

ACL2 User Support

Matt Kaufmann (speaker)
J Strother Moore

The University of Texas at Austin

April 26, 2016

INTRODUCTION

- ▶ ACL2: **A Computational Logic for Applicative Common Lisp**".

INTRODUCTION

- ▶ ACL2: **A Computational Logic for Applicative Common Lisp**".
- ▶ Home page:
<http://www.cs.utexas.edu/users/moore/acl2/>

INTRODUCTION

- ▶ ACL2: **A Computational Logic for Applicative Common Lisp**".
- ▶ Home page:
<http://www.cs.utexas.edu/users/moore/acl2/>
- ▶ Distribution site for ACL2 system and *community books* (recent count: 4,765):
<https://github.com/acl2/acl2>

INTRODUCTION

- ▶ ACL2: **A Computational Logic for Applicative Common Lisp**".
- ▶ Home page:
<http://www.cs.utexas.edu/users/moore/acl2/>
- ▶ Distribution site for ACL2 system and *community books* (recent count: 4,765):
<https://github.com/acl2/acl2>
- ▶ Today I'll discuss our focus on ACL2 users.

INTRODUCTION

- ▶ ACL2: **A Computational Logic for Applicative Common Lisp**".
- ▶ Home page:
<http://www.cs.utexas.edu/users/moore/acl2/>
- ▶ Distribution site for ACL2 system and *community books* (recent count: 4,765):
<https://github.com/acl2/acl2>
- ▶ Today I'll discuss our focus on ACL2 users.
 - ▶ Very high-level, but feel free to ask for details

FOCUS ON USERS

FOCUS ON USERS

- ▶ Responsiveness to requests

FOCUS ON USERS

- ▶ Responsiveness to requests
- ▶ Debugging tools (both for proofs and for programming)

FOCUS ON USERS

- ▶ Responsiveness to requests
- ▶ Debugging tools (both for proofs and for programming)
- ▶ Documentation: web and emacs, system and books;
[DEMO]: (DEBUGGING and GL)

FOCUS ON USERS

- ▶ Responsiveness to requests
- ▶ Debugging tools (both for proofs and for programming)
- ▶ Documentation: web and emacs, system and books;
[DEMO]: (DEBUGGING and GL)
- ▶ Helpful output, including:
 - ▶ Useful error and warning messages
 - ▶ Gag-mode [DEMO]
 - ▶ Iprinting [DEMO]

FOCUS ON USERS

- ▶ Responsiveness to requests
- ▶ Debugging tools (both for proofs and for programming)
- ▶ Documentation: web and emacs, system and books;
[DEMO]: (DEBUGGING and GL)
- ▶ Helpful output, including:
 - ▶ Useful error and warning messages
 - ▶ Gag-mode [DEMO]
 - ▶ Iprinting [DEMO]
- ▶ Support for efficient programming;
e.g., Centaur writes many tools in the ACL2 programming language

FOCUS ON USERS

- ▶ Responsiveness to requests
- ▶ Debugging tools (both for proofs and for programming)
- ▶ Documentation: web and emacs, system and books;
[DEMO]: (DEBUGGING and GL)
- ▶ Helpful output, including:
 - ▶ Useful error and warning messages
 - ▶ Gag-mode [DEMO]
 - ▶ Iprinting [DEMO]
- ▶ Support for efficient programming;
e.g., Centaur writes many tools in the ACL2 programming language
- ▶ Support for several platforms

RELEASE NOTES (1)

Quoting from a paper to appear in the *Philosophical Transactions of the Royal Society*:

RELEASE NOTES (1)

Quoting from a paper to appear in the *Philosophical Transactions of the Royal Society*:

The release note records indicate that Centaur has been engaged with ACL2 since 2009. ACL2 Version 3.5 was released in May, 2009, and so we start our statistics gathering there and go through Version 7.2, released in January, 2016. We count the changes requested by Centaur, by category.

Changes to Existing Features	95
New Features	44
Heuristic and Efficiency Improvements	22
Bug Fixes	72
Changes at the System Level	18

RELEASE NOTES (2)

The changes on the preceding slide include:

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:
 - ▶ Lisp-level: wall-clock time, Common Lisp compiler directives like inlining, . . .

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:
 - ▶ Lisp-level: wall-clock time, Common Lisp compiler directives like inlining, . . .
 - ▶ Faster I/O

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:
 - ▶ Lisp-level: wall-clock time, Common Lisp compiler directives like inlining, . . .
 - ▶ Faster I/O
 - ▶ New read macros supporting Verilog syntax

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:
 - ▶ Lisp-level: wall-clock time, Common Lisp compiler directives like inlining, . . .
 - ▶ Faster I/O
 - ▶ New read macros supporting Verilog syntax
 - ▶ Fundamental new features like *congruent* and *nested single-threaded objects* (stobj), as well as *patterned congruences*
 - used in our x86 work, as are *abstract stobj*s

RELEASE NOTES (2)

The changes on the preceding slide include:

- ▶ Improved facilities for documentation, warning, and error messages
- ▶ New features, e.g.:
 - ▶ Lisp-level: wall-clock time, Common Lisp compiler directives like inlining, . . .
 - ▶ Faster I/O
 - ▶ New read macros supporting Verilog syntax
 - ▶ Fundamental new features like *congruent* and *nested single-threaded objects* (stobj), as well as *patterned congruences*
 - used in our x86 work, as are *abstract stobj*s
- ▶ Performance improvements

RELEASE NOTES (3)

Release note counts:

<i>2012, Version 5.0:</i>	125
<i>2012, Version 6.0:</i>	53
<i>2013, Version 6.1:</i>	26
<i>2013, Version 6.2:</i>	41
<i>2013, Version 6.3:</i>	36
<i>2014, Version 6.4:</i>	27
<i>2014, Version 6.5:</i>	49
<i>2015, Version 7.0:</i>	82
<i>2015, Version 7.1:</i>	52
<i>2016, Version 7.2:</i>	91

CODE SIZE

7.7 MB	<i>2004, Version 2.8</i>	12.4 MB	<i>2011, Version 4.3</i>
7.7 MB	<i>2004, Version 2.9</i>	13.4 MB	<i>2012, Version 5.0</i>
8.3 MB	<i>2006, Version 3.0</i>	14.1 MB	<i>2012, Version 6.0</i>
8.5 MB	<i>2006, Version 3.1</i>	14.1 MB	<i>2013, Version 6.1</i>
8.9 MB	<i>2007, Version 3.2</i>	14.1 MB	<i>2013, Version 6.2</i>
9.3 MB	<i>2007, Version 3.3</i>	14.3 MB	<i>2013, Version 6.3</i>
9.8 MB	<i>2008, Version 3.4</i>	14.3 MB	<i>2014, Version 6.4</i>
10.2 MB	<i>2009, Version 3.5</i>	14.4 MB	<i>2014, Version 6.5</i>
10.3 MB	<i>2009, Version 3.6</i>	14.7 MB	<i>2015, Version 7.0</i>
11.4 MB	<i>2010, Version 4.0</i>	14.8 MB	<i>2015, Version 7.1</i>
11.4 MB	<i>2010, Version 4.1</i>	14.9 MB	<i>2016, Version 7.2</i>
11.7 MB	<i>2011, Version 4.2</i>	15.1 MB	<i>4/22/2016</i>

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

This change was made to support usage in industry:

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

This change was made to support usage in industry:

- ▶ Intel was interested in including ACL2 as part of a verification product shipped outside Intel, but GPL made that impossible.

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

This change was made to support usage in industry:

- ▶ Intel was interested in including ACL2 as part of a verification product shipped outside Intel, but GPL made that impossible.
- ▶ Rockwell Collins also expressed interest:

“FWIW, I continue to hope earnestly for BSD before all is said and done.”

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

This change was made to support usage in industry:

- ▶ Intel was interested in including ACL2 as part of a verification product shipped outside Intel, but GPL made that impossible.
- ▶ Rockwell Collins also expressed interest:

“FWIW, I continue to hope earnestly for BSD before all is said and done.”

... and after the change:

LICENSING

The license is now more permissive: a 3-clause BSD license, formerly GPL.

This change was made to support usage in industry:

- ▶ Intel was interested in including ACL2 as part of a verification product shipped outside Intel, but GPL made that impossible.
- ▶ Rockwell Collins also expressed interest:

“FWIW, I continue to hope earnestly for BSD before all is said and done.”

... and after the change:

“Thank you for changing the ACL2 license to the 3-clause BSD license. It’s a really big deal for us (and our lawyers)!”

CONCLUSION

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971
 - ▶ Application work continues too (e.g., next talk explores x86 modeling)

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971
 - ▶ Application work continues too (e.g., next talk explores x86 modeling)
- ▶ ACL2 is increasingly subjected to heavy use in industry

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971
 - ▶ Application work continues too (e.g., next talk explores x86 modeling)
- ▶ ACL2 is increasingly subjected to heavy use in industry
 - ▶ AMD, Centaur, Oracle, Intel, Kestrel, Rockwell Collins

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971
 - ▶ Application work continues too (e.g., next talk explores x86 modeling)
- ▶ ACL2 is increasingly subjected to heavy use in industry
 - ▶ AMD, Centaur, Oracle, Intel, Kestrel, Rockwell Collins

... as well as in academia and government.

CONCLUSION

- ▶ Research continues in automated reasoning at UT Austin.
 - ▶ Continues the Boyer-Moore program started in 1971
 - ▶ Application work continues too (e.g., next talk explores x86 modeling)
- ▶ ACL2 is increasingly subjected to heavy use in industry
 - ▶ AMD, Centaur, Oracle, Intel, Kestrel, Rockwell Collins... as well as in academia and government.
- ▶ ACL2 has become a reasonably mature tool, but continued use presents continued opportunities for making users happier!