The Numbers

- 13 Teams participated
- 18 (+2 historical) Solvers entered the main track
  4(+1) in the incremental track
- 32 logics (2 logics with no eligible benchmarks)
  25 logics had participation from more than one team
- 67426 main track competition benchmarks (out of 137648 total)
- 339714 job-pairs executed (+ some repeats)
- ~1 week x 147 nodes of compute time
- 1 new sibling competition (SL-COMP) organized

Record numbers!
• Some initial startup problems, partly bugs, partly user error, but otherwise

StarExec worked great

• Required porting tools to StarExec – thanks Tjark and David

• Thanks to Aaron Stump for prompt help when problems or questions arose

• Continuing to run major jobs with long (10 hour) timeouts to resolve sat/unsat status of unknown benchmarks
## Solver participation – 2014

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NEW</td>
<td></td>
<td>12</td>
<td>12</td>
<td>9</td>
<td>13</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>Abziz...</td>
<td>Cairo U.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boolector</td>
<td>JKU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>CVC/CVCLite/CVC3</td>
<td>NYU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>CVC4</td>
<td>NYU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MathSat-HeavyBV</td>
<td>Trento</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MathSAT 3,4,5</td>
<td>FBK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMTInterpol</td>
<td>U. Freiburg</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SONOLAR</td>
<td>U. Bremen</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STP, simplifyingSTP, STP2</td>
<td>Stanford, MIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4Simp</td>
<td>U. Melbourne</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tiffany de Wintmonte</td>
<td>U. Melbourne</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>opensmt</td>
<td>U. Lugano</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>veriT</td>
<td>UFRN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z3</td>
<td>MSR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AProVE NIA</td>
<td>RWTH Aachen</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MiniSMT</td>
<td>U. Innsbruck</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>test_pmathsat</td>
<td>FBK-IRST</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>barcelogic</td>
<td>UPC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>beaver</td>
<td>UC Berkeley</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>clsat</td>
<td>Washington U.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sateen</td>
<td>U. Col.-Boulder</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sword</td>
<td>U. Bremen</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yices</td>
<td>SRI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alt-Ergo</td>
<td>CNRS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ArgoLib</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fx7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ario</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ExtSat</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jat</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NuSMV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sammy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Benchmarks & Logics

- Many new benchmarks added
  - 137648 main track benchmarks in 34 divisions but 35202 are easy and 35020 are unknown, leaving 67426 for competition
  - 9925 benchmarks for incremental track in 8 divisions

- Thanks to many contributors

- Thanks to Morgan Deters, Clark Barrett for curation and uploading
  - Checked and reclassified the benchmarks, resulting in the expansion to 34 divisions
Incremental track

- Sorry, *data not yet reduced*…
  - *But it will be*
Parallel vs. Sequential

- Emphasized sequential timing since we weren’t sure that solvers were implemented or tuned for parallel solving.

- This is question for future competition design.
Teams & Solvers

Some first-time participants:

- Hristina Palikareva, Cristian Cadar: Kleaver-STP, Kleaver-portfolio (QF_ABV)
- Tung Vu Xuan: raSAT (QF_NRA)
- Mate Soos: STP-Crypto-MiniSat4 (QF_BV) second place in QF_BV
Kleaver

The constraint solver of the symbolic execution engine KLEE:

High-Level Optimizations

- Mathematical simplifications
- Expression canonicalization
- Grouping constraints into independent subsets
- Cache, which exploits subset/superset relations among constraint sets to determine satisfiability of subsequent queries
Teams & Solvers

Some relatively new participants or returning after a few years:

- M. A. Abziz: two portfolio solvers (QF_BV)
- Carsten Fuhs: AProVE (QF_NIA)
Teams & Solvers

Other regulars in single divisions:

- T. Hansen: 4Simp (QF_BV)
- Antti Hyvarinen: OpenSMT2 (QF_UF)
- Florian Lapschies: SONOLAR (QF_BV, QF_ABV)
OpenSMT2

- OpenSMT is a GPL-licensed SMT solver
- The development is coordinated at the University of Lugano in Switzerland
- Version 2 has been under development since summer 2012
  - Native support for the SMTLIB2 standard
  - Separation of the abstract term dag from the theory related representations (such as EUF terms)
  - Compact representation and efficient memory management for the data types including Enodes
  - Currently support for QF_UF (but more is to come)

We are looking for a person interested in doing a PhD on a project related to parallelized SMT solving!
Teams & Solvers

Other regulars in single divisions:

- A. Biere, et al.: Boolector (QF_BV)
  
  winner QF_BV

Boolector (2 variations) (QF_ABV)

winner QF_ABV
Boolector at the SMTCOMP’14
Aina Niemetz, Mathias Preiner, Armin Biere

Major changes since SMTCOMP’12:
• new lemmas on demand (LOD) engine: array operations and arrays as lambda terms and uninterpreted functions
• don’t care reasoning to speed up LOD
  • Boolector (justification)
  • Boolector (dual propagation)

Further improvements:
• support for SMT-LIB v2 macros (define-fun)
• new model generation algorithm (fixes performance drop of older versions)
• internal model validation for satisfiable instances
• cloning support (cf. cloning in Lingeling)
• API call tracing (record/replay sequences that trigger erroneous behavior)
• model-based testing
• fixes in both rewrite engine and the incremental API
• fixed and reenabled the previously disabled unconstrained optimization
Entrants in many divisions:

- Clark Barrett, Morgan Deters: (all 32 divisions)
  CVC4 – *winner in 14 divisions*
  CVC3 – *winner in 3 divisions*

- Pascal Fontaine, David Deharbe: (17 divisions)
  veriT – *winner in UFLRA*

- Bruno Dutertre: (15 divisions)
  Yices2 – *winner in 10 divisions*
  *(back after a few years’ absence)*

- Jochen Hoenicke, Jürgen Christ: (8 divisions)
  SMTInterpol – *winner in QF_LIA*
CVC4 (NYU and U Iowa)

Clark Barrett (NYU)  Cesare Tinelli (U Iowa)

Theories
Arithmetic, Arrays, Bit-vectors, Inductive Data Types, Quantifiers, Sets, Strings, Uninterpreted Functions

Features
Models, Proofs, Open-Source, BSD License, Portfolio mode, Variety of API’s

Performance in SMT-COMP (all divisions, after bug-fix)
Top solver in 9 divisions
(AUFLIA, AUFNIRA, LRA, QF_AUFBV, QF_LIA, QF_LRA, QF_UFNIA, UF, UFLIA)
Overall score (all divisions): 65.56 (Z3: 73.97)
Excluding non-linear: 55.57 (Z3: 54.82)
The veriT solver

http://www.veriT-solver.org

David Déharbe, Pablo Federico Dobal and Pascal Fontaine
Loria, INRIA, Université de Lorraine (France) and UFRN (Brazil)

What is new:
- improved efficiency on UF and LRA (still space for improvement)
- stabilized on many categories
- To do: LIA, better combinations, better quantifiers

Goals:
- UF, LIA, LRA, NRA (Redlog), NIA, combinations and quantifiers
- for verification platforms B, TLA+
- Proofs!
Further Thoughts

**Solvers:**
- First-time entrants had some trouble with system configurations – getting a static build of a tool and getting it to work on StarExec
- Two entrants dropped out after expressing initial intention

**Benchmarks:**
- Still need more benchmarks; some divisions have relatively few

**Competition:**
- StarExec allowed us to run all eligible benchmarks
- Continuing to run jobs to resolve unknown benchmarks
- Revise scoring – more emphasis on timing?
- Parallel or sequential?
- Better support needed for incremental benchmarks
- Separate measure of performance on quick jobs?

**Teams:**
- Congratulations on your accomplishments
- Thanks for your participation
SL-COMP’14

Competition of solvers for Separation Logic
Input Theory

Separation Logic [O’Hearn, Reynolds et al. CSL’01, LICS’02]

fragment of
Symbolic Heaps with Recursive Definitions

\[ \Phi ::= \prod \land \Sigma \]
\[ \prod ::= \text{X}=\text{Y} \mid \text{X} \neq \text{Y} \mid \prod \land \prod \]
\[ \Sigma ::= \text{emp} \mid \text{X} \mapsto \{(f_0, Y_0), \ldots\} \mid \Sigma \star \Sigma \mid \text{P}(Y_0, \ldots) \]

\[ \text{P}(E, \ldots) \iff Z_0. \prod_0 \land \Sigma_0 \lor \ldots \lor \exists \forall Z_k. \prod_k \land \Sigma_k \]
Input Theory

Separation Logic [O’Hearn, Reynolds et al. CSL’01, LICS’02]

fragment of

Symbolic Heaps with Recursive Definitions
Benchmarks

- **Problems:**
  - Checking satisfiability 25%
  - Checking entailment validity 75%

- **Kind of recursive definitions (division):**
  - acyclic singly linked lists (ls)  sll0a  59%
  - fixed (nll, dll, skl, …)  FDB  6%
  - user-defined  UDB  35%

- **Origin:**
  - crafted  41%
  - random  59%
Competition Rules

- Input format in SMTLIBv2
  - theory QF_S
  - semantics discussed in smtcomp14-sl@googlegroups
  - benchmarks available in github project smtcomp14-sl

- Use of pre-processors for some solvers

- No scrambling of benchmark problems

- Solvers running on Star-Exec

- Same score computation as in SMT-COMP’14
Solvers

- **Asterix** *(TUM and MPI, Germany and UCL, UK)*
  - J. Navarro Perez and A. Rybalchenko

- **Cyclist-SL** *(UCL, UK)*
  - J. Brotherston, N. Gorogiannis, and R. L. Petersen

- **SLEEK** *(NUS, Singapore)*
  - Q.L. Le and W.N. Chin

- **SLIDE** *(Verimag, France and VeriFIT, Czech Rep.)*
  - A. Rogalewicz, R. Iosif, and T. Vojnar

- **SLSAT** *(UCL, UK)*
  - J. Brotherston, C. Fuhs, N. Gorogiannis, and J. Navarro Perez

- **SPEN** *(LIAFA, France and VeriFIT, Czech Rep.)*
  - C. Enea, O. Lengal, M. Sighireanu, and T. Vojnar
### Results

<table>
<thead>
<tr>
<th></th>
<th>SII0a(sat)</th>
<th>SII0a(=&gt;)</th>
<th>FDB(=&gt;)</th>
<th>UDB(sat)</th>
<th>UDB(=&gt;)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Asterix</strong></td>
<td>1</td>
<td>1</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td><strong>Cyclist-SL</strong></td>
<td>--</td>
<td>4</td>
<td>2</td>
<td>--</td>
<td>1</td>
</tr>
<tr>
<td><strong>SLEEK</strong></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td><strong>SLIDE</strong></td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>2</td>
</tr>
<tr>
<td><strong>SLSAT</strong></td>
<td>4</td>
<td>--</td>
<td>--</td>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td><strong>SPEN</strong></td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

Future Work

- Re-defining the SL theory in SMTLIBv2
- Including more benchmarks
  - in existing divisions
  - more divisions, e.g., SL + AI
  - from program analysis and verification tools
- Other problems
  - sat witness
  - (bi-)abduction
Thanks

- David Cok

- Clark Barrett and Cesare Tinelli

- Solver providers: Nikos Gorogiannis, Ondrej Lengal, Le Quang Loc, Juan Navarro Perez, Chin Wei Ngan, Adam Rogalewicz, Radu Iosif, Andrey Rybalchenko, Tomas Vojnar, Constantin Enea.

- Group list support: Josh Berdine, Thomas Wies, Christoph Haase.