Developing Reliable Software is Impossible!

Dr. Ricky E. Sward

The MITRE Corporation

Developing Reliable Software is Impossible!

- We can't ensure operational systems are 100% free of errors
- NIST 2002: SW errors cost the US ***$59.2 billion*** annually
- How do we build trust and reliability in software?
  - Increase the amount of Testing, Validation and Verification
  - Enforce strict policies and procedures
  - Use Formal Methods to prove certain aspects of the software

Civilian uses for Unmanned Aircraft are emerging.

- Unmanned Aircraft Systems (UAS)
  - May include safety critical sub-systems, such as flight control and navigation systems
  - Civilian uses are emerging and UAS may soon fly in manned airspace

- In US, FAA approves all aerospace software-based systems using DO-178C
  - Formal Methods activities can replace V&V
  - UAS makers have already started adhering to DO-178C
  - May include Formal Methods in development process

Crop dusting UAS in Japan

Developing Reliable Software is Impossible!

- UAS software development market is very competitive, must deliver
  - Need ways to quickly train software engineers on Formal Methods
  - Need processes that incorporate Formal Methods in development
  - Need tools for proof obligations and formal requirements specification

- UAS Command and Control (C2)
  - Man-on-the-loop versus Man-in-the-loop
  - Relieves UAS operator of mundane, tedious tasks
  - Must scrutinize systems for reliability, level of trust

Many universities approved to fly drones

- Increased requirements for security testing and certification
  - Beneficial to include requirements as part of the SW process
  - Formal Methods achieve higher security assurance levels

- Building error-free, reliable software may indeed be impossible
- Formal Methods help reduce V&V, increase assurance levels
- Provide tools and processes to build Formal Methods into software development