

on the JVM - do you ditch Java and go with
Kotlin, Scala, Groovy, Frege or do you keep
the Java faith?

Russel Winder

@russel_winder

russel@winder.org.uk

<https://www.russel.org.uk>

Russel Winder

- Ex theoretical physicist
- Ex UNIX systems programmer
- Ex academic:
 - Parallel programming
 - Software development and programming
 - HCI, UI, UX
- Ex Company director
- Ex Independent consultant
- Ex analyst
- Ex author
- Ex expert witness
- Ex trainer



Personal Programming History - 1/2

- Fortran
- Assembly Language
- Algol68
- Pascal
- C
- C++
- Scheme/Lisp
- Miranda/Haskell
- Java
- Python

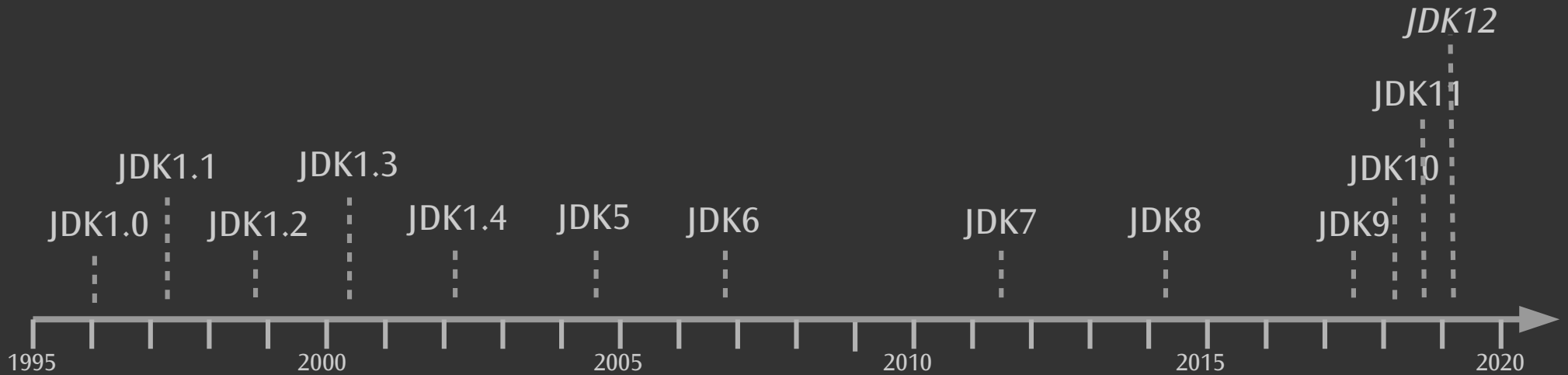
Personal Programming History - 2/2

- Ruby
- Groovy
- Scala
- Go
- Frege
- Kotlin
- Ceylon
- JavaScript/ECMAScript
- Rust



Introduction

A Bit of Java History



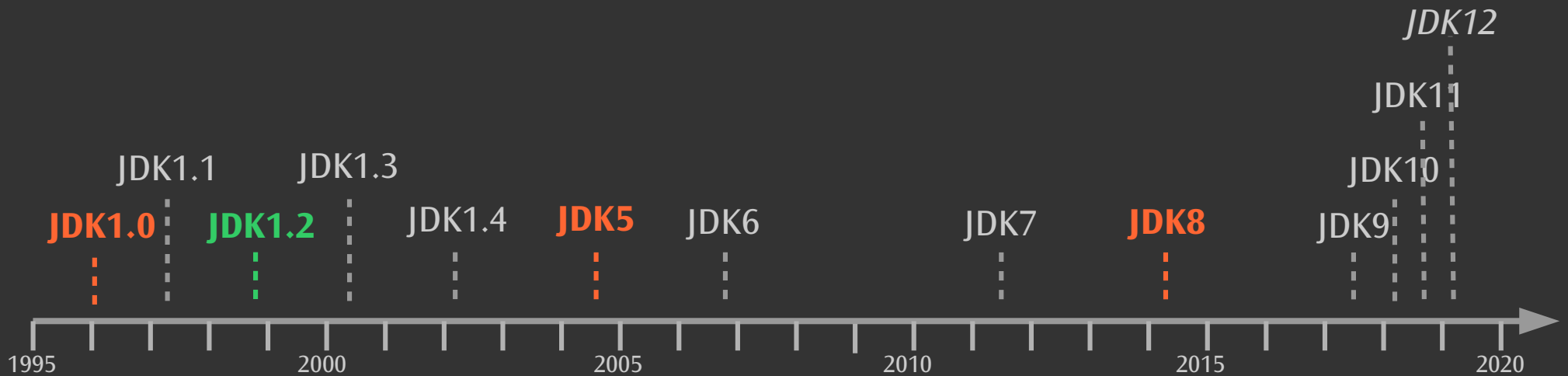
A Bit of Detail

- 2004 – Java 5 introduces:
 - generics, and
 - Type erasure.

A Bit More Detail

- 2014 – Java 8 introduces:
 - Lambda Expressions,
 - Method references,
 - Streams, and
 - `CompletableFuture`.

A Bit of Java History



Java is...

- ...an "old" programming language steeped in "imperative" and "object oriented" history.
- ...a programming language, evolving **very** slowly.
- ...not really keeping up with modern programming ideas and techniques.

The Java Platform is...

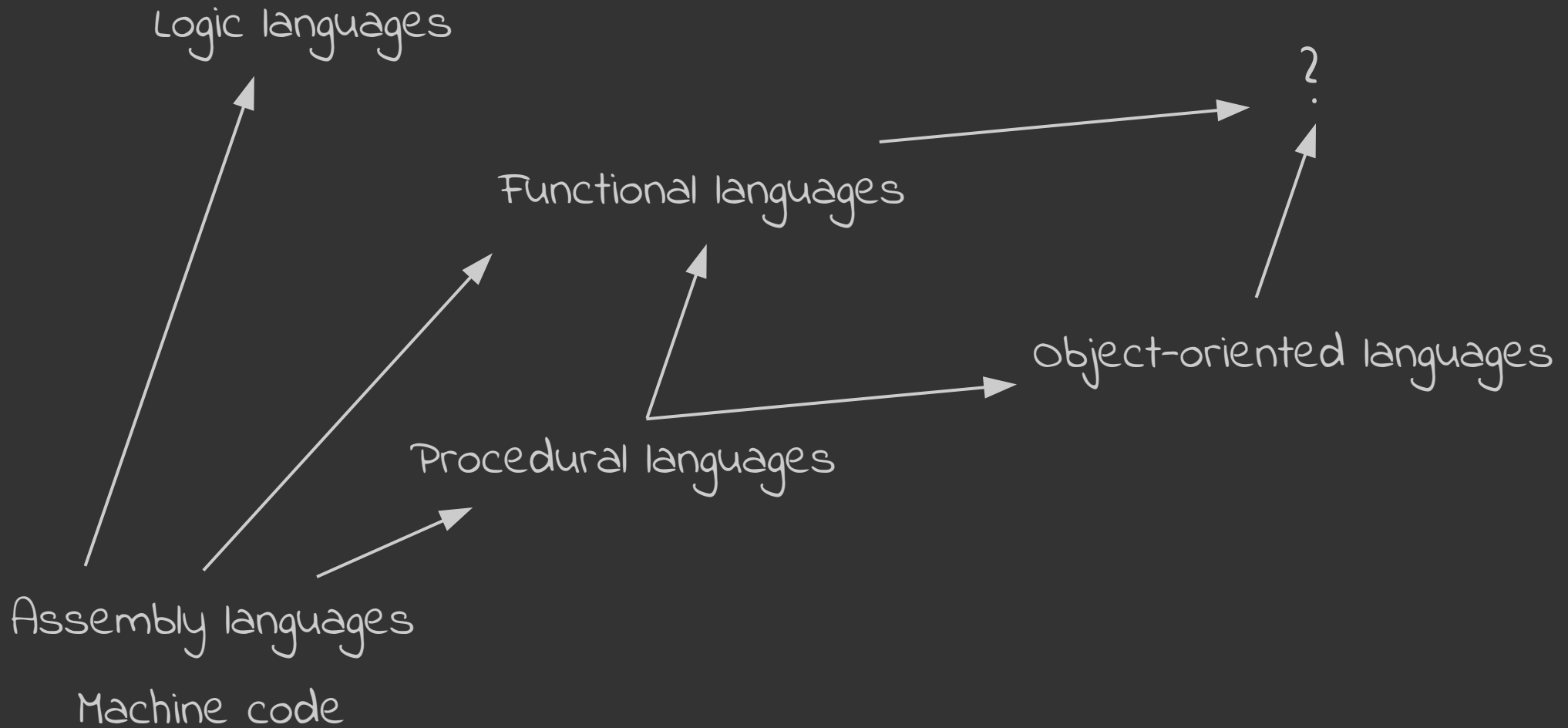
- ...a mature and extensive hardware-independent platform.
- ...evolving **very, very** slowly.
- ...a platform for many programming languages:
 - Java, Scala, Kotlin, Ceylon, Frege, **Groovy**, Fantom, Gosu, ...
 - **Groovy**, Clojure, JRuby, Jython, Golo, ...

Some Philosophy

Is Change Good?

Is Change Good?

- Change threatens backward compatibility, backward compatibility is determined to be sacrosanct, ergo **no**.
- Change enables new knowledge, new techniques, new tools, ergo **yes**.



Backward compatibility is the
enemy of progress.

Backward compatibility is the
excuse of the lazy to avoid work.

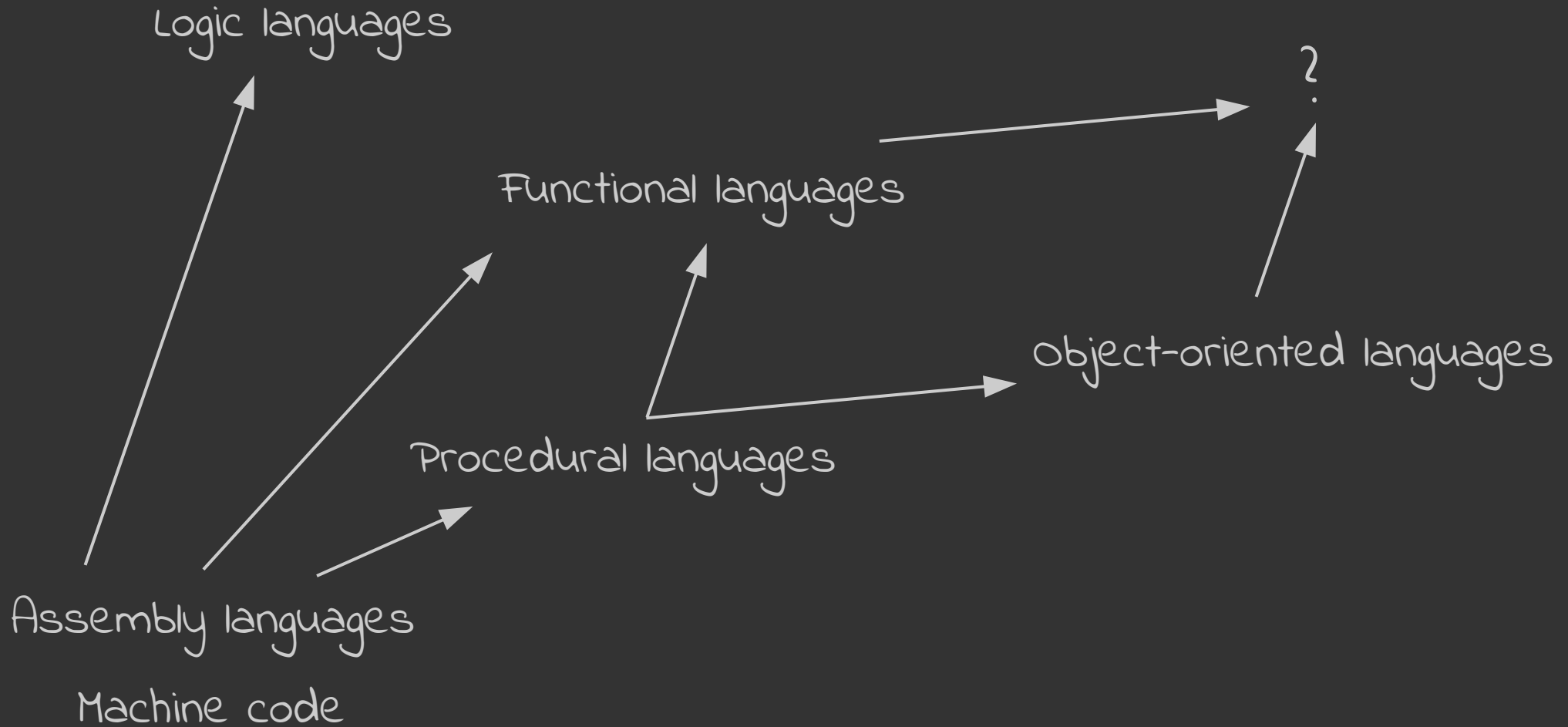
Backward compatibility is the
way applications become
outdated and broken.

Is backward compatibility the
biggest threat to Java?

Moving On

Distinguish Java the language from
JVM and Java Platform.

If Java will not change...



Scala, touted as a replacement for Java, incorporates functional programming into an object oriented language.

However...

Many took to Scala, but
many found it unsatisfactory.

Kotlin and Ceylon are two of the results.

The Two Languages

- Kotlin

- Interwork with Java allowing incremental change of codebase.
- Influenced by Java, Groovy, Scala.
- Superior type system to Java.

- Ceylon

- Replace Java but use the Java Platform.
- Module based from the outset.
- Far superior type system to Java.

Both Kotlin and Ceylon
Can compile to JVM or JavaScript.

Both Kotlin and Ceylon are
languages of the 2010s.

Hello world has to be done

```
/**
 * The Hello World program in Java
 *
 * @author Russel Winder
 */
public class HelloWorld_Java {
    public static void main(final String[] args) {
        System.out.println("Hello World.");
    }
}
```

Top level functions.

```
/**
 * The Hello World program in Kotlin.
 *
 * @author Russel Winder
 */
fun main(args:Array<String>) {
    println("Hello World.")
}
```

```
"The Hello World program in Ceylon."
by("Russel Winder")
shared
void run() {
    print("Hello World.");
}
```

Doing everything with classes and packages as in Java (and Scala) seen as too restrictive.

Adding top level functions, etc. seen as right, make the compiler do the work of creating classes for the JVM.

Fibonacci is required

Code

Let's look at anagrams
or word count

Code

Some Further Thoughts

Operator overloading was deemed too hard for programmers to cope with by the inventors of Java.

Most JVM languages since have put it back, one way or another.

Language evolution by replacement.

Kotlin and Ceylon represent two possible futures on the JVM.

Headlines

- Kotlin:

- More declarative than Java.
- Less code.
- Mixed language codebase.
- More expressive than Java.

- Ceylon:

- More declarative than Java.
- Modules.
- Far better type system than Java.
- More expressive than Java.

Both Kotlin and Ceylon run on Android.

Both Kotlin and Ceylon target JVM or JS.

A Challenge to You

Try all this in Java.

on the JVM - do you ditch Java and go with
Kotlin, Scala, Groovy, Frege or do you keep
the Java faith?

Russel Winder

@russel_winder

russel@winder.org.uk

<https://www.russel.org.uk>

End Note

The JVM is dead,

Long live the GraalVM

on the JVM - do you ditch Java and go with
Kotlin, Scala, Groovy, Frege or do you keep
the Java faith?

Russel Winder

@russel_winder

russel@winder.org.uk

<https://www.russel.org.uk>