# Industrial Use of ACL2: Applications, Achievements, Challenges, and Directions

**J Strother Moore and Marijn J.H. Heule**

http://www.cs.utexas.edu/users/moore/acl2


ARCADE in Gothenburg, Sweden August 6, 2017

# Why ACL2 is Successful in Industry

- that was the goal of the project
- efficient, executable logic/programming language with native verifier
- dual-use bit- and cycle-accurate models
- access to Common Lisp programming (via trust tags)
- automatic prover with "a human in the loop"
- rugged, well documented, free, open source form, many useful books, and a fairly unrestrictive license.
- coherent user community devoted to making mechanized verification practical
- industry needs help

# Industrial Achievements

ACL2 achievements include the verification of:

- all elementary floating-point arithmetic on the AMD Athlon
- all elementary floating-point arithmetic on the AMD Opteron
- a silicon implementation of a JVM chip by Rockwell Collins
- the Rockwell Collins AAMP7 crypto chip
- the Green Hills operating system
- the Centaur Technology, Inc., Verilog design for the VIA Nano floating point adder
- floating point designs at Oracle and ARM

**All verifications were performed in-house by full-time employees**

# Directions (or Weaknesses Reported by Industry)

- inefficient execution of some primitives
- inconvenient as a scripting language
- does not support visualization/graphics tools

Note: Industry's complaints about ACL2 rarely concern absence of strong typing, explicit quantifiers, partial functions, or HOL

# Question 1: Users vs Value

Reviewer: "*real industry penetration of automated deduction tools is achieved if the tools are routinely used by people who have little to no understanding of their inner workings*"

Do we measure the success of a theorem prover (or any tool) in an industry by how many people use it or by the value it brings to the industry?

If a single person saves a company half-a-billion dollars by using an obscure tool but is the only person in the company using the tool, do we deem the tool doomed to obscurity?

# Question 2: Industrial vs Academic Interests

If you want to build a theorem prover used by industry, listen to what industrial users want.

Is anyone working on a formally defined programming language with accompanying proof engine that supports scripting and GUIs and is rugged and powerful enough to model, simulate and prove properties of large digital artifacts like the x86 or the Java Virtual Machine?

# Industrial Use of ACL2: Applications, Achievements, Challenges, and Directions

**J Strother Moore and Marijn J.H. Heule**

http://www.cs.utexas.edu/users/moore/acl2


ARCADE in Gothenburg, Sweden August 6, 2017