A Holistic Look at Testing Autonomous Systems

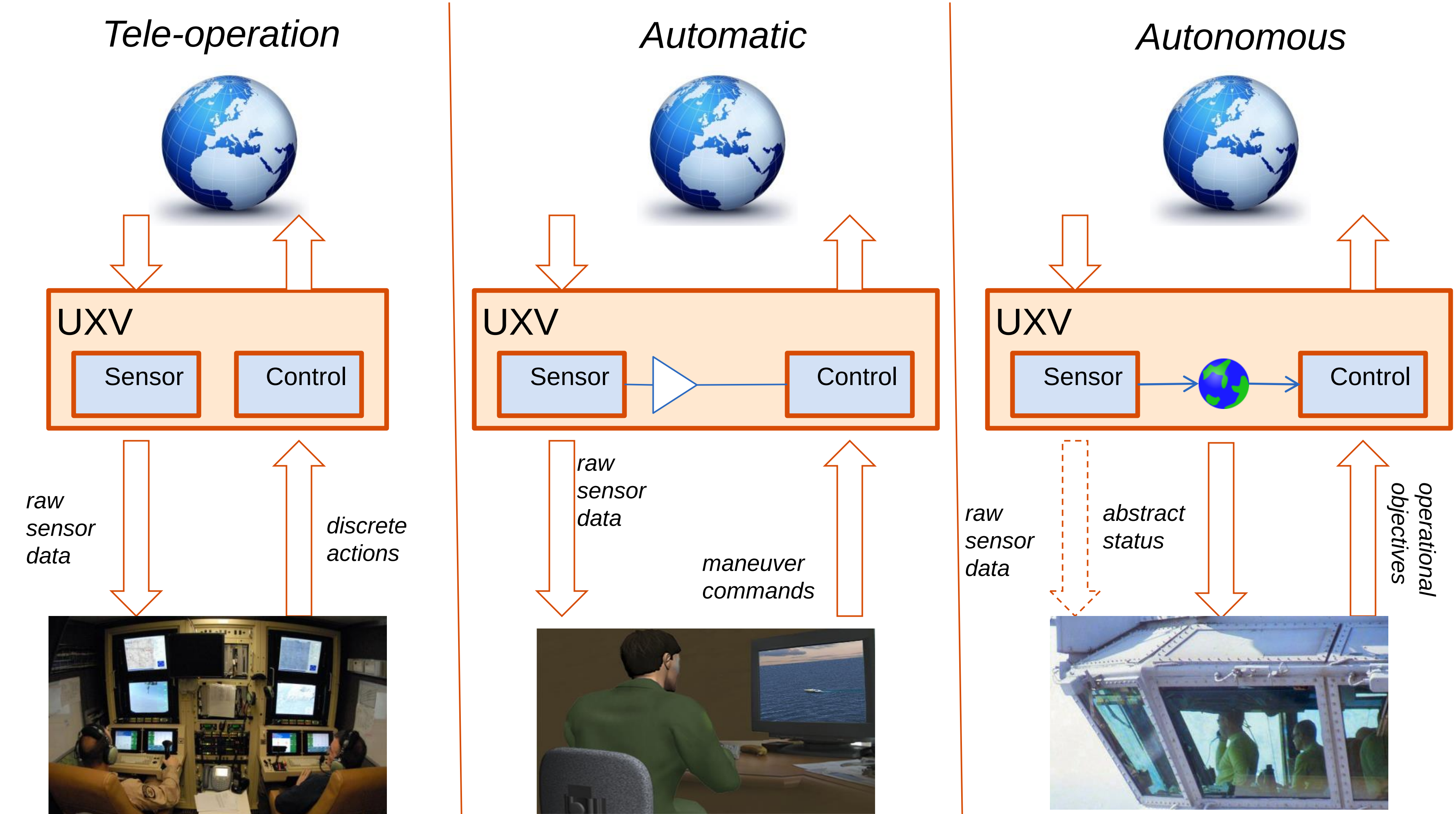
31st Annual National Test and Evaluation Conference

3/3/2016

David Scheidt david.scheidt@jhuapl.edu

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Autonomous Systems



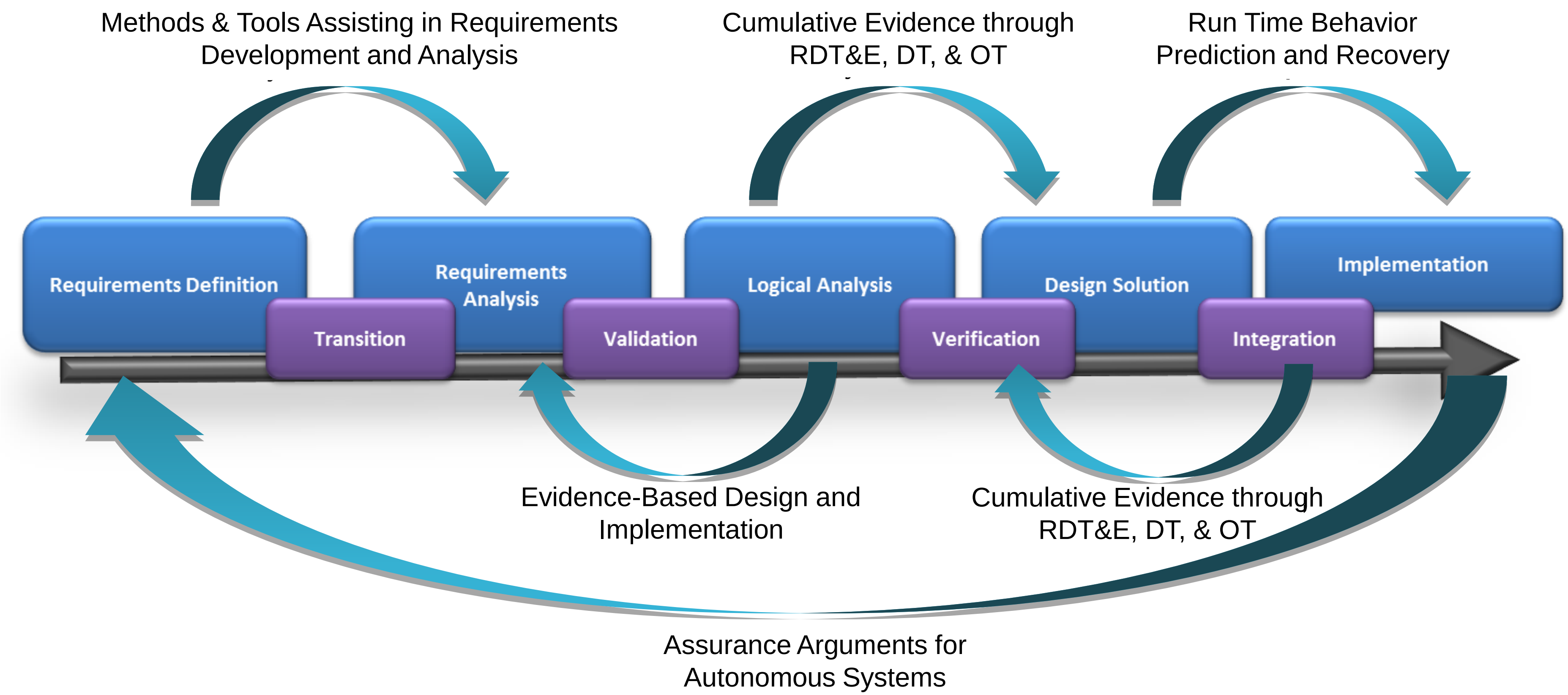
Operator interface similar to pilot to plane interface

Operator interface similar to pilot to autopilot interface

Operator interface similar to commander to pilot interface

Autonomy Test & Evaluation Challenge

T&E Throughout the Entire Engineering Process



OSD Autonomy COI Test Evaluation Verification and Validation Working Group, Technology Investment Strategy 2015-2018, June 2015.

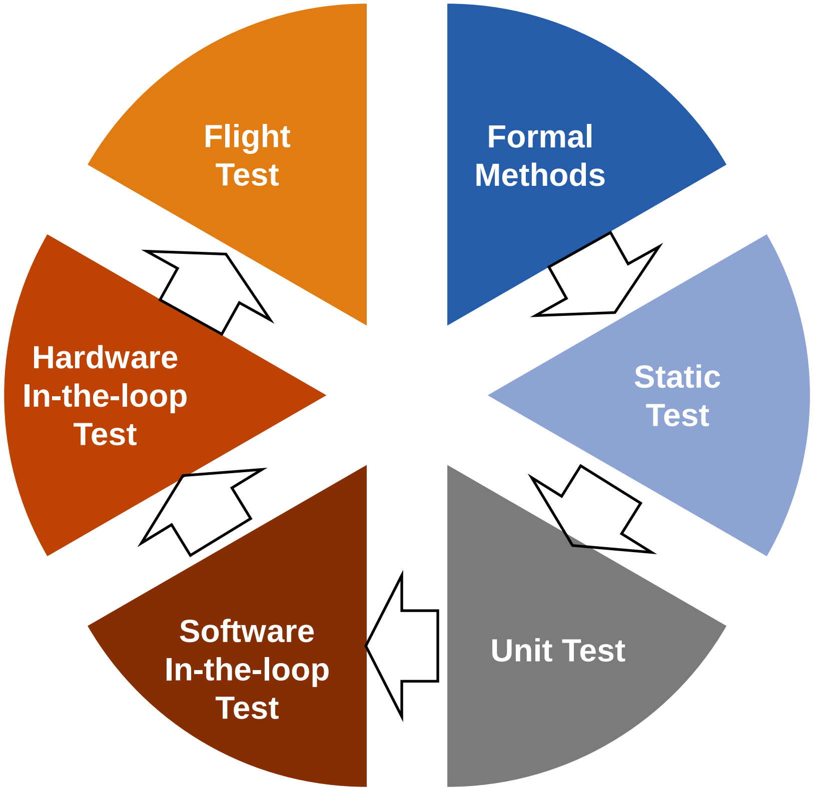
Measuring Autonomous Systems

- **Objective –** Autonomous system metrics are use to provide trust. Measurement-derived analysis should provide operators with insight into mission capability as a function of operating conditions.
- **Measuring the "level of autonomy" is not useful [ref. DSB 2012/2015]**
- **Autonomy == Decisions -** Measuring autonomy requires measurement of autonomous system *decisions* within the context of the system's physical plant and the current operating conditions. Applicable metrics may be derived from:
  - **Command and Control Theory [Alberts & Hayes]**
  - **Control Theory**
  - **Information Theory [Shannon]**
  - **Game Theory and Decision Science**
- Measures of Performance
  - **Mission Objectives that will be satisfied**
  - **Mission Constraints that will be avoided**
- Measures of Effectiveness
  - **Quantitative assessment of MOP**

New analytical methods are required because…

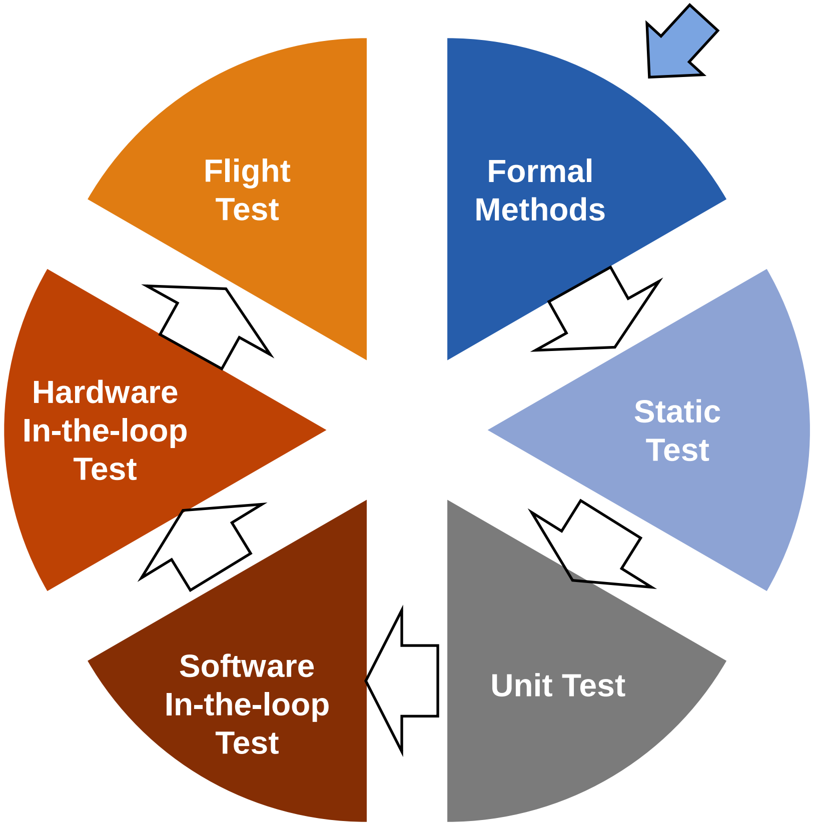
Statistical analysis of autonomous systems operating in adversarial conditions is not valid without an accurate model of the adversary's cognitive performance.

TEVV Process



Torens, C., Adolf, F. (2014), "V&V of Automated mission planning for UAS", *NATO SCI-274 Workshop Verification and Validation of Autonomous Systems*, Imperial College, London, June 24-25.

Formal Methods – Analyzing the Algorithm



Torens, C., Adolf, F. (2014), "V&V of Automated mission planning for UAS", *NATO SCI-274 Workshop Verification and Validation of Autonomous Systems*, Imperial College, London, June 24-25.

Formal methods

"Formal Methods" describes a set of mathematically rigorous techniques for proving properties of software systems.

Theorem Proving – Proves that during an algorithm's execution algorithm desired invariants will hold.

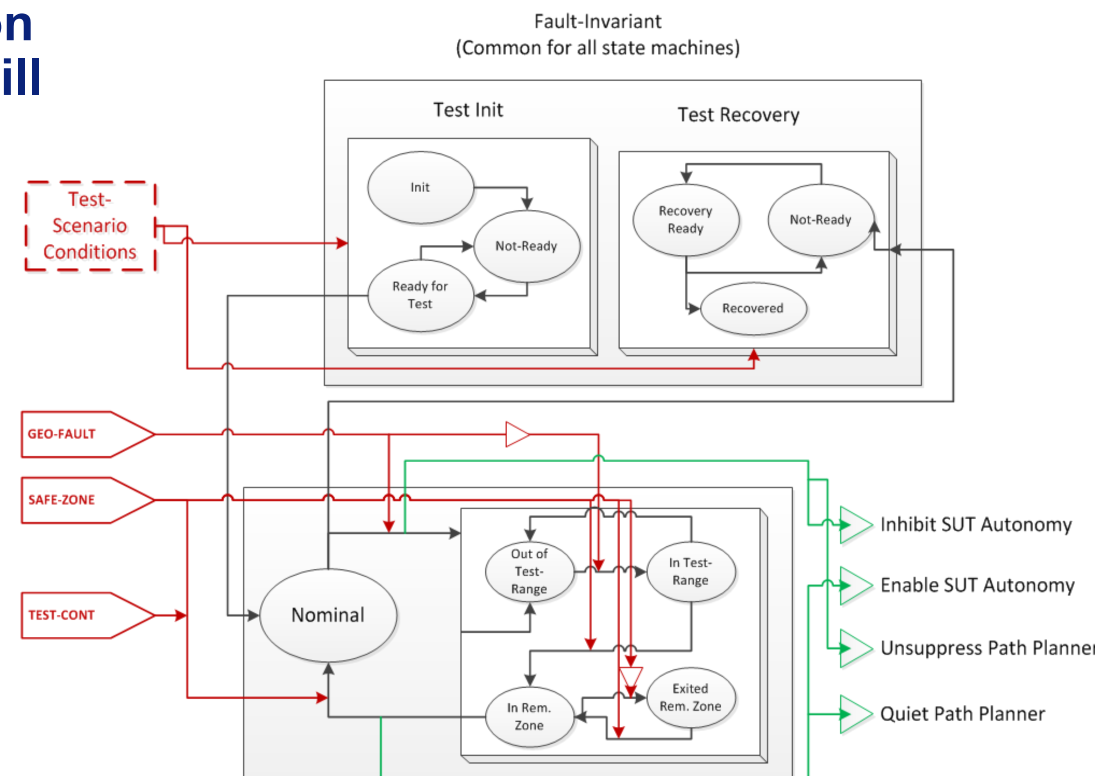
- **Correctness**
- **Satisifiability**

Model Checking – Proves that a model used by the reasoning system exhibits desired properties

- **Self-Consistancy**

Example Hoare Triples

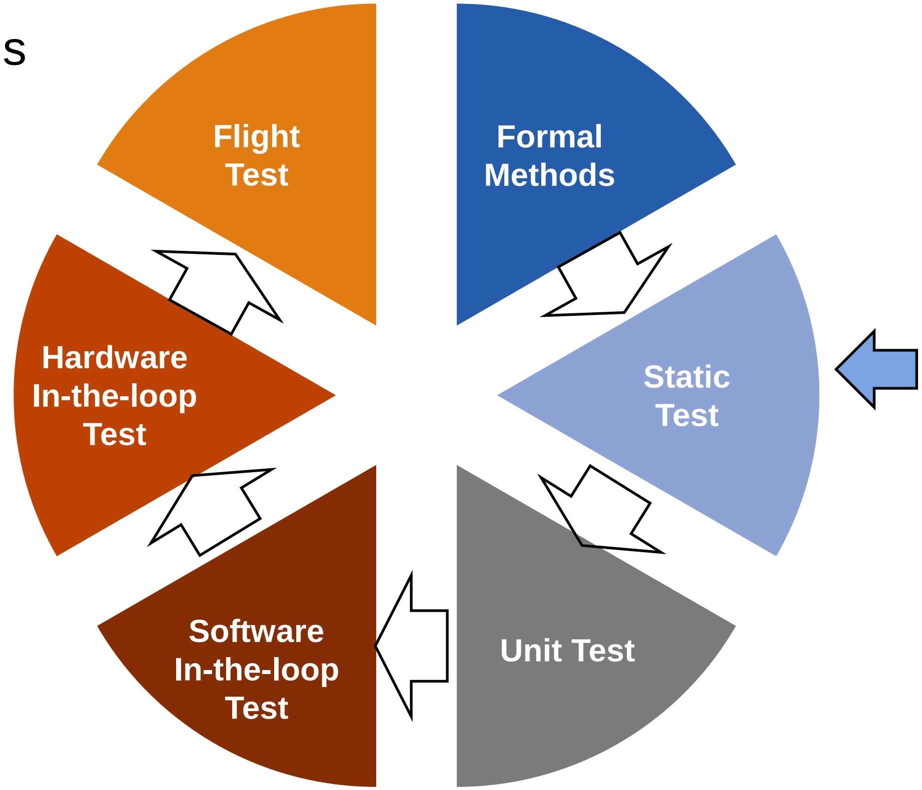
$$true\ \{x := 5\}\ x = 5$$
$$x = y\ \{x := x + 3\}\ x = y + 3$$
$$x > 0\ \{x := x * 2\}\ x > -2$$
$$x = a\ \{if\ (x < 0) then\ x := -x\}\ x = |a|$$
$$false\ \{x := 3\}\ x = 8$$
$$true\{while\ true\ do\ x := x + 1\} false$$



Static Testing – Testing the Implementation of the Algorithm

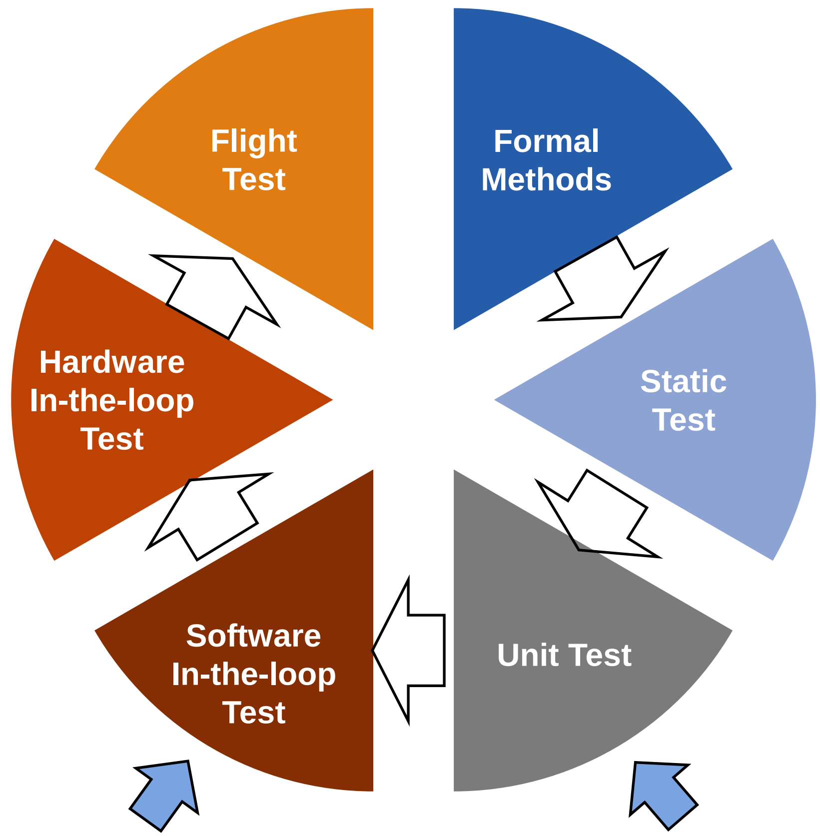
Software Engineering Methods

- Coverage Analysis
- Function Point Analysis



Torens, C., Adolf, F. (2014), "V&V of Automated mission planning for UAS", *NATO SCI-274 Workshop Verification and Validation of Autonomous Systems*, Imperial College, London, June 24-25.

Unit Testing and System-wide Software In-the-loop Testing

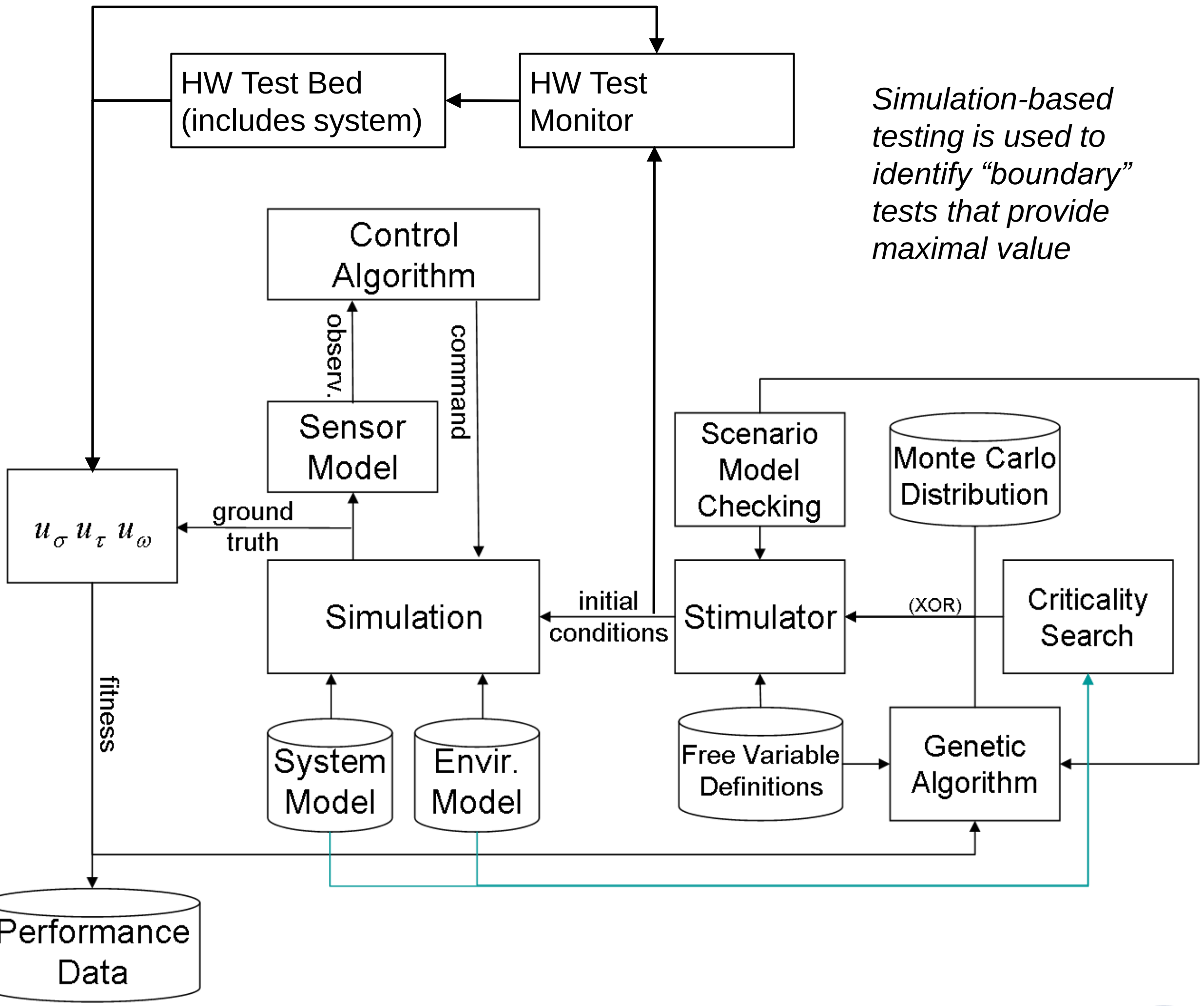


Torens, C., Adolf, F. (2014), "V&V of Automated mission planning for UAS", *NATO SCI-274 Workshop Verification and Validation of Autonomous Systems*, Imperial College, London, June 24-25.

Mixed HWIL and Simulation-based Testing of Autonomous Systems

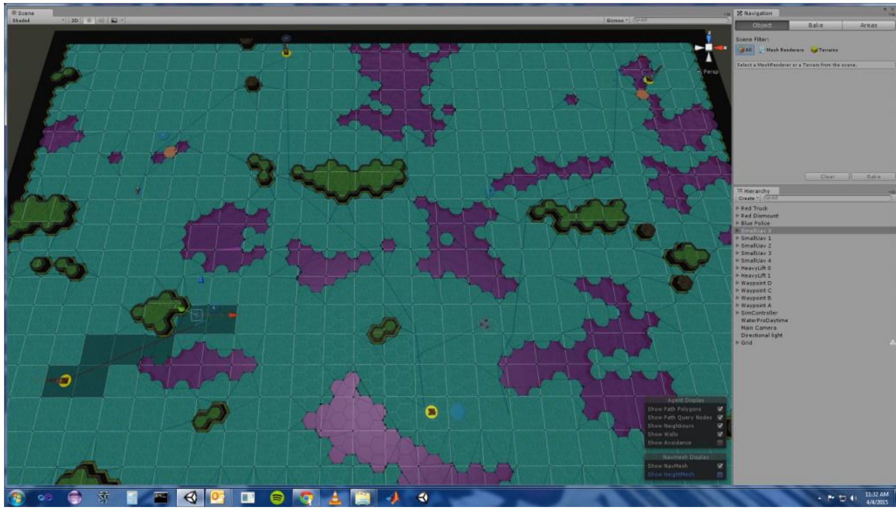
Automated Test Generation

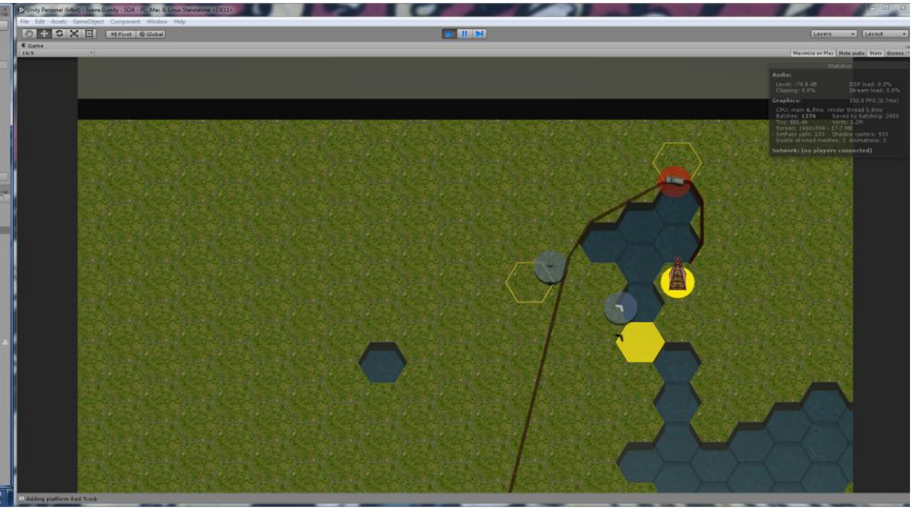
- Monte Carlo
  - Even Coverage
  - Fair Comparison
  - Inefficient
- T-Wise Testing (not shown)
  - Complete n^{th} Order Coverage
  - Efficient
  - Predictable
- Genetic Algorithm
  - Approximates Limitations
  - Highly Efficient
  - Not Predictable
- Criticality
  - Provides Provably Correct Limitations



Simulation-based testing is used to identify "boundary" tests that provide maximal value

Software Simulations for Autonomy Testing





M&S Toolkit that models individual actor knowledge and decision-making

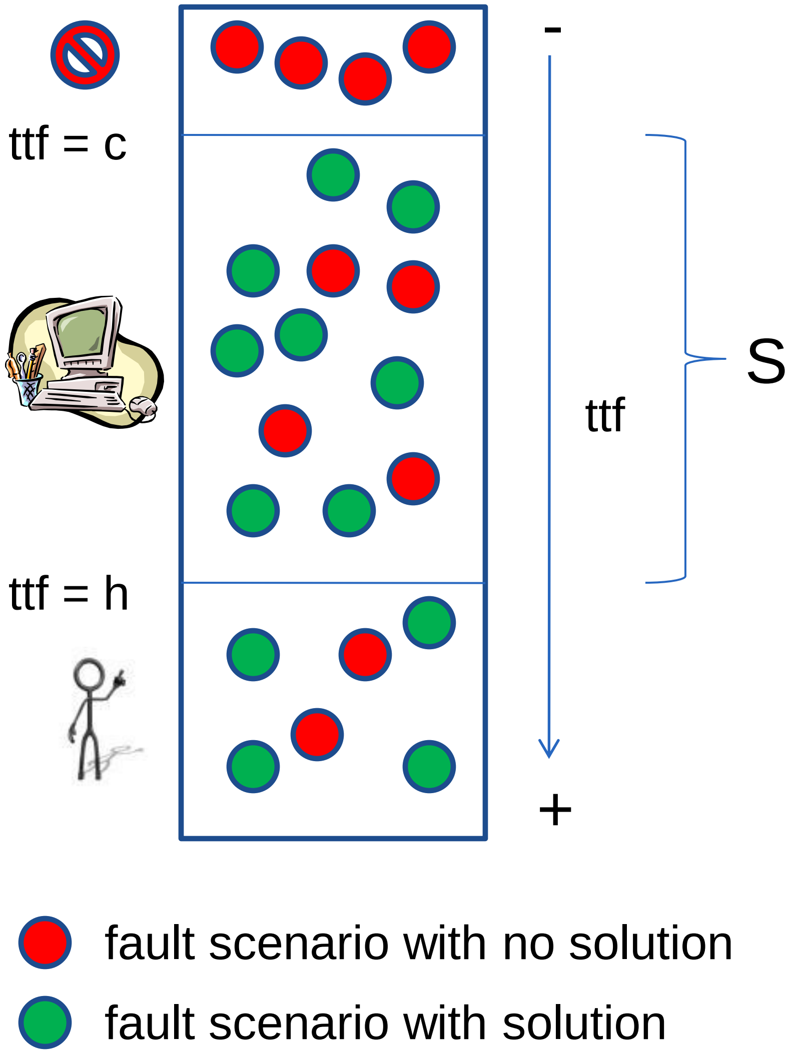
Modeling fidelity must be equal to or greater than the level of fidelity used by the unmanned vehicle’s reasoning engine.

Since cognitive algorithms typically operate with abstractions these tools should by low fidelity

Criticality-based Testing

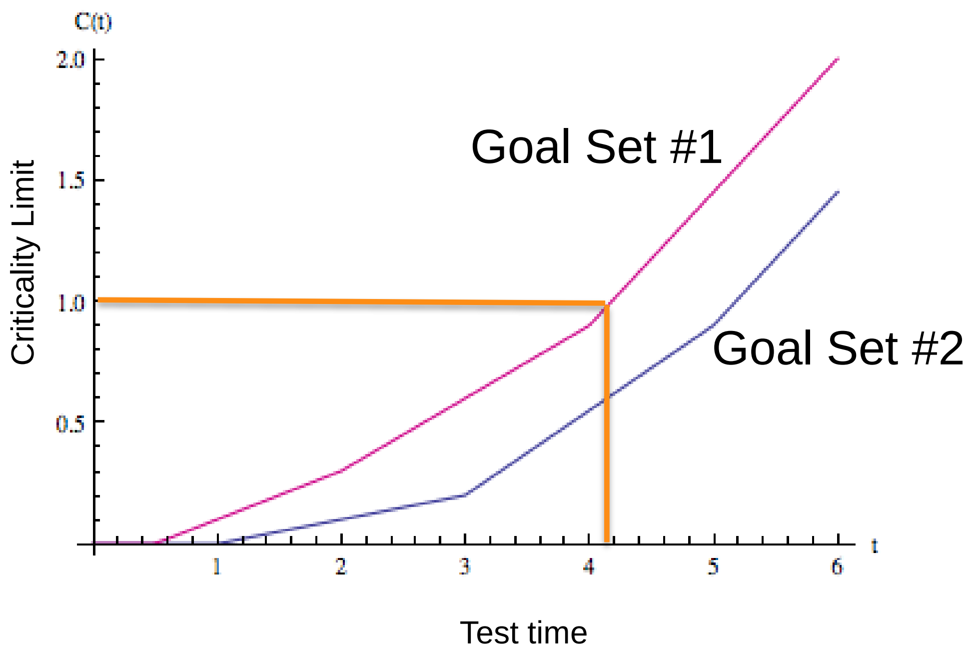
- Although we cannot exhaustively test any controller, perhaps we emphasize test scenarios for which no human intervention is possible:

- A fault scenario will cause a critical failure in ttf seconds
- A human can resolve a fault scenario in h seconds
- A controller can resolve a fault scenario in c seconds. For most faults, we assume c << h
- An ‘ideal’ controller will solve all fault scenarios for which ttf < h
- Thus, identify and test all the fault scenarios S that have a solution and for which ttf < h



ttf = time to mission failure

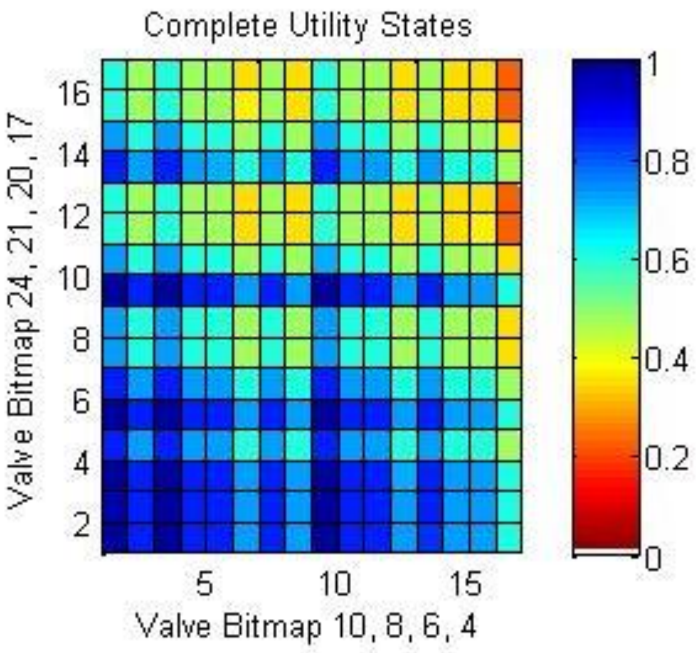
Criticality Testing Metrics



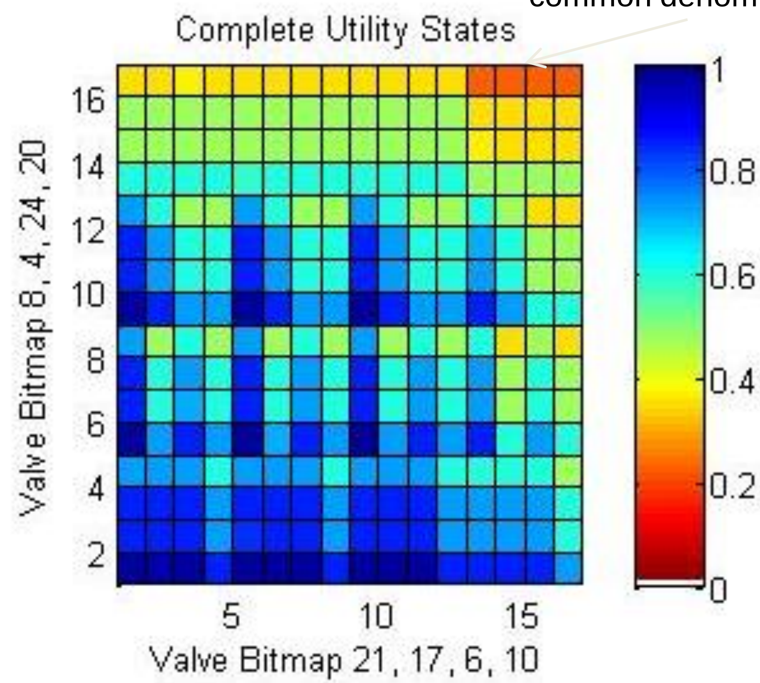
- After 4.17 seconds of testing all failure combinations capable of causing a catastrophic failure within 1.0 seconds had been tested.

Visual techniques for data analysis

Defines risk conditions through the lowest common denominator



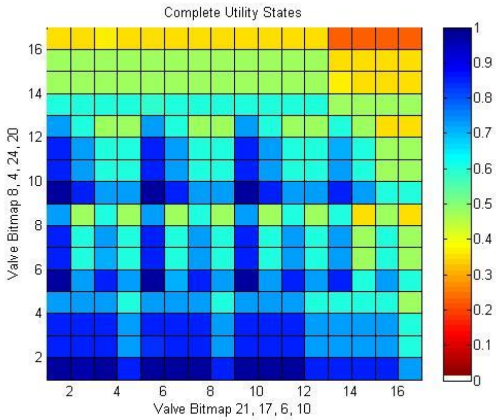
Unprocessed



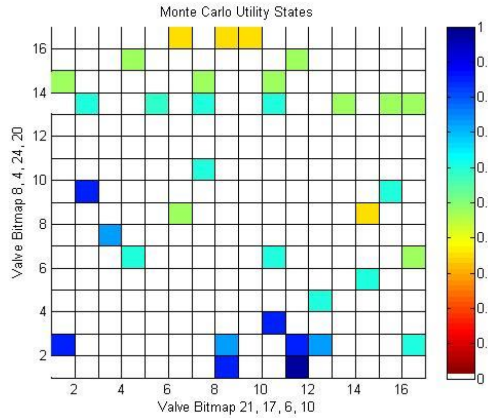
Post-processed

- **Processing a visual map exposes the most influential states**
- **This provides a clustering of critical test cases to be "examined"**

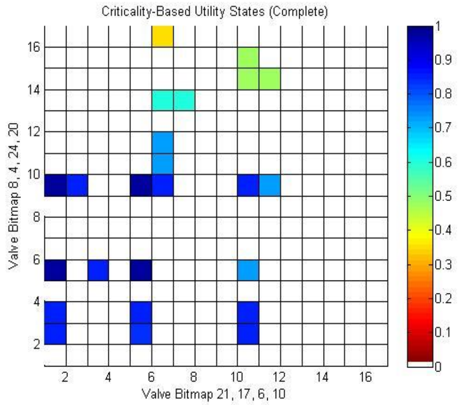
Comparison of Stimulation Techniques



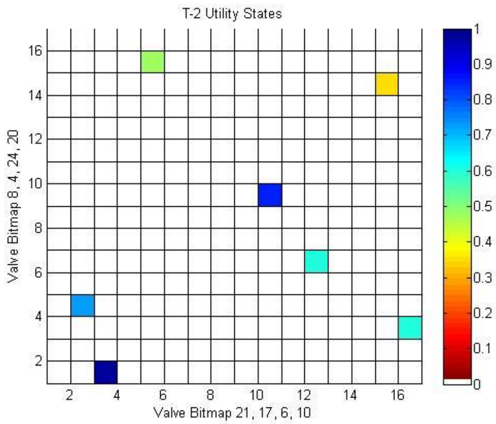
Ground Truth



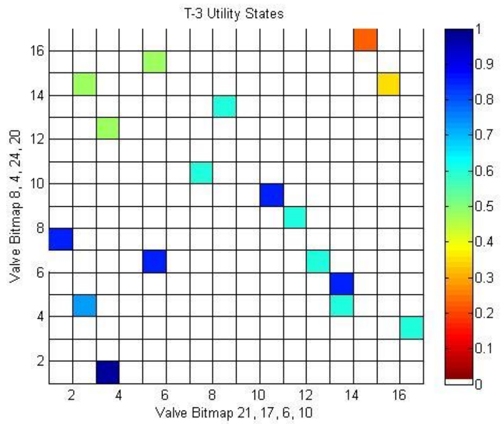
Monte Carlo – 13% exploration



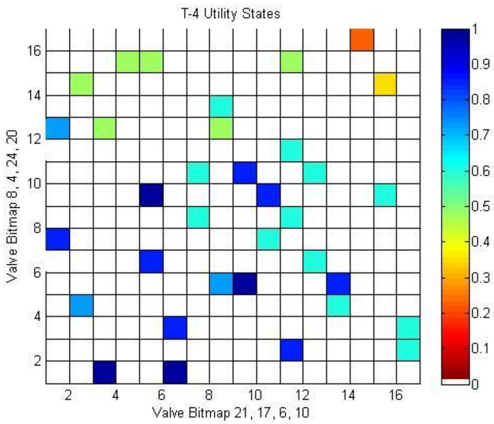
Criticality – 6% exploration



T-wise, 2T – 3% exploration

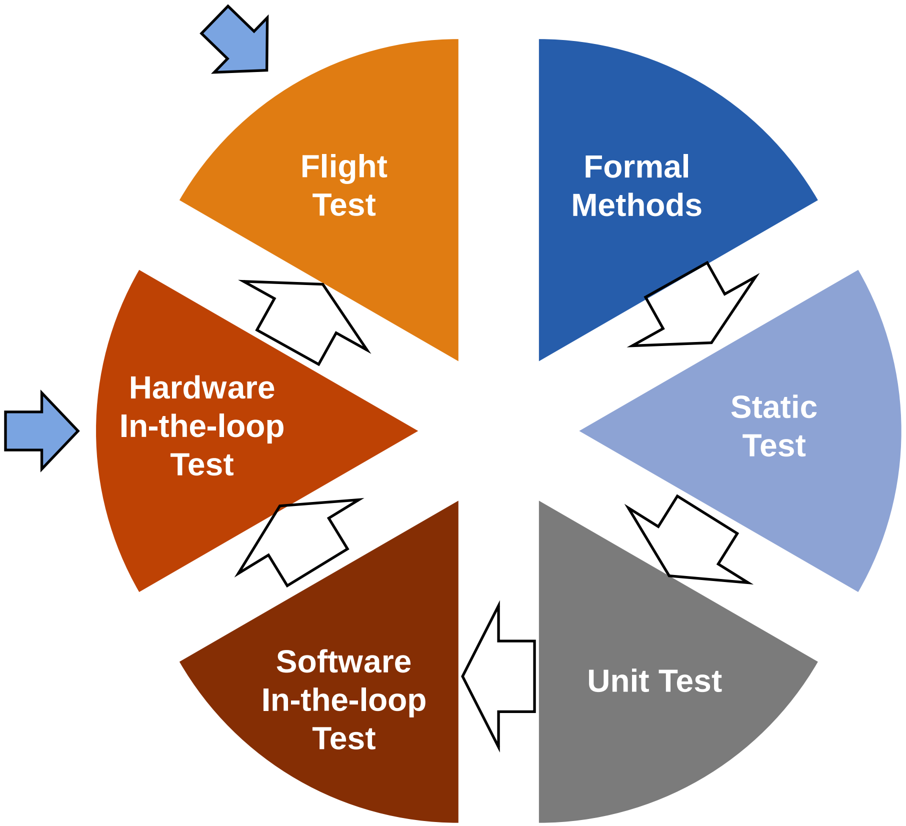


T-wise, 3T – 6% exploration



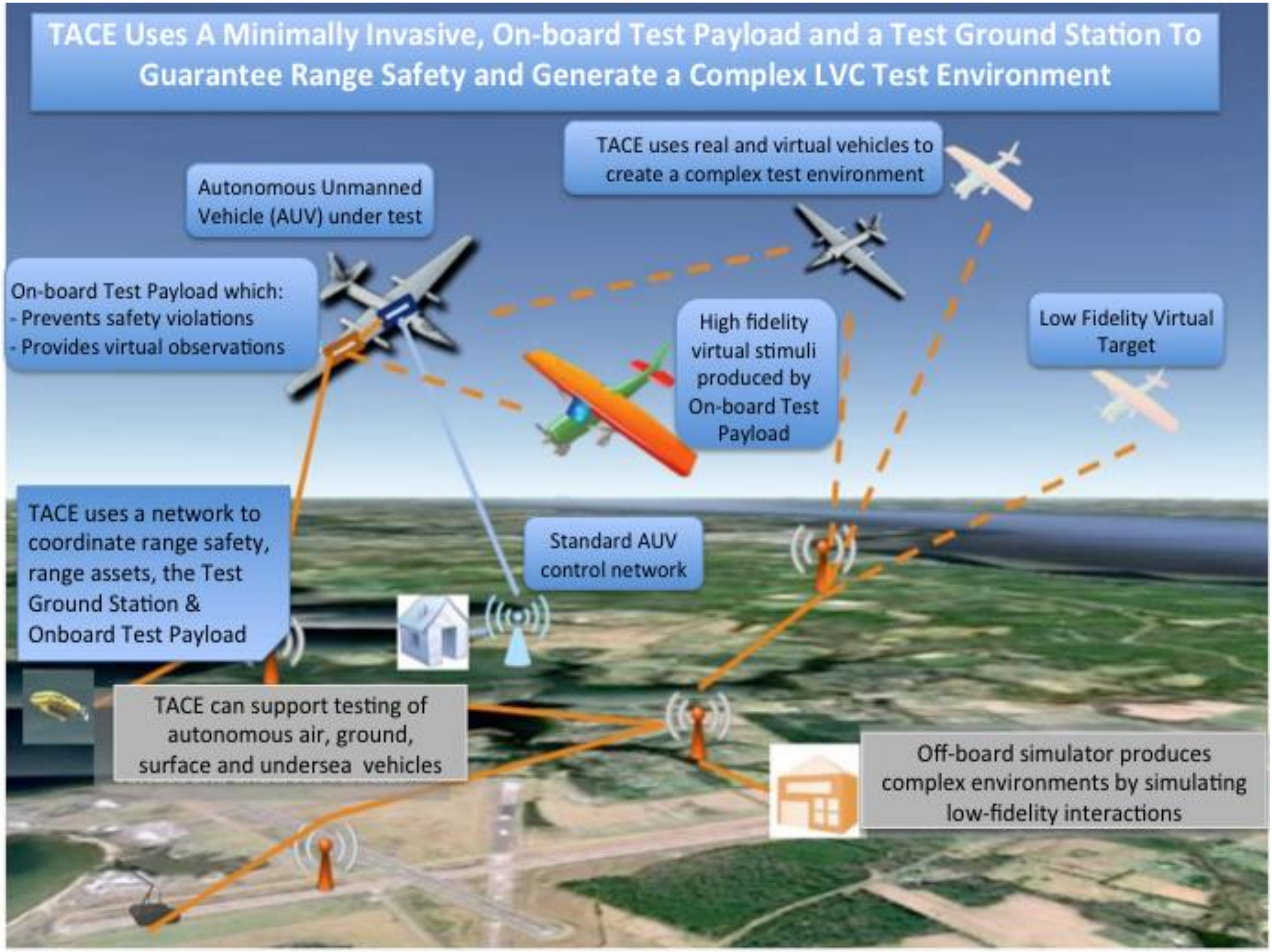
T-wise, 4T – 13% exploration

Hardware in-the-loop Testing (Bench and Flight)

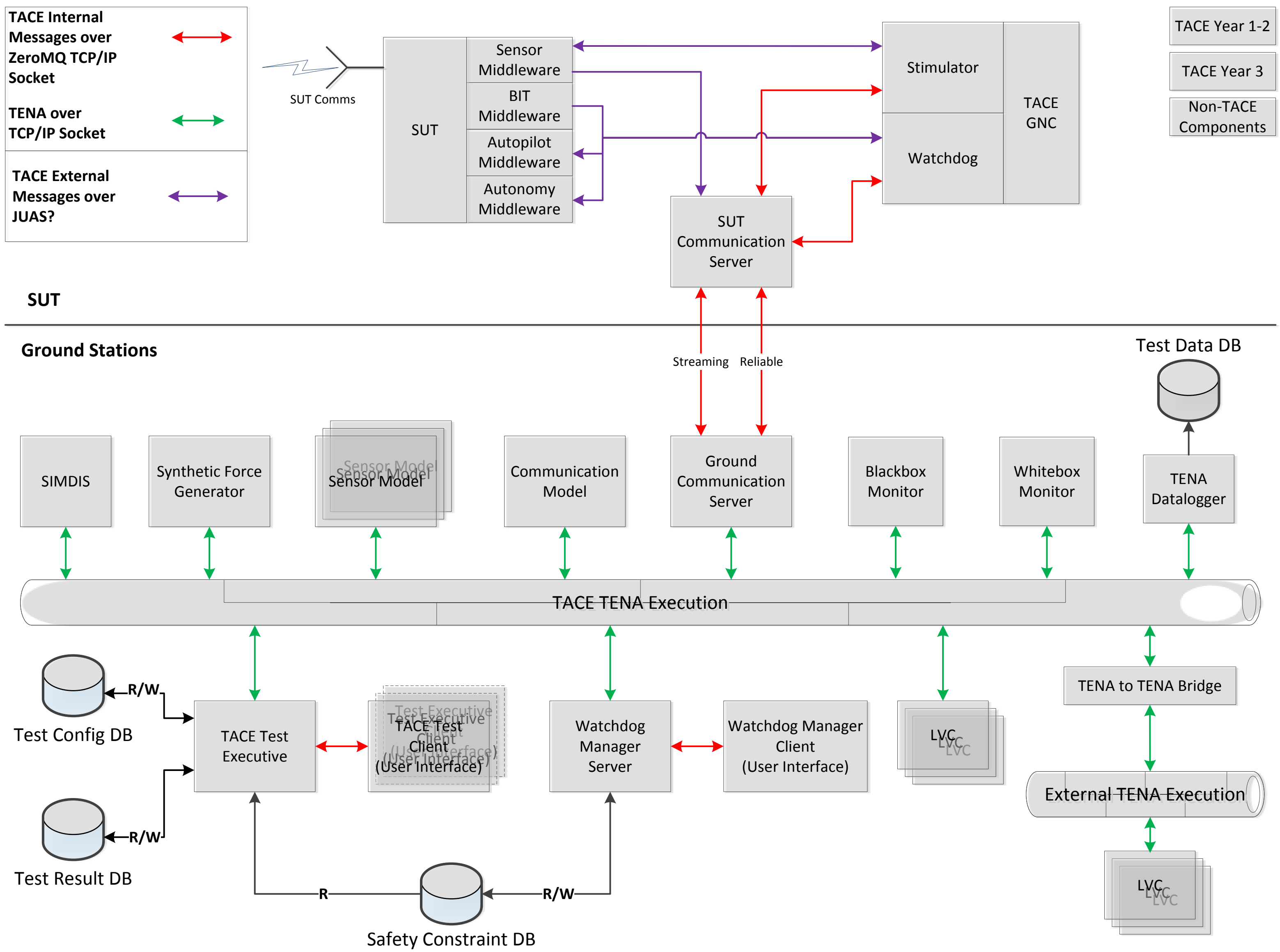


Torens, C., Adolf, F. (2014), "V&V of Automated mission planning for UAS", *NATO SCI-274 Workshop Verification and Validation of Autonomous Systems*, Imperial College, London, June 24-25.

Testing of Autonomous Systems in Complex Environments (TACE)



TACE System Architecture



DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

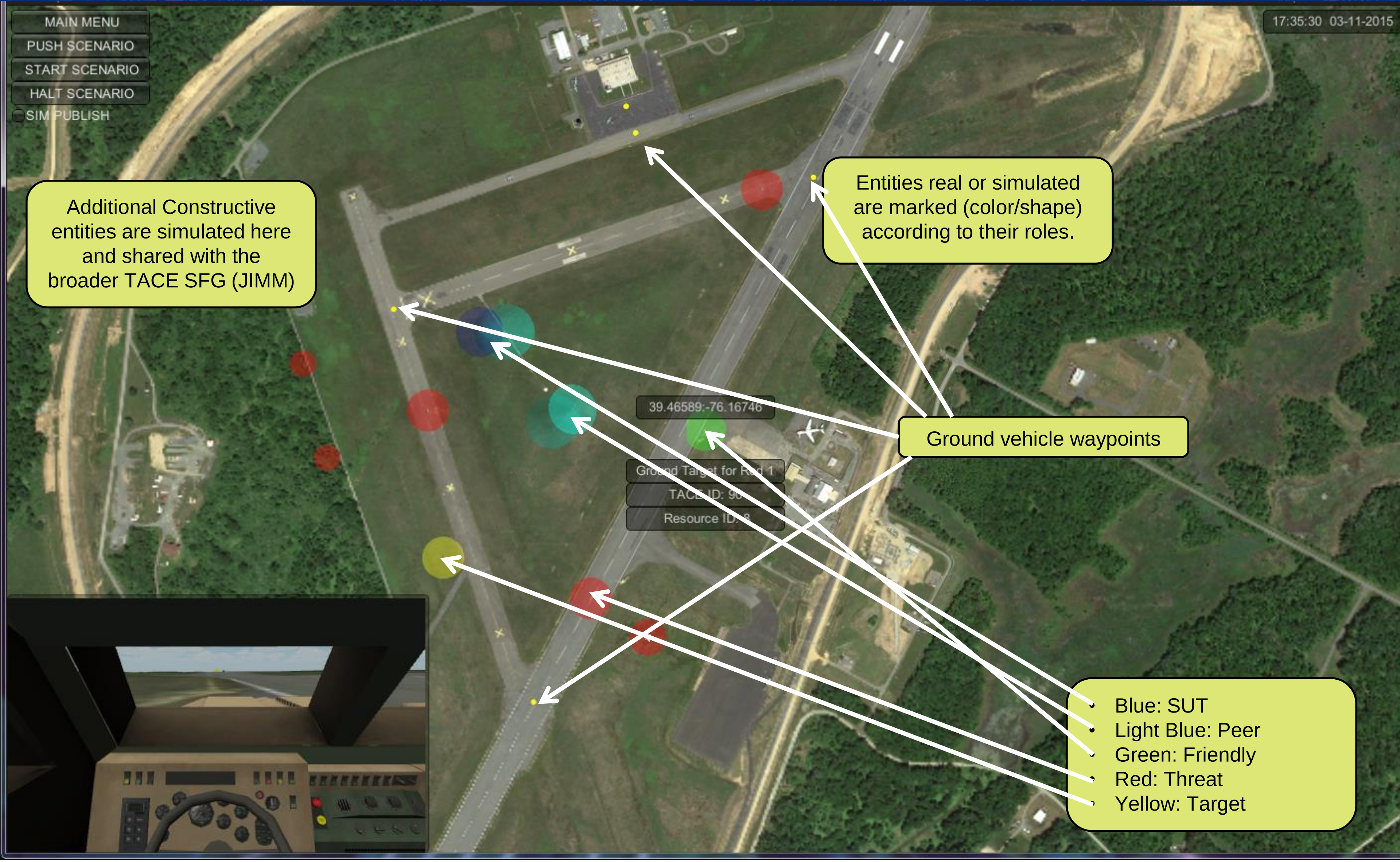
TACE Flight Tests at Aberdeen Test Center

Five Test Events with Multiple Sorties Were Executed during January/February 2014

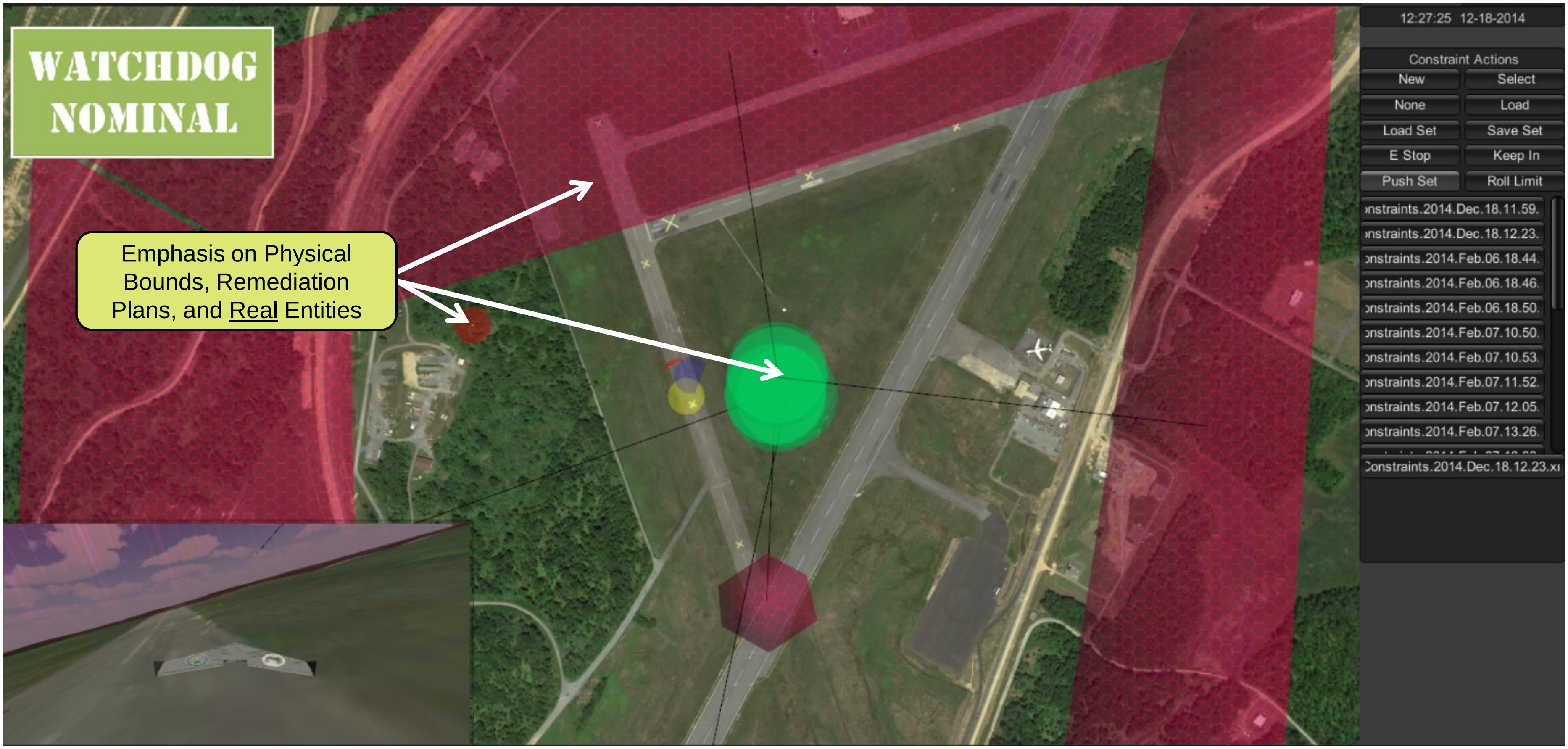
APL Test Team on the tarmac at Phillips Army Airfield (PAAF) Aberdeen Test Center (ATC)

Hand launch of the Procerus research AUV controlled by JHU/APL's Autonomy Tool Kit (ATK)

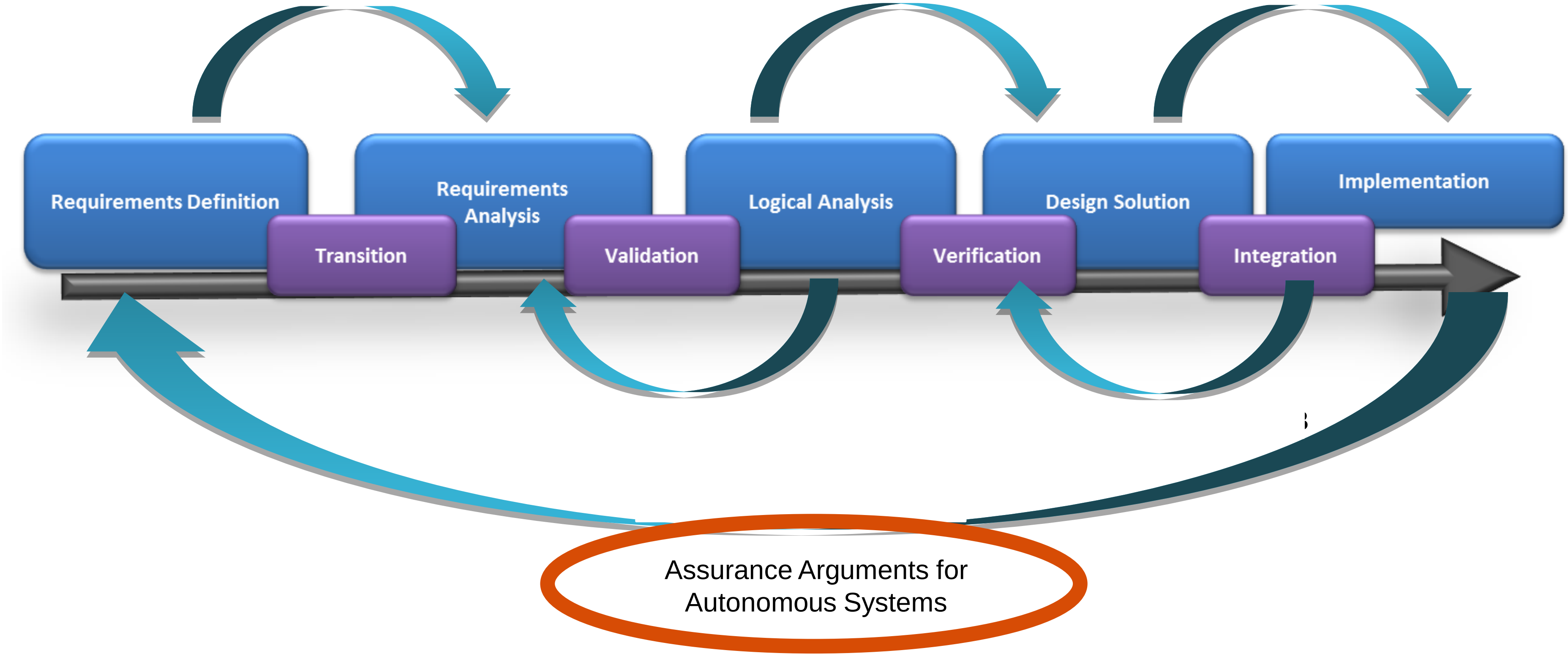
Test Manager Display



Safety Manager Display



Putting it all together – Making the Assurance Argument



Required Research – How do we make a compositional argument that combines

- Licensure – Empirical Evidence from experienced "in the wild"
- Experimental Evidence – Software in-the-loop and Controlled Hardware in-the-loop
- Formal Proof of Correctness

Cognitive Systems Engineering – How do we integrate unit tests into a system-wide argument?

Testing as a Lifetime Sport – For those Autonomous systems that learn, testing doesn't end with operational testing.

Acknowledgements

Thanks to our sponsors:

Test Resource Management Center Unmanned and Autonomous Systems Test (TRMC UAST) Program

ONR Machinery Automation Program

APL

JOHNS HOPKINS

APPLIED PHYSICS LABORATORY