

Kotlin für den Buildmaster

@dirk.dittert



Agenda

- **Kotlin in fünf Minuten**
- **Kotlin DSLs für TeamCity**
- **Gradle 3.5 mit Build Skripten in Kotlin**
- **Kobalt – Buildskripte mit 100% Kotlin**

Was ist Kotlin?

- Eine statisch typisierte Programmiersprache. Programme können in JVM Bytecode und Javascript übersetzt werden
- Ziele der Sprache:
 - Spaß bei der Entwicklung
 - Flache Lernkurve
 - Nahtlose Integration mit Java
 - Maximale Unterstützung durch die IDE
- Pragmatische Lösungsansätze
- Vater der Sprache ist Andrey Breslav (JetBrains)

<http://try.kotlinlang.org>

The screenshot shows the try.kotlinlang.org web IDE interface. At the top, there are social media icons for Google+, GitHub, and JetBrains (JB). To the right, there are utility buttons: 'Shortcuts', 'Convert from Java', and 'Fullscreen'. The main editor area is titled 'Simplest version.kt' and contains the following Kotlin code:

```
1 /**
2  * We declare a package-level function main which returns Unit and takes
3  * an Array of strings as a parameter. Note that semicolons are optional.
4  */
5
6 fun main(args: Array<String>) {
7     println("Hello, world!")
8 }
```

The left sidebar shows a navigation menu with 'Examples' expanded to 'Hello, world!' and 'Simplest version.kt' selected. Below the code editor, a status bar indicates 'Compilation completed successfully' and 'On-the-fly type checking' is disabled. At the bottom, there are tabs for 'Problems view', 'Console', and 'Generated classfiles', and a footer note: 'This demo is running on Kotlin v1.0.1-2'.

Hello World!

```
package tfo
```

```
data class KotlinEvent(var name: String = "JUG", var ort: String = "Nürnberg")
```

```
fun main(args: Array<String>) {
```

```
    val event = KotlinEvent("Javaland", "Brühl")
```

```
    println("Herzlich willkommen ${event.name} in ${event.ort}!")
```

```
}
```

```
> Herzlich willkommen Javaland in Brühl!  
>  
> Process finished with exit code 0
```

TeamCity Konfigurationen als Kotlin DSL

- Kotlin eignet sich sehr gut für DSLs
 - statisch typisiert → Codevervollständigung, Gehe zu Deklaration, Typen
 - Extension Functions → Erweiterung beliebiger Typen
 - Annotationen → Unterstützung der Codevervollständigung der IDE
- TeamCity (Continuous Integration System von JetBrains) kann Konfigurationen als Kotlin Code aus einem Repository laden
- Ermöglicht dynamische Generierung, Unit Tests, Wiederverwendung

DEMO

Kotlin in Gradle

- Entwickler wünschen sich Code Vervollständigung für die Gradle DSL
- Unterstützung von Kotlin wurde im Mai 2016 angekündigt
- Aktuell: Gradle 3.5 RC 1 mit `gradle-script-kotlin` in Version 0.8.0
- Infos: <https://github.com/gradle/gradle-script-kotlin/releases>

DEMO

Fazit zu Kotlin in Gradle

- Debugging funktioniert immer noch nicht!
„Accelerate developer productivity“ → ohne Debugger schwer...
- Momentan ist die Dokumentation spärlich (Beta!), aber Beispiele helfen bei der Einarbeitung deutlich
- Dritter Ansatz (Groovy, Java API, Kotlin) mit neuen Besonderheiten?
- Produktiver Einsatz? Warten...

Kobalt

- Kobalt ist ein Build System, das durch Gradle und Maven inspiriert wurde und vollständig in Kotlin implementiert ist
- Autor: Cédric Beust (manchen bekannt als Autor von TestNG)
- Informationen: <http://beust.com/kobalt/home/index.html>
Source: <https://github.com/cbeust/kobalt/>

DEMO

Fazit zu Kobalt

- Kobalt ist nicht nur Buildtool, sondern auch ein Framework zu Programmierung von Buildskripte
- Viele pragmatische Designentscheidungen (z.B. können Default Repositories global konfiguriert werden)
- Neue Projekte können über Templates erstellt werden
- Einfaches aber für viele Probleme angemessenes Plugin-Konzept
- IDE Plugin für IntelliJ (Funktionsumfang momentan noch eingeschränkt und auch etwas buggy)
- Aktive Entwicklung und extrem schnelles und detaillierte Hilfestellung unter <https://kotlinlang.slack.com/> (Channel #kobalt)



Vielen Dank für Ihre Aufmerksamkeit!