

# Making shell scripts Groovy



# about me

- programmer + DevOps enthusiast
  - Java, Groovy, learning Clojure
  - Linux, Ansible, Liquibase, Maven
- co-organizing software craftsmanship groups:
  - Ruhrgebiet + Düsseldorf
  - coderetreats, coding dojos

 georgberky

 georg.berky@valtech-mobility.com



# my journey to Groovy



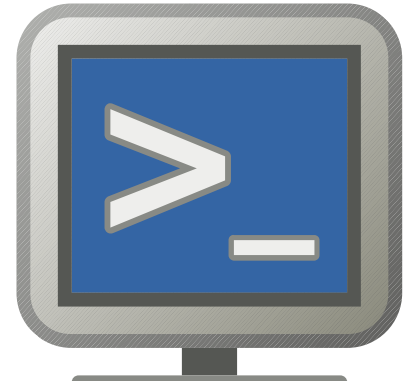
- problem in production
- when contacting a remote system via HTTP
- not enough logging
- but access to the machine
- everything works in curl

# my journey to Groovy

- problem has to be inside the JVM
- google: “run Java as a script”
- Groovy!
- some tinkering, some dependencies
- ⇒ missing certificate in Java’s keystore :)

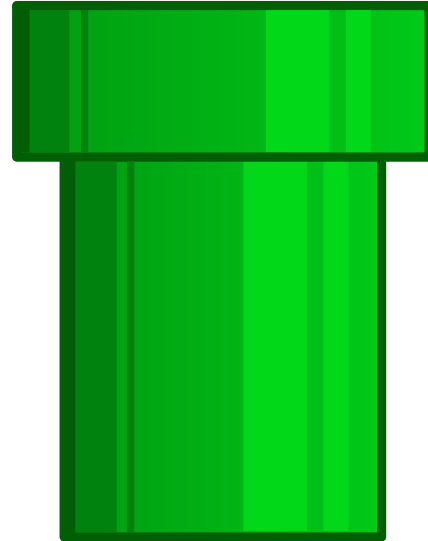
# today's agenda

- → live coding
- shebang + groovyserver
- pipes
- dependency management
- parameter parsing
- external commands
- XML/JSON
- example: scripting with git and HTTP API
- bonus: FIFOs
- bonus: logging



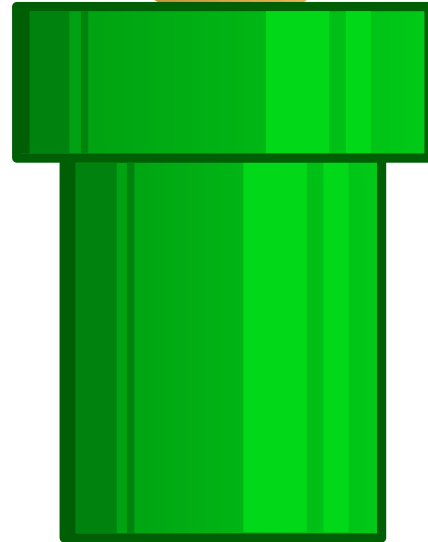
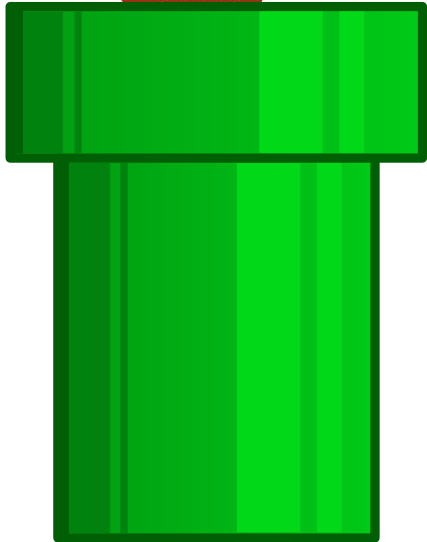
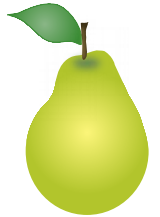
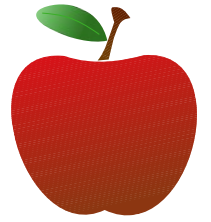
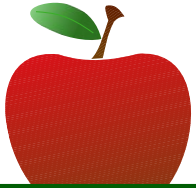
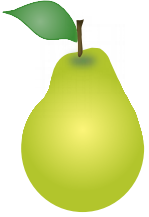
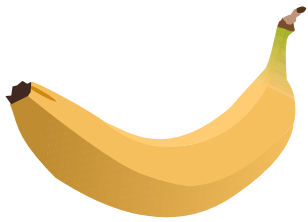
# pipes

- connect two processes
- live outside your root file system
- lifetime: as long as the process creating it
- writing end + reading end
- ends support UNIX file interface



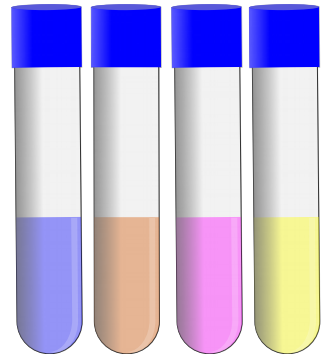
# protocol for pipes

- one line is one message
- line buffered
- last producer disconnects: EOF
- preserve order



# testing

- batteries included: JUnit 3 – 5
- power assert
- test doubles: closure + map coercion
- ~~MockFor, StubFor~~  
only in dynamic context  
scripts: compiles







# dependency management

- Groovy Grape
- built on top of Apache Ivy
- grabs (transitive) dependencies
- default: from Maven Central
- adds them to classpath at runtime

# dependency management

- `@Grab('org.apache.commons:commons-lang3:3.8.1')`
- `@Grab(group='org.apache.commons',  
module='commons-lang3',  
version='3.8.1')`
- **at runtime:** `Grape.grab( ... )`
- **groovysh:** `:grab '...'`
- **CLI:** `grape install <group> <artifact> <version>`

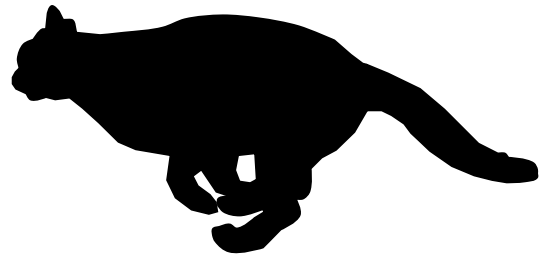
# command line options

- Groovy Commons CliBuilder
- `@Grab('org.codehaus.groovy:groovy-cli-commons:2.5.5')`
- `import groovy.cli.commons.CliBuilder`
- supports short and long options
- now supports typed arguments



# external commands

- Groovy JDK extension
- method `execute()` on `String`
- e.g. `"ls -l".execute()`
- creates and runs instance of `Process`
- method `getText()` on `Process`
- method `eachLine()` on `String`



# XML/JSON

- XmlSlurper/JsonSlurper
- nodes become properties
- XML attributes accessible via `parent.@attributeName`
- `finding`, `looping` etc.



# further reading

- Georg Berky: “Groovy Shell Scripting” (Never Code Alone Blog)
  - <http://tinyurl.com/groovy-shell-1>
  - <http://tinyurl.com/groovy-shell-2>
  - <http://tinyurl.com/groovy-shell-3>
- Hubert Klein-Ikkink (mrhaki): “Groovy Goodness”
  - <https://mrhaki.blogspot.com/>
- Groovy Documentation:
  - <http://groovy-lang.org/documentation.html>



# thank you!



 georgberky

 georg.berky@valtech-mobility.com