

Functional

- Functional programming models computation as the application of mathematical functions
- Clojure FP eschews the complexities (e.g. category theory, monads, etc.) of Haskell and Scala in favor of practicality
- Data pipelining via functions is a ubiquitous Clojure concept
- Java got more functional with Lambdas

Complete OO Support

- gen-class, proxy, reify, definterface, etc. make creation, design, and instantiation of Java classes easy
- defrecords allow the creation of value classes
- Clojure is more polymorphic than Java
 - interfaces work as Java
 - protocols support early- or late-binding single dispatch
 - multimethods support arbitrary dispatch
- Calling Java classes and APIs in Clojure is very simple
- In practice, you usually just code as data
- Calling Clojure from Java is straightforward
 - Caveat: Special care must be taken to prevent type erasure of generic types