German Research Center for Artificial Intelligence

A Web Interface for Isabelle: The Next Generation

Christoph Lüth and Martin Ring

Introduction

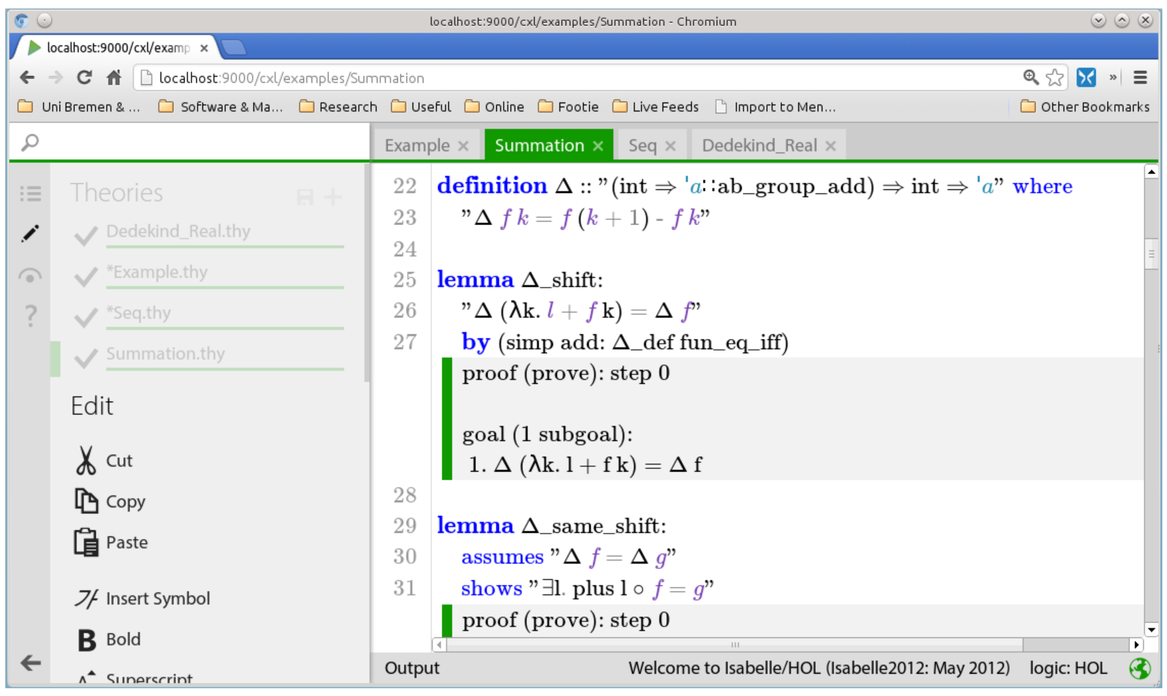
In the past years the role of web development has completely changed. Today, with the finalization of the HTML5 standard in sight, it's possible to create complex, interactive and comfortable desktop-like user interfaces living in any modern browser on a classic computer, a tablet or even a smartphone.

Web interfaces do not use much resources on the client side, are portable and mobile and require no effort to set up and use. There can be no missing requirements.

Breaking new ground

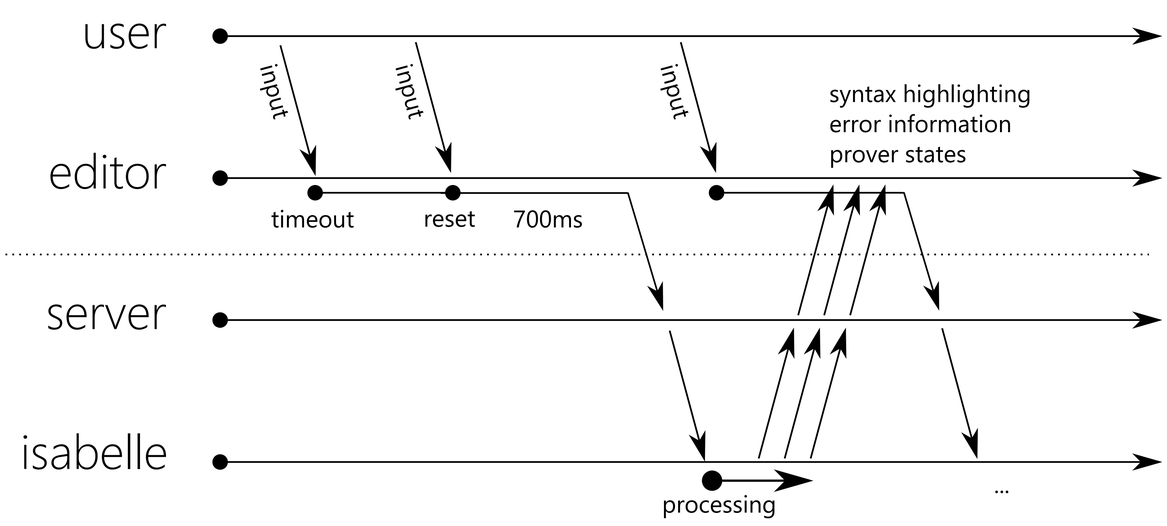
These new possibilities lead to the idea of a next-generation user interface for a contemporary interactive theorem prover.

Our question (and challenge) was: how far can we push that idea of combining two completely different worlds?



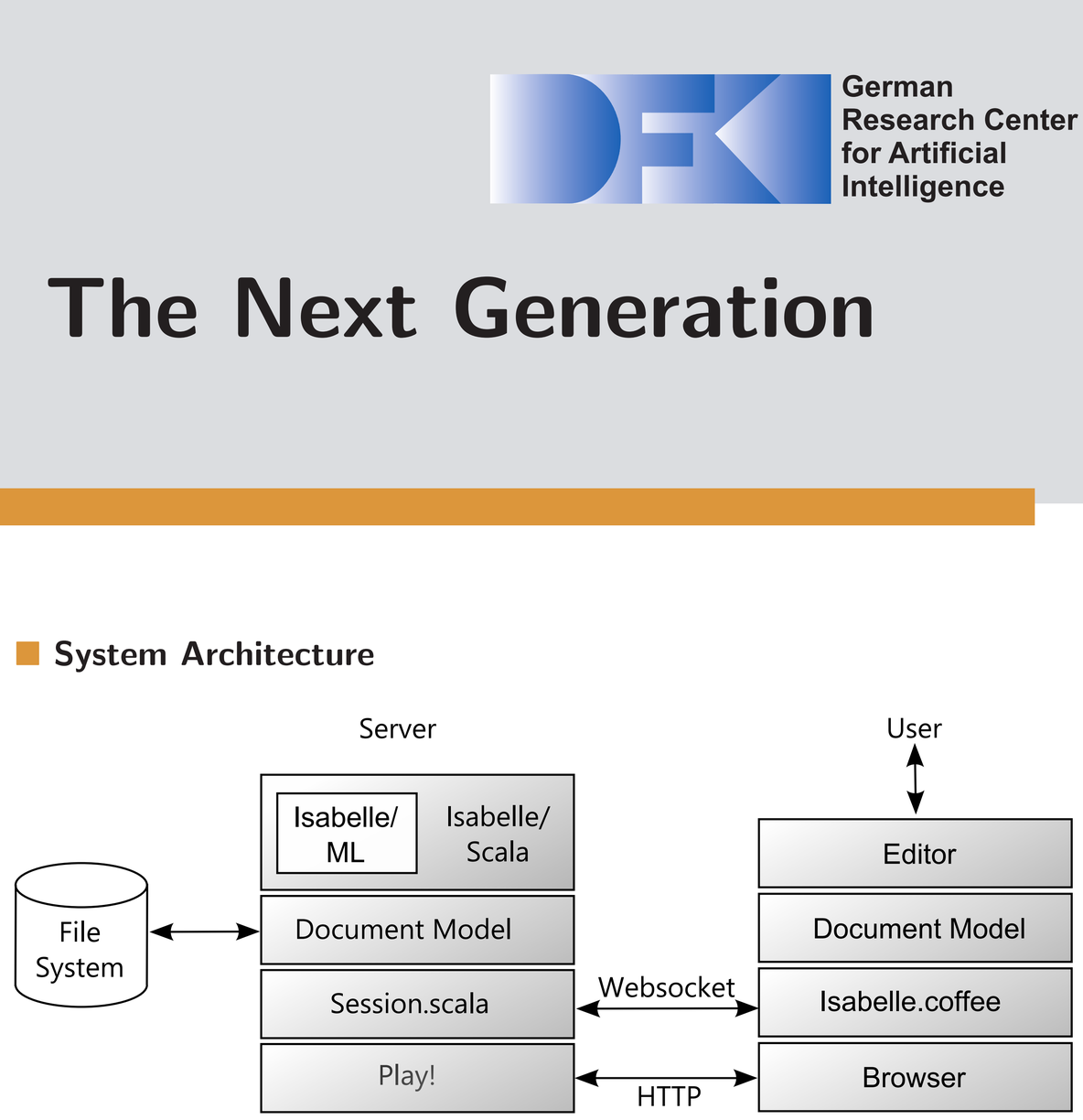
Implementation

The basic system architecture is clear: we need a web server to connect with Isabelle on one side, and with web browsers on the other side. Hence, the questions to address are: Firstly, how to connect Isabelle with a web server, and secondly, how to use a browser to edit Isabelle theory Files?



The single most important design constraint was the asynchronous communication between server and browser

System Architecture



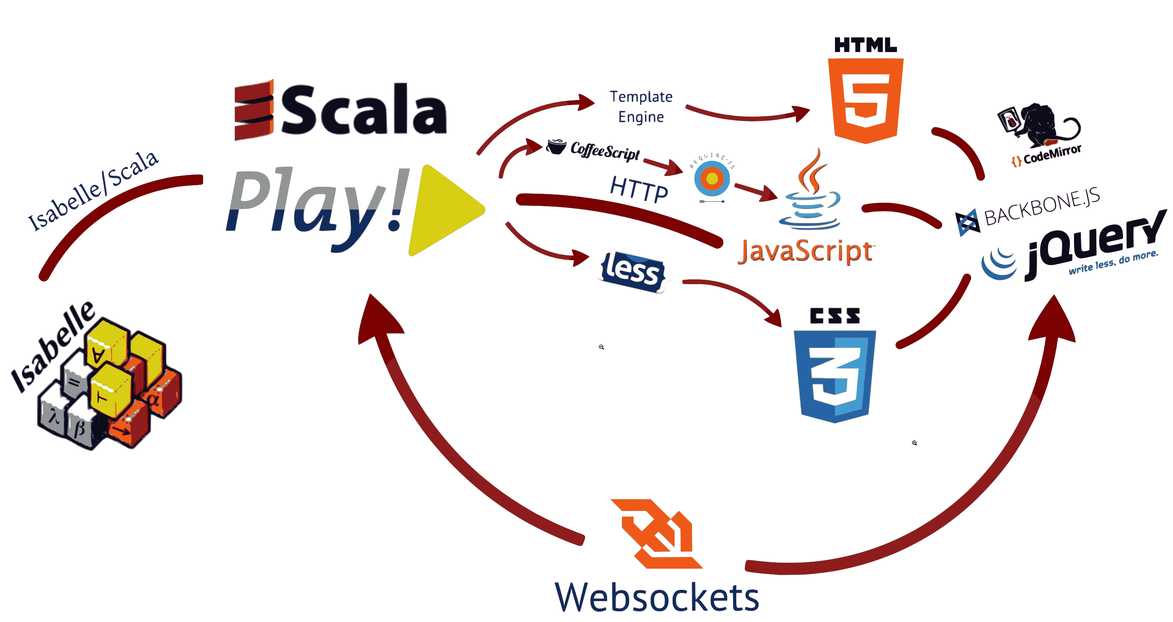
The Editor

The editor needs to be able to display special symbols; perform on-the-fly replacements; use variable-width fonts; allow super- and subscripts; and allow tooltips for text spans. We extended the CodeMirror editor and use the MathJax fonts, which results in a seamless editing environment for mathematical notation on the web.

Communication

The asynchronous document model is implemented by two modules, which run on the server and in the browser respectively. They communicate via WebSockets, using a self-developed thin dynamically typed communication layer, which allows to call JS functions from Scala nearly as if they were native, and vice versa.

A lot of players



Clide ...

- has a much richer user experience than previous web interfaces
- has a better rendering of mathematical notation than other interfaces
- equals them in terms of responsiveness
- is easier to set up and use

Web technology is ready for theorem proving! Try it here:

`http://clide.informatik.uni-bremen.de`

(Use a recent WebKit-based client)

Research supported by

Bundesministerium für Bildung und Forschung

Contact: Christoph Lüth and Martin Ring
Cyber Physical Systems
DFKI Bremen

E-mail: {martin.ring,christoph.lueth}@dfki.de
Website: http://clide.informatik.uni-bremen.de