Tidbits

Section 8

In this Section, we are going to take a look at...

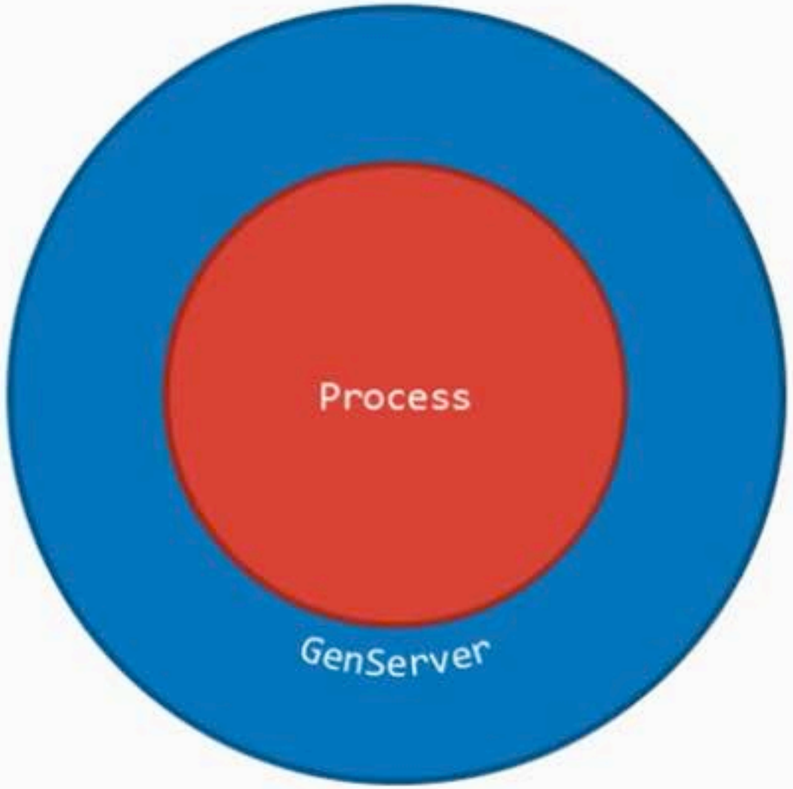
- Agents and tasks
- Umbrella applications
- Releases

Agents and Tasks

In this Video, we are going to take a look at...

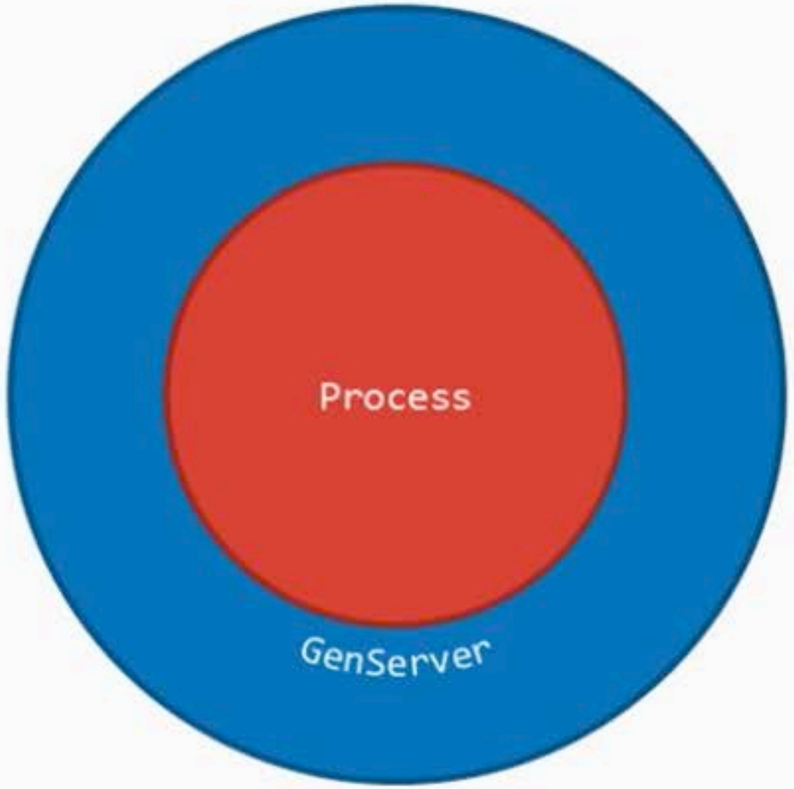
- Agents
- Tasks

Recap – The GenServer



Recap – The GenServer

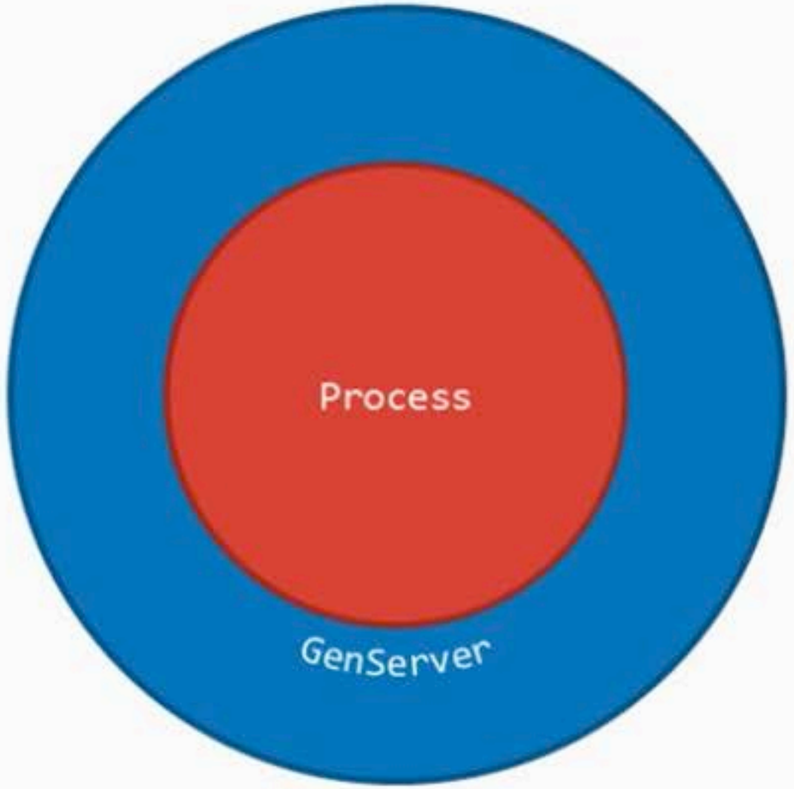
Simple wrap around state



Short-lived asynchronous computation

Recap – The GenServer

Simple wrap around state



Short-lived asynchronous computation

Agent

Simple wrapper around state.

Provides functions for retrieving and updating said state.

```
➜ Projects iex
Erlang/OTP 20 [erts-9.0.4] [source] [64-bit] [smp:8:8] [ds:8:8:10] [async-threads:10
] [hipe] [kernel-poll:false] [dtrace]

Interactive Elixir (1.5.1) - press Ctrl+C to exit (type h() ENTER for help)
iex(1)> {:ok, agent} = Agent.start_link(fn -> [] end)
{:ok, #PID<0.86.0>}
iex(2)> state = Agent.get(agent, fn state -> state end)
[]
iex(3)> Agent.update(agent, fn state -> ["hello" | state] end)
:ok
iex(4)> state = Agent.get(agent, fn state -> state end)
["hello"]
iex(5)> state = Agent.get(agent, fn state -> Enum.at(state, 0) |> String.upcase() en
d)
"HELLO"
iex(6)> state = Agent.get_and_update(agent, fn state -> {Enum.at(state, 0), []} end)

"hello"
iex(7)> state = Agent.get(agent, fn state -> state end)
[]
iex(8)>
```

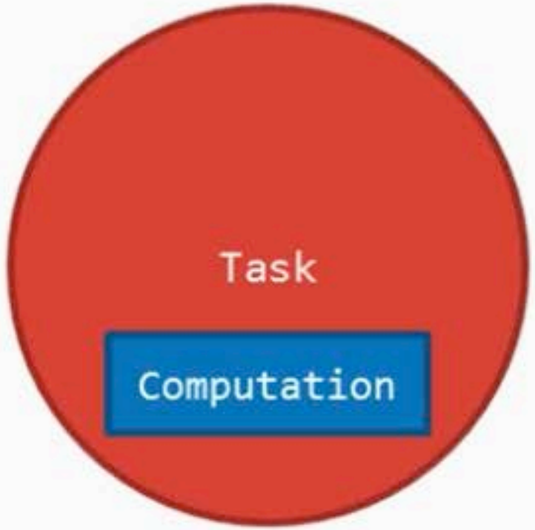
```
:ok
iex(4)> state = Agent.get(agent, fn state -> state end)
["hello"]
iex(5)> state = Agent.get(agent, fn state -> Enum.at(state, 0) |> String.upcase() en
d)
"HELLO"
iex(6)> state = Agent.get_and_update(agent, fn state -> {Enum.at(state, 0), []} end)

"hello"
iex(7)> state = Agent.get(agent, fn state -> state end)
[]
iex(8)> Agent.st
start/1          start/2          start/3          start/4
start_link/1     start_link/2     start_link/3     start_link/4
stop/1           stop/2           stop/3
iex(8)> Agent.stop(agent)
:ok
iex(9)> state = Agent.get(agent, fn state -> state end)
** (exit) exited in: GenServer.call(#PID<0.86.0>, {:get, #Function<6.99386804/1 in :
erl_eval.expr/5>}, 5000)
    ** (EXIT) no process: the process is not alive or there's no process currently a
ssociated with the given name, possibly because its application isn't started
    (elixir) lib/gen_server.ex:774: GenServer.call/3
iex(9)>
```

Task

Asynchronous units of computation.

Provides functions for launching and waiting on tasks.



```
➜ Projects iex
Erlang/OTP 20 [erts-9.0.4] [source] [64-bit] [smp:8:8] [ds:8:8:10] [async-threads:10
] [hipe] [kernel-poll:false] [dtrace]

Interactive Elixir (1.5.1) - press Ctrl+C to exit (type h() ENTER for help)
iex(1)> task = Task.async(fn -> "Hello" end)
%Task{owner: #PID<0.84.0>, pid: #PID<0.86.0>,
 ref: #Reference<0.1570111398.2060713991.48356>}
iex(2)> result = Task.await(task)
"Hello"
iex(3)> result = Task.await(task)
** (exit) exited in: Task.await(%Task{owner: #PID<0.84.0>, pid: #PID<0.86.0>, ref: #
Reference<0.1570111398.2060713991.48356>}, 5000)
    ** (EXIT) time out
    (elixir) lib/task.ex:491: Task.await/2
iex(3)> result = Task.await(task, 1000)
** (exit) exited in: Task.await(%Task{owner: #PID<0.84.0>, pid: #PID<0.86.0>, ref: #
Reference<0.1570111398.2060713991.48356>}, 1000)
    ** (EXIT) time out
    (elixir) lib/task.ex:491: Task.await/2
iex(3)>
```