

U3L6TFEJF : <@U0564EGNY> nice! mind sharing the solution?
 U0564EGNY : I suppose the map should be mapv
 U0564EGNY : hmm, not sure how this is actually working without that :wink:
 U0564EGNY : oh, and the outer seq needs to be there because our tree may have multiple paths, duh
 U0564EGNY : ok, I have a bug when adding another path in the tree. I'm flattening at the wrong place
 U3L6TFEJF : here's my attempt with Specter:``

```
(def TreeValues
  (s/recursive-path [] p
    (s/if-path map?
      [(s/collect-one s/FIRST s/FIRST) s/MAP-VALS p]
      s/STAY)))
...

```

```
U3L6TFEJF : ``user => (s/select TreeValues m)
[{:a :b :c 1} {:a :b 2}]
user => (s/select TreeValues {:a {:fields {:b {:options [{:label "foo" :value "bar"}]}}}})
[{:a :fields :b :options [{:label "foo" :value "bar"}]}]
...

```

U3L6TFEJF : I haven't figured out yet how to not include the leaf itself
 U0564EGNY : map over the results with `butlast`
 U3L6TFEJF : just noticed that mine is not correct either, I get `[:a :b 2]` instead of `[:a :d 2]`
 U0J9LVB6G : <@U0564EGNY> Here's one implementation.
 ...

```
(defn all-paths [m]
  (letfn [(step [acc current-path x]
            (reduce-kv (fn [acc k v]
                        (let [path (conj current-path k)]
                          (if (map? v)
                              (step acc path v)
                              (conj acc path))))
                      acc, x))]
    (step [] [] m)))

(all-paths {:a {:b {:c 1} :d 2}}) ;=> [[:a :b :c] [:a :d]]
...

```

U0564EGNY : thanks, I'll try that on my data set
 U071CG4QY : <<https://stackoverflow.com/questions/21768802/how-can-i-get-the-nested-keys-of-a-map-in-clojure>>
 U3L6TFEJF : sweet, there you have a working version of what I was trying to do:``(def TreeValues
 (s/recursive-path [] p
 (s/if-path map?
 [s/ALL (s/collect-one s/FIRST) s/LAST p]
 s/STAY)))

user=> (map butlast (s/select TreeValues {:a {:b {:c 1} :d 2}}))
([[:a :b :c] [:a :d]])
...

U3JURM9B6 : what is the best data format for exchanging data between clojure and python?
 U064X3EF3 : depends on what your needs are, but you should look at <<https://github.com/cognitect/transit-format>>
 U1Z5X06NP : hey, has anybody ever had trouble requiring clj-time.types in a ns? I have the latest version from clojars but I get an exception when I compile complaining the ns clj-time.types doesn't exist or can't be found on the class path.
 U1Z5X06NP : the namespace clearly exists tho in the github repo
 U0NCTKEV8 : yeah, but the latest in git doesn't have to match the latest jar on clojars
 U0NCTKEV8 : there are also a few clj-time artifacts on clojars, are you sure you are using the one that maps to the github repo you are looking at?
 U0NCTKEV8 : you should check `lein deps :tree`
 U0NCTKEV8 : if any of your dependencies require one of these other clj-time artifacts (which unfortunately likely have the same namespace names) you will get all kinds of weird behavior

U1Z5X06NP : I followed the clojars link on github, maybe they just didn't push a newer artifact to maven?