



Development of a Language and Tools for a Distributed Messaging System

Master Project Proposal, WS 2015/2016
Software Architecture Group, Prof. Dr. Robert Hirschfeld

From Wiki-like to Synchronous Collaboration

Lively Kernel is a Smalltalk-like self-supporting development environment in the Web-browser, which allows Web-applications to be developed in a lively way. Since the environment is shared, the feedback cycles among collaborating users are also shortened. Therefore, a Lively Kernel-based wiki allows for development, sharing, and reuse of each other's creations in an environment which can evolve while being used.

To allow for the development of distributed applications and synchronous collaboration, Lively Kernel provides a message passing system called Lively2Lively. This system allows to pass messages directly from one Web page to another.



<http://lively-kernel.org/>

Project Objectives

In Lively2Lively message sending and receiving is implemented as a JavaScript library, which requires writing code in a continuation passing style via promises or callbacks. Recent developments in programming languages like Dart have shown that such code does not need to be written manually, but that it can be generated automatically.



The goal of this project is to design and implement a language extension to JavaScript that supports writing code using asynchronous Lively2Lively messages. Since it will be used in an interactive development environment, tool support for writing and debugging code that uses the messaging system is a core requirement.

Prof. Dr. Robert Hirschfeld, Jens Lincke

Contact

Software Architecture Group, Potsdam

<http://www.hpi.de/swa>

Robert Krahn and Dan Ingalls

SAP Labs, Communications Design Group San Francisco