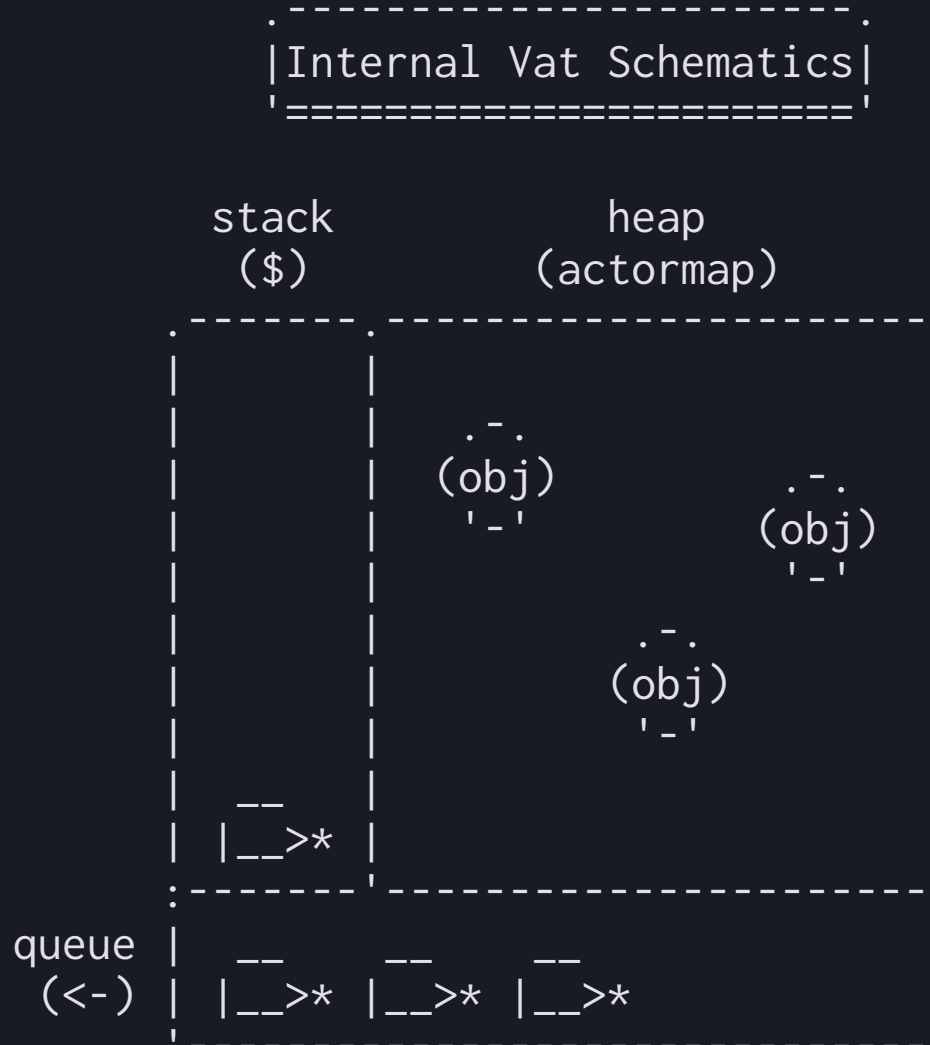
The Hybrid Worldview ("The Vat Model")



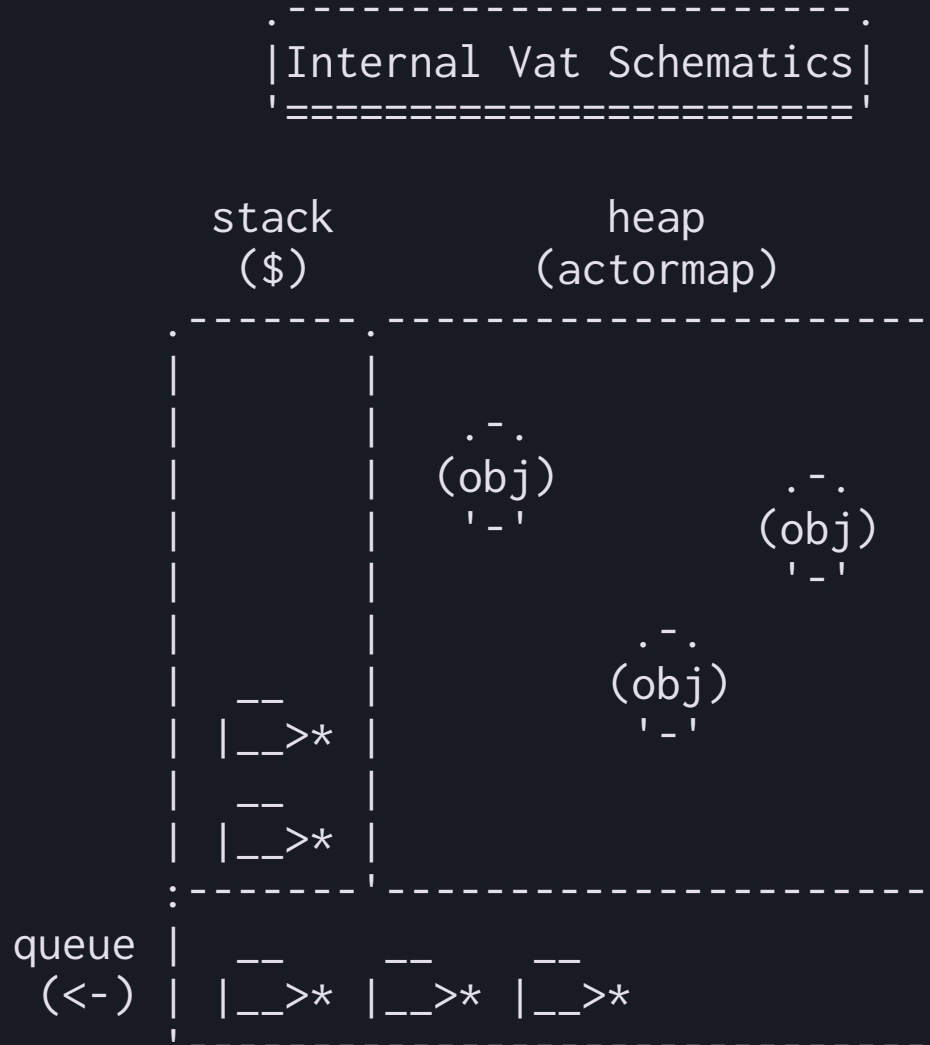
The Hybrid Worldview ("The Vat Model")



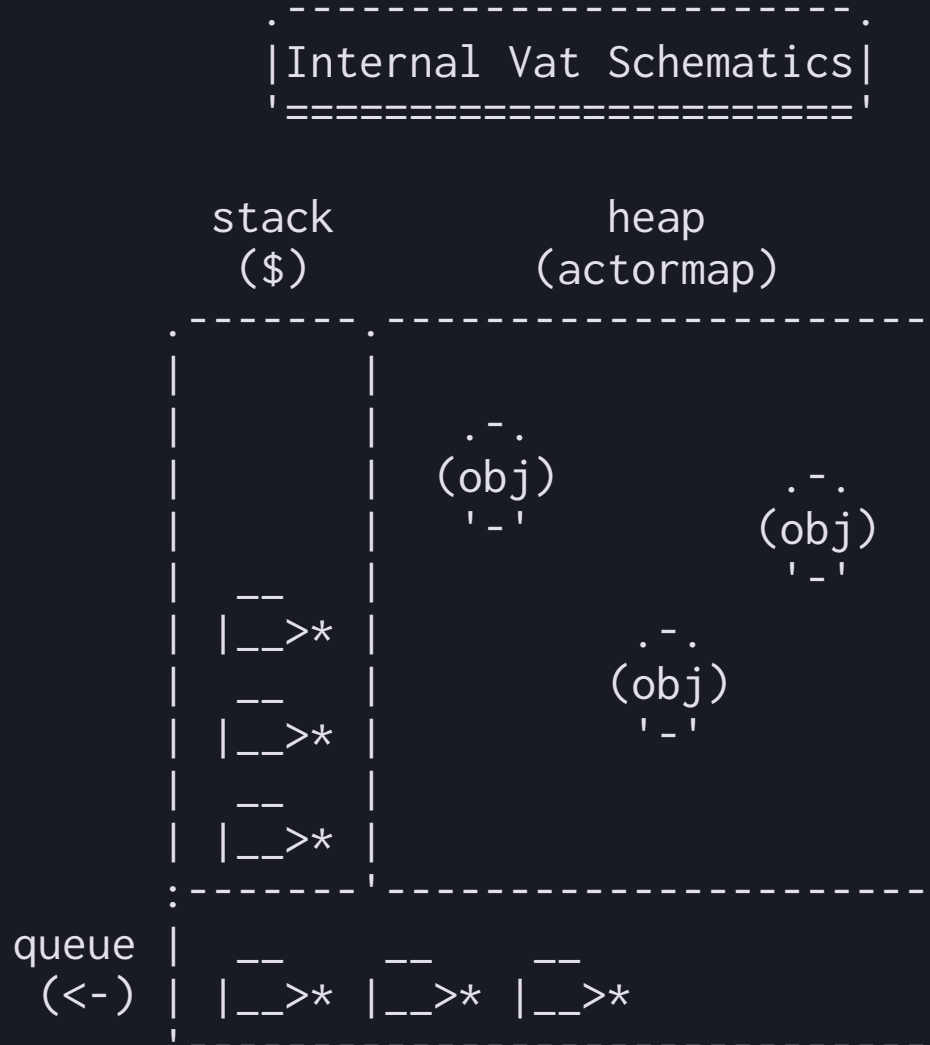
The Hybrid Worldview ("The Vat Model")



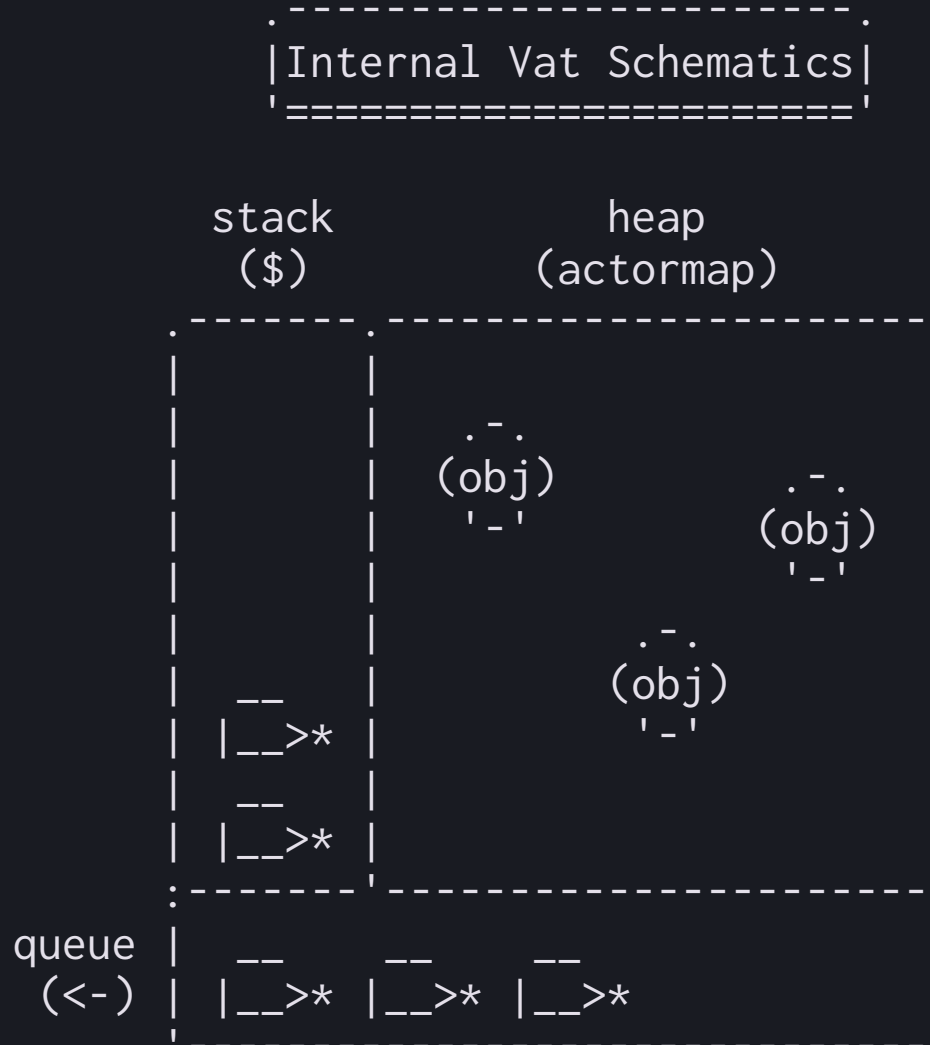
The Hybrid Worldview ("The Vat Model")



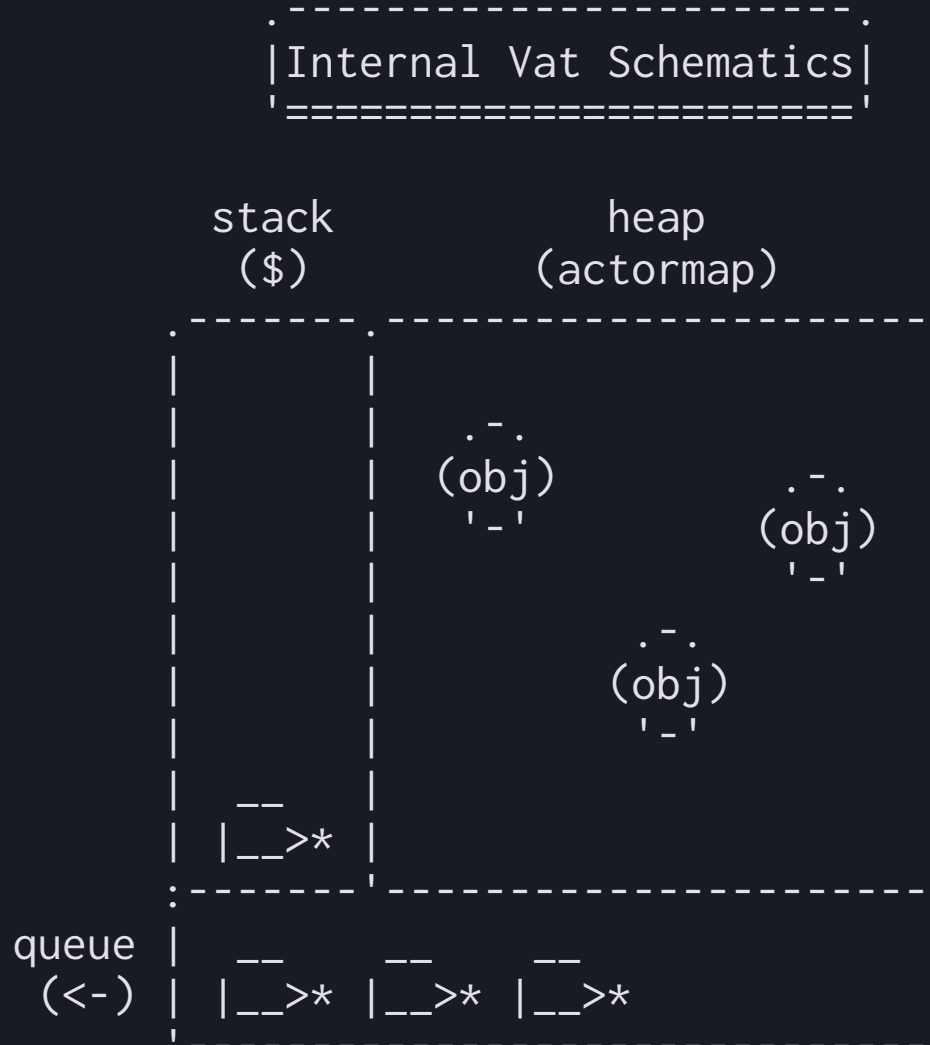
The Hybrid Worldview ("The Vat Model")



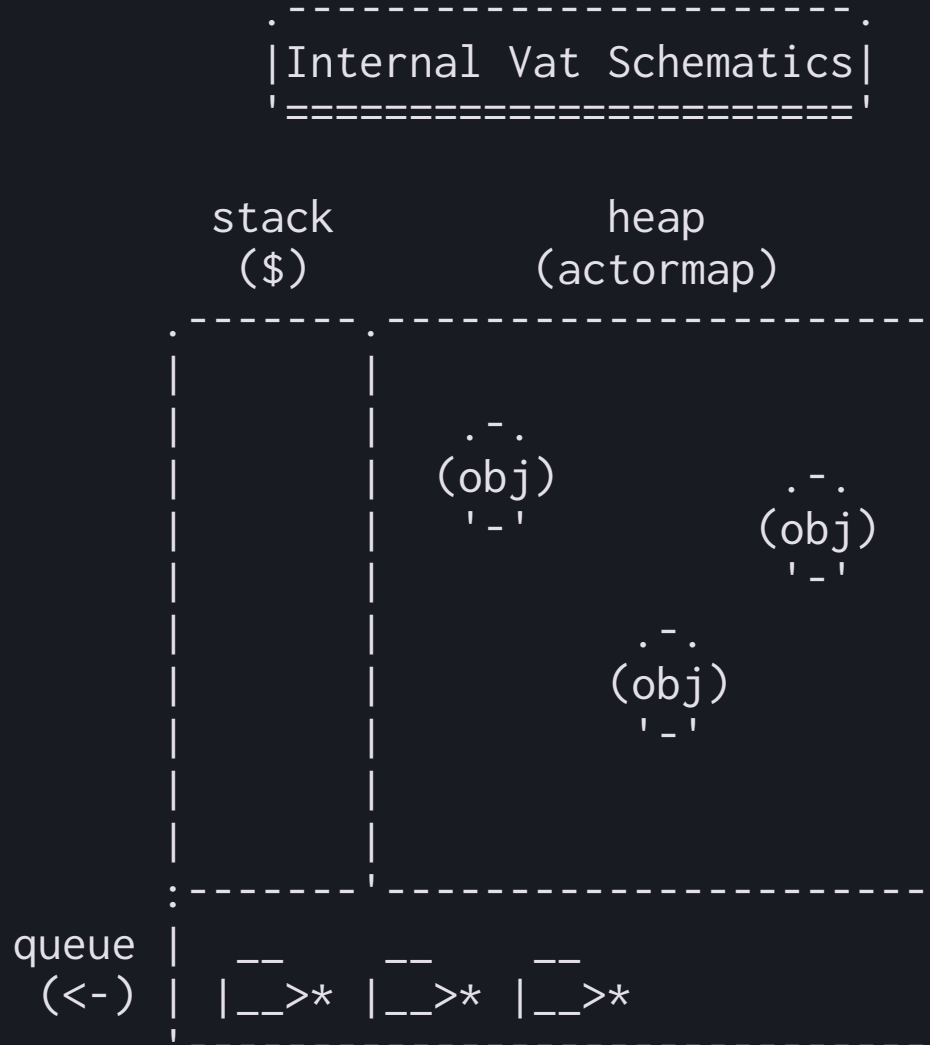
The Hybrid Worldview ("The Vat Model")



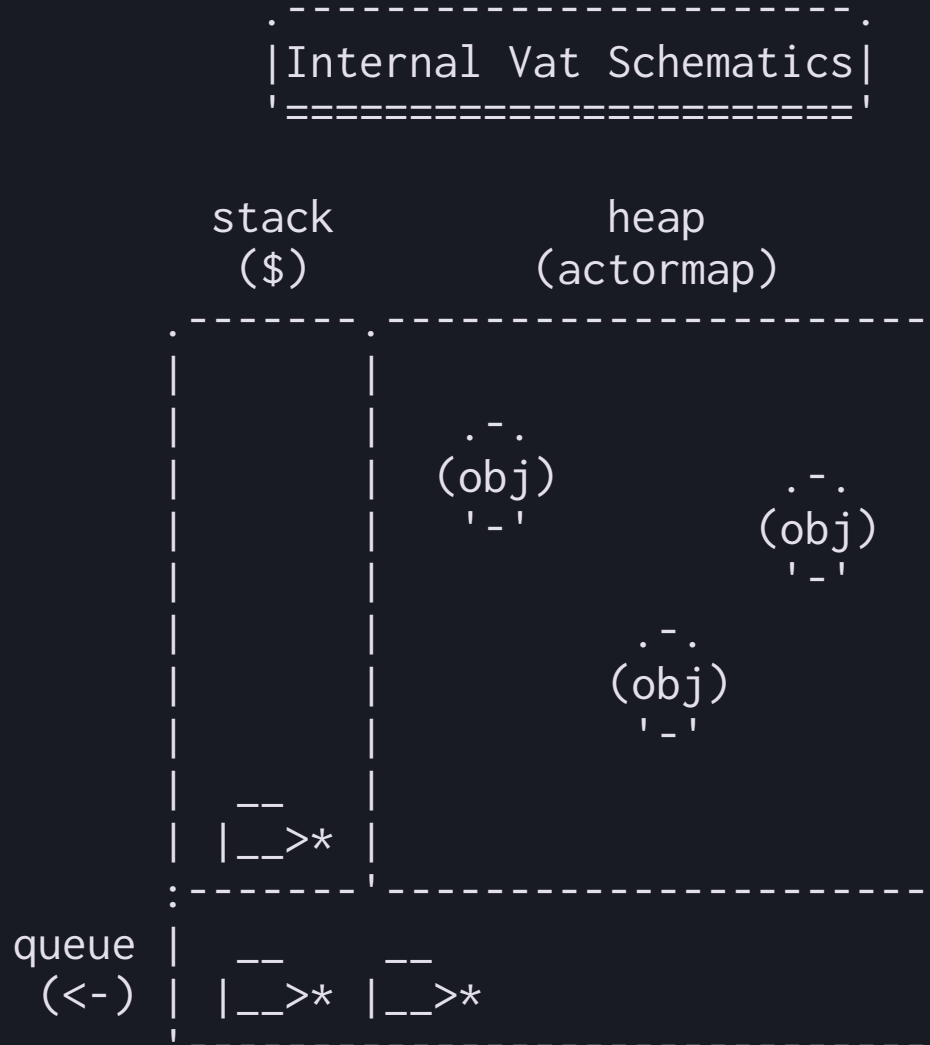
The Hybrid Worldview ("The Vat Model")



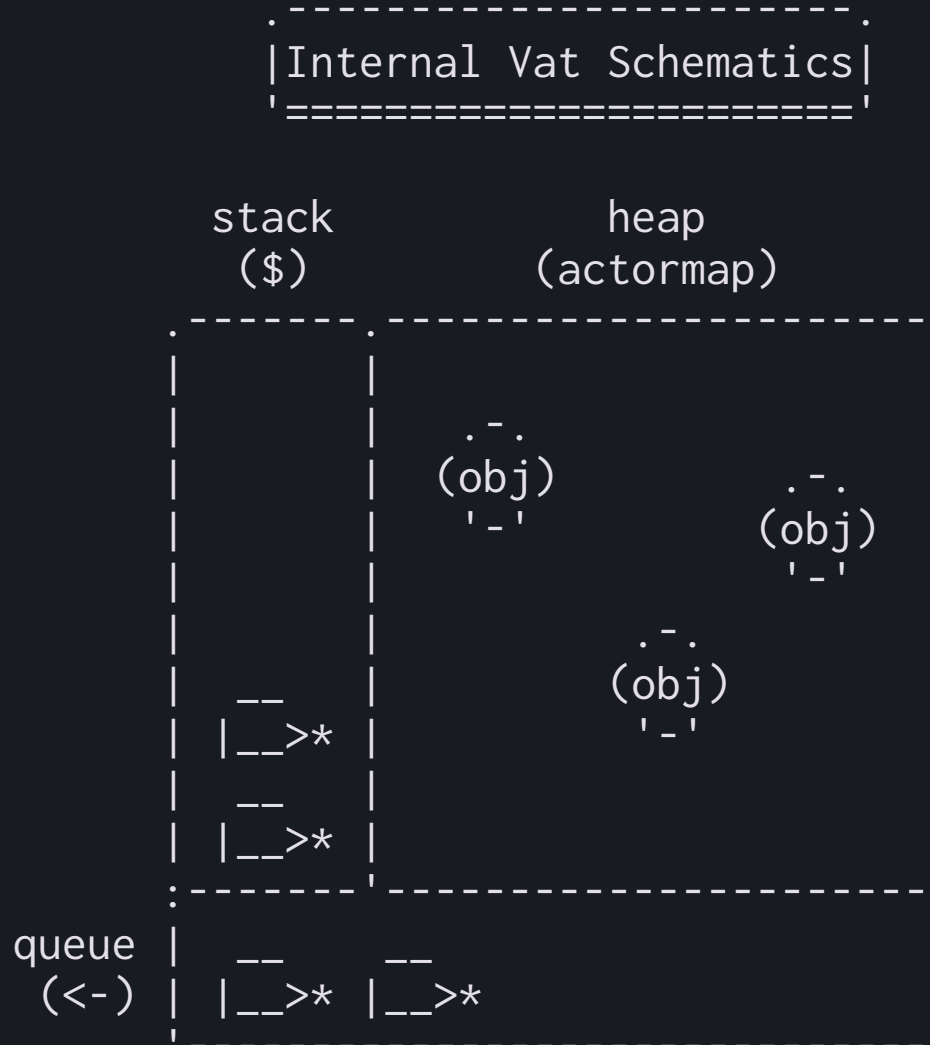
The Hybrid Worldview ("The Vat Model")



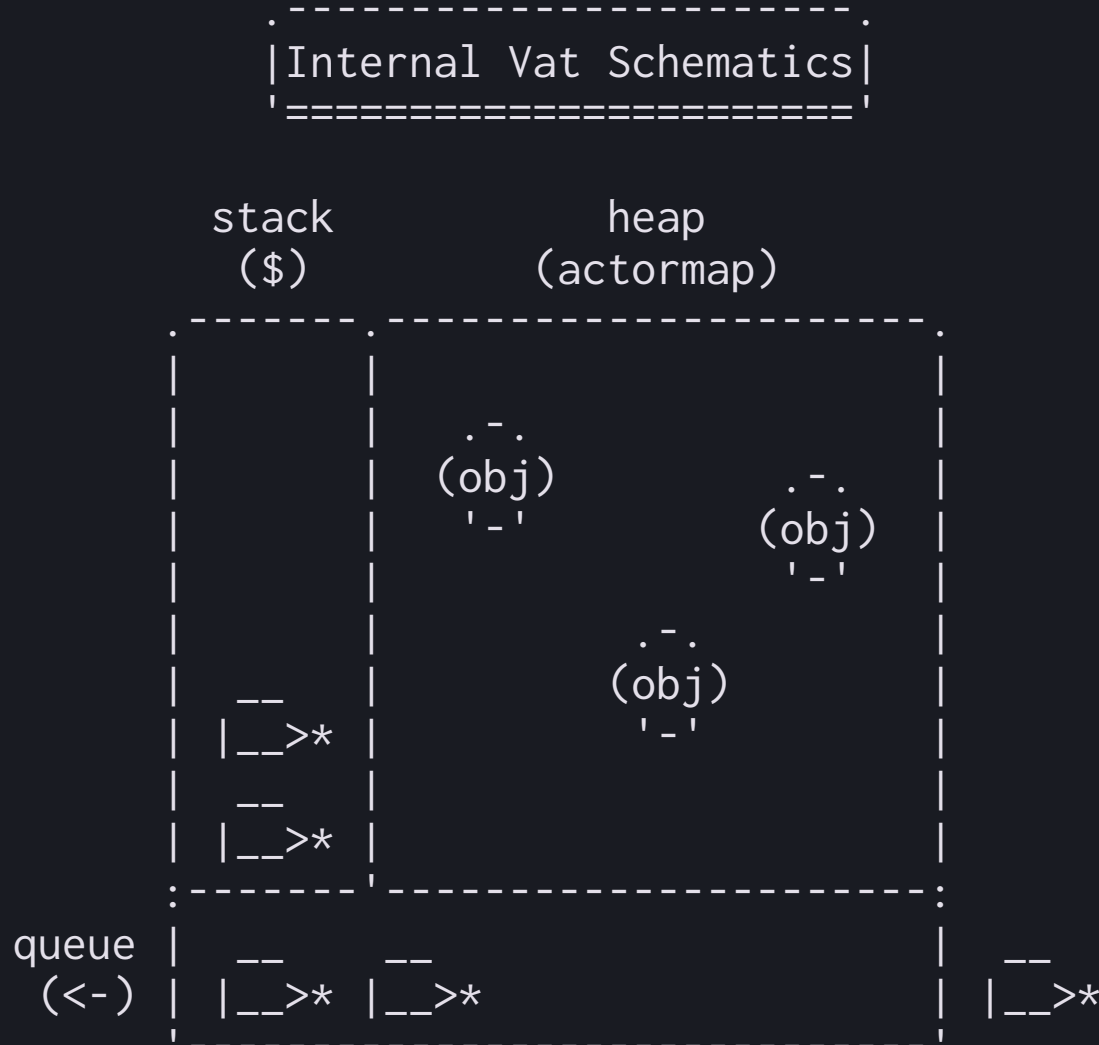
The Hybrid Worldview ("The Vat Model")



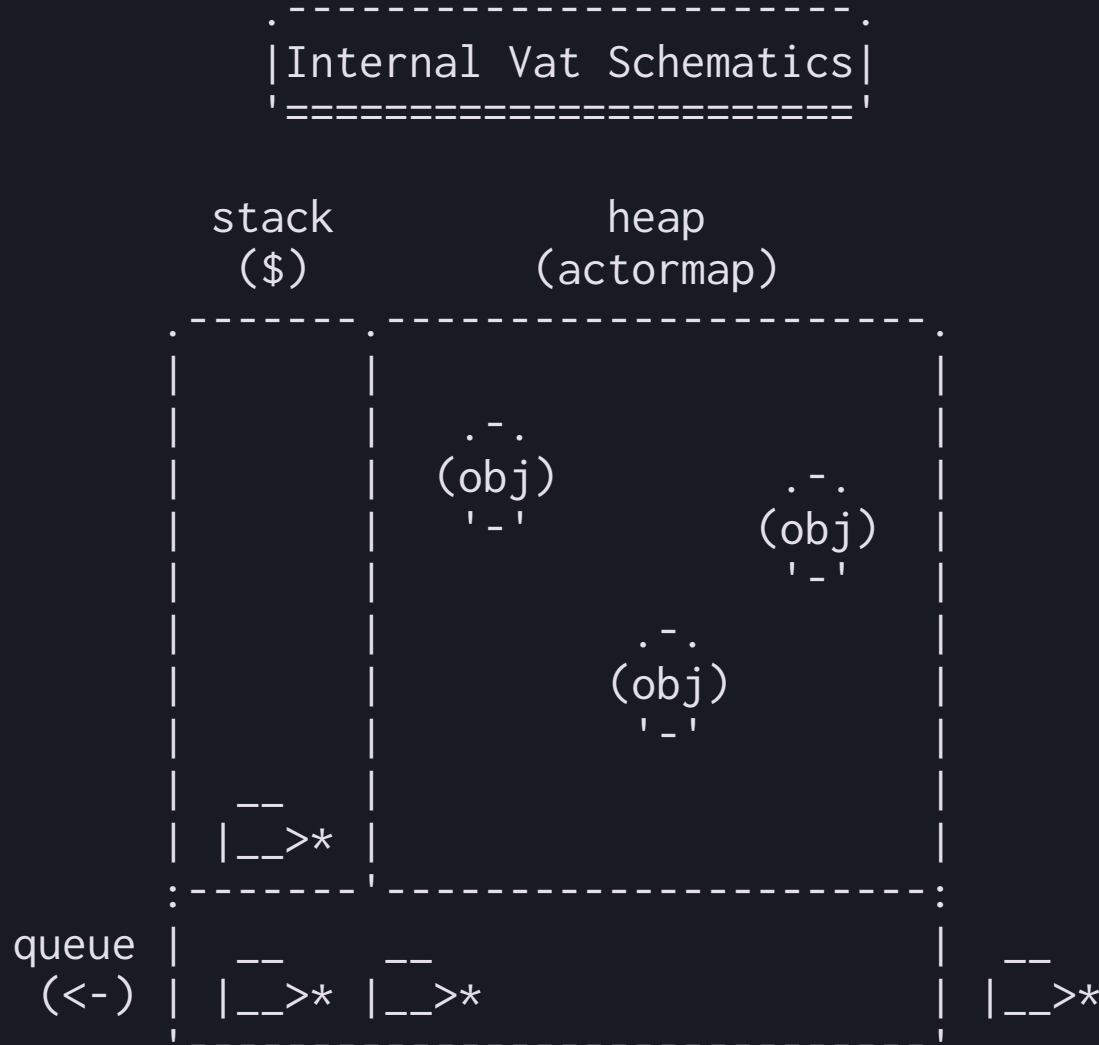
The Hybrid Worldview ("The Vat Model")



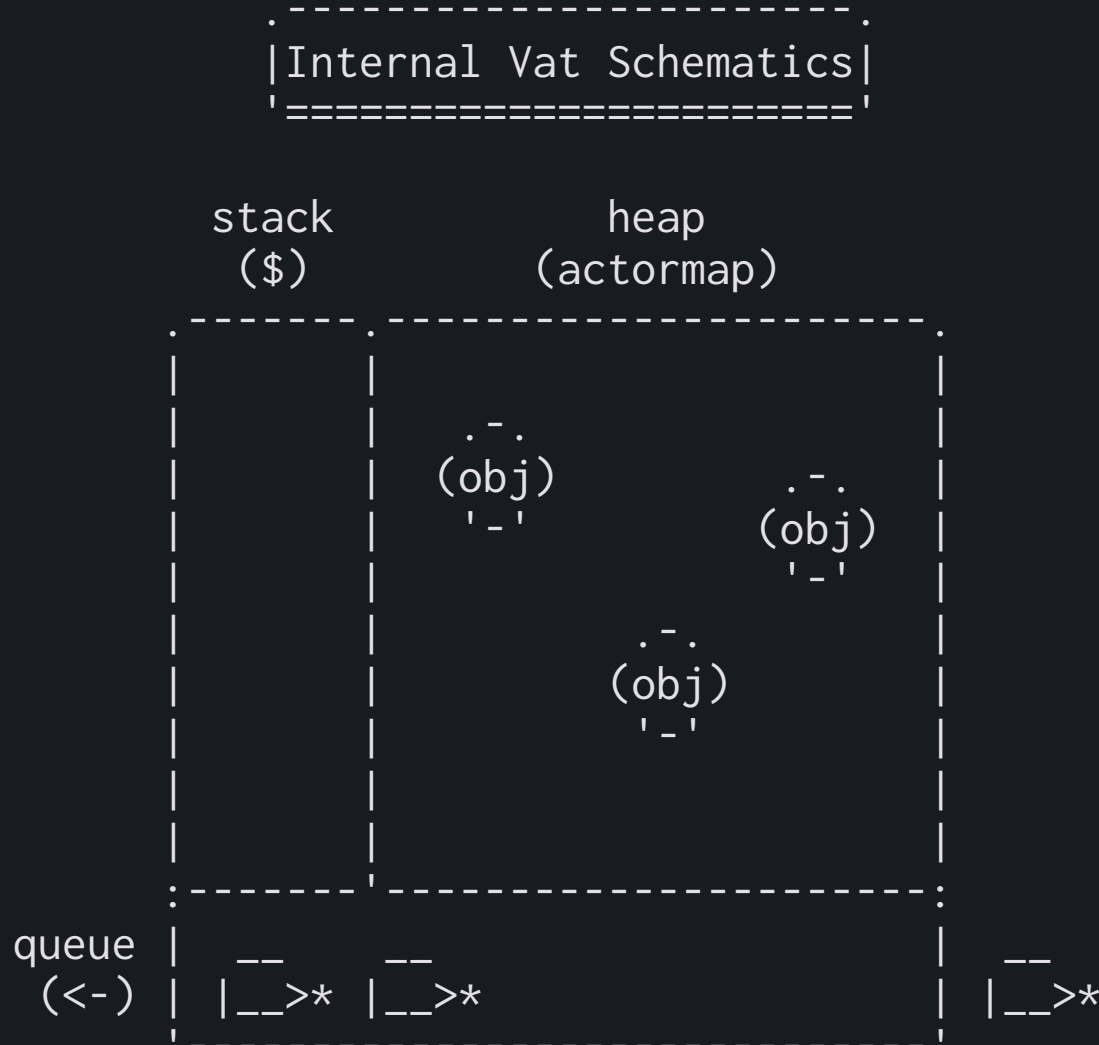
The Hybrid Worldview ("The Vat Model")



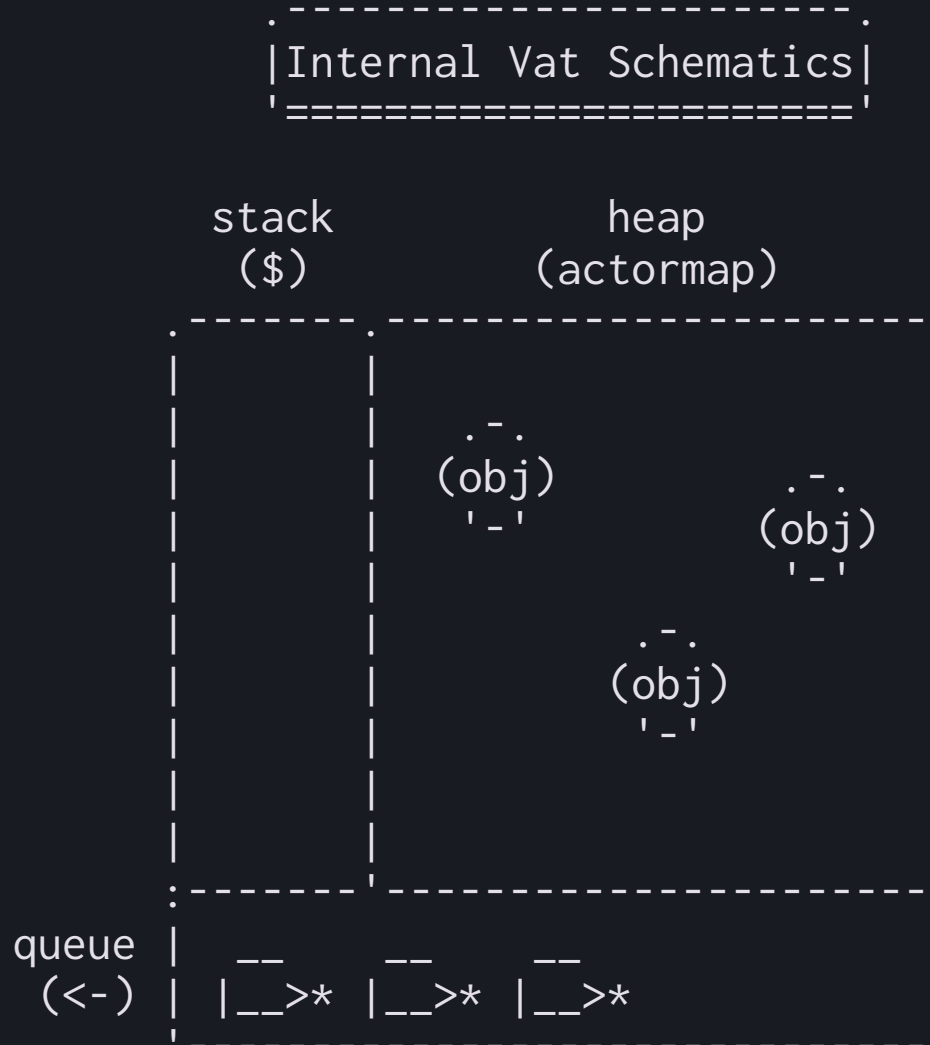
The Hybrid Worldview ("The Vat Model")



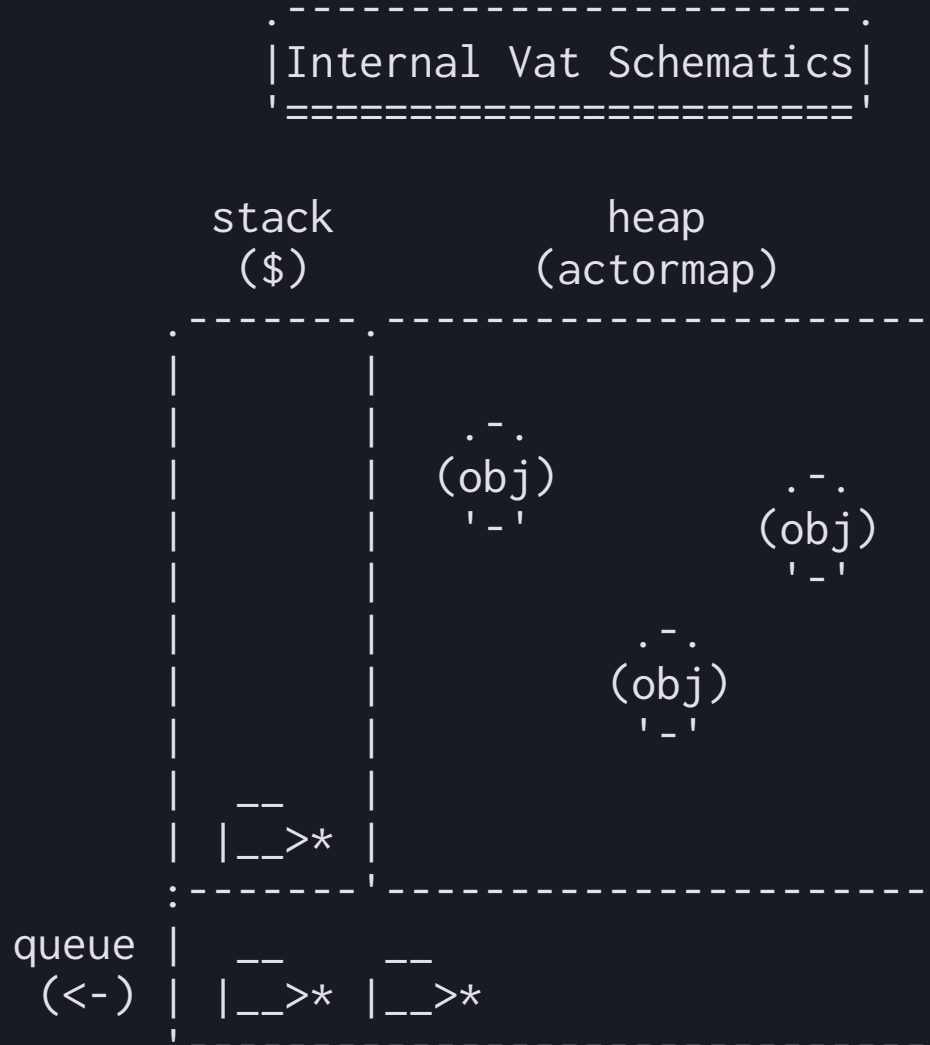
The Hybrid Worldview ("The Vat Model")



The Hybrid Worldview ("The Vat Model")



The Hybrid Worldview ("The Vat Model")



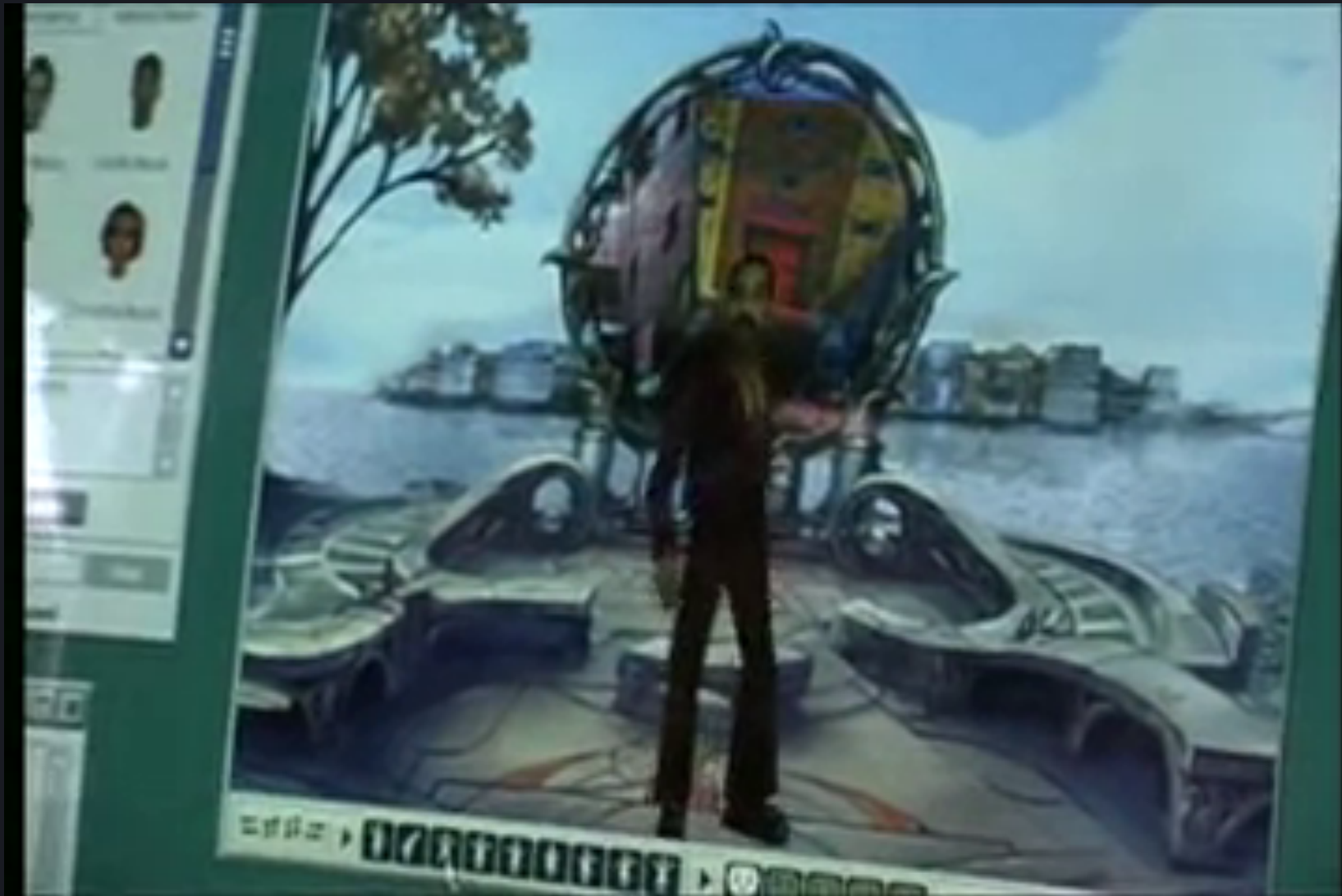
The Vat Model Combines Both Worlds

Call-Return (\$)	Eventual Send (<-)
Same vat only	Any actor, anywhere
Immediate	Eventual
Return values	Promises
Transactions	Distributed w/o deadlocks



Fantasy

decentralized virtual worlds



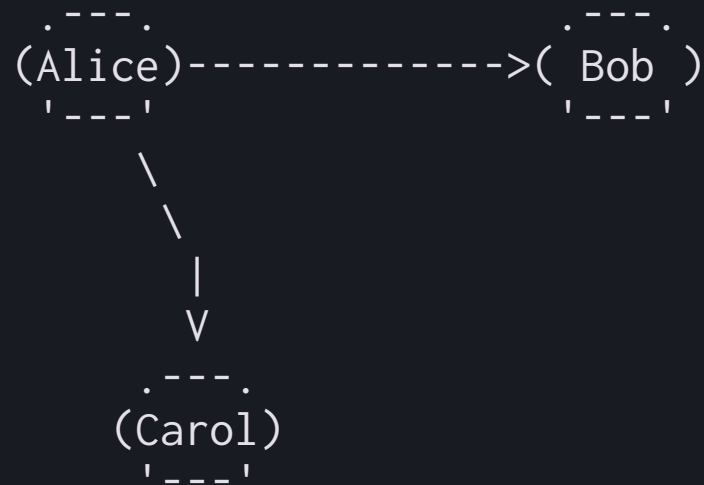


Open Source Distributed Capabilities

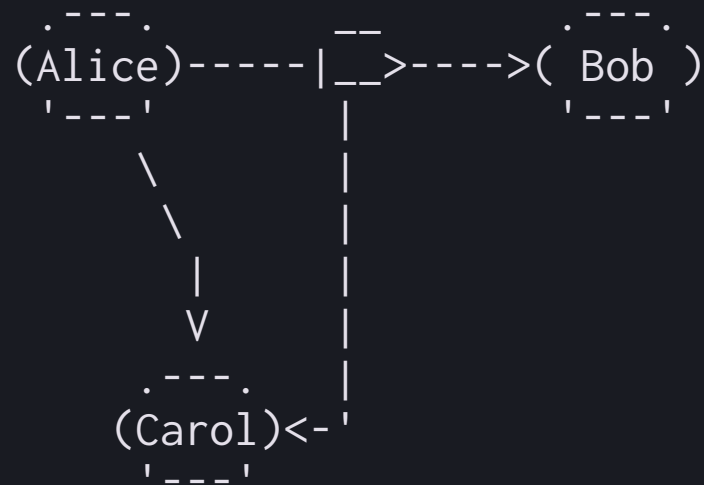
Welcome to *ERights.org*, home of **E**,
the secure distributed persistent language
for capability-based smart contracting.

[Quick Start](#) | [What's New?](#) | [What's **E**?](#)
[Smart Contracts](#) | [History & Talks](#) | [Feedback](#)

Object Capability Security



Object Capability Security



Object Capability Security



Lambda: The Ultimate Security Model

A.I. Memo No.
1564

MASSACHUSETTS INSTITUTE OF
TECHNOLOGY
ARTIFICIAL INTELLIGENCE
LABORATORY

March
1996

**A Security Kernel Based on the Lambda
Calculus**

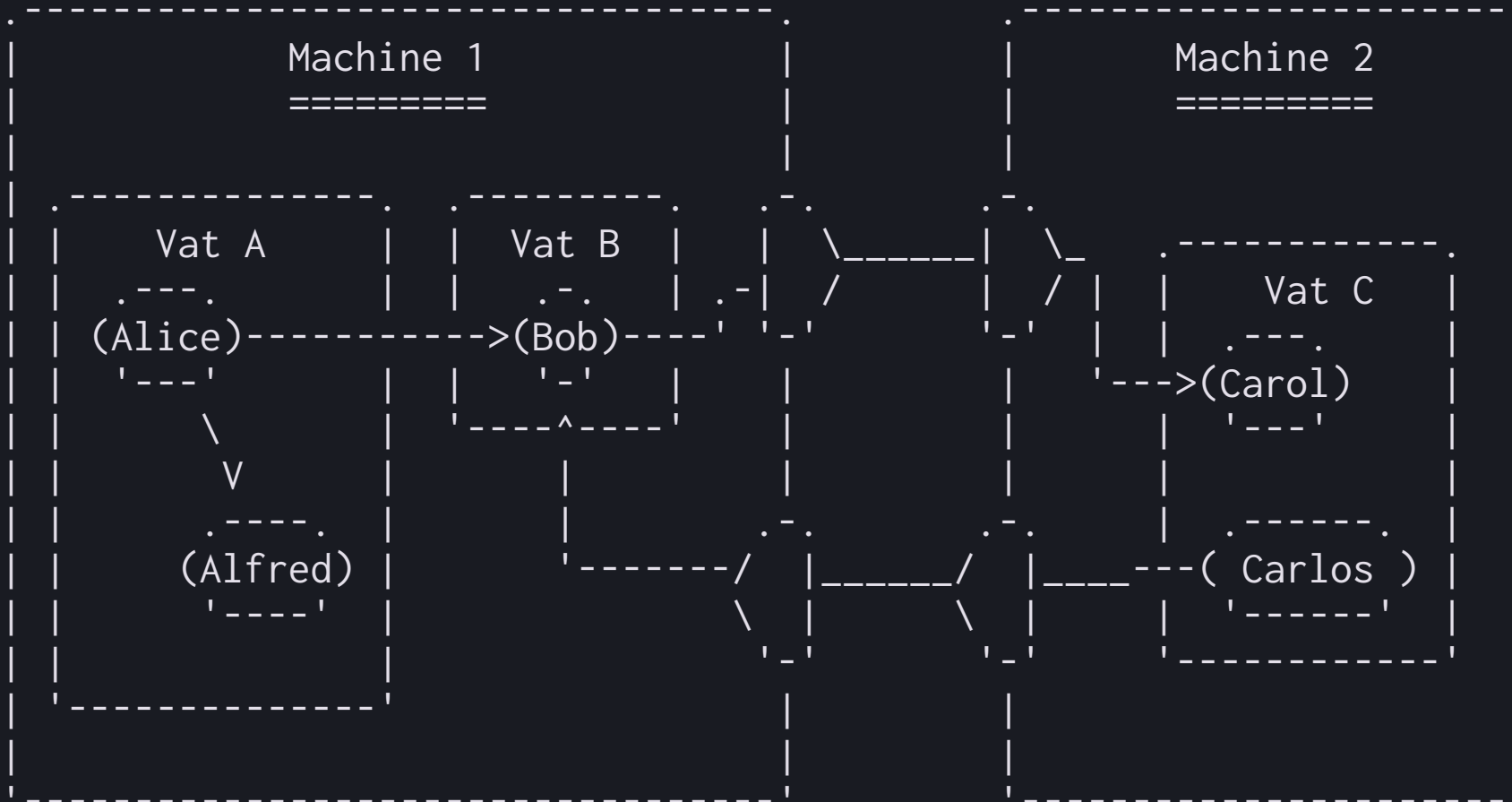
Jonathan A. Rees

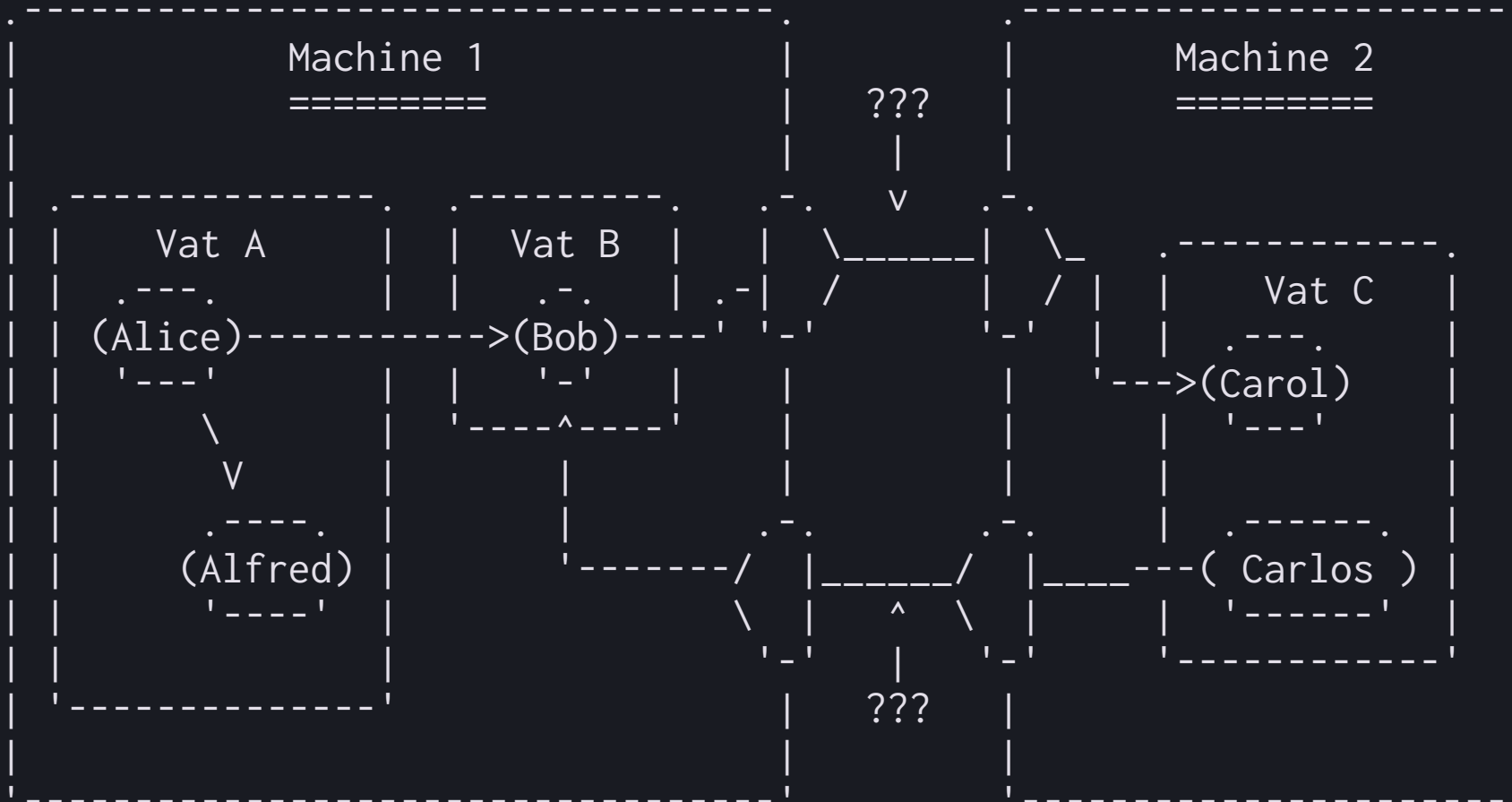
This publication can be retrieved by anonymous ftp to [publications.ai.mit.edu](ftp://publications.ai.mit.edu).

Abstract

Cooperation between independent agents depends upon establishing a degree of security. Each of the cooperating agents needs assurance that the cooperation will not endanger resources of value to that agent. In a computer system, a computational mechanism can assure safe cooperation among the system's users by mediating resource access according to desired security policy. Such a mechanism, which is called a *security kernel*, lies at the heart of many operating systems and programming environments.

What's in your scope? That's your authority.





* 4th wall breakage again... *