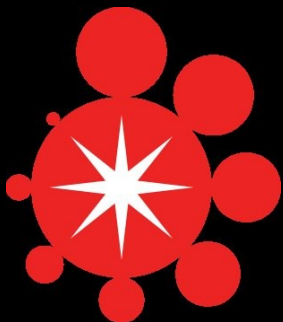


StarExec

Design Abstractions, Demonstration

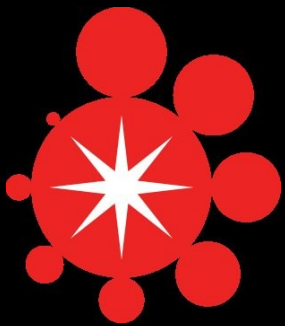
Ben McCune, **Aaron Stump**, Cesare Tinelli
CS, The University of Iowa

Geoff Sutcliffe
CS, University of Miami



StarExec: Shared Logic-Solving Infrastructure

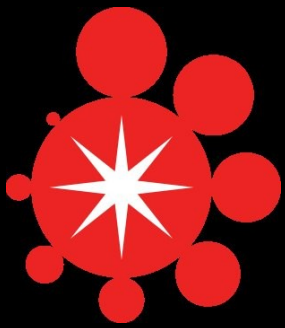
- **Web service** for
 - Hosting benchmark libraries
 - Running competitions
 - Evaluating solvers
- Users can
 - Upload solvers, benchmarks
 - Run jobs on **compute cluster** (200 nodes planned)



In this talk

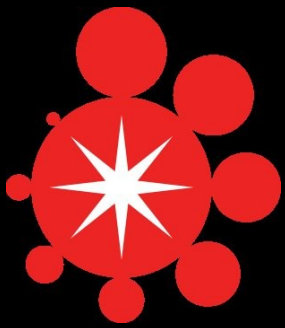
- Design abstractions in StarExec
- Demo

<http://www.starexec.org>



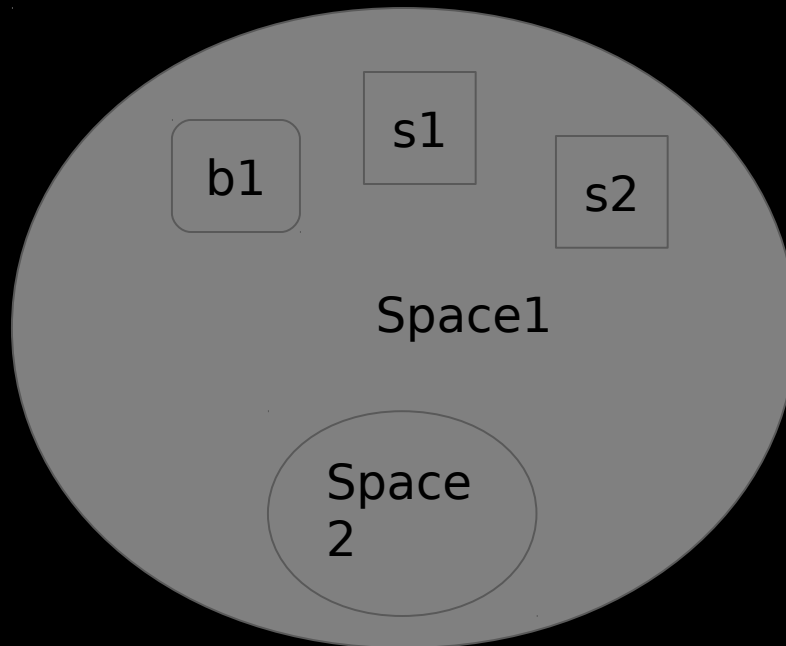
Primitives

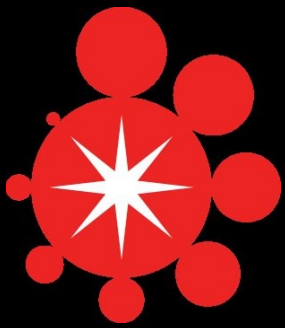
- Benchmarks
- Solvers
- Jobs
- Users



Spaces

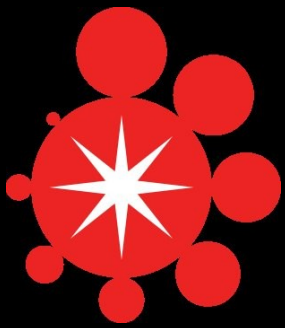
Contain primitives, subspaces





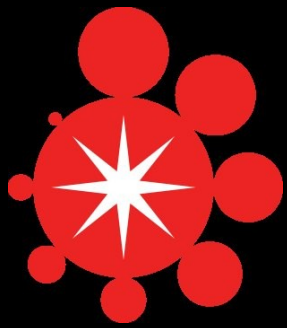
Communities

- For different kinds of logic solvers
 - TPTP, SMT, SAT, etc.
- Communities are special spaces
- Other spaces are subspaces
- New community members get a private subspace



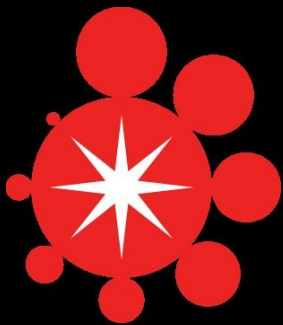
Permissions

- Add and remove
- For spaces and primitives
- **Space Leaders** control access



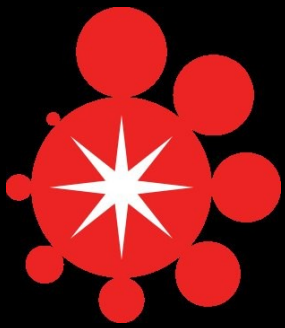
Community Leadership

- Approve new members
- Provide benchmark validators, job post-processors
- Set community defaults for jobs



Benchmarks

- Upload via a .zip archive
- Creates parallel space structure
- Benchmarks validated on upload
- Validators can extract attributes
 - key-value pairs
 - e.g., subvariety of logic, expected result (sat/unsat), other characteristics

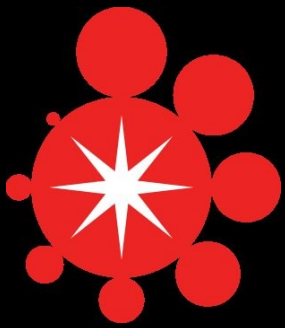


Solvers

- Can have multiple configuration scripts
- Configurations invoke actual solver

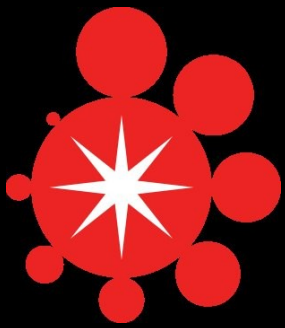
```
#!/bin/bash
```

```
./z3 -smt2 $1
```



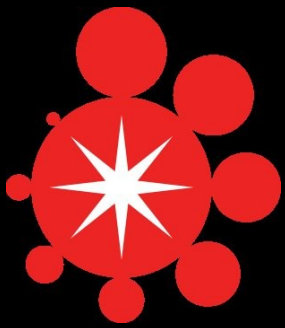
Running a Job

- Jobs initiated within spaces
- Job-pair = 1 solver on 1 benchmark
- Multiple ways to collect job-pairs from space hierarchy
- Job-pairs dispatched to compute nodes
- Results collected incrementally



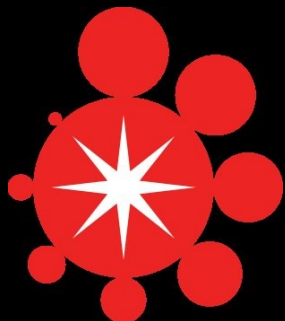
Job Results

- Solver output post-processed for attributes
- Job results can be downloaded
 - .zip files, Excel spreadsheets
 - All outputs from solvers, or
 - All attributes collected



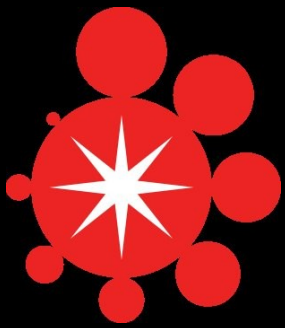
System Design

- **Head nodes**
 - Run web service
 - Send jobs to compute nodes
- **Compute nodes**
 - execute job-pairs
- Networked storage



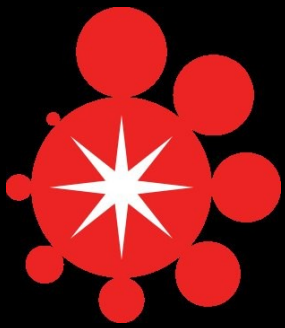
Compute Nodes

- 32 Hewlett-Packard SL230s with:
 - 2 Intel 2.4GHz quad-core processors
 - 128 GB RAM
 - 1TB local disk
- Funds for around 150-200 nodes total



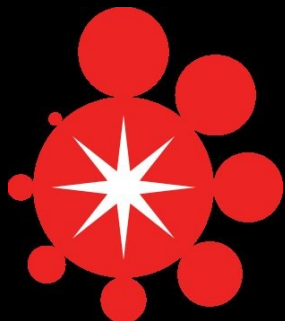
Technologies

- Front end: JSP, Javascript/jQuery
- Backend: Java, MYSQL
- Apache Tomcat as web server and servlet container
- Oracle GridEngine to schedule job-pairs.



Status

- First hardware purchase
 - 3 head nodes, 32 compute nodes
 - NetApp storage (mirrored 22TB)
- Software almost ready
- Public release this fall
- Demo today on dev cluster



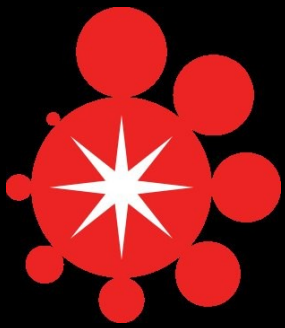
Acknowledgments

Support

- The **National Science Foundation**
- The University of Iowa

Development team (past and present)

- **Benton McCune, Tyler Jensen**
- Todd Elvers, Clifton Palmer, Vivek Sardeshmukh, Skylar Stark, Ruoyu Zhang
- JJ Urich, Hugh Brown (sys admin)



Demonstration

<http://www.starexec.org>