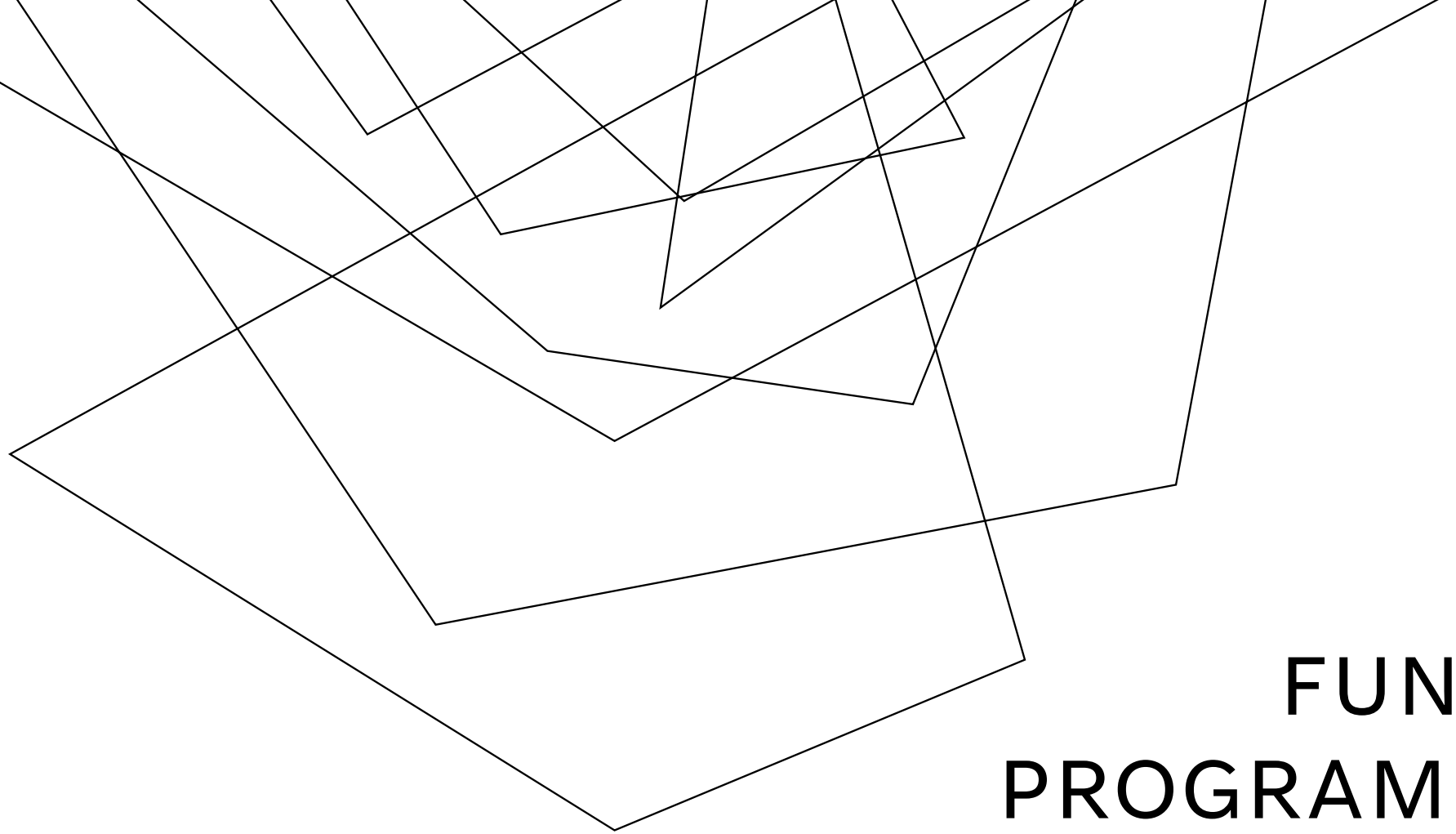


FUNCTIONAL PROGRAMMING ON JVM

Victor Osório

AGENDA

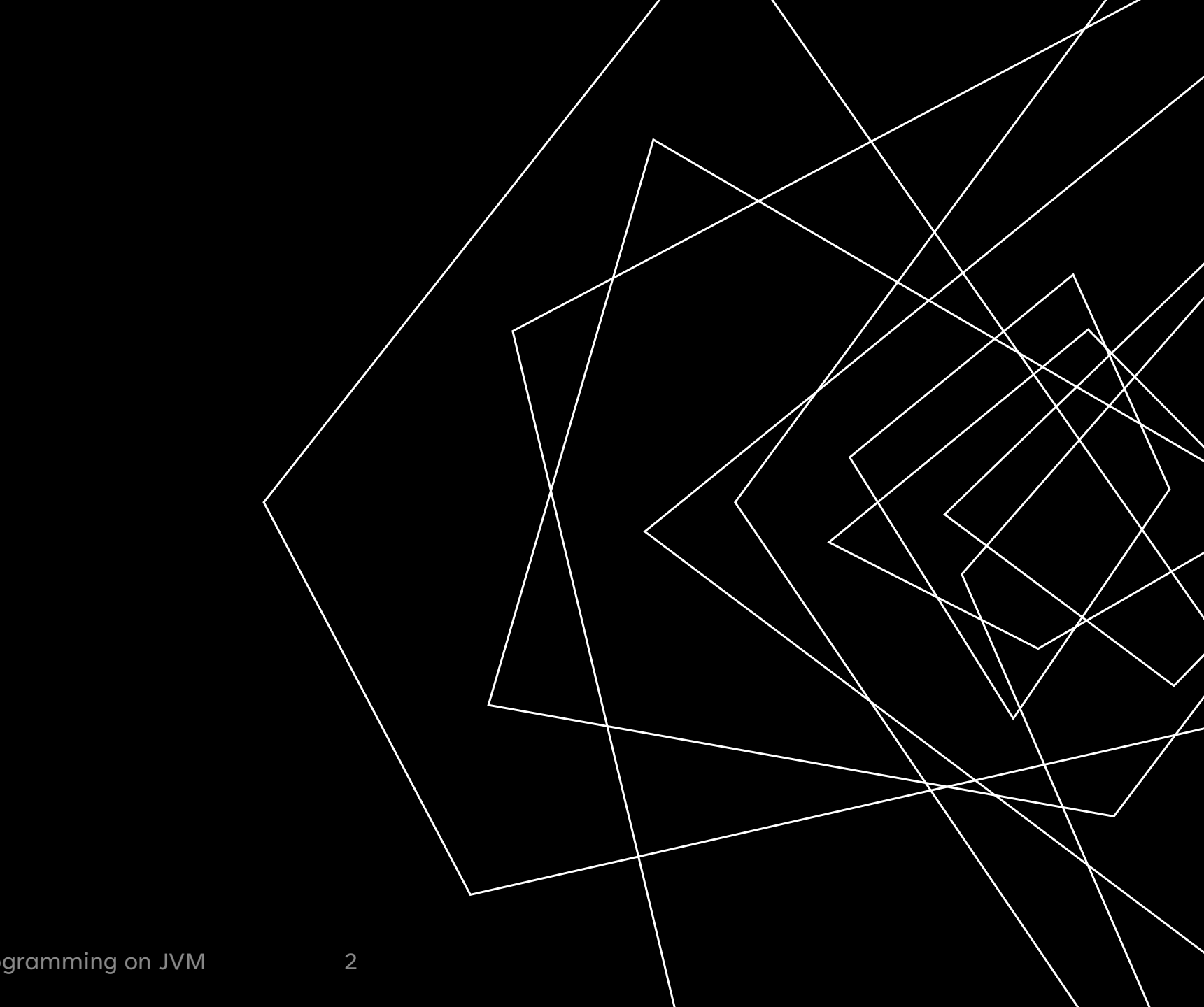
What is the JVM

Clojure

Scala

FP in Java

Summary



WHAT IS THE JVM

Java Virtual Machine is a Virtual Machine created to run Java Bytecode in any Operating System.

It allows **Write once, Runs anywhere** (WORA) and, in fact, is **Compile once, Runs anywhere**.

IMPORTANT JVM FEATURES

- Rich environment for monitoring/deploy
- A new release of the JVM is release every 6 months
- This is a very stable and optimized (JIT optimizes the code at runtime)
- Highly specified platform (latest release: https://docs.oracle.com/javase/specs/jvms/se19/html/index.html)

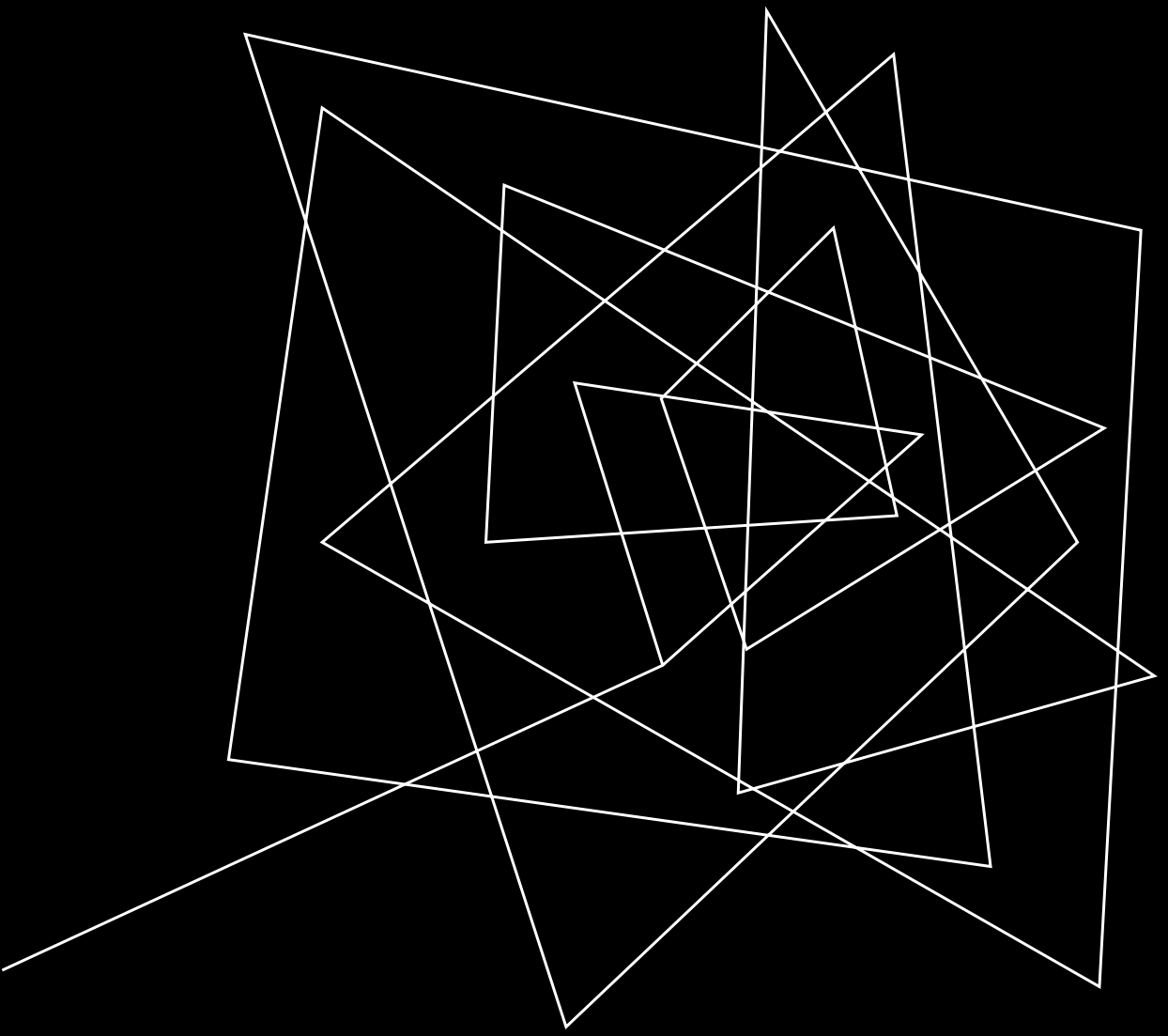
GAME CHANGER FEATURE

With an open specification anyone can create a new JVM language, using any paradigm

- Stability
- Interoperability
- Use any Java library

LANGUAGE SUPPORT

Object-Oriented ⟷ Functional



CLOJURE

An almost Lisp Language

CLOJURE GOALS

- Embrace an industry-standard platform
- Modernize LISP
- FP with Immutables

CLOJURE FEATURES

First-class functions

Functions can be arguments

Functions exists besides classes

Functions have arity overloading

Immutable Data Structures

Clojure implements a set of immutable data structures

Designed for Concurrency

Dynamic typing with common functions

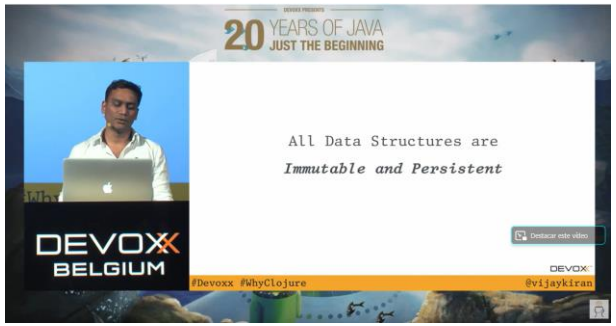
Recursive Looping

JVM do not allow Tail Recursion

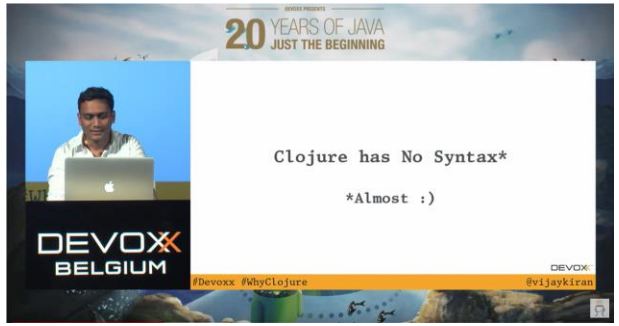
Clojure allows it by *recur* operator that does a constant-sapce recursive looping.

WHY CLOJURE? BY VIJAY KIRAN

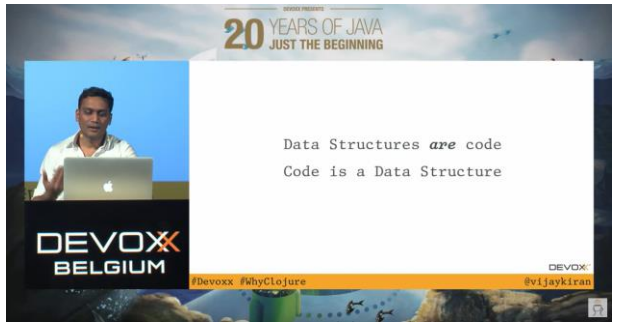
All Data Structures are **Immutable** and **Persistent**



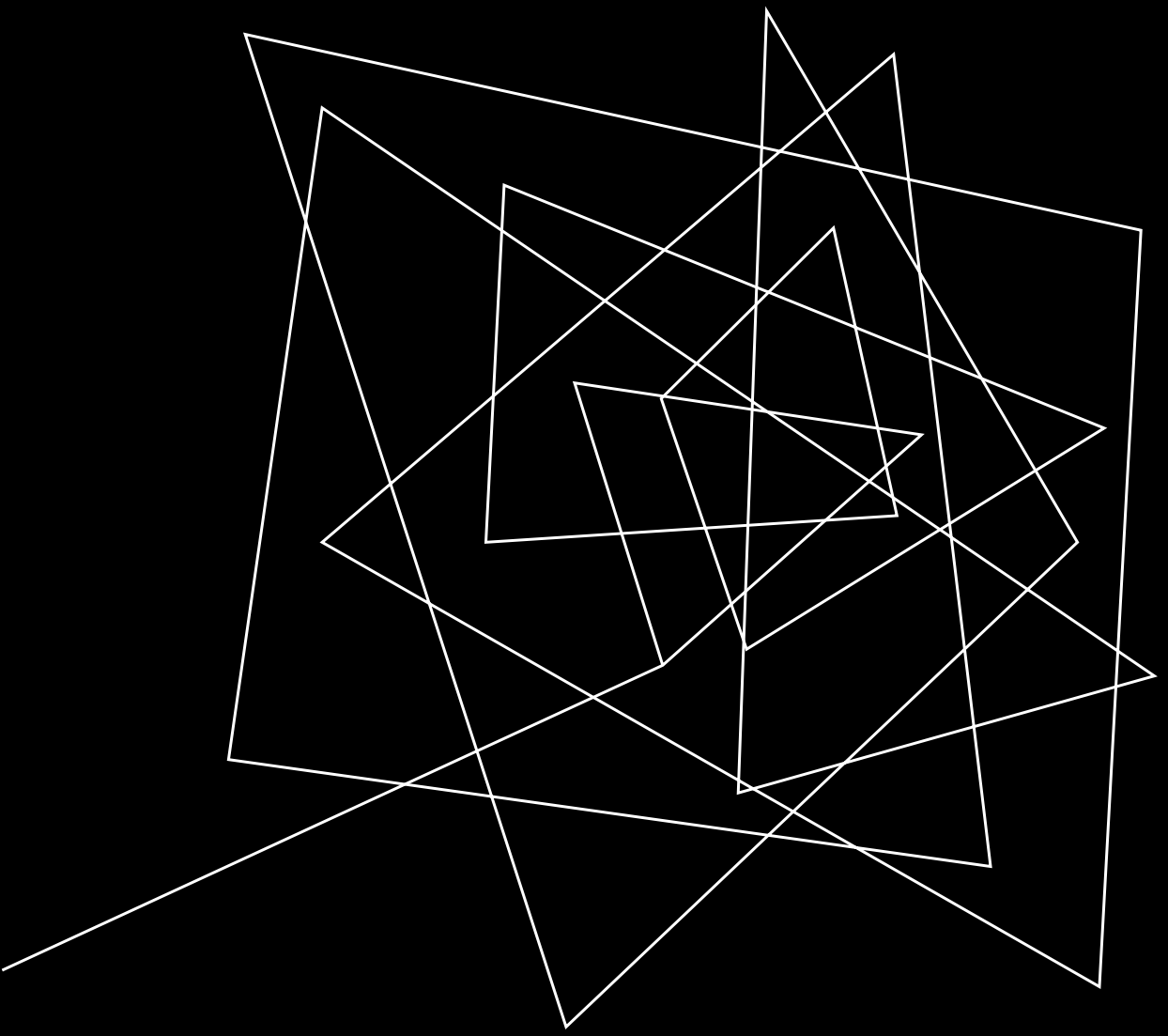
Clojure has **No Syntax**



Data Structures are code and Code is Data Structure



https://www.youtube.com/watch?v=SLRSOyR47Ro



SCALA

FP/OOP fusion

SCALA GOALS

- Concise and readable syntax
- Support the fusioon of FP and OOP
- Statically-typed but feels dynamic
- Expression based not statement based

SCALA FEATURES

First-class functions

Functions can be arguments

Functions exists besides classes

Higher-order functions

Lambdas (anonymous funcions)

Function Composition

Immutable Data Structures

Clojure implements a set of immutable data structures

Designed for Concurrency

Contextual abstraction

Allow omit boilerplate code

“Given” and “Using” (Scala 3) clauses implements “implicit” (Scala 2).

Extension Methods

Multiversal Equality

Context Functions

IMPORTANT PROJECTS

Apache Kafka

High-Performance message broker and streaming platform.

The broker is written in Scala

Akka

Actor's Reactive Platform

Build resilient Message Driven Applications

Apache Spark

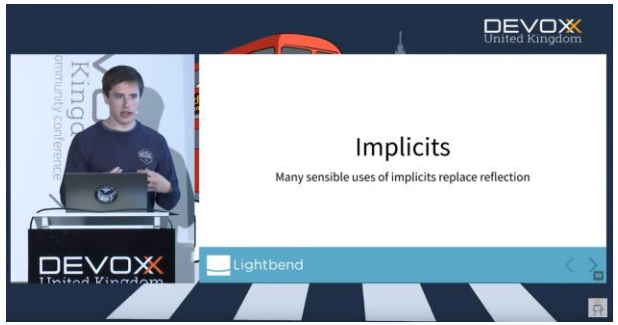
Data Processing Platform for Big Data

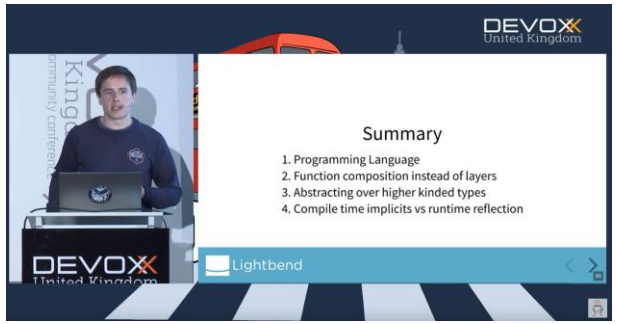
Classical MapReduce

SCALA LANGUAGE FEATURES

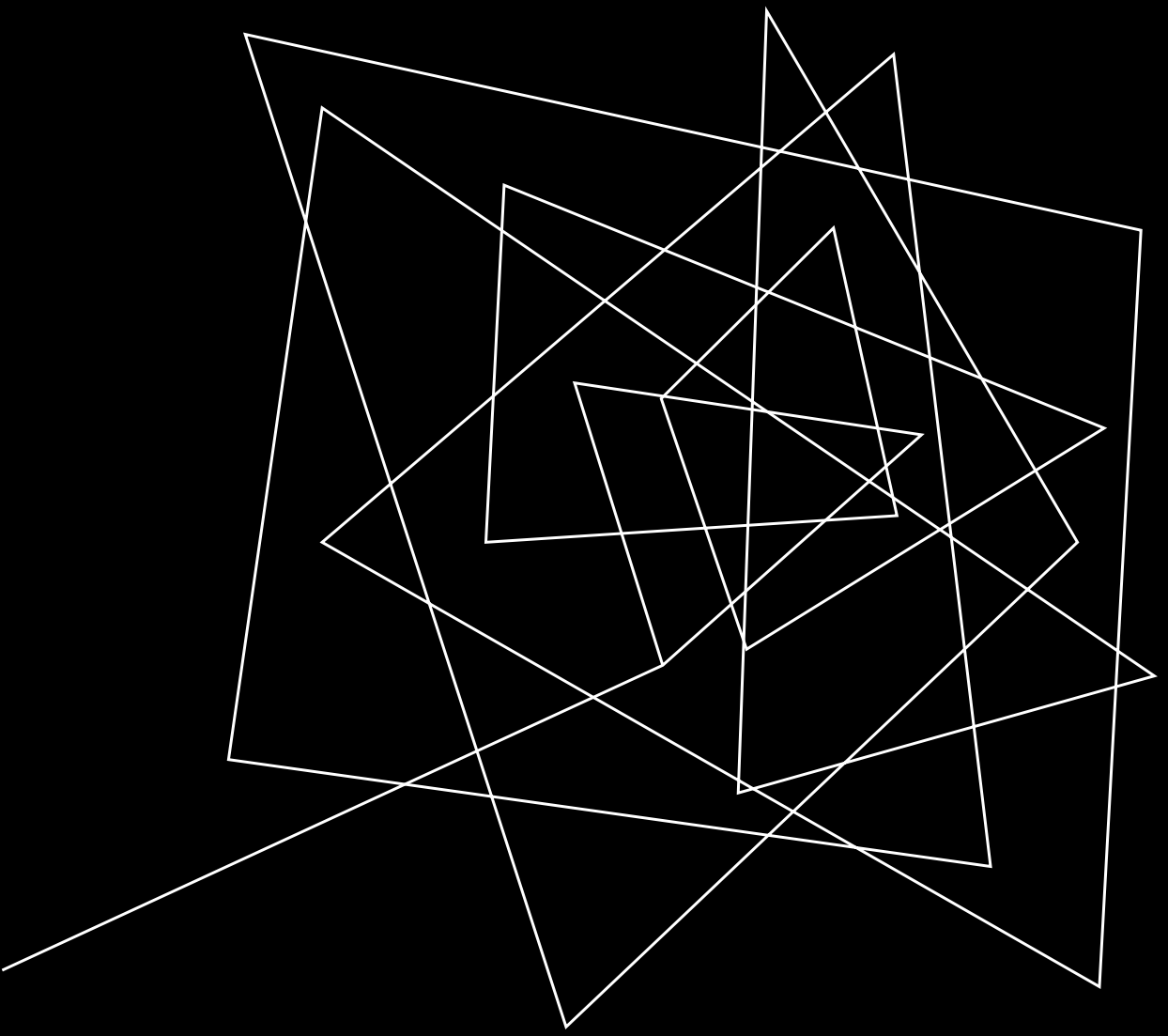
Function composition

Implicits





https://www.youtube.com/watch?v=4QIgEMvUfIE



FP IN JAVA

Java is going functional

JAVA PROJECTS

- Project Amber (Ongoing): Add “productivity-oriented” features to Java Language
- Project Lambda (JDK 8): High-Order Functions, Lazy Evaluation

PROJECT AMBER FEATURES

Patten Matching

Instanceof

Switch Expression

Type Patterns with Guarded Patterns

Record Patterns

Sealed Class

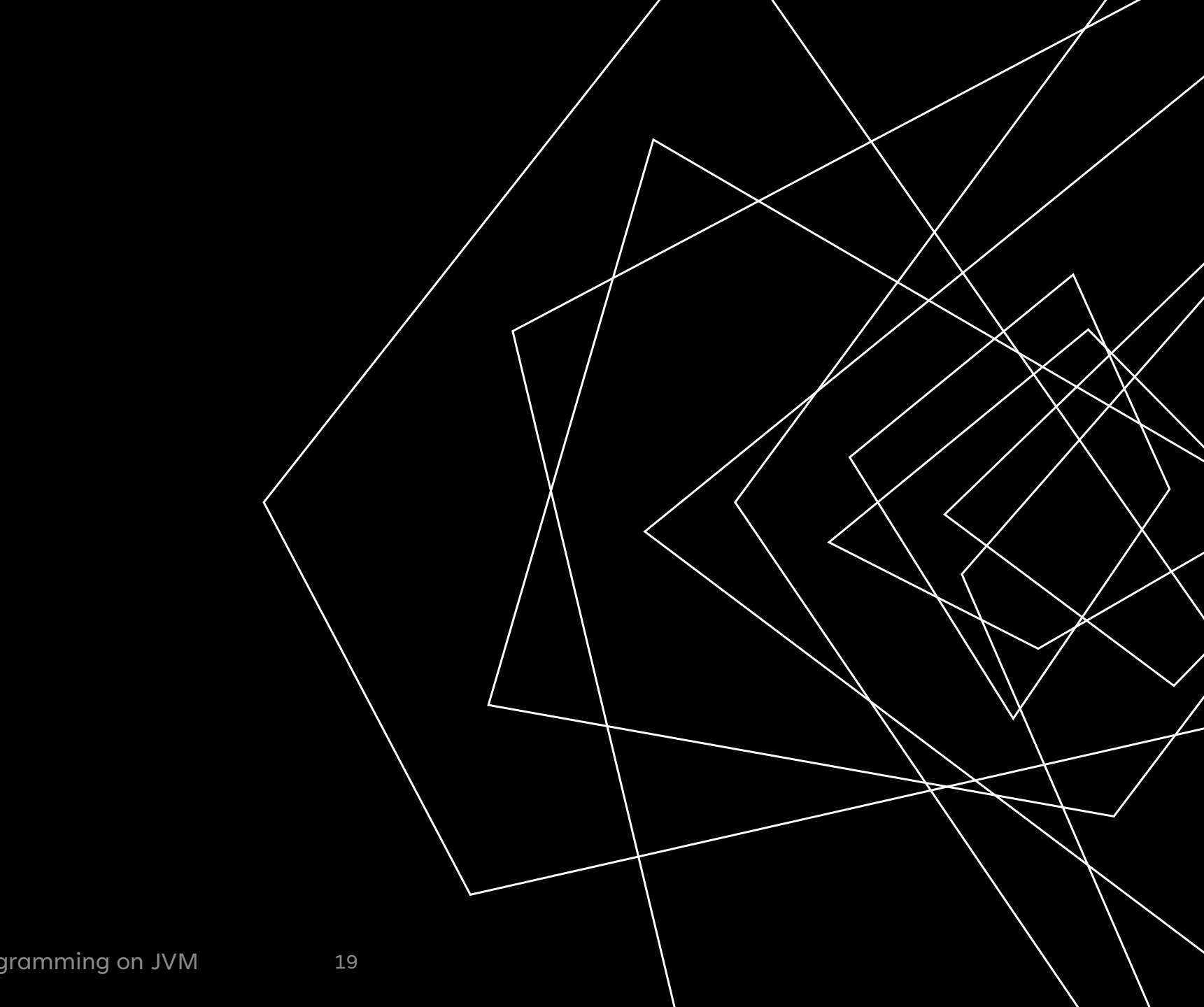
Prevents accidental inheritance

Allow Type Pattern Matching

Records

Data immutability as a language feature

SUMMARY



JAVA IS GOING FUNCTIONAL

JVM spec allows creating new languages that forces Java adopt new language features

Scala is forcing Java going functional