

# Current Understanding

---

- Science Team tests science
- For strings of science routines the science testing must be end-to-end
- There are lots of things inside science algorithms that are not “science”
- Testing is going to have to be a **cooperative effort**

# Software Test Process

---

- Evolving understanding of test process
- Shared responsibility, TM, SDST, ECS
- More work than you may have planned
- Requires test data sets

# Example Problems - 1

---

- Science team may not know how to provide test inputs to drive all of the options in preceding parts of processing strings
- SDST does not know how to provide test inputs to drive all of the options in the science code

# Example problems - 2

---

- Science testing may not be enough to assure code is robust in operational sense:
  - Did you test for correct date and input?
  - Did you check for ancillary data flaws?
  - Did you check for missing data?
  - Did you check all the QA flags from preceding steps?

# What is Coming -1

---

- Interaction between Science Team and SDST to define testing program
- Science Team to provide information about tests they have conducted:
  - what was tested
  - test data sets
  - test results

# What is coming - 2

---

- SDST to provide suggestions about what things (other than science) should be tested
- SDST and Science Team to work out test data needs
- SDST to assist in providing test data.
- SDST to conduct flow and string tests, evaluation to be joint responsibility