

Agents

(Proposal or spec clarification?)

Lars T Hansen, Mozilla

Ecma TC39 March 2016

(axis-of-eval.org/shmem/agents-formatted.html)

Overview

Agents are “bundles of execution machinery”

Shared memory spec must address interactions of concurrent executions

ES2016 does not have sufficient machinery

“Agents” restate existing machinery, add more

Motivation

Agent interaction shows up in the SAB spec

- An agent can block (wait until woken)
- An agent can unblock (wake) other agents
- These ideas show up in the semantics
- Having a concrete idea of agent is necessary

Other uses for agents in SAB spec too

What's in it?

General ideas

- Agent properties; the Agent Record
- Execution thread; concurrency; forward progress

For shared memory

- Agent clusters -- agents that share memory
- Suspend-together semantics for clusters
- Termination requirements for clusters

Agent definition

An “agent” is a bundle:

- set of execution contexts
- execution context stack
- running execution context
- job queues
- Agent Record holding properties
- execution thread, which may be shared

All ES code runs in a *surrounding agent*

Agent Record

The Agent Record has some simple properties:

- `[[LittleEndian]]`
- `[[State]]`
- `[[CanBlock]]`
- `[[Signifier]]`
- `[[IsLockFree1]]`, `[[IsLockFree2]]`

Could also hold execution engine state!

Agent Semantics

Agents:

- are independent
- have a forward-progress guarantee
- can block

Agents that share an execution thread:

- have less independence
- have a soft forward-progress guarantee
- can't block

Agent Clusters

Agents that are in principle able to communicate through shared memory form an agent cluster.

Agents in a cluster share `[[LittleEndian]]` and `[[IsLockFree]]` values and have distinct `[[Signifier]]` values.

Agents in a cluster are coupled for suspension, termination.

Suspension and Termination

Agents in a cluster must suspend together

- This limits memory sharing

Agents in a cluster must be:

- Killed together, or
- There must exist a kill notification mechanism

(Discuss - are these rules viable?)

Open questions

Merge execution state into Agent Record?

Move the parts about shared memory back to the shared memory spec? (Agent cluster, suspension and termination concerns)

Is the agents spec a full proposal or an adjustment to the existing spec?

Status & Progress

Spec has received attention (Domenic, Mark)

Spec is fairly stable

The easy bits could be turned into a PR now

The hard bits must wait for the shmem spec