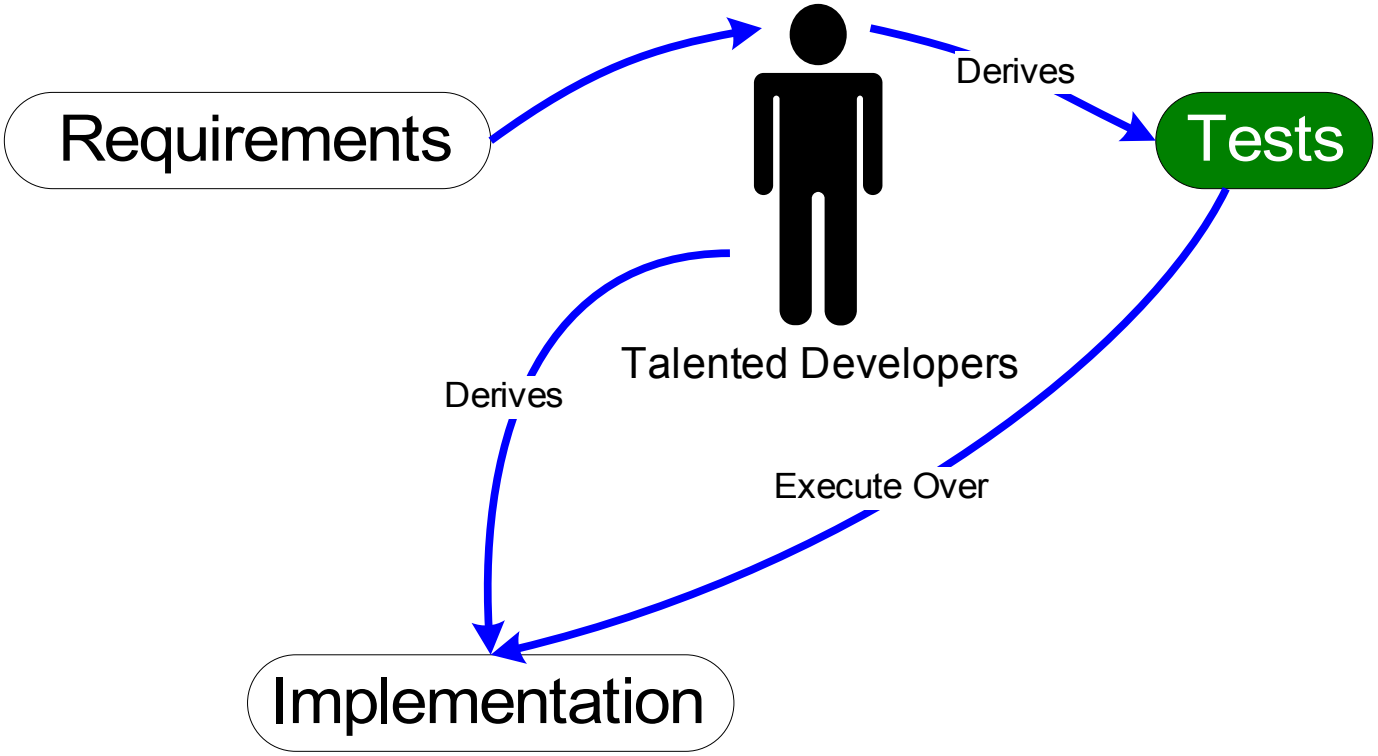
Coverage Metrics for Requirements-Based Testing: Evaluation of Effectiveness

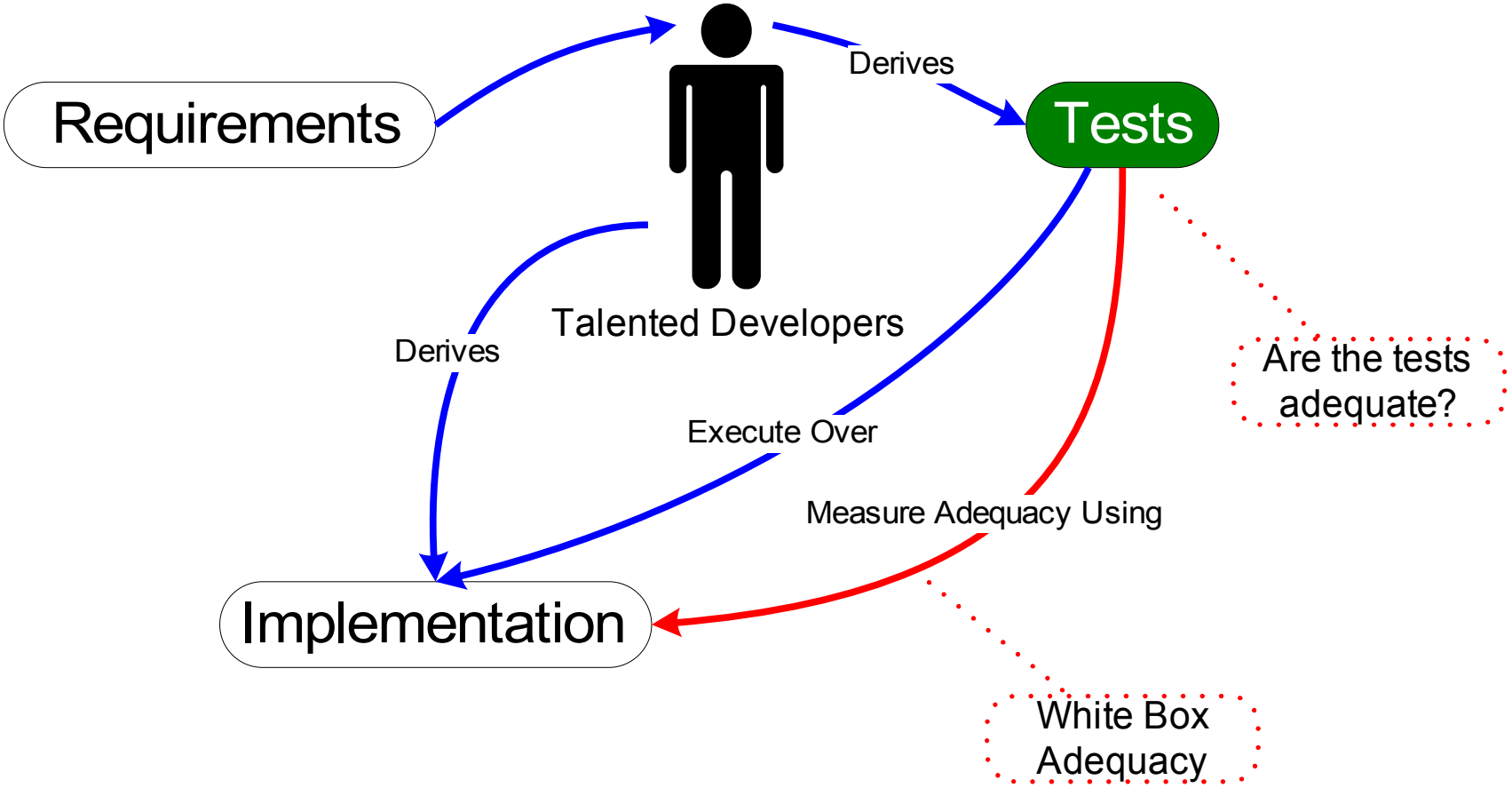
Matt Staats, Michael W. Whalen, Mats Heimdahl
University of Minnesota
Ajitha Rajan
Laboratoire d'Informatique de Grenoble

NFM 2010

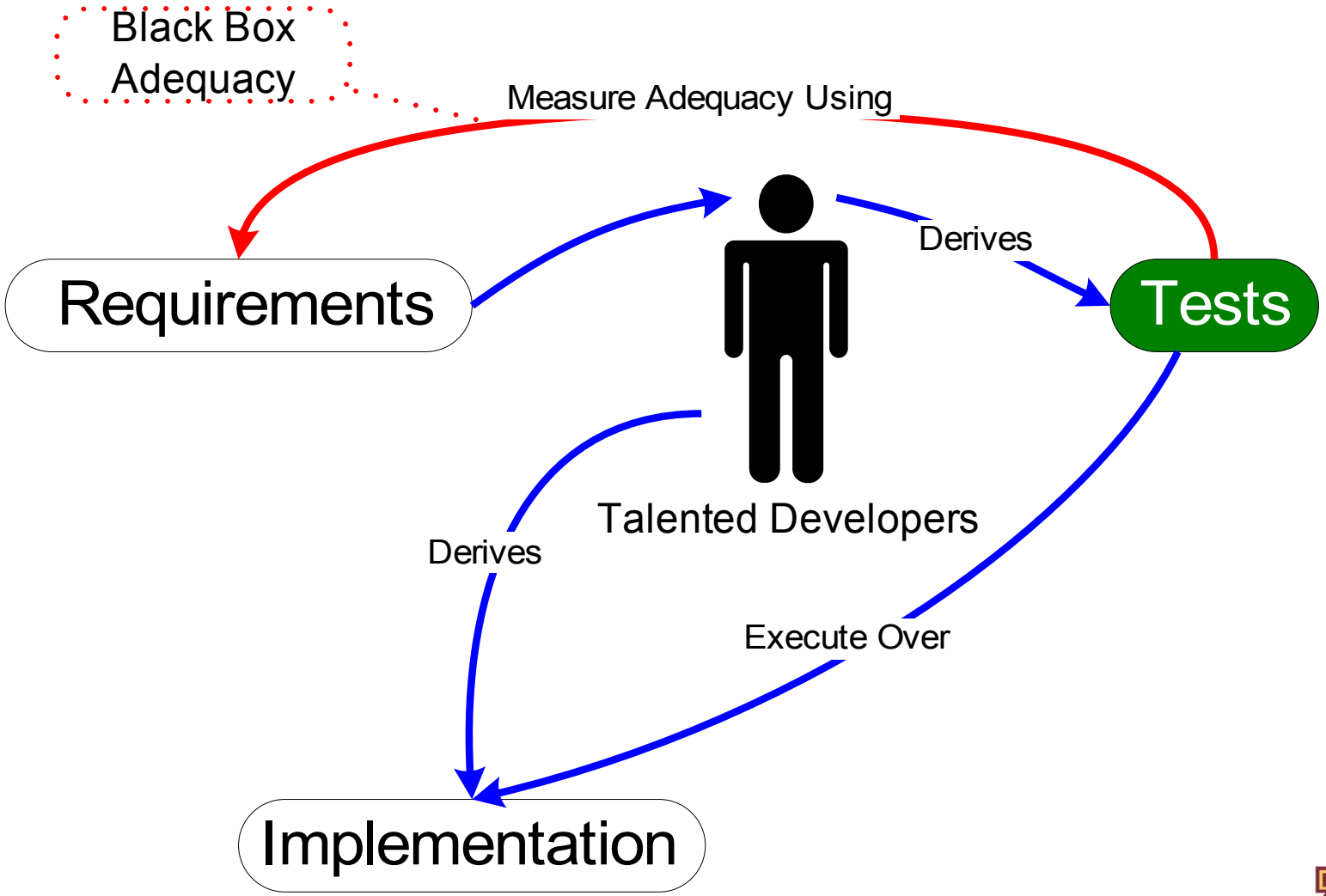
Testing Process



Testing Process



Testing Process Revised



Formalized Requirements

"If the onside FD cues are off, the onside FD cues shall be displayed when the AP is engaged"

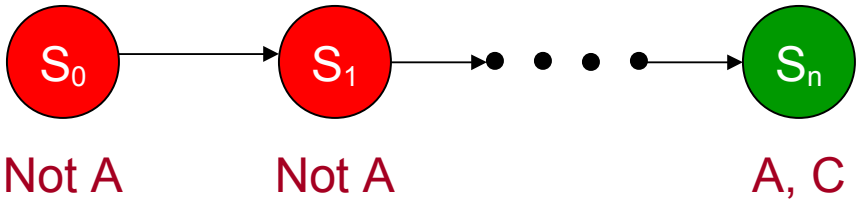
$$G((\neg\ Onside\_FD\_On\ \wedge\ \neg\ Is\_AP\_Engaged) \rightarrow X(Is\_AP\_Engaged \rightarrow Onside\_FD\_On))$$

- Possible Coverage Metrics
  - *Naïve requirements coverage*: Single test case that demonstrates that requirement is satisfied
    - Prone to "dumb" tests, e.g., execution in which AP is never engaged.
  - More rigorous metrics are necessary

Antecedent Coverage

- Many of the requirements in the FGS are of the form :
  - *Globally if 'A' occurs then 'C' will occur*

    G (A → C)
  - Two ways of satisfying (A → C)
    - A is false
    - A is true and C is true
- Antecedent Coverage – Test cases will exercise the antecedent.



Antecedent Coverage

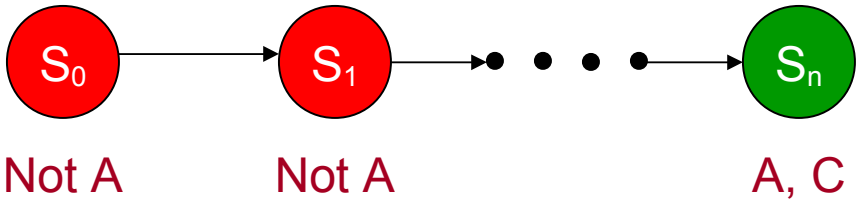
- Many of the requirements in the FGS are of the form :
  - *Globally if 'A' occurs then 'C' will occur*

    $G\,(A \rightarrow C)$
  - Two ways of satisfying $(A \rightarrow C)$
    - A is false
    - A is true and C is true

What if:

$A \vee B \rightarrow C$

- Antecedent Coverage – Test cases will exercise the antecedent.



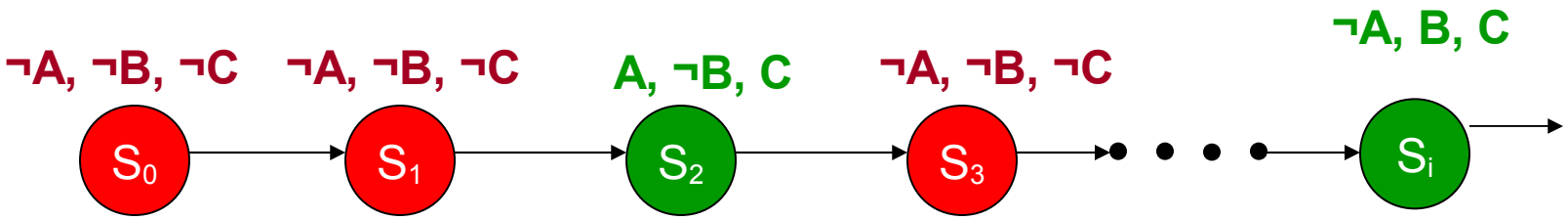
Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path

Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path

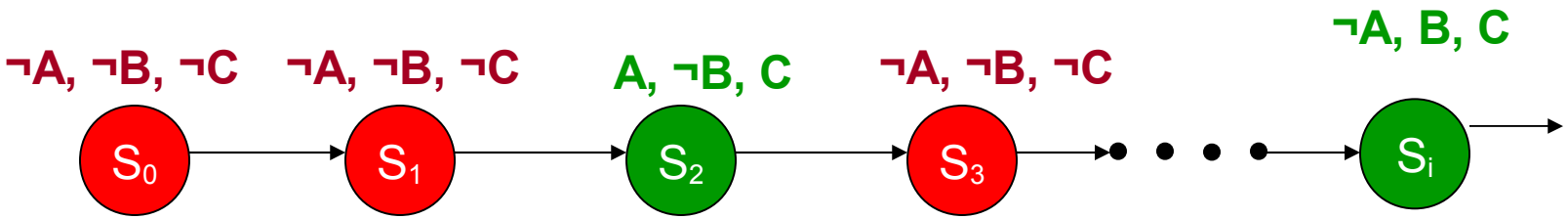
Example LTL property - $G(A \vee B \rightarrow C)$.



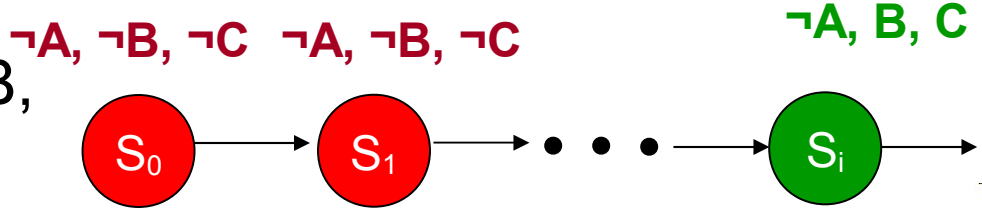
Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path

Example LTL property - $G(A \vee B \rightarrow C)$.



To show independence of B,



Study Goals

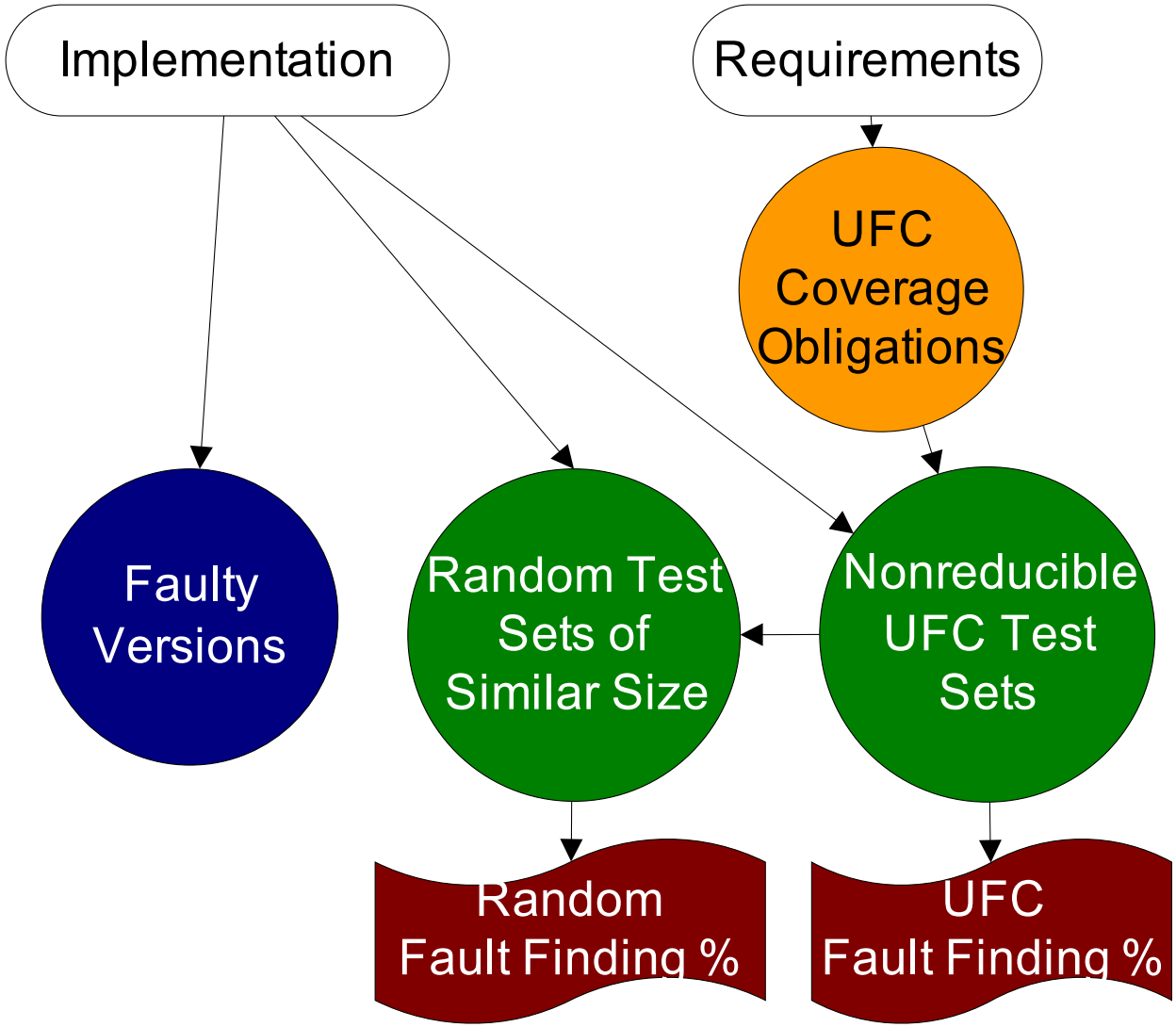
- Is *subsumption* between these metrics indicative of practical effectiveness?
- Are these coverage metrics *any good?*

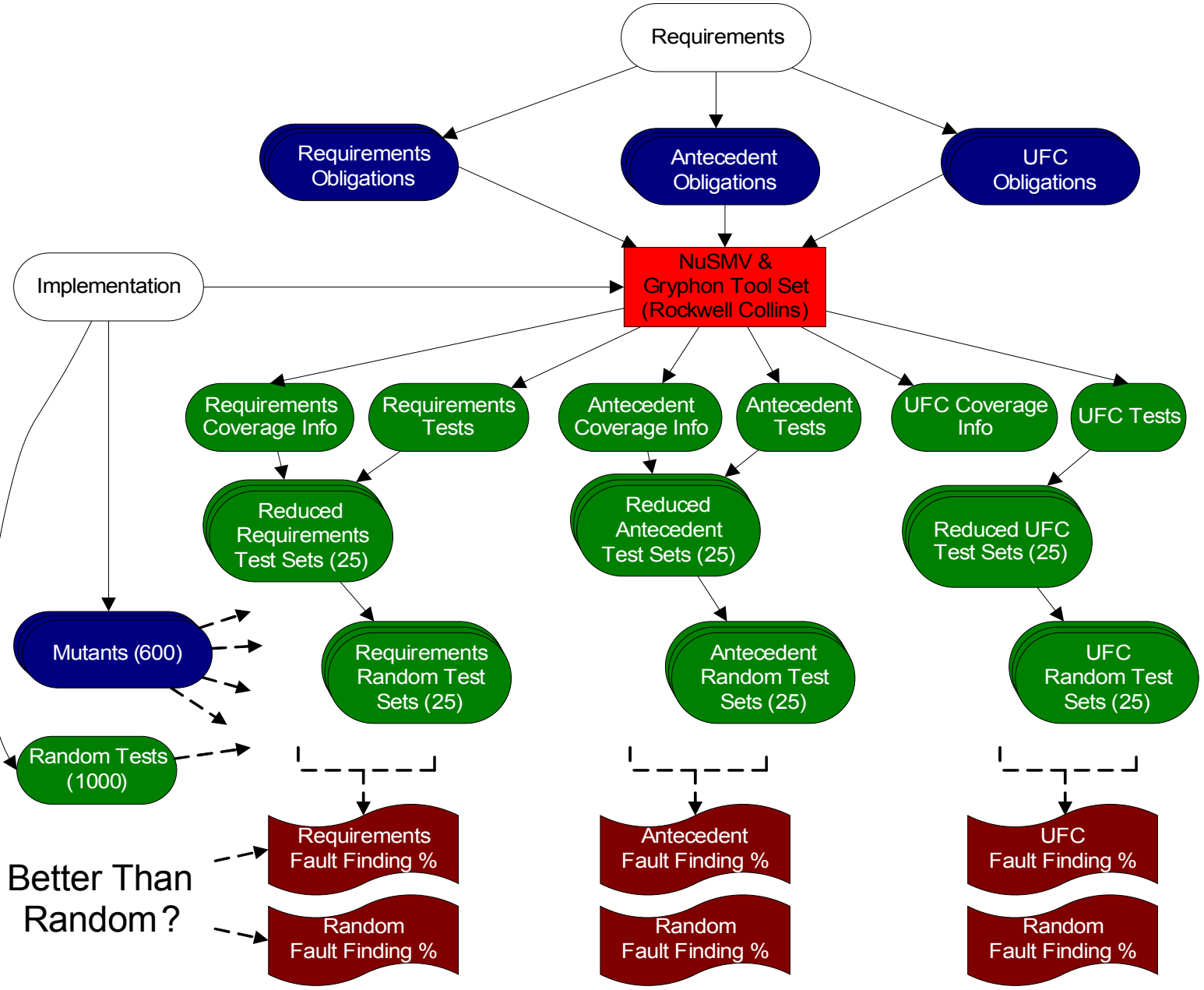
Case Examples

- Avionics systems courtesy of Rockwell Collins
  - Simulink, translated to Lustre
- Includes "good" set of LTL requirements

	# Simulink Subsystems	# Blocks	# Requirements
DWM_1	3,109	11,439	170
DWM_2	128	429	41
Vertmax_Batch	396	1,453	294
Latctl_Batch	120	718	110

Basic Experimental Setup

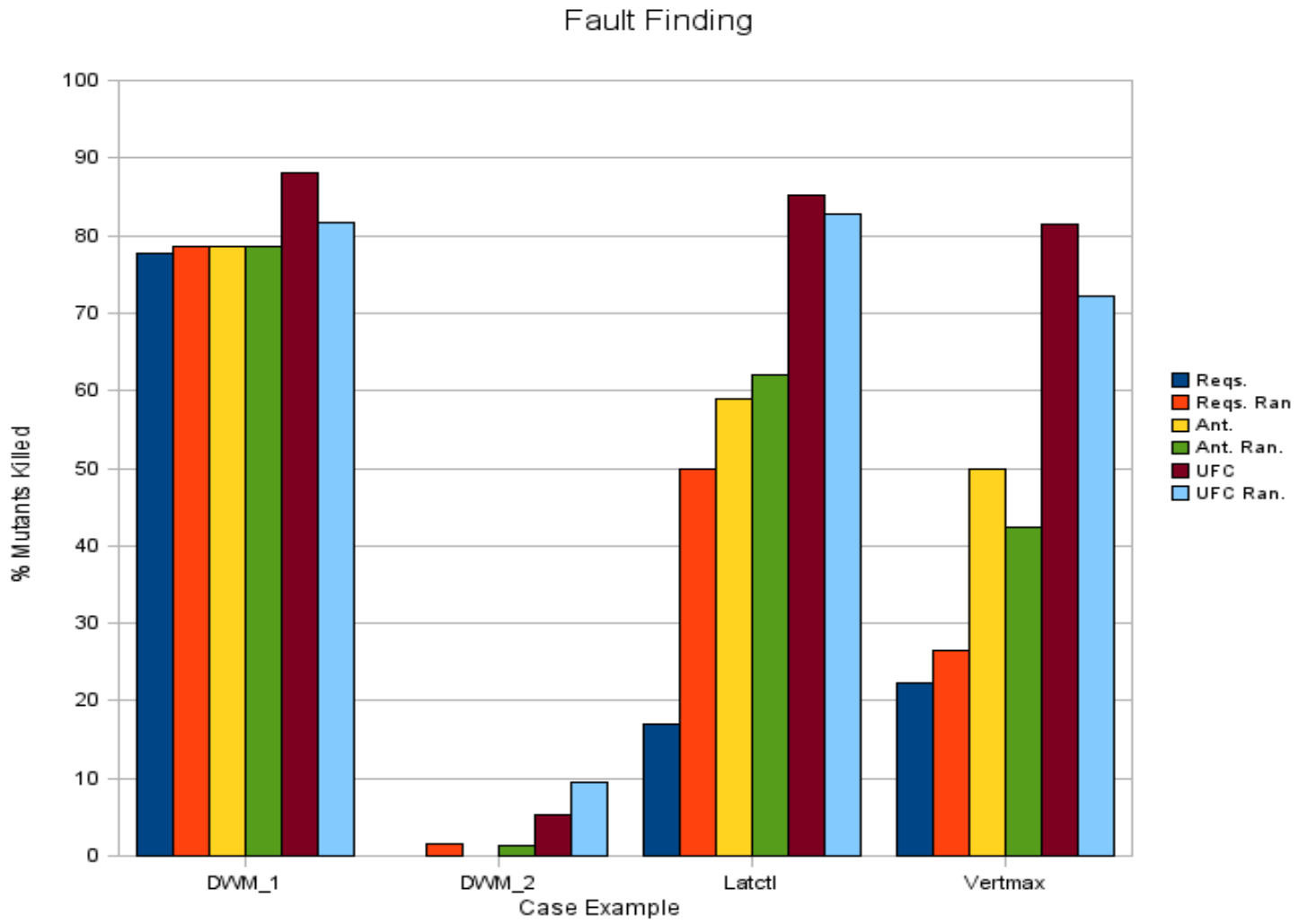



Requirements
Requirements Obligations
Antecedent Obligations
UFC Obligations
Implementation
NuSMV & Gryphon Tool Set (Rockwell Collins)
Requirements Coverage Info
Requirements Tests
Antecedent Coverage Info
Antecedent Tests
UFC Coverage Info
UFC Tests
Reduced Requirements Test Sets (25)
Reduced Antecedent Test Sets (25)
Reduced UFC Test Sets (25)
Mutants (600)
Requirements Random Test Sets (25)
Antecedent Random Test Sets (25)
UFC Random Test Sets (25)
Random Tests (1000)
Requirements Fault Finding %
Antecedent Fault Finding %
UFC Fault Finding %
Random Fault Finding %
Random Fault Finding %
Random Fault Finding %
Better Than Random?

Results

- Does subsumption relate to fault finding effectiveness?
  - YES! (Mostly)
- Are these coverage metrics effective measures of adequacy?
  - NO!
    - For requirements and antecedent coverage
  - YES!
    - For UFC coverage, for 3 of 4 systems

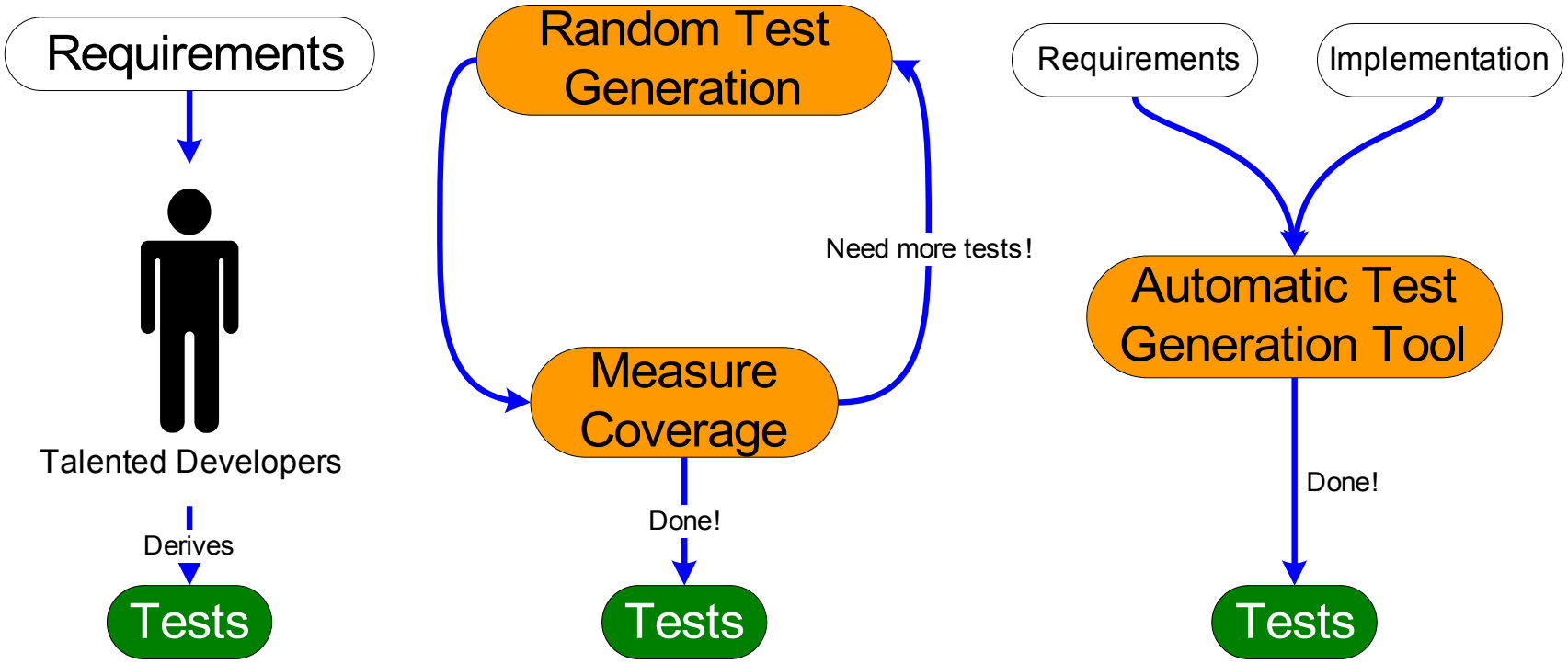
Results


Fault Finding
% Mutants Killed
100
90
80
70
60
50
40
30
20
10
0
DVWM_1
DVWM_2
Latctl
Vertmax
Case Example
Reqs.
Reqs. Ran
Ant.
Ant. Ran.
UFC
UFC Ran.

New Questions

- Why do test sets satisfying requirements and antecedent coverage perform poorly relative to random testing?
- Why does UFC's effectiveness as an adequacy measurement vary between systems?

Test Generation Approaches



Weak Coverage Metrics

- Easy to cheat
- Major problem when using counterexample based test generation
  - Counterexamples intended to be simple traces
  - Simple traces make bad tests
- Counterexample based test generation worst case behavior
  - Positive results are positive
  - Negative results are misleading
- Still, satisfying requirements/antecedent coverage not indicative of good tests

Sensitivity to Requirements Structure

- Problem: UFC based temporal and Boolean operators
- DWM_2 system using many relational and arithmetic operators

Original:

```
LTLSPEC G(var_a > (
        case
            foo : 0 ;
            bar : 1 ;
        esac +
        case
            baz : 2 ;
            bpr : 3 ;
        esac
    ));
```

Revised:

```
LTLSPEC G(var_a > (
        case
            foo & baz : 0 + 2 ;
            foo & bpr : 0 + 3 ;
            bar & baz : 1 + 2 ;
            bar & bpr : 1 + 3 ;
        esac
    ));
```


Conclusion

- Evaluated three black box coverage metrics using 4 realistic avionics system
- UFC only useful coverage metric
  - However, UFC is not useful for all combinations of requirements and systems

Questions

UFC Coverage

- $\mathbf{G}(A)^+ = \{A \text{ U } (a \wedge \mathrm{G}(A)) \mid a \in A^+\}$
  $\mathbf{G}(A)^- = \{A \text{ U } a \mid a \in A^-\}$
- $\mathbf{F}(A)^+ = \{\neg A \text{ U } a \mid a \in A^+\}$
  $\mathbf{F}(A)^- = \{\neg A \text{ U } (a \wedge \mathrm{G}(\neg A)) \mid a \in A^-\}$
- $(A \text{ U } B)^+ =$
  $\{(A \wedge \neg B) \text{ U } ((a \wedge \neg B) \wedge (A \text{ U } B)) \mid a \in A^+\} \cup$
  $\{(A \wedge \neg B) \text{ U } b \mid b \in B^+\}$

  $(A \text{ U } B)^- =$
  $\{(A \wedge \neg B) \text{ U } (a \wedge \neg B) \mid a \in A^-\} \cup$
  $\{(A \wedge \neg B) \text{ U } (b \wedge \neg(A \text{ U } B)) \mid b \in B^-\}$
- $\mathbf{X}(A)^+ = \{\mathrm{X}(a) \mid a \in A^+\}$
  $\mathbf{X}(A)^- = \{\mathrm{X}(a) \mid a \in A^-\}$