SATE VI
Ockham Sound Analysis Criteria

31 May 2017

https://samate.nist.gov/SATE6OckhamCriteria.html

William of Ockham
Source: Wikipedia

Certain trade names and company products are mentioned in the text or identified. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology (NIST), nor does it imply that the products are necessarily the best available for the purpose.

“… program testing can be a very effective way to show the presence of bugs, but is hopelessly inadequate for showing their absence”

– Edsger W. Dijkstra

“The Humble Programmer”,
CACM, 15(10):864, October 1972.

What is “Sound”?

- Informally, we use “sound” to mean absolutely correct reasoning, in contrast with heuristics.
- That is, you can depend on the tool’s *findings*.

Why Have An Ockham Track?

- Sound analyzers only handle moderate-sized programs and a few weaknesses.
- *But all findings are correct – you can depend on them.*
- The Ockham Criteria highlights the strengths of sound analyzers.

The Criteria

1. The tool is claimed to be sound.
2. For at least one weakness class and one test case the tool produces findings for a minimum of 75% of buggy sites OR of non-buggy sites.
3. Even one incorrect finding disqualifies a tool for this SATE.

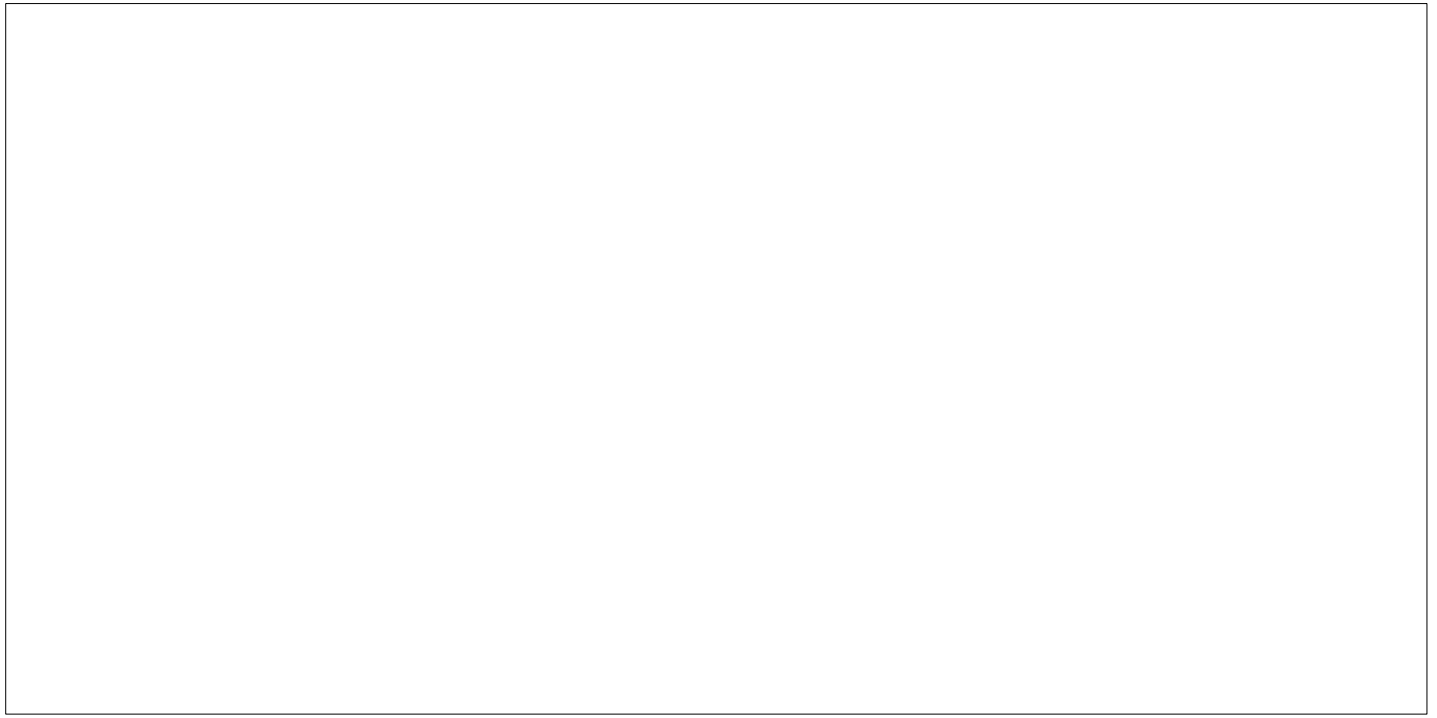
PROPOSED

Some Details of the Criteria

- No manual editing of the tool output is allowed. No automated filtering specialized to a test case or to SATE is allowed, either.
- A finding is a definitive report. In other words, that the site has a specific weakness (is buggy) or that the site does not have a specific weakness (is not buggy).
- Sound means every finding is correct.
- A tool may have optional settings that cause unsound analysis.
- The tool defines what is a class of weaknesses and what are sites.
- A test case may be one of the large programs. Small synthetic test cases may be grouped to reach the threshold.
- There must be a minimum of 100 appropriate sites in the test case.
- We will determine that findings are correct (or incorrect) by simple programs.
- All reasoning is based on models, assumptions, definitions, etc. (collectively, "models"). If there are unexpected findings, we (the tool maker and the Ockham committee) will decide if they result from a reasonable model difference or whether they are incorrect. To satisfy the SATE VI Ockham Criteria, any such differences must be publicly reported.

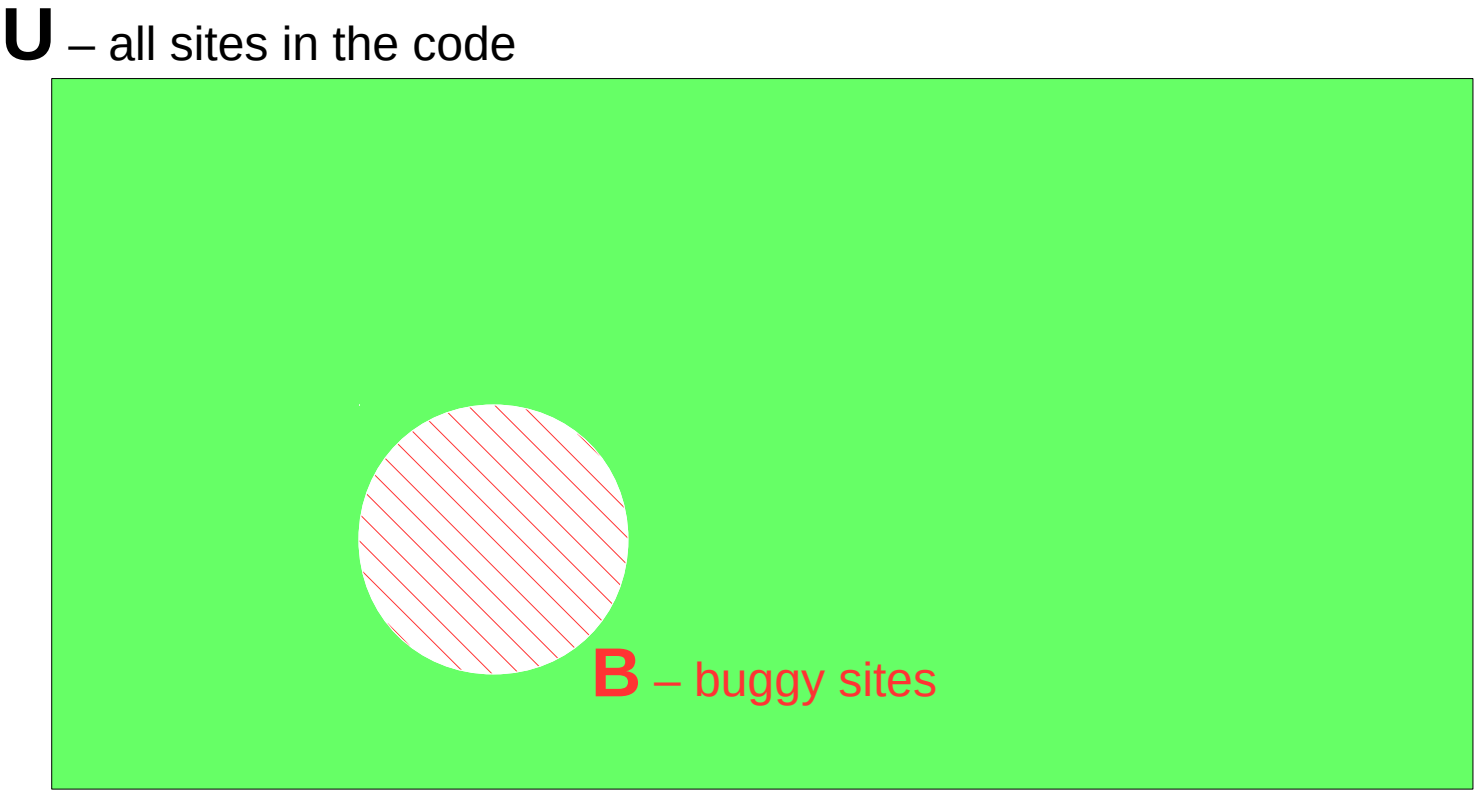
What is a “Finding”?

U – all sites in the code



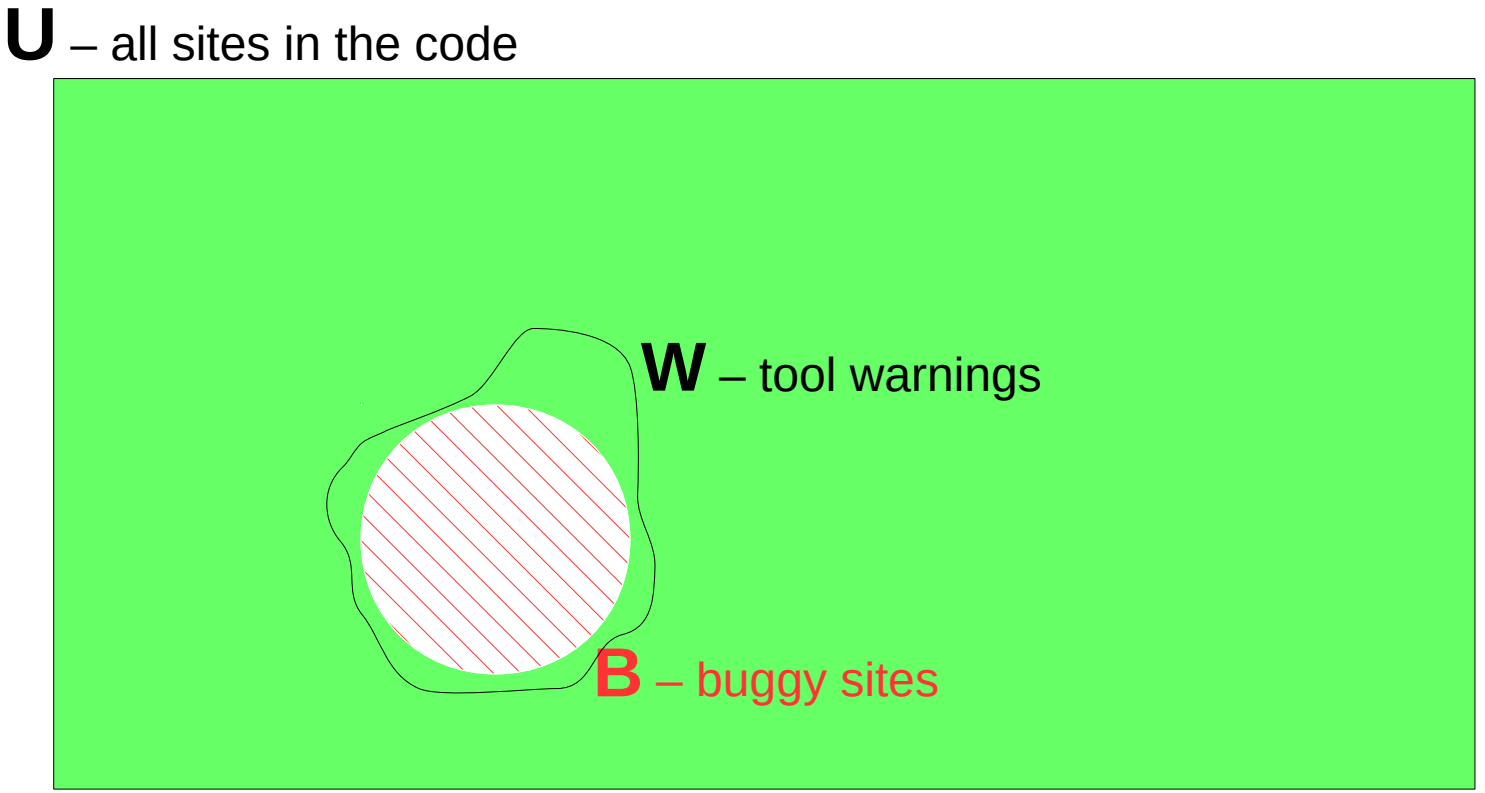
What is a “Finding”?

- The code has some bugs and some good code.



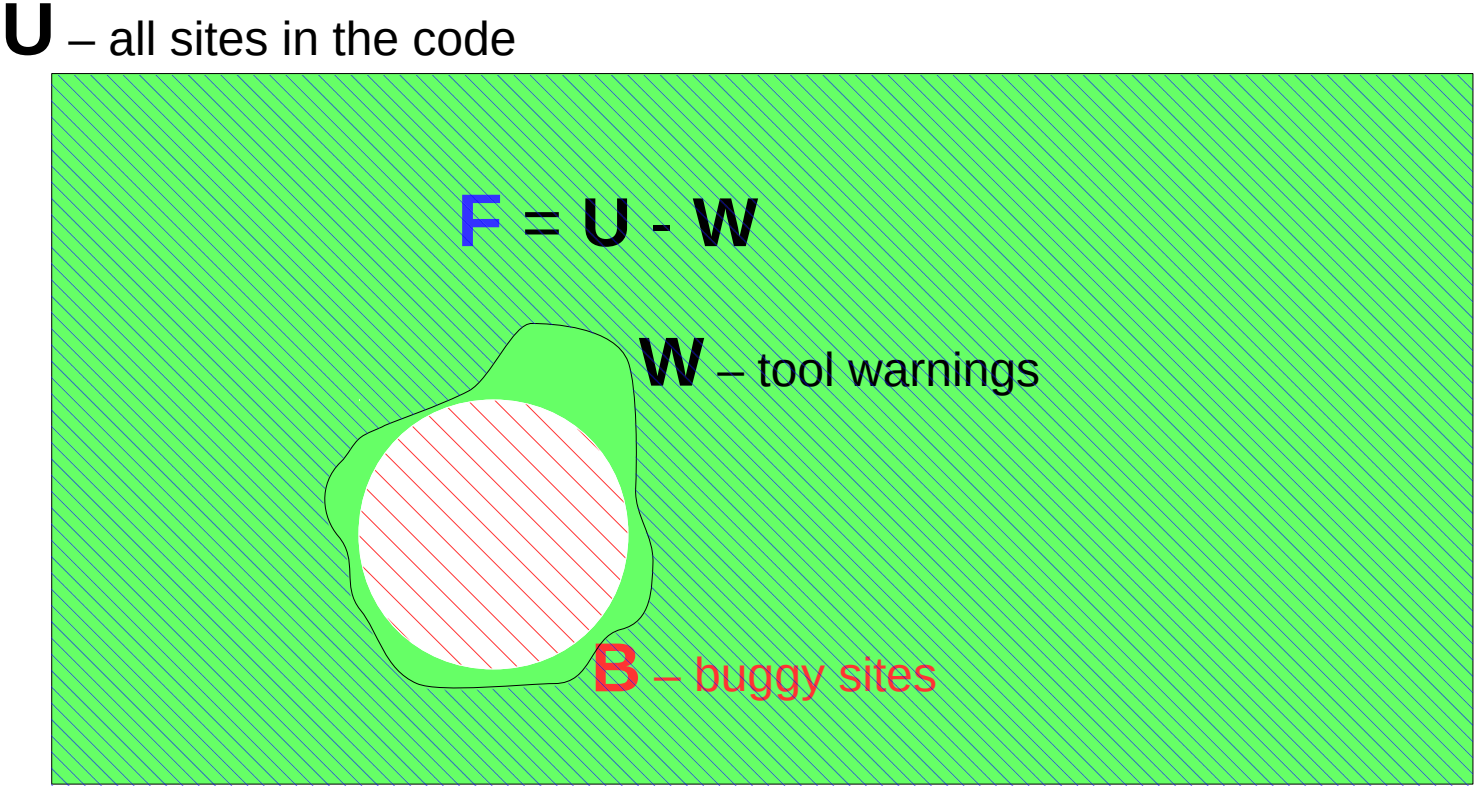
What is a “Finding”?

- The tool’s warnings may overapproximate, because it must summarize program states. Any location *without* a warning is definitely fine.



What is a “Finding”?

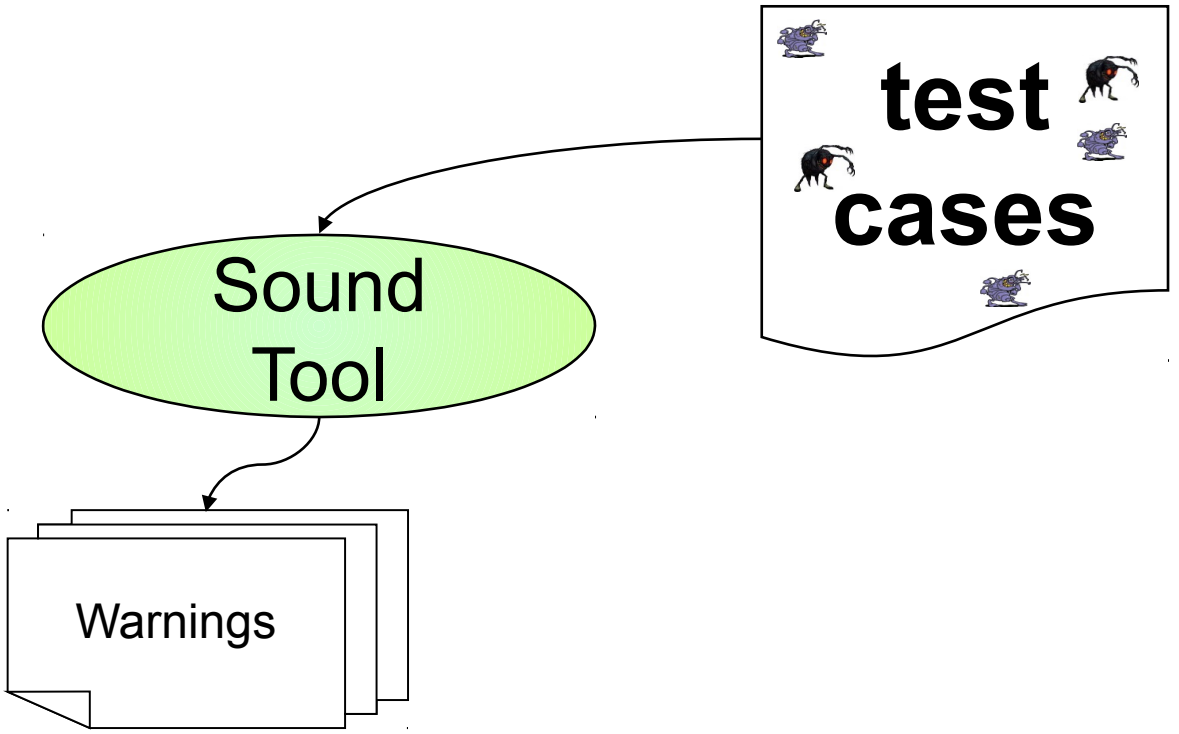
- The tool’s *findings* are sites without warnings.



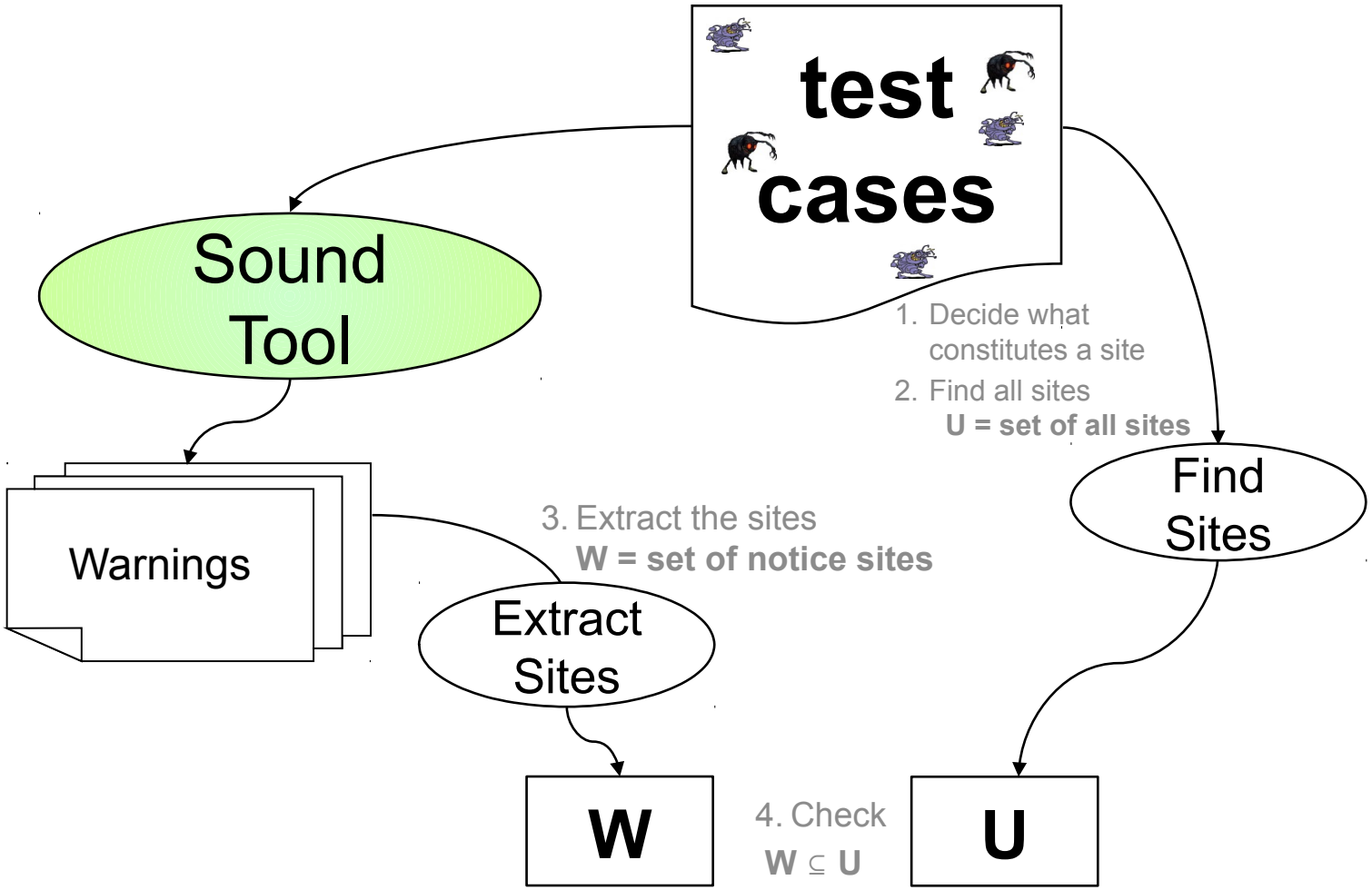
What is a “Site”?

- A site is a location in code that a fault may occur.
  - *In other words, places to check for this bug*.
- Examples:
- Buffer overflow (BOF) sites are:
  - use of [ ] operator
  - use of unary * (dereference) operator with arrays
  - use of string library functions, such as strcpy() or strcat().
- Integer underflow sites are:
  - use of -- or binary -
  - use of * (multiply)

Flow Overview

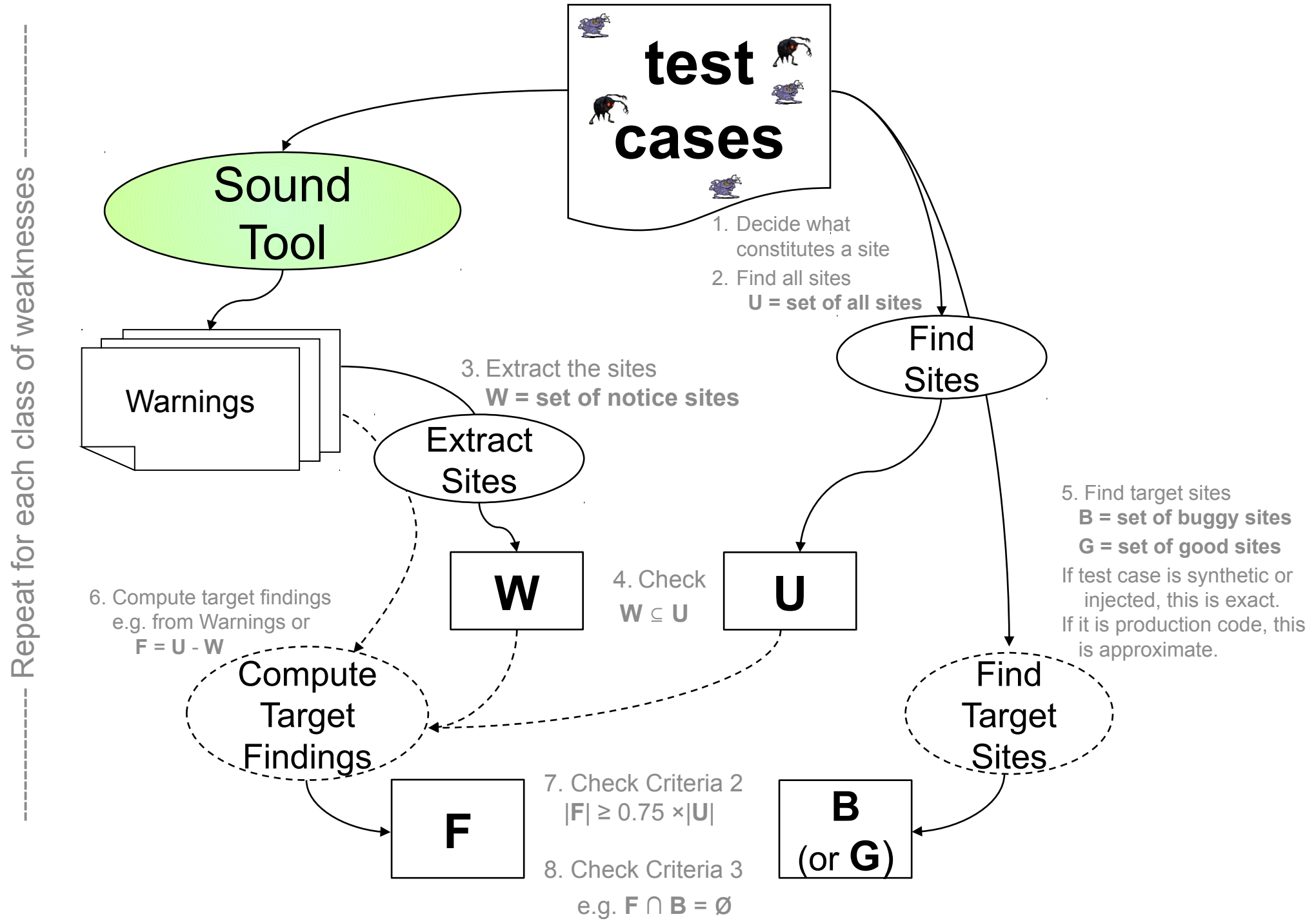


Flow Overview



---------- Repeat for each class of weaknesses ----------

Flow Overview

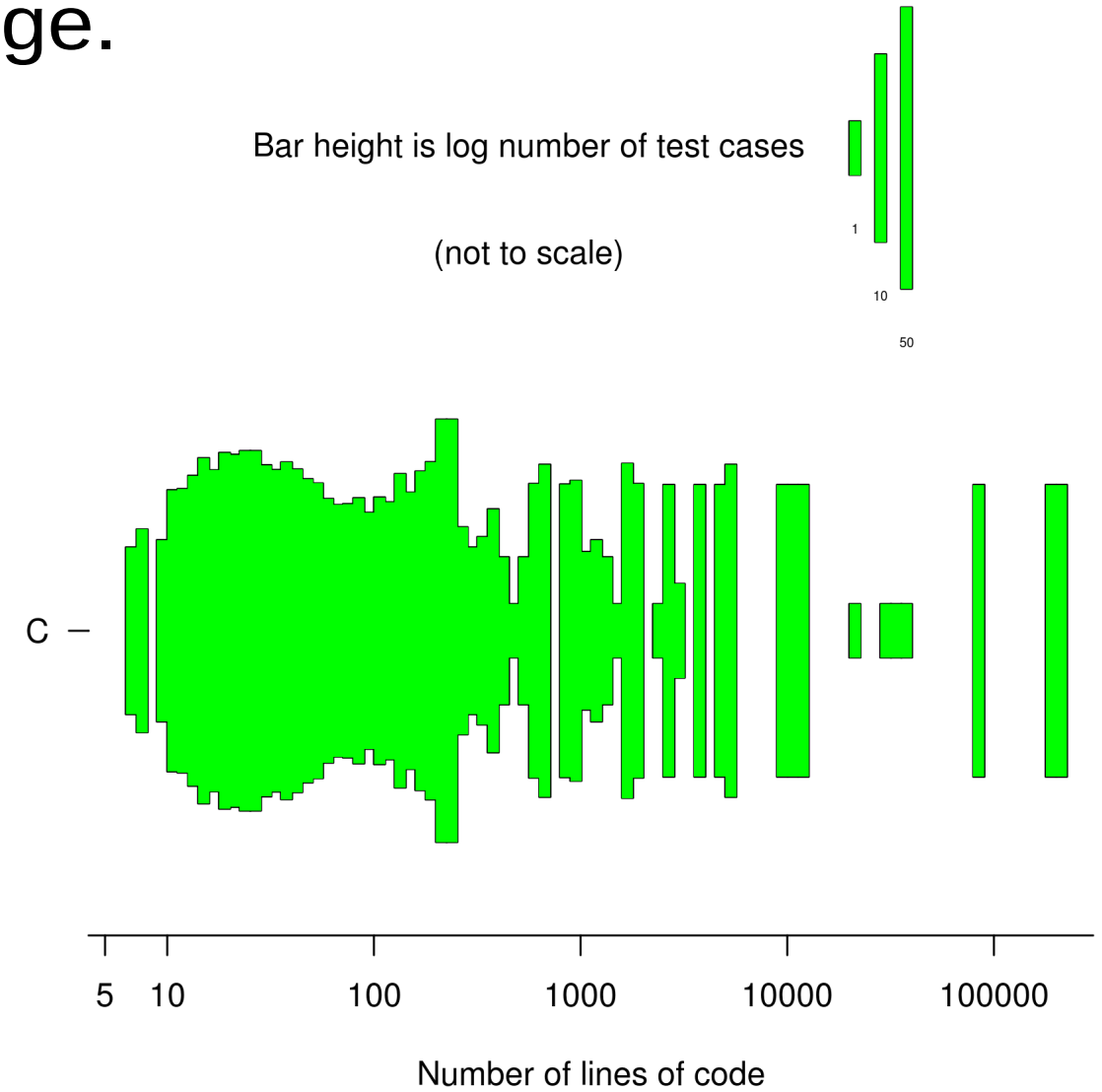


Selecting Test Cases

- Test cases must be large enough to matter.
- Tool maker selects their test cases from those we offer.
- Some possibilities:
  - SATE VI classic or mobile track cases
  - Juliet 1.3 (16k cases)
  - SV-COMP (2k cases)
  - DARPA Cyber Grand Challenge/Trail of Bits (< 245 cases)

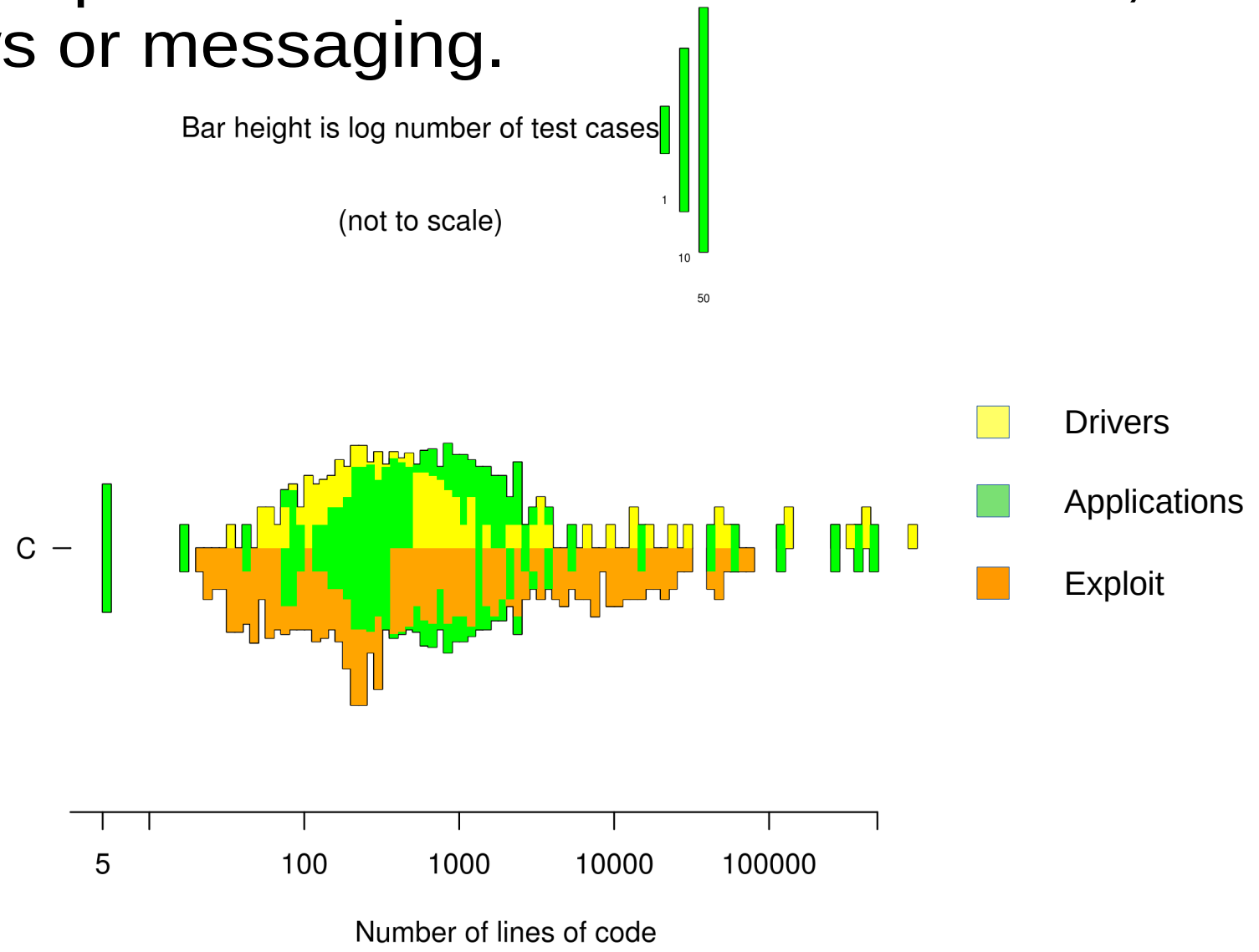
SV-COMP cases

- From the automatic Software Verification challenge.



DARPA Cyber Grand Challenge / Trail of Bits cases

- Small, simple version of interactive services, like news or messaging.



Possible Ockham Track Schedule

- Ockham test cases ready by 10 July 2017
- Tool reports returned by mid-August 2017
- Ockham team analysis begins by 1 September 2017
  - Communicate with tool makers to understand weakness classes and their sites.
- Finish analysis by January 2018
- Experience workshop March 2018

- Note: Classic track test cases will be available September 2017 at the earliest.

PROPOSED