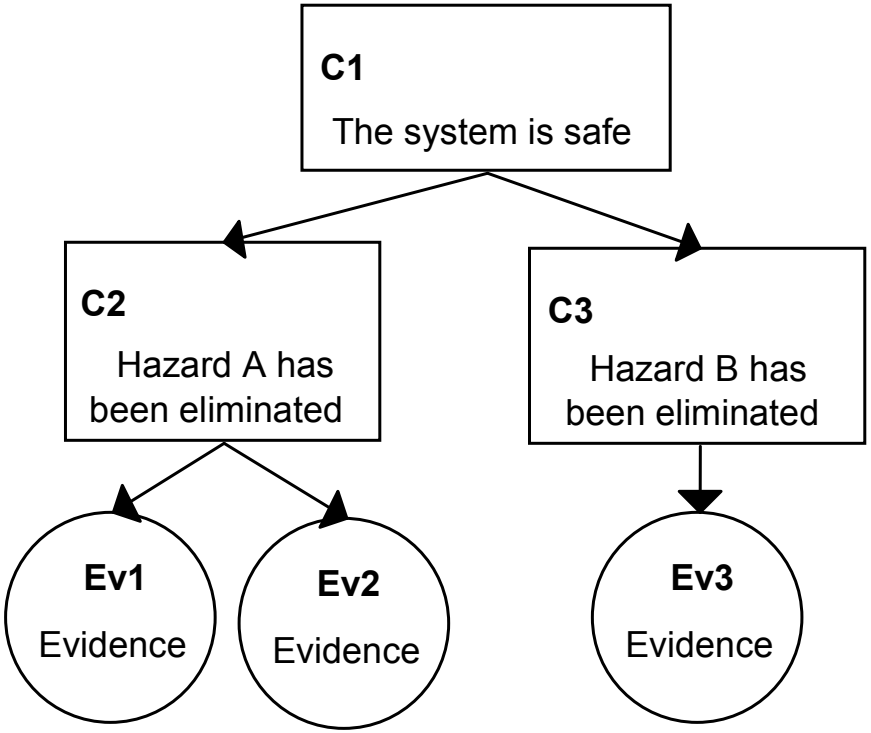
Assessing Confidence in an Assurance Case

John Goodenough
Charles B. Weinstock
Ari Z. Klein

December 6, 2011

The Problem



How confident in C1? Why?

What does it mean to have confidence?

What could be done to improve confidence? Why?

The Philosophy of Confidence

Scientific, legal hypothesis testing

- Which hypothesis is best supported by the evidence?
- A distinguished history, starting with Aristotle*
  - Pascal (1654) (finite number of equally likely outcomes)
  - Bayes (1763) (belief)
  - Francis Bacon (1620) (eliminative induction)

Assurance cases pose a somewhat different problem

- Not comparing different hypotheses
- How well is a given hypothesis (claim) supported

Eliminative induction (Bacon [1620], Cohen [1970])

- As reasons for doubt are eliminated, confidence (belief) grows
- If we have no reasons for doubting a claim, we must accept its validity

* David A. Schum, **The Evidential Foundations of Probabilistic Reasoning**, 1994.

Deductive Reasoning

Bermuda-born ⇒ British subject

Harry was Bermuda-born

∴ Harry is a British subject

Deductive Reasoning

Bermuda-born ⇒ British subject Generalization; inference rule

Harry was Bermuda-born

∴ Harry is a British subject

Deductive Reasoning

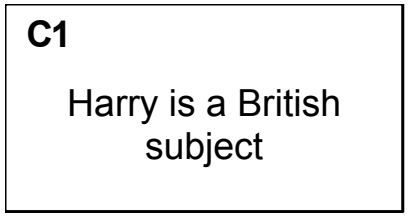
Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	

Deductive Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion

Deductive Reasoning

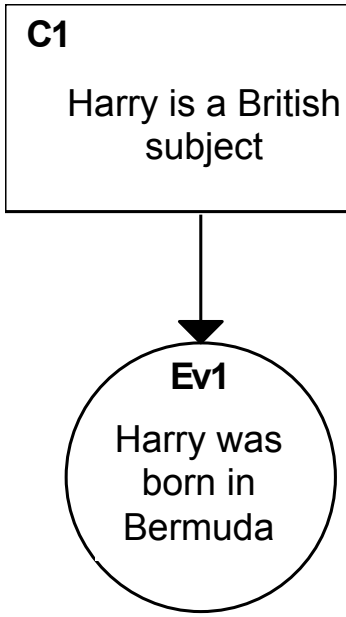
Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



Conclusion

Deductive Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



Conclusion

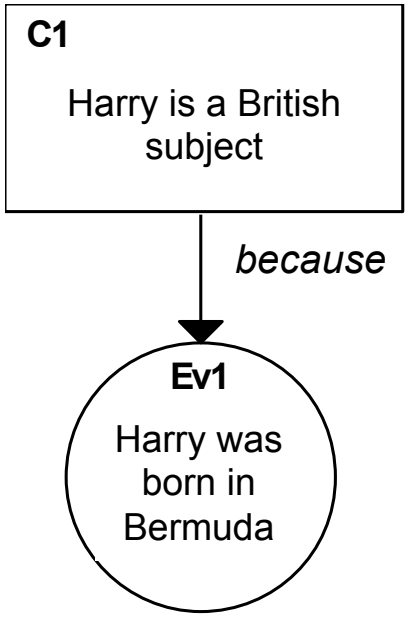
Premise

Deductive Reasoning

Bermuda-born ⇒ British subject — Generalization; inference rule

Harry was Bermuda-born — Premise

∴ Harry is a British subject — Conclusion



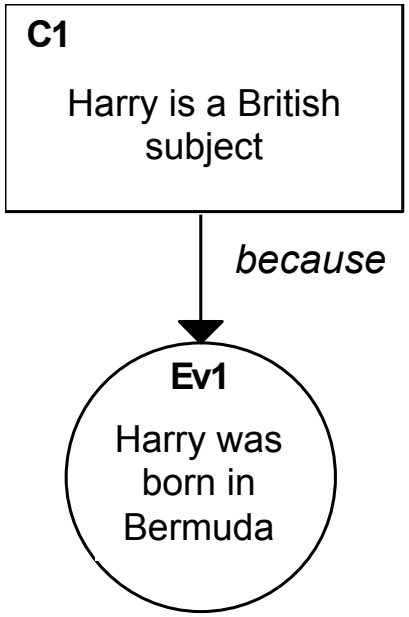
Conclusion

Bermuda-born ⇒ British subject

Premise

Defeasible Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



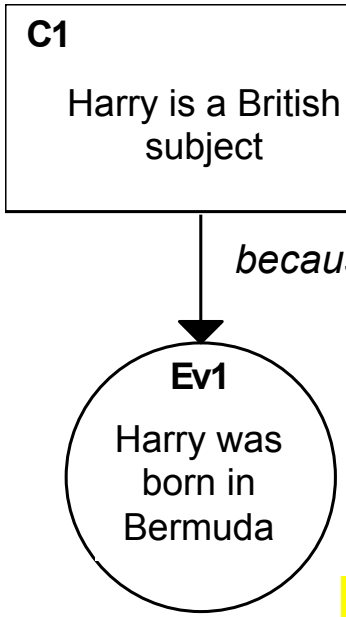
Conclusion

Bermuda-born ⇒ British subject

Premise

Defeasible Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



Conclusion

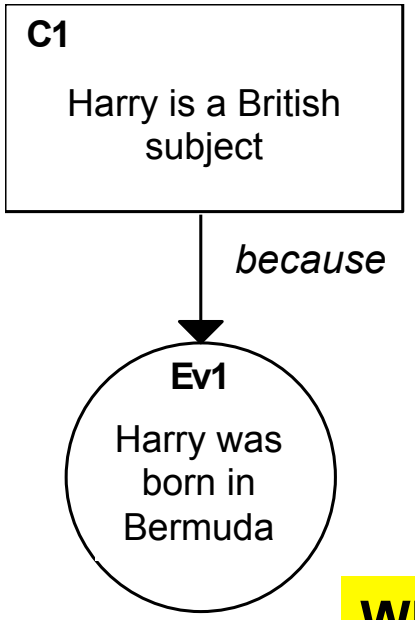
Bermuda-born ⇒ British subject

Premise

What additional information can put the conclusion in doubt?

Defeasible Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



Conclusion

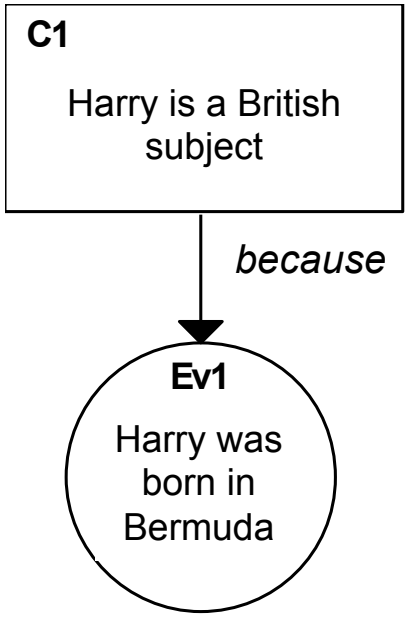
Bermuda-born ⇒ British subject
unless R, S, T, …

Premise

What additional information can put the conclusion in doubt?

Defeasible Reasoning

Bermuda-born ⇒ British subject	Generalization; inference rule
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



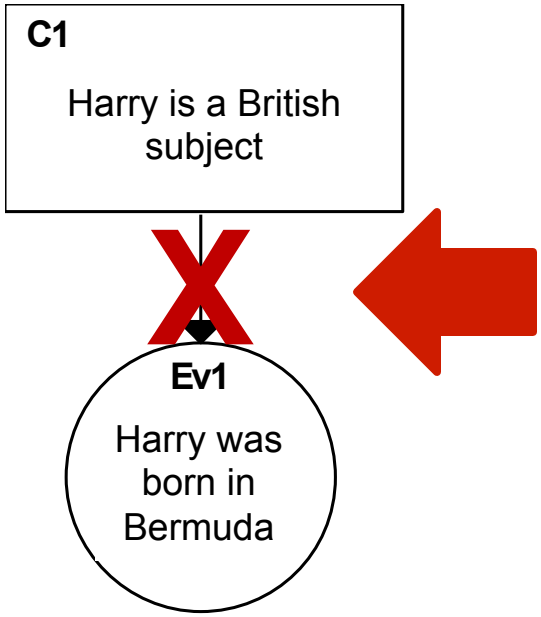
Conclusion

Bermuda-born ⇒ British subject
unless R, S, T, …

Premise

Defeasible Reasoning

Bermuda-born ⇒ British subject	Parents were not Bermuda citizens
Harry was Bermuda-born	Premise
∴ Harry is a British subject	Conclusion



Conclusion

Undercutting defeater

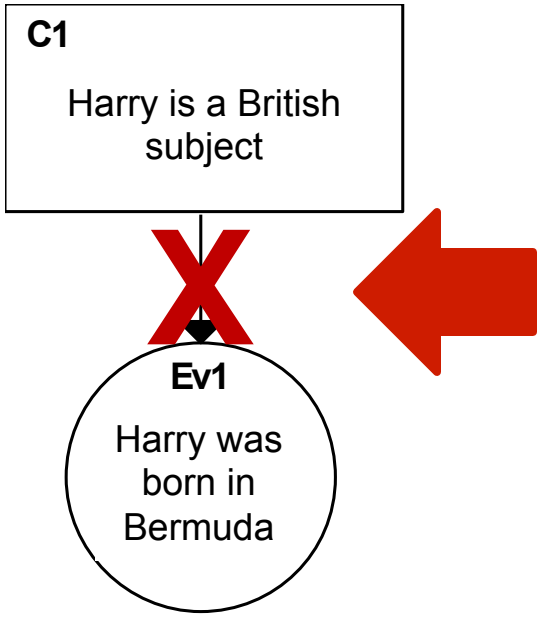
Premise

Defeasible Reasoning

Bermuda-born ⇒ British subject — Parents were not Bermuda citizens

Harry was Bermuda-born — Premise

∴ Harry is a British subject — Conclusion



Conclusion

Undercutting defeater

Premise

Defeasible Reasoning

Bermuda-born ⇒ British subject

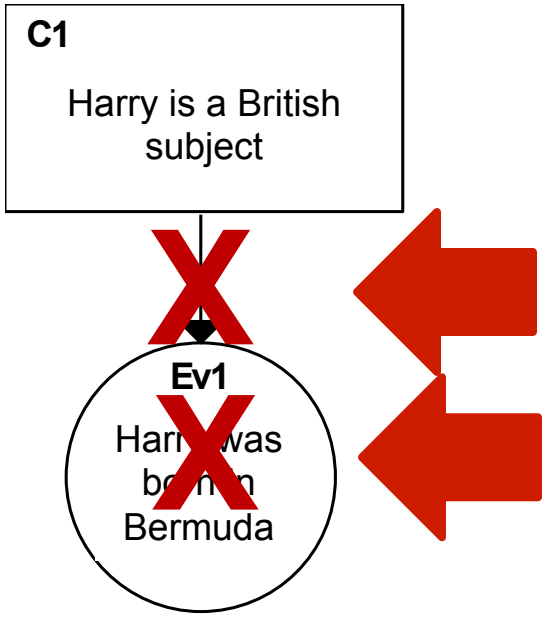
Harry was Bermuda-born

∴ Harry is a British subject

Parents were not Bermuda citizens

Harry was actually born in London

Conclusion



Conclusion

Undercutting defeater

Undermining defeater

Defeasible Reasoning

Bermuda-born ⇒ British subject

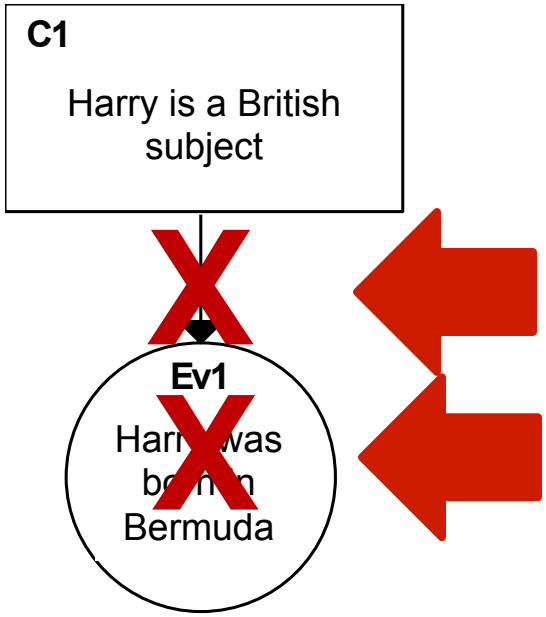
Harry was Bermuda-born

∴ Harry is a British subject

Parents were not Bermuda citizens

Harry was actually born in London

Conclusion



Conclusion

Undercutting defeater

Undermining defeater

Defeasible Reasoning

Bermuda-born ⇒ British subject

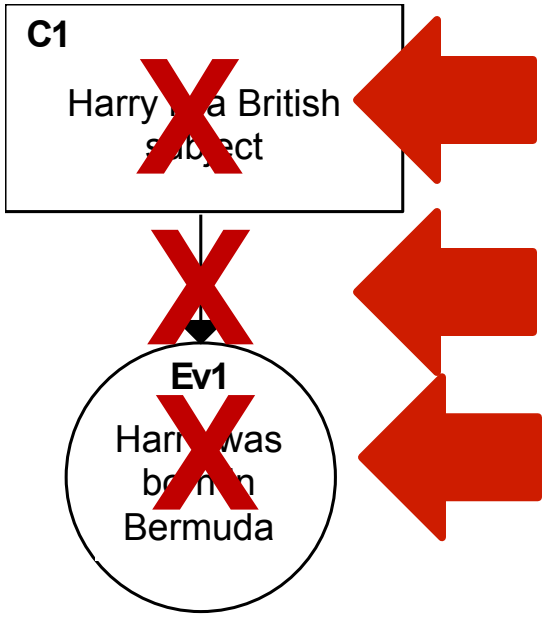
Harry was Bermuda-born

∴ Harry is a British subject

Parents were not Bermuda citizens

Harry was actually born in London

Harry renounced his citizenship



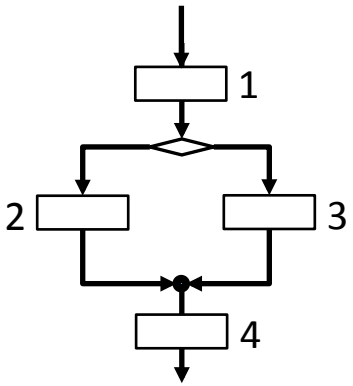
Rebutting defeater

Undercutting defeater

Undermining defeater

Enough About Harry; Give Me Some Code!

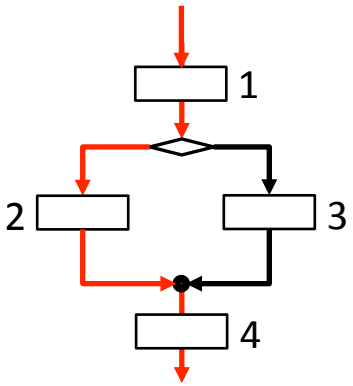
Egregious error: *Every* execution of a statement containing an egregious error will fail



C1

There are no egregious errors in the program

Enough About Harry; Give Me Some Code!



C1

There are no egregious errors in the program

C2

There are no egregious errors in any basic block

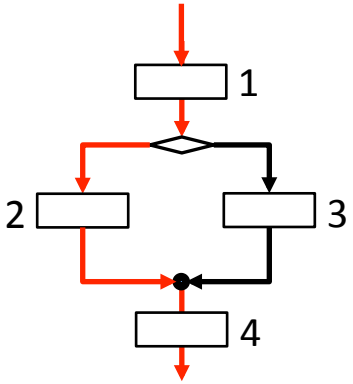
Four Rebutting Defeaters

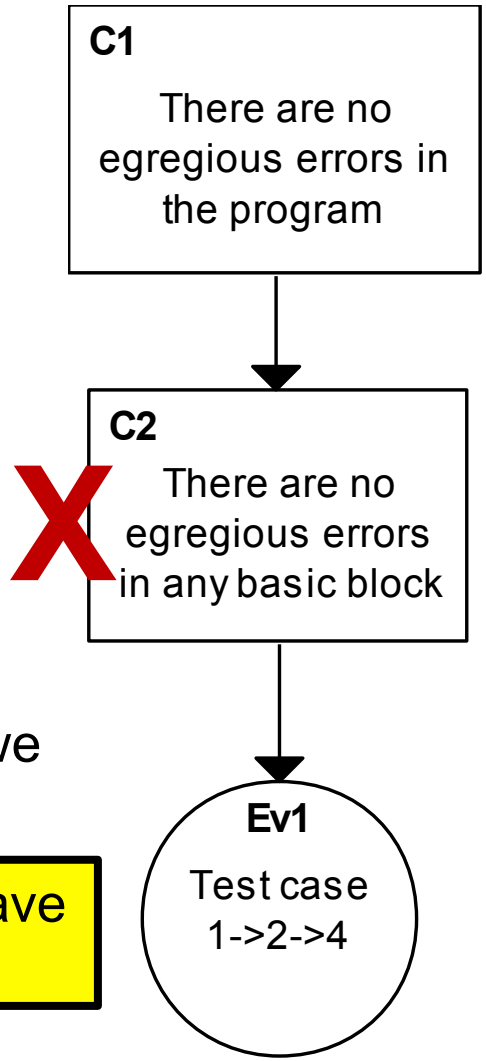
~~1) BB1 execution always fails~~

~~2) BB2 execution always fails~~

3) BB3 execution always fails

~~4) BB4 execution always fails~~





- Confidence increases as we eliminate defeaters
- Confidence is a measure of how much info we have about defeaters and their elimination

We have 75% of the information needed to have complete confidence in the claim

What about other defeaters?

Some undermining defeaters

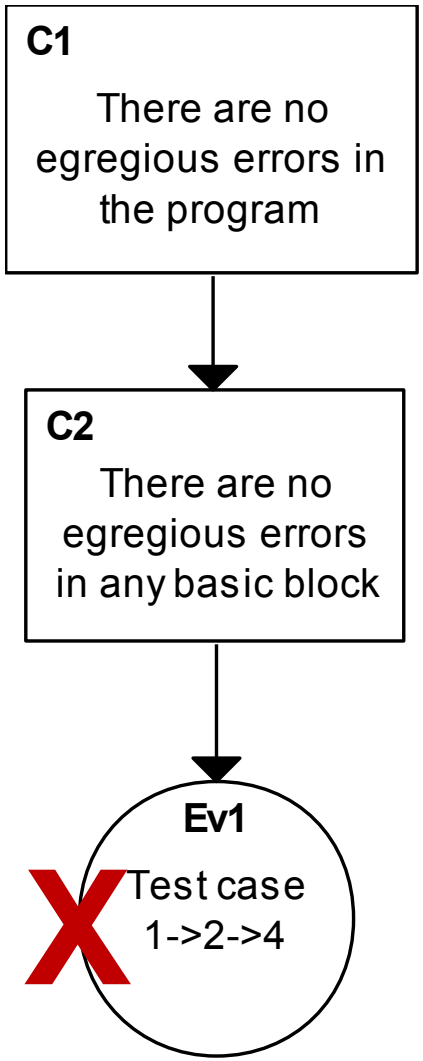
- Test oracle does not report test success accurately
- The test results do not apply to the current version of the code
- Assertions about what basic blocks have been executed are unreliable

Eliminating these defeaters

- Evidence/analysis showing oracle is reliable
- Evidence/analysis of the configuration management mechanisms being used
- Evidence/analysis shows reports of basic block executions are reliable

Note that the defeaters are <u>independent</u>

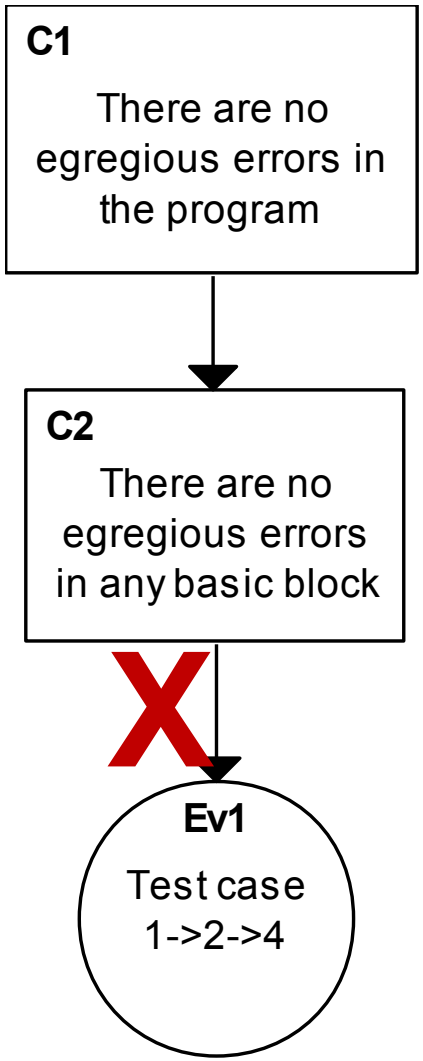
- Eliminating one does not eliminate others



What about other defeaters?

An undercutting defeater

- What is the generalization being used here?
  - If all basic blocks are successfully executed (at least once), there are no egregious errors in any basic block
- Defeater: None; this is logically equivalent to the definition of egregious error



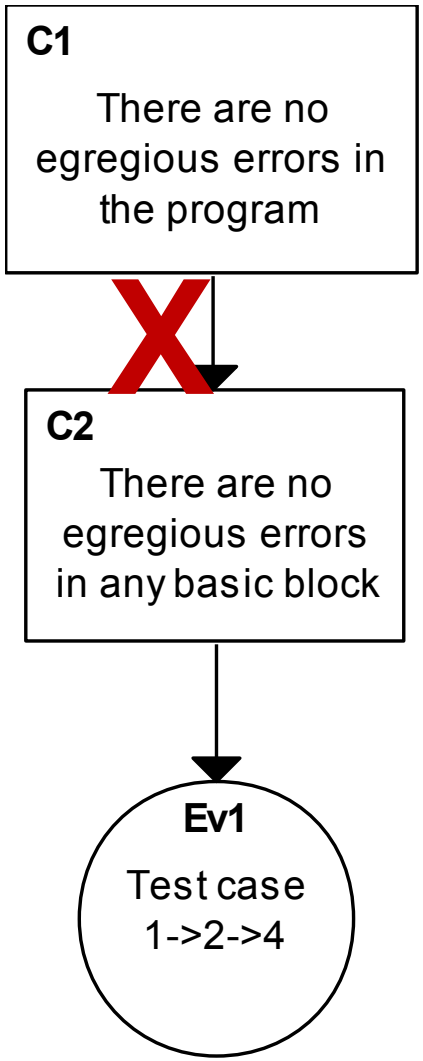
What about other defeaters?

An undercutting defeater

- What is the generalization being used here?
  - No egregious errors in any basic block implies there are no egregious errors in the program
- Defeater: Not all basic blocks are identified

Eliminating the defeater

- Analysis of the method for finding basic blocks ensures none will be overlooked



Summary of Defeaters

8 defeaters for C2

- Four rebutting defeaters (BB execution success)
- Three undermining defeaters (oracle, configuration mgmt, BB assertion)
- Deductive undercutting defeater (is defeated by deduction)

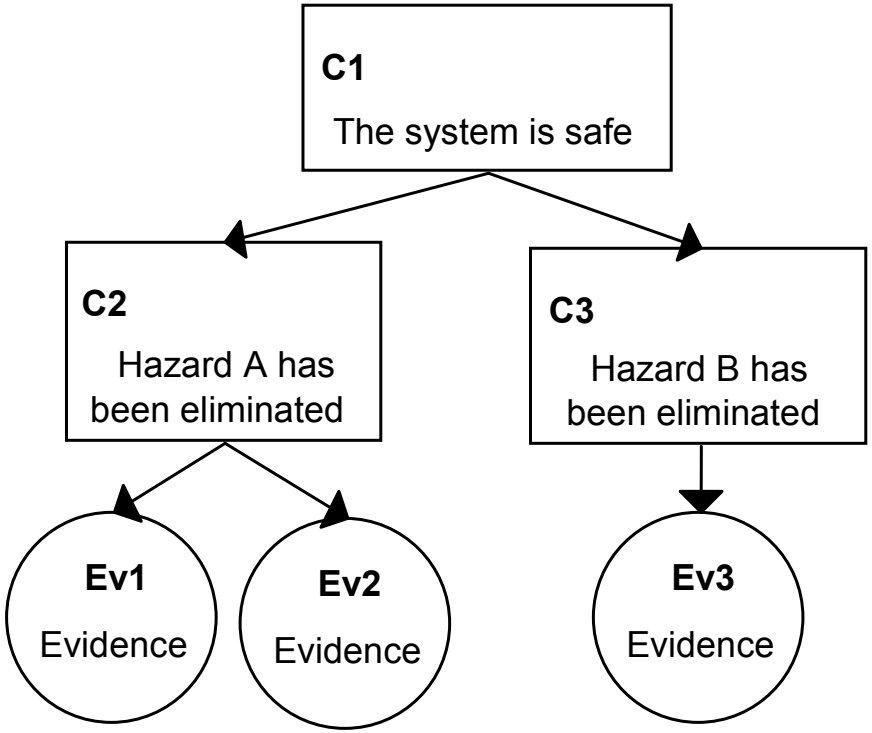
Test case eliminates 3 (rebutting defeaters)

- In the absence of information about the other defeaters, the Baconian probability (belief) is 4 out of 8, expressed as 4/8
  - 4/8 is not a fraction; it is a measure of missing information
  - 0/8 does not mean the claim is false; it means we have no reason to believe the claim

If no reasons for doubt have been eliminated, we have no confidence in a claim

If we eliminate all reasons for doubt, we have no reason to doubt

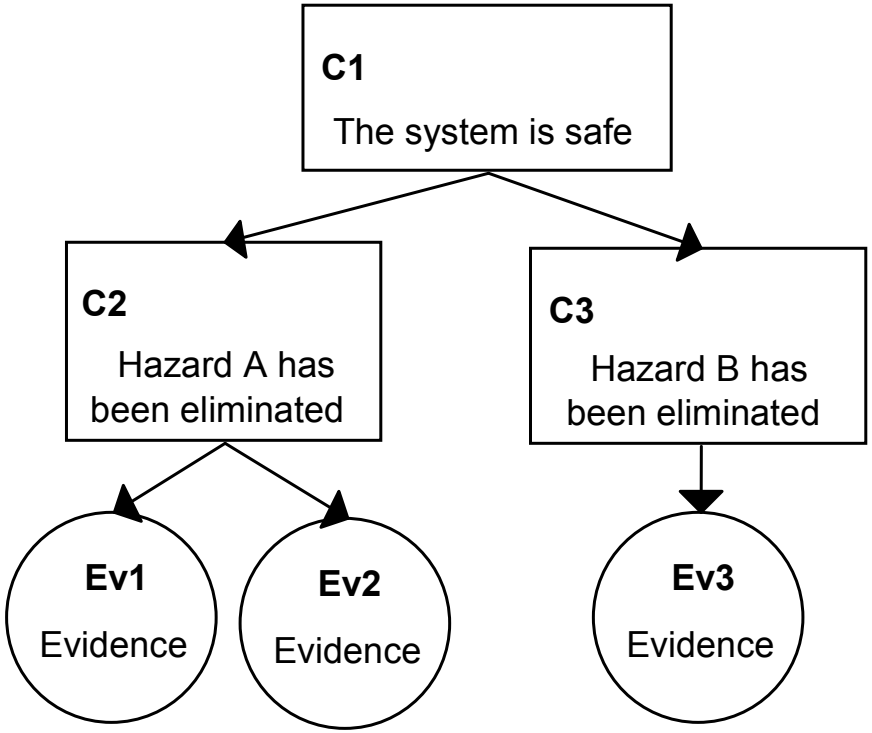
The Problem



How confident in C1? Why?

Number of eliminated defeaters

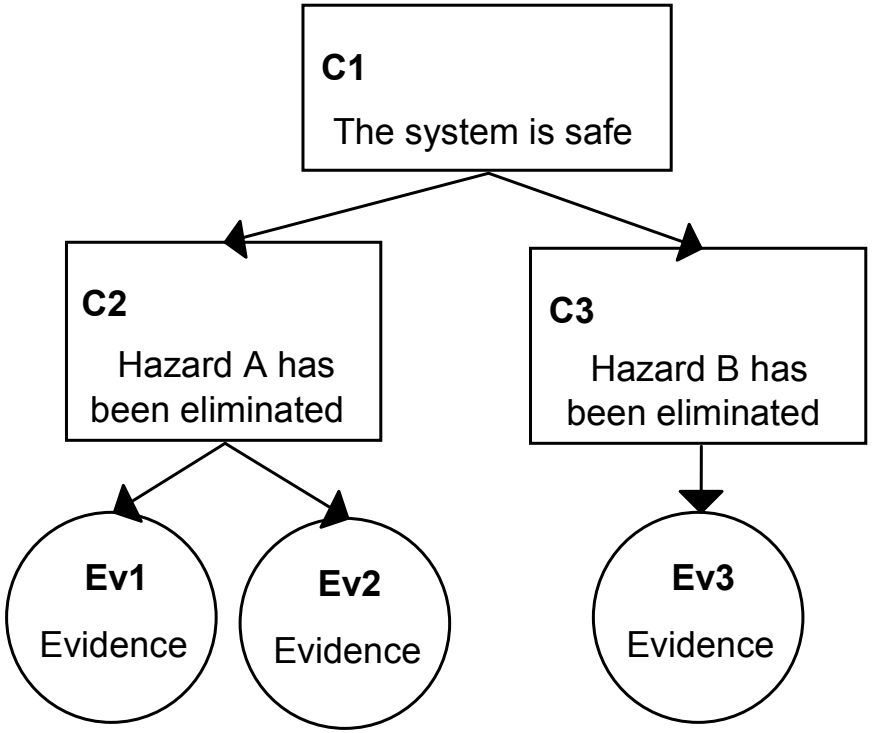
The Problem



How confident in C1? Why? Number of eliminated defeaters

What does it mean to have confidence? No reason to doubt

The Problem

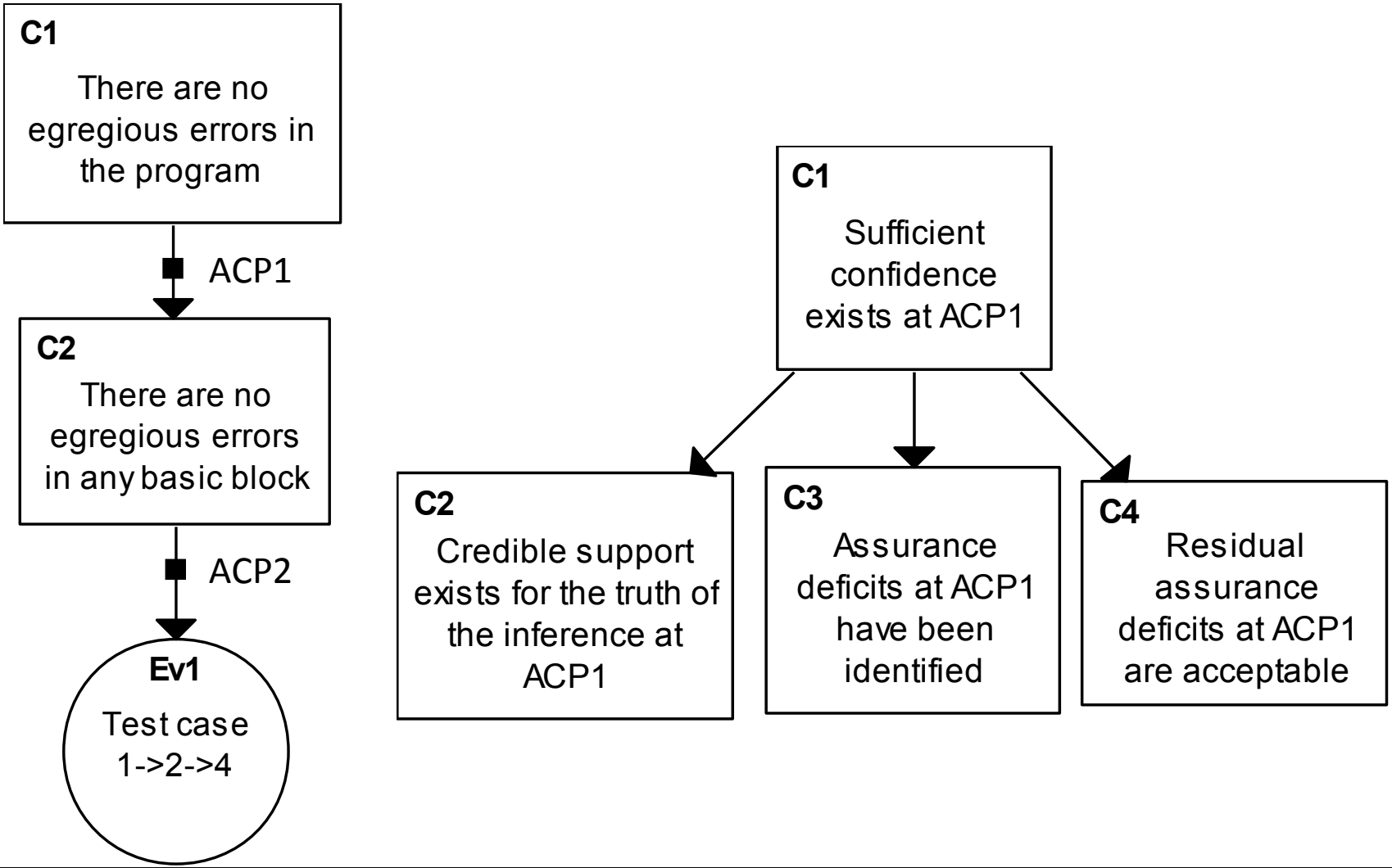


How confident in C1? Why? Number of eliminated defeaters

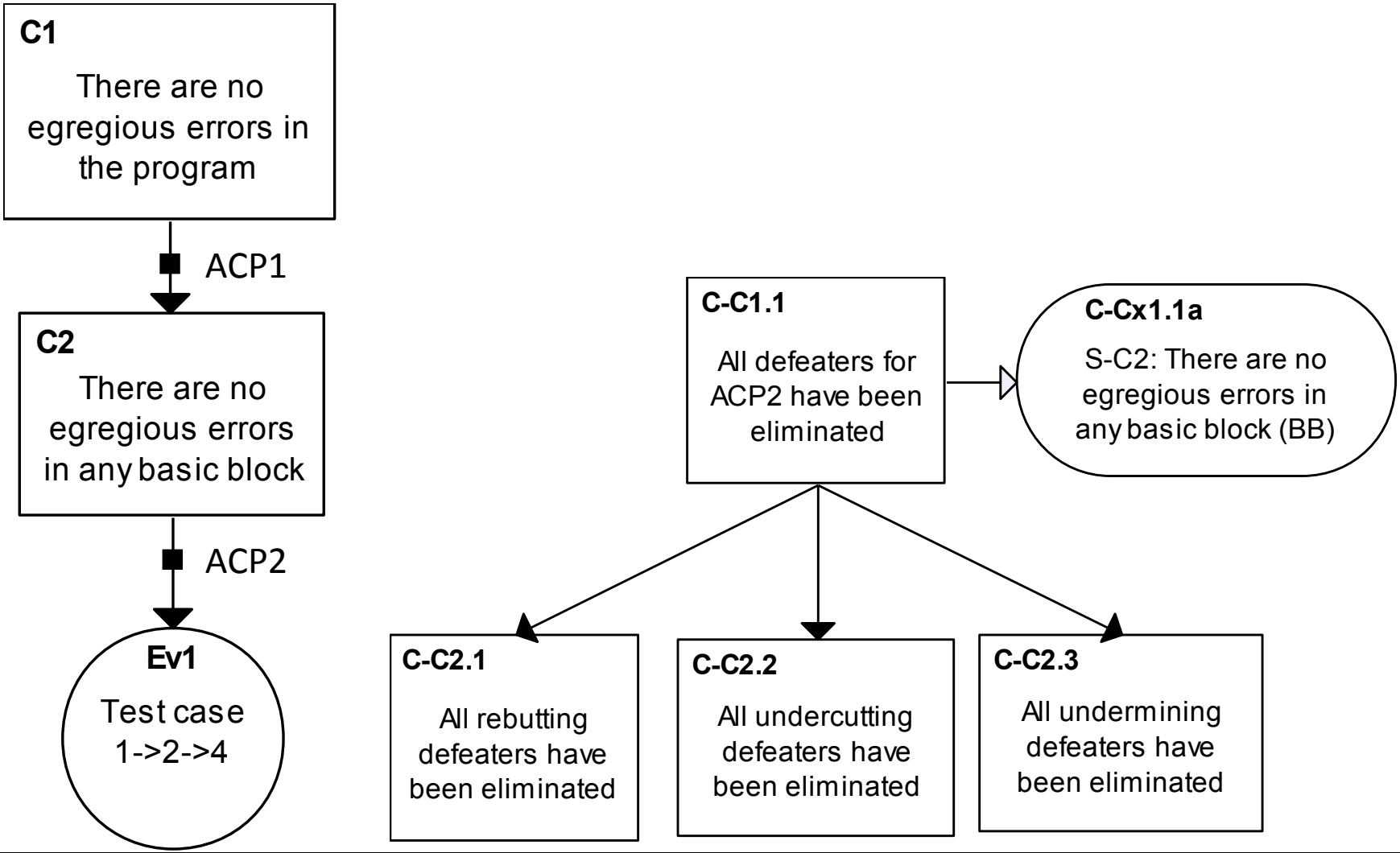
What does it mean to have confidence? No reason to doubt

How to improve confidence? Why? Eliminate more defeaters

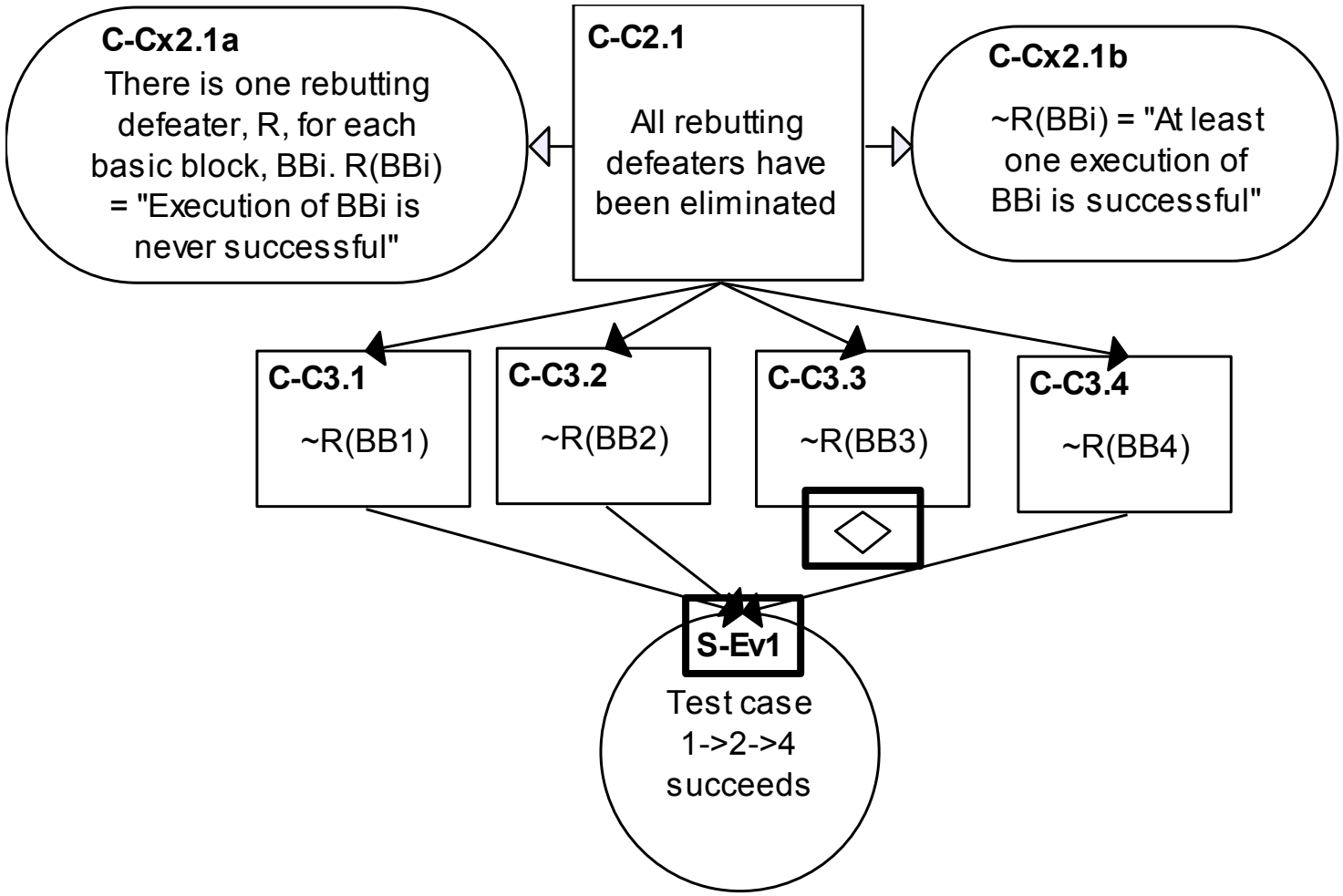
Building the Confidence Case



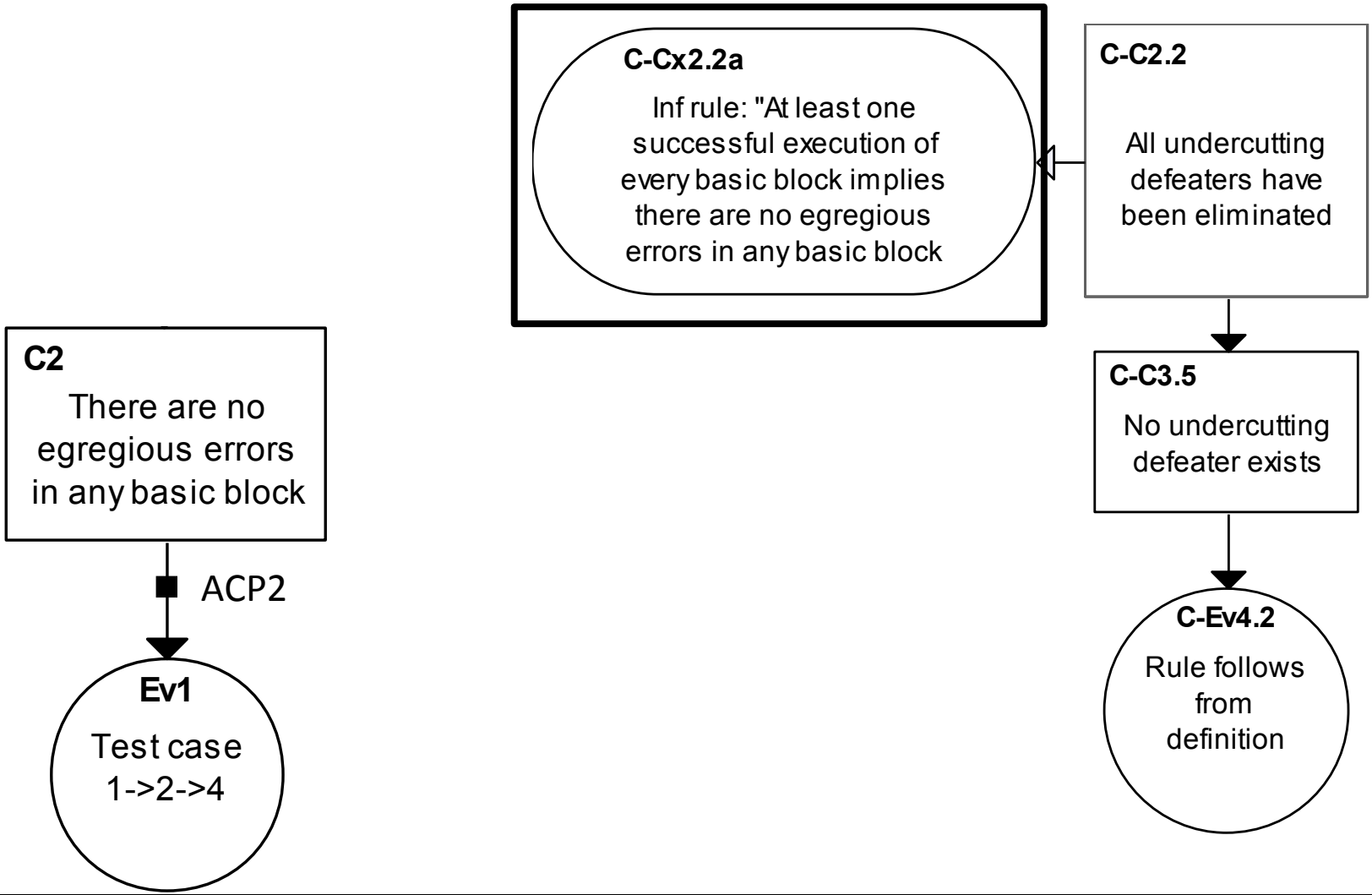
Building the Confidence Case



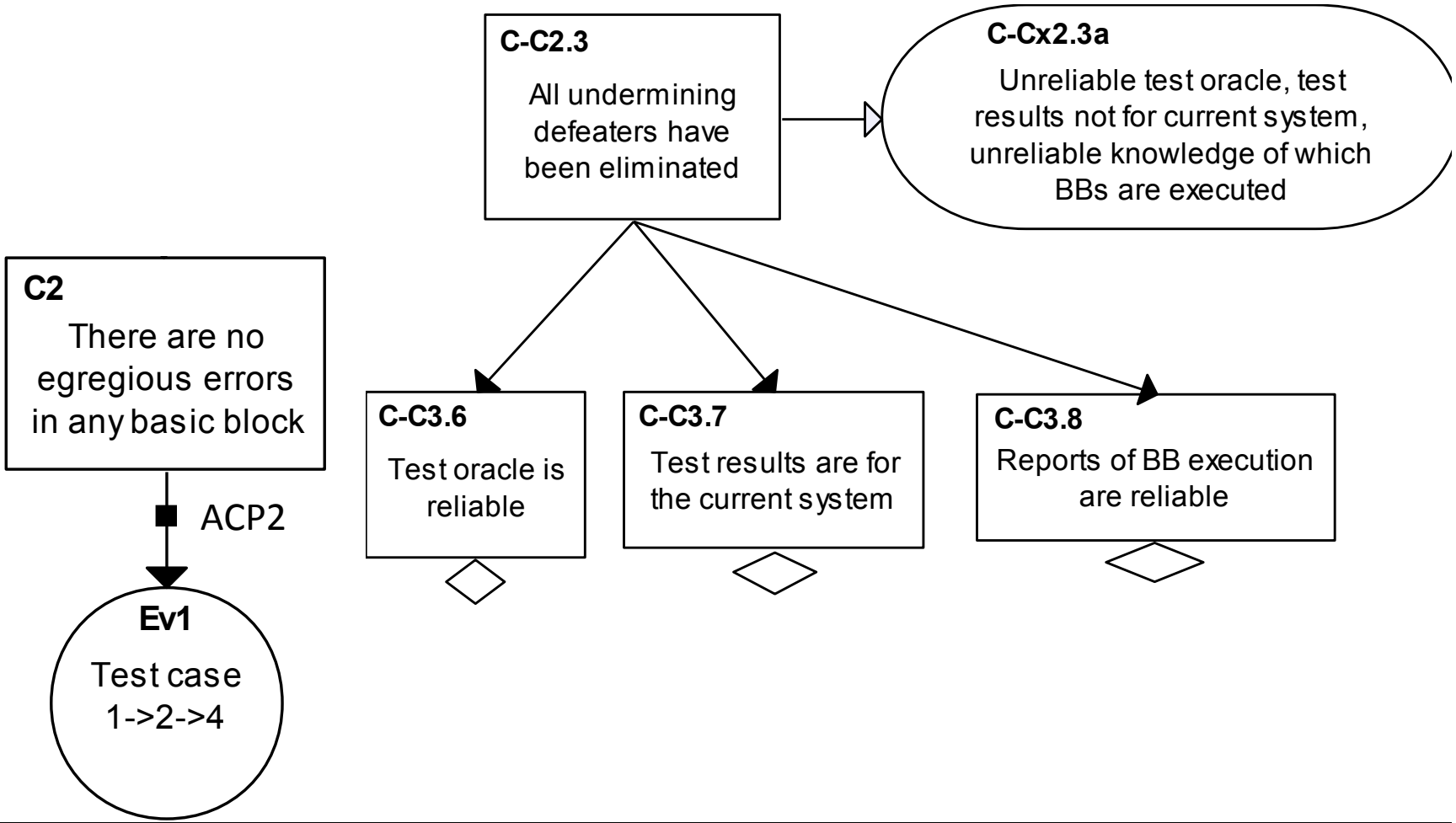
Building the Confidence Case



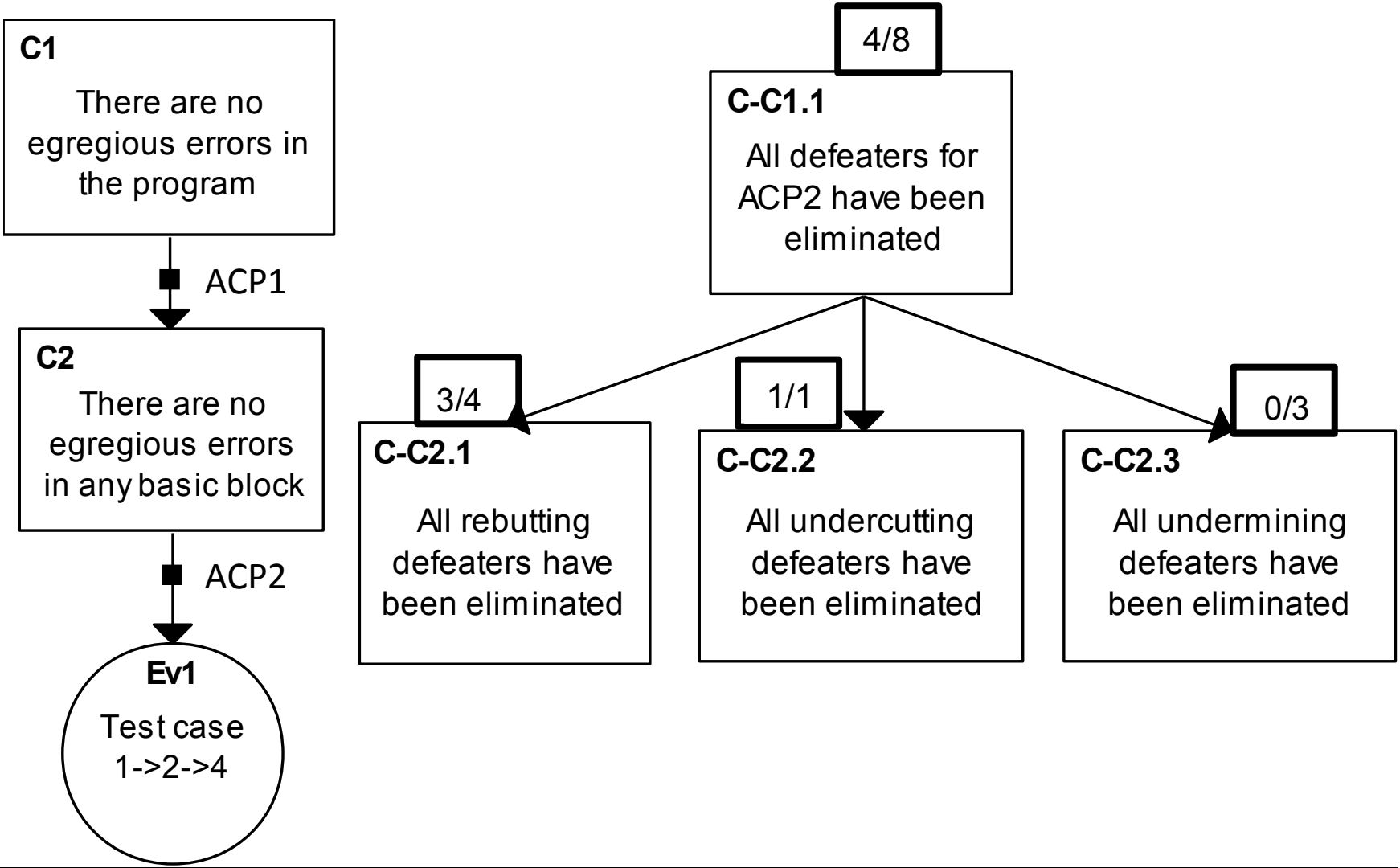
Building the Confidence Case



Building the Confidence Case



Building the Confidence Case



Other Implications of this Approach

Multi-legged arguments

- The combined arguments eliminate more defeaters

Taking into account defeater likelihood

- Eliminating some defeaters gives more confidence than others

Evaluating the strength/value of evidence in terms of how many defeaters are eliminated or the importance of the eliminated defeaters

- Irrelevant evidence eliminates no defeaters
- Powerful evidence eliminates more defeaters than weaker evidence

Undermining defeaters address the “trustworthiness” of evidence

Summary: A Basis for Assessing Confidence

Confidence is the degree of belief in a claim as measured by the number of eliminated defeaters

As more defeaters are eliminated, confidence grows (eliminative induction)

The three types of defeaters suggested by defeasible reasoning give a basis for finding defeaters

A confidence argument shows how identified defeaters are eliminated and gives insight into the basis for confidence in a system claim

Some References

David A. Schum, The Evidential Foundations of Probabilistic Reasoning, 1994

J. Cohen, An Introduction to the Philosophy of Induction and Probability, Clarendon, 1989.

J. Cohen, The Probable and the Provable, Clarendon, 1977.

J. Pollock, “Defeasible Reasoning,” in Reasoning: Studies of Human Inference and Its Foundations, J.E. Adler and L.J. Rips (eds.), Cambridge University Press, 2008.

S. Toulmin, “The Uses of Argument,” Cambridge University Press, 1958.

Contact

John B. Goodenough
Fellow
Telephone: 412-268-6391
Email: jbg@sei.cmu.edu

U.S. Mail:
Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-3890

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.