HELLO!

I am Gabriele Santomaggio

@gsantomaggio

WHY?

The BEAM virtual machine is different

First 3 minutes

4

10 MINUTES LATER.. ...TRYING TO UNDERSTAND

Definition

Elixir leverages the Erlang VM, known for running low-latency, distributed and fault-tolerant systems, while also being successfully used in web development and the embedded software domain.

Elixir is a dynamic, functional language designed for building scalable and maintainable applications.
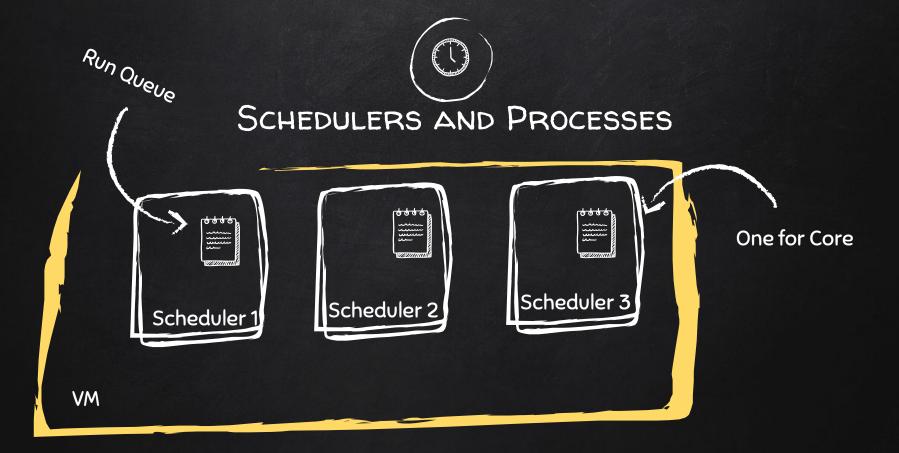
Main Concepts

✘ Nodes
✘ No threads
✘ Processes ⟵ OS STYLE
✘ No shared memory
✘ Message passing
✘ Built-in distribution
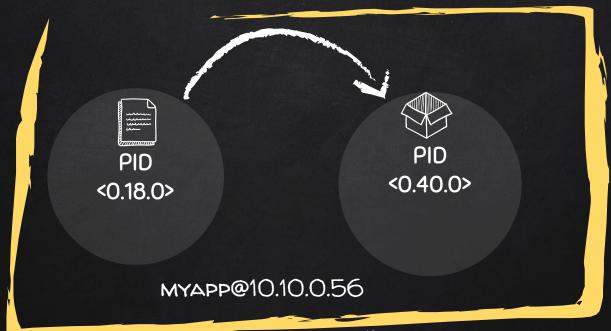✘ Supervisors

Preemptive — scheduler

- one process will run for a while, and then the scheduler of the virtual machine will suspend it and let another process run.

- In a multicore or a distributed system we can achieve true parallelism

Run Queue

Schedulers and Processes

One for Core

Scheduler 1

Scheduler 2

Scheduler 3

VM

Processes Messages — same beam

PID
<0.18.0>

PID
<0.40.0>

MYAPP@10.10.0.56
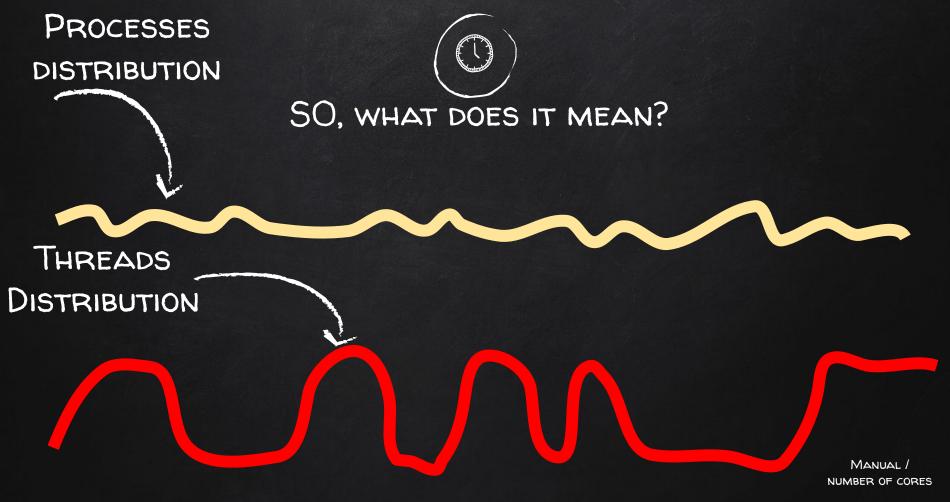
Processes Messages – across beams

PID
<0.18.0>

PID
<0.40.0>

MYAPP@10.10.0.56

MYAPP@10.10.0.24

Processes distribution

SO, WHAT DOES IT MEAN?

Threads Distribution

Manual / number of cores
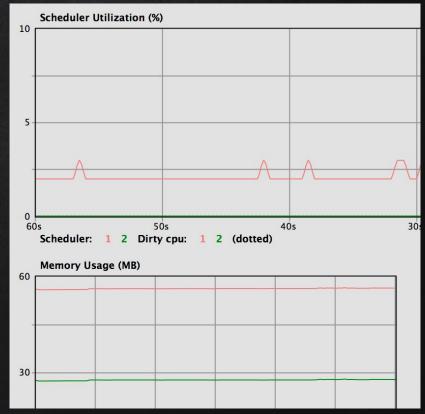
13

The BEAM as OS

✘ Tuning the VM (number of cores, processes ....)
✘ Etop/shell/inet commands
✘ Observer (sort of task manager)
✘ Remote shell using another node
✘ Remote shell using ssh
✘ Add remove modules
✘ Kill processes

8 → SCHEDULERS ← 2

Left panel:

Scheduler Utilization (%)

Scheduler: 1 2 3 4 5 6 7 8   Dirty cpu: 1 2 3 4 5 6 7 8

Memory Usage (MB)

Right panel:

Scheduler Utilization (%)

Scheduler: 1 2   Dirty cpu: 1 2   (dotted)

Memory Usage (MB)

Demo time ..

Access on the Elixir Application using remsh:

```
$ erl –sname debug  –remsh webapp@mac
    Interactive Elixir (1.7.3)
    iex(webapp@mac)1>
```

Built-in distribution..

Running BEAM App
(restapp@10.10.0.23)

Counter.status()

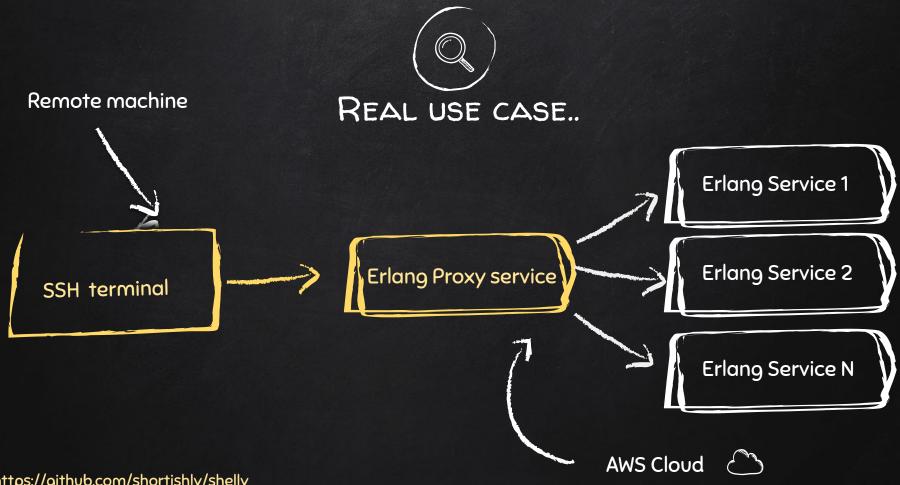*Counter.report()*

RPC call

app@10.10.0.89

... WHAT CAN I DO?

Remote Shell

SSH

Running BEAM App:
ex:rabbit@10.10.1.67

Profile/Trace

Add custom module
Ex : ssh Module

Kill processes

Demo time ..

Connect to RabbitMQ using remsh:

```
$ erl –sname debug  –remsh rabbit@mac
Erlang/OTP 21 [erts–10.0.8]
(rabbit@mac)1>inet.i().
```

REAL USE CASE..

Remote machine

SSH terminal

Erlang Proxy service

Erlang Service 1

Erlang Service 2

Erlang Service N

AWS Cloud

https://github.com/shortishly/shelly

20

THANKS!

Any questions?

You can find me at
@gsantomaggio