

SCALING ERLANG WEB APPLICATIONS

100 TO 100K USERS AT ONE WEB SERVER

Fernando Benavides (*@elbrujohalcon*)

Inaka Labs

February 16, 2012

WHO AM I?

TODO: some funny stuff about Argentina, me, Erlang, elbrujohalcon... maybe some pictures

└ Who am I?

TODO: some funny stuff about Argentina, me, Erlang,
elbrujahalcon... maybe some pictures

Brief review of my story, why am I an Erlang programmer, how much I know about web applications and scalability

ABOUT INAKA

TODO: Inaka's Info

└ About Inaka

TODO: Inaka's Info

Brief review of Inaka's story, the systems we develop and why scalability matters to us

OUTLINE

1 THE CHALLENGE

- Description
- Scope

2 THE PLAN

- Finding The Initial Boundaries
- Blackbox Tests
- Erlang Tuning
- Adding Nodes

3 TIPS AND TRICKS

- TCP Tuning
- OTP
- Other Stuff

4 FINAL WORDS

- Summary
- Other stuff

OUTLINE

1 THE CHALLENGE

- Description
- Scope

2 THE PLAN

- Finding The Initial Boundaries
- Blackbox Tests
- Erlang Tuning
- Adding Nodes

3 TIPS AND TRICKS

- TCP Tuning
- OTP
- Other Stuff

4 FINAL WORDS

- Summary
- Other stuff

OUTLINE

1 THE CHALLENGE

- Description
- Scope

2 THE PLAN

- Finding The Initial Boundaries
- Blackbox Tests
- Erlang Tuning
- Adding Nodes

3 TIPS AND TRICKS

- TCP Tunning
- OTP
- Other Stuff

4 FINAL WORDS

- Summary
- Other stuff

OUTLINE

1 THE CHALLENGE

- Description
- Scope

2 THE PLAN

- Finding The Initial Boundaries
- Blackbox Tests
- Erlang Tuning
- Adding Nodes

3 TIPS AND TRICKS

- TCP Tuning
- OTP
- Other Stuff

4 FINAL WORDS

- Summary
- Other stuff

We will work on the scalability of a *web* project that has a *HTTP API* and keeps clients *connected* to the server for *long periods* of time.

Examples:

- Social sites
- Chat sites
- Sports sites

We will work on the scalability of a *web* project that has a *HTTP API* and keeps clients *connected* to the server for *long periods* of time.

Examples:

- Social sites
- Chat sites
- Sports sites

We will work on the scalability of a *web* project that has a *HTTP API* and keeps clients *connected* to the server *for long periods* of time.

Examples:

- Social sites
- Chat sites
- Sports sites

We will work on the scalability of a *web* project that has a *HTTP API* and keeps clients *connected* to the server for *long periods* of time.

Examples:

- Social sites
- Chat sites
- Sports sites

We will work on the scalability of a *web* project that has a *HTTP API* and keeps clients *connected* to the server for *long periods* of time.

Examples:

- Social sites
- Chat sites
- Sports sites

We will work on

- OTP behaviours
- TCP connections
- mochiweb
- Underlying system configurations

*We will **not** deal with*

- Multiple machines/nodes
- Databases

We will work on

- OTP behaviours
- TCP connections
- mochiweb
- Underlying system configurations

*We will **not** deal with*

- Multiple machines/nodes
- Databases

GENERAL CONSIDERATIONS

- Be sure it's working
- Automate your clients
- Keep a human watching
- Be patient

GENERAL CONSIDERATIONS

- Be sure it's working
- **Automate your clients**
- Keep a human watching
- Be patient

GENERAL CONSIDERATIONS

- Be sure it's working
- Automate your clients
- **Keep a human watching**
- Be patient

GENERAL CONSIDERATIONS

- Be sure it's working
- Automate your clients
- Keep a human watching
- Be patient

GENERAL CONSIDERATIONS

- Be sure it's working
- Automate your clients
- Keep a human watching
- Be patient

GOALS

TODO: this stage goals

STEPS

TODO: this stage steps

GOALS

TODO: this stage goals

STEPS

TODO: this stage steps

GOALS

TODO: this stage goals

STEPS

TODO: this stage steps

GOALS

TODO: this stage goals

STEPS

TODO: this stage steps

OS TWEAKS

TODO: Copy from the article

ERLANG TWEAKS

TODO: Copy from the article on listeners
TODO: Copy from the article on inbound TCP connections
TODO: Copy from the article on outbound TCP connections

GEN_EVENT

TODO: Copy from the article on sup_handler
TODO: Copy from the article on long delivery queues

GEN_SERVERS

TODO: Copy from the article on timing out
TODO: Copy from the article on too much memory
TODO: Copy from the article on taking too long to initialize

SUPERVISORS

TODO: Copy from the article

PROCESS REGISTRATION

TODO: Copy from the article

TIMERS

TODO: Copy from the article

LOGGING

TODO: Copy from the article

SUMMARY

TODO: Summary

OTHER STUFF

THAT WE LEFT OUT OF THIS PRESENTATION

TODO: List of other scalability stuff we left out

Any questions?


```
-spec fact(integer()) -> integer().  
fact(N) ->  
    lists:fold(fun(X, F) ->  
                F * X  
            end, 1, lists:seq(1,N)).
```