Linking Netlogo to Java & Clojure

some general advice, feel free to take it or leave it...

linking NL to CLJ and/or Java

1. use the sock2 extension (see index at s573859921.websitehome.co.uk/pub/netlogo/cogs_nl_tools.htm) & read the user guide;
2. invoke NL directly from CLJ and use CLJ to call procedures in NL – this works nicely but there is currently no documentation on this.

exchanging data structures

socket communication is textual (Strings). If you want to exchange data structures (ie: a list, etc), convert them to Strings on the writer end then evaluate them on the reader end to rebuild them.

In NL use runresult...

```
observer> show (runresult "[1 \"banana\" 5 9]")
observer: [1 "banana" 5 9]
```

in CLJ use readstring & eval...

```
user=> (read-string "'(cat sat bat)")
(quote (cat sat bat))
user=> (eval (read-string "'(cat sat bat)"))
(cat sat bat)
```

sending commands/program code

Same idea as exchanging data.

In NL use run or runresult (depending on whether you are using a reporter ie: you expect a result)...

```
observer> show runresult "one-of [red blue green]"
observer: 55   ;; green has a value of 55

observer> run "ask patches [set pcolor (grey + 1)]"
```

NB: NL evaluates in any context (observer, agent, link, etc) and in the scope of run/runresult so you can access local vars

In CLJ use read-string & eval...

```
user=> (eval (read-string "(first '(mango melon peach))"))
mango
```

...but note that all CLJ evaluation takes place in the global context.