

Polyglot development using GraalVM

- Twitter analysis

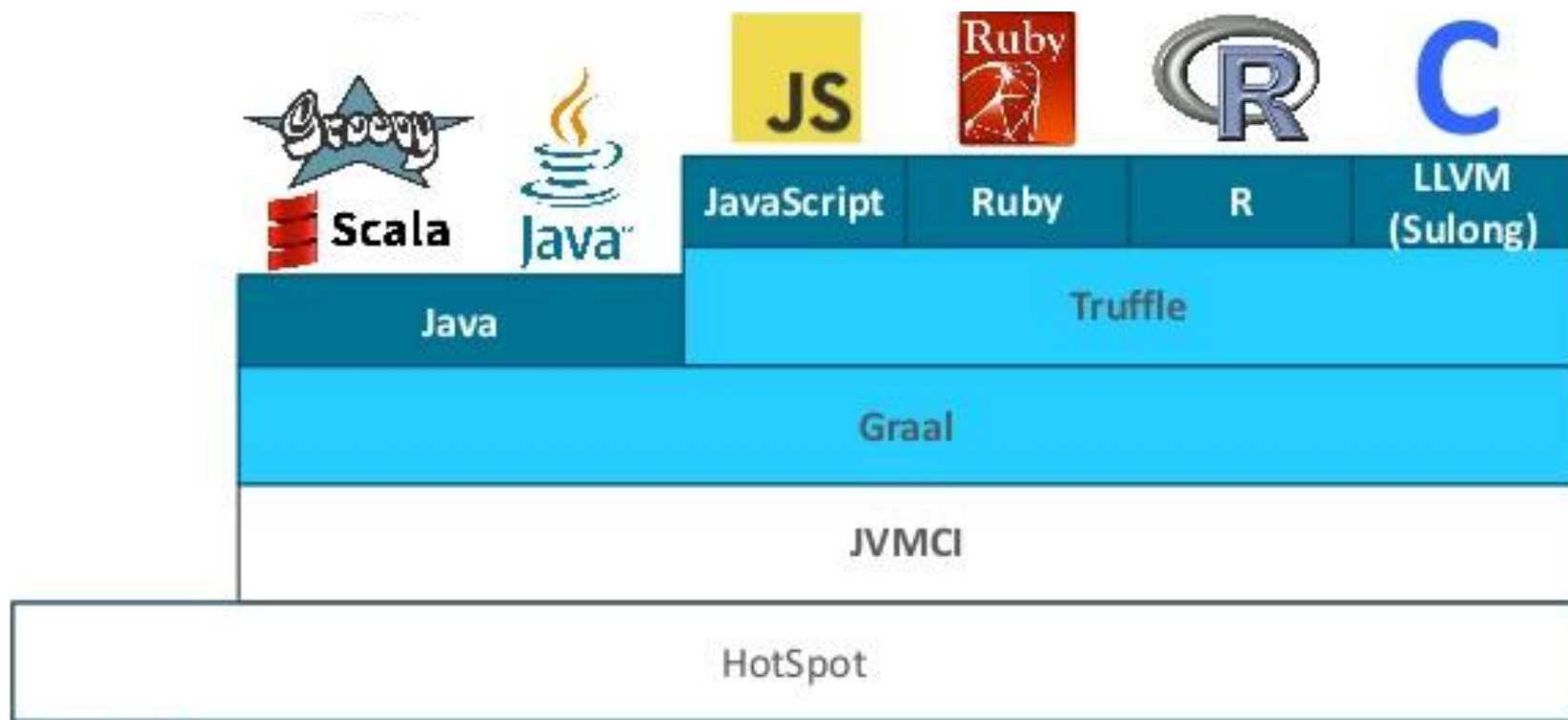
SCG Seminar Project: Flurin Truebner

Supervisor: Boris Spasojević

Project Goal

- Polyglot Twitter analysis
 - Node.js
 - Java
 - Ruby
 - R
- Using GraalVM

GraalVM



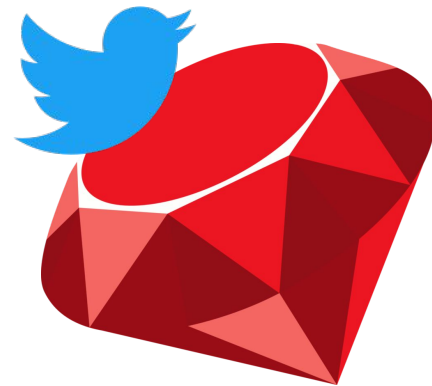
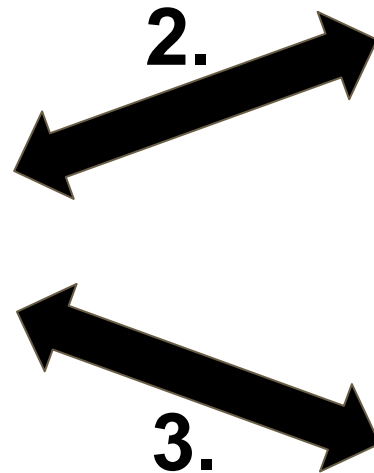
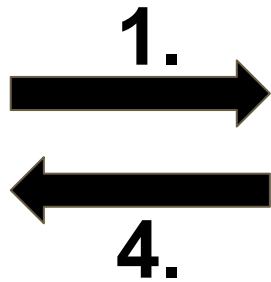
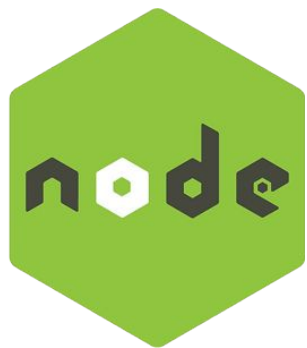
GraalVM 1.0

- [GraalVM.org](https://graalvm.org)
- Polyglot API
- Currently using 0.32

Some benefits of the GraalVM

- Interoperability
 - Less overhead
 - Polyglot API
- Performance
 - Truffle implementations
 - graal > c2 for modern workloads (streams)
- Tooling
- Native Images

Project



Demo

Node.js -> Java



Nodejs:

```
const JavaHostType = Java.type('JavaHost');  
var JavaHost = new JavaHostType();
```

```
res.send(JavaHost.tweetSentiment(searchTerm, tweetCount));
```

Java:

```
public String tweetSentiment(String searchTerm, int tweetCount){}
```


Java -> Truffle Languages (Ruby, R)

Java:

```
srcFile = new File("twitterGet.rb");  
srcLang = Source.findLanguage(srcFile);  
source = Source.newBuilder(srcLang, srcFile).build();  
rbTwitterGet = context.eval(source);
```

```
Value tweets = rbTwitterGet.execute(searchTerm, tweetCount);  
String rPlot = rSentAna.execute(tweets).asString();
```

Ruby:

```
def getTweets(searchTerm, tweetCount)  
  tweets = client.user_timeline(searchTerm, count: tweetCount)
```

Polyglot Value in Java



```
if(value.canExecute()) { value.execute(); }  
  
(value.isString()) ? value.asString() : null;  
  
(value.hasArrayElements()) ? value.getArrayElement(0) : null;
```

Problems / Limitations

- Missing Examples / Documentation
- Truffle Languages
 - e.g. library support

Conclusion

- Easy to setup
- Polyglot API works out of the box
- Fast
- Motivation

Questions ?