Testing Concurrency Runtime via a Stochastic Stress Framework

Atilla Gunal

Technical Computing Group

Microsoft

Agenda

- The Product
  - Concurrency Runtime
- The Problem
  - Testing concurrent software
- The Model
  - A stochastic stress framework
- The Results
  - Bugs found
  - Effectiveness of framework
  - Comparison with similar tools

The Product:
Concurrency Runtime

Concurrency Runtime

- Runtime for parallel execution
  - Parallel_for(1, 10, Foo)
- Dynamic
  - Composed of interacting components
- Performance centric
- Ocean of race conditions

The Problem:
Testing Concurrent Software

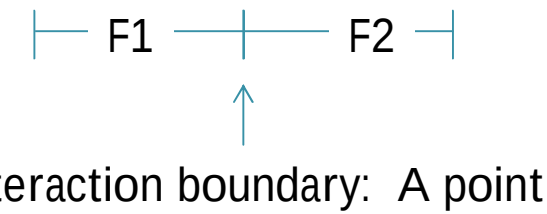
Problem: State Space Explosion

Number of Functionalities	Enumeration of Interactions	Number of Interactions
1	F1	1
2	F1, F2, F1F2, F2F1	4
3	F1, F2, F3, F1F2, F1F3, …	15
…	…	…
N	F1, F2, …, FN, F1F2, F1F3, … F1F2F3, F1F2F4, … … F1F2F3…FN, F2F1F3…FN, …	N + Nx(N-1) + Nx(N-1)x(N-2) + … N!

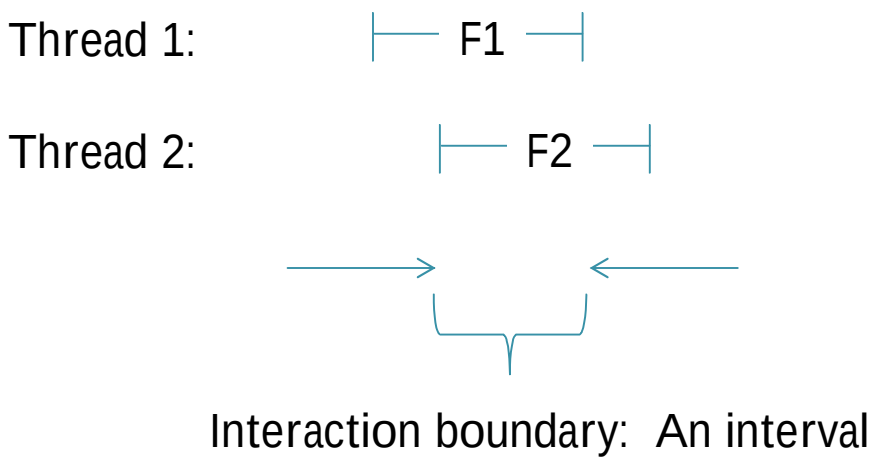
It's hard to test software!

Problem: Thread Interactions

- Interaction boundary for serial software



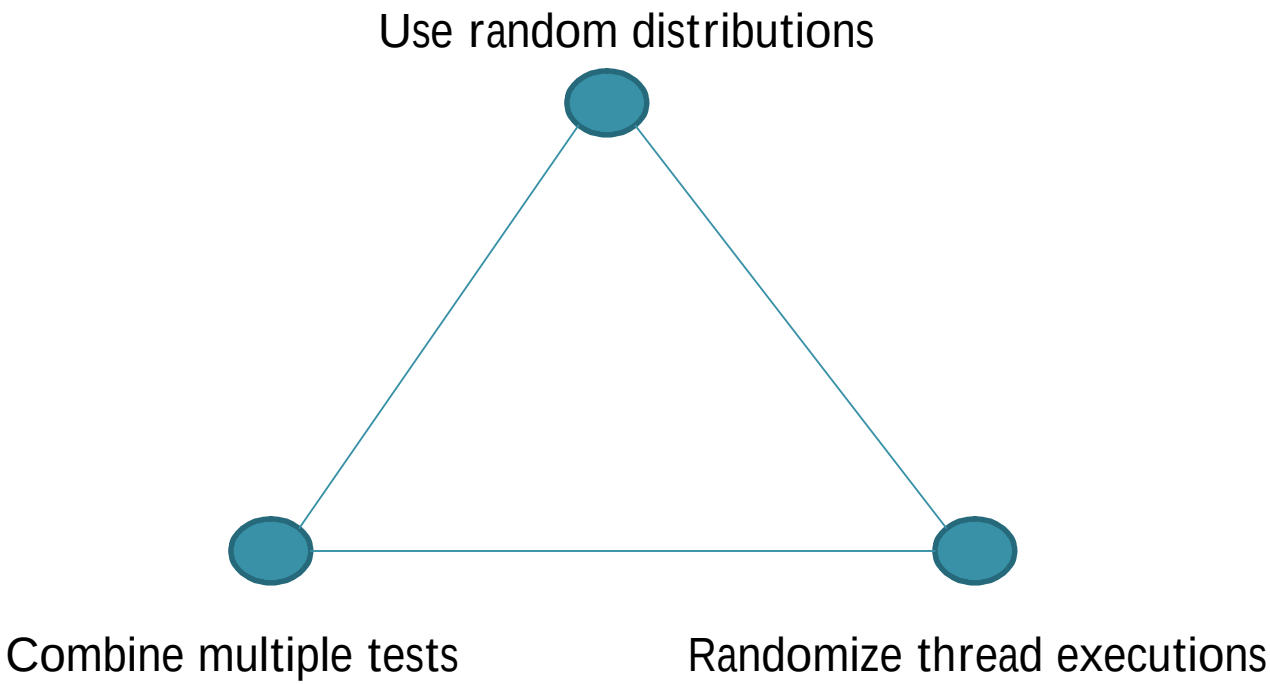
- Interaction boundary for concurrent software



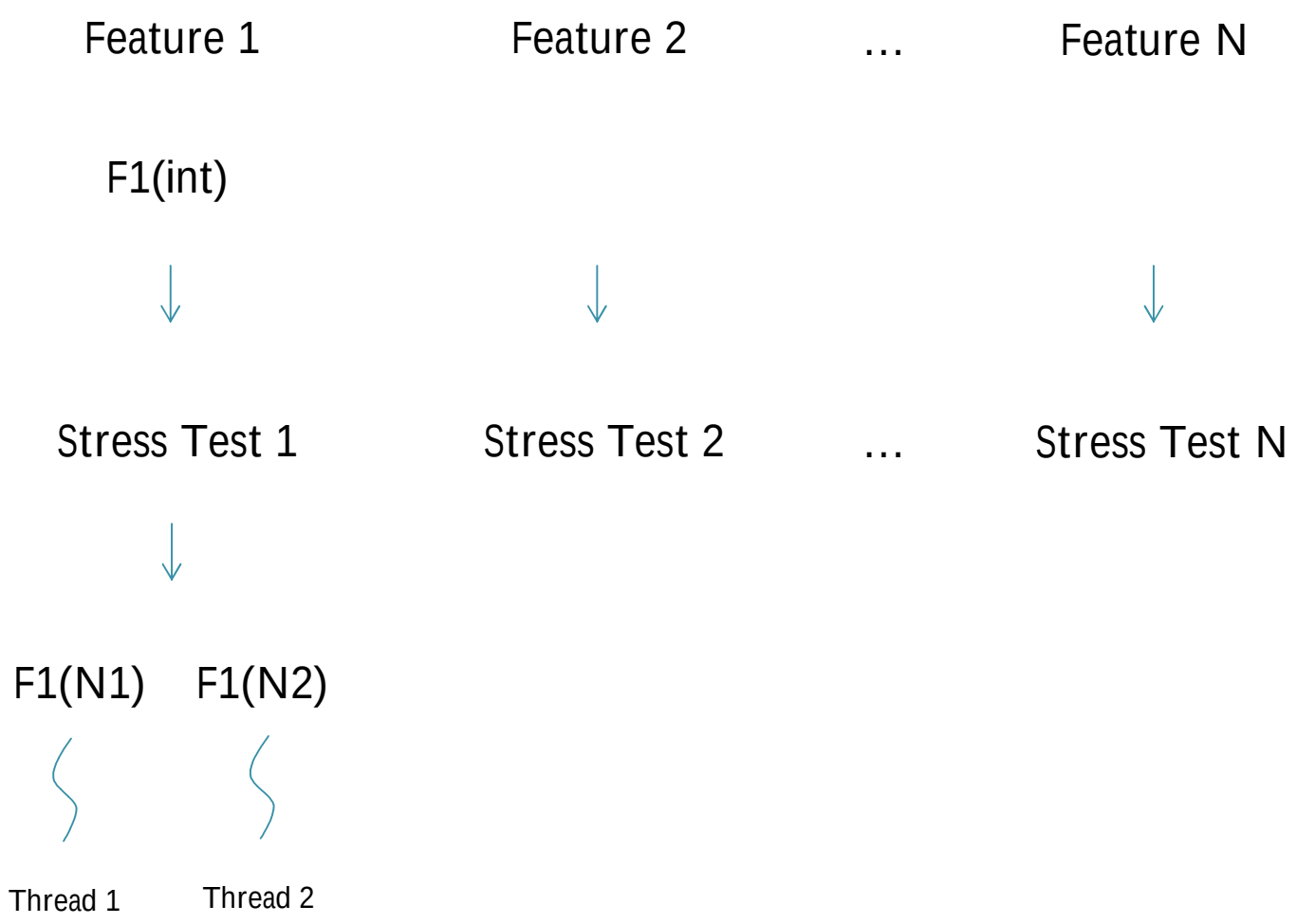
It's even harder to test concurrent software!

The Model:
Stochastic Stress Framework

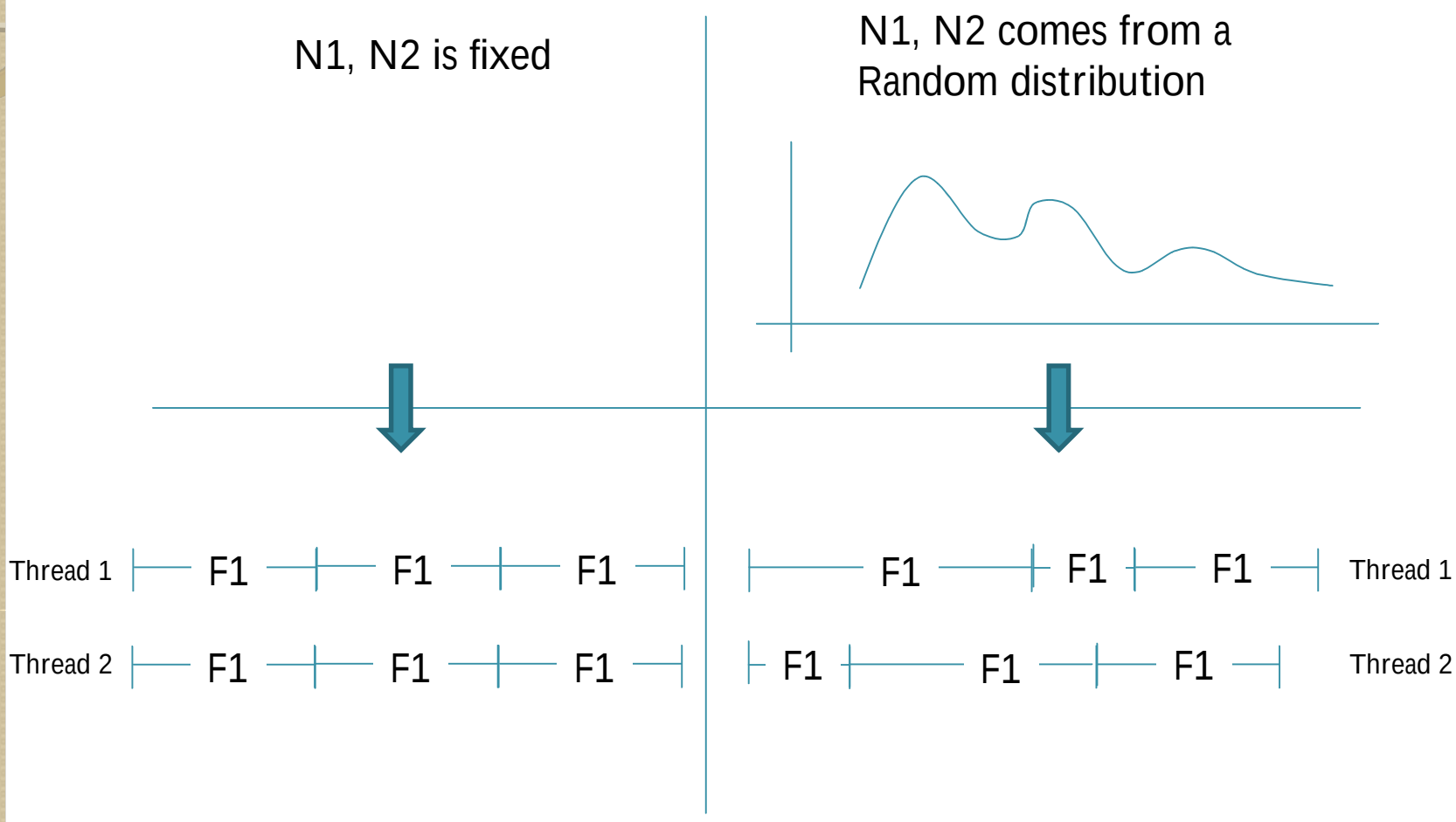
Key Abilities of the Model



Ability 1: Use Random Distributions

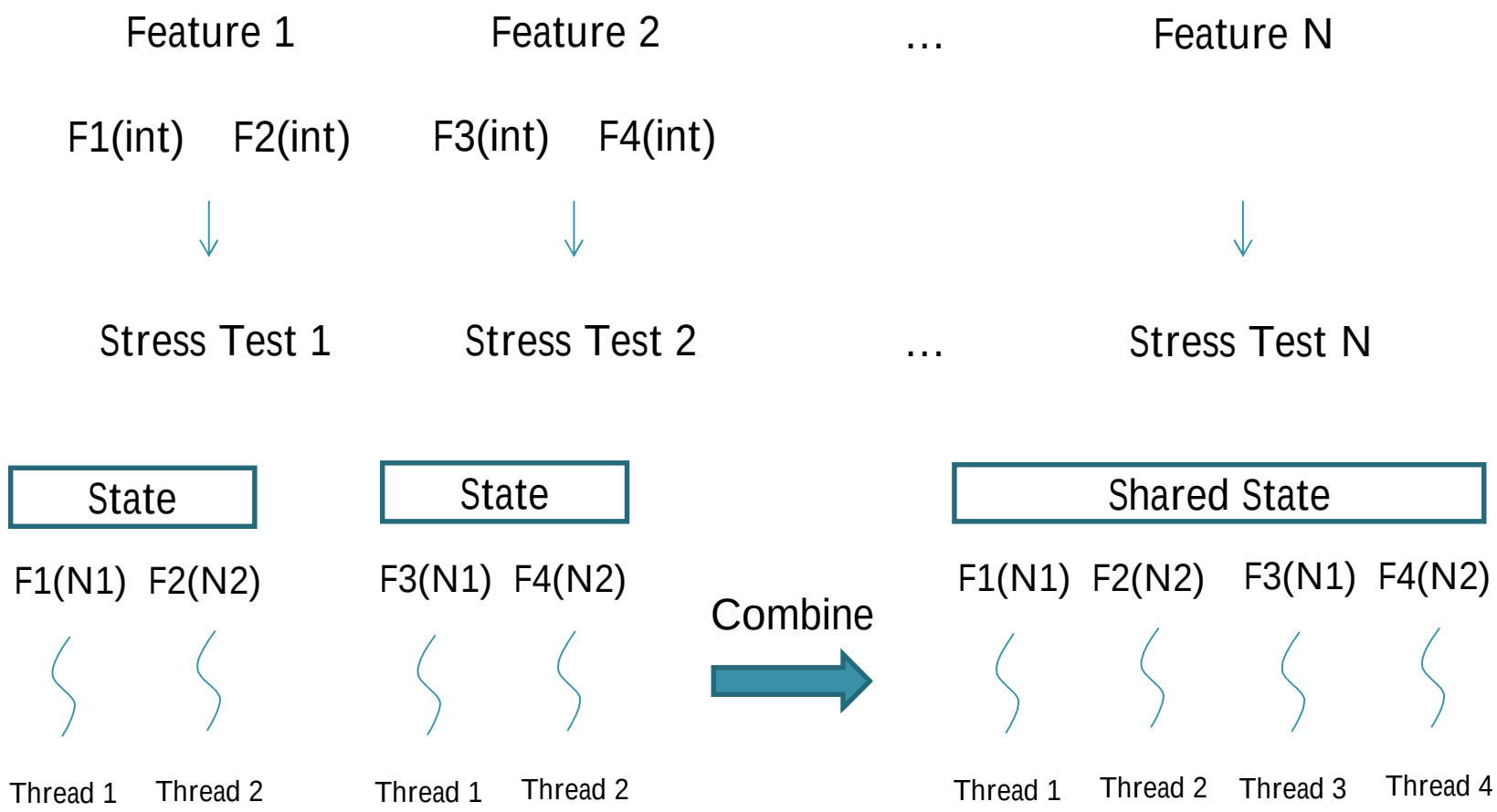


Use Random Distributions



Improves thread interactions!

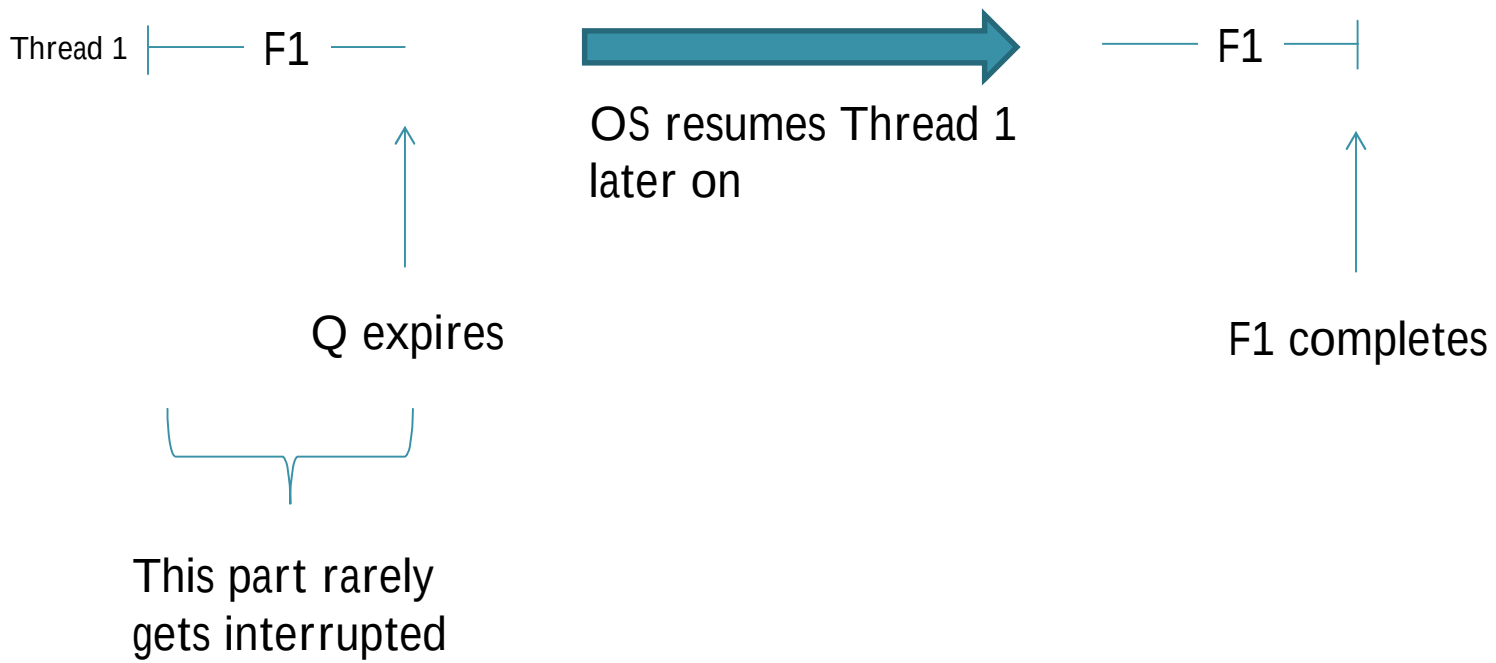
Ability 2: Combine Multiple Tests



Copes with state space explosion!

Ability 3: Randomize Thread Executions

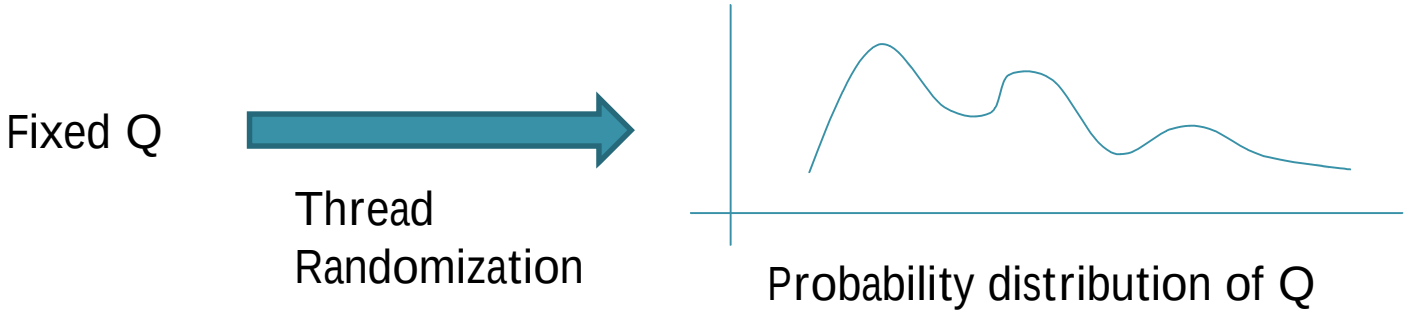
- Reason for thread interruption
  - Thread blocks
  - Thread quantum (Q) expires



Randomize Thread Executions

- Pseudo code of the tool

```
Pick a random thread
Suspend that thread
Sleep for some random time
Resume thread
```



Improves thread interactions!

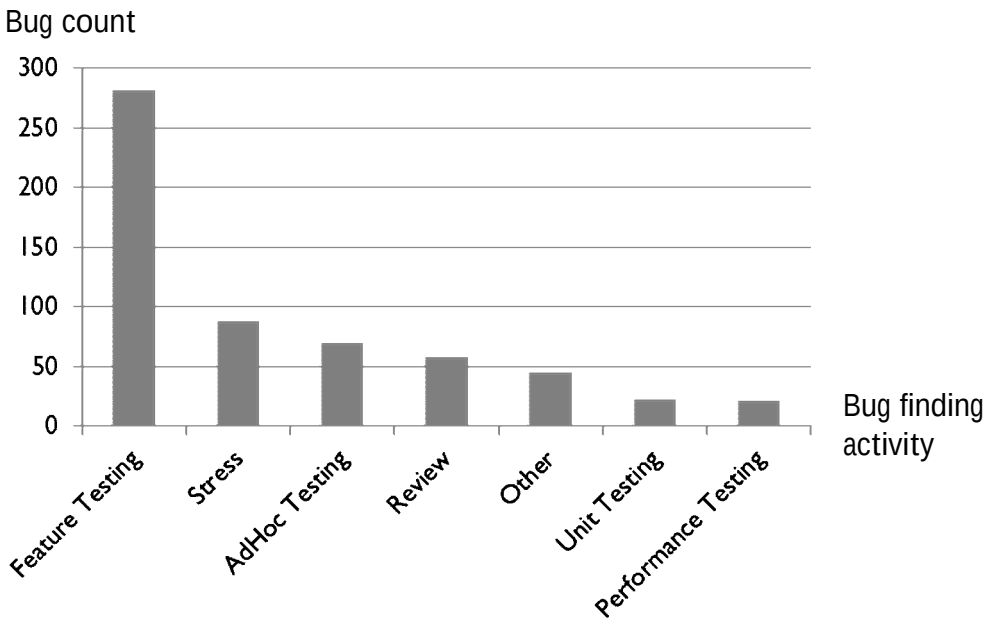
Summary of Problems and Abilities

Problems	Abilities
State space explosion	Combine multiple tests
Interactions between threads	Use random distributions Apply thread randomization

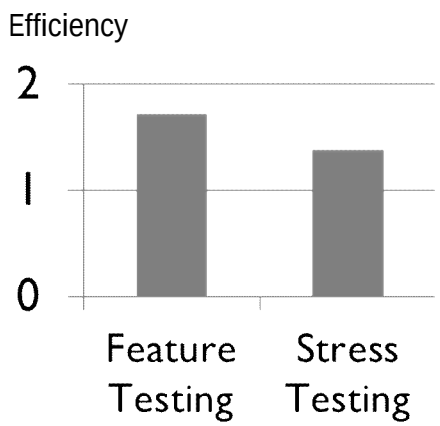
The Results

Efficiency of The Stress Tests

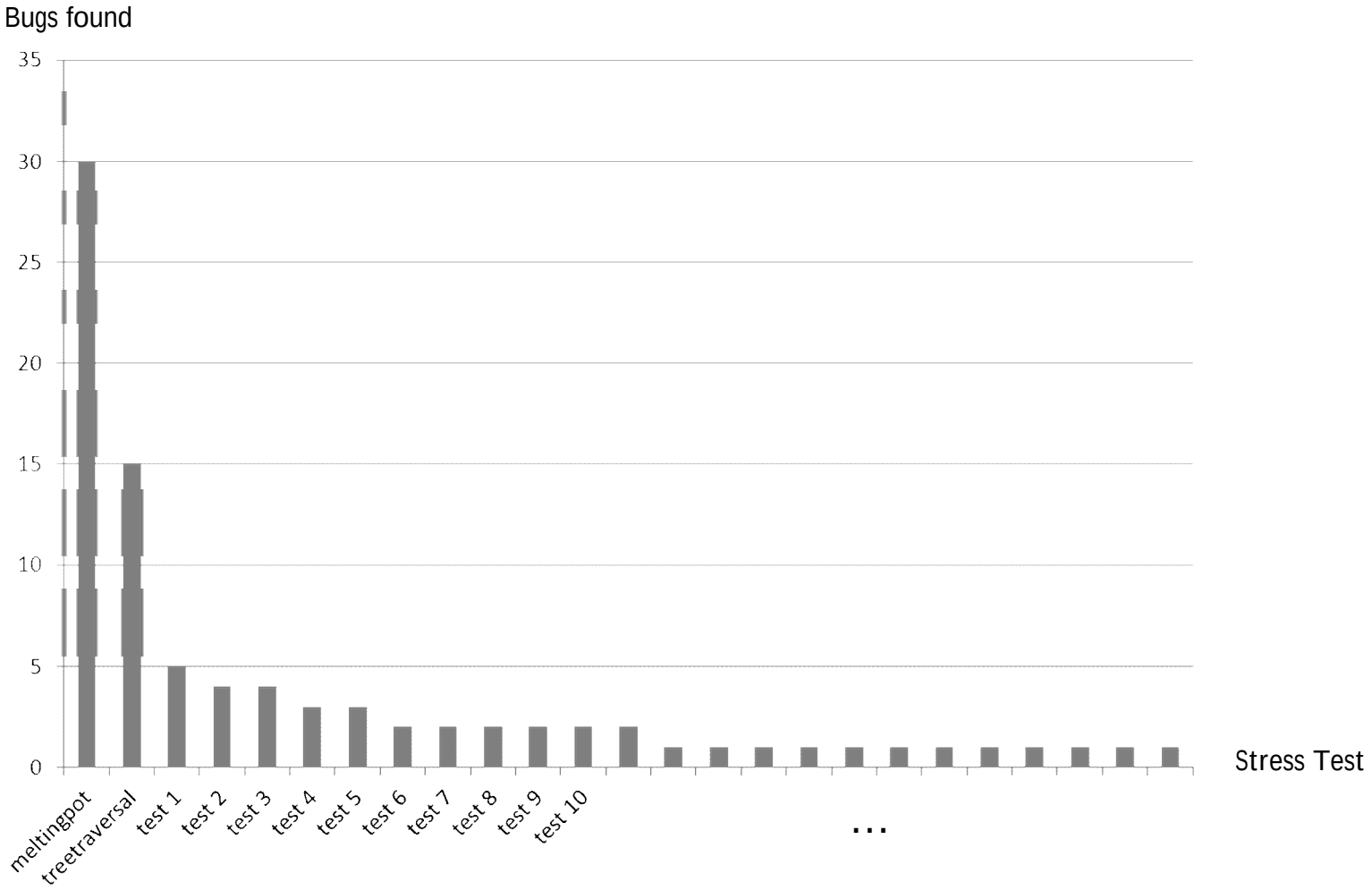
$$Efficiency = \frac{Output}{Input} = \frac{Bugs\ found}{Lines\ of\ test\ code}$$



Bug finding activity	Number of Bugs Found	1000 Lines of code
Feature Testing	282	163
Stress	88	63



Effectiveness of stress tests



Comparison

- Repetitive Functional Execution
  - Limited Randomization
  - Limited shared runtime state
- Cuzz
  - Smarter thread randomization tool
  - Running stress under Cuzz – no issues found

Conclusion

- Key takeaways to cope with concurrency
  - Combine multiple tests
  - Apply random distributions
  - Randomize thread executions

Questions?