

Messaging

- Message functions (send and receive) are first class members of the language:

```
def rpc(name, request) do
  send(name, {self(), request})

  receive do
    {_name, response} -> response
  end
end
```

```
receive do
  {from, request} ->
    response = handleRequest(request)
    send(from, {self(), response})
end
```

Tail Recursion

- If the last thing a function does is call itself, there's no need to make the call. Instead, the runtime simply jumps back to the start of the function. If the recursive call has arguments, then these replace the original parameters.

```
def loop(name, state) do
  receive do
    {from, request} ->
      {response, newState} = handleRequest(request, state)
      send(from, {name, response})
      loop(name, newState)
  end
end
```