SBT

The Past - The Present - The Future

Felix Amerbauer
@felixamerbauer
https://github.com/felixamerbauer/vienna-scala-user-group-sbt

Agenda

- History of build tools
- SBT Basics
- SBT Technology Previews
- SBT 1.0
- Survey

Build Tools History - Shell Scripts

```
#!/bin/bash
while [ $# -gt 0 ] ; do
  case $1 in
  -h*|--h*)cat<<EOMSG
Usage: $(basename $0) file_to_compile
  Compiles and runs a go program found in a single file
EOMSG
      exit 0;;
  *) file=$1; shift ;;
  esac
done
if [ -z "$file" ] ; then
  echo "You must specify a file to compile"
  exit 1
fi
case $file in
  *.go) echo "OK with file $file";;
  *) echo "File must be named _something_.go"; exit 1;;
esac
base=$(basename $file)
echo "Compiling $file"
6g $file
echo "Linking $base.6"
6l $base.6
echo "Running 6.out"
./6.out
echo "Done running"
```

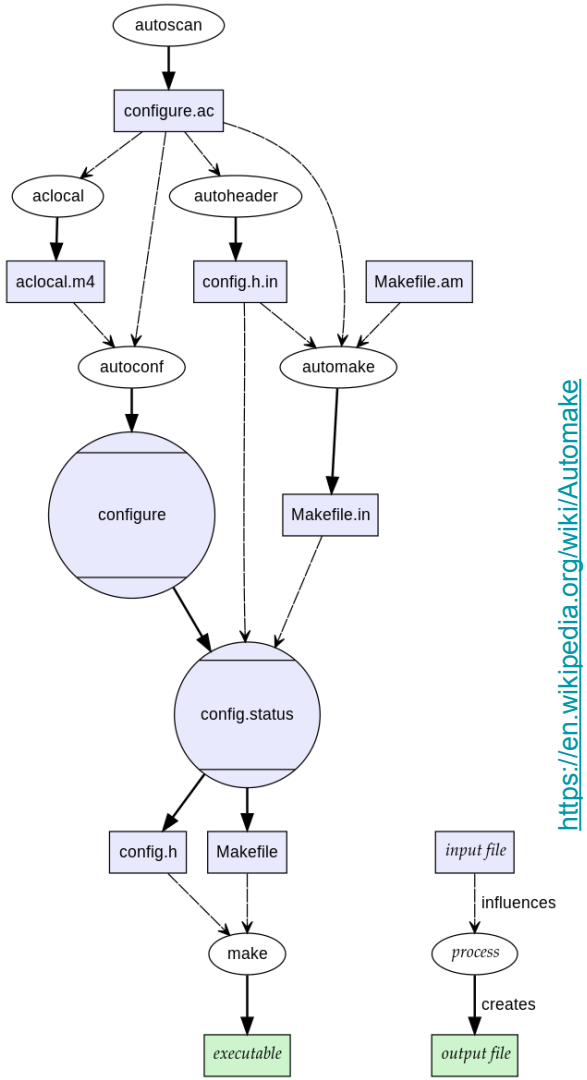
https://www.daniweb.com/software-development/shell-scripting/threads/340254/shell-script-to-build-files

Build Tools History - make (1976)

```
CC = cc
LD = ld

prog: foo.o bar.o
	$(LD) -o prog foo.o bar.o

foo.o: foo.c
	$(CC) -c foo.c

bar.o: bar.c
	$(CC) -c bar.c
```

https://de.wikipedia.org/wiki/Make

Build Tools History - Autoconf/Automake (1991/1996)



https://en.wikipedia.org/wiki/Automake

Build Tools History - 2000 -


```xml
<?xml version="1.0"?>
<project name="Hello" default="compile">
    <target name="clean" description="remove intermediate files">
        <delete dir="classes"/>
    </target>
    <target name="clobber" depends="clean" description="remove all artifact files">
        <delete file="hello.jar"/>
    </target>
    <target name="compile" description="compile the Java source code to class files">
        <mkdir dir="classes"/>
        <javac srcdir="." destdir="classes"/>
    </target>
    <target name="jar" depends="compile" description="create a Jar file for the application">
        <jar destfile="hello.jar">
            <fileset dir="classes" includes="**/*.class"/>
            <manifest>
                <attribute name="Main-Class" value="HelloProgram"/>
            </manifest>
        </jar>
    </target>
</project>
```

Build Tools History - 2002 - Maven™

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>Maven Quick Start Archetype</name>
  <url>http://maven.apache.org</url>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.8.2</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
```

https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html

Build Tools History - 2007 - 

```
apply plugin: 'java'
apply plugin: 'eclipse'

sourceCompatibility = 1.5
version = '1.0'
jar {
    manifest {
        attributes 'Implementation-Title': 'Gradle
Quickstart',
                   'Implementation-Version':
version
    }
}

repositories {
    mavenCentral()
}
```

```
dependencies {
    compile group: 'commons-collections', name:
'commons-collections', version: '3.2'
    testCompile group: 'junit', name: 'junit',
version: '4.+'
}

test {
    systemProperties 'property': 'value'
}

uploadArchives {
    repositories {
       flatDir {
           dirs 'repos'
       }
    }
}
```

https://docs.gradle.org/current/userguide/tutorial_java_projects.html

Build Tools History - 2008 - SBT

- dependency based programming
- internal DSL
- package/library management
- default build / convention over configuration

- interactivity
- parallel by default
- abstracts over test tools
- Scala goodies
  - incremental compilation
  - cross-compilation
  - cross-publishing

https://docs.gradle.org/current/userguide/tutorial_java_projects.html

Build Tools History - 2015 - Bazel

- formerly known as Blaze
- multi-language
  - C++, Java, Objective-C
  - planned support for Scala, Rust...
- single repository
- incremental builds
- force all dependencies to the same version for all projects
- custom language BUILD

- 2 billion lines of code
- 85 TB
- 35 milion commits
- repository for 25.000 Googlers

http://www.wired.com/2015/09/google-2-billion-lines-codeand-one-place/
http://bazel.io/faq.html

SBT - Basics

- tasks
  - typed input / output
  - can be defined at runtime
- settings
  - typed
  - set when loading
- configurations
  - namespaces
- commands
  - more powerful than tasks
  - can modify state of build
- dependencies
  - resolver
- default build
  - configured tasks
    - compile - package - publish
  - structure on disk
- subprojects
  - structure
  - performance

Demo - Minimal build file

Demo - Custom Tasks

SBT - Testing

- out of the box: Specs / Spec2 / ScalaCheck / ScalaTest
- extensibiltiy: https://github.com/sbt/junit-interface

SBT - Publishing

- local Ivy / Maven
- remote Ivy / Maven
- 3rd party plugins for Fat JAR, Docker...

SBT - Extending SBT

- Tasks
- Plugins
- integrate 3rd party libraries
- use SBT libraries
    - IO, user input

SBT - 3rd party plugins

http://www.scala-sbt.org/0.13/docs/Community-Plugins.html

- IDE
- Testing
- Code coverage
- Static code analysis
- One JAR
- Release
- (Cloud)-Deployment
- Monitoring integration
- Web / Frontend
- Documentation
- Dependencies
- Build interoperability
- Utility and system
- Database
- Code generator
- (Games)
- Android
- iOS
- OSGi
- (Plugin bundles)

SBT - 0.13.5+ Technology Previews

- eviction warnings
- enable disable plugins
- name hashing
- cached resolution (minigraph)
- Natural whitespace handling
- none/multiple main class detection
- Sane way to support bytecode manipulation
- syntax improvements

```
lazy val foo = project settings sharedSettings
```

- Sequential tasks
- project-level dependency exclusion
- Cross-version support for Scala sources

```
src/main/scala-2.11/
```

- sbt scala version 2.10.5 for 0.13.9

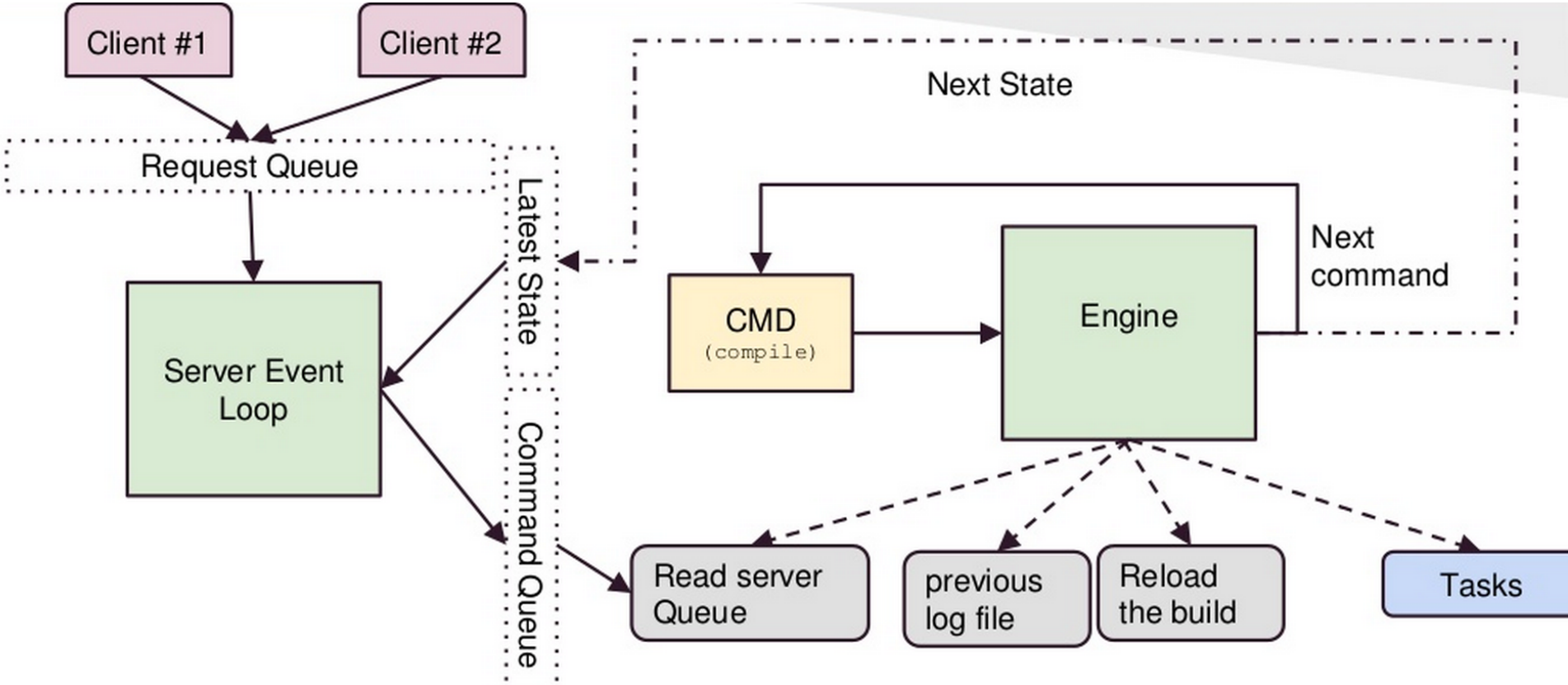
http://www.scala-sbt.org/0.13/docs/sbt-0.13-Tech-Previews.html

SBT - 1.0

- build.scala deprecated (https://goo.gl/SuozmQ)
- stability
  - changes 0.7 -> 0.10
  - binary compatibility (plugins for 1.0.0 work 1.x.y)
  - source compatibility (build.sbt)
- modularization
  - many candidates
  - https://github.com/sbt/io
  - https://github.com/sbt/librarymanagement
- server - idea
  - central build management
  - clients like IDE,terminal
- server - challenges
  - discovery
  - disconnects
  - crashes
  - migrating existing plugins
    - https://github.com/sbt/sbt-core-next
  - protocol
    - https://github.com/sbt/serialization
  - user input
  - blocking commands
  - watchdog

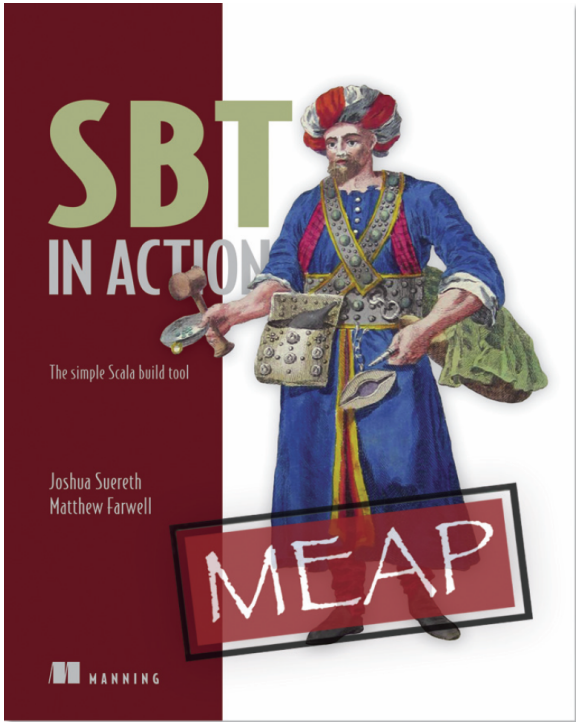
http://de.slideshare.net/EugeneYokota/road-to-sbt-10-paved-with-server

SBT - 1.0 - Architecture



http://de.slideshare.net/EugeneYokota/road-to-sbt-10-paved-with-server

Resources



- https://www.manning.com/books/sbt-in-action
- http://www.scala-sbt.org/documentation.html
- http://stackoverflow.com/search?tab=newest&q=%23sbt

- https://groups.google.com/forum/#!forum/sbt-dev
- https://github.com/sbt/sbt
- https://github.com/jsuereth
- https://github.com/eed3si9n
- https://gitter.im/sbt/sbt

- https://www.parleys.com/tutorial/road-sbt-1-0-paved-server-1
- http://de.slideshare.net/EugeneYokota/road-to-sbt-10-paved-with-server

Survey - Your Build Tool(s) For Scala?

- SBT (?/22)
- Maven (?/22)
- ...
- IDE - Eclipse (?/22)
- IDE - IDEA (?/22)
- None (?/22)

Survey - Your Favorite SBT plugins?

- sbt-assembly
- ...

Survey - Not Yet Started SBT Projects

- Felix: DB -> Slick Mapping -> TypeScript Classes
- Felix: Optimize Fat JARs with ProGuard
- ...