The Actor Model

1. Actor Model Introduction
2. Rust Actor Model Frameworks
3. Intro to Tauri and its Actor Model Approach

Introduction to Actors:

actors encapsulate state and behaviour

The actor model in computer science is a mathematical model of concurrent computation that treats the actor as the universal primitive of concurrent computation. In response to a message it receives, an actor can: make local decisions, create more actors, send more messages, and determine how to respond to the next message received. Actors may modify their own private state, but can only affect each other indirectly through messaging.

Wikipedia Page

So what is an actor?

An Actor is the fundamental unit of computation. Actors communicate solely through messages in an asynchronous fashion. An actor can perform three distinct actions based on the message it receives:

- send a finite number of messages to other actors
- create a finite number of new actors
- change its state or designate the behavior to be used for the next message it receives

Actors interact with each other by passing messages. There is no assumed order to the above actions, and they could be carried out concurrently. Two messages that are sent concurrently can arrive in either order.

https://riker.rs/actors/

What does an actor model system need?

- Supervisor
- Message Channels
- Messages
- Actors

> Share state by communicating messages instead of communicating via shared state.

A few Types of Channels

- Point to Point (simplex)
- Pub-Sub
- Message Bus (duplex)
- Intermediary facets
- ...

Some Message Patterns

- Command Message (tell the recipient what to do)
- Document Message (just share data)
- Event Message (pub-sub)

Book Recommendation:

Reactive Messaging Patterns with the Actor Model - Vaugn Vernon

Important concerns

- Only messages
- Async
- Internal state
- Validation
- Supervision
- Design your Channels like an ecosystem

Rust Actor Models

When choosing an actor model, consider your entire system and specifically the devices upon which the actors will be living.

When building your actors, try to architect them such that they do one thing and do that well.

Dining Philosophers

- Actix - https://actix.rs/
  - Fully featured
  - Battle tested (actix_web)
- Riker - https://riker.rs/ (great intro to actors)
  - Flexible
  - Best practices
- Bastion - https://bastion.rs/
  - Opinionated
- Stage - https://gitlab.com/encounter-vtt/stage/stage-core
  - Early alpha
- Kay - https://github.com/aeplay/kay
  - Very performant
  - WASM compatible