Why Clojure Matters

Creighton Kirkendall

Principal Consultant SEI

Email: ckirkendall@gmail.com

Twitter: @crkirkendall

Github: https://github.com/ckirkendall

Why Clojure Matters

- Lisp Refresher
- Concurrency & State
  - Java Concurrency and Mutability
  - Immutability & Persistent Data Structures
  - Clojure Example
- Web Development
  - The Death of MVC
  - Pinot Noir (thirsty?)

Why Clojure Matters

Clojure Syntax (lisp)

Name	Clojure Syntax	Java Equivalent
Symbols	atom, foo-bar, *foo*, etc.	Variables Names
Literals	42, "foo", nil, true, false, \c, :foo	Same
Keywords	:foo (like symbols, prefixed with colon)	None
Lists	(a b c) & '(a b c)	LinkedList
Vectors	[a b c]	Array
Maps (hashes)	{:a 1 :b 1} or {:a 1, :b 2}	Map
Sets	#{:a :b :c}	Set

Why Clojure Matters

Clojure Syntax (lisp)

Clojure Syntax	Java Equivalent
(def a “test”)	String a=”test”;

(fn [x] (println x))
 	
#(println %1)
 	
(def tmp (fn [x] (println x))
 	
(defn tmp [x] (println x))
	No parallel for in-line functions
	
	public void tmp(Object x){
	System.out.println(x);
	}

(tmp “test”)	tmp(test);

Why Clojure Matters

Java and Concurrency

It's the mutable state, stupid. All concurrency issues boil down to coordinating access to mutable state. The less mutable state, the easier it is to ensure thread safety.

Java Concurrency in Practice

Why Clojure Matters

Java and Concurrency

Immutable objects are automatically thread-safe. Immutable objects simplify concurrent programming tremendously. They are simpler and safer, and can be shared freely without locking or defensive copying.

Java Concurrency in Practice

Why Clojure Matters

Java and Concurrency

A program that accesses a mutable variable from multiple threads without synchronization is a broken program

Java Concurrency in Practice

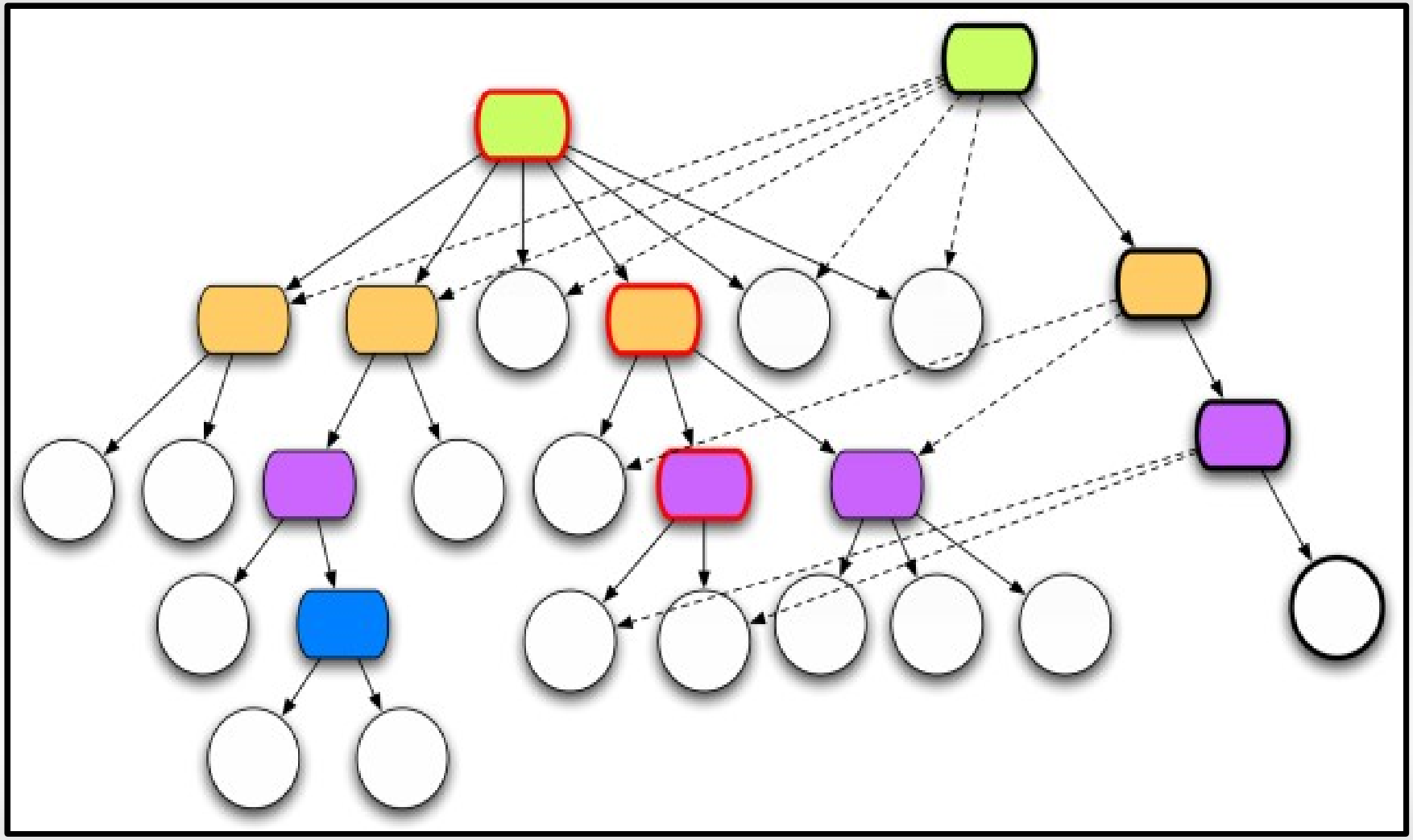
Why Clojure Matters

Java and Concurrency

Example

Why Clojure Matters

Immutability & Persistent Data Structures



Why Clojure Matters

Clojure Mutable State

- Atoms
  - swap!
- Refs
  - dosync
- Agents
  - send

Why Clojure Matters

Clojure Concurrency

Example

Why Clojure Matters

The Death Of MVC

- What is wrong with MVC.
  - Limited User Experience
- The Rise of Service Oriented UIs (AJAX, JSON)
  - Rich UI
  - Rise of JavaScript

Why Clojure Matters

Clojure and Clojurescript

Example

Why Clojure Matters

Cincy f(P)

Q & A