

# Isabelle/Scala System Programming

Makarius Wenzel  
TU München

August 2009

# Motivation

## General aims:

- Renovate and reform traditional “LCF-style” theorem proving for coming generations of users and tool developers
- Catch up with technological shifts, e.g. advanced user-interfaces, parallel computing
- Support novel models for interactive proof checking

## Possible applications:

- Web client, based on server-side prover component
- Powerful proof editor, or “Prover IDE”
- Advanced document preparation, with rich semantic information

# Scala — <http://www.scala-lang.org>

## What is Scala anyway?

- *The Next Big Thing* in the JVM world (The Hype of 2010?)
- Nice integration of the best of
  - **object-oriented programming** (many steps beyond Java)
  - **higher-order functional programming** (many improvements over ML and Haskell, despite some compromises)
- Native support for “domain specific languages”

## Isabelle/Scala:

- Scala/JVM wrapping for the Isabelle process
- Integral part of Isabelle/Pure sources, `.ML` and `.scala` side-by-side
- Usual Isabelle/ML conventions carry over to Isabelle/Scala (forExampleWeDoNotUseMixedCaseIdentifiers)

# Isabelle/System layers

## Bottom-up structure:

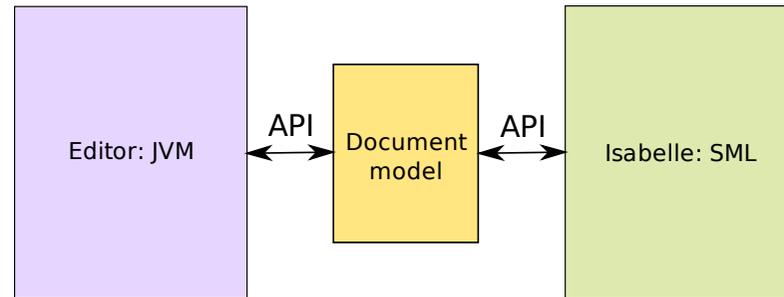
1. ML compiler and runtime system —  
Standard ML with tight integration into `lsar`
2. Posix system glue based on `bash` and `perl` —  
works uniformly on Linux, Mac OS, Windows (via Cygwin)
3. Scala/JVM wrapping — platform independent `.jar`

# Examples

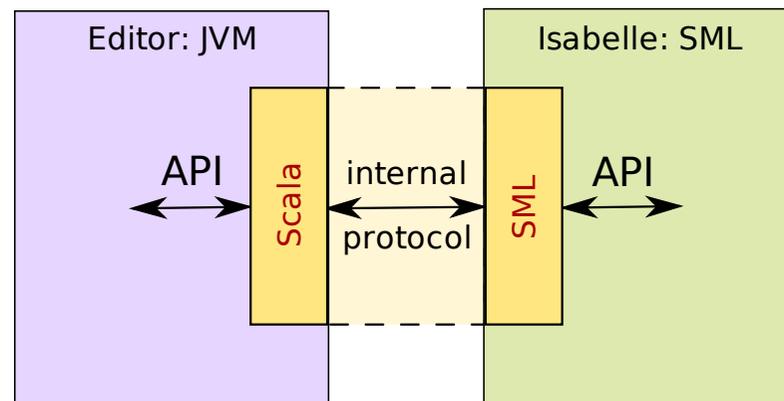
- Isabelle System
- Isabelle Process
- Basic Document Model

# Emerging interface architecture

## Conceptual view:



## Implementation view:



# Interface architecture

## Characteristics:

- + Regular API, based on internal protocol.
- + Supports mixed environments: Scala/JVM vs. SML.
- + Conceptual advances in proof document model:  
parallel checking, asynchronous interaction.
- Significant effort for design and initial implementation.
- Provers need to be adapted to interface needs.
- ++ Focussed towards particular platform: Scala/JVM

## Example: Isabelle/jEdit

- jEdit plugin written in Scala
- Basic support for forthcoming proof document model
- Discontinues typical “Proof General” interaction
- Contributors: Johannes Hölzl, Fabian Immler, Makarius Wenzel