

# What's special about Elixir?

@sasajuric  
[aircloak.com](https://aircloak.com)



**Saša Jurić** @sasajuric · 23 Aug 2015

Coincidentally, I recently said that "the most important aspect of Elixir is in fact Erlang VM" [infoq.com/articles/elixi...](http://infoq.com/articles/elixi...)



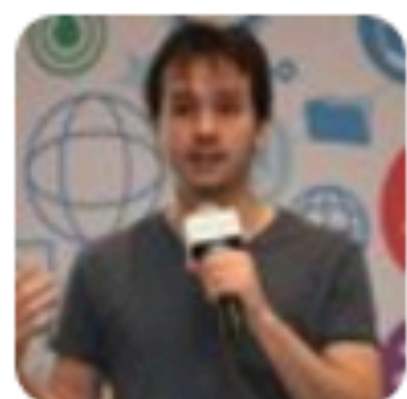
1



10



9



**José Valim**

@josevalim

Following

@sasajuric Precisely! If there was no Erlang VM, there would be no Elixir. Elixir is just a small drop. ;)

magic source

**BEAM**

magic beneficiaries

Erlang Elixir LFE

OTP GenStage

Cowboy Poolboy

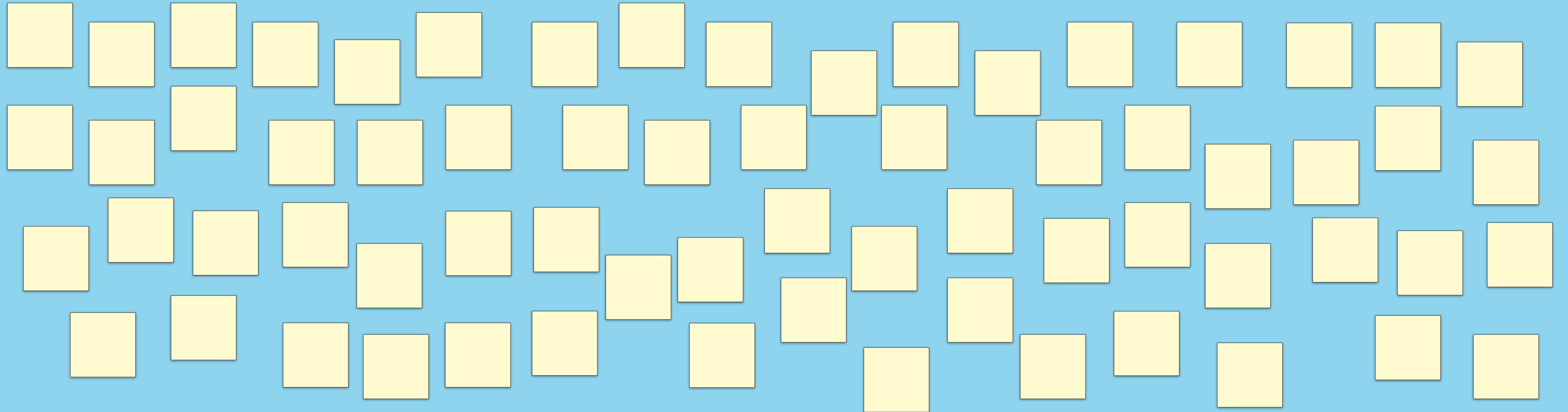
Plug Ecto

Phoenix

long-running system  
many tasks  
soft real-time

~~finite time program  
all-or-nothing  
hard real-time~~

# BEAM



scheduler

scheduler

scheduler

scheduler

CPU

CPU

CPU

CPU

demo

**scheduler**

p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>
----------------	----------------	----------------

**scheduler**

p <sub>4</sub>	p <sub>5</sub>	p <sub>6</sub>
----------------	----------------	----------------

**scheduler**

p <sub>7</sub>	p <sub>8</sub>	p <sub>9</sub>
----------------	----------------	----------------

**scheduler**

p <sub>10</sub>	p <sub>11</sub>	p <sub>12</sub>
-----------------	-----------------	-----------------

$p_1$

1 ms

$p_2$

1 ms

$p_3$

1 ms

$p_1$

1 ms

$p_2$

1 ms

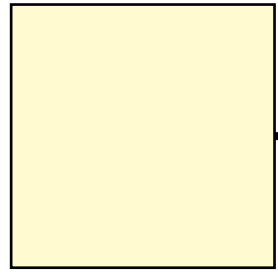
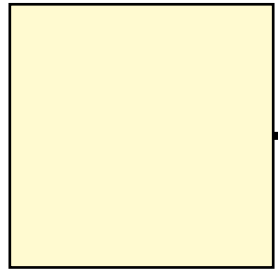
$p_3$

1 ms



demo

shared-nothing concurrency



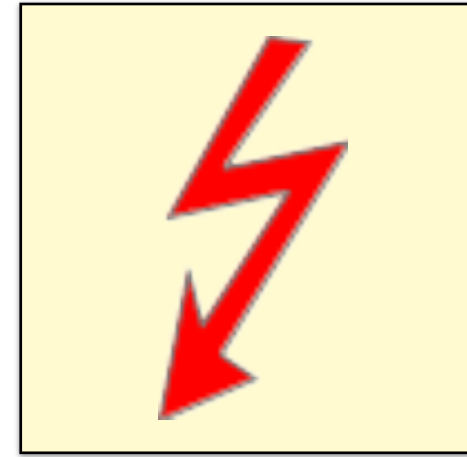


demo

**process**

heap

resources



process-owned “stuff” is released

**activity**

runtime entity  
with identity



demo

# **cancellation**

run the job in a separate process  
kill the process to cancel it

```
# run in a separate process
```

```
task = Task.async(fn -> ... end)
```

```
# wait for the result
```

```
case Task.yield(task, :timer.seconds(5)) do
```

```
  {:ok, result} ->
```

```
    # the result arrived on time
```

```
    do_something(result)
```

```
  nil ->
```

```
    # timeout -> kill the process
```

```
    Task.shutdown(task, :brutal_kill)
```

```
end
```

higher-level abstractions

backend

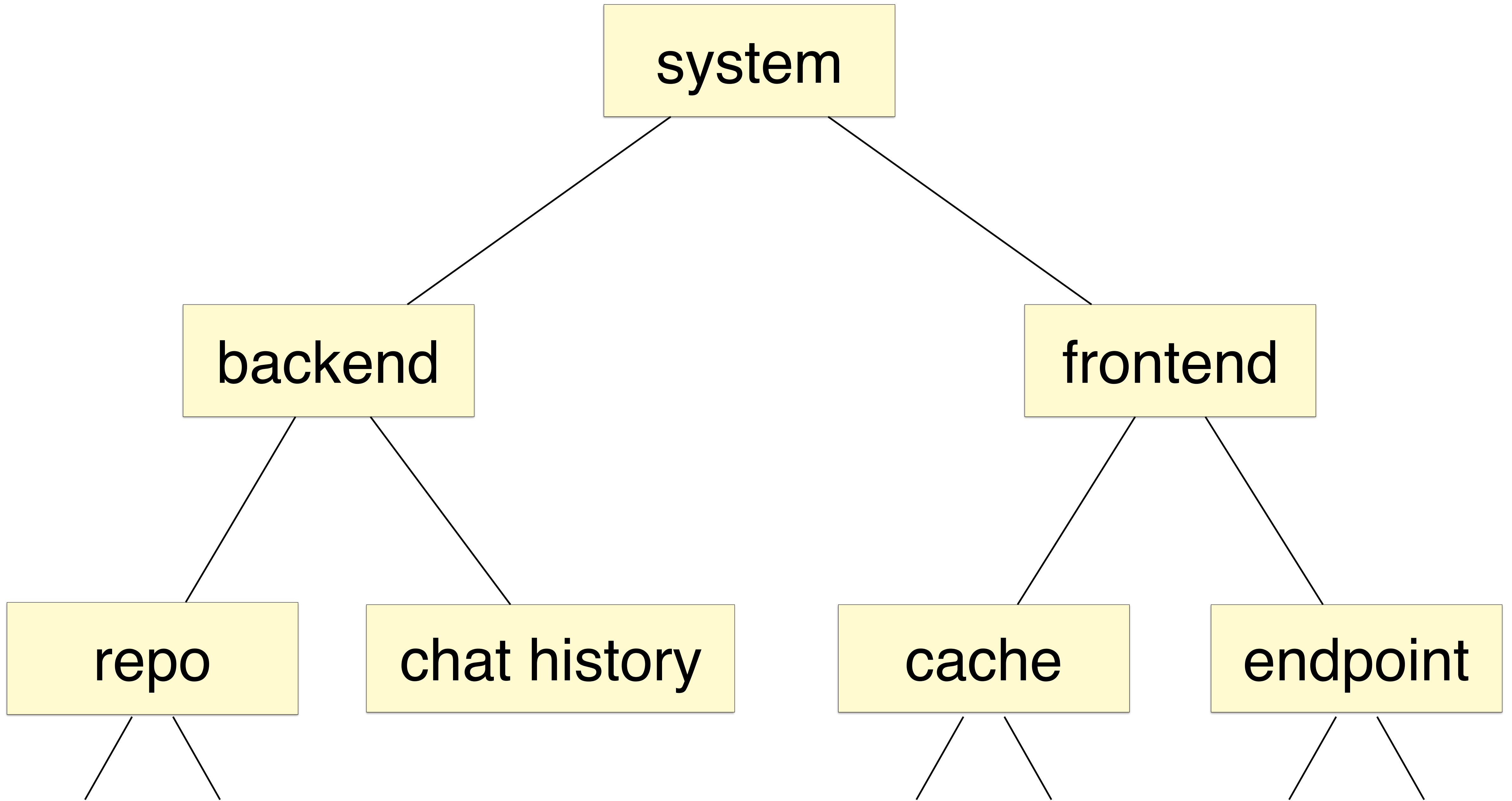
frontend

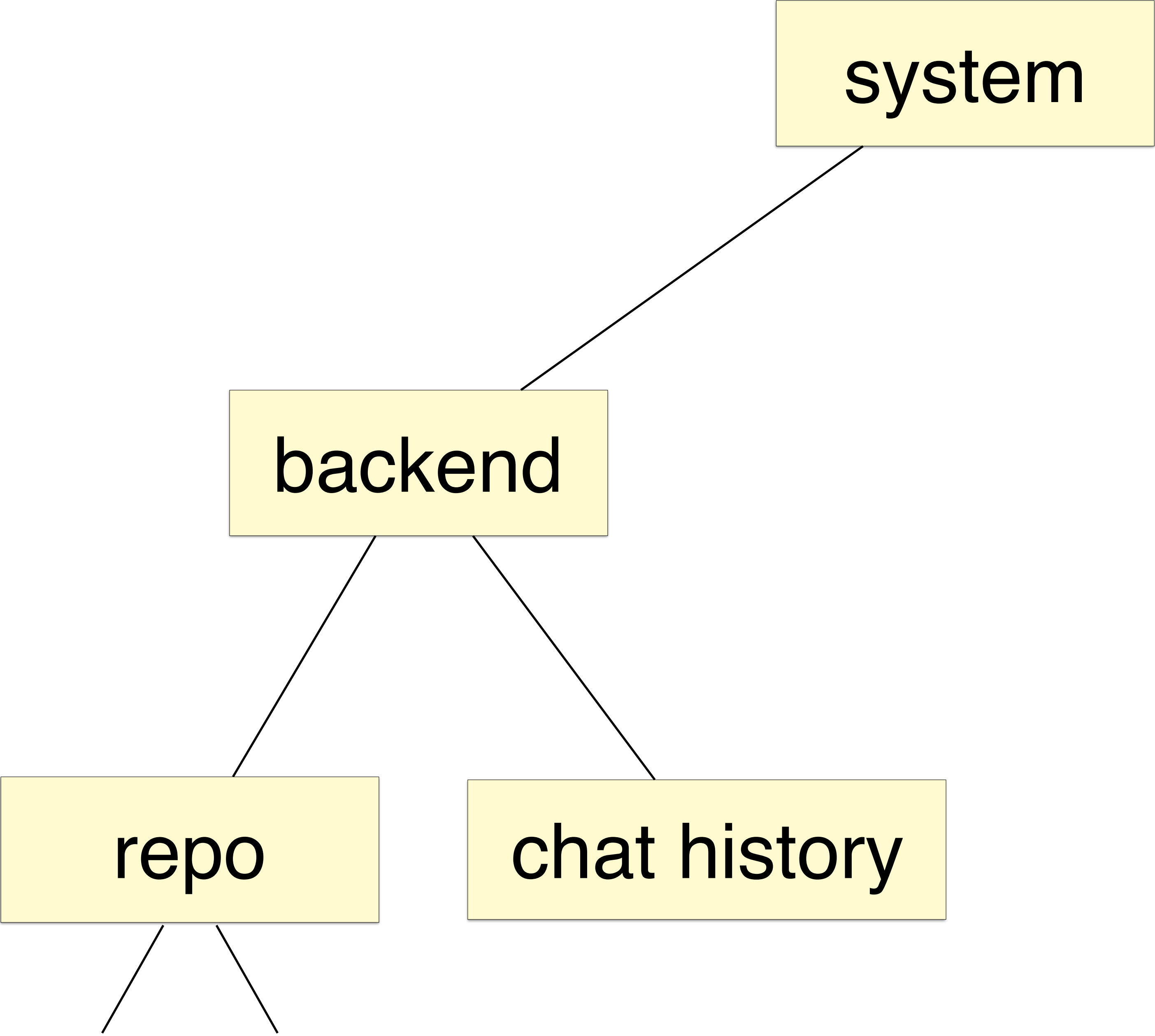
chat history

repo

endpoint

cache



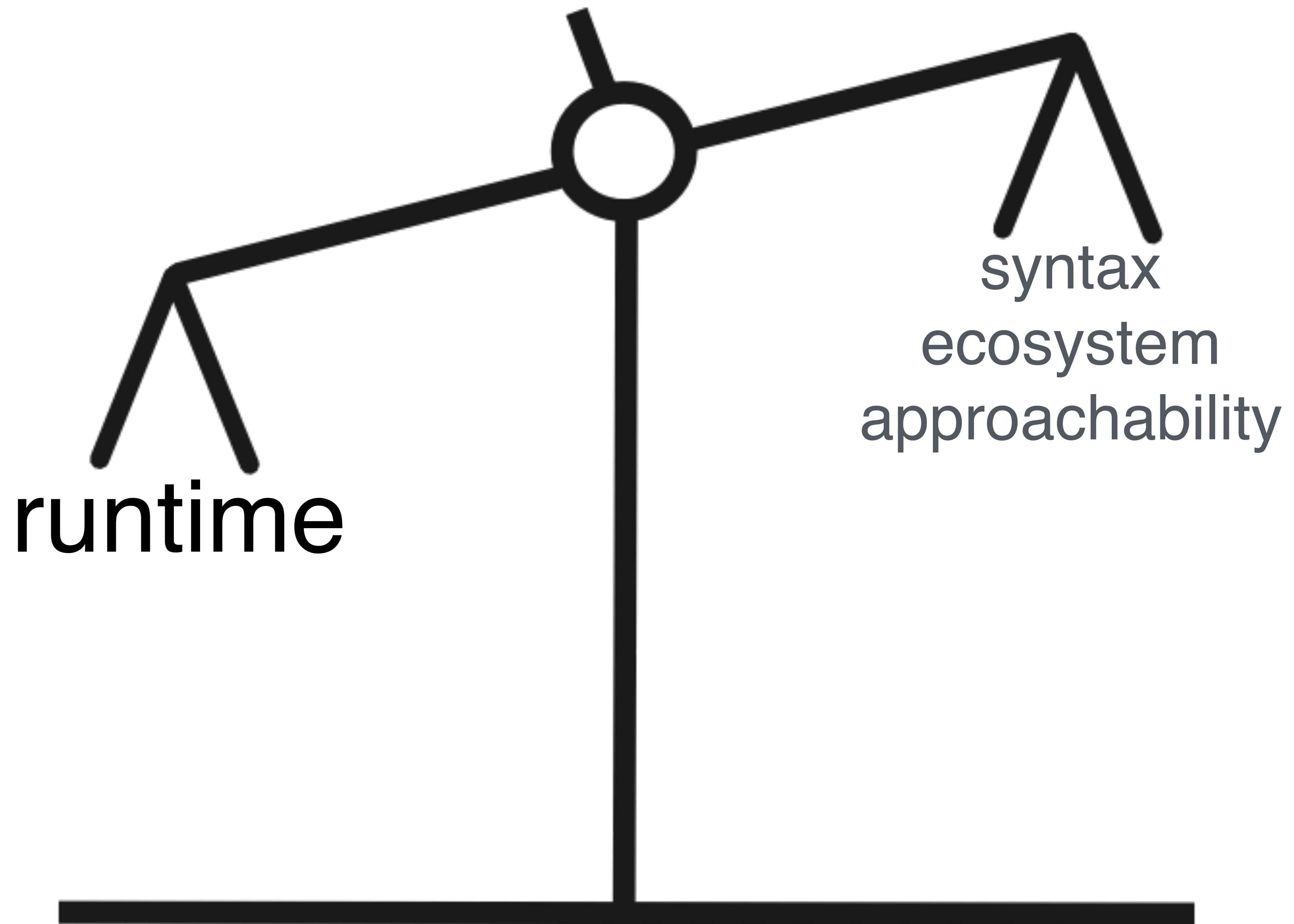


# **supervisor foundations**

lightweight concurrency  
shared-nothing concurrency  
frequent context switching  
activities as runtime citizens



syntax  
ecosystem  
approachability



<https://bit.ly/beam-resources>

Scales of Justice image taken from clipartfox.com