

Working with Implicits



Harit Himanshu

@harittweets



Overview



Understanding how **Implicit Conversion** works

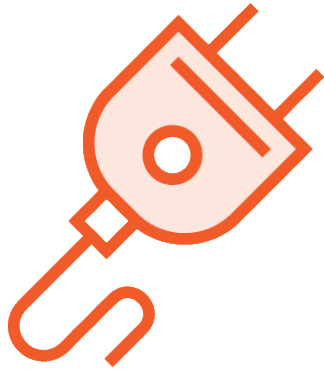
Understanding the Rules for **Implicit Conversion**

Working with the Application of **Implicits**

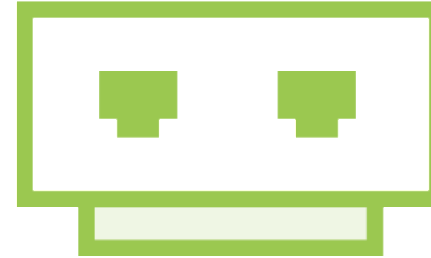
Project Demo



Use of Implicits



Type Available



Type Requested



Rules for Implicit Conversion

Marking Rule

Scope Rule

One-at-a-time Rule

Explicit-First Rule



Marking Rule

Only definitions marked *implicit* are available



Scope Rule

An inserted **implicit conversion** must be in scope as a single identifier, or be associated with the source or target type of the conversion



One-at-a-time Rule

Only one **implicit** is inserted



What's One-at-a-time?

`convertToType1(i) + j`



`convertToType2(convertToType1(i)) + j`



Explicit-First Rule

Whenever code type checks as it is written, **no implicits** are attempted



Where Are Implicitly Applied?

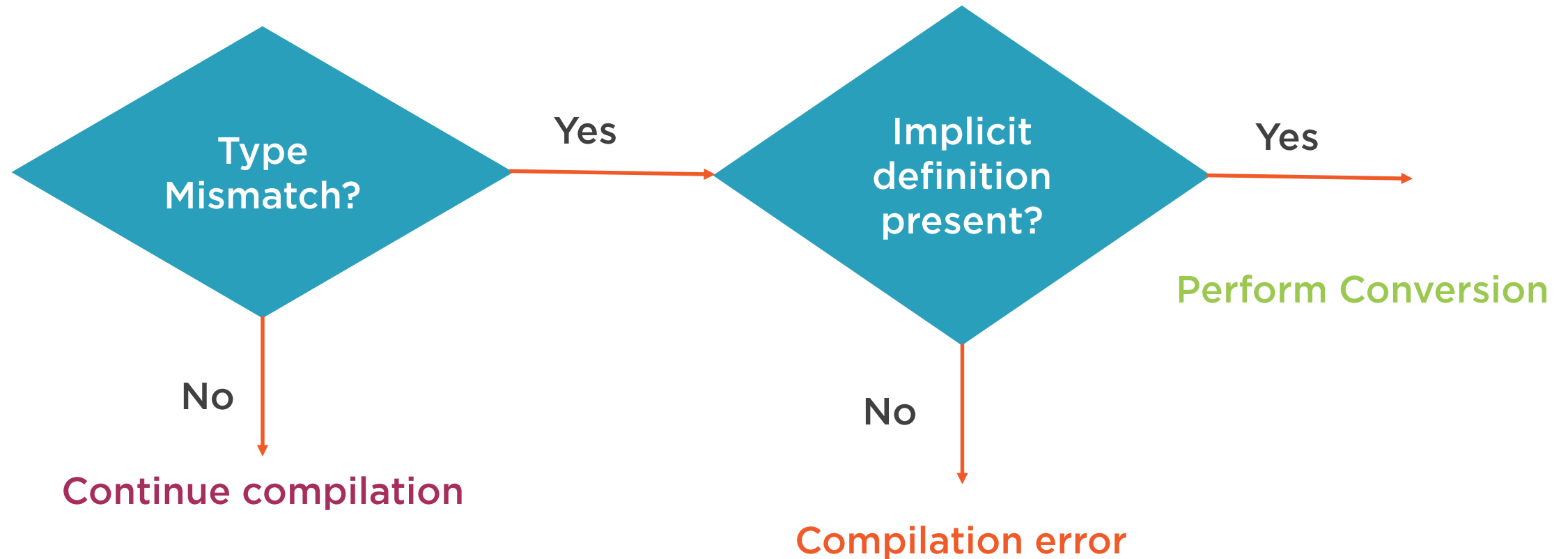
**Converting to
Expected Type**

**Converting the
Receiver**

**Implicit
Parameters**



Converting to an Expected Type



Converting the Receiver

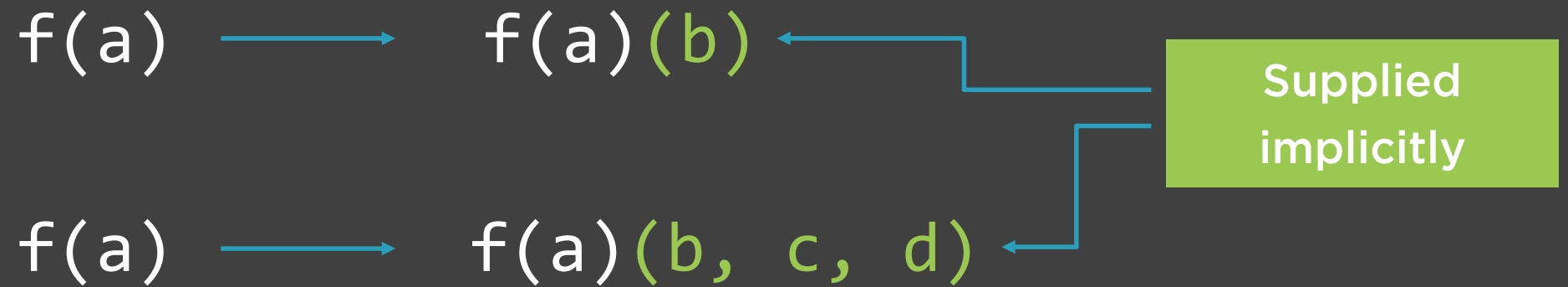
customerBalance.pretty



receiver



Implicit Parameters



How Implicit Parameters Work?

```
implicit val b: B = 10  
implicit val c: C = "Hello"  
implicit val d: D = true
```

```
def myFunction(a: A)(implicit b: B, c: C, d: D) = ???
```



Demo



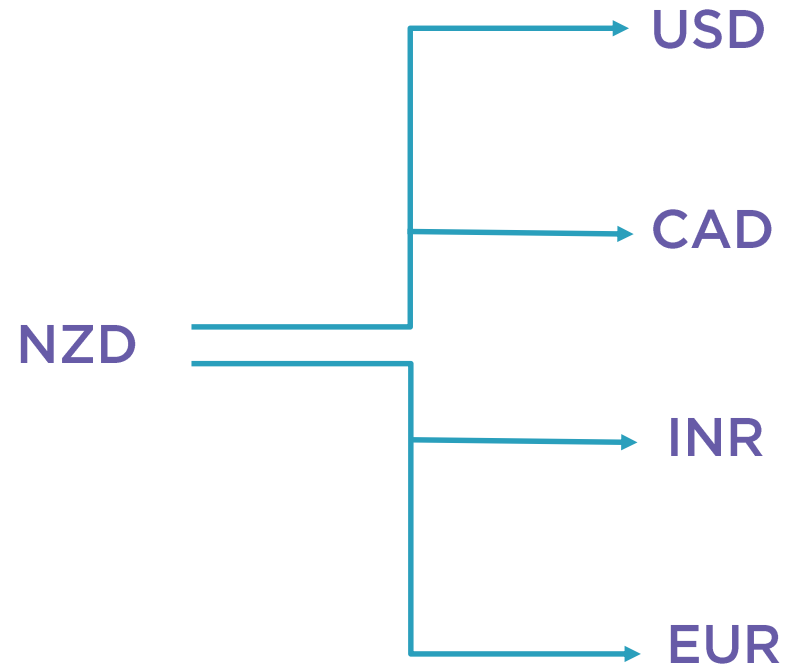
Extending the **AccountService** Trait

Adding a new **Currency** class

Refactoring **BankofScala** entrypoint



Buying Foreign Currency



Summary



Understanding how **Implicit Conversion** works

Understanding the Rules for **Implicit Conversion**

Working with the Application of **Implicits**

Project Demo

