Week 12: Scala

Martin Odersky

Scala's Type System from A Programmer Perspective

In the last three weeks of this course, we study Scala's type system.

Compared to what we have seen before, we will see that we have to extend the type system with a form of type that refers to a value, e.g. $x.T$.

With these *term dependent types*, one obtains better language expressiveness.

This is demonstrated by two papers which we will study in this week.

Next week we study the formalization of the core of Scala's type system.

Papers to Read

Scalable Component Abstractions.

Martin Odersky and Matthias Zenger, submitted for publication.

- This paper gives an overview of some of the less conventional constructs in Scala's type system.
- It shows how these constructs can be used in new techniques for component abstraction and composition.

Indendently Extensible Solutions to the Expression Problem.

Matthias Zenger and Martin Odersky. Proc. 12th ACM Workshop on Foundations of Object-Oriented Languages (FOOL 12), Jan 2005.

- This paper studies a fundamental problem in development of extensible software.
- How can we extend systems at the same time with new data variants and with new operations?