WHAT JAVA CAN LEARN FROM HASKELL AND VICE VERSA

NICK LINKER

NOVEMBER 2016, ISSART

1 WHAT JAVA CAN LEARN FROM HASKELL

1.1 HASKELL IS A BIG LANGUAGE

1. Expressive (though terse) syntax
2. Immutable data structures
3. Pure functions *with currying*
4. No nulls
5. Algebraic Data Types and pattern matching
6. Type classes
7. Concurrency
8. Tools: GHCI, stack

1.2 THE MOST IMPACT IS DONE BY

1. Syntax, type inference
2. ADT & pattern matching
3. Type classes
4. Pure functions and explicit side effects

2 WHAT HASKELL CAN LEARN FROM JAVA

2.1 ANYONE KNOWS JAVA

- Much lower learning curve.
- The first time it is really hard to get even simple things.
- Haskell forces to control side effects even when it is not needed.
- Forces to spend much more time on resolving compiler errors.
- Sometimes errors are hard to comprehend
- Advanced types (type families and higher kinded types) appear even in ordinary things (like running REST server)
- Haskell compiler implements a lot of extensions
- Stackoverflow
- However it is hard to get it done right
- `cow.eat(grass)`, `grass.beEatenBy(cow)` or `field.interact(cow, grass)`
- subtype polymorphism impossible to

2.2 CONVENIENT TOOLING

- Developer experience is good
- Intellij Idea
- Although not always good from the long-term perspective

3 TO USE OR NOT TO USE?

1. Use in the domains when it is really shine: compiler construction, provable correct code.
2. There are a lot of compilers written in Haskell: GHC, Corrode, Elm, Agda, Kaleidoscope, Purescript, Pandoc.
3. Many libraries to support compiler constructions.
4. It is good for small teams with eagerness to use FP.
5.

4 TO LEARN OR NOT TO LEARN?

1. Yes!
2. Many FP concepts are here is the most explicit way.
3. It is very different.
4. Get used to type systems other to Java.
5. Concurrency and parallelism.
6. Monads.

5 QUESTIONS?