

Global Precipitation Measurement IV&V

GPM IV&V - ITC Testing Process and Analysis
September 14, 2011

Amy Robinson

amy.l.robinson@ivv.nasa.gov

681-753-5297

Bryan Cox

bryan.a.cox@ivv.nasa.gov

681-753-5330

Adel Lagoy

adelbert.c.lagoy@ivv.nasa.gov

281-317-7472



IV&V PM: Eric Sylvania
Lead Analyst: Amy Robinson

Developer: GSFC/JAXA
Launch Date: July 2013



Mission Objectives:

- Advance precipitation measurement capability from space
- Improve knowledge of precipitation systems, water cycle variability, and fresh water availability
- Improve climate and hydrometeorological modeling, prediction, and 4-D climate reanalysis

Instruments:

- Dual-Frequency Precipitation Radar (DPR)
- GPM Microwave Imager (GMI)

NASA Center: **GSFC**

Website: gpm.gsfc.nasa.gov

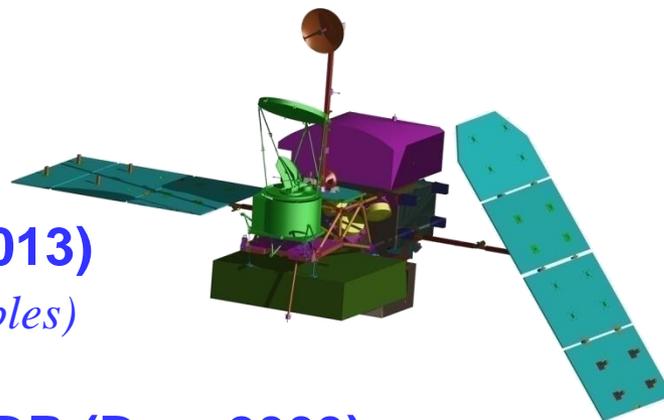
Major Developers: **JAXA; Ball; GSFC**

Launch: **Tanegashima (July, 2013)**

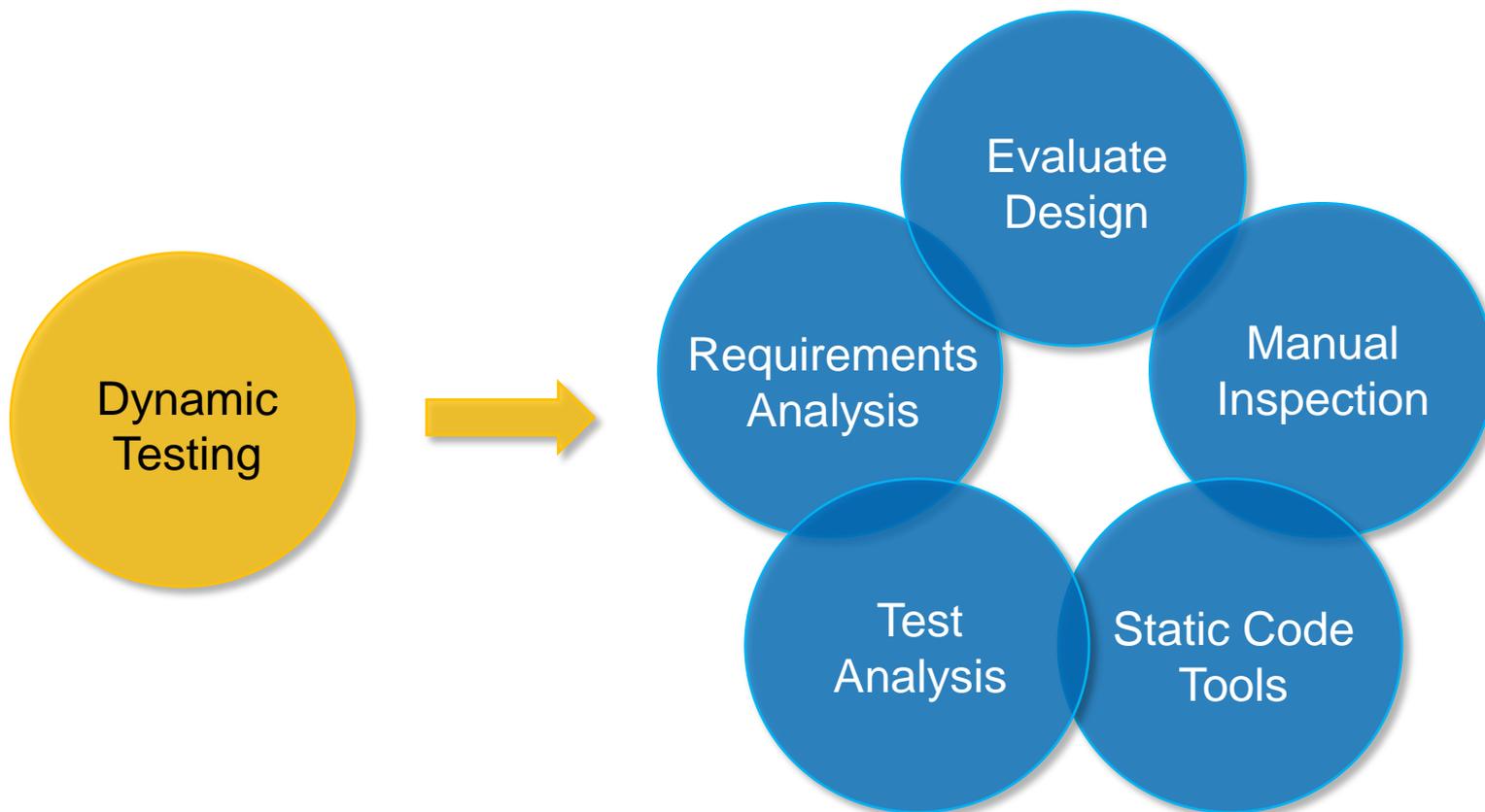
Mission Duration: **3 years** (*5 years consumables*)

Mission Phase: **Implementation**

Mission Status: **Completed Mission CDR (Dec, 2009)**



- How can dynamic testing complement current IV&V analyses?

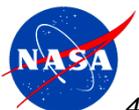


How does dynamic testing fit into IV&V?



- **To perform dynamic testing, we need to come up with tests**
 - Expand the project's test scope
 - Based on delivered test artifacts
 - Determine if project's tests completely verified requirements
 - IV&V generated additional scenarios to verify implementation
 - Avoid duplicating project tests
 - No additional value
 - Off-nominal/adverse condition testing
 - Scenarios to determine if the software behaves appropriately to
 - Invalid commands
 - Invalid data received from other onboard software or hardware entities
 - Invalid table uploads
 - Other vulnerabilities to handling bad inputs
 - Operator awareness
 - “What if?” scenarios
 - Explore unintended behaviors in the software
 - What behaviors does the software allow and are they appropriate/desired?

IV&V testing provides additional test coverage



requirements
test artifacts

Expand the project's test scope

Are scenarios defined to fully verify requirements?

Off-nominal/adverse condition tests

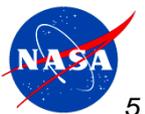
Does project's testing cover adverse scenarios?

"What if?" scenarios

Does the software let me do...?

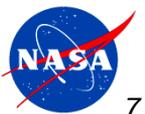
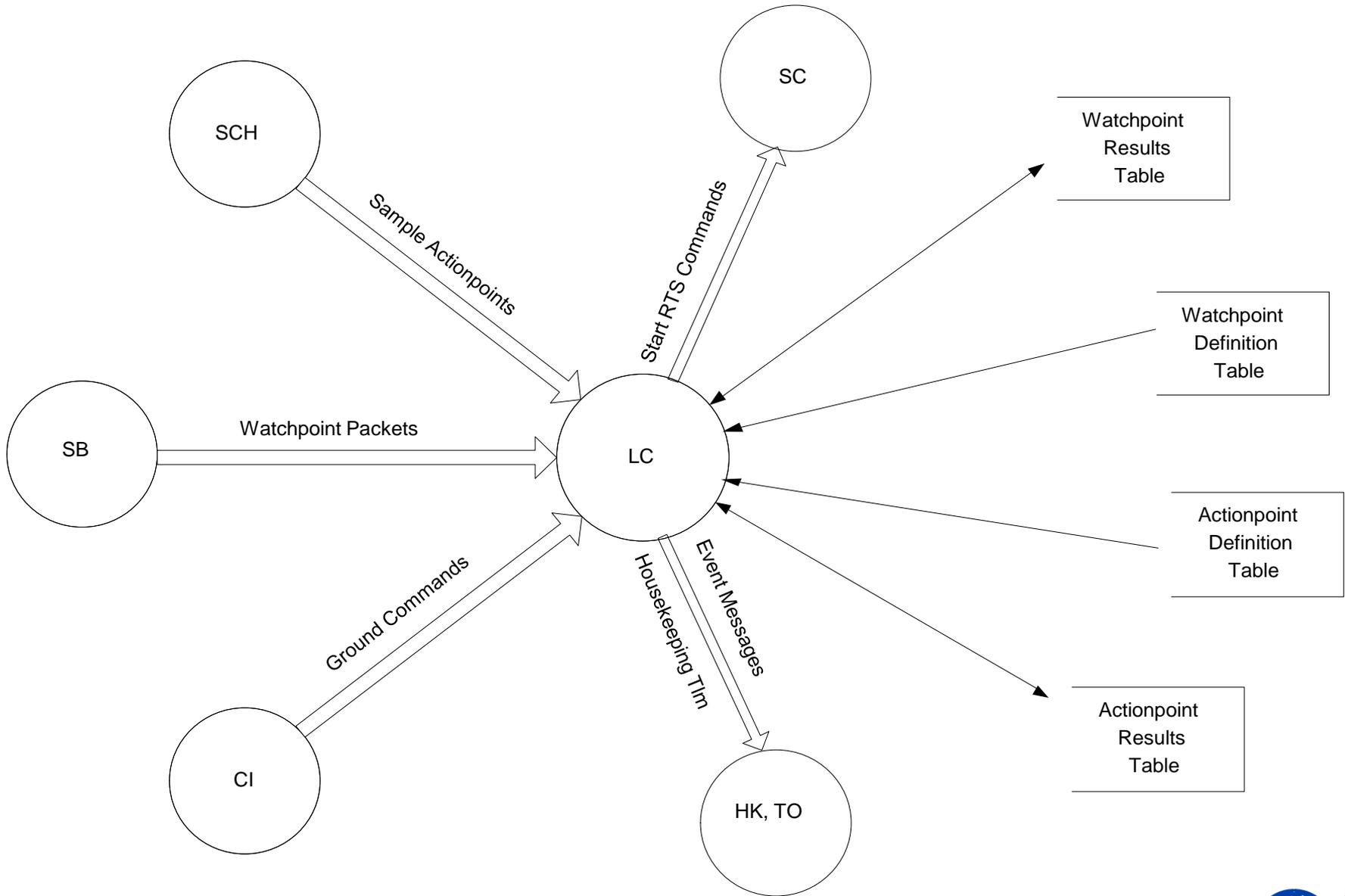
IV&V test scenarios

IV&V testing builds on the project's test campaign



- **Limit Checker (LC) application**
 - Monitors data and takes action when necessary
 - Essential to Fault Detection and Correction capabilities
- **Why did IV&V choose LC to test in ITC?**
 - Ability to test in current ITC configuration
 - Only requires GPM FSW, no other hardware/sensor/effector simulations needed
 - GPM project test results availability
 - LC is CFS
 - Mature application even at build 2.0 (early GPM build)
- **Approximately 140 test cases performed by the project**
- **IV&V performed traditional analysis on LC**
 - Requirements validation
 - Manual requirements implementation analysis
 - Static code analysis using Flexelint, Klockwork, CodeSonar, Understand
 - Test analysis performed in parallel to dynamic testing





- Tables define the data points LC needs to monitor
- Test results showed not all combinations demonstrated

		<		<=		>		>=		=		!=	
		pass	fail										
8-bit	signed	Green	Yellow	Yellow	Yellow	Yellow	Green	Green	Green	Yellow	Yellow	Yellow	Yellow
	unsigned	Yellow	Yellow	Green	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Yellow	Yellow
16-bit	sign-BE	Yellow	Yellow	Green	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Green	Yellow
	sign-LE	Yellow	Yellow	Green	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Yellow	Yellow
	un-BE	Green	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Yellow	Yellow	Green	Yellow	Yellow
	un-LE	Yellow	Yellow	Yellow	Yellow	Green	Yellow	Green	Yellow	Yellow	Yellow	Green	Yellow
32-bit	sDou-BE	Yellow	Green	Green	Yellow								
	sDou-LE	Green	Green	Yellow	Green	Yellow	Yellow						
	uDou-BE	Yellow	Yellow	Green	Green	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Yellow
	uDou-LE	Yellow	Yellow	Yellow	Green	Yellow							
	float-BE	Yellow	Green	Green	Yellow	Yellow							
	float-LE	Yellow	Green	Yellow	Yellow	Yellow	Yellow						

BE – big endian, LE – little endian

- LC performed as expected

 project test
 IV&V test

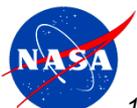
In this example, IV&V added 113 test cases



- **Since LC accepts operator commands, how does it react to invalid ones?**
 - Requirements specify response to bad commands
 - Test analysis shows this was mostly done by project
 - IV&V's additional test cases included
 - 1 test for rejecting a command with an invalid message length
 - 3 tests for command parameter boundaries/invalid command parameters
 - 21 tests for disallowed application and actionpoint states/invalid transitions
- **LC monitors incoming bus messages, how does it react to bad ones?**
 - Bad sample AP #
- **LC is table driven, how does it react to invalid table data?**
 - Additional testing by IV&V to investigate table validation behavior
 - 4 test cases performed by project
 - Approximately 70 additional tests performed by IV&V

- **Dynamic testing allows the analyst to see how the software “really” works**
 - Helps analyst come up with “what if” scenarios
 - Helps analyst explore unintended behaviors in the software
 - Requirements/documentation may not be clear on how software should behave in all circumstances
- **Re-commanding LC app state/Actionpoint states**
 - No requirement/design data indicates this should be prevented/allowed
 - Watchpoint/Actionpoint behavior- is this the correct?
- **Uploading Watchpoint/Actionpoint definition tables with empty entries**
 - Is allowed by the flight software- is this correct?
 - Or should this be added as a check in the table validation function?
- **Uploading the HS application monitoring table with no entries**
 - A blank table prevents HS from monitoring any applications for suspicious behavior
 - Should this be prevented?

IV&V analyzed unintended behaviors through dynamic testing

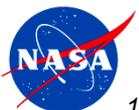


- **Did dynamic testing provide IV&V additional confidence that the SW behaves as expected?**
 - Software operated nominally as it should(Q1)
 - Mostly covered by project
 - Software either blocked bad data or handled the invalid data in an appropriate way(Q2, Q3)
 - IV&V increased this type of testing coverage
 - Software works as expected under nominal and adverse conditions
 - We expect LC to be at this state given that it is CFS code that is reused and reviewed over again by other projects
- **How does dynamic testing increase confidence of the FSW?**
 - With dynamic testing, we can see the code actually functioning and processing the way we expect it to
 - We also receive a more accurate understanding of how the code is working at real time and how it responds to bad data or improper processing
 - We have more confidence that the code will behave properly during flight

- **How can dynamic testing complement current IV&V analyses?**
 - Static Code Analysis
 - Easily find issues such as buffer overflows and null pointers
 - Dynamic testing allows IV&V evaluate the issue impact more accurately
 - Also allows IV&V to prove actual issues exist when there is a dispute
 - Manual Inspection
 - Determine if all requirements are implemented correctly
 - May also find additional code not traceable to any documentation
 - Dynamic testing allows IV&V to perform additional tests more quickly than analytical methods
 - Also allows us to determine that these tests perform as we expect
 - Test Analysis
 - Used in developing what to test in dynamic testing
 - Needed so we don't retest the same things as the project
 - Dynamic testing adds an additional layer of confidence in the judging the code as ready for flight

- **What are the advantages of dynamic testing on GPM's FSW? Limitations?**
 - Advantages of GPM dynamic testing in ITC
 - Increased confidence that the code will work as it should and be able to handle adverse conditions. Seeing is believing.
 - Increase accuracy and knowledge of how the code functions as a whole and how each of the different subsystems function together
 - We can test the limits of the code further than before
 - We can expand the test coverage at our own will and help the project cover more testing if they have time constraints
 - Limitations of GPM dynamic testing in ITC
 - Learning curve on understanding the ground system and test script syntax
 - Availability of simulators that simulate GNC and instrument FSW
 - Not having completeness limits test capabilities
 - Amount of resources needed to develop and run test scenarios

- **Results**
 - Test analysis showed most requirements fully verified by project
 - IV&V performed approximately 220 additional tests for LC
 - Added tests to increase verification confidence
 - Tested off-nominal/adverse conditions
 - Investigated unintended behaviors
 - All test outcomes were as expected
- **Issues/observations from dynamic testing**
 - Issue remains on Watchpoint Results Table not correctly displaying the value that tripped Watchpoint
 - Can inconvenience operator, but LC evaluates Watchpoint properly
 - Issue remains on passive RTS counter does not increment properly
 - Another counter is incremented instead
 - Operator inconvenience
- **Traditional analysis and dynamic testing results indicate that no significant issues exist for the LC application**



Questions?

Bryan Cox

bryan.a.cox@ivv.nasa.gov

681-753-5330

Adel Lagoy

adelbert.c.lagoy@ivv.nasa.gov

281-317-7472

