# LEO-II — A Cooperative Higher-Order–First-Order ATP

Christoph Benzmüller, Articulate Software

(currently funded by DFG grant BE 2501/6-1)

# LEO-II — A Cooperative Higher-Order–First-Order ATP

Thanks to:

- 'E'-inside
  (Stephan Schulz)

# LEO-II — A Cooperative Higher-Order–First-Order ATP

Thanks to:

- 'E'-inside (Stephan Schulz)
- 'THF0'-outside (Geoff Sutcliffe)

# LEO-II — A Cooperative Higher-Order–First-Order ATP

Thanks to:

- 'E'-inside
  (Stephan Schulz)
- 'THF0'-outside
  (Geoff Sutcliffe)
- Frank Theiss
  parser, shared term datastructure, indexing

# LEO-II — A Cooperative Higher-Order–First-Order ATP


Thanks to:

- 'E'-inside (Stephan Schulz)
- 'THF0'-outside (Geoff Sutcliffe)
- Frank Theiss

Further thanks to:

- Larry Paulson
- Arnaud Fietzke
- Chad Brown
- Jasmin Blanchette
- ...

# LEO-II — A Cooperative Higher-Order–First-Order ATP


Thanks to:

- 'E'-inside (Stephan Schulz)
- 'THF0'-outside (Geoff Sutcliffe)
- Frank Theiss

Further thanks to:

- Larry Paulson
- Arnaud Fietzke
- Chad Brown
- Jasmin Blanchette
- ...


Key aspects of LEO-II:

extensional HO-RUE-Resolution
extensional HO-pre-unification (depth-bounded)
OTTER like loop
cooperation with FO-ATP (E)
written in OCAML

# LEO-II — A Cooperative Higher-Order–First-Order ATP


Thanks to:

- 'E'-inside (Stephan Schulz)
- 'THF0'-outside (Geoff Sutcliffe)
- Frank Theiss

Further thanks to:

- Larry Paulson
- Arnaud Fietzke
- Chad Brown
- Jasmin Blanchette
- ...


Key aspects of LEO-II:

extensional HO-RUE-Resolution
extensional HO-pre-unification (depth-bounded)
OTTER like loop
cooperation with FO-ATP (E)
written in OCAML

Why did I win?

very simple relevance filtering + parameter scheduling