HOL4 users' workshop

Tuesday 10th – Wednesday 11th June 2025

Arm, Cambridge, UK

Contacts: Hrutvik Kanabar, Hrutvik.Kanabar@arm.com
Eleni Vafeiadi Bila, Eleni.VafeiadiBila@arm.com

List of attendees

* = *virtual*

*Alex Shkotin	*ACM, USA*
Albert Rizaldi	*PlanV GmbH, Germany*
Andreas Lööw	*Imperial College London, UK*
Anthony Fox	*Arm, UK*
Daniel Nezamabadi	*ETH Zurich, Switzerland* and *Chalmers, Sweden*
Didrik Lundberg	*KTH, Sweden* and *Saab AB, Sweden*
Eleni Vafeiadi Bila	*Arm, UK*
Gergely Buday	*University of Sheffield, UK*
Hrutvik Kanabar	*Arm, UK*
*Irvin Ng	*Temasek Poly Student, Singapore*
*Johannes Åman Pohjola	*Chalmers, Sweden* and *University of Gothenburg, Sweden*
*Konrad Slind	*unaffiliated*
Magnus Myreen	*Arm, Sweden*
Mario Carneiro	*Chalmers, Sweden*
*Michael Norrish	*ANU, Australia*
Mohammad Abdulaziz	*King's College London, UK*
*Nikos Alexandris	*Chalmers, Sweden*
Pascal Lasnier	*University of Cambridge, UK*
Ramana Kumar	*Verifereum, UK*
Robert Söldner	*University of York, UK*
Thaïs Baudon	*University of Kent, UK*
Thibaut Pérami	*University of Cambridge, UK*
Thomas Bauereiss	*University of Cambridge, UK*
Vineet Rajani	*University of Kent, UK*
Yong Kiam Tan	*NTU, Singapore*

Programme – day 1

09:00–09:30 **Arrival and admission**

09:30–10:00 **Lightning introductions**

10:00–10:45 **Tips + tricks: using `AncestryData`**

Michael Norrish (ANU, Australia)

Do you want to stash data in theory files so that your Wonderful Tool™ can use per-theory information later? Data that you might like to stash includes things like custom theorems that you will use as rewrites, but might also be meta-data about constants like how it should be parsed/printed.

Once upon a time you could directly (but rather tediously) write SML code into `xTheory.sml` files to do this, but that is (i) gross; (ii) really gross; and (iii) now impossible. I will explain the "right" way to achieve this objective, with examples from the system (the simplifier, the parser and others), and some of the nice features the technology enables.

10:45–11:15 **Break**

11:15–12:00 **Documentation** ***discussion***

Led by: Anthony Fox (Arm, UK)

We all know that well written and accessible *documentation is super important for the success of tools/systems. Existence of (reasonably good) documentation used to be one of HOL4's strengths. But we've been sitting on our laurels, as things have not changed much in the last 20+ years. There are the big PDF manuals (Description, Tutorial and Reference); the help `.doc` files (which can be converted to HTML and `.txt`); and an assortment of guides/cheatsheets. Other tools/systems show us that there's tons of room for improvement. Can we draw inspiration from these? Can we spread the burden of overhauling things, given that it is nobody's "day job"?*

12:00–13:00 **Lunch**

Programme – day 1 (continued)

13:00–13:30 **GOL in GOL in HOL: using `cv_compute` on Conway's Game of Life**

Magnus Myreen (Arm, Sweden)

I will give a short presentation of GOL in GOL in HOL which has been accepted at ITP. I'll explain how HOL's new `cv_compute` function made this work possible.

13:30–13:45 **Brack: verified compilation of Scheme to CakeML**

Pascal Lasnier (University of Cambridge, UK)

Brack performs a CPS transform from Scheme to CakeML, which is directed by its small-step semantic definition. Semantic preservation is a simulation proof, which reconciles small- and big-step semantics by relating Scheme small-step semantic states to ML CPS expressions.

13:45–14:15 **A HOL-to-ACL2 Connection**

Konrad Slind (unaffiliated)

In recent work, Matt Kaufmann has been formalizing ZFC set theory in ACL2. Since ZFC is expressive enough to represent the HOL logic, this creates a good opportunity for us to map HOL formalizations over to ACL2. We discuss the current design and implementation of such a translation.

14:15–14:30 **Finite automata theory**

Konrad Slind (unaffiliated)

Finite state automata theory is a building block of theoretical computer science, and also provides the formal basis for regexp search, lexers, and other important system tools. I discuss some issues with formalizing automata theory, and also explore how it can be used to build arithmetic decision procedures.

14:30–14:45 **Verifereum + verified compilation for Vyper**

Ramana Kumar (Verifereum, UK)

Current status, future plans, and opportunities for collaboration on Verifereum: the project enabling proof of implementation correctness for Ethereum applications in HOL4.

14:45–15:15 **Break**

Programme – day 1 (continued)

15:15–15:45 **Towards a modern IDE experience for HOL**

Mario Carneiro (Chalmers, Sweden)

In this talk, I'll discuss (and demo) recent work done on the VSCode mode for HOL to make it easier to use it interactively, with live diagnostics, editor hovers for types and jump to definition. I will also discuss future plans and the HOL changes to support these features.

15:45–16:15 **Compiling formal network semantics: lessons from CakeML and HOL4P4**

Didrik Lundberg (KTH, Sweden *and* Saab AB, Sweden)

I will present some results, challenges, practical experiences and lessons learned from recent experiments using CakeML with our P4 semantics.

16:15–16:45 **Secure compilation of the Declassification Core Calculus**

Thaïs Baudon (University of Kent, UK)

Declassification Core Calculus (DeCC), Rajani et al. CSF'25, is a recent modal type theory for reasoning about rich security properties with deliberate information disclosure, a.k.a. declassification. In this work, we look at the problem of compiling DeCC to a calculus with nonmodal information flow types which only supports vanilla noninterference (and no declassification). We show that our compilation preserves both types and semantics. Additionally, we also show that relaxed semantic declassification (soundness criteria for the source, DeCC) can be recovered from noninterference (soundness criteria of the target), along with additional properties of our compilation scheme. This work is being mechanized in the HOL4 theorem prover.

19:00 **Workshop dinner @ Rice Boat**

Programme – day 2

09:00–09:30 **Arrival and admission**

09:30–10:30 **UI/UX** ***discussion***

Led by: Mario Carneiro (Chalmers, Sweden)

Currently, HOL has a long way to go before it will look attractive for newcomers, and editor experience is a major factor in this. How can we reimagine the HOL editing experience to feel more modern? Are there specific pain points with current tooling? Let's lay out a path from here to the shiny future HOL we would like to see.

10:30–11:00 **Break**

11:00–12:00 **Package management** ***discussion***

Led by: Daniel Nezamabadi (ETH Zurich, Switzerland *and* Chalmers, Sweden)

Designing a system to ease building/downloading and installing HOL4 and working with HOL theories. More context is in issue #1392. Let's discuss the design and try to get it to the point of being implementable.

12:00–13:00 **Lunch**

13:00–14:00 **The future of HOL4 development and use** ***discussion***

Led by: Ramana Kumar (Verifereum, UK)

> *"HOL ... is completely moribund with all the legacy code holding it back" – Mario Carneiro*
>
> *"I'd be willing to bet you'd have more users if the documentation was better, because it's so inaccessible right now" – yatchan*

The HOL4 community (i.e. people working on or using the system) is small, and correspondingly disadvantaged in the competition for funding, attention, and growth with other theorem proving systems like Lean, Rocq, and Isabelle. Should we pack up and migrate our projects to one of these other systems? Or what might we be able to do to spin up a flywheel of contributions towards HOL4?

Programme – day 2 (continued)

14:00–14:45 **What is the next HOL platform?** ***discussion***

Led by: Konrad Slind (unaffiliated)

Poly/ML has been the main platform for HOL for quite some time. It has served us well, but it will probably not be around forever and some day David Matthews will stop developing and supporting it. We should try to think through the possibilities.
Questions:

1. *Should we stay with an ML? Eventually join in maintenance/dev of Poly/ML if/when Dave Matthews stops working on it? Shift to Candle? Shift to OCaml?*
2. *Maybe we should shift to a mainstream language and build whatever tooling is needed to get a productive environment?*
3. *What about merging with HOL Light? Good idea, or no? How would it work?*
4. *This isn't research work, so not a task academic HOL users can spend a lot of time on. (Maybe I am wrong?) How to pay for the work to get done?*

14:45–15:15 **Break**

15:15–16:15 **A guiding light for HOL4** ***discussion***

Led by: Ramana Kumar (Verifereum, UK)

What do we want HOL4 to become? Could the future include a merger with Candle? Let's discuss the items on an initial Zulip wishlist from Ramana, find points that need further design work, points of contention and consensus, and points that are motivating or inspiring for present and future HOL developers.

16:15– **Discussion and end of day**