Web Computing in Clean

Kristóf Harmat, Péter Tandi, Zoltán Horváth
Martin Wierich, Rinus Plasmeijer

Department of General Computer Science, University Eötvös Loránd, Budapest
krizsa@ludens.elte.hu, tandi@ludens.elte.hu, hz@ludens.elte.hu
Faculty of Mathematics and Computer Science, University Nijmegen
martinw@cs.kun.nl, rinus@cs.kun.nl

Abstract

The nature of Web computing is inherently different in many aspects even from the more conventional distributed computing over local area networks. We investigate the applicability of Cardelli's Service Combinators as an extension of the pure functional language Clean in this paper. Our work is based on Clean's recently developed TCP library. The result, a Service Combinator library of Clean provides a tool for interoperation using HTTP protocol[5] between Clean applications and CGI applications residing on Web servers written in any language (for example C, Java or Clean).

1 Introduction

Nowadays we use the Web more often in our work and everyday life. If we would like to access a service or a resource in the Web our "browsing" behavior is different from our behaviour using a service in our local computer. We have to behave differently because the architecture of the Web is different in many aspects from the Neumann computer architecture. The Web is some kind of distributed object system. If somebody wants to design a programming language for this distributed object system, some special problems arise.

The nature of Web computing is inherently different in many aspects even from the more conventional distributed computing over local area networks. An appropriate model of Web computing takes into account the delays caused by long physical distances, lack of referential integrity, the varying bandwidth, the possible failure of nodes, the large scale heterogeneity of the underlying network media and mobility of networking devices [1].

If you think about the URL as a network pointer, it does not always point to the same data and sometimes doesn't point anywhere. So this pointer is different from the "normal" pointers which can be found in the third or fourth generation programming

---