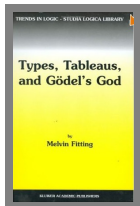


# Formalization, Mechanization and Automation of Gödel's Proof of God's Existence

Christoph Benzmüller and Bruno Woltzenlogel Paleo

November 1, 2013



$$\frac{\text{Axiom 3} \quad \frac{\frac{\forall \varphi. [P(\varphi) \rightarrow \Diamond \exists x. \varphi(x)]}{P(G) \rightarrow \Diamond \exists x. G(x)} \forall_E}{\Diamond \exists x. G(x)} \rightarrow_E$$

A gift to **Priest Edvaldo** and his church in Piracicaba, Brazil

**SPIEGEL ONLINE WISSENSCHAFT** Login | Registrierung

Politik | Wirtschaft | Panorama | Sport | Kultur | Netzwerk | Wissenschaft | Gesundheit | einestages | Karriere | Uni | Schule | Reise | Auto

Nachrichten > Wissenschaft > Mensch > Mathematik > Formel von Kurt Gödel: Mathematiker bestätigen Gottesbeweis

## Formel von Kurt Gödel: Mathematiker bestätigen Gottesbeweis

Von Tobias Hürter



picture-alliance/Imago/ Wiener Stadt- und Landesbibliothek

Kurt Gödel (um das Jahr 1935): Der Mathematiker hielt seinen Gottesbeweis Jahrzehntlang geheim

**Ein Wesen existiert, das alle positiven Eigenschaften in sich vereint. Das bewies der legendäre Mathematiker Kurt Gödel mit einem komplizierten Formelgebilde. Zwei Wissenschaftler haben diesen Gottesbeweis nun überprüft - und für gültig befunden.**

Montag, 09.09.2013 - 12:03 Uhr

Drucken [Versenden] [Markieren]

Jetzt sind die letzten Zweifel ausgeräumt: Gott existiert tatsächlich. Ein Computer hat es mit kalter Logik bewiesen - das MacBook des Computerwissenschaftlers Christoph Benzmüller von der Freien Universität Berlin.

## Germany

- Telepolis & Heise
- Spiegel Online
- FAZ
- Die Welt
- Berliner Morgenpost
- Hamburger Abendpost
- ...

## Austria

- Die Presse
- Wiener Zeitung
- ORF
- ...

## Italy

- Repubblica
- L'Espresso
- ...

## India

- DNA India
- Delhi Daily News
- India Today
- ...

## US

- ABC News
- ...

## International

- Spiegel International
- Yahoo Finance
- United Press Intl.
- ...

## SCIENCE NEWS

HOME / SCIENCE NEWS / RESEARCHERS SAY THEY USED MACBOOK TO PROVE GOEDEL'S GOD THEOREM

### Researchers say they used MacBook to prove Goedel's God theorem

Oct. 23, 2013 | 8:14 PM | [1 comments](#)

Are we in contact with Steve Jobs? No

Do you really need a MacBook to obtain the results? No

Is Apple sending us money? No  
(but maybe they should)

## Def: **Ontological Argument/Proof**

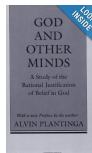
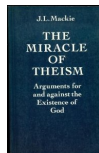
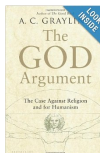
- \* deductive argument
- \* for the existence of god
- \* starting from premises, which are justified by pure reasoning, i.e. they do not depend on observation in the world.

## Existence of God: different types of arguments/proofs

- a posteriori (use experience/observation in the world)
  - teleological
  - cosmological
  - moral
  - ...
- a priori (based on pure reasoning, independent)
  - **ontological argument**
    - definitional
    - modal
    - ...
  - other a priori arguments

## *Wohl eine jede Philosophie kreist um den ontologischen Gottesbeweis*

(Adorno, Th. W.: Negative Dialektik. Frankfurt a. M. 1966, p.378)



Rich history on ontological arguments (pros and cons)

... Anselm v. G. Th. Aquinas Descartes Spinoza Leibniz Hume Kant Hegel Frege Hartshorne Malcolm Lewis Plantinga Gödel ...

Anselm's notion of God:

*"God is that, than which nothing greater can be conceived."*

Gödel's notion of God:

*"A God-like being possesses all 'positive' properties."*

To show by logical reasoning:

*"(Necessarily) God exists."*

## Different Interests in Ontological Arguments:

- **Philosophical:** Boundaries of Metaphysics & Epistemology
  - We talk about a metaphysical concept (God),  
● but we want to draw a conclusion for the real world.
  - Necessary Existence (NE): metaphysical NE vs. logical NE vs. modal NE
- **Theistic:** Successful argument should convince atheists.
- **Our:** Can computers (theorem provers) be used
  - to formalize the definitions and axioms?
  - to verify the arguments step-by-step?
  - to fully automate (sub-)arguments?

*“Computer-assisted Theoretical Philosophy”*

Main challenge: No provers for *Higher-order Modal Logic* (HML)

Our solution: Embedding in *Higher-order Classical Logic* (HOL)

[BenzmüllerPaulson, Logica Universalis, 2013]

What we did (rough outline for remaining presentation!):

A: pen and paper: detailed natural deduction proof

B: formalization: in classical higher-order logic (HOL)

automation: theorem provers LEO-II and SATALLAX

consistency: model finder NITPICK (NITROX)

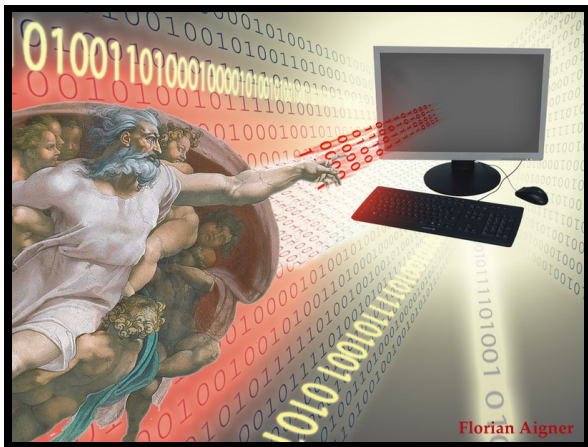
C: step-by-step verification: proof assistant Coq

D: automation & verification: proof assistant ISABELLE

Did we get new results?

Yes — let's discuss later!





## **Part A:** Informal Proof and Natural Deduction Proof

## ToDo: Improve Resolution

Ontologische Beweise

Feb 10, 1970

P(q) q is positive (i.e.  $q \in P$ )

At 1  $P(q) \cdot P(p) \supset P(q \cdot p)$  At 2  $P(p) \supset P(q)$

P1  $G(x) \equiv (q) [P(q) \supset q(x)]$  (God)

P2  $\varphi \text{ Ess. } x \equiv (\psi) [\psi(x) \supset N(\exists y) [\varphi(y) \supset \psi(y)]]$  (Essence of x)  
 $p \supset Nq = N(p \supset q)$  Necessity

At 2  $P(p) \supset NP(p)$   
 $\sim P(p) \supset N \sim P(p)$  } because it follows from the nature of the property

Th.  $G(x) \supset G \text{ Ess. } x$

Df.  $E(x) \equiv (p) [\varphi \text{ Ess. } x \supset N \exists x q(x)]$  necessary Existence

Ax 3  $P(E)$

Th.  $G(x) \supset N(\exists y) G(y)$

hence  $(\exists x) G(x) \supset N(\exists y) G(y)$

"  $M(\exists x) G(x) \supset M N(\exists y) G(y)$

"  $\supset N(\exists y) G(y)$

M = possibly

any two instances of x are nec. equivalent  
 exclusive or \* and for any number of summands

$M(\exists x) G(x)$  means <sup>the system of</sup> all pos. prop. is. com-  
 patible This is true because of:

At 4:  $P(q) \cdot q \supset N \psi \supset P(\psi)$  which impl.

~~hence~~  $\begin{cases} x=x & \text{is positive} \\ x \neq x & \text{is negative} \end{cases}$

But if a system S of pos. prop. were incomp.  
 It would mean that the sum prop. S (which  
 is positive) would be  $x \neq x$

Positive means positive in the moral sense  
 hence (independently of the accidental structure of  
 the world). <sup>Only in the at. time</sup> It means  
 also <sup>pure</sup> "attribution" as opposed to "privation"  
 (or containing privation). - This is important for the proof

$\exists x \neg \varphi$  privation of (x)  $N \neg \varphi(x)$  ~~otherwise~~  $\varphi(x) \supset x \neq x$

hence  $x \neq x$  positive ~~not~~  $x=x$  <sup>is</sup> necessary At

the ~~system of pos. prop.~~

<sup>does</sup> x i.e. the normal form in terms of elem. prop. contains  
 members without negation.

- A1 Either a property is positive or its negation is (never both):  
 $\forall\phi[P(\neg\phi) \leftrightarrow \neg P(\phi)]$
- A2 A property necessarily implied by a positive property is positive:  
 $\forall\phi\forall\psi[(P(\phi) \wedge \Box\forall x[\phi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]$
- T1 Positive properties are possibly exemplified:  
 $\forall\phi[P(\phi) \rightarrow \Diamond\exists x\phi(x)]$
- D1 A *God-like* being possesses all positive properties:  
 $G(x) \leftrightarrow \forall\phi[P(\phi) \rightarrow \phi(x)]$
- A3 The property of being God-like is positive:  $P(G)$
- C Possibly, God exists:  $\Diamond\exists xG(x)$
- A4 Positive properties are necessarily positive:  
 $\forall\phi[P(\phi) \rightarrow \Box P(\phi)]$
- D2 An *essence* of an individual is a property possessed by it and necessarily implying any of its properties:  
 $\phi \text{ ess } x \leftrightarrow \phi(x) \wedge \forall\psi(\psi(x) \rightarrow \Box\forall y(\phi(y) \rightarrow \psi(y)))$
- T2 Being God-like is an essence of any God-like being:  
 $\forall x[G(x) \rightarrow G \text{ ess } x]$
- D3 *Necessary existence* of an individual is the necessary exemplification of all its essences:  
 $E(x) \leftrightarrow \forall\phi[\phi \text{ ess } x \rightarrow \Box\exists y\phi(y)]$
- A5 Necessary existence is a positive property:  $P(E)$
- T3 Necessarily, God exists:  $\Box\exists xG(x)$

$$\mathbf{D1:} \ G(x) \equiv \forall\varphi.[P(\varphi) \rightarrow \varphi(x)]$$

$$\mathbf{D2:} \ \varphi \text{ ess } x \equiv \varphi(x) \wedge \forall\psi.(\psi(x) \rightarrow \Box\forall x.(\varphi(x) \rightarrow \psi(x)))$$

$$\mathbf{D3:} \ E(x) \equiv \forall\varphi.[\varphi \text{ ess } x \rightarrow \Box\exists y.\varphi(y)]$$

$$\begin{array}{c}
 \begin{array}{c}
 \mathbf{A3} \\
 \overline{P(G)}
 \end{array}
 \quad
 \frac{
 \frac{
 \overline{\forall\varphi.\forall\psi.[(P(\varphi) \wedge \Box\forall x.[\varphi(x) \rightarrow \psi(x)]) \rightarrow P(\psi)]}
 }{
 \mathbf{T1:} \ \forall\varphi.[P(\varphi) \rightarrow \Diamond\exists x.\varphi(x)]
 }
 }{
 \mathbf{C1:} \ \Diamond\exists x.G(x)
 }
 \\
 \\
 \begin{array}{c}
 \mathbf{A1b} \\
 \overline{\forall\varphi.[\neg P(\varphi) \rightarrow P(\neg\varphi)]}
 \end{array}
 \quad
 \begin{array}{c}
 \mathbf{A4} \\
 \overline{\forall\varphi.[P(\varphi) \rightarrow \Box P(\varphi)]}
 \end{array}
 \quad
 \begin{array}{c}
 \mathbf{A5} \\
 \overline{P(E)}
 \end{array}
 \\
 \frac{
 \mathbf{T2:} \ \forall y.[G(y) \rightarrow G \text{ ess } y]
 }{
 \mathbf{L1:} \ \exists z.G(z) \rightarrow \Box\exists x.G(x)
 }
 \quad
 \mathbf{S5} \\
 \overline{\forall\xi.[\Diamond\Box\xi \rightarrow \Box\xi]}
 \\
 \frac{
 \mathbf{L1:} \ \exists z.G(z) \rightarrow \Box\exists x.G(x)
 }{
 \mathbf{L2:} \ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)
 }
 \\
 \frac{
 \mathbf{C1:} \ \Diamond\exists x.G(x) \quad \mathbf{L2:} \ \Diamond\exists z.G(z) \rightarrow \Box\exists x.G(x)
 }{
 \mathbf{T3:} \ \Box\exists x.G(x)
 }
 \end{array}$$

$$\frac{A \vee B \quad \begin{array}{c} \overline{A} \\ \vdots \\ C \end{array} \quad \begin{array}{c} \overline{B} \\ \vdots \\ C \end{array}}{C} \vee_E$$

$$\frac{A \quad B}{A \wedge B} \wedge_I$$

$$\frac{\begin{array}{c} \overline{A} \\ \vdots \\ B \end{array} \quad n}{A \rightarrow B} \rightarrow_I^n$$

$$\frac{A}{A \vee B} \vee_{I_1}$$

$$\frac{A \wedge B}{A} \wedge_{E_1}$$

$$\frac{B}{A \rightarrow B} \rightarrow_I$$

$$\frac{B}{A \vee B} \vee_{I_2}$$

$$\frac{A \wedge B}{B} \wedge_{E_2}$$

$$\frac{A \quad A \rightarrow B}{B} \rightarrow_E$$

$$\frac{A[\alpha]}{\forall x.A[x]} \forall_I$$

$$\frac{\forall x.A[x]}{A[t]} \forall_E$$

$$\frac{A[t]}{\exists x.A[x]} \exists_I$$

$$\frac{\exists x.A[x]}{A[\beta]} \exists_E$$

$$\neg A \equiv A \rightarrow \perp$$

$$\frac{\neg\neg A}{A} \neg\neg_E$$

$$\frac{\alpha : \boxed{\begin{array}{c} \vdots \\ A \end{array}}}{\Box A} \Box_I$$

$$\frac{\Box A}{t : \boxed{\begin{array}{c} A \\ \vdots \end{array}}} \Box_E$$

$$\frac{t : \boxed{\begin{array}{c} \vdots \\ A \end{array}}}{\Diamond A} \Diamond_I$$

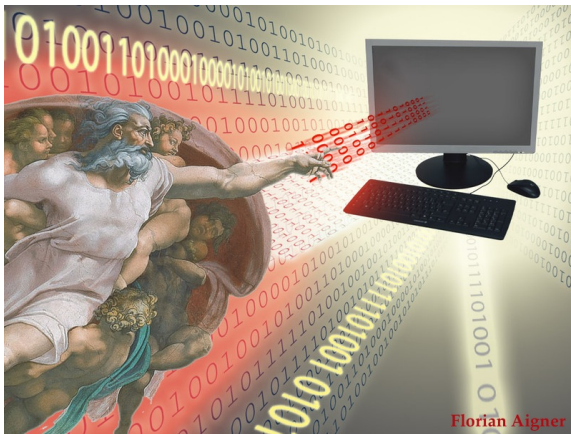
$$\frac{\Diamond A}{\beta : \boxed{\begin{array}{c} A \\ \vdots \end{array}}} \Diamond_E$$

$$\Diamond A \equiv \neg \Box \neg A$$

Christoph Benz Müller and Bruno Woltzenlogel Paleo

$$\begin{array}{c}
 \frac{\psi(x)^6 \quad \frac{\psi(x) \rightarrow \Box P(\psi)}{\Box P(\psi)} \Pi_2}{\Box P(\psi)} \rightarrow E \\
 \frac{\Box P(\psi) \quad \frac{\frac{\Box P(\psi)^7 \quad \frac{P(\psi) \rightarrow \forall x.(G(x) \rightarrow \psi(x))}{\forall x.(G(x) \rightarrow \psi(x))} \rightarrow E}{\Box \forall x.(G(x) \rightarrow \psi(x))} \Box I}{\Box P(\psi) \rightarrow \Box \forall x.(G(x) \rightarrow \psi(x))} \rightarrow I \\
 \frac{\Box P(\psi) \rightarrow \Box \forall x.(G(x) \rightarrow \psi(x))}{\psi(x) \rightarrow \Box \forall x.(G(x) \rightarrow \psi(x))} \rightarrow E
 \end{array}$$





## Part B:

formalization: in classical higher-order logic (HOL)

automation: theorem provers LEO-II and SATALLAX

consistency checking: model finder NITPICK (NITROX)

Main challenge: No provers for *Higher-order Modal Logic* (HML)

Our solution: Embedding in *Higher-order Classical Logic* (HOL)

Then use existing HOL theorem provers for reasoning in HML

[BenzmüllerPaulson, Logica Universalis, 2013]

Previous empirical findings:

Embedding of *First-order Modal Logic* in HOL works well

[BenzmüllerOttenRaths, ECAI, 2012]

[Benzmüller, LPAR, 2013]

**HML**    $\varphi, \psi ::= \dots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x \varphi \mid \exists x \varphi \mid \forall P \varphi$

- Kripke style semantics (possible world semantics)

**HOL**    $s, t ::= C \mid x \mid \lambda x s \mid s t \mid \neg s \mid s \vee t \mid \forall x t$

- meanwhile very well understood
- Henkin semantics vs. standard semantics
- various theorem provers do exist

interactive: Isabelle/HOL, HOL4, Hol Light, Coq/HOL, PVS, ...

automated:    TPS, LEO-II, Satallax, Nitpick, Isabelle/HOL, ...

**HML**  $\varphi, \psi ::= \dots \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \Box\varphi \mid \Diamond\varphi \mid \forall x \varphi \mid \exists x \varphi \mid \forall P \varphi$

**HOL**  $s, t ::= C \mid x \mid \lambda x s \mid s t \mid \neg s \mid s \vee t \mid \forall x t$

**HML** in **HOL**: **HML** formulas  $\varphi$  are mapped to **HOL** predicates  $\varphi_{t \rightarrow o}$

$\neg$	=	$\lambda\varphi_{t \rightarrow o} \lambda s_t \neg\varphi s$	<b>Ax</b>
$\wedge$	=	$\lambda\varphi_{t \rightarrow o} \lambda\psi_{t \rightarrow o} \lambda s_t (\varphi s \wedge \psi s)$	
$\rightarrow$	=	$\lambda\varphi_{t \rightarrow o} \lambda\psi_{t \rightarrow o} \lambda s_t (\neg\varphi s \vee \psi s)$	
$\Box$	=	$\lambda\varphi_{t \rightarrow o} \lambda s_t \forall u_t (\neg r s u \vee \varphi u)$	
$\Diamond$	=	$\lambda\varphi_{t \rightarrow o} \lambda s_t \exists u_t (r s u \wedge \varphi u)$	
$\forall$	=	$\lambda h_{\mu \rightarrow (t \rightarrow o)} \lambda s_t \forall d_\mu h d s$	
$\exists$	=	$\lambda h_{\mu \rightarrow (t \rightarrow o)} \lambda s_t \exists d_\mu h d s$	
$\forall$	=	$\lambda H_{(\mu \rightarrow (t \rightarrow o)) \rightarrow (t \rightarrow o)} \lambda s_t \forall d_\mu H d s$	
<b>valid</b>	=	$\lambda\varphi_{t \rightarrow o} \forall w_t \varphi w$	

The equations in **Ax** are given as axioms to the **HOL** provers!

## Example

HML formula

HML formula in HOL

expansion,  $\beta\eta$ -conversion

expansion,  $\beta\eta$ -conversion

expansion,  $\beta\eta$ -conversion

$\Diamond \exists x G(x)$

valid  $(\Diamond \exists x G(x))_{l \rightarrow o}$

$\forall w_l (\Diamond \exists x G(x))_{l \rightarrow o} w$

$\forall w_l \exists u_l (rwu \wedge (\exists x G(x))_{l \rightarrow o} u)$

$\forall w_l \exists u_l (rwu \wedge \exists x Gxu)$

## What are we doing?

In order to prove that  $\varphi$  is valid in HML,

$\rightarrow$  we instead prove that valid  $\varphi_{l \rightarrow o}$  can be derived from Ax in HOL.

This can be done with interactive or automated HOL theorem provers.

Expansion: user or prover may flexibly choose expansion depth

# Proof Automation and Consistency Checking: Demo!

```
Terminal — bash — 125x32
MacBook-Chris %
MacBook-Chris %
MacBook-Chris % ./call_tptp.sh T3.p

Asking various HOL-ATPs in Miami remotely (thanks to Geoff Sutcliffe)

MacBook-Chris % agsyH0L---1.0 : T3.p +++++ RESULT: S0T_7L4x_Y - agsyH0L---1.0 says Unknown - CPU = 0.00 WC = 0.02
LE0-II---1.6.0 : T3.p +++++ RESULT: S0T_E4SCha - LE0-II---1.6.0 says Theorem - CPU = 0.03 WC = 0.09
Satallax---2.7 : T3.p +++++ RESULT: S0T_kVZ1cB - Satallax---2.7 says Theorem - CPU = 0.00 WC = 0.14
Isabelle---2013 : T3.p +++++ RESULT: S0T_xa0gEp - Isabelle---2013 says Theorem - CPU = 14.06 WC = 17.73 SolvedBy = auto
TPS---3.120601S1b : T3.p +++++ RESULT: S0T_R0Egsg - TPS---3.120601S1b says Unknown - CPU = 33.56 WC = 41.57
Nitrox---2013 : T3.p +++++ RESULT: S0T_WGY1Tx - Nitrox---2013 says Unknown - CPU = 75.55 WC = 49.24

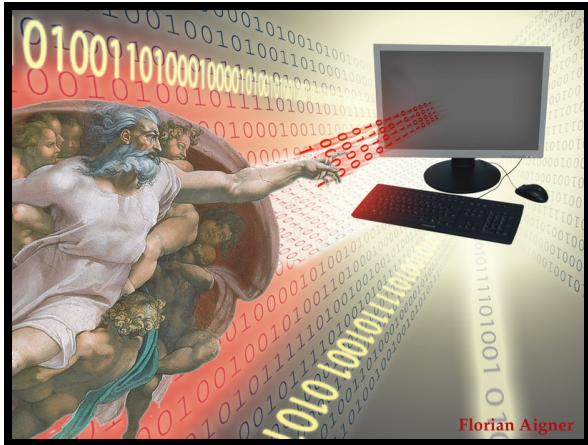
MacBook-Chris %
MacBook-Chris % ./call_tptp.sh Consistency.p

Asking various HOL-ATPs in Miami remotely (thanks to Geoff Sutcliffe)

MacBook-Chris % agsyH0L---1.0 : Consistency.p +++++ RESULT: S0T_ZtY_7o - agsyH0L---1.0 says Unknown - CPU = 0.00 WC = 0.00
Nitrox---2013 : Consistency.p +++++ RESULT: S0T_HUz10C - Nitrox---2013 says Satisfiable - CPU = 6.56 WC = 8.50
TPS---3.120601S1b : Consistency.p +++++ RESULT: S0T_fpJxTM - TPS---3.120601S1b says Unknown - CPU = 43.00 WC = 49.42
Isabelle---2013 : Consistency.p +++++ RESULT: S0T_6Tpp9i - Isabelle---2013 says Unknown - CPU = 69.96 WC = 72.62
LE0-II---1.6.0 : Consistency.p +++++ RESULT: S0T_dY10sj - LE0-II---1.6.0 says Timeout - CPU = 90 WC = 89.86
Satallax---2.7 : Consistency.p +++++ RESULT: S0T_Q9WSLf - Satallax---2.7 says Timeout - CPU = 90 WC = 90.50

MacBook-Chris %
```

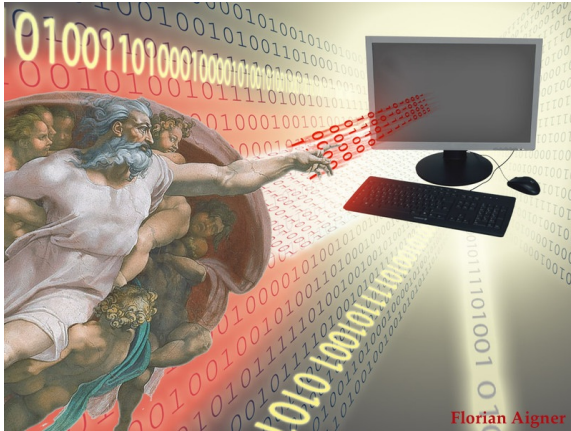
Provers are called remotely in Miami — no local installation needed!



## Part C: Formalization and Verification in Coq

- Goal: verification of the natural deduction proof
  - Step-by-step formalization
  - Almost no automation (intentionally!)
- Interesting facts to note:
  - Embedding is transparent to the user
  - Embedding gives labeled calculus for free





## **Part D:** automation & verification: proof assistant ISABELLE



[Home](#)
[Overview](#)
[Installation](#)
[Documentation](#)
[Community](#)

**Site Mirrors:**  
[Cambridge \(UK\)](#)  
[Munich \(DE\)](#)  
[Sydney \(AU\)](#)

[Home-FU](#)
[2012-Watson](#)
[Homepage](#)
[2012-FOL](#)
[SPIEGEL](#)
[2012-FOL-Home](#)
[GMail](#)
[Google Maps](#)
[M&M](#)
[SigmaOnline](#)
[Kita](#)
[Sigma](#)
[Kalender](#)
[Beliebt](#)
[Google Maps](#)

# Isabelle



UNIVERSITY OF  
CAMBRIDGE  
Computer Laboratory



TECHNISCHE  
UNIVERSITÄT  
MÜNCHEN

## What is Isabelle?

Isabelle is a generic proof assistant. It allows mathematical formulas to be expressed in a formal language and provides tools for proving those formulas in a logical calculus. Isabelle is developed at University of Cambridge ([Larry Paulson](#)), Technische Universität München ([Tobias Nipkow](#)) and Université Paris-Sud ([Makarius Wenzel](#)). See the [Isabelle overview](#) for a brief introduction.

## Now available: Isabelle2013



Download for  
Mac OS X



[Download for Linux](#) - [Download for Windows](#)

### Some highlights:

- Improvements of Isabelle/Scala and Isabelle/Edit Prover IDE.
- Advanced build tool based on Isabelle/Scala.
- Updated manuals: Isar-ref, implementation, system.
- Pure: improved support for block-structured specification contexts.
- HOL tool enhancements: Sledgehammer, Lifting, Quickcheck.
- HOL library enhancements: HOL-Library, HOL-Probability, HOL-Cardinals.
- HOL: New BNF-based (co)datatype package.
- Improved performance thanks to Poly/ML 5.5.0.

See also the cumulative [NEWS](#).

## Distribution & Support

Isabelle is distributed for free under the BSD license. It includes source and binary packages and documentation, see the detailed [installation instructions](#). A vast collection of Isabelle examples and applications is available from the [Archive of Formal Proofs](#).

Support is available by ample [documentation](#), the [Isabelle community Wiki](#), and the following mailing lists:

- [isabelle-users@cl.cam.ac.uk](#) provides a forum for Isabelle users to discuss problems, exchange information, and make announcements. Users of official Isabelle releases should [subscribe](#) or see the [archive](#) (also available via [Google groups](#) and [Narkive](#)).
- [isabelle-dev@in.tum.de](#) covers the Isabelle development process, including intermediate repository versions, and administrative issues concerning the website or testing infrastructure. Early adopters of [repository versions](#) should [subscribe](#) or see the [archive](#) (also available at [mail-archive.com](#) or [gmane.org](#)).

Last updated: 2013-03-09 12:21:39

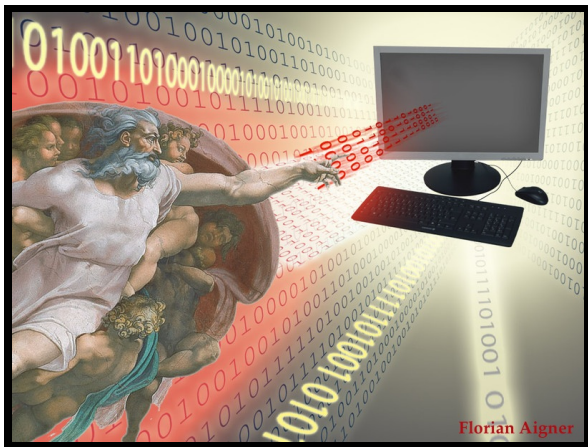
Isabelle

- bla

What have we done

- bla

See the handout (generated from the Isabelle source file).

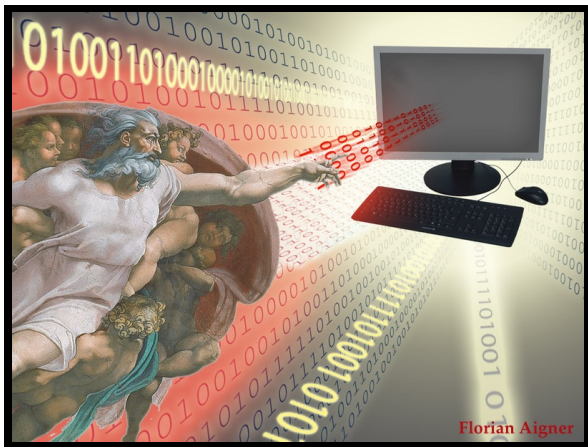


## Part E: Criticisms







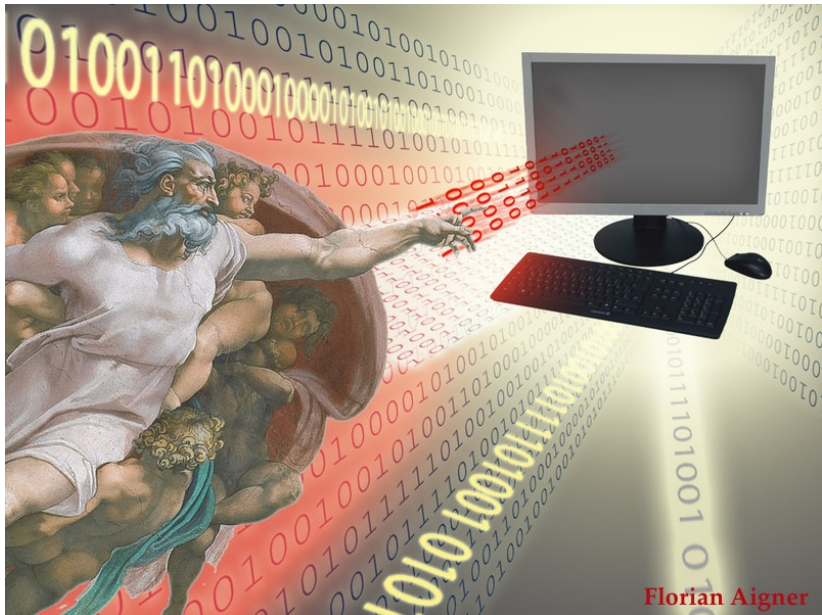


## Part F: Conclusions



- K sufficient for T1, C1 and T2
- S5 not needed for T3
- KB sufficient for T3
- A simpler new proof of C1
- Gödel's original axioms (without conjunct  $\phi(x)$  in D2) are inconsistent
- Scott's axioms are consistent
- For T1, only half of A1 (A1a) is needed
- For T2, the other half (A1b) is needed

- Infra-structure for reasoning with modal logic using existing proof assistants and higher-order automated theorem provers
- A new natural deduction calculus for higher-order modal logic



**Florian Aigner**