

U5ZAJ15P0 : thus my interest

U5ZAJ15P0 : I didn't know 4clojure; thanks!

U5ZAJ15P0 : Out of curiosity, what has your experience been going from Erlang to Clojure <@U067BPAB1> ?

U5ZAJ15P0 : It's one of those languages (Erlang) that is on my "to learn" list but never quite got around to doing it

U5ZAJ15P0 : And did you have a good time with Erlang?

U067BPAB1 : Clojure is a much better language. Erlang is more interesting from a concurrency perspective than an FP one, since the FP there is completely incidental (it was deemed necessary to implement the platform) rather than intentional. Both teach you something new, so if you feel like learning something else after Clojure I can recommend it. I enjoyed my time with Erlang but after 4+ years of it I think I've exhausted the platform, and I personally don't think it's the right fit in 90% of modern distributed/concurrent system development as a lot of the big selling points (failure handling, concurrency etc.) have been implemented outside of the VM these days and I just think that's the right way to do it instead of having to think about these explicitly within a service.

U06QSF3BK : hmm, seems that eastwood doesn't like defrecord forms produced by [org.clojure/clojure "1.9.0-alpha17"]

U060FKQPN : what's the issue?

U5WDMBTUJ : Hello! Can anyone help me understand the use-case of defrecord? Everytime I write a code, it is usually enough with just a function, but I'm afraid I'm missing the good parts of it!

U2PGHFU5U : Joy of Clojure is fun, and from a practical point of view I loved Learning ClojureScript, but that's about ClojureScript

U06QSF3BK : ```== Eastwood 0.2.4 Clojure 1.9.0-alpha17 JVM 1.8.0_131Directories scanned for source files:

src test

== Linting lol.core ==

Entering directory `/private/tmp/lol'

src/lol/core.clj:3:1: suspicious-expression: condp called with 3 args. (condp pred test-expr expr) always returns expr.

Perhaps there are misplaced parentheses?``` where source is ```0% cat src/lol/core.clj

(ns lol.core)

(defrecord Lol [])```

U06CM8C3V : <@U5ZAJ15P0> I'm planning to learn Erlang too. Well, ok, I'll probably learn Elixir instead, but that's derived from Erlang.

U2PGHFU5U : When running around 1000 simultaneous clj-http/get requests with `:async? true` I get this error:

```Exception in thread "async-dispatch-6" Exception in thread "async-dispatch-5" java.lang.OutOfMemoryError: unable to create new native thread```

Is there a way to circumvent this?

U5ZAJ15P0 : <@U06CM8C3V> yes Elixir looks interesting!

U3JURM9B6 : <@U04V70XH6> : is it public if your team uses GAE or AWS ?

U06QSF3BK : hrm, having hard time trying to teach eastwood to ignore defrecord

U06QSF3BK : ```(disable-warning {:linter :suspicious-expression

:for-macro 'clojure.core/defrecord

:reason "Clojure 1.9 defrecord output causes false warning"})``` doesn't seem to work

U0GN0S72R : what's the proper spelling for a new version of `foo-bar`? `foo-bar2` or `foo-bar-2`?

U06QSF3BK : hmm, not actually clojure 1.9 problem

<<https://github.com/jonase/eastwood/blob/eb58d6fb2628e1d178d4b8d2c46d4788b2a20094/cases/testcases/suspicious.clj#L8-L11>>

U06QSF3BK : so the actual use for a defrecord without fields was a `com.stuartsierra.component`

U37NPE2H0 : gfredericks: seems like it would be the former in most cases

U37NPE2H0 : `foo-bar-2`, I think, more strongly implies it was intended to be a distinct symbol with a similar naming scheme, while `foo-bar2` is more clearly a mutation of the original `foo-bar`