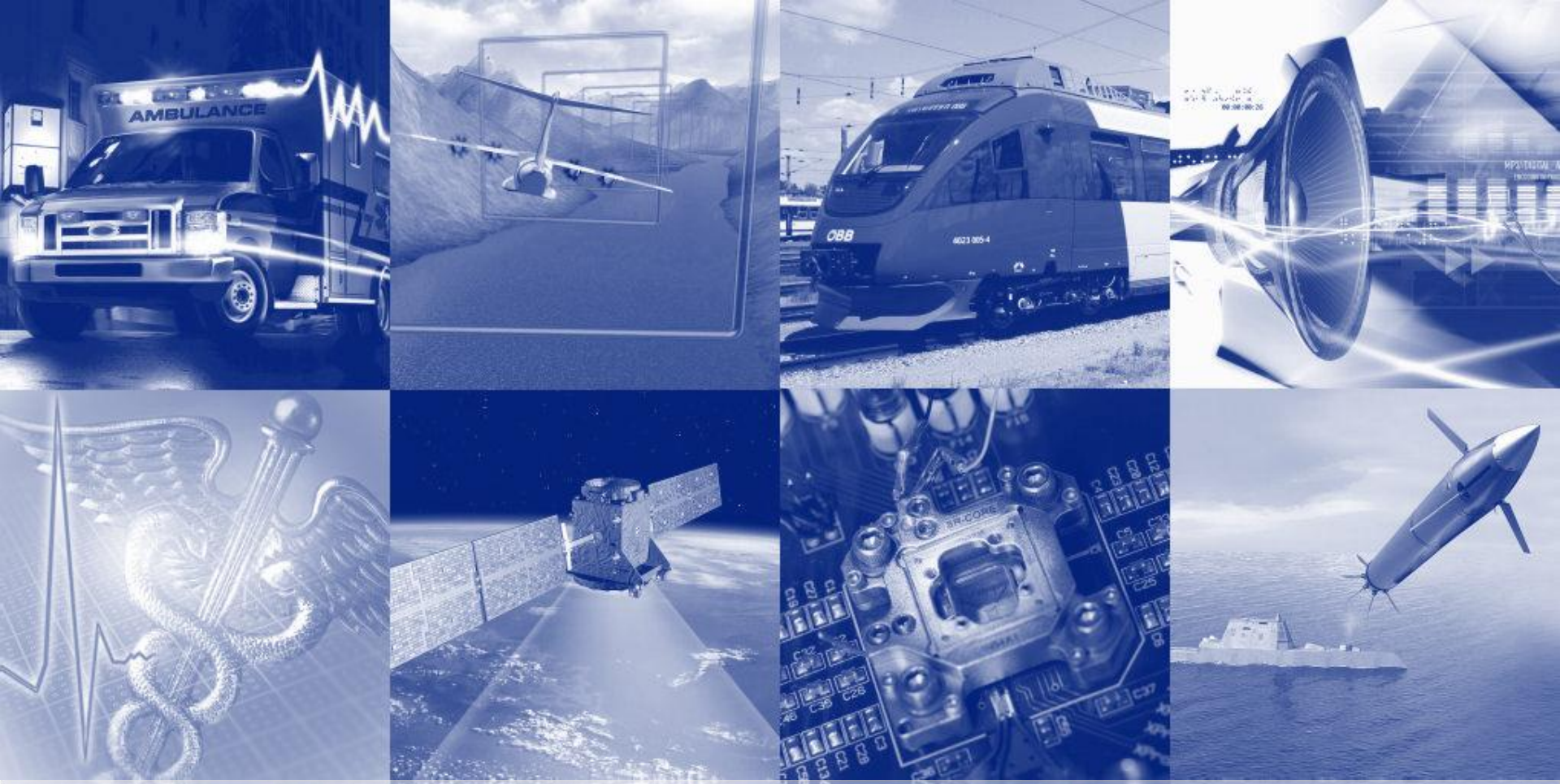
VECTOR software

Using VectorCAST/C++ with Renesas HEW and the RX63N

Ryan Lavering
Field Applications Engineer

Agenda

- About Vector Software
- Overview of VectorCAST tool suite
  - Unit / Integration Testing
  - Code Coverage Analysis
  - Target and Simulator Testing
  - Continuous Integration
- Lab Intro and VectorCAST Demonstration
- Interactive Lab
- Wrap-up



About Vector Software

VECTOR software

Bio – Ryan Lavering

- Field Applications Engineer, Vector Software
- B.S. Computer Engineering, Cal Poly, San Luis Obsipo
- Seven years experience in embedded industry
- Expertise:
  - RTOS
  - Compilers
  - Debug
  - Integration
  - Testing

Our Company

- Vector Software develops embedded software testing products
  - U.S. Based company founded in 1990 by embedded developers
    - *First product released in 1994 for Lockheed Martin's C-130J "Super" Hercules*
  - Corporate headquarters in Rhode Island
  - Sales and support office worldwide
    - *Providence*
    - *Boston*
    - *Baltimore*
    - *Jacksonville*
    - *Denver*
    - *Phoenix*
    - *Los Angeles*
    - *Düsseldorf*
    - *London*

solid reference accounts in multiple industries

Aerospace | Defense	Automotive	Medical	Railway	Industrial
DO-178B | ED-12B	ISO 26262	FDA | IEC 62304	CENELEC | EN 50128	IEC 61508

Our Clients

Aerospace | Defense

Boeing
Bell Helicopter
Belcan
BAE Systems
Cessna Aircraft
EADS
Electric Boat
Goodrich Aerospace
General Dynamics
Hamilton Sundstrand
Honeywell
ITT Communications
India Space Research
Korean Aerospace
Lockheed Martin
L-3 Communications
Northrop Grumman
Pratt & Whitney
Raytheon
Rockwell Collins
Rolls Royce
Saab Group
Thales
Turkish Aerospace
U.S. Air Force
XMobots Robotic

DO-178B | ED-12B

Automotive

Autoeversystems
Beko
Caterpillar
Chang'an Automotive
Delphi
Dongfeng Automotive
ENSO
Fujitsu
Hyundai
Magna Powertrain
Magneti Marelli
MOTONIC
Siemens
Tsinghua University
VALCO
Yazaki

ISO 26262

Medical

Abbott Laboratories
Acist Medical
Baxter Healthcare
Beckman Coulter
Boston Scientific
Biosafe
Cardinal Health
Covidien
Enteromedics
GE Healthcare
Hoana Medical
Hospira Medical
Impact
Medtronic
Philips Medical
Smiths Medical

FDA | IEC 62304

Railway

ABB
Allied Telesis
Alcatel
Ansaldo Signal
Alstom
Bombardier
CAP Gemini
Elin EBG
Electro-Motive Diesels
GE Transportation
SafeTran
Skoda Electric
Safetran Systems
Triconex
Union Switch & Signal

CENELEC | EN 50128

Industrial

Analog Devices
Arçelik
Bechtel Bettis
Berthold Technologies
British Energy
Caterpillar
Changzhou Scale Ltd.
Endress+Hauser
Flextronics
Foster Miller
General Electric
HAWE Hydraulik
ICS Tripex
Invensys
Ingersoll Rand
Itron
Krauss-Maffei
Mettler Toledo AG
MEI Group
NXP
NetApp
Oak Systems
Sanmina
ViaSat
ZIEHL

IEC 61508

Companies who…

- *have safety or mission critical applications*
- *have processes which specify rigorous testing*
- *have unit and integration testing requirements*
- *need the ability to automate regression testing*
- *have requirements to prove code coverage*
- *realize that NOT testing… **is NOT an option***

Our Solution

Full Embedded Software Testing

- *Unit and Integration testing*
- *Code coverage analysis through lifecycle*
- *Automated regression testing*
- *Testing on embedded targets and simulators*
- *Qualification materials for FAA, FDA, and others*

True Embedded Test Automation

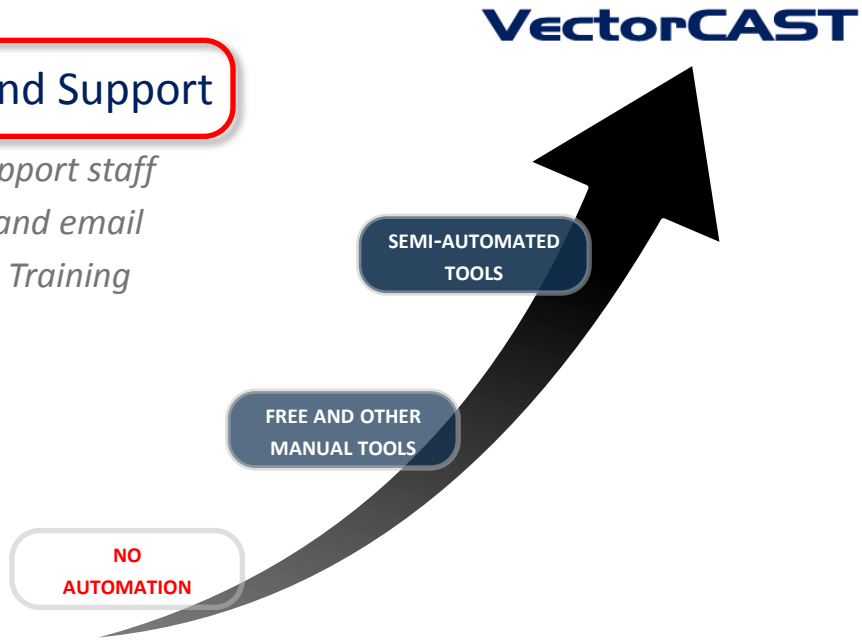
- *"Easiest to Use" test tool on the market*
- *Easy continuous integration and test for repeatability*
- *Integration with best of breed static analysis tools*

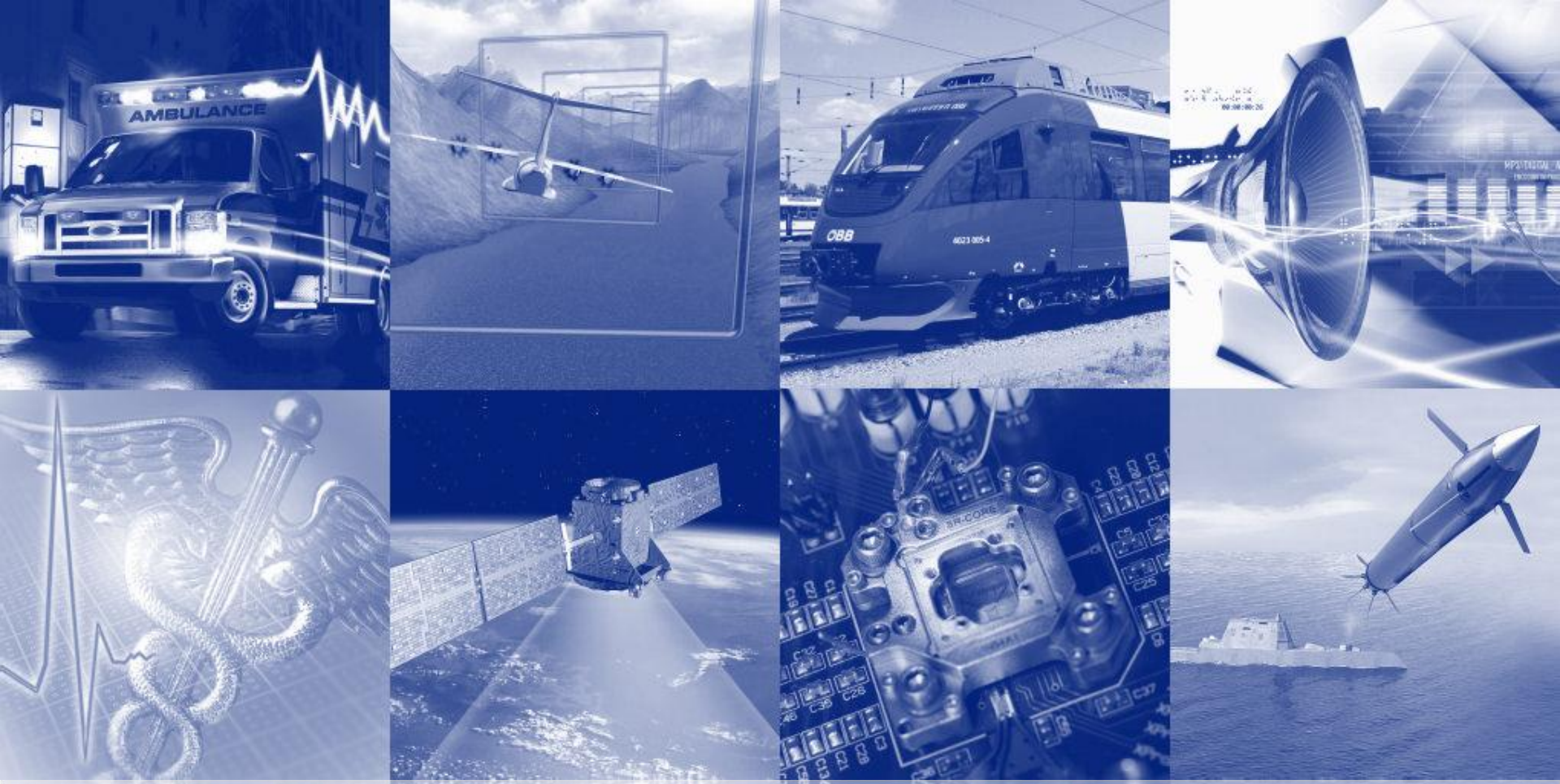
World-Class Service and Support

- *Our engineers are our support staff*
- *Local support via phone and email*
- *Consulting and Technical Training*

Benefits of using VectorCAST

- *Reduced Time-To-Market*
- *Easier Certification / Qualification*
- *Verifiable savings of time and money*

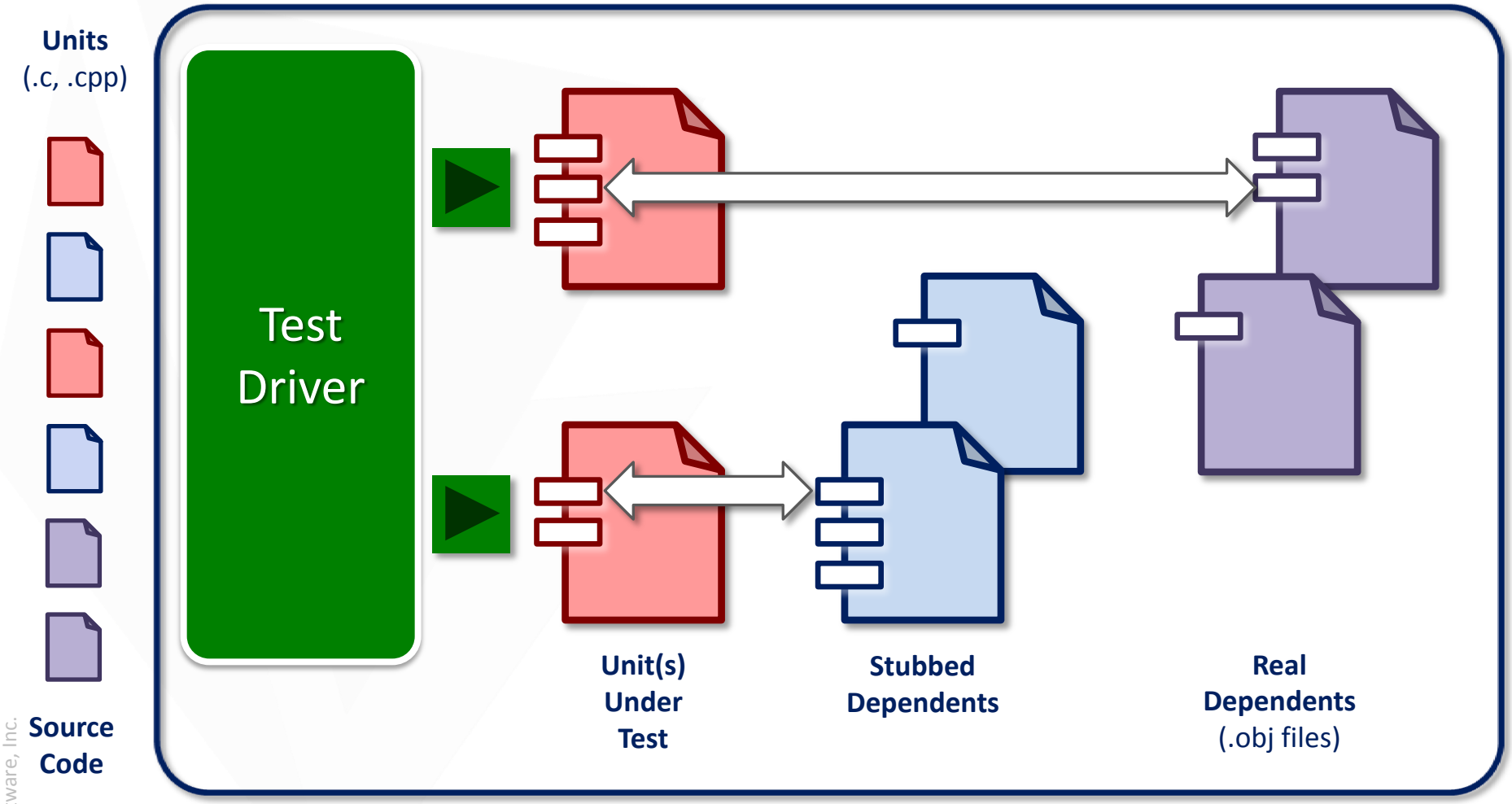




Using VectorCAST with Renesas HEW

VECTOR software

Unit and Integration Testing

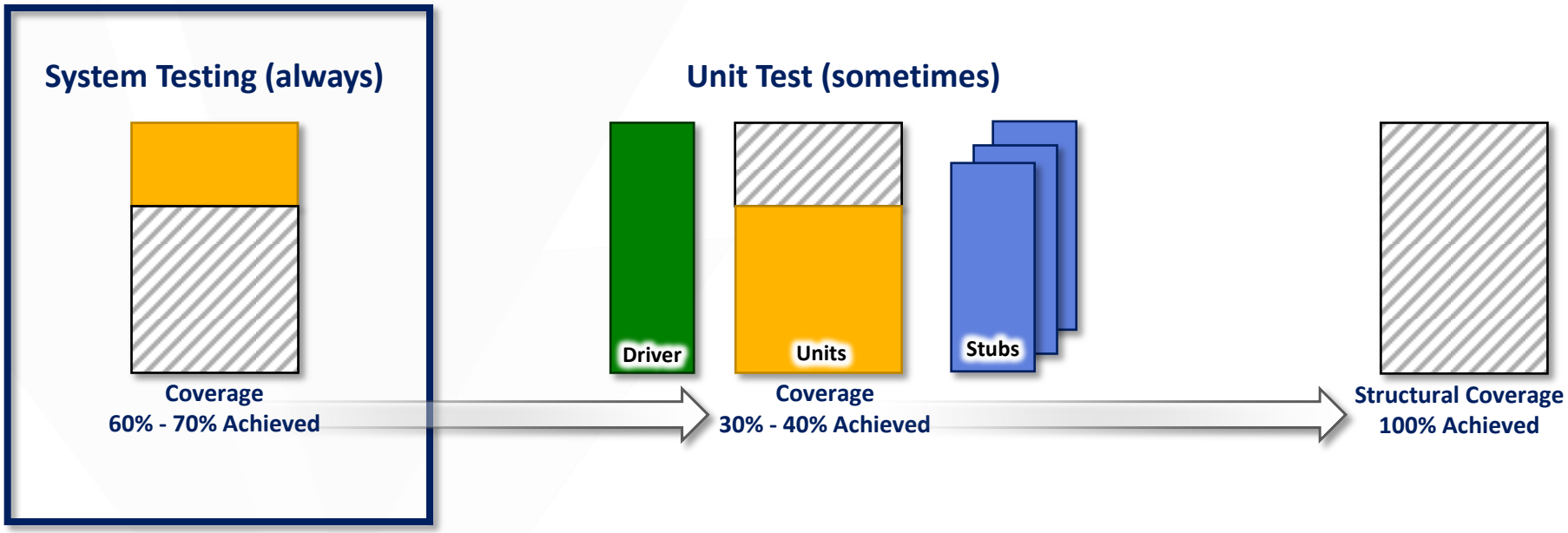


VectorCAST builds test harnesses automatically

VectorCAST/C++

- Unit Testing
  - Testing smallest piece of testable software
  - Unit is C/C++ source file
  - Requires generation of test stubs and drivers
  - Driver simulates calling unit and stub(s) simulates called unit
  - Enables generation of tests to ensure complete code coverage
- Integration Testing
  - Logical extension of unit testing
  - Allows testing of all units that make up a functional process
  - Requires multiple units under test at the same time
  - Identifies problems when units are combined

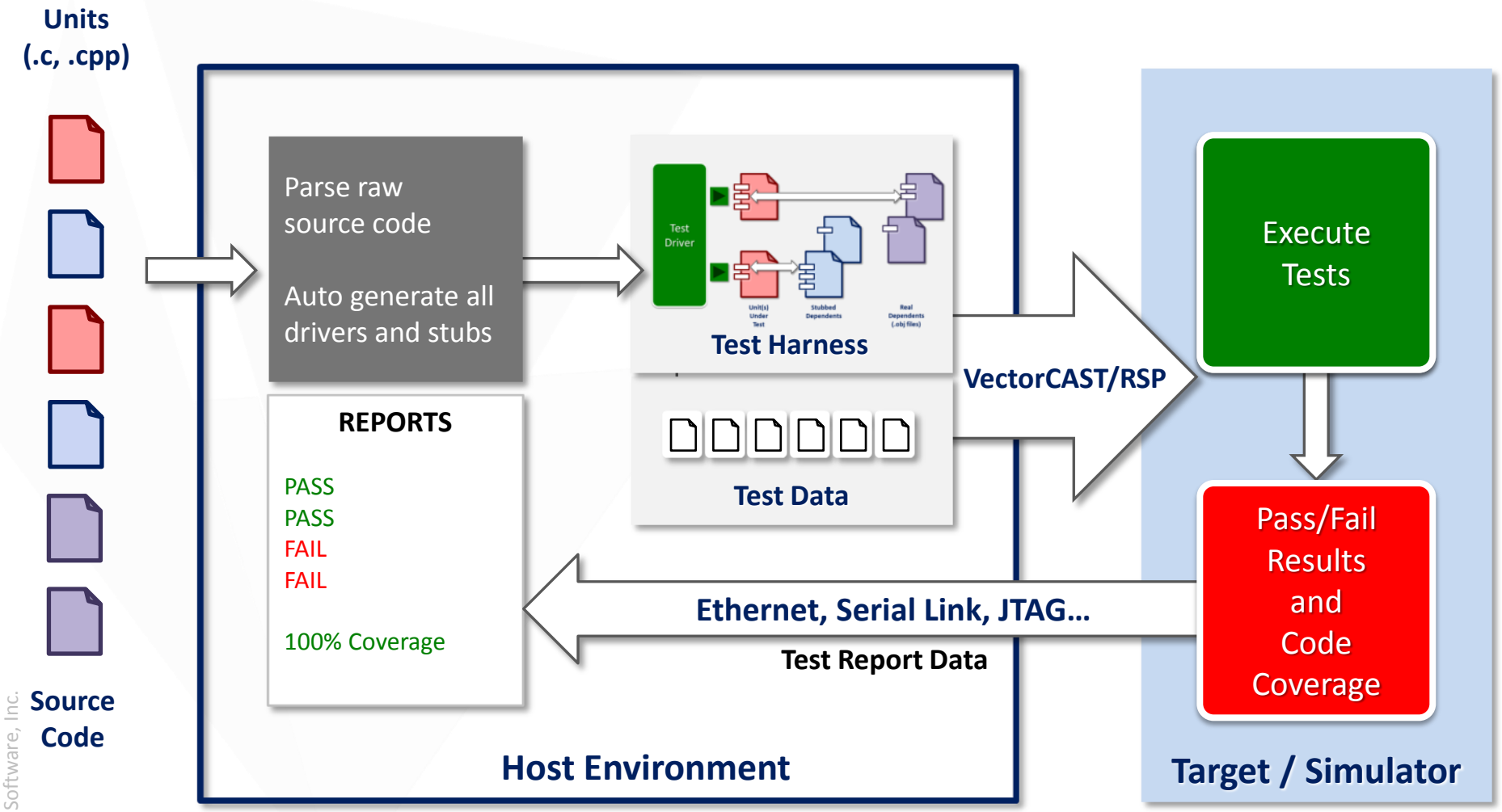
Achieving 100% Structural Coverage



VectorCAST/Cover

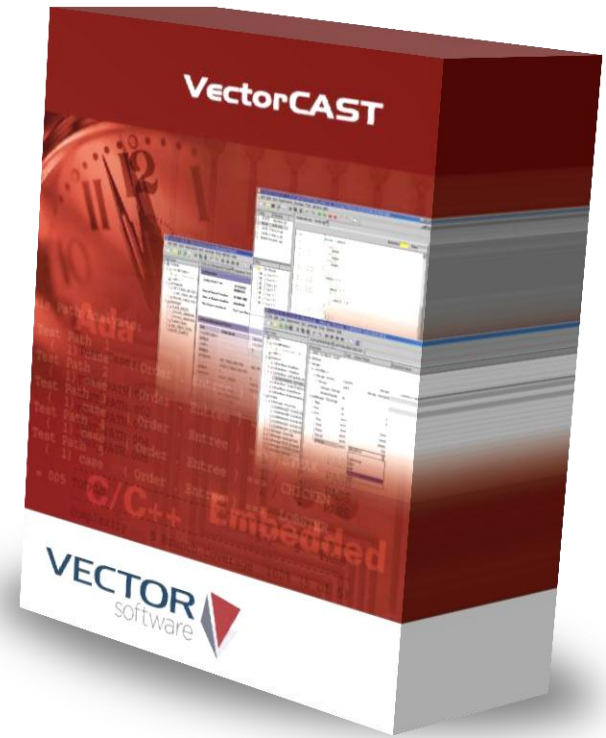
- Code Coverage
- System / Functional Test Verification
  - Running your existing tests to see what code is being executed
  - System tests can be executed different ways:
    - *from scripts, manually through GUI, simulator,…*
  - Generally performed by a dedicated test group
- Issues with System Test
  - Unlikely that full code coverage will be achieved
  - 60-70% coverage is common
  - Coverage should be combined with unit test coverage to achieve 100%

Embedded Target / Simulator Testing

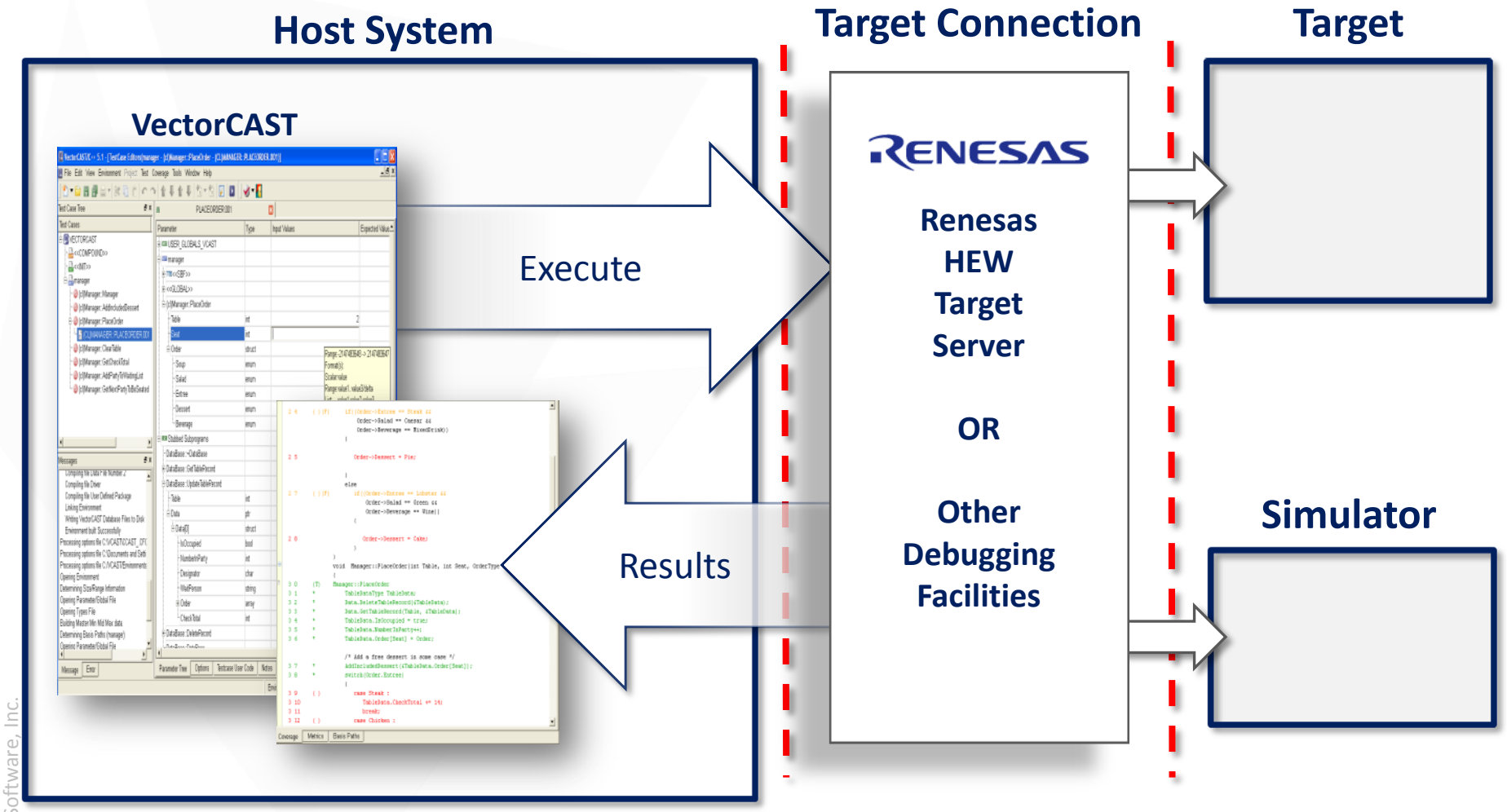


VectorCAST/RSP

- Runtime Support Package
- Target or Simulator Testing
  - Used with VectorCAST/C++
  - Cross-Compiler specific test-harness generation for execution on a target or simulator
  - Automatic download of test harnesses and test cases for execution on a target
  - Execution controlled from GUI provides an ease-of-use not common to component target testing
  - Customized I/O facilities on both the host and target sides allow communication between the host and target processors for test reporting



VectorCAST/RSP for Renesas HEW Target



Why is Regression Testing Important?

- Code can be re-factored over time
  - The biggest reason that engineers don't re-factor code is the lack of adequate test cases
  - Re-factoring code is risky – I might break something!
  - Robust test infrastructure means code can be re-factored with confidence
- Leverages testing investment across the life cycle
  - Testing once makes no sense
  - Testing once per release cycle makes no sense
  - Testing Constantly makes sense
- Bugs are easiest to fix on the day that they are introduced.
  - Finite number of patches to be reviewed
  - Conversely months later it may take days to find the source of the bug
- Test cases are kept in sync with code
  - Minimal maintenance required if tests are constantly maintained
  - Huge effort if tests are updated only at release time

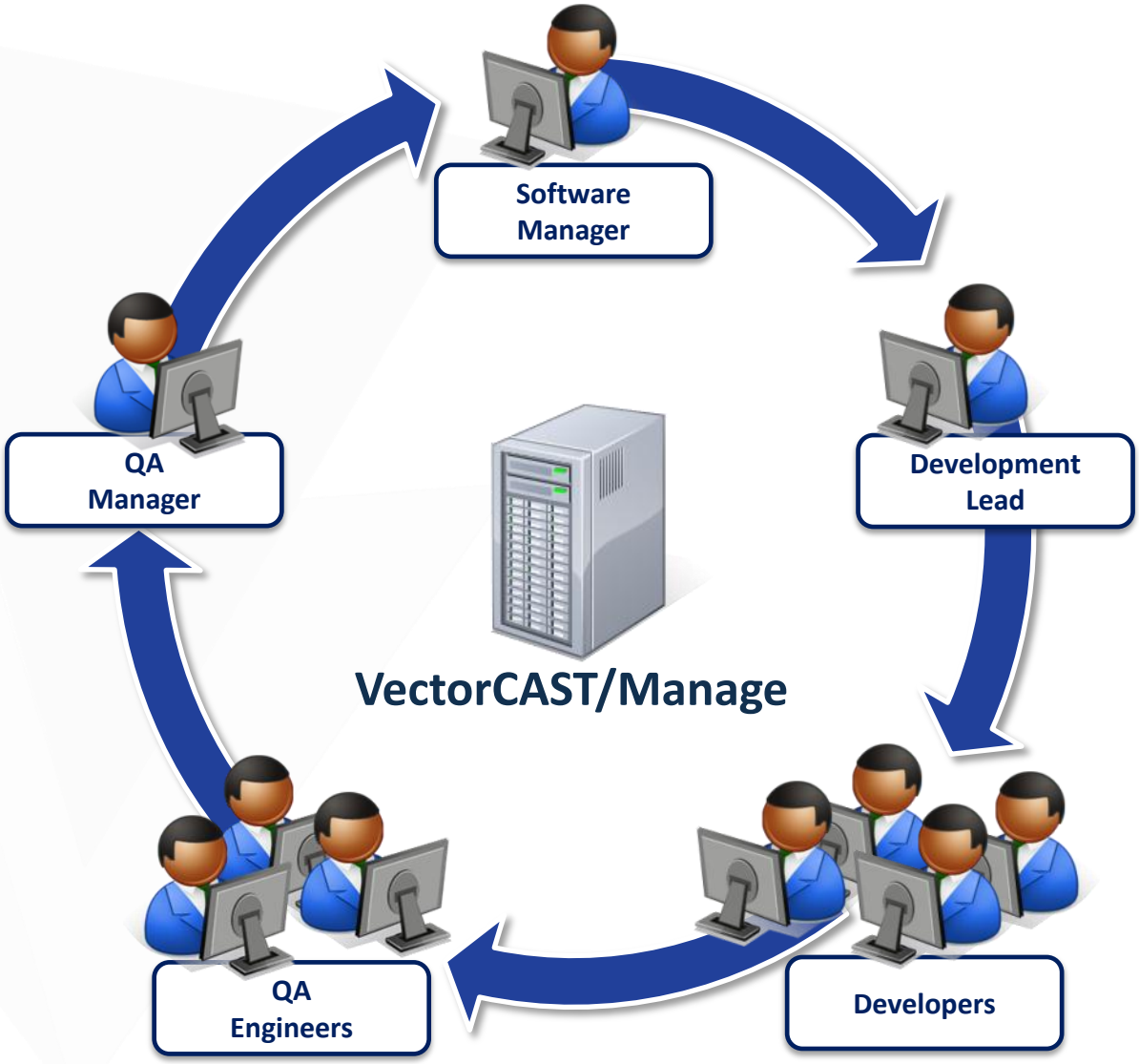
Regression Testing must be…

- Quick
  - Hardware is relatively cheap
  - 10K of hardware is 1 engineer for 1 month
  - Time is of essence
- Robust
  - Run frequently; constant testing
- Support Parallel Testing
  - Want each branch to be tested completely, under 1 hour

VectorCAST/Manage

- Regression Testing
- Testing multiple source code configurations
  - Application deployed on different physical targets
  - Application built with multiple tool chains
  - Application built with multiple compile options
- Testing multiple source code versions
  - Production, Beta, or R&D code
- Testing across multiple machines or architectures
  - Test suite may take weeks to run on one machine
  - Application may be deployed on multiple operating systems (Windows, Linux)
- Testing what has changed
  - Interim testing of only those source files that have changed
  - Based on file checksums or change dates

VectorCAST/Manage User Community

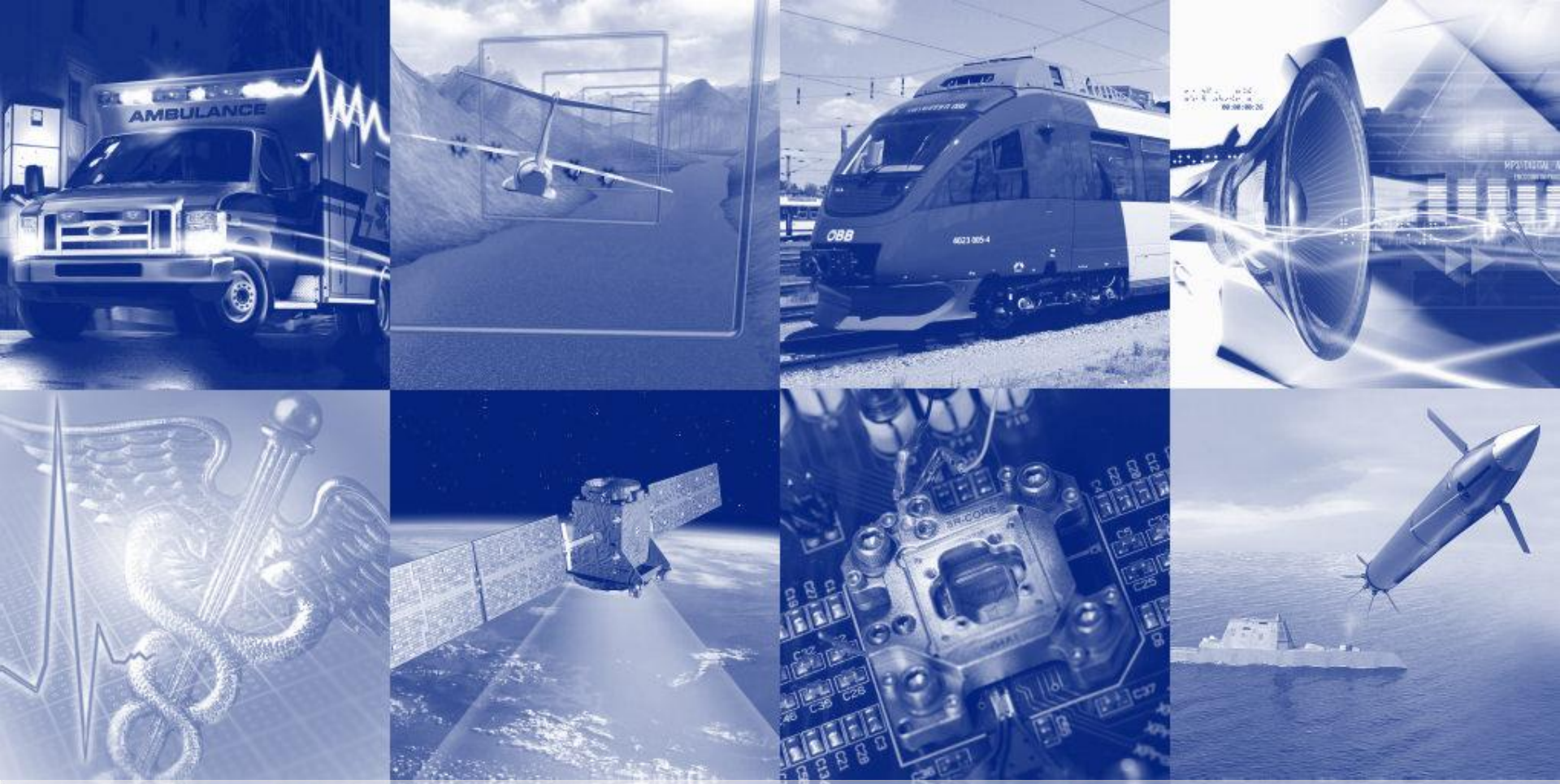


How Regression Testing Results are Analyzed

- QA Manager
  - Reviews HTML Over-Night Reports
  - Assigns Each Test Failure to a QA Engineer to investigate

- QA Engineers
  - Run Test Manually to Confirm Test Failure
  - Test Failure can be a Regression or an Intended Change in Functionality
  - Analyze Commit Log to determine cause of failure
  - Create an Internal Bug Report and Assign responsible developer

- Developers
  - Dispatch Debug Build to Build Servers
  - Run Failing Test on Debug Build in Development Environment
  - Confirm Breakage
  - Determine Corrective Action



> Lab Introduction and VectorCAST Demonstration

VECTOR software

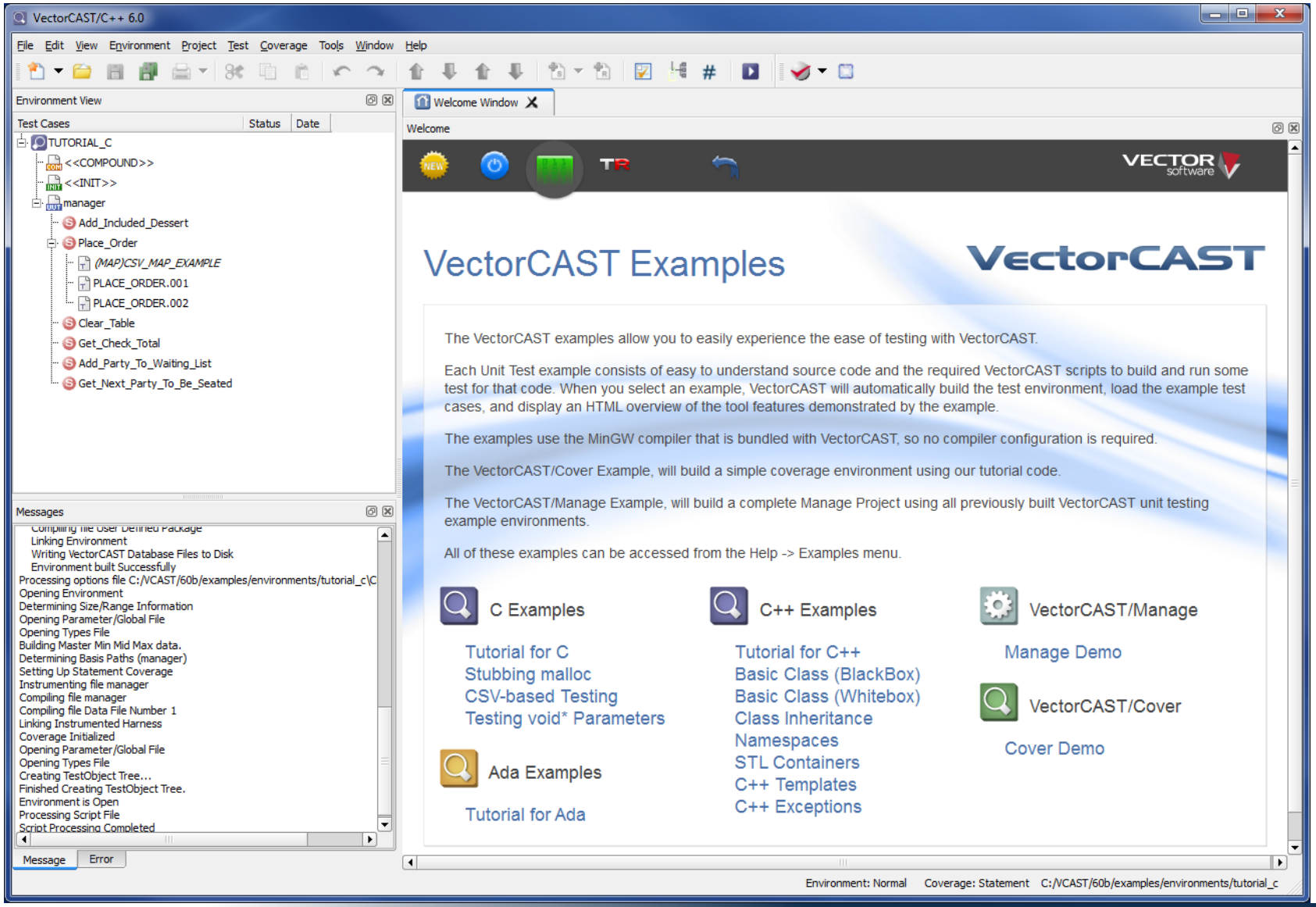
Overview

- Intro to Tutorial Code
- Setting up VectorCAST for use with HEW
- Runtime Support Package (RSP) details
- Configuration and basic tool usage (demo)

Introduction to Tutorial Code

- In this lab, we'll be using the VectorCAST tutorial code
- Simulation of a restaurant point-of-sale application
  - Code handles various elements of managing a table
  - Place order, get check total, clear table, specials, etc.
- Located in VectorCAST install dir (C:\VCAST\Tutorial)
- We'll mainly focus on manager.c and database.c

Setting up VectorCAST for use with HEW



Setting up VectorCAST for use with HEW

One time:

- Prep HEW and template workspace for execution

For each environment directory:

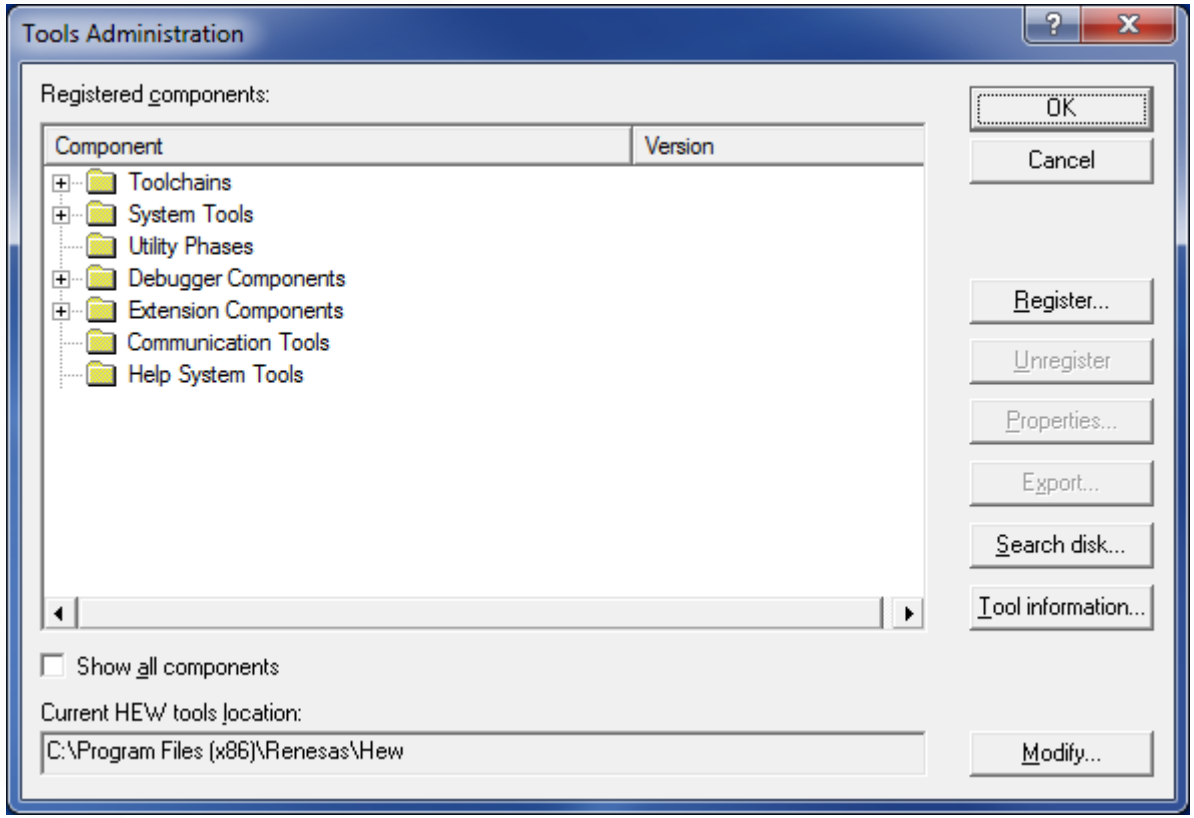
- Launch VectorCAST
- Select working directory
- Create new unit test environment
- Select RX600 compiler template
  - Designed to work with a range of RX600 processors, including RX62N and RX63N

For more environments:

- Just create new unit test environment
  - Compiler template is automatically filled in

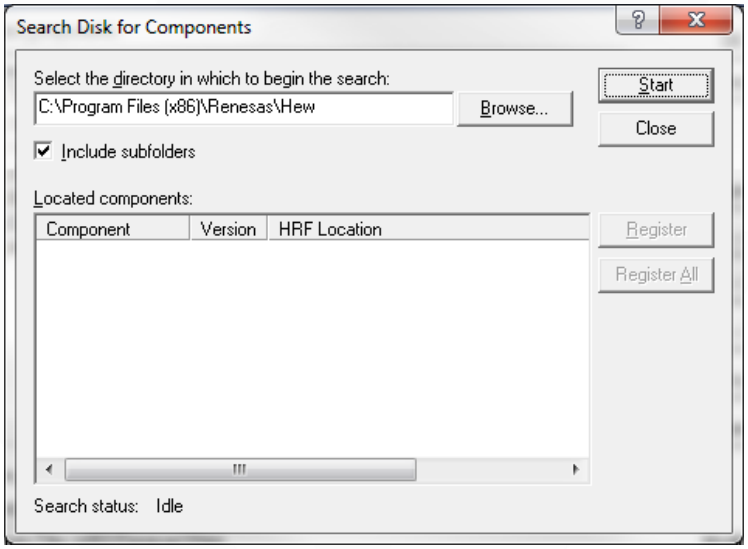
Preparing HEW and Template Workspaces for Execution

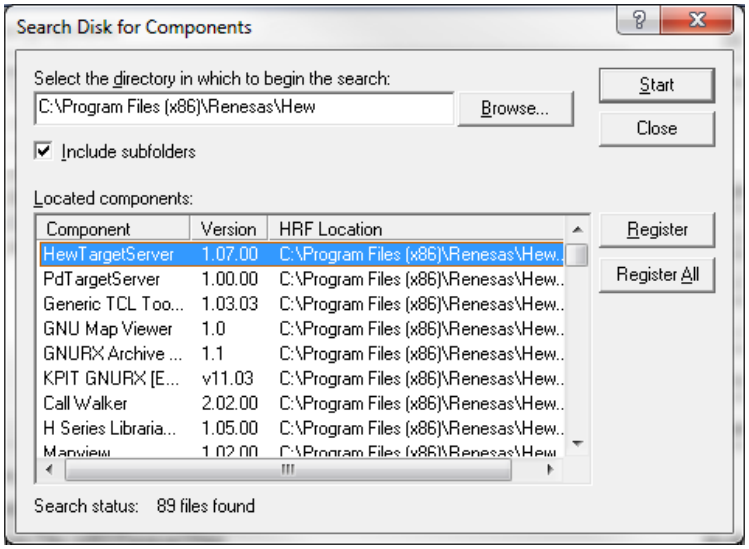
- Enable HEW Target Server (COM scripting support)
- Tools => Administration



Preparing HEW and Template Workspaces for Execution

- Enable HEW Target Server (COM scripting support)
- Click “Search disk” to find components to register

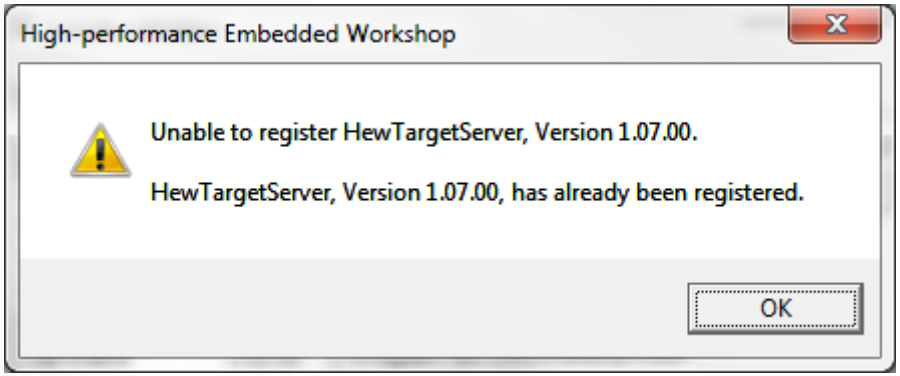




- Select HewTargetServer and click “Register”

Preparing HEW and Template Workspaces for Execution

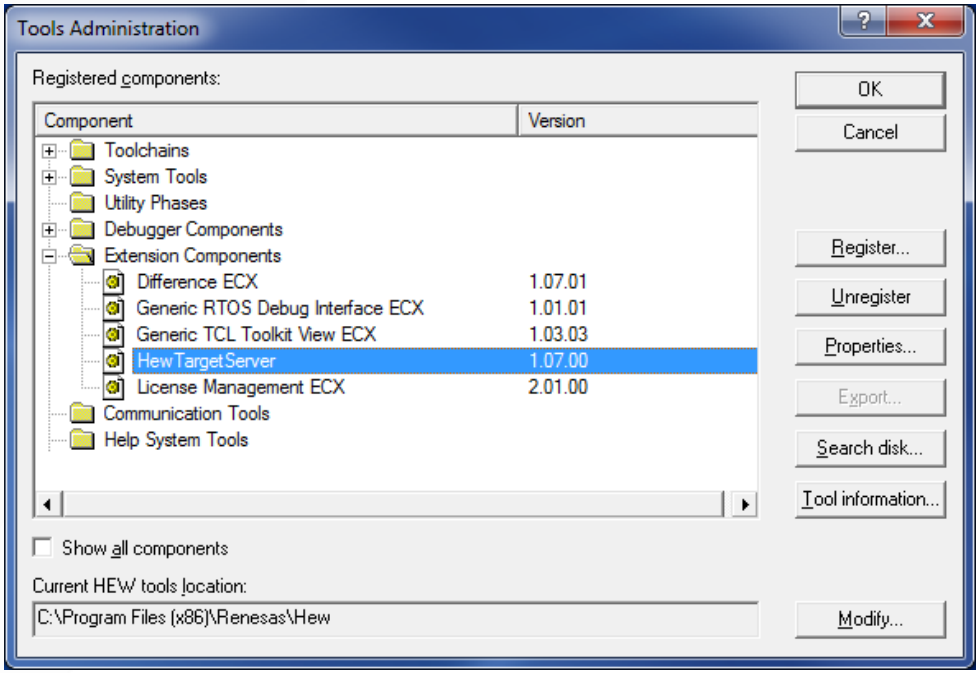
- Enable HEW Target Server (COM scripting support)
- If component already registered, you'll see this:



- That's fine. Just click OK.

Preparing HEW and Template Workspaces for Execution

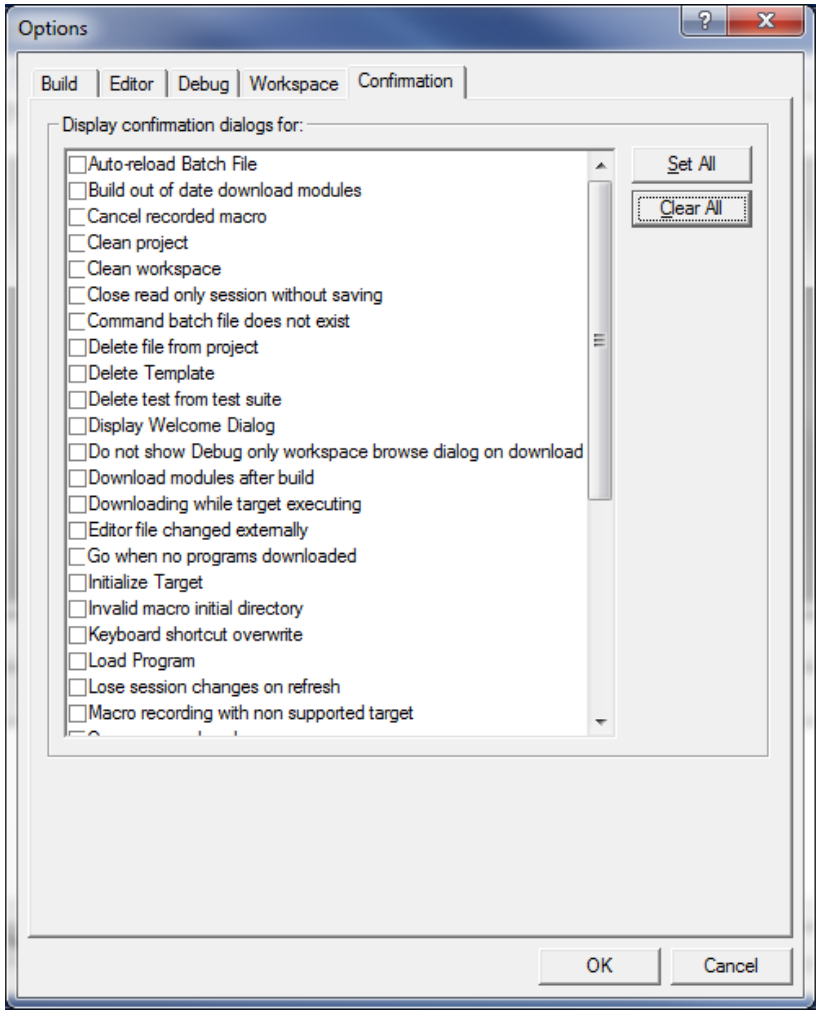
- Enable HEW Target Server (COM scripting support)
- Shut down HEW and re-launch, then confirm that HewTargetServer is listed in the Tools Administration dialog
  - Should be under Extension Components



  - Occasionally, the registration will not stick the first try, so it's good to verify that the registration takes hold before moving on.

Preparing HEW and Template Workspaces for Execution

- Disable HEW confirmation dialogs to avoid breaking script automation
- Setup => Options => Confirmation tab
- Click “Clear All”
- Click OK



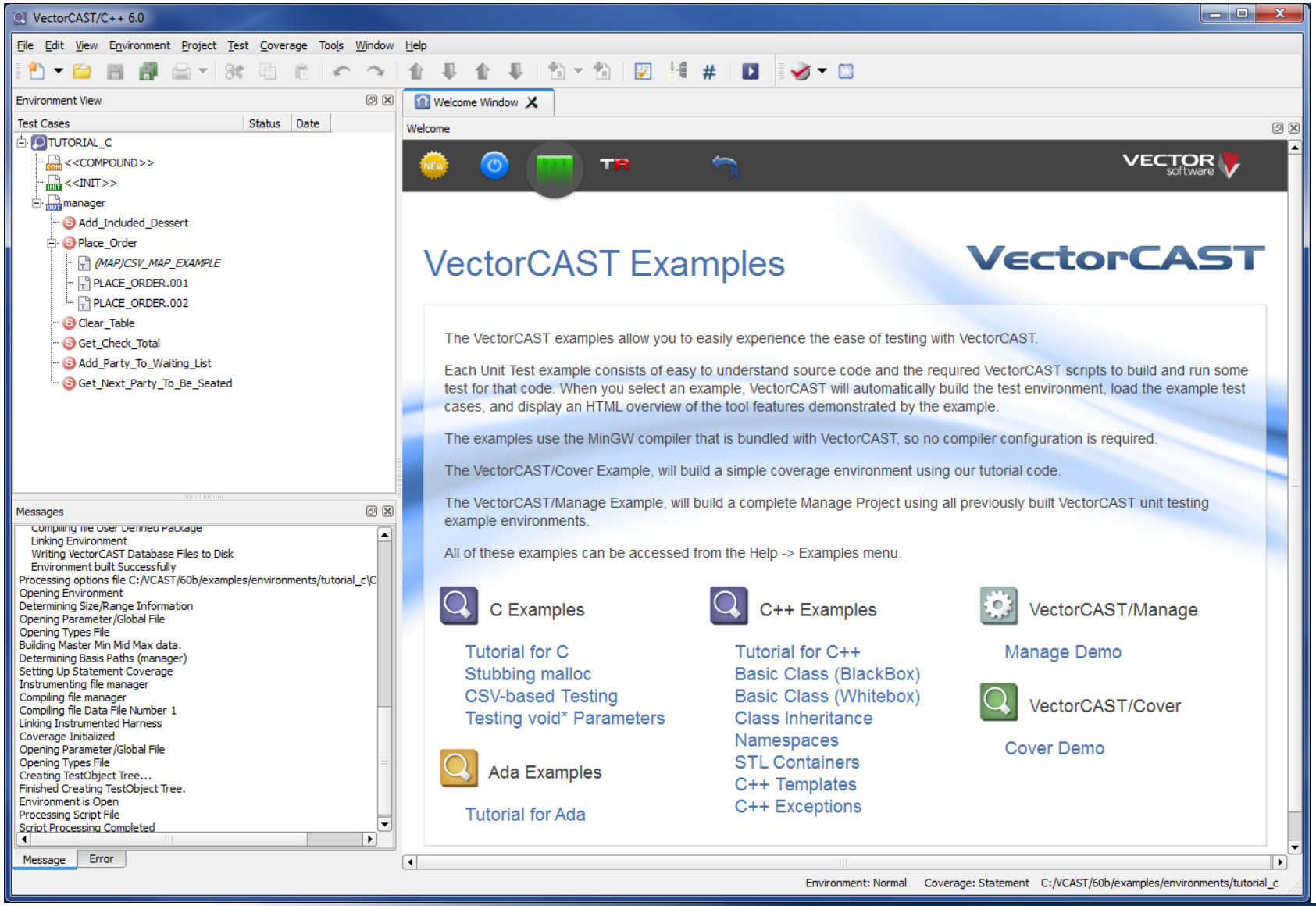
Preparing HEW and Template Workspaces for Execution

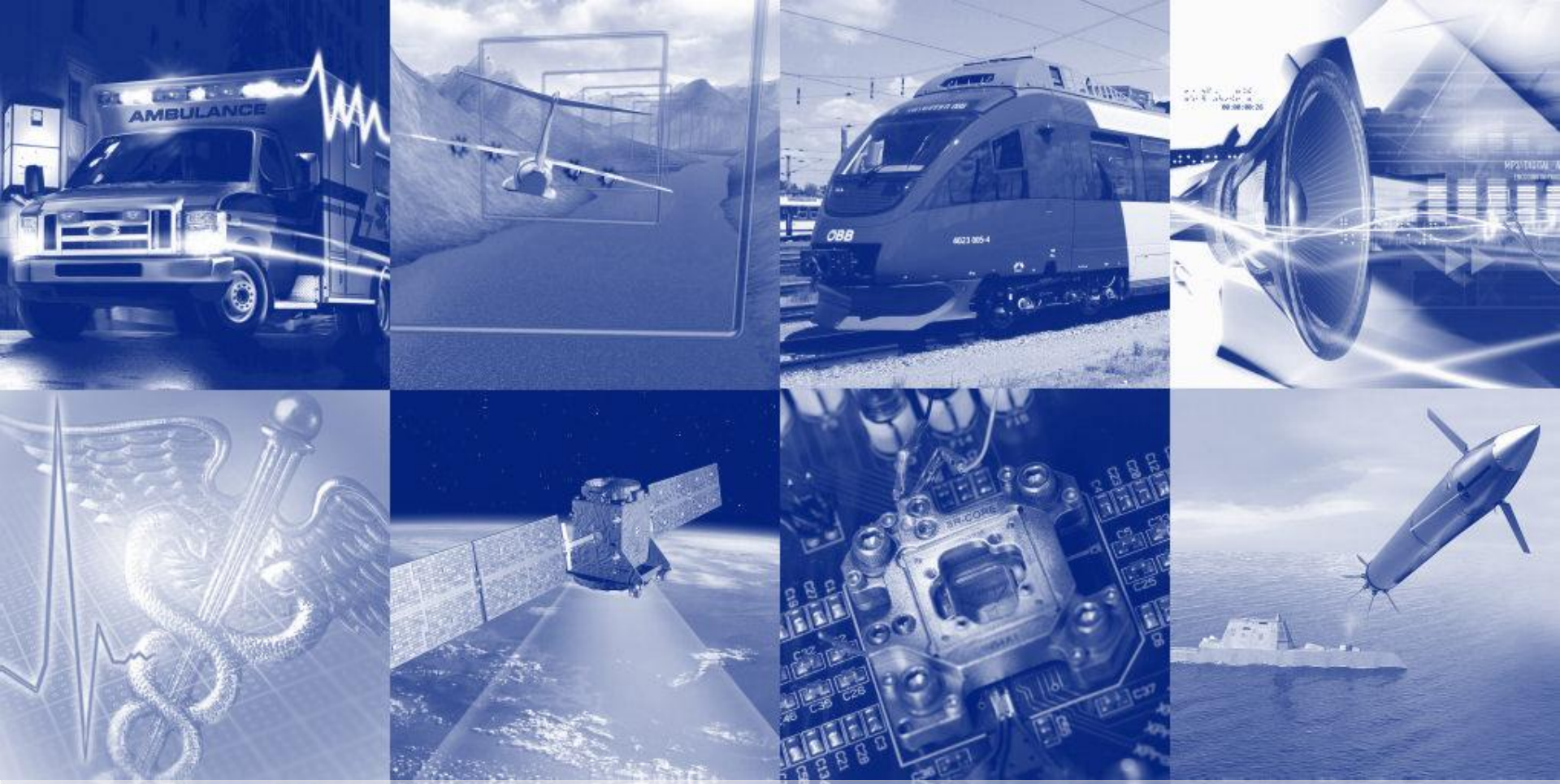
- Open VectorCAST template workspaces to update internal paths
- Templates are located under VectorCAST installation directory (default: C:\VCAST)
  - Simulator: C:\VCAST\DATA\renesas\rx\RX600\hew_sim\hew_sim.hws
  - J-Link target: C:\VCAST\DATA\renesas\rx\RX600\hew_target\hew_target.hws
- Open template workspace in HEW
  - Template will automatically to connect to simulator or target
  - On target connection, if extra connection dialogs are shown, click option at bottom to hide the dialog in the future.

RX63N HEW Runtime Support Package (RSP) details

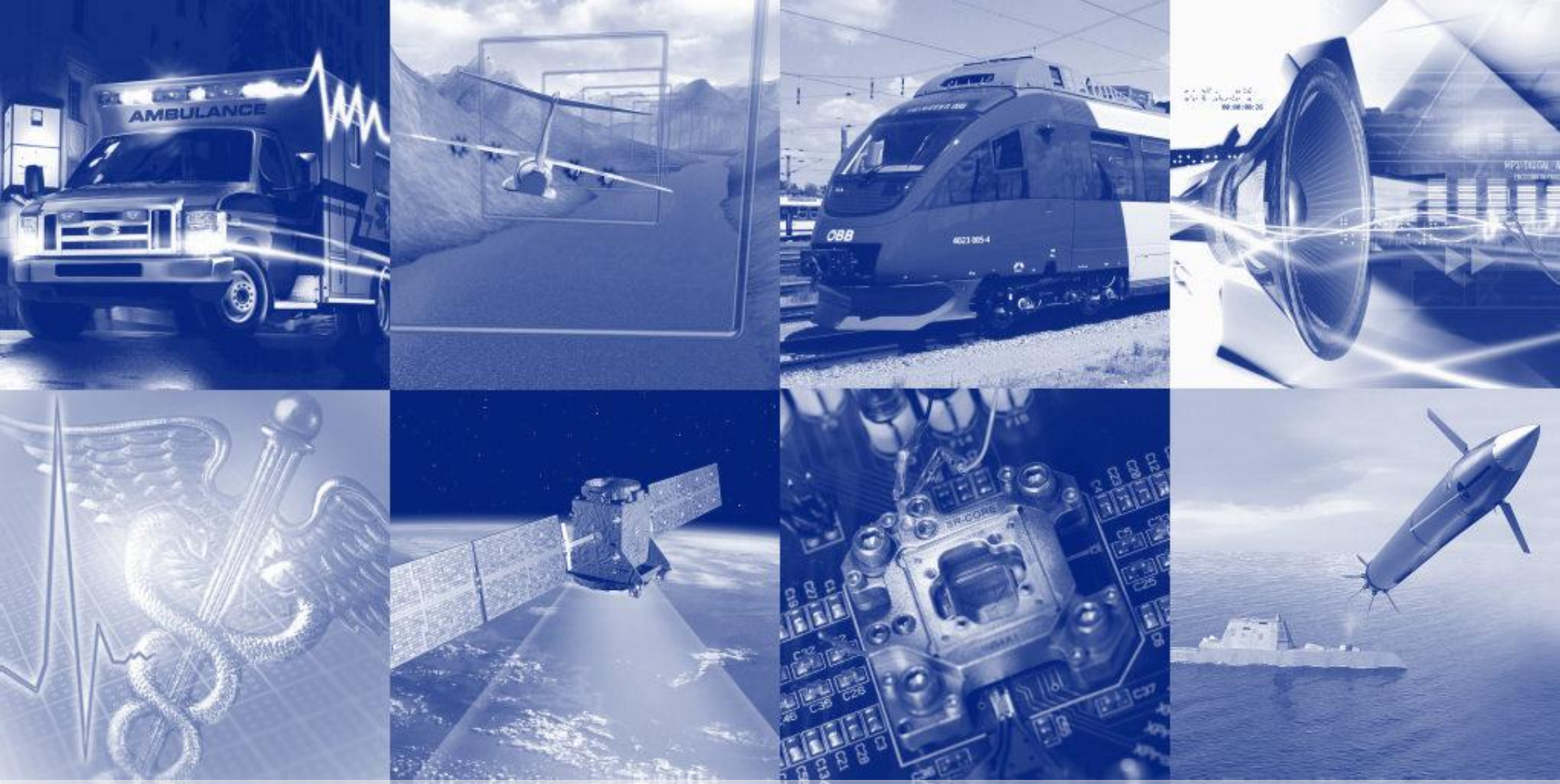
- Interface program is implemented using HEW Target Server (COM) interface
- Renesas_Interface.exe:
  - Launches or connects to HEW instance
  - Opens VectorCAST template workspace
  - Connects to target (simulator or board)
    - *Templates are provided for each connection type*
  - Downloads test harness
  - Sets up I/O channel and logging
  - Runs test harness
  - Cleans up target and returns control to VectorCAST
- Prerequisites:
  - HEWTargetServer plugin must be enabled
  - Some HEW configuration settings must be tweaked to ensure correct operation
  - Template workspace must be opened once in HEW to reconfigure local paths
  - This setup only needs to be done once per machine

Configuration and Basic Tool Usage (Demo)



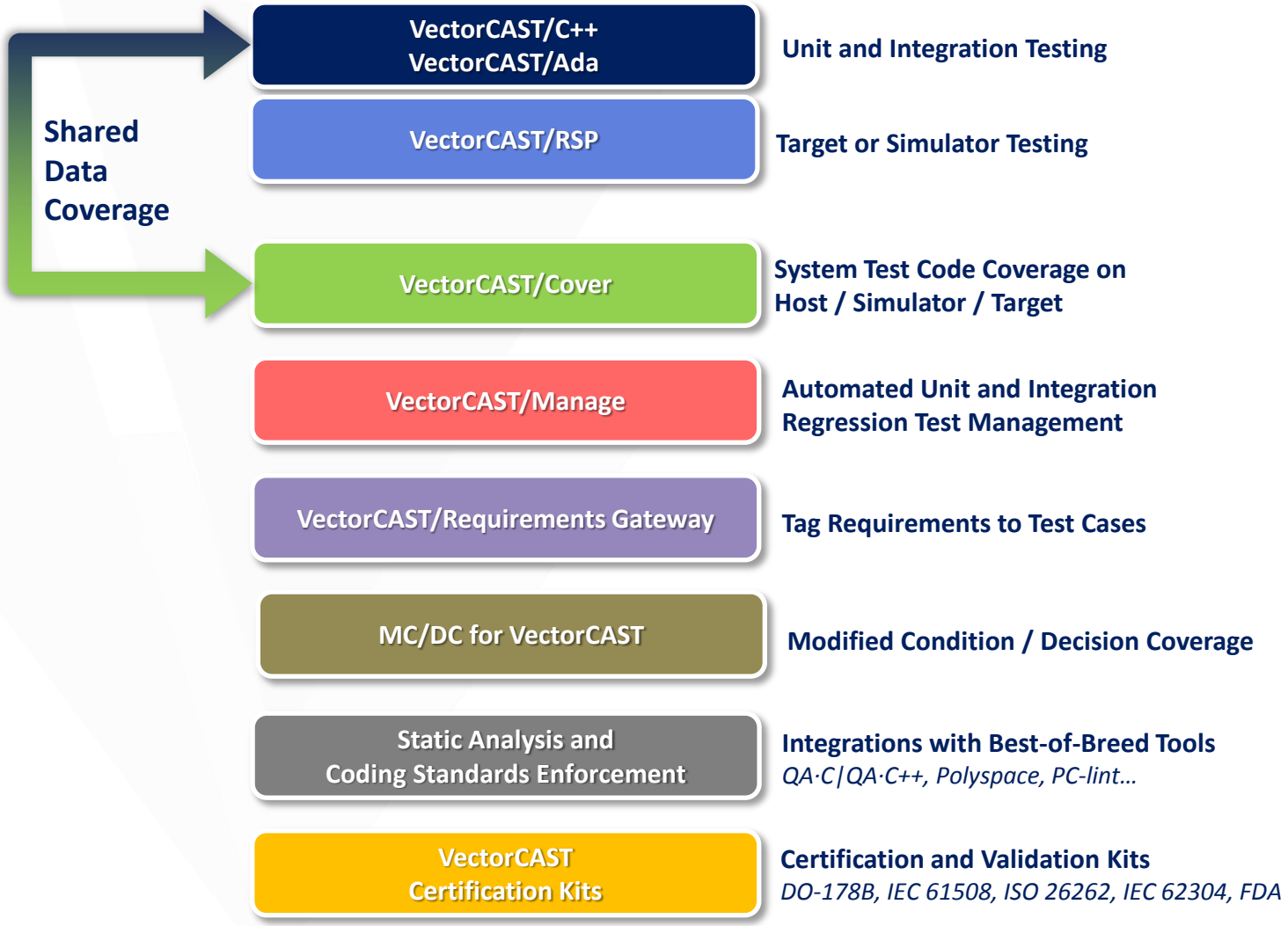


> Interactive Lab – See Handout for Instructions



Wrap-up Discussion

VectorCAST Product Suite Overview



Vector Software Advantage

1. ***Experienced Team***
2. ***Proven Track Record***
3. ***Solution Centric Approach***
4. ***Committed To Your Success***

Connect With Us

Vector Software Website
vectorcast.com

Automated Software Testing Blog
vectorcast.com/blog

User Driven Community
testresponsibly.com

VectorCAST Professional Network
linkedin.com

VectorCAST Fans
facebook.com/vectorsoftware

News and Events
twitter.com/vectorsoftware

Tutorials and Demonstrations
youtube.com/vectorcast

VECTOR software

T: +1.401.398.7185
F: +1.401.398.7186
E: info@vectorcast.com