

Java SE 8

Einführung

Java: wesentliche Merkmale

- Ausführung von *Architektur-unabhängigem Bytecode* in einer *Virtuellen Maschine (JVM)*
- *Plattformunabhängigkeit*
- *Objektorientierung*
- Unterstützung für andere (auch dynamische!) *Sprachen in der Java Virtual Machine*

Java: Design-Ziele

- eine *einfache, objektorientierte, verteilte* und vertraute Programmiersprache
- *robust* und *sicher*
- *architekturneutral* und *portabel*
- leistungsfähig
- interpretierbar, parallelisierbar und dynamisch

Java: Design-Ziele

„einfach“

- Gegenüber C, C++ oder C# reduzierter Sprachumfang
- Aus der Erfahrung heraus fehleranfällige oder schwierige Sprachkonstrukte entfallen
- Keine Operatorüberladung
- Keine Mehrfachvererbung
- Keine explizite Speicherverwaltung

Java: Design-Ziele

„objektorientiert“

- Java ist *strikt* objektorientiert, d.h. es existiert kein Code außerhalb eines Objekts
- Objekte werden generell mit Klassen beschrieben, bevor sie erzeugt werden können
- Eine Klasse stellt sozusagen den Bauplan für Objekte mit identischer Struktur und identischem Verhalten dar.

Java: Design-Ziele

„verteilt“

- Unterstützung für Netzwurkkommunikation
- Verteilte/entfernte Methodenaufrufe, d.h. ein Objekt kann Methoden eines anderen Objekts auf einem über das Netzwerk erreichbaren Rechner aufrufen
- Unterstützung auch für Kommunikation auf höherem Abstraktionsniveau: Webservices, REST-Services u.a.

Java: Design-Ziele

„vertraut“

- Syntax ist an C angelehnt
- Klassenbibliotheken orientieren sich an Smalltalk
- Entwurfsmuster („GoF Design Patterns“ von E. Gamma et. al.) werden aufgegriffen

Java: Design-Ziele

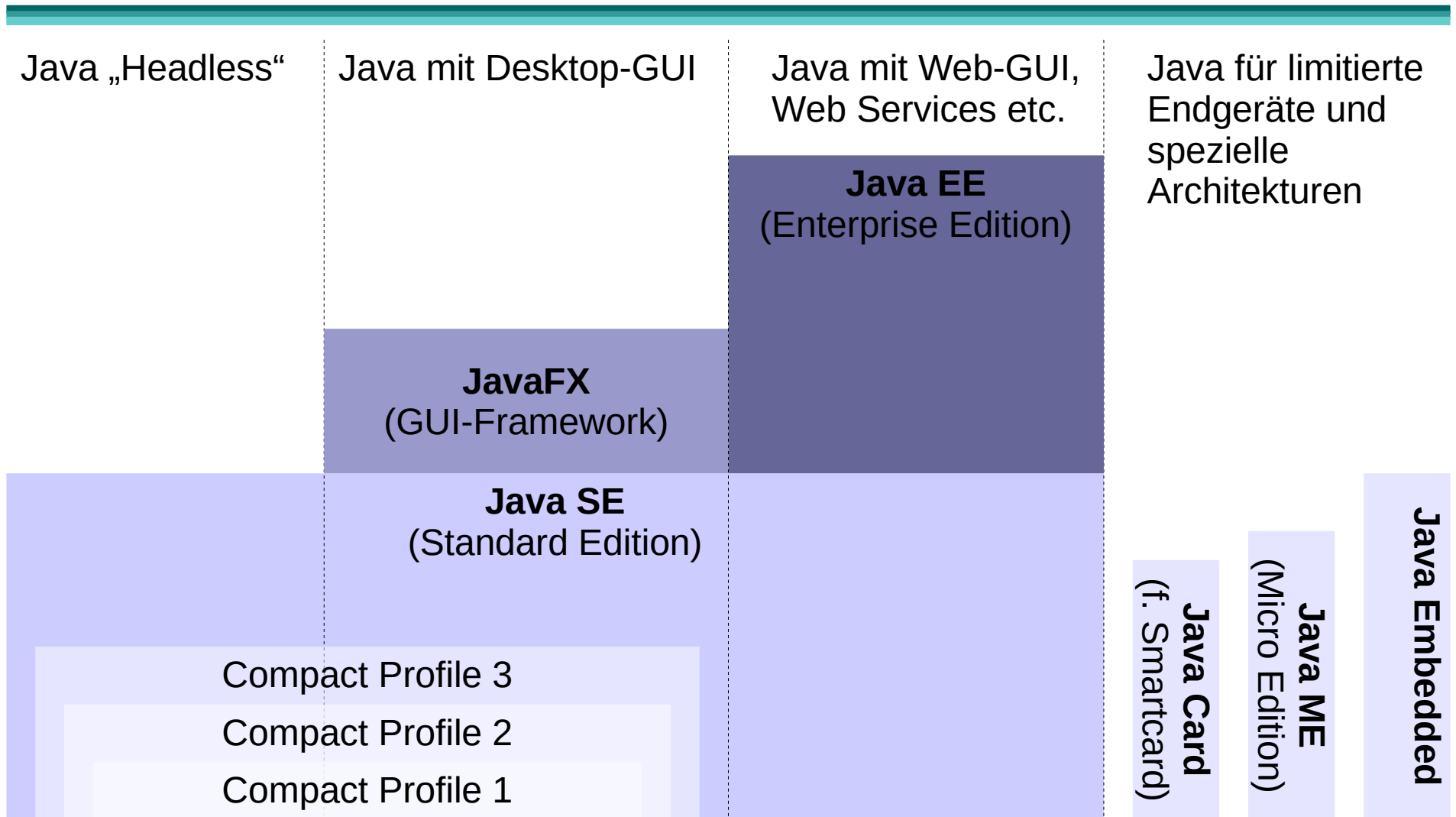
„robust“

- Wahrscheinlichkeit versehentlicher Fehler soll reduziert werden
- Strikte und starke Typisierung
- Automatische Speicherbereinigung („Garbage Collection“)
- Keine Zeigerarithmetik
- Keine Operator-Überladung

Java-Plattformen

- Java SE – Standard-Edition
- Java EE – Java Enterprise Edition
- Java ME – Java Micro Edition
- Java SE/ME Embedded
- Java Card – Java auf Smart Cards
- Java SE für ARM – inzwischen Standard
- Java SE für iOS/Android/Windows Mobile?
→ <http://openjdk.java.net/projects/mobile/>

Java-Plattformen



Java installieren

■ JRE – Java Runtime Environment

- Java Virtual Machine
- Java-Laufzeit-Bibliotheken

■ JDK – Java Development Kit

- Enthält ein komplettes JRE
- ... und die Entwicklungstools (Compiler, Debugger, ...)

Die nächsten Schritte

■ „Hands on!“

- Entwicklungsumgebung
- Code, Übersetzung und Ausführung
- Dokumentationserstellung mit javadoc

■ Grundlegendes zur Syntax

■ Elementare Bibliotheken

Hinweise zur Syntax

I

- Java ist *case sensitive*.
- Anweisungen müssen *immer* mit Semikolon terminiert werden.
- Standard-Encoding für Quellcode ist UTF-8.

... und außerdem:

- Variablen-Namen beginnen mit Buchstaben, niemals mit `_` oder `$` (Konvention).

Hinweise zur Syntax

II

- Für Bezeichner-Namen ist *camel case-Notation* üblich.
- Klassen-Namen dürfen zusätzlich nicht mit Zahlen beginnen.
- Package-Namen orientieren sich an der *revers domain name*-Konvention.
- Eigene Package-Namen beginnen niemals mit *java*, *javax* oder *sun*.

Entwicklungsumgebungen

- NetBeans IDE
 - Eclipse
 - IntelliJ IDEA
 - Oracle JDeveloper
 - ...
-

Quellen

- <http://www.oracle.com/technetwork/java/>
 - <http://docs.oracle.com/javase/tutorial/>
 - <http://openbook.rheinwerk-verlag.de/javainsel/>
 - <http://openbook.rheinwerk-verlag.de/java8/>
 - Inden, M.: „Der Weg zum Java-Profi“. dpunkt.verlag 2017, 4. Aufl.
-