Performance & Optimisation on Scaja.js

Damien Engels, Tristan Overney
Sébastien Doeraene, Martin Odersky (Supervisors)

January 20, 2016

Chapter 1

Introduction

Nowadays, a huge part of our computer utilisation is through a web browser and web apps. Developping such apps is most commonly done in JavaScript as it is pretty much universally supported. Scala.js allows you to develop those kind of programs without the hassle of using JavaScript directly, with the type safety of your regular Scala and still keep a great inter operability with the many existing JavaScript libraries. Since its conception, Scala.js has seen a lot of optimization and rewriting done in the sole purpose to increase its performances and reduce the size of its generated code. In this project, we were tasked to first implement or port a benchmarking framework to Scala.js to have a reliable way to compare two implementations of the same feature and be able to tell which one is the best in a relatively effortless manner. After that we took a deep dive in the Scala.js tools and were tasked to implement several optimisation to the generated javascript code. We first implemented the constant folding for the binary operator *String_+*, described in chapter 3. Then we rewrote the instance checking with a tag system and eventually we worked on the inlining constructor in specific cases; both in order to reduce the amount of method calls.

Chapter 2

Scalameter for ScalaJS

Chapter 3

String_+ constant folding

Chapter 4

Instance test optimisation

Chapter 5

Constructor inline on effectively final classes

Chapter 6

Going further