FUNCTIONAL PROGRAMMING PATTERNS

Michael Bevilacqua-Linn

Sr. Software Architect

Comcast (CIM)

http://pragprog.com/book/mbfpp/



mblinn.com

IWROTETHIS!



Functional Programming Patterns

in Scala and Clojure

Write Lean Programs for the JVM



PATTERNS?

Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma Richard Helm Ralph Johnson John Vlissides



Cover art © 1994 M.C. Escher / Cordon Art - Baarn - Holland. All rights reserved.

Foreword by Grady Booch



*

ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

PATTERNS?

A Pattern Language

Towns · Buildings · Construction



Christopher Alexander Sara Ishikawa · Murray Silverstein wm Max Jacobson · Ingrid Fiksdahl-King Shlomo Angel

PATTERNS?

I.) Independent Regions77.) House For A Couple

247.) Paving With Cracks Between The Stones

FUNCTIONAL PROGRAMMING?



- Purity Same input, same output
- Known Range and Domain

SOME DOMAINS LEND THEMSELVES TO PURITY

scala> println("hello, world")
hello, world

scala> println("hello, world")
sod off

SOME DON'T

Google			
some randon	n search	0	Ų
	Google Search	I'm Feeling Lucky	

RANDOM WEBSITE CIRCA 1998



RANDOM WEBSITE CIRCA 1998





99 AND 44/100 PURE



WHAT PATTERNS?

- Functional Interface
- Iterator
- Command
- Focused Mutability
- Customized Control Flow

FUNCTIONAL INTERFACE

"We'll give all interfaces that have just one method a name: functional interfaces."

http://cr.openjdk.java.net/~briangoetz/lambda/lambda-state-4.html



SERENDIPITY

New ideas go through stages of acceptance, both from within and without. From within, the sequence moves from "barely seeing" a pattern several times, then noting it but not perceiving its "cosmic" significance, then using it operationally in several areas, then comes a "grand rotation" in which the pattern becomes the center of a new way of thinking, and finally, it turns into the same kind of inflexible religion that it originally broke away from.

Alan Kay - ''The Early History Of Smalltalk'' © 1993 ACM

ITERATOR

"Provide a way to access the elements of an aggregate object sequentially without exposing its underlying representation."

Design Patterns: Elements of Reusable Object-Oriented Software

COMMAND

"Provide a way to access the elements of an aggregate object sequentially without exposing its underlying representation."

Design Patterns: Elements of Reusable Object-Oriented Software



FOCUSED MUTABILITY

Use mutable data structures in small, performance sensitive parts of a pro- gram hidden inside of a function, while still using immutable data throughout the majority.

Me

CUSTOM CONTROL FLOW

Create focused, custom control flow abstractions.

Me

CONCLUSION CAT CONCLUDES

