The Stack Trace Analysis Tool
Enabling Million-way Debugging

Extensive-Scale Debugging Challenges
- Many control channels
- Large data volumes
- Excessive data analysis overhead
- Scalable results presentation

STAT Incrementally Debugs To Narrow Down the Search Space
- Spatial stack traces across tasks
- Temporal stack traces over time
- Traces gathered with varying level of detail
- Function name, source line, PC

STAT Successfully Identified Bug at 1 Million MPI Tasks on Sequoia
- Identifies processes with similar traces
- Equivalence classes color encoded
- Representative subset fed into traditional debugger for root cause analysis

MRNet and Efficient Data Structures Enable Scalable Analysis
- Uses Rose compiler for static analysis of code
- Identify loops and loop ordering variables
- Uses StackwalkerAPI for runtime information
- Gather stack traces with source file name and line number
- Extract program variables for loop ordering

Temporal Ordering Analysis Identifies the Root Cause of Hangs
- Runs on Linux Clusters, IBM BlueGene systems, and Cray platforms.
- http://www.paradyn.org/STAT/STAT.html
- Source available at https://outreach.scidac.gov/projects/stat/

Additional Information
- This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344.
- Barton P. Miller
  University of Wisconsin
- Dorian C. Arnold
  University of New Mexico
- Dong H. Ahn, Bronis R. de Supinski, Gregory L. Lee, Matthew LeGendre, Martin Schulz
  Lawrence Livermore National Laboratory

Single outlier determined to be faulty hardware