



THAT ONE,

FOR NOW

@bgmarx





*PHOENIX  
REQUEST*

A stylized landscape featuring purple wireframe mountains and a large red sun. The word "PROCESSES" is written in white cursive across the center.

*PROCESSES*

A stylized landscape featuring purple wireframe mountains and a large red sun. The scene is set against a dark background with a grid of white lines forming the terrain. The sun is a large, solid red circle in the upper center. The mountains are composed of various shades of purple and blue, with white lines indicating their structure. The overall aesthetic is futuristic and digital.

SCHEDULELERS



*SINGLE*  
*SCHEDULE*



MULTIPLE  
SCHEDULES  
(SMAP)



*PHOENIX*  
*REQUEST*





# Phoenix Framework

[Get Started](#)

## Welcome to Phoenix!

A productive web framework that does not compromise speed and maintainability.

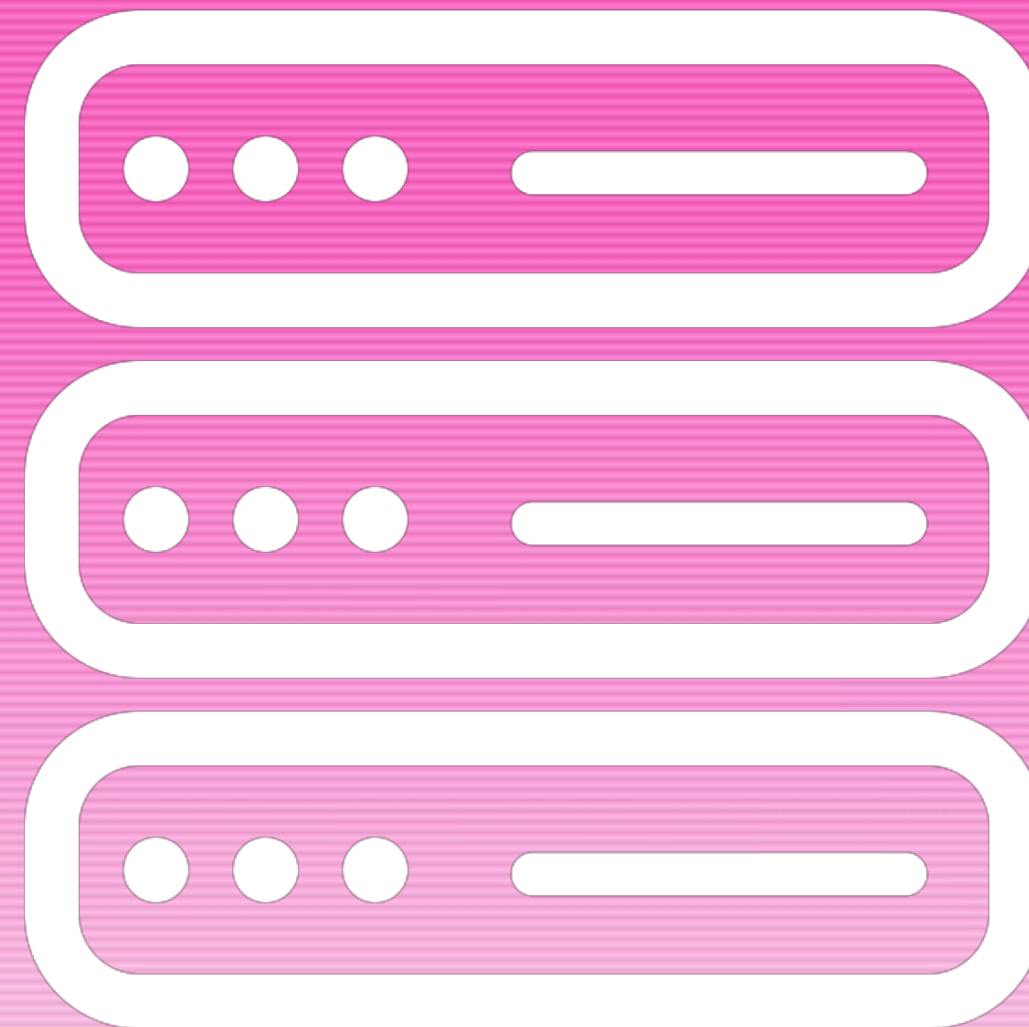
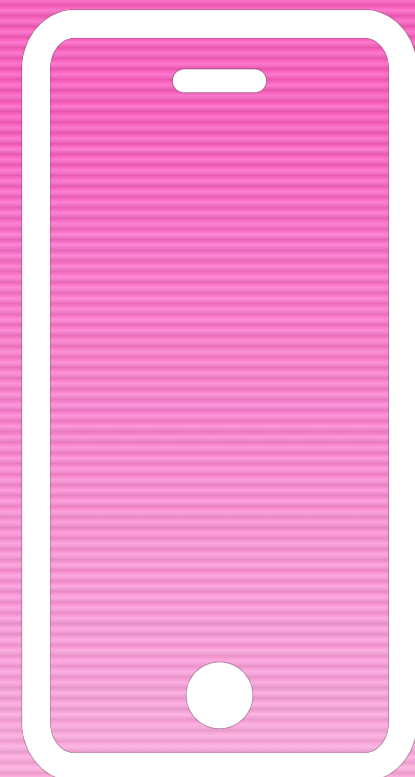
### Resources

- [Guides](#)
- [Docs](#)
- [Source](#)

### Help

- [Mailing list](#)
- [#elixir-lang on freenode IRC](#)
- [@elixirphoenix](#)

transformations



- endpoint
- plugs
- router
- pipeline
- controller
- view
- template



*PHOENIX*  
*REQUEST*

A stylized landscape featuring purple wireframe mountains and a large red sun. The word "PROCESSES" is written in white cursive across the center.

*PROCESSES*

# LIGHTWEIGHT

## System statistics / limit

Atoms:	22458 / 1048576 (2 % used)
Processes:	248 / 262144 (0 % used)
Ports:	8 / 65536 (0 % used)
ETS:	58 / 8192 (1 % used)
Distribution buffer busy limit:	1048576

# ISOLATED

In simple relativity theory the concept of simultaneity just does not exist. The point is that in reality objects do not share state, I believe its not a good idea to model what cannot exist in reality in software ...

Everything that can be achieved with sharing and locks can also be achieved with pure message passing and no locks. The [sic] is the Erlang way.

EXECUTES

CODE

# FOUR BLOCKS OF MEMORY

MAILBOX

HEAP

PCB

STACK



# MAILBOX

Pid	Name or Initial Func	Reds	Memory	MsgQ	Current Function
<0.256.0>	Elixir.Plug.Supervisor	1	7104	0	gen_server:loop/7
<0.241.0>	ranch_sup	1	7104	0	gen_server:loop/7
<0.221.0>	hackney_sup	1	7104	0	gen_server:loop/7
<0.203.0>	Elixir.Logger.Supervisor	1	7104	0	gen_server:loop/7
<0.380.0>	erlang:apply/2	2581	8756	0	timer:sleep/1
<0.145.0>	Elixir.Hex.State	1	8844	0	gen_server:loop/7
<0.202.0>	application_master:start_it/4	1	8888	0	application_master:loop_it/4
<0.394.0>	observer_port_wx:init/1	2	8904	0	wx_object:loop/6
<0.247.0>	cowboy_clock	265	8932	0	gen_server:loop/7

PROCESSES

CONTROL BOARD

```
    Uint32 rcount; /* suspend count */
    int schedule_count; /* Times left to reschedule a low prio
process */
    Uint reds; /* No of reductions for this process */
    Eterm group_leader; /* Pid in charge (can be boxed) */
    Uint flags; /* Trap exit, etc (no trace flags anymore) */
    Eterm fvalue; /* Exit & Throw value (failure reason) */
    Uint freason; /* Reason for detected failure */
    Eterm ftrace; /* Latest exception stack trace dump */
```

[HTTPS://GITHUB.COM/ERLANG/OTP/BLOB/OTP-21.0/ERTS/EMULATOR/BEAM/  
ERL\\_PROCESS.H#L927](https://github.com/erlang/otp/blob/OTP-21.0/erts/emulator/beam/erl_process.h#L927)

# PROCESS STATES

FREE

WAITING

RUNNABLE

RUNNING

EXITING

GARBING

SUSPENDED

garbing computer science



**All**

Images

Videos

News

Shopping

More

Settings

Tools

About 120,000 results (0.45 seconds)

Did you mean: **garbage** computer science

# "NORMAL" STATES

WAITING

RUNNABLE

RUNNING

```
iex(5)> :erlang.process_info self(), :status  
{:status, :running}
```

A stylized landscape featuring purple wireframe mountains and a large red sun. The word "PROCESSES" is written in white cursive across the center.

*PROCESSES*



The background features a stylized, low-poly landscape. The mountains are rendered in shades of purple and blue, with a grid-like pattern of white lines. A large, bright red sun is positioned in the upper center, set against a dark, starry sky. The overall aesthetic is modern and digital.

SCHEDULELERS



*SINGLE*  
*SCHEDULE*

```
iex(1)> :erlang.system_info:(smp_support)
```

```
true
```

```
iex(2)> :erlang.system_info :schedulers_online
```

```
4
```

```
~> iex --erl "+S 1"
```

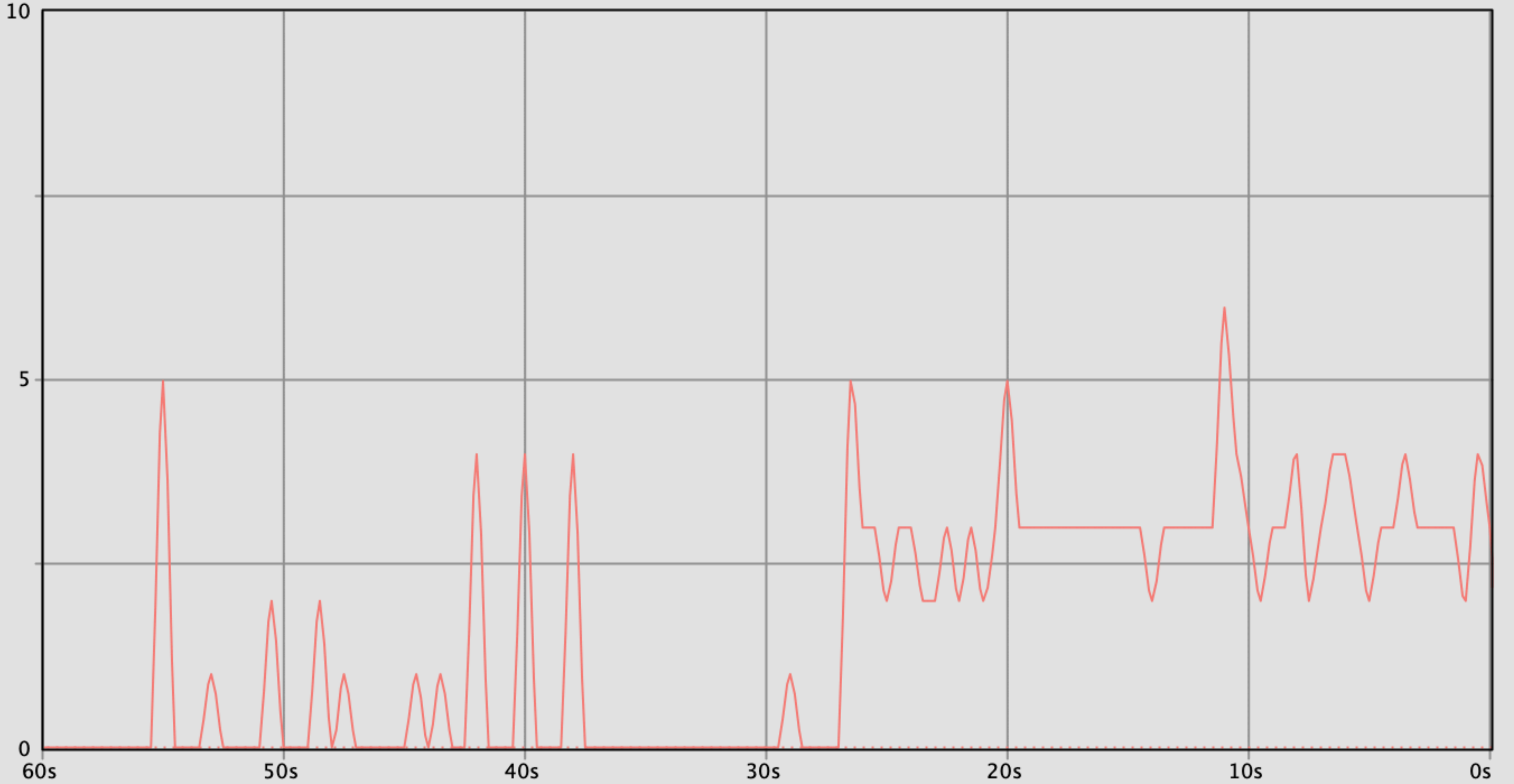
```
Erlang/OTP 21 [erts-10.0] [source] [64-bit] [smp:1:1] [ds:1:1:10] [async-threads:1] [hipe]
```

```
Interactive Elixir (1.7.3) - press Ctrl+C to exit (type h() ENTER for help)
```

```
iex(1)> :erlang.system_info :schedulers_online
```

```
1
```

# Scheduler Utilization (%)



Scheduler: 1 Dirty cpu: 1 (dotted)

```
iex(25)> Process.sleep(10_000)
[info] GET /
[debug] Processing with SimpleWeb.PageController.index/2
Parameters: %{}
Pipelines: [:browser]
[info] Sent 200 in 150µs
```



PREEMPTIVE\*



# REDUCTIONS





4000

REDUCTIONS

```
#DEFINE CONTEXT_REDS 4000 /* SWAP PROCESS OUT AFTER THIS NUMBER */
```



[HTTPS://GITHUB.COM/ERLANG/OTP/BLOB/MASTER/ERTS/EMULATOR/BEAM/ERL\\_VM.H#L39](https://github.com/erlang/otp/blob/master/erts/emulator/beam/erl_vm.h#L39)

# RUN QUEUE

WAITING



RUNNABLE



RUNNING



SCHEDULER



RUN QUEUE

TASK 1

TASK 2

TASK N



```
iex(11)> :erlang.statistics(:run_queue)  
0
```

SCHEDULER



RUN QUEUE

TASK 1

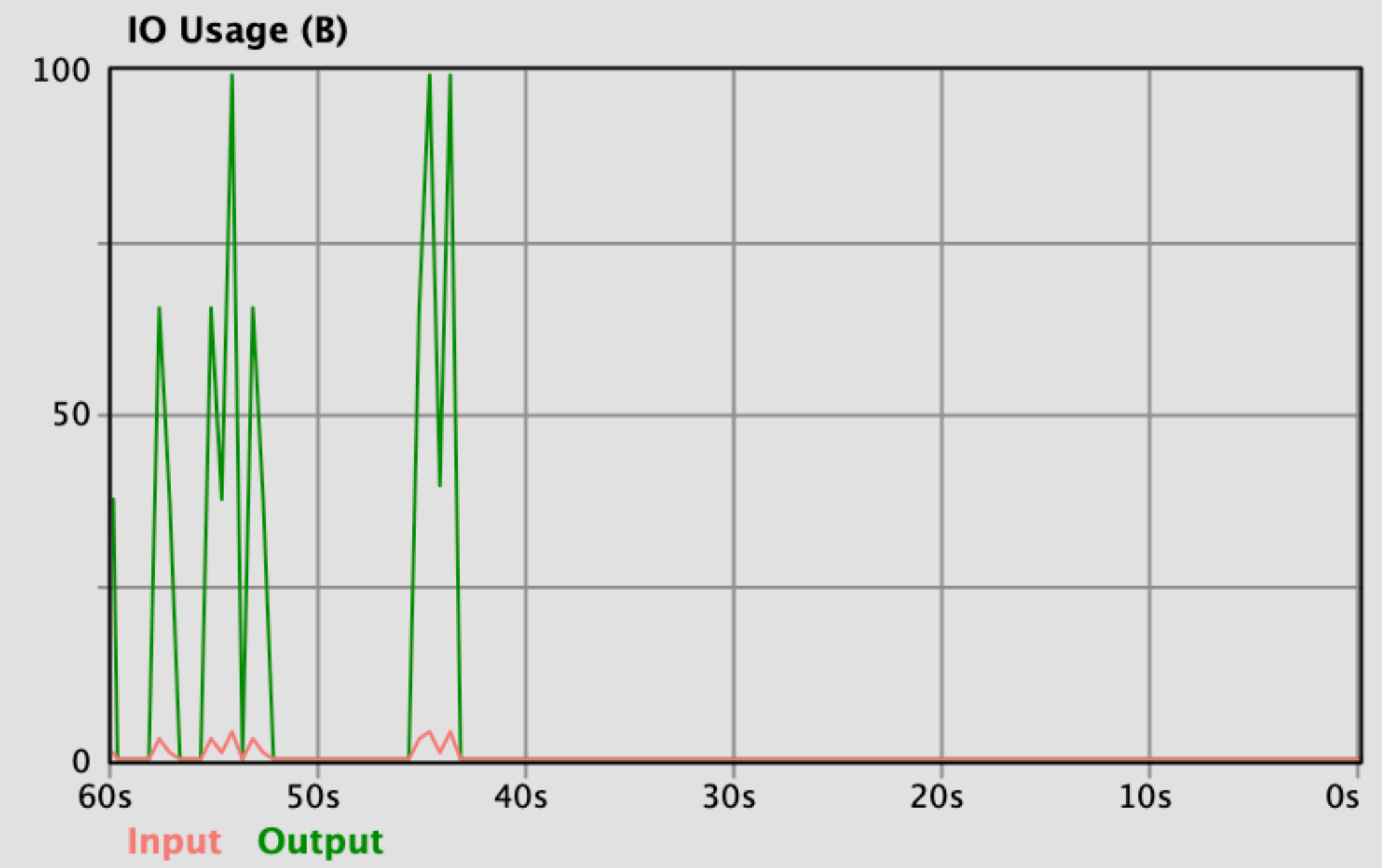
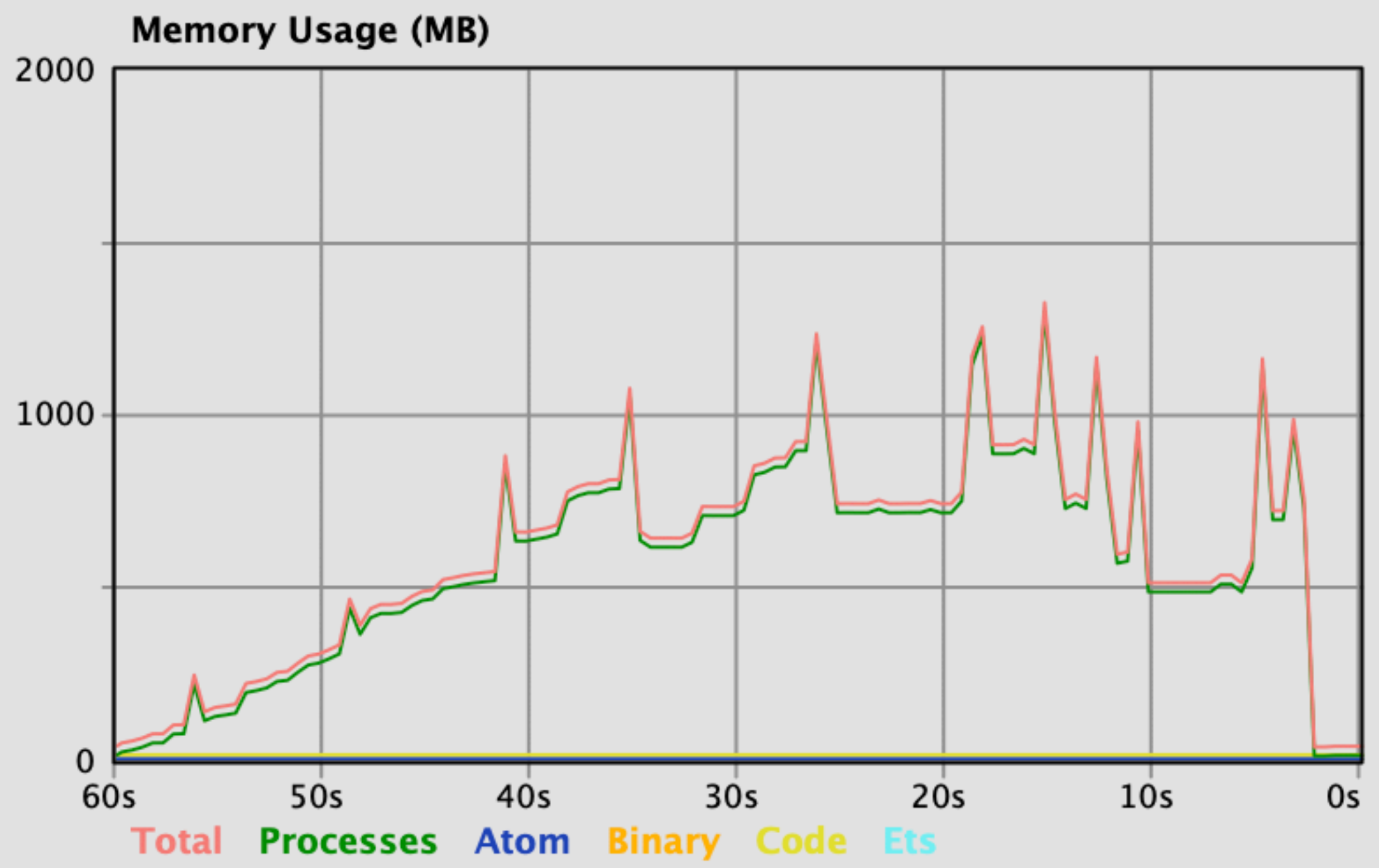
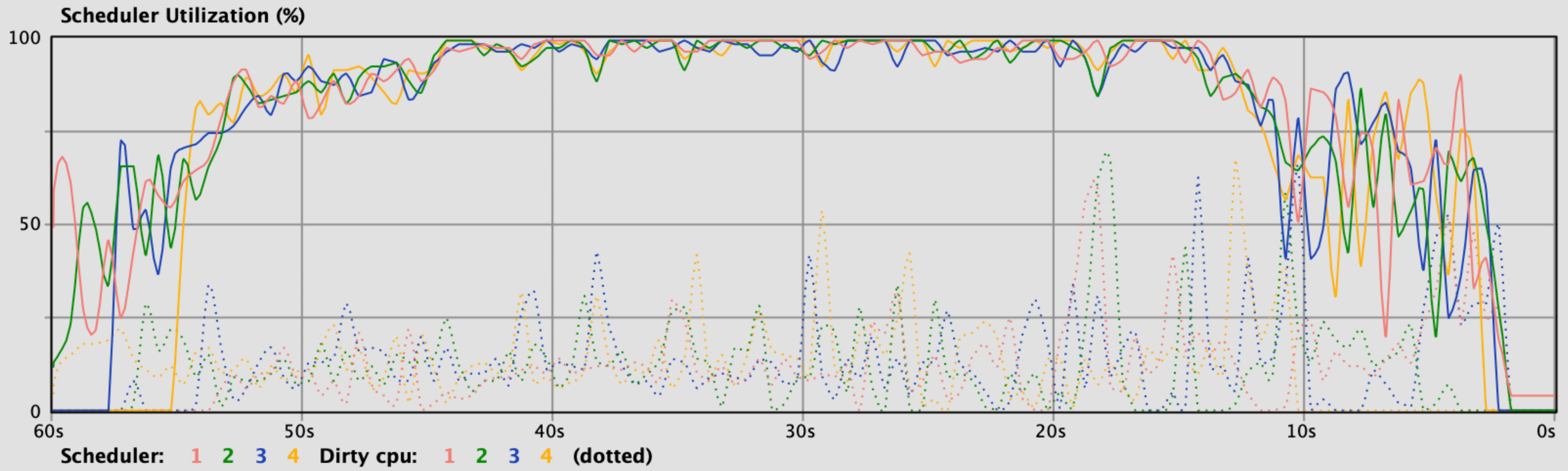
TASK 2

TASK N



The background features a large, solid red circle in the upper center. Below and around it are several purple, faceted, geometric shapes that resemble crystalline structures or low-poly mountains. The overall aesthetic is modern and digital.

MULTIPLE  
SCHEDULES  
(SMAP)



# RUN QUEUE

SCHEDULER



MAX PRIORITY

HIGH PRIORITY

NORMAL & LOW



LOAD

BALANCING



TASK

STEALING



# MIGRATION LOGIC



SCHEDULER 1



SCHEDULER 2



# MIGRATION LOGIC

SCHEDULER 1

RUN QUEUE 1



SCHEDULER 2

RUN QUEUE 2



SCHEDULER N

RUN QUEUE N



*FITNEFLY*

*TUNED*



# SCHEDULER FLAGS

```
elixir --erl "+sbt ns" #etc
```

[HTTP://ERLANG.ORG/DOC/MAN/ERL.HTML#FLAGS](http://erlang.org/doc/man/erl.html#flags)

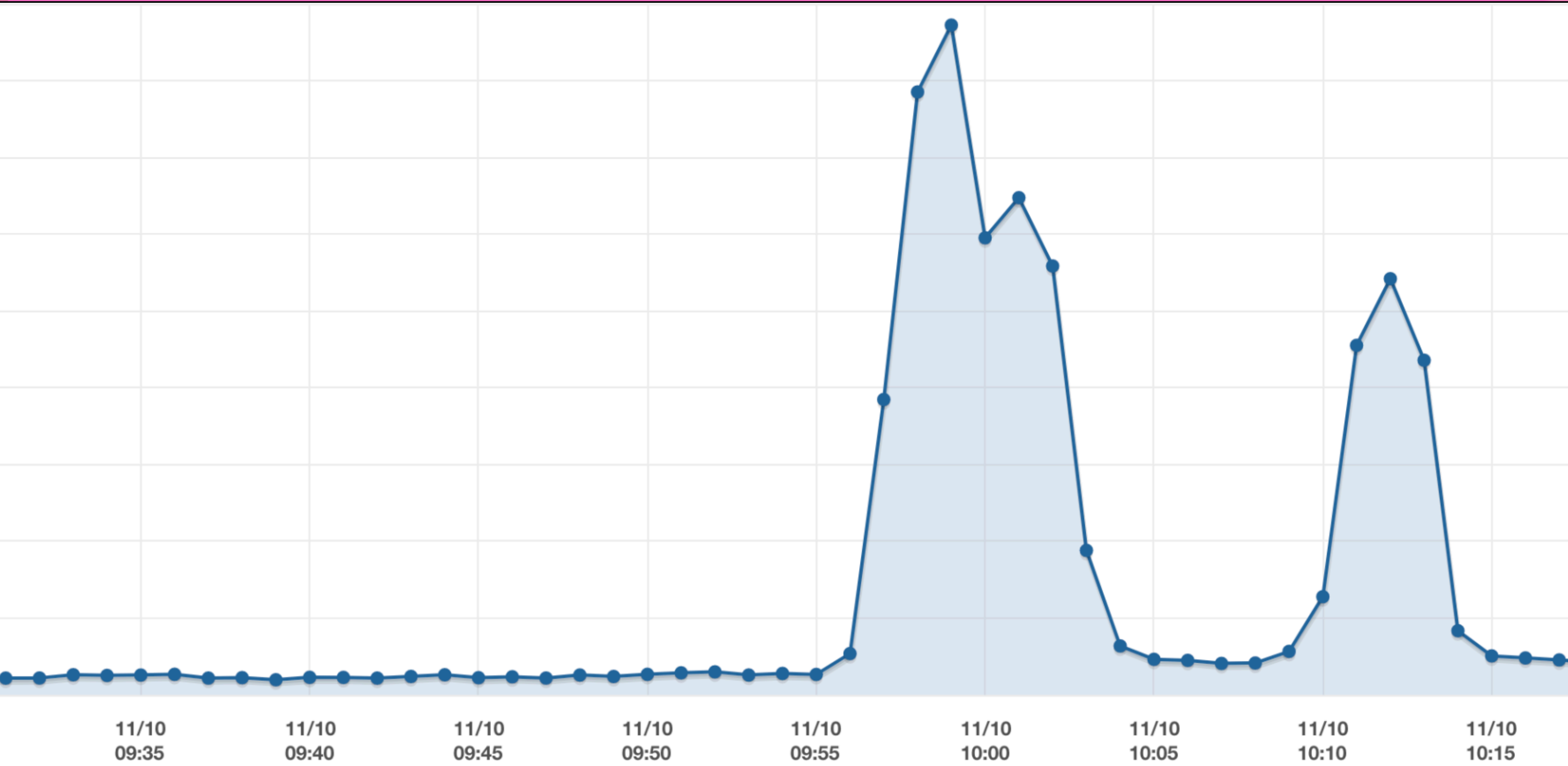
# MOCHIT SCHEDDILLERS



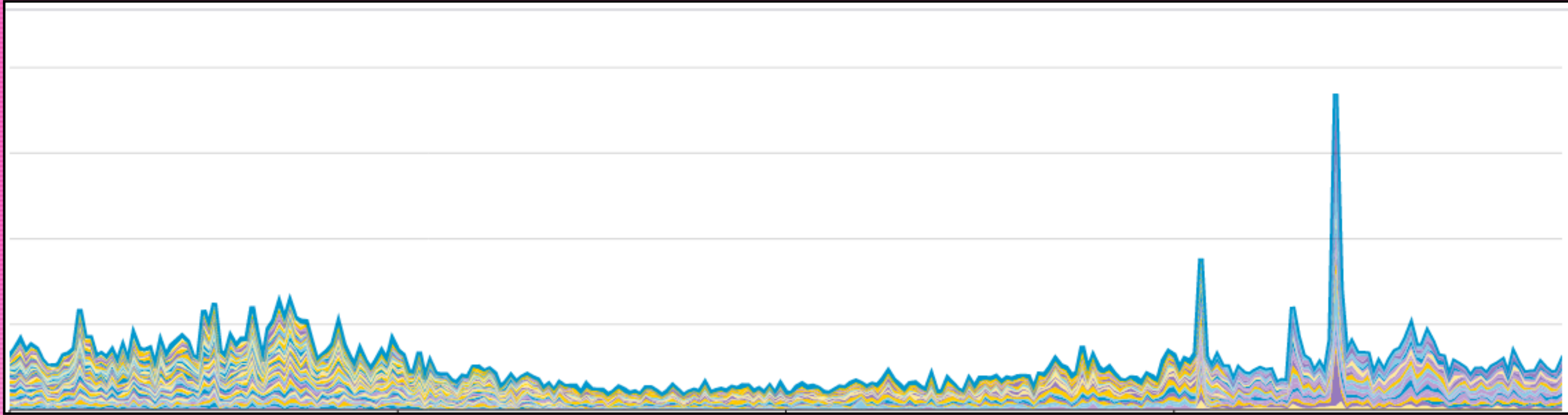
HOW HAS

THIS FELLED!

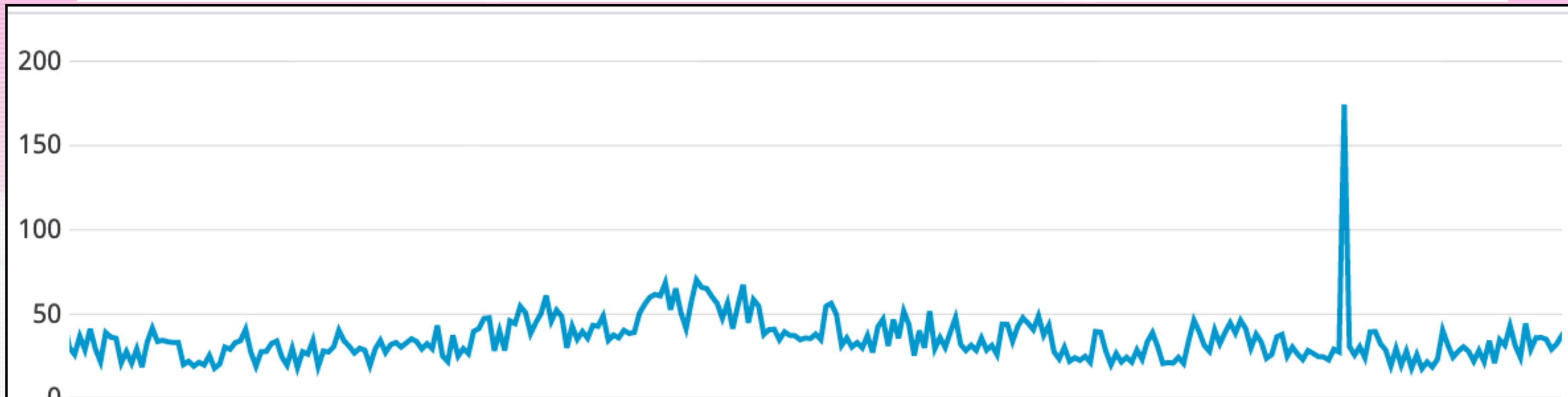




# RUN QUEUE



# LATENCY



REQUEST

SCHEDULER ASSIGNED

ADDED TO RUN QUEUE

TRANSFORMATIONS

RESPONSE

# IMAGE CREDITS

<https://wallpaper-gallery.net/single/vaporwave-background-hd-14.html>

<http://www.desktopimages.org/preview/770453/1920/1080/>

[https://www.wallpapervortex.com/  
wallpaper-52083\\_1\\_miscellaneous\\_digital\\_art\\_vaporwave\\_image\\_glitch.html](https://www.wallpapervortex.com/wallpaper-52083_1_miscellaneous_digital_art_vaporwave_image_glitch.html)

<https://www.deviantart.com/epix0r/art/R-o-a-d-W-a-y-1920-x-1080-HD-629299503>

<https://wallpaper-gallery.net/single/glitch-wallpaper-hd-11.html>

# RESOURCES

The BEAM Book - <https://github.com/happi/theBeamBook>

<https://jlouisramblings.blogspot.com/2013/01/how-erlang-does-scheduling.html>

<http://www.erlang-factory.com/upload/presentations/105/KennethLundin-ErlangFactory2009London-AboutErlangOTPandMulti-coreperformanceinparticular.pdf>

<https://hamidreza-s.github.io/erlang/scheduling/real-time/preemptive/migration/2016/02/09/erlang-scheduler-details.html>

DZIEKUJĘ

*@bggmarx*