

蒋航的博客

Scala编程爱好者♥

scala的反射

如果当初Scala没有选择基于Jvm，那么Scala是否还能取得今天的成就吗？Jvm为Scala带了稳健强大的性能，同时也无法避免类型擦除的约束。

作为Jvm上的先进语言，Scala在生成字节码时，编译器附加了额外的类型信息，及时class的泛型参数被擦除了，scala仍然可以获取泛型信息。

主要存在三种api:

- `TypeTag`，可获取一个类型的全部信息，包括高阶类型，比如 `List[List[List[String]]]` 类型。
- `ClassTag`，可获取类型的部分信息。 `ClassTag[List[List[String]]]`，仅可得到类型擦除后的类的类型，也就是 `scala.collection.immutable.List`。
- `WeakTypeTag` 是可以用于获取抽象类型，比如 `def foo[A]=List.empty[A]`，想要获取这个抽象类型 `A`，需要使用 `WeakTypeTag`。

```
→ scala
Welcome to Scala 2.12.2 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_141).
Type in expressions for evaluation. Or try :help.

scala> import scala.reflect.runtime.universe._
import scala.reflect.runtime.universe._

scala> val tt=typeTag[List[List[String]]]
tt: reflect.runtime.universe.TypeTag[List[List[String]]] =
TypeTag[scala.List[scala.List[String]]]

scala> import scala.reflect._
import scala.reflect._

scala> val ct=classTag[Map[String,Int]]
ct: scala.reflect.ClassTag[Map[String,Int]] = scala.collection.immutable.Map

scala> ct.runtimeClass
res1: Class[_] = interface scala.collection.immutable.Map
```

利用编译器获取类型信息

之前说过，scala编译器在编译器全部保存了相关的类型信息，仅仅需要借助隐式参数由编译器传入即可。

```
import scala.reflect.runtime.universe._

def typeInfo[T](x: T)(implicit tag: TypeTag[T]) = tag.tpe match {
  case TypeRef(preType: Type, symbol: Symbol, typeParams: List[Type]) =>
    println(s"preType.typeSymbol : ${preType.typeSymbol}")
    println(s"preType : ${preType}")
    println(symbol.fullName)
    println(typeParams)
}

typeInfo(Map("1"->1))
//打印如下::
preType.typeSymbol : package immutable
preType : scala.collection.immutable.type
scala.collection.immutable.Map
List(java.lang.String, Int)
```

在以上程序中，`preType` 是x对象的前缀路径的类型，这里的 `scala.collection.immutable` 是一个 package。`symbol` 就是参数x的符号信息，`typeParams` 是x对应的类型的类型参数，对于 `Map[String,Int]`，那么其类型参数有 `java.lang.String` 和 `Int`。

公告

昵称：蒋航
园龄：9个月
粉丝：0
关注：4
+加关注

2018年9月						
日	一	二	三	四	五	六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

最新随笔

1. Akka中的Message Stash
2. scala的反射
3. Pipeline in scala——给scala添加管道操作
4. Akka Cluster简介与基本环境搭建
5. Akka Serialization
6. akka-stream与actor系统集成以及如何处理随之而来的背压问题
7. AVL树的单双旋转操作
8. Run a task only once in (akka) cluster
9. scala写算法-List、Stream、以及剑指Offer里部分题目基于scala解法
10. scala写算法-快排

随笔分类

akka(4)
akka-cluster(2)
scala(9)
scalajs(1)
数据结构(5)

一句话解释scala的反射机制:编译时期额外类型信息，此类型信息可以用编码显式给出，也可以用编译器推断。以下的例子可以说明:

```
def getType[T](x: T)(implicit tag: TypeTag[T]): Type = tag.tpe
```

现有一个函数 `getType(x)`，可以此获取某个对象 `x` 的类型信息。显式给出类型为 `Any`，编译器则传入隐式 `tag:TypeTag[Any]` 对象。

```
println(getType[Any](List(1, 2, 3)))
//打印 Type[Any]
println(getType(List(1, 2, 3)))
//打印 Type[List[Int]]
```

Type之间的比较操作

type之间的比较主要有三种类型:

- 比较两个类型是否是继承关系
- 判断两个类型之间的相等性
- 给定某一确定的类型的成员方法或者属性

现在有两个trait，分别是 `A` 和 `B`，且 `A` 是 `B` 的父级。

```
trait A
trait B extends A
```

```
val aType = typeOf[A]
val bType = typeOf[B]
println(aType == bType) //false
println(aType <:= bType) //false
println(bType <:= aType) //true
println(aType <:= aType) //true
```

对于给定的两个 `Type` 实例，`<:=` 方法可以用于比较两个类型是否具有父子类型关系，作用与Java中的 `isInstance` 类似。

当使用 `==` 比较两个类型时，类型别名与实际类型被认为是不同类型，而 `:=` 会用最根本的类型去比较。

```
type PersonName=String
val t1= typeOf[PersonName]
val t2= typeOf[String]
t1 == t2 //false
t1 := t2 //true
```

`TypeTag` 可以等效的看作为scala2.10版本以前的 `Manifest`，`ClassTag` 也可以等效的看作为scala2.10版本之前的 `ClassManifest`。

知难行易

原创博文，[请勿转载](#)

[我的又一个博客hangscer.win](#)

分类: scala

好文要顶 关注我 收藏该文





蒋航
关注 - 4
粉丝 - 0

+加关注

« 上一篇：[Pipeline in scala——给scala添加管道操作](#)
» 下一篇：[Akka中的Message Stash](#)

posted @ 2018-02-23 21:08 蒋航 阅读(534) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

- 【推荐】超50万VC++源码: 大型组态工控、电力仿真CAD与GIS源码库！
- 【免费】要想入门学习Linux系统技术，你应该先选择一本适合自己的书籍
- 【前端】SpreadJS表格控件，可嵌入应用开发的在线Excel
- 【直播】如何快速接入微信支付功能

随笔档案

2018年2月 (2)
2017年12月 (12)

阅读排行榜

1. scala-Future和Promise(610)
2. scala的反射(534)
3. Akka Cluster简介与基本环境搭建(504)
4. scalajs_初体验(449)
5. Akka Serialization(163)



最新IT新闻:

- 外媒：TCL考虑收购ASM太平洋25%股权 价值约10亿美元
 - 瓜子二手车发声明：法院驳回人人车请求禁止瓜子投放广告语申请
 - 微软证实将停止Windows 10 Office Mobile应用的开发
 - 支付宝全球独宠中国锦鲤今日开抽 网友：抽我我不怕疼
 - 前百度音乐总经理王磊确认加盟腾讯音乐 出任总经理
- » 更多新闻...



最新知识库文章:

- 为什么说 Java 程序员必须掌握 Spring Boot ？
 - 在学习中，有一个比掌握知识更重要的能力
 - 如何招到一个靠谱的程序员
 - 一个故事看懂“区块链”
 - 被踢出去的用户
- » 更多知识库文章...