What are Clojure's four tools for concurrency?

# What is a ref?

# What are agents?

Reference or dereference objects.

"Re-assign" the reference.

How do you execute a transaction?

What properties to transactions guarantee?

What property is not guaranteed?

What does it mean for two transactions to be coordinated?

How can this be achieved?

What's the best way to update a ref in a single step?

Why might one use `conj` instead of `cons` when working with a list?

What is the signature of a suitable update function for `alter`?

What is the fundamental approach of STM's Multiversion Concurrency Control approach?

Why does it get expensive?

How can one improve upon the performance of `alter`?

Add a validation function to a ref to maintain transaction consistency.

Create an atom.