

**In 80 Tagen um die Welt
oder
wie Scala mich
veränderte**

Andreas Boss



Jörg Bächtiger

Finnova

consistent	everything is an object, everything can be nested
flexible	every operator is a method call (and vice versa)
scalable	libraries can implement new language constructs
type-safe	fully statically typed at compile-time
light-weight	usable like a dynamically-typed scripting language
expressive	lots of semantics expressable with very less syntax
concise	mostly free of „syntactic sugar“ in syntax
simple(r)	no special syntax for arrays, no specialized "switch", etc
object-oriented	unified object system, no primitive types, no statics
functional	higher-order functions, pattern matching, etc
interoperable	call Java from Scala and with care also vice versa
scriptable	supports on-the-fly compilation and Unix she-bang
sophisticated	very elegant and sophisticated collection library design
extensive	leverages from existing Java ecosystem of libraries

Fun

Learn new stuff

Become smarter

Motivation

Quality

Attract new employees

Push others

Workshop (in-house)

Conference videos

Coursera

How we started

Tooling

Workshop (in-house)

Conference videos

Coursera

How we started

Get coached

Tooling

Training course

Write productive code

Understandability

Freedom of choice

Laggards

SBT

Management

Challenges

Embrace functional paradigm

Mocking

To become competent you must feel bad

Hubert Dreyfus

Quality

Learning curve

**It is a process and
takes time**

Interoperable

Experiences

Less code

Java-Style Scala

Concise

Freedom

Elegant

Scale in-house

Further training

Use in new projects

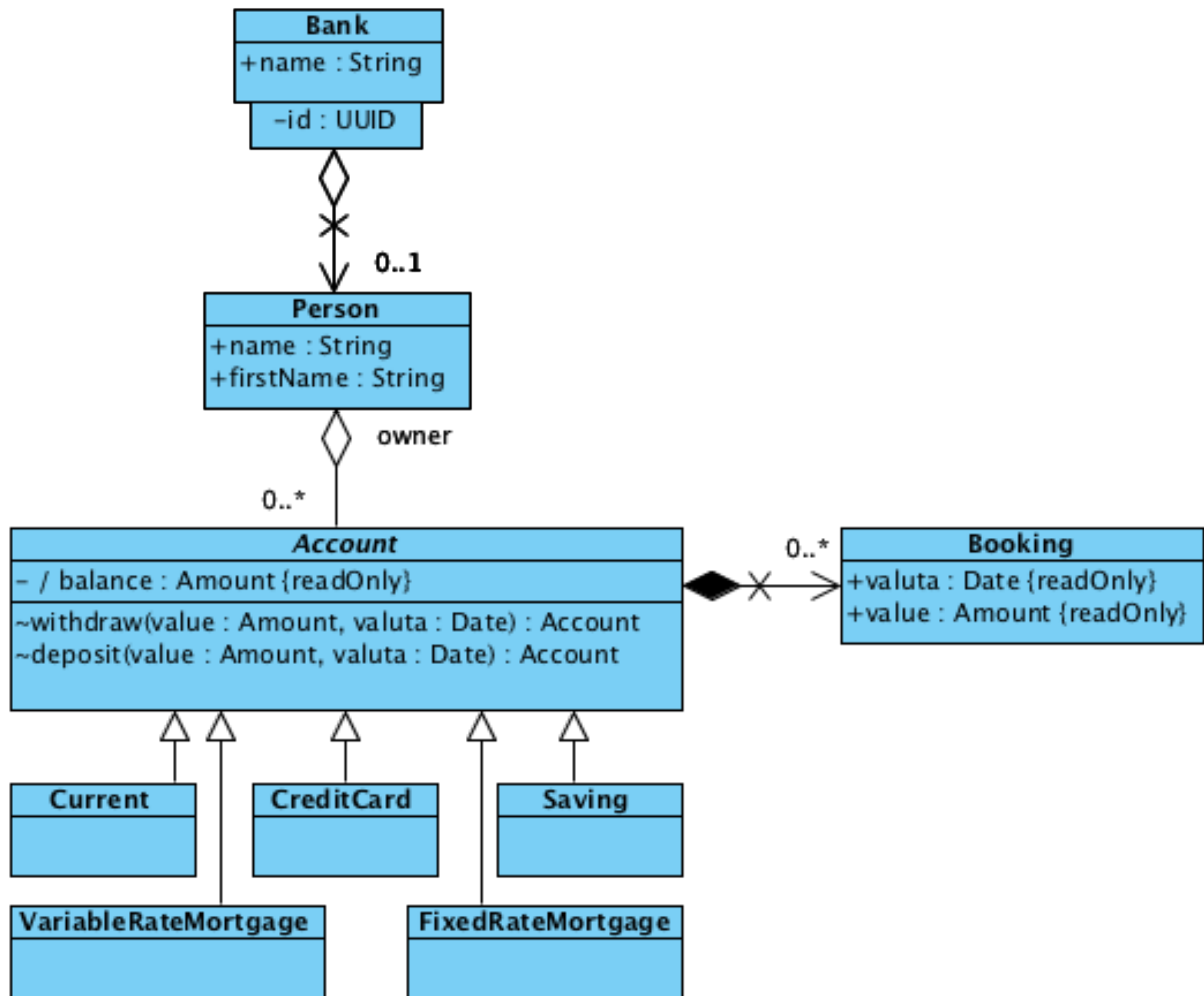
Prospects

Enabling people

Use more scala features

New technologies

Live coding



Value semantic

Single writer principle

Complexity reduction

Better performance

Immutability

Failure atomicity

Cacheable

Inherently thread-safe

**Thank you
for your attention**

Questions?