Elegant actor systems with xtra

tl;dr:

- docs.rs/xtra
- In-process actor library
- No network communication like akka
- Take advantage of Rust's type system
- Not my project (I am just a fan and contributor)
- High churn at the moment but 0.6 is coming!

tl;smtc (show me the code):

```rust
/// (1) Define the actor struct
struct Greeter;

/// (2) Define lifecycle behaviour
impl xtra::Actor for Greeter { type Stop = (); async fn stopped(self) -> Self::Stop {} }

/// (3) Define a message
struct Greet(&'static str);

/// (4) Define message handling behaviour
#[async_trait]
impl xtra::Handler<Greet> for Greeter {
  type Return = String;

  async fn handle(&mut self, msg: Greet, ctx: &mut xtra::Context<Self>) -> Self::Return {
    format!("Hello {}!", msg.0)
  }
}

/// (5) Make sure it works
#[tokio::test]
async fn can_greet() {
  let address = Greeter.create(None).spawn_global();

  let greeting = address.send(Greet("Rust-Sydney")).await.unwrap();

  assert_eq!("Hello Rust-Sydney!", greeting);
}
```

How is xtra different from `tokio::spawn(async { })`?

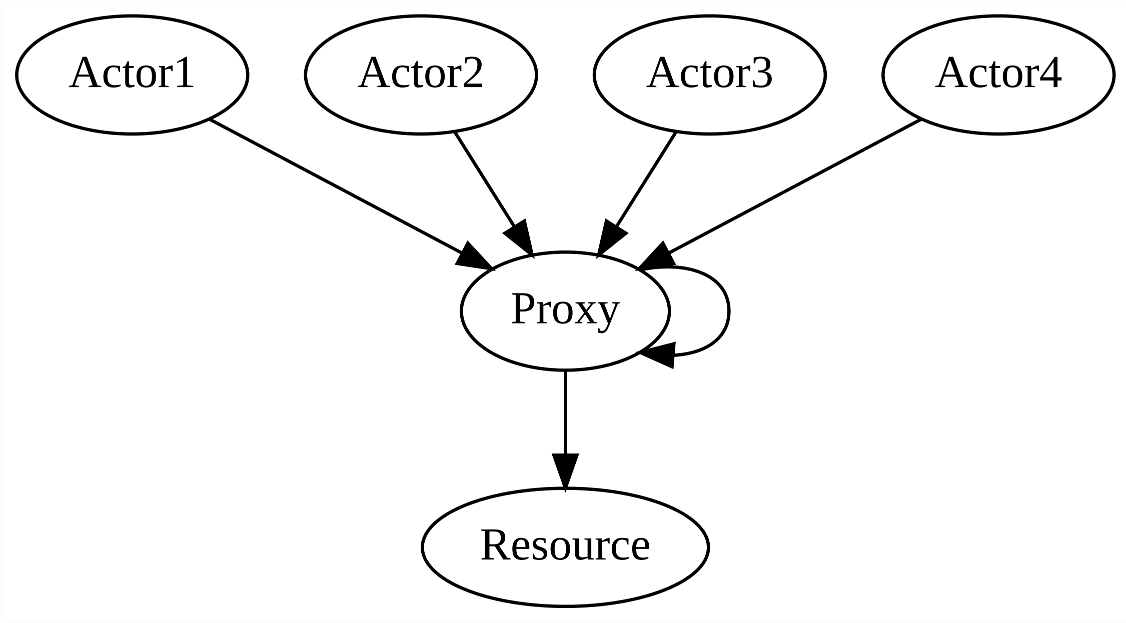
- Actors are tasks on steroids
  - Send messages to a task once spawned
  - Defined lifecycle functions allow for supervisors: *Have you tried turning it off and on again?*
  - Naming tasks as actors is easier for humans
- Easier instrumentation with prometheus / jaeger

Pattern: Send message to yourself

```rust
struct MyActor; // The actor.
struct Ping; // The message.

#[async_trait]
impl xtra::Actor for MyActor {
   type Stop = ();

   async fn started(&self, ctx: &mut Context<Self>) {
      // Grab an address to ourselves.
      let address = ctx.address().expect("we just started");

      // Spawn a task to ping ourselves every 100ms.
      tokio::spawn(async move {
          // Make sure the task stops if the actor stops, otherwise memory-leak!
          while let Ok(()) = address.send(Ping).await {
              tokio::time::sleep(Duration::from_millis(100)).await;
          }
      });
   }

   async fn stopped(self) -> Self::Stop { }
}
```

Pattern: Local-cache proxy

1. Refresh a local cache on interval
2. Other components can retrieve local copy without network request



Wrap-up

- xtra: In-process and thus type-safe actor library
- Small footprint
- async-await native
- Modular

The end

- github.com/Restioson/xtra
- #xtra-community:matrix.org
- Questions?