The Theorem Prover Museum

Conserving the System Heritage of Automated Reasoning

Michael Kohlhase

Abstract [1]
                                                         EdN:1

1 Introduction

Theorem provers are software systems that can find or check proofs for conjectures given in some logic. Research in theorem proving systems started with Newell and Simon's "logic theorist" 1955 [NS56] – one of the earliest systems in the then-emerging field of Artificial Intelligence – and has led to a succession of systems since. Today, more than 60 years later, the CADE ATP system competition [CASC] attracts 15-20 systems annually. Automated reasoning systems have applications ranging from the verification of mathematical results, via program synthesis/verification, the Semantic Web, all the way to the discovery of unfair trading rules in darkpools of investment banks.

Theorem provers are complex software systems that have pushed the envelope of artificial intelligence and programming, and as such they constitute important cultural artefacts.[2]   EdN:2

With the ongoing wave of retirements of the original principal investigators there is good chance that these systems are lost, when their group servers are shut down. The following incident is unfortunately quite typical. When – ten days after Herbert Simon's passing in February 2001 – the author tried to find a copy of the source code of the Logic Theorist in Simon's scientific estate at CMU, all tapes and printouts had already been discarded – only the written materials and notes

---

Computer Science, FAU Erlangen-Nürnberg
`http://kwarc.info/kohlhase`

[1] EDNOTE: tbw

[2] EDNOTE: give some examples, some things that were first done in theorem provers, e.g. the programming language ML which heavily influenced modern typed functional programs was introduced as a meta-language of the LCF theorem prover by Robin Milner. Its type system was motivated by the idea that proofs could be programmed, if the type of proofs can only contain logically valid proofs.

were being catalogued in the CMU library. Fortunately, report P-868 of the Rand Corporation [NS56], where the program was conceived contains the full printout of the code. Otherwise we would only be able to read about this seminal program, but not be able to study the artefact itself.

EdN:3

In other cases, we may not have been so lucky; see [TPL] for a list of theorem provers believed lost. This is a great loss to[3]

2 A Museum of Theorem Prover Source Code

This article reports on an initiative started by the author in spring 2016 to help conserve the source code of theorem provers: the "therorem prover museum", a collection of GitHub repositories repositories with source code of systems, together with a web site that presents them and organizes the process of acquiring more.

The term "museum" in the title may be sound bit ambitious, since the exhibition and didactic interpretation of the theorem provers is beyond the scope of the initiative (and perhaps abilities of the founder). But the foremost function of any museum is the conservation of artefacts, which is what the "theorem prover museum" project intends to do. Once the source code is preserved, historians of science and engineering can start to do research on it and create multiple user interfaces to present it to the public.

Note that it is not the purpose of the museum to keep the theorem proving systems running (in many cases the compilers and dependencies have moved on, making this very difficult). But only to archive the source code for academic study. This is a well-considered design decision, taken to lower the barrier of archiving systems here. Again, once the source code is preserved – i.e. made public by the original authors – other enthusiasts can revive it. Indeed this has already happened, triggered by the act of exposing the source.

EdN:4

The concept of the theorem prover museum is compatible with the Software Heritage initiative [SH], and contributes to it since it is based on GitHub repositories.[4]

3 The Museum

The actual "theorem prover museum" consists of a simple web site that features a list of GitHub repositories that contain the actual source code. The repositories are collected in the GitHub organization `theoremprover-museum` `https://github. com/theoremprover-museum`

This site is the front-end to a collection of source code repositories for theorem provers

---

[3] EDNOTE: explain why!, history of science/engineering.

[4] EDNOTE: also way what swMath has done with the special category

4 Call for Contributions

To exhibit your system here or suggest a system for inclusion (most wanted list, systems believed lost), please contact Michael Kohlhase. Contributors, community/contact, project/issues.

5 Conclusion

Acknowledgements [5]   EdN:5

References

[CASC]   *The CADE ATP System Competition. The World Championship for Automated Theorem Proving.* URL: `http://www.cs.miami.edu/~tptp/CASC/` (visited on 09/07/2012).

[NS56]   Allen Newell and Herbert A. Simon. *The Logic Theory Machine, A complex information processing system.* Tech. rep. P-868. Rand Corporation, 1956. URL: `http://shelf1.library.cmu.edu/IMLS/MindModels/logictheorymachine.pdf`.

[SH]   *Software Heritage.* URL: `https://www.softwareheritage.org/` (visited on 04/07/2017).

[TPL]   *Theorem Provers believed lost.* URL: `http://theoremprover-museum.github.io/systems-believed-lost.html` (visited on 09/02/2017).

---

[5] EDNOTE: Jörg,...