

U06B8J0AJ : Now, spec seems like the right tool for conforming. However, I've also heard tales of spec being able to destructure as well.

U06B8J0AJ : Where can I read about how the destructuring works, specifically? If I can avoid hand-disassembling the stain upon humanity that is the output of `parse`, I'd be delighted.

U050487DQ : <@U0BKWMG5B> invite sent

U06B8J0AJ : I just discovered zippers

U5MMY8M7X : Anyone used clojure transducers for unfolds (anamorphisms)? All talk I've seen is on folds (reducers / catamorphisms).

U1C03090C : So I'm working on getting a plugin system working in clojure. Based on the loading system on <@U050CBXUZ> (<<https://yogthos.net/posts/2015-01-15-A-Plugin-System-in-Clojure.html>>). I can get the file on the classpath, but when I try and run the function defined by the .edn file packaged with the plugin, it gives me the error: `Could not locate 'oss\_world\_example/core\_\_init.class' or 'oss\_world\_example/core.clj' on classpath.`

U1C03090C : I checked in the jar and `oss\_world\_example/core\_\_init.class` is definitely there (you can download the example jar I'm using here: <<https://filebin.ca/3VLfGd71KesW/project.jar>>). What gives? How is it finding the .edn file but not the classes?

U050MP39D : <@U1C03090C> what does the code in those files look like?

U1C03090C : <@U050MP39D> `oss-world-example.core` looks like this:

U1B0DFD25 : <@U1C03090C> You might have an extra quote character somewhere?

U050MP39D : fwiw, I can load that ns from that jar fine```

```
11326-storage:look bfabry$ java -cp ~/.m2/repository/org/clojure/clojure/1.8.0/clojure-1.8.0.jar:project.jar clojure.main
Clojure 1.8.0
user=> (<http://clojure.java.io/resource[clojure.java.io/resource> "oss_world_example/core__init.class")
#object[java.net.URL 0x2ca26d77 "jar:file:/private/tmp/look/project.jar!/oss_world_example/core__init.class"]
user=> (require '[oss-world-example.core])
nil
user=> oss-world-example.core/entry
#object[oss_world_example.core$entry 0x24c22fe "oss_world_example.core$entry@24c22fe"]
user=>
```
```

U050MP39D : ```user=> (oss-world-example.core/entry)

Entered the example plugin.

nil

```

U1B0DFD25 : Perhaps in the edn file.

U050MP39D : so my bet is you don't actually have the jar on the cp

U050MP39D : the edn file is using underscores for the namespace, which is incorrect

U050MP39D : I'd have to see your loader code to know whether you compensate for that

U050MP39D : but, while that's a problem, it's not a problem that should yield the error you got. the error you got very specifically says the file does not exist in the cp

U050MP39D : (it could be finding the edn file because it's accessible by some other cp entry like .)

U1B0DFD25 : I think the names in `could not locate` error message should not start with a quote, so check for single quote in the edn file such as:```

```
{:description "Markdown parser"
 :init 'cryogen-markdown.core/init}
```
```

U050MP39D : ^ that's it

U050MP39D : I missed the quote