# JAVA 11

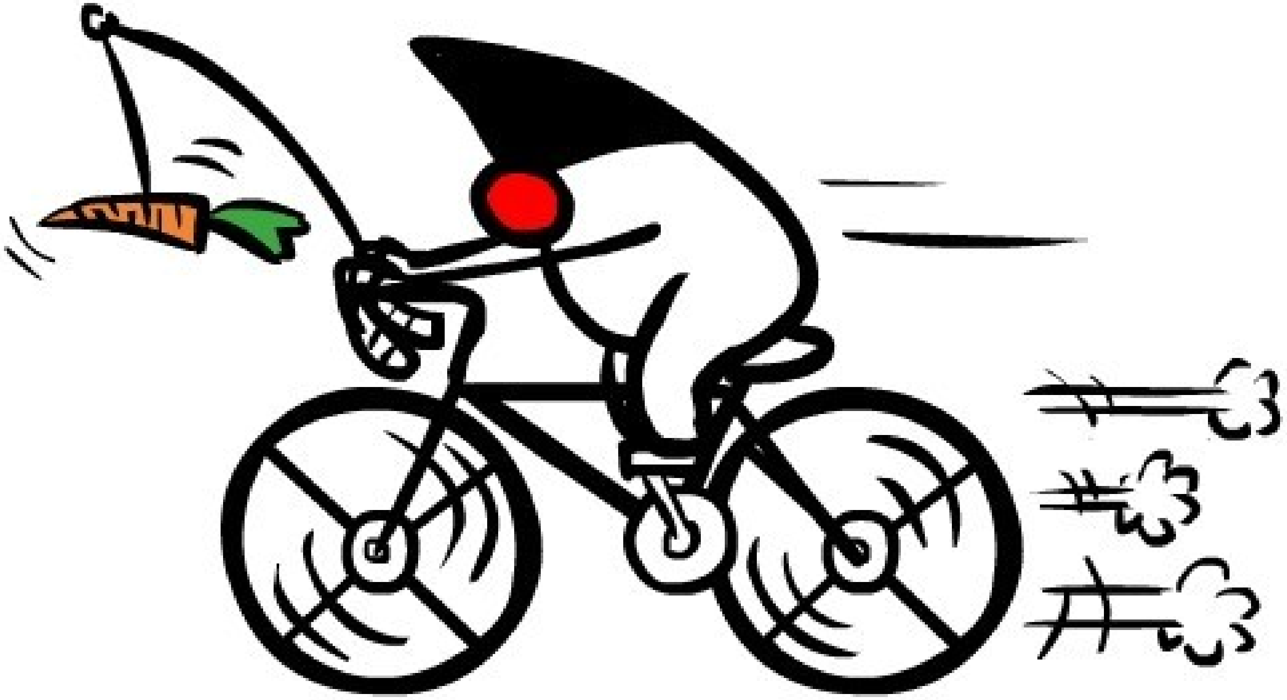
# JVM INTERNALS

Łukasz Sikora
sikora.lukasz.sl@gmail.com

# Agenda

- JEP 309: Dynamic Class-File Constants
- JEP 181: Nest-Based Access Control
- JEP 315: Improve Aarch64 Intrinsics

# Constant Dynamic - „CONDY”



# Road to condy

- Java 7:
  - invokedynamic
  - bootstrap methods
- JDK-8161256 & JDK-8161250
- Java 11:
  - ldc changes, constant dynamic pool

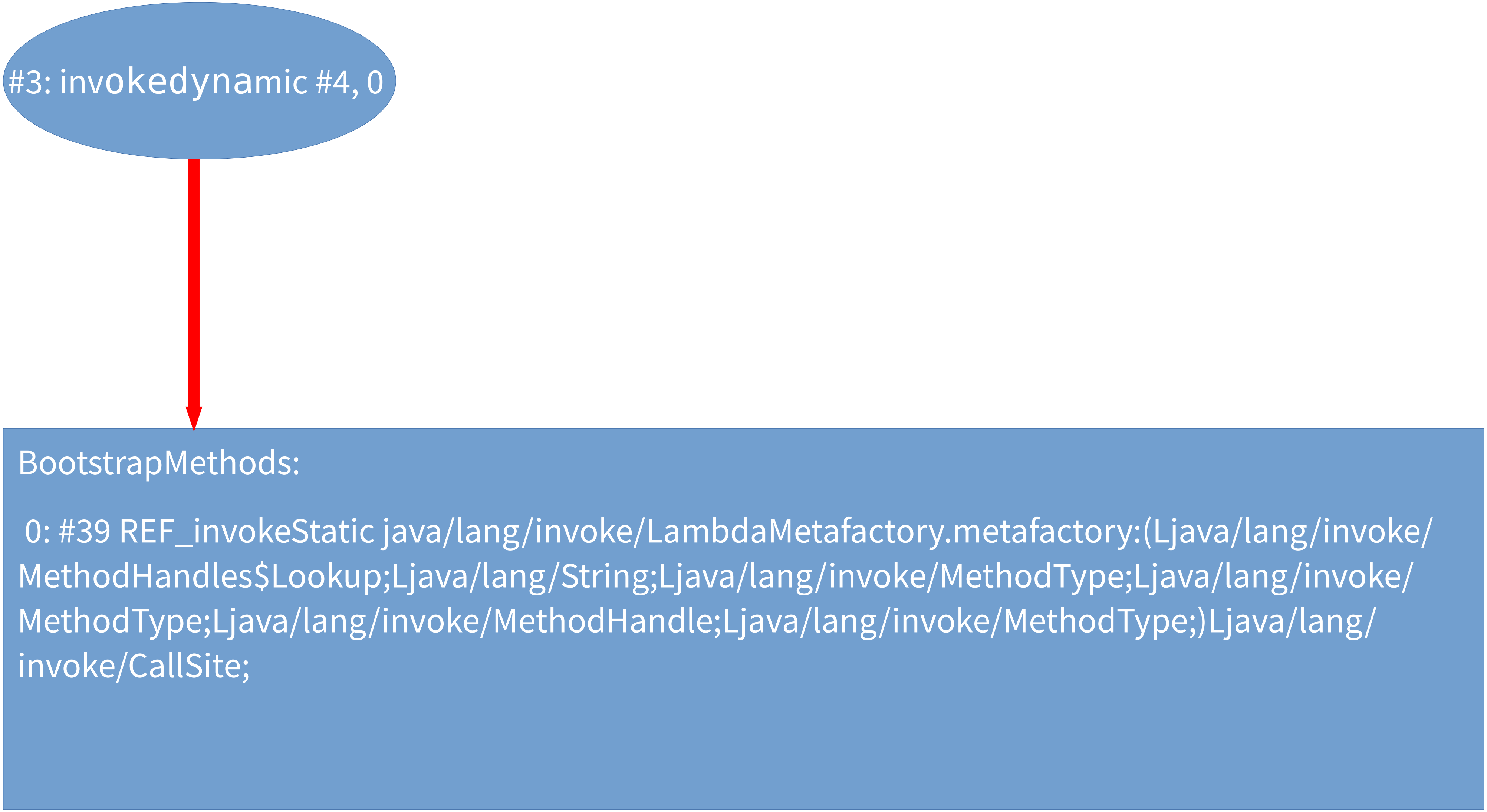
# Constant pools

```
Constant pool:
   #2 = Methodref          #5.#22      // java/lang/Object."<init>":()V
   #4 = Class              #25            // org/intrinsics/Nest$Birdie
   #5 = Class              #26            // java/lang/Object
   #6 = Utf8               this$0
  #21 = NameAndType        #6:#7         //this$0:Lorg/intrinsics/Nest;
  #22 = NameAndType        #8:#27         // "<init>":()V
  #23 = Class              #28            // org/intrinsics/Nest
  #25 = Utf8               org/intrinsics/Nest$Birdie
  #26 = Utf8               java/lang/Object
  #27 = Utf8               ()V
  #28 = Utf8               org/intrinsics/Nest
  #29 = Utf8               access$000
  #30 = Utf8               (Lorg/intrinsics/Nest;)Ljava/lang/String;
```

# Indy

#3: invokedynamic #4, 0

# Indy

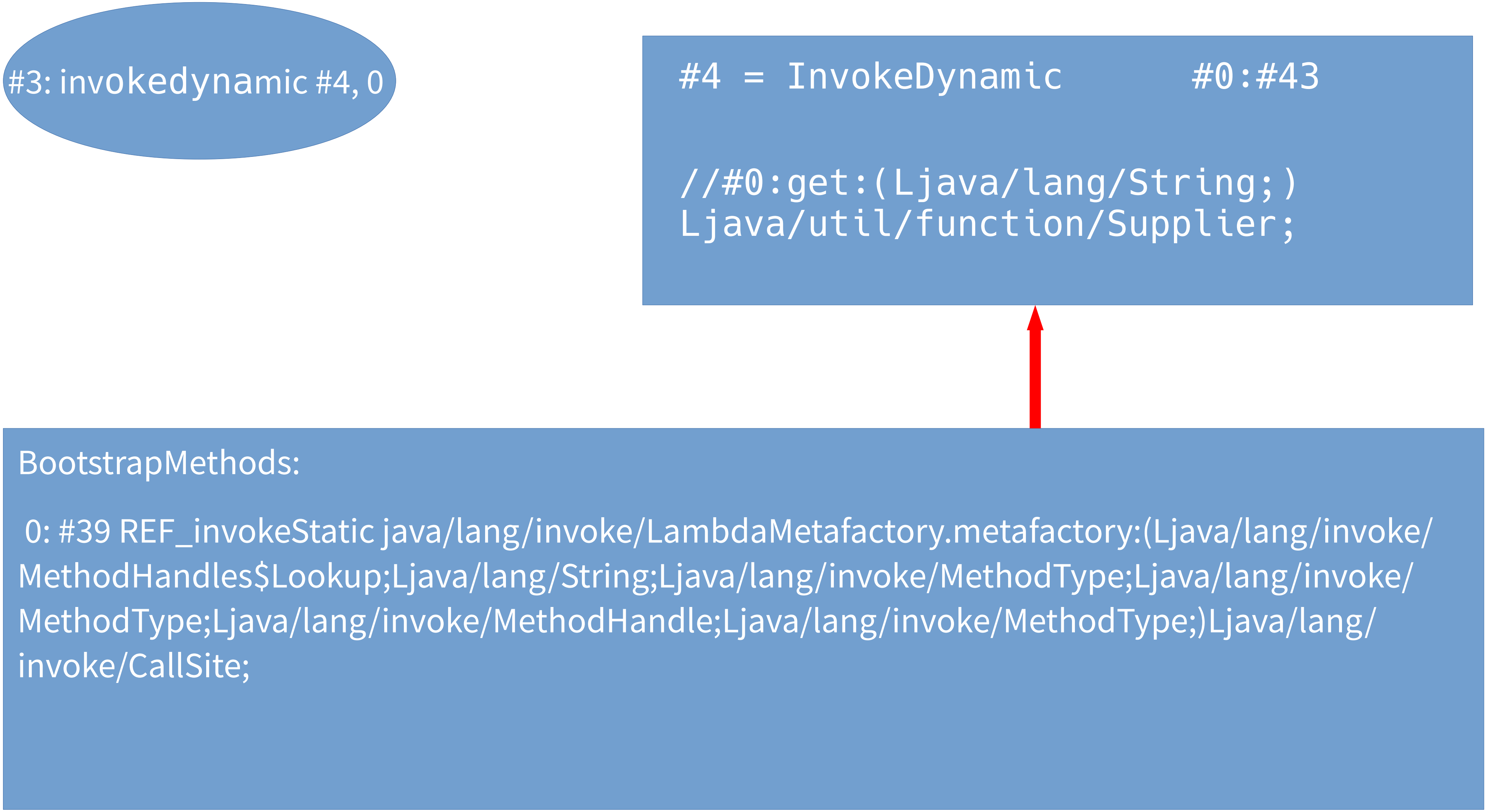


#3: invokedynamic #4, 0

BootstrapMethods:

0: #39 REF_invokeStatic java/lang/invoke/LambdaMetafactory.metafactory:(Ljava/lang/invoke/MethodHandles$Lookup;Ljava/lang/String;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodHandle;Ljava/lang/invoke/MethodType;)Ljava/lang/invoke/CallSite;

# Indy



#3: invokedynamic #4, 0

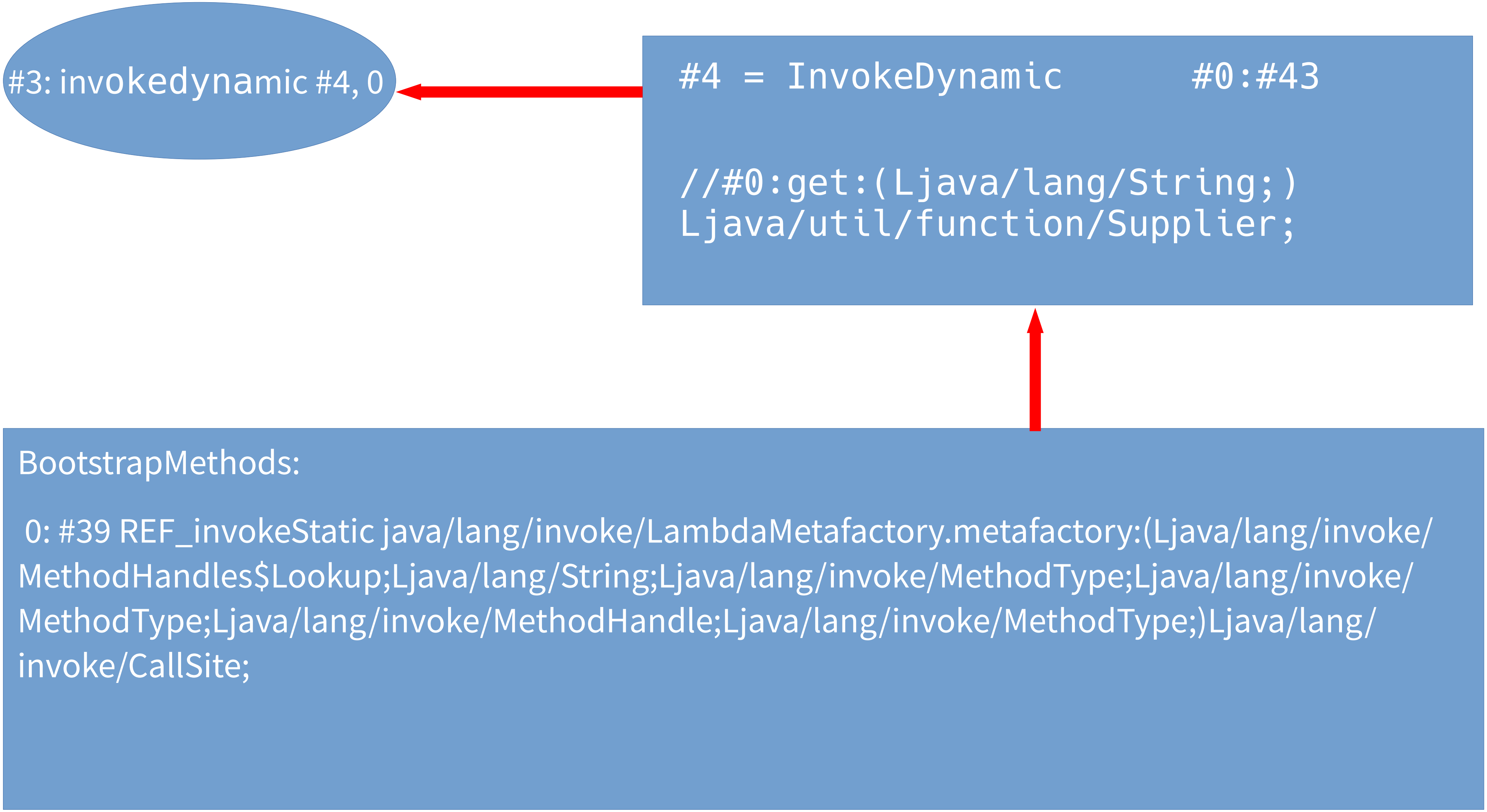
```
#4 = InvokeDynamic      #0:#43

//#0:get:(Ljava/lang/String;)
Ljava/util/function/Supplier;
```

BootstrapMethods:

0: #39 REF_invokeStatic java/lang/invoke/LambdaMetafactory.metafactory:(Ljava/lang/invoke/MethodHandles$Lookup;Ljava/lang/String;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodHandle;Ljava/lang/invoke/MethodType;)Ljava/lang/invoke/CallSite;

# Indy



```
#3: invokedynamic #4, 0
```

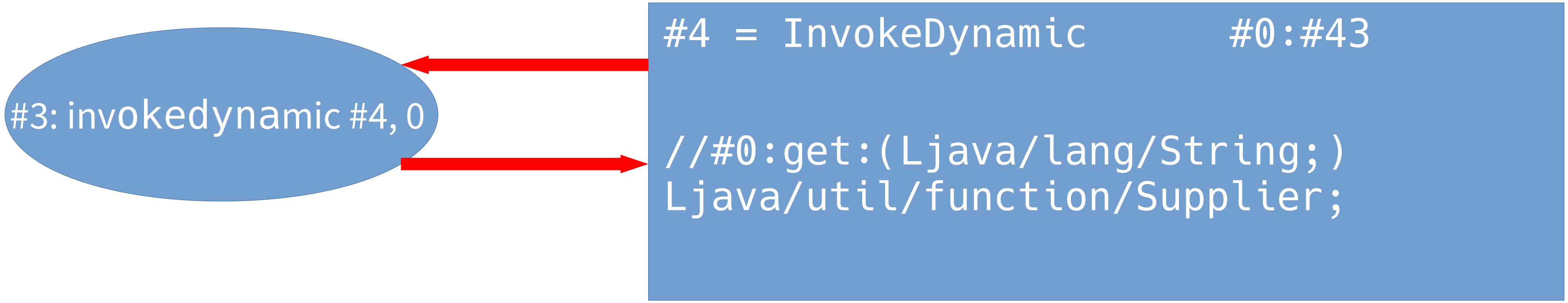
```
#4 = InvokeDynamic      #0:#43

//#0:get:(Ljava/lang/String;)
Ljava/util/function/Supplier;
```

```
BootstrapMethods:

 0: #39 REF_invokeStatic java/lang/invoke/LambdaMetafactory.metafactory:(Ljava/lang/invoke/
MethodHandles$Lookup;Ljava/lang/String;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/
MethodType;Ljava/lang/invoke/MethodHandle;Ljava/lang/invoke/MethodType;)Ljava/lang/
invoke/CallSite;
```

# Indy



#3: invokedynamic #4, 0

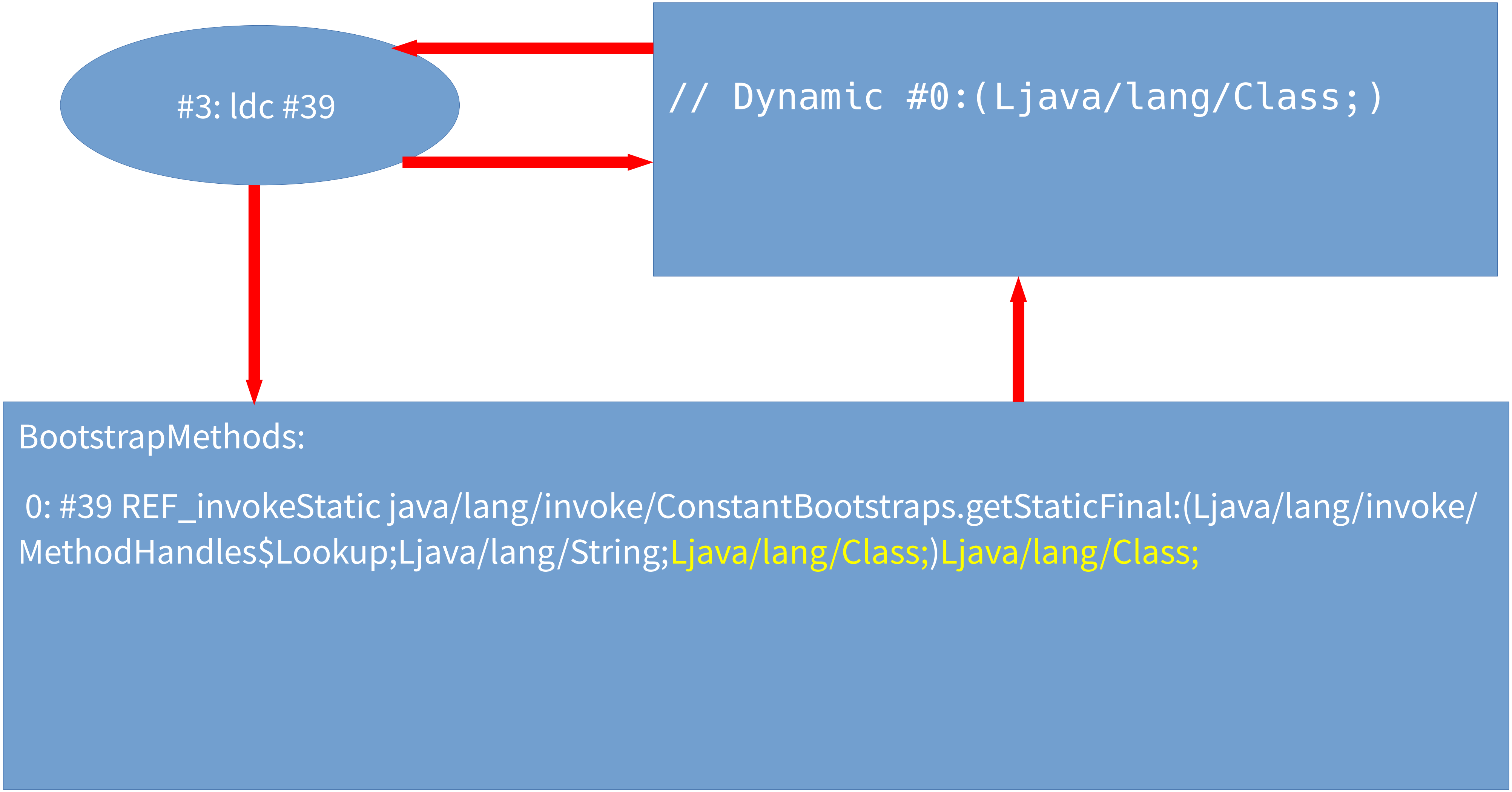
```
#4 = InvokeDynamic      #0:#43

//#0:get:(Ljava/lang/String;)
Ljava/util/function/Supplier;
```

BootstrapMethods:

 0: #39 REF_invokeStatic java/lang/invoke/LambdaMetafactory.metafactory:(Ljava/lang/invoke/MethodHandles$Lookup;Ljava/lang/String;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodType;Ljava/lang/invoke/MethodHandle;Ljava/lang/invoke/MethodType;)Ljava/lang/invoke/CallSite;

# Condy



#3: ldc #39

```
// Dynamic #0:(Ljava/lang/Class;)
```

BootstrapMethods:

0: #39 REF_invokeStatic java/lang/invoke/ConstantBootstraps.getStaticFinal:(Ljava/lang/invoke/MethodHandles$Lookup;Ljava/lang/String;Ljava/lang/Class;)Ljava/lang/Class;

# Why bother

**Things that can get to constant pool with constant dynamic:**

- **null**
- **enum constants (JDK-8161250)**
- **primitives (long.class)**
- `VarHandle`

# Why bother

- raw bytes (instead of byte array)
- lambdas
- value types
- any object that with resolved references is runtime constant

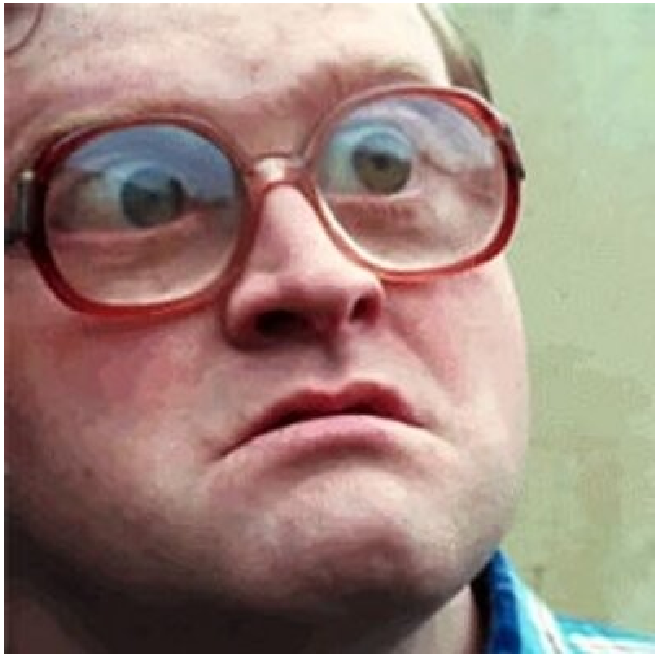
# Nestmates

# The problem

- **Java language spec:**

  **Nested class has access to private members of outer class**

- **JVM:**



# Private member access pre 11

```java
package org.intrinsics;

public class Nest {
    private String nest = "comfy";

    public class Birdie{

        public String getNest(){
          return nest;
        }
    }
}
```

# Pre java 11 - Birdie

```
  public java.lang.String getNest();
    descriptor: ()Ljava/lang/String;
    flags: (0x0001) ACC_PUBLIC
    Code:
      stack=1, locals=1, args_size=1
         0: aload_0
         1: getfield      #1                  // Field
this$0:Lorg/intrinsics/Nest;
         4: invokestatic  #3                  // Method
org/intrinsics/Nest.access$000:(Lorg/intrinsics/Nest;)Ljava/lang/Stri
ng;
         7: areturn
      LineNumberTable:
        line 9: 0
      LocalVariableTable:
        Start  Length  Slot  Name   Signature
            0       8     0  this   Lorg/intrinsics/Nest$Birdie;
}
```

# Pre Java 11 - Nest

```
  static java.lang.String access$000(org.intrinsics.Nest);
    descriptor: (Lorg/intrinsics/Nest;)Ljava/lang/String;
    flags: ACC_STATIC, ACC_SYNTHETIC
    Code:
      stack=1, locals=1, args_size=1
         0: aload_0
         1: getfield      #1                  // Field nest:Ljava/lang/String;
         4: areturn
      LineNumberTable:
        line 3: 0
      LocalVariableTable:
        Start  Length  Slot  Name   Signature
            0       5     0    x0   Lorg/intrinsics/Nest;
}
SourceFile: "Nest.java"
InnerClasses:
     public #7= #6 of #4; //Birdie=class org/intrinsics/Nest$Birdie of class
org/intrinsics/Nest
```

# Java 11 – Nest.class

```
NestMembers:
  org/intrinsics/Nest$Birdie
InnerClasses:
  public #7= #6 of #4;                    // Birdie=class org/
intrinsics/Nest$Birdie of class org/intrinsics/Nest
```

# Java 11 - Nest$Birdie

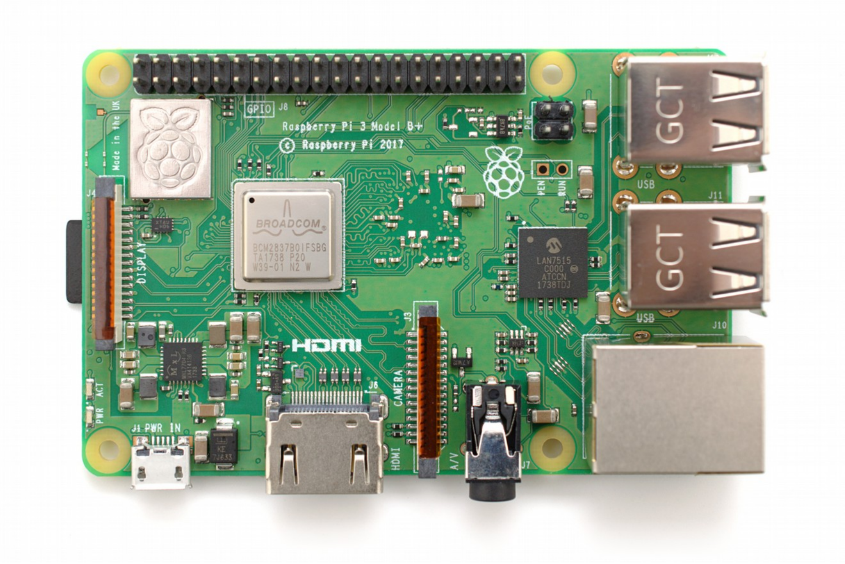
```
  public java.lang.String getNest();
    descriptor: ()Ljava/lang/String;
    flags: (0x0001) ACC_PUBLIC
    Code:
      stack=1, locals=1, args_size=1
         0: aload_0
         1: getfield      #1                  // Field
this$0:Lorg/intrinsics/Nest;
         4: getfield      #3                  // Field
org/intrinsics/Nest.nest:Ljava/lang/String;
         7: areturn
      LineNumberTable:
        line 9: 0
      LocalVariableTable:
        Start  Length  Slot  Name   Signature
            0       8     0  this   Lorg/intrinsics/Nest$Birdie;
}
SourceFile: "Nest.java"
NestHost: class org/intrinsics/Nest
InnerClasses:
  public #14= #4 of #22;                  // Birdie=class
org/intrinsics/Nest$Birdie of class org/intrinsics/Nest
```

# Aarch64=Android?

# Not this time

Testing

Intrinsics performance will be tested on Cavium ThunderX, ThunderX2 and Cortex A53 hardware using JMH benchmarks.



# What was changed?

**AArch64**

- **java.lang.Math:**
  - sin (sine trigonometric function)
  - cos (cosine trigonometric function)
  - log (logarithm of a number)

# And

- **Use the ARM NEON instruction set.**
- **UseSIMDForMemoryOps**

# Impact

- String::compareTo
- String::indexOf,
- StringCoding::hasNegatives,
- Arrays::equals,
- StringUTF16::compress,
- StringLatin1::inflate
- Checksum calculations.

# Summary

- **Lambdas nad value objects go contant in future**
- **class<nest<package<module<jar**
- **Aarch64 is even faster**

# Q&A

# Resources

- https://openjdk.java.net/jeps/181
- https://openjdk.java.net/jeps/309
- https://openjdk.java.net/jeps/315
- https://www.youtube.com/playlist?list=PLX8CzqL3ArzVnxC6PYxMlngEMv3W1pIkn