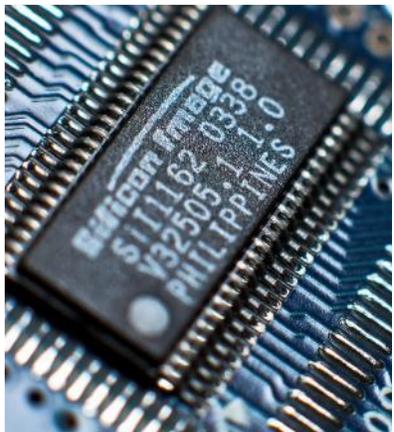




Software Test Automation for High Quality Software



Agenda



VectorCAST

Solutions for Safety and Business Critical Embedded Applications



Aerospace



Automotive



Medical



Industrial



Railway



Business-Critical



About Vector Software



Organizations world-wide use our test automation platform to reduce the development time, risk, and cost of delivering reliable, secure, and compliant software.

Our Company

- Founded in 1990
 - First product released for Lockheed Martin's C-130J Super Hercules
- Corporate headquarters in Rhode Island, USA
 - US HQ – Providence, Europe HQ – London, Asia HQ – Shanghai
 - Worldwide Presales, Sales, and Technical Support – Direct and Indirect
 - Development in US, UK, and India
 - 90+ Employees
 - 100+ Partners
 - 900+ Customers
- Global Services
 - On-site training, best practice workshops, project delivery



Vector Software Global Services

- Mentoring Clients to Self Sufficiency
- Expertise
 - Deep domain knowledge of software test
 - World-wide technical staff
- Typical Engagements
 - **On-Site Training:** classroom lecture and hands-on exercises
 - **Project Consulting:** using VectorCAST with actual project code
 - **Best Practices Workshops:** maximizing your return on investment
 - **Continuous Test:** automated testing infrastructure implementation
 - **Baselining:** creating a repeatable testing infrastructure for legacy applications based on how the code operates today



Our Markets



Aerospace



Automotive



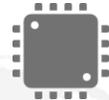
Medical



Industrial



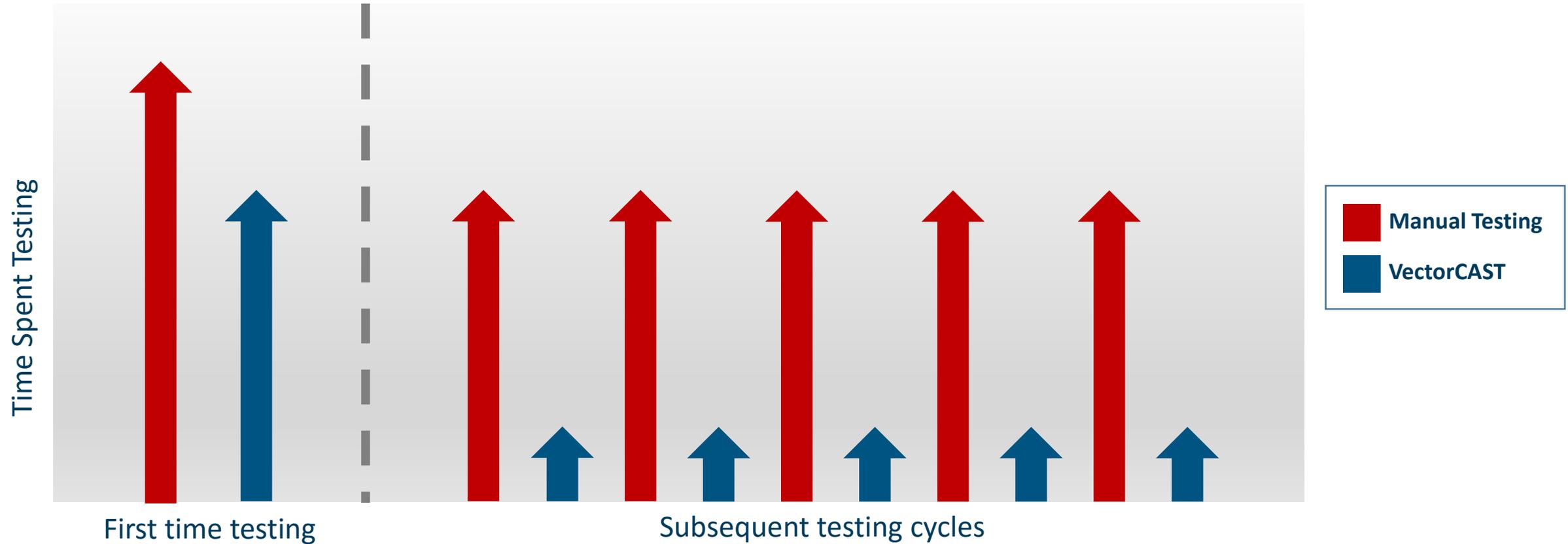
Railway



Business-Critical



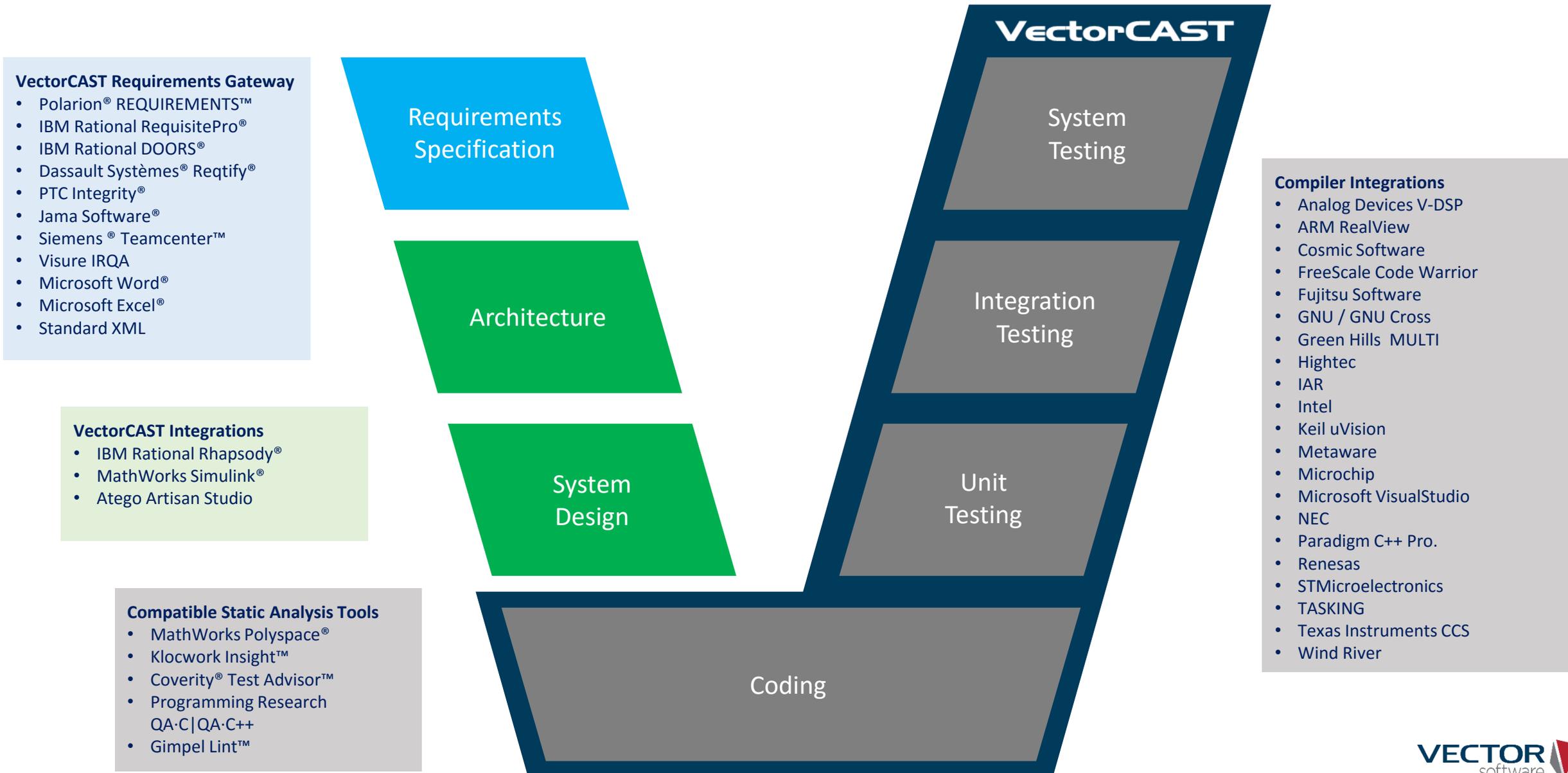
Benefits of VectorCAST Test Automation



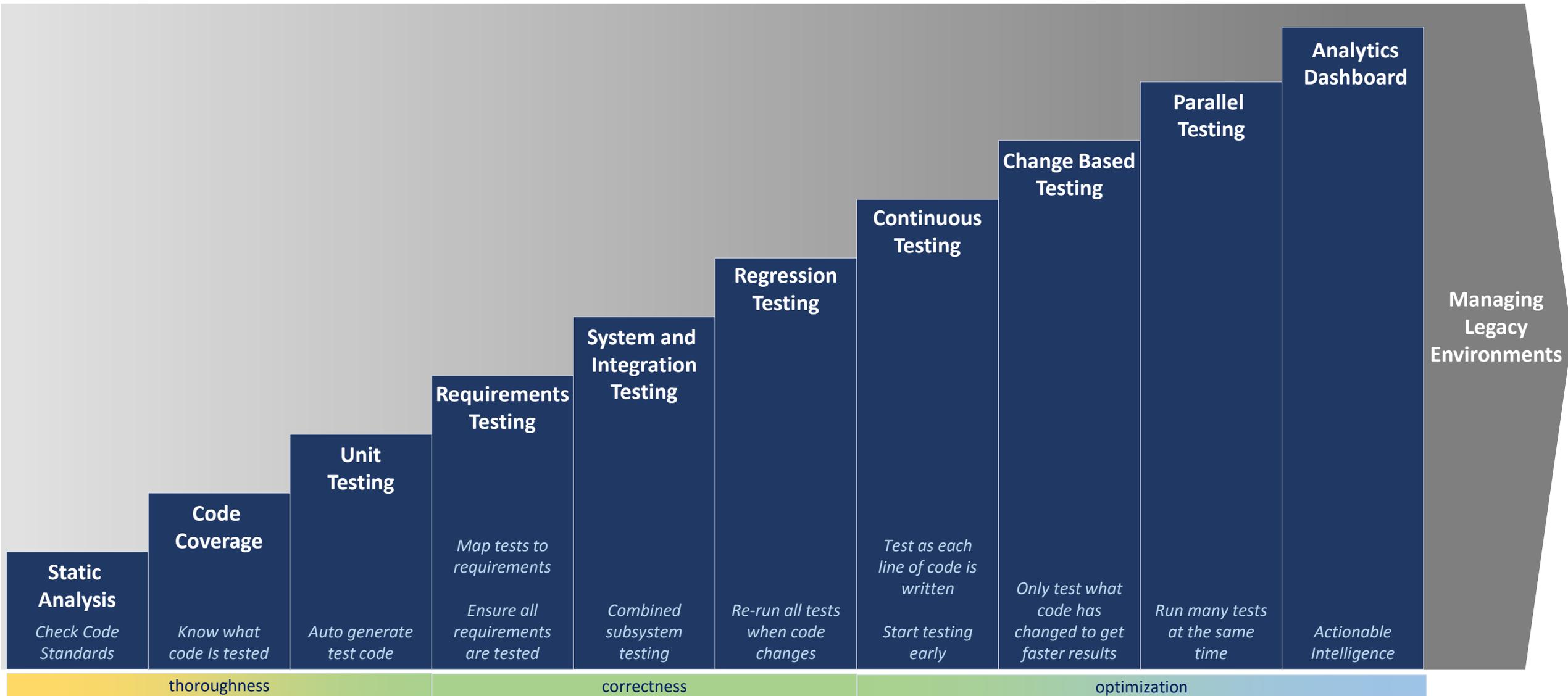


The VectorCAST Solution

Where does VectorCAST Fit in Your Environment?

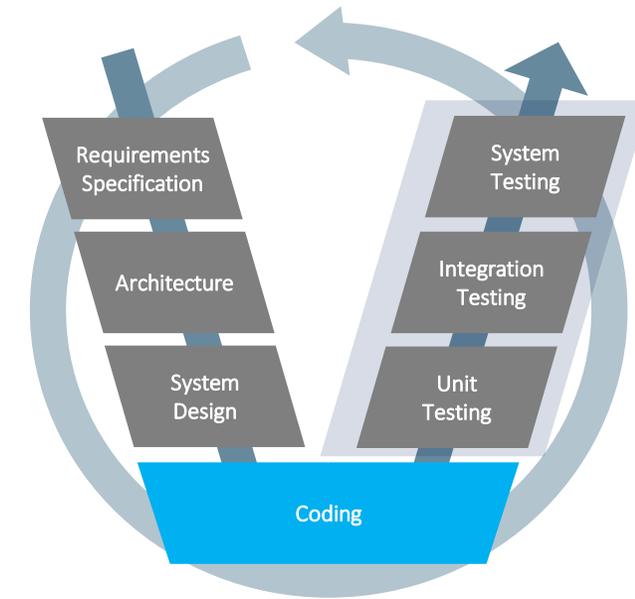


Where are you with your Test Process?



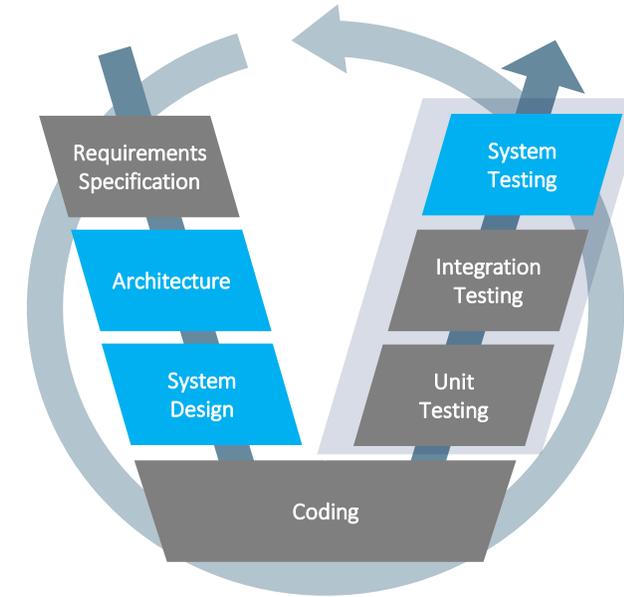
Static Analysis

- A range of tools for C, C++, and Ada to complement VectorCAST dynamic testing
- Enforce Coding Standards
 - MISRA C, MISRA C++, in-house
- Find bugs
 - Divide by zero, array out of bounds
- Formal methods to prove software safety
- Certified for use in safety-critical systems

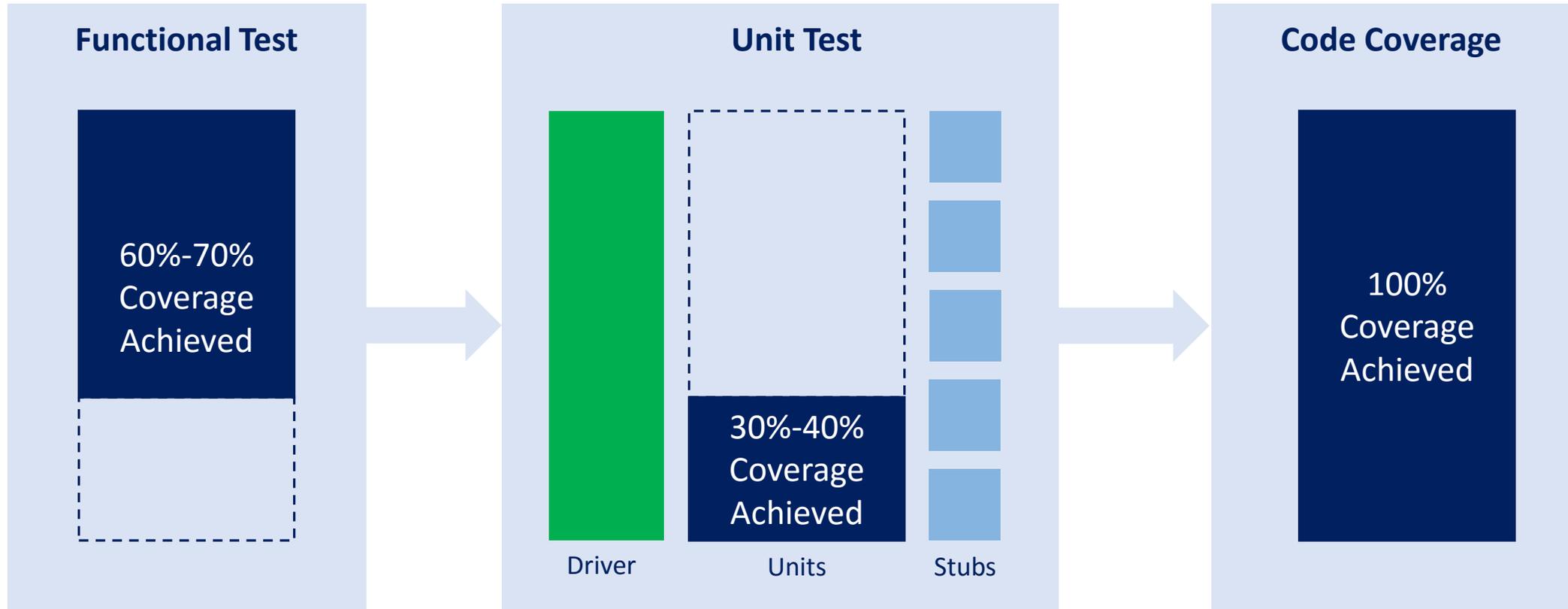


Code Coverage Across Lifecycle

- Provides insight into the completeness of testing
- Reports show whether released applications are fully tested
- Forces developers to review code as tests are developed
- Purpose-built with small footprint for embedded target environments
- Coverage By Analysis
 - Augment measured coverage with manual analysis to achieve the mandated 100% code coverage.
 - Intuitive editor allows users to provide analysis for statements, branch outcomes, or MC/DC pairs



Code Coverage



Combining System and Unit Test for 100% Code Coverage

Step 1:
Conduct system test and measure code coverage

```
manager.cpp
Statements 64% Branches 42% Pairs 0%
1 0 (T) Manager::PlaceOrder
1 1 * Data.DeleteTableRecord(&TableData);
1 2 * Data.GetTableRecord(Table, &TableData);
1 3 * TableData.IsOccupied = true;
1 4 * TableData.NumberInParty++;
1 5 * TableData.Order[Seat] = Order;
1 6 * switch (Order.Entree) {
1 7 (T) case Steak :
1 8 * TableData.CheckTotal += 14;
1 9 * break;
1 10 (T) case Chicken :
1 11 * TableData.CheckTotal += 10;
1 13 ( ) case Lobster :
1 14 * TableData.CheckTotal += 18;
1 15 * break;
1 16 ( ) case Pasta :
1 17 * TableData.CheckTotal += 12;
1 18 * break;
1 19 ( ) default :
1 20 * break;
1 22 * Data.UpdateTableRecord(Table, &TableData);
2 0 (T) void Manager::ClearTable(int Table)
2 1 * Data.DeleteRecord(Table);
3 0 (T) int Manager::GetCheckTotal(int Table)
3 1 * TableDataType TableData;
3 1 * Manager::GetCheckTotal
3 1 * Data.DeleteTableRecord(&TableData);
```

VectorCAST/Cover

Step 2:
Augment coverage with more unit tests

Table	short
Seat	short
Order	struct
Soup	enum
Salad	enum
Entree	enum

```
2 6 * switch(Order.Entree)
2 7 (T) {
2 8 * case NO_ENTREE :
2 9 * break;
2 10 (T) case STEAK :
2 11 * TableData.CheckTotal = TableData.CheckTotal + 14.0;
2 12 * break;
2 13 (T) case CHICKEN :
2 14 * TableData.CheckTotal = TableData.CheckTotal + 10.0;
2 15 * break;
2 16 (T) case LOBSTER :
2 17 * TableData.CheckTotal = TableData.CheckTotal + 18.0;
2 18 * break;
2 19 ( ) case PASTA :
2 20 * TableData.CheckTotal = TableData.CheckTotal + 12.0;
2 21 * break;
```

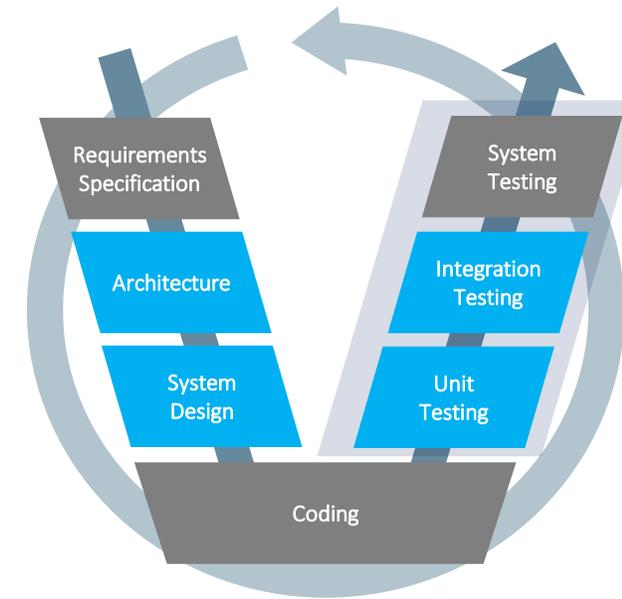
VectorCAST/C++

Step 3:
100% Code Coverage

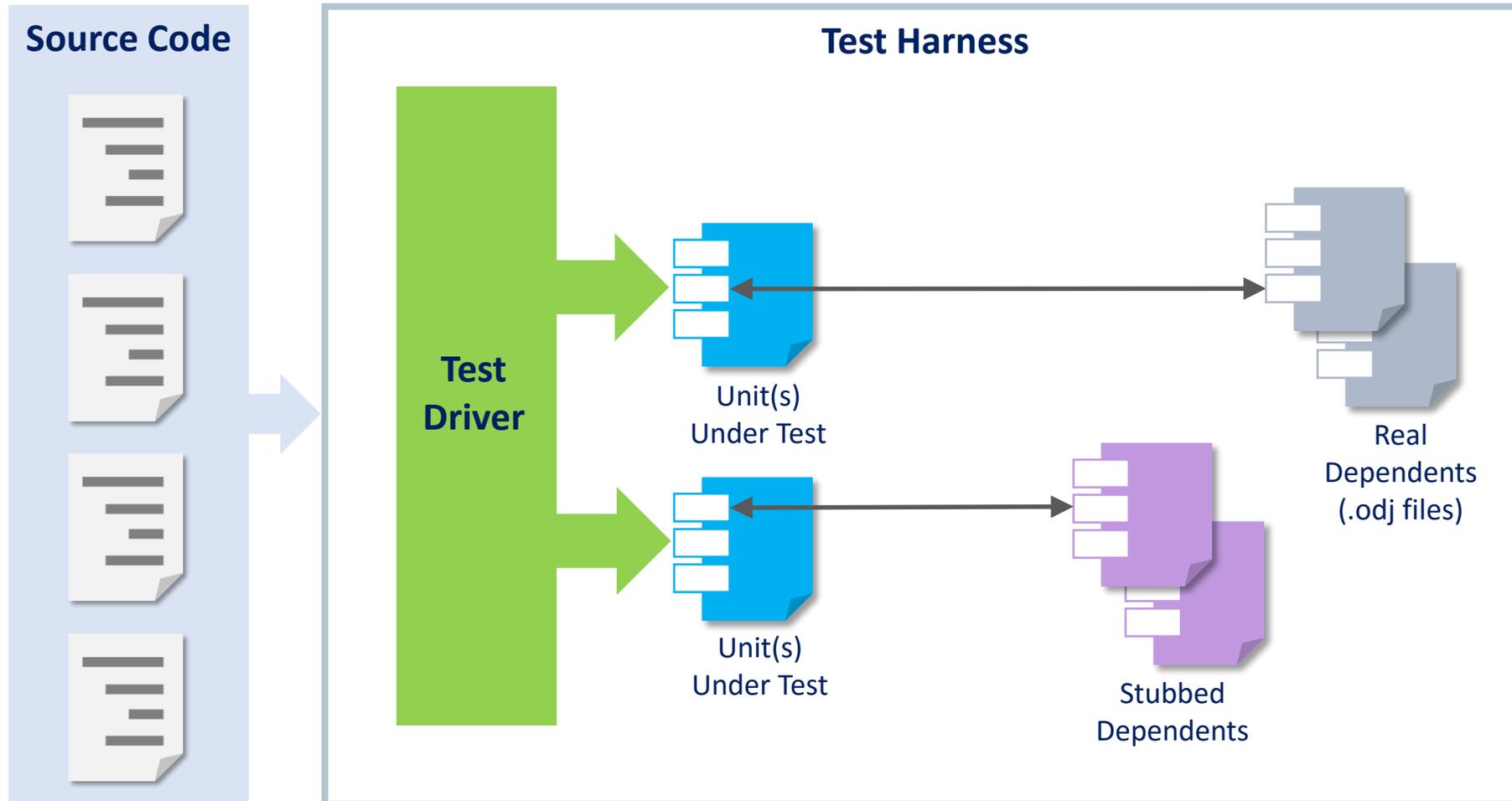
```
manager
Statements 100% Branches 100%
2 2 * TableData.IsOccupied = v_true;
2 3 * TableData.NumberInParty = TableData.NumberInParty + 1;
2 4 * TableData.Order[Seat] = Order;
2 5 * /* Add a free dessert in some cases */
2 6 * AddIncludedDessert(&TableData.Order[Seat]);
2 7 * switch(Order.Entree)
2 8 * {
2 9 (T) case NO_ENTREE :
2 10 * break;
2 11 (T) case STEAK :
2 12 * TableData.CheckTotal = TableData.CheckTotal + 14.0;
2 13 * break;
2 14 (T) case CHICKEN :
2 15 * TableData.CheckTotal = TableData.CheckTotal + 10.0;
2 16 * break;
2 17 (T) case LOBSTER :
2 18 * TableData.CheckTotal = TableData.CheckTotal + 18.0;
2 19 * break;
2 20 (T) case PASTA :
2 21 * TableData.CheckTotal = TableData.CheckTotal + 12.0;
2 22 * break;
2 23 * }
2 24 * UpdateTableRecord(Table, TableData);
2 25 * return 0;
```

Unit Testing

- Moves testing from a manual process to engineering rigor
- Makes developer testing a streamlined and repeatable process
- Proven reduction in cost by automated generation of test harness and stubs
- Intuitive user interface for generating test cases
- Can be used on both new and legacy applications
- Supports C, C++, and Ada



C, C++, Ada Embedded Unit Testing



Compiler, Debugger, RTOS, and Emulator

- Easily run unit tests and capture code coverage on embedded targets or simulators
- From bare-metal up to fully featured OS
- Seamless integration with cross-development tools
- Allows push button on-target and simulator testing
- Integrated with over **300** compiler, target, and runtime combinations



Q

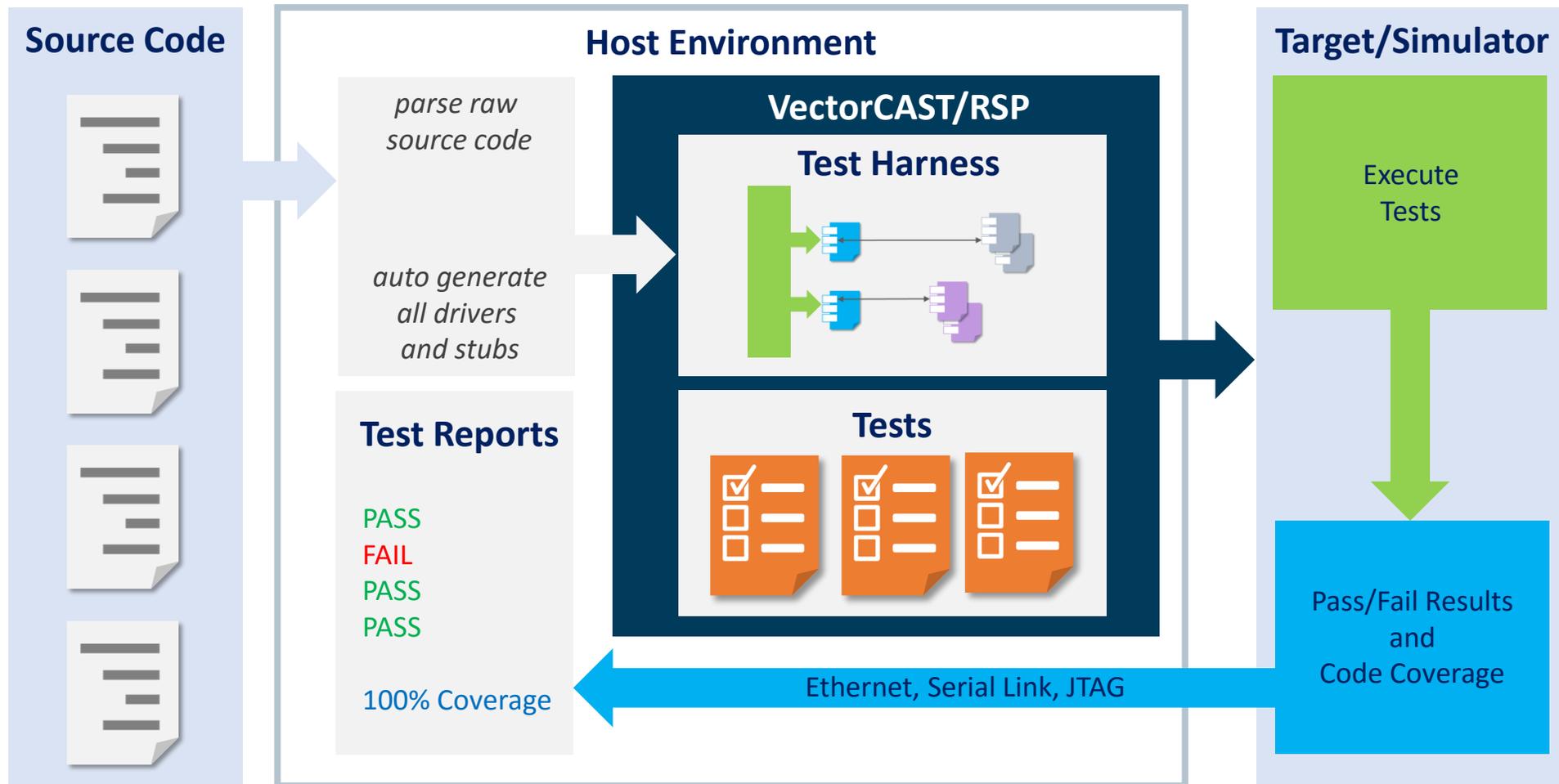
What do all these companies have in common?

A

None of these companies offer a test automation platform.

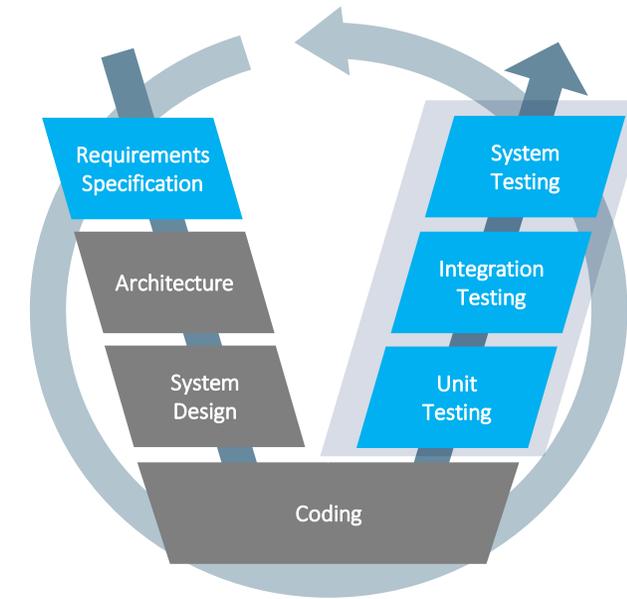
VectorCAST provides test automation to all of these environments.

On-Target Embedded Unit Testing



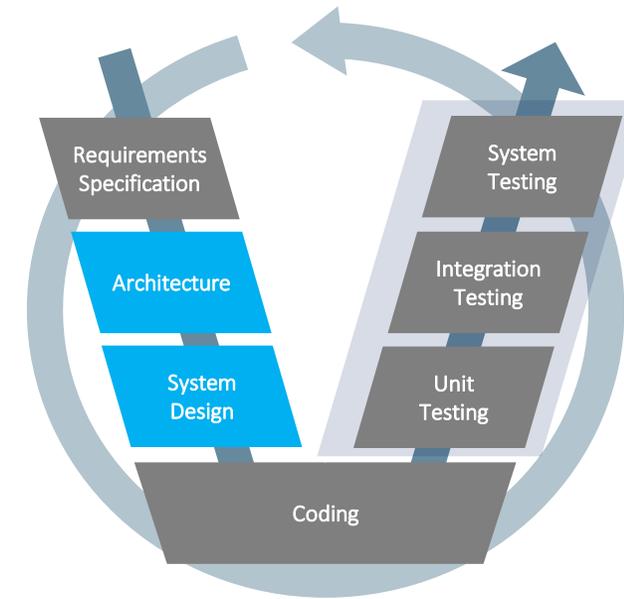
ALM and Requirements Gateway

- Provides traceability between software requirements, test cases, and code coverage
- Permits the flow of data between a requirements management tool and the VectorCAST testing tools
- Provides visibility to ensure requirements are being tested and the results of the tests
- Integrated with the most popular requirements tools
- Satisfies traceability requirements of regulated industry standards



Model-Based Development

- Create VectorCAST test environments from code generated from modelling tools
- Automatically convert model-based tests to code-based tests in VectorCAST
- Provides projects the ability to test in a target environment



PTC[®]
aTegO[®]

MathWorks[®]
MATLAB[®]
& SIMULINK[®]

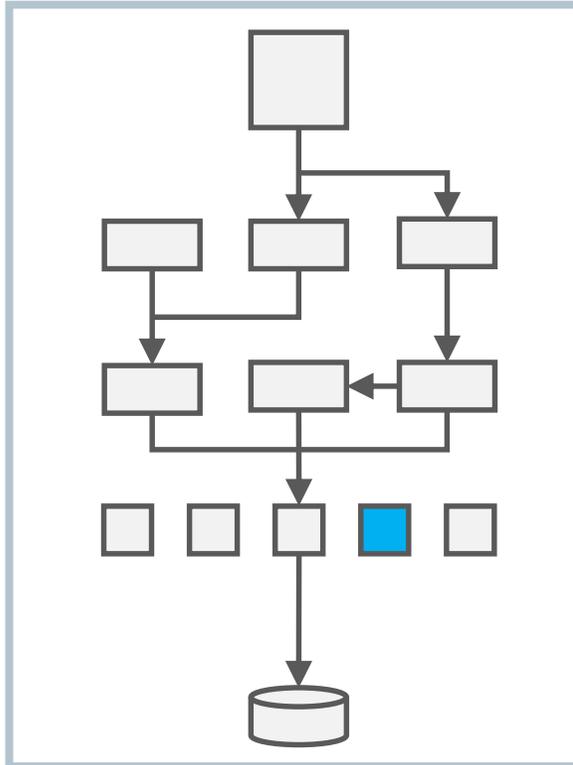
ANSYS[®]

IBM Rhapsody[®]

VECTOR[®]
software

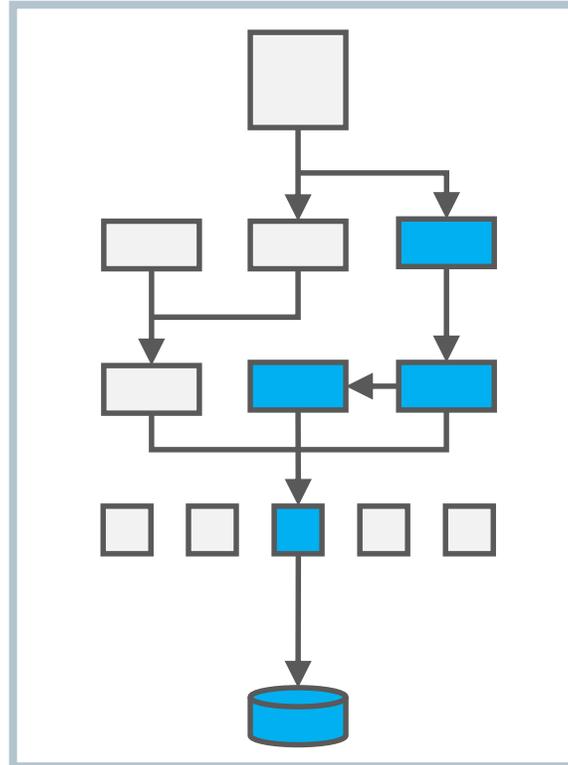
C, C++, Ada Embedded Integration Testing

Unit Testing



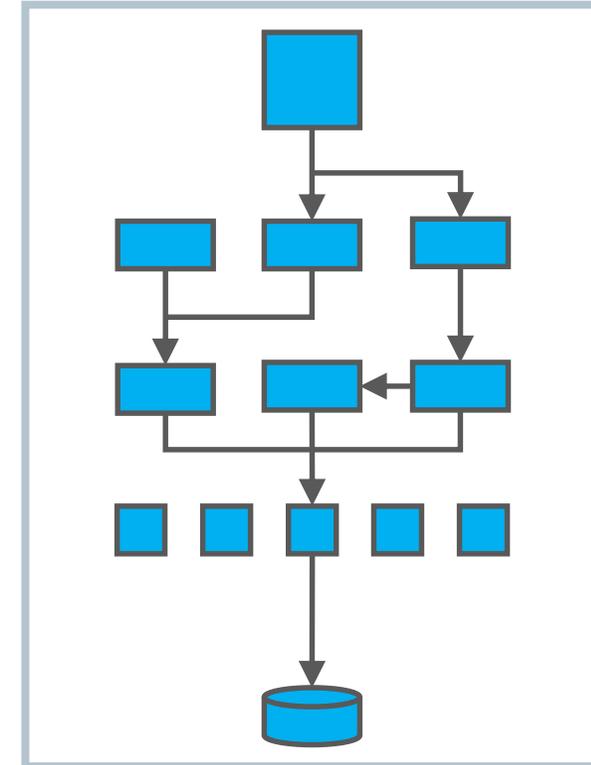
Individual units or modules are tested. It involves testing of source code by developers.

Integration Testing



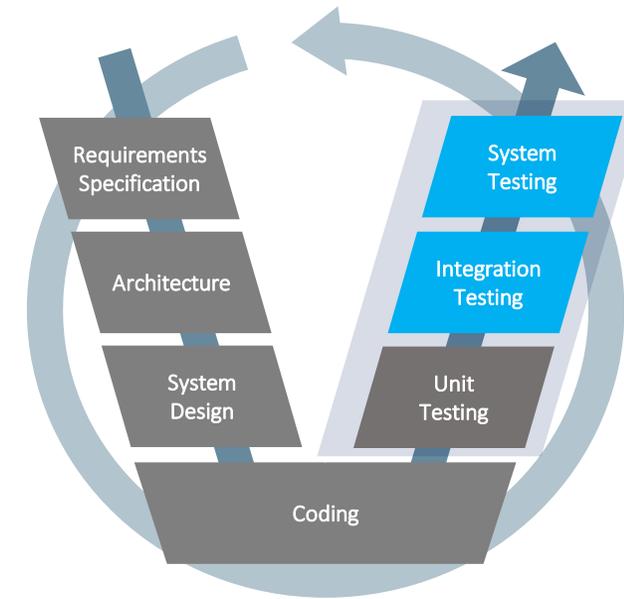
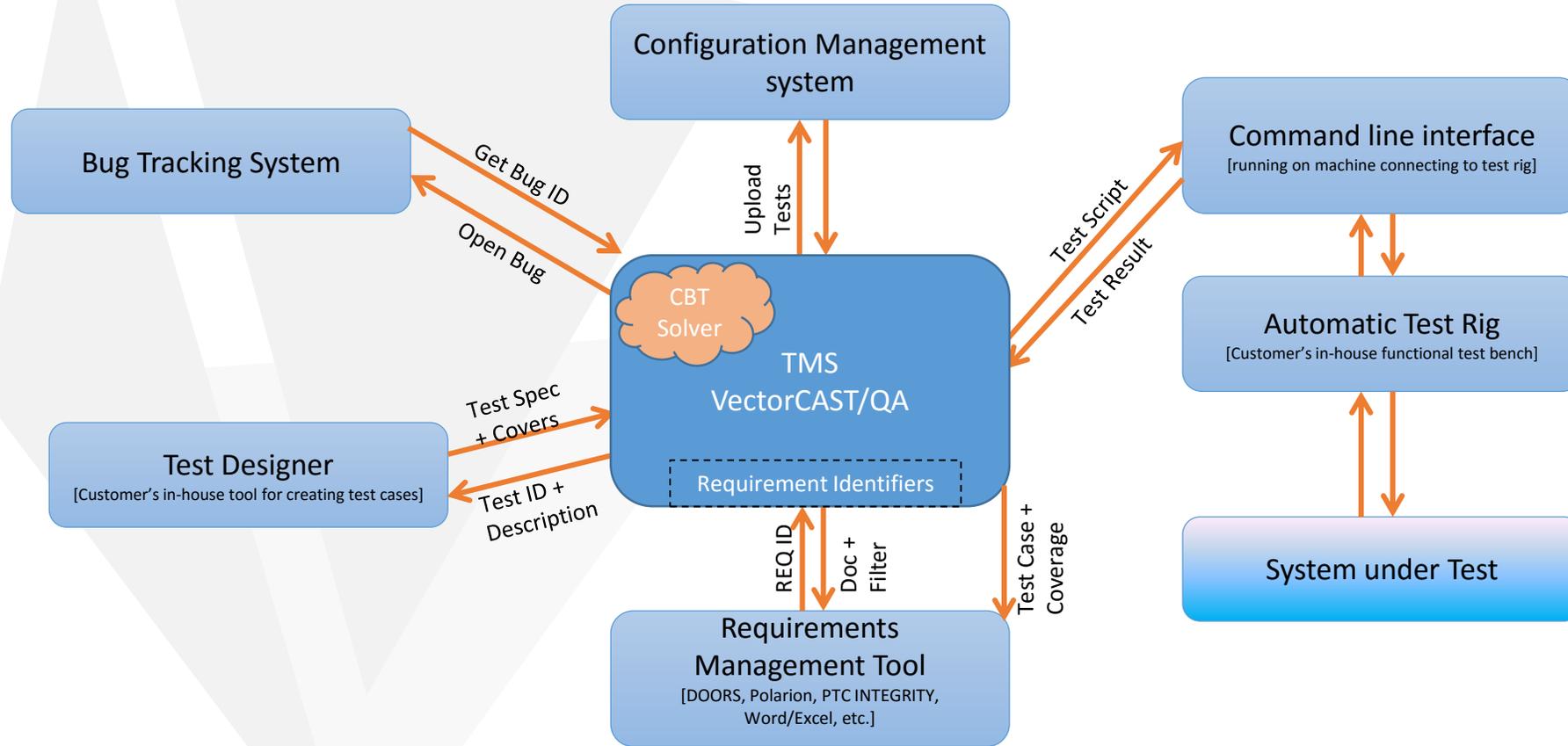
Individual modules are grouped together and tested. The purpose is to determine that modules are working as expected once they are integrated.

System Testing



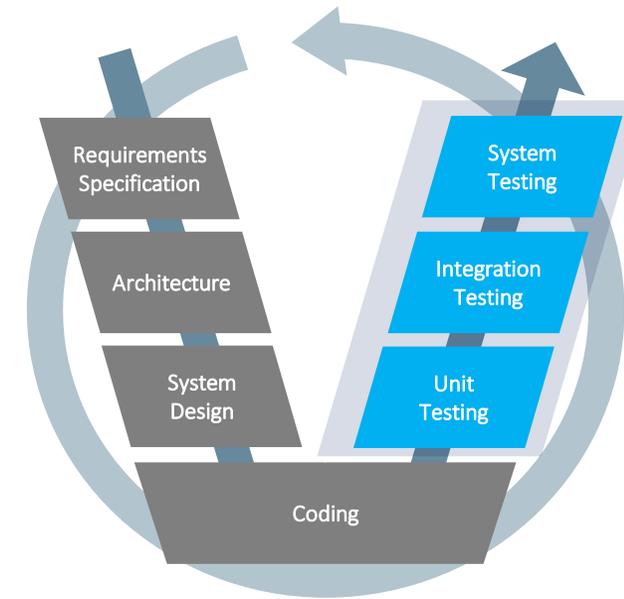
Testing is performed on the whole system by checking whether the system or application meets the requirement specification document.

Integration and System Test Automation



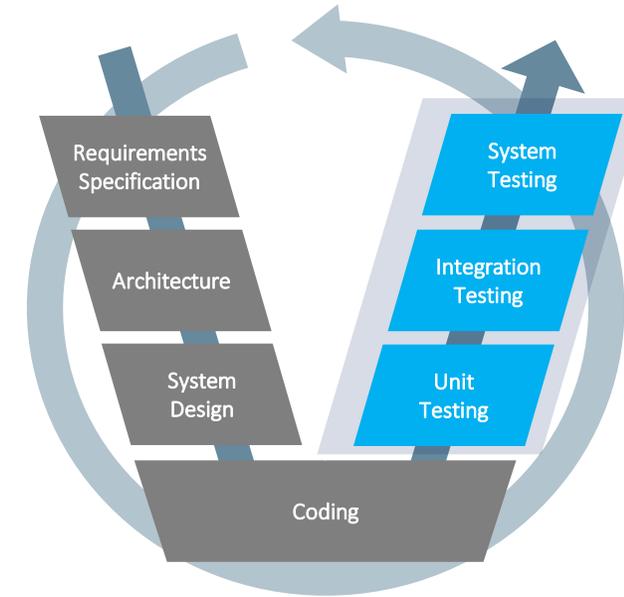
Test Management Interface

- Provides a single point-of-control for all unit and integration test activities
- Provides a single point of code coverage data aggregation across all phases of the testing lifecycle
- Helps with Release Readiness assessment
- Test common code across multiple configurations, easily check test results with a different compiler



Continuous Testing

- All VectorCAST tools have a command line interface
 - Versatile and ready for integration into any CI system
- Jenkins Integration
 - Creates jobs to run VectorCAST tests
 - View coverage and pass/failure trends
- Allows tests to be easily run by every developer on every source change



 **Bamboo**



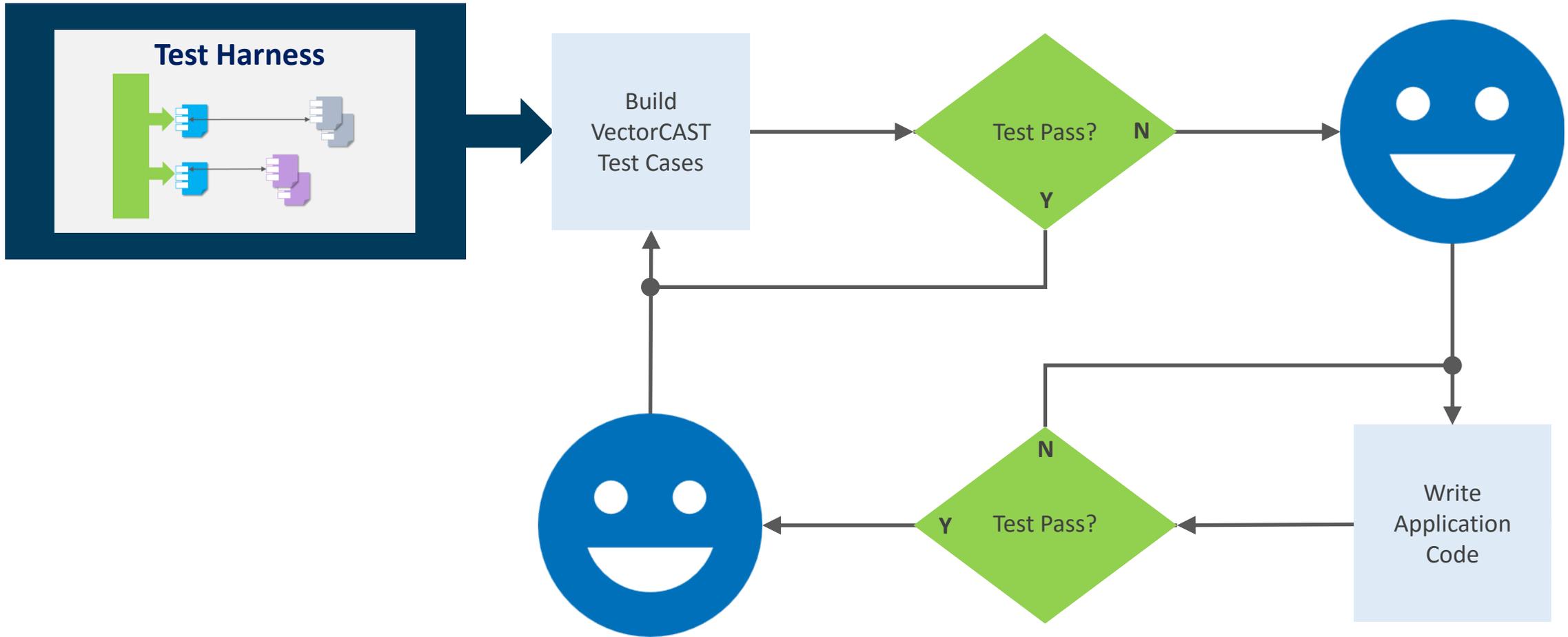
Jenkins



Hudson

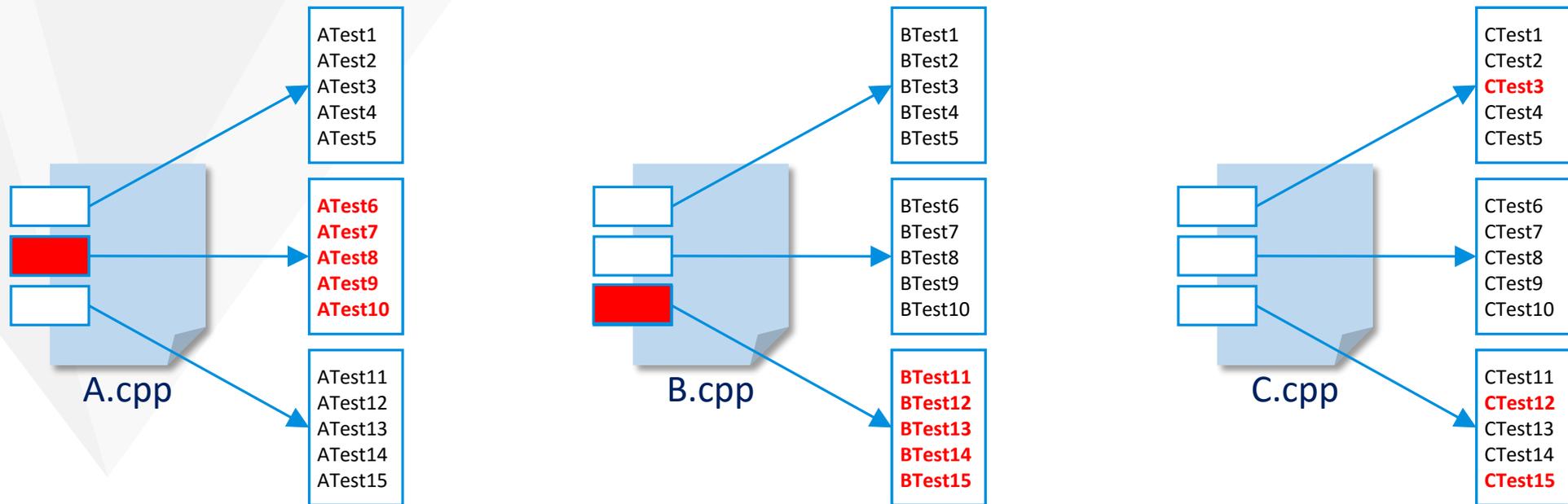
VECTOR
software

Continuous Testing Agile / Test Driven Development

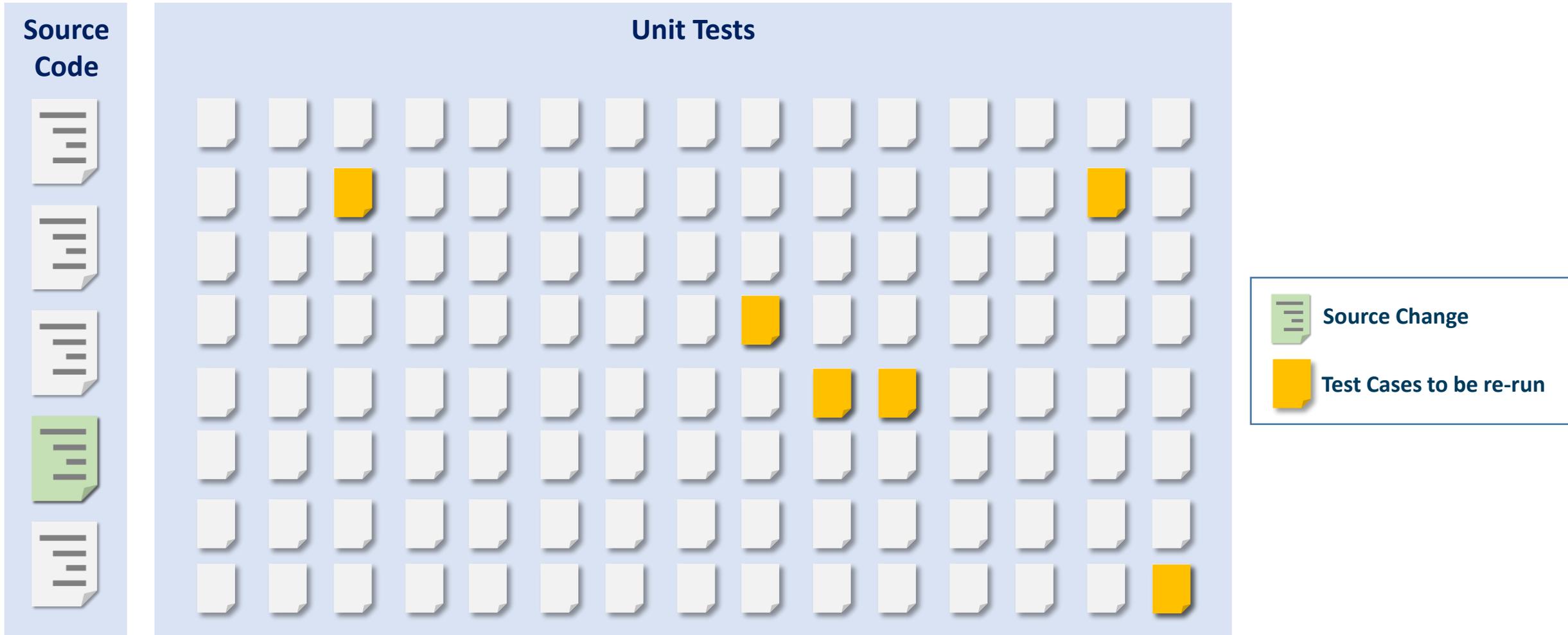


Change Based Testing

- Comparing changes is key to assessing risk
- Determine if a code change affects other parts of the system
- Prioritize tests based on risk, change, and criticality of modules
 - Change-based testing permits prioritized tests of modified modules
 - Regression testing ensures changes do not introduce new faults



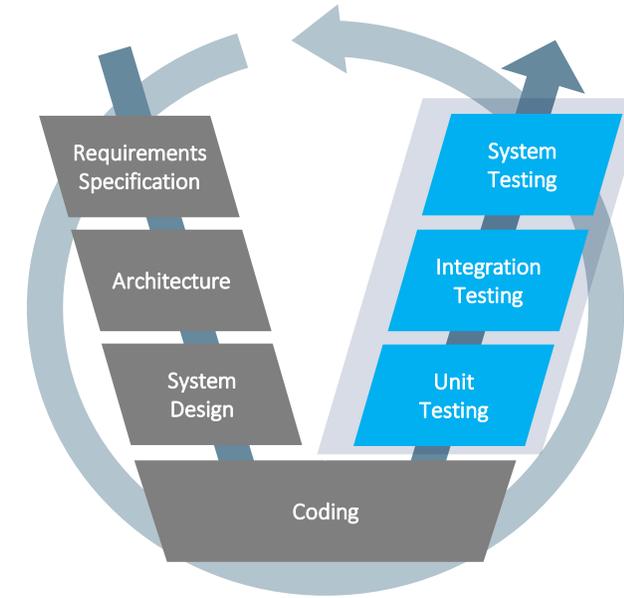
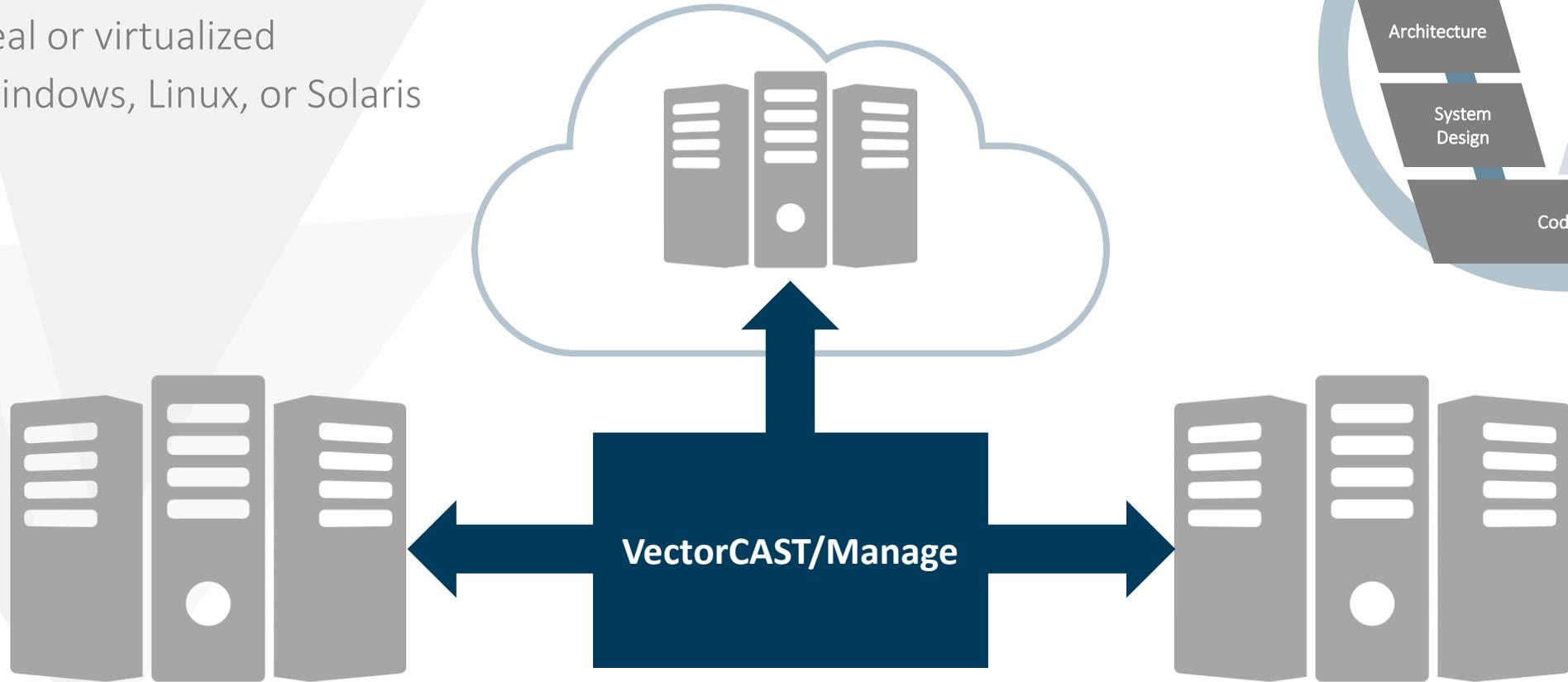
Change Based Testing – “Test Less”, “Fail Faster”



VectorCAST determines which test cases have been affected by a source change

Parallel Testing

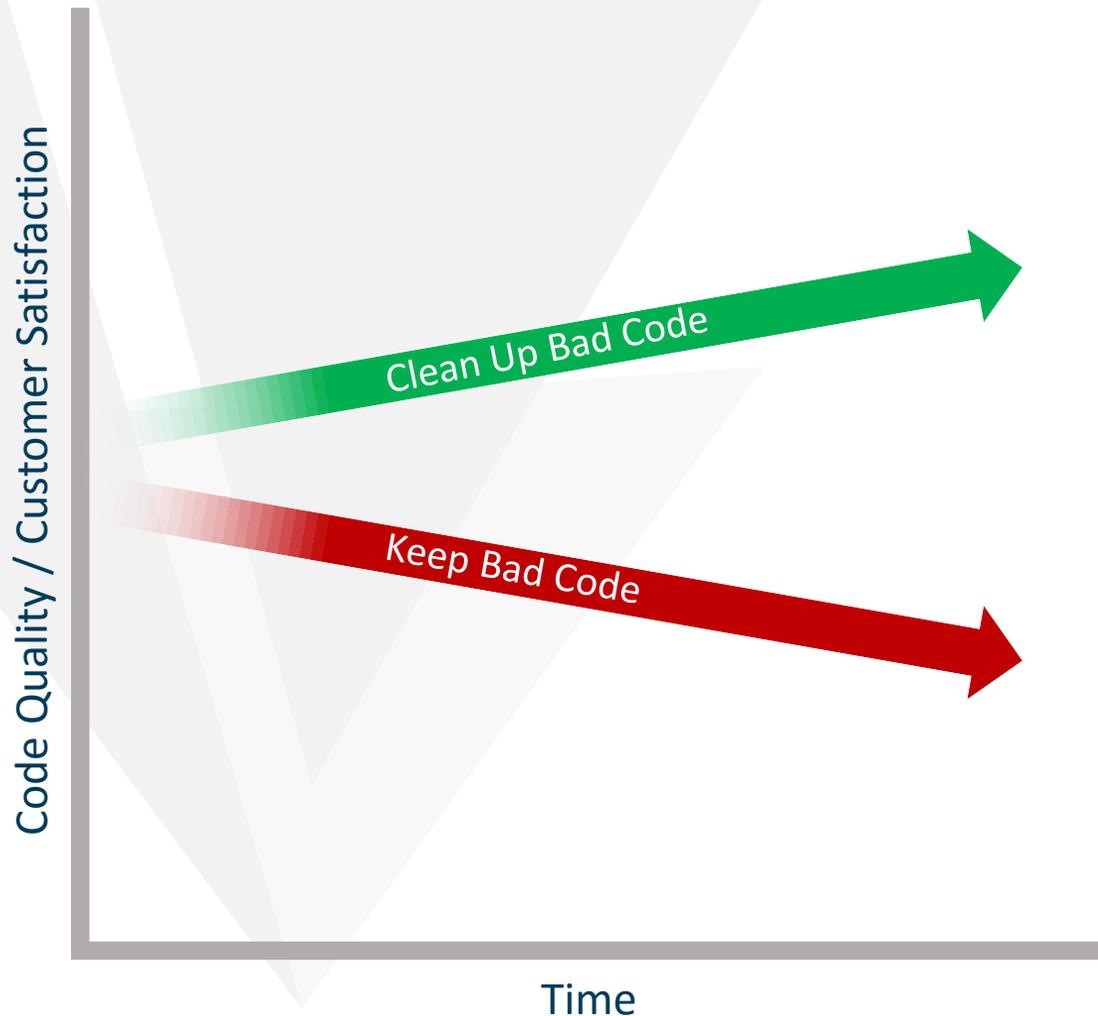
- User controlled execution of tests on clusters
 - Local or remote
 - Real or virtualized
 - Windows, Linux, or Solaris



Parallel Testing

- Jenkins support
 - VectorCAST integrates with Jenkins to allow for continuous integration and test
 - Perfect for large projects with many users and a lot of tests
 - Overnight or complete application test execution can be reduced from days to hours
 - Impact of Change analysis can be performed on the master project, greatly reducing the time it takes to identify regression errors
 - Speeds overall project testing time and reduces late-in-the-project side effects
- Wind River Simics support
 - Massively parallel testing
 - Used together with CI and VectorCAST

What is Technical Debt?



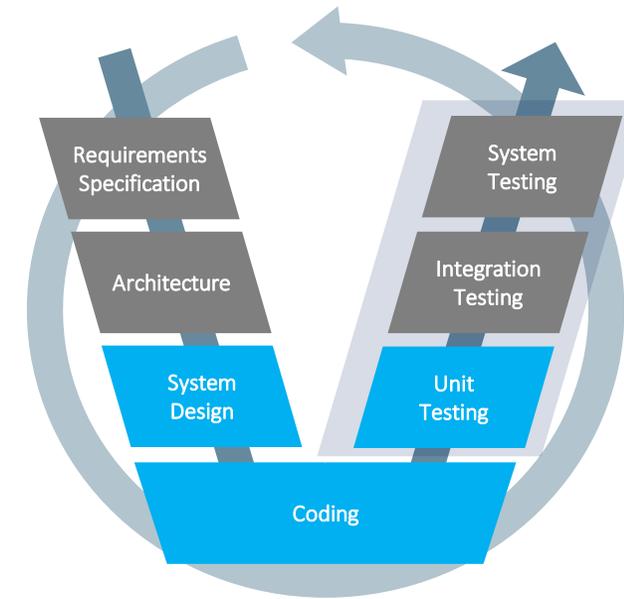
“Although immature code may work fine and be completely acceptable to the customer, excess quantities will **make a program unmasterable**, leading to extreme specialization of programmers and finally an **inflexible product**...”

“Every minute spent on not-quite-right code counts as interest on that debt. Entire engineering **organizations can be brought to a stand-still under the debt load** of an unconsolidated implementation, object-oriented or otherwise.”

Ward Cunningham, *The WyCash Portfolio Management System*
March 26, 1992

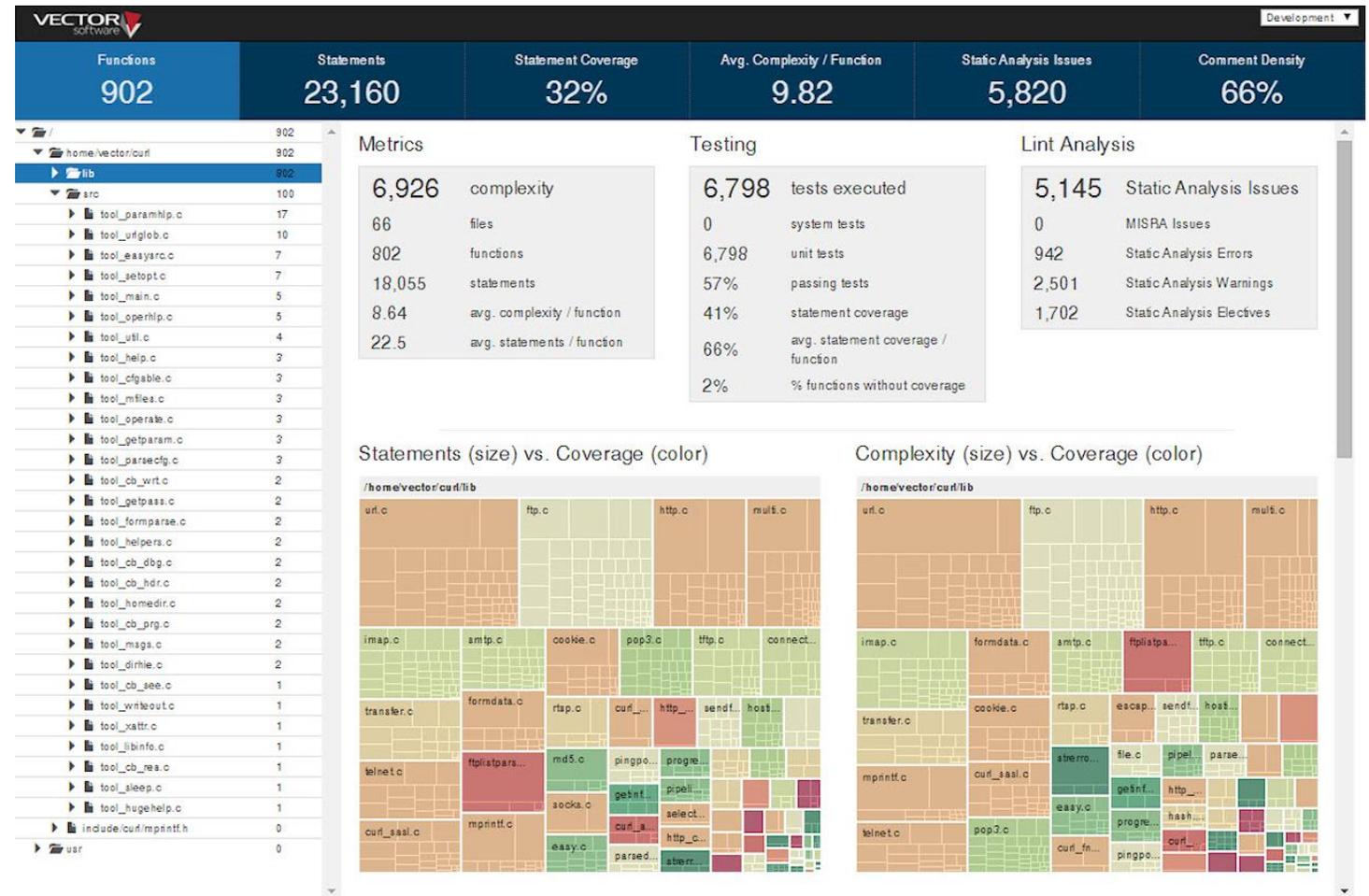
Legacy Code Baselineing

- Inadequately tested legacy projects can leverage automatic test case generation to baseline existing functionality
- Making the assumption that a fielded legacy application exhibits correct behavior
- Eliminate risk by automating testing on legacy code
 - Analyze project build
 - Automatically create test harnesses
 - Auto-generate test cases and capture expected results
 - Execute tests and capture code coverage
- Continue development or refactor



Web-based Dashboard

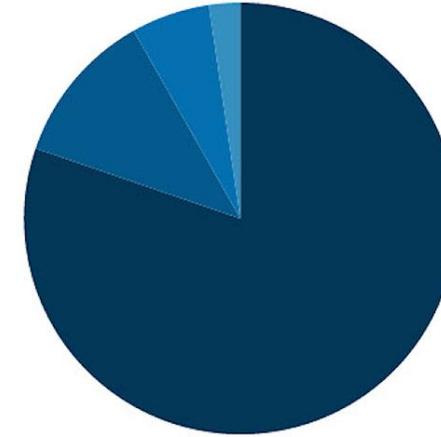
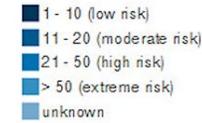
- Web-based dashboard view of software quality and test metrics
 - Fully customizable based on organization's goals
 - Drill-down details for each unit
 - Heat map identification of risk areas
 - Built in connectors for VectorCAST and third-party data
 - Active Development and Testing views



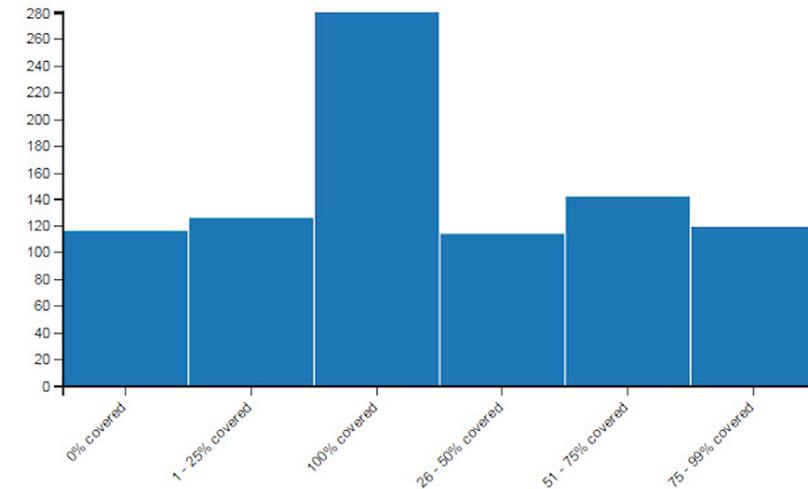
Quality Metrics

- Quantifiable code quality metrics for actionable intelligence
 - Make better release decisions
 - Assess and mitigate risk
 - Prioritize and allocate resources for maximum efficiency
 - Reduce technical debt
 - Increase test case quality
 - Analyze trends

Complexity Risk



Coverage Spread



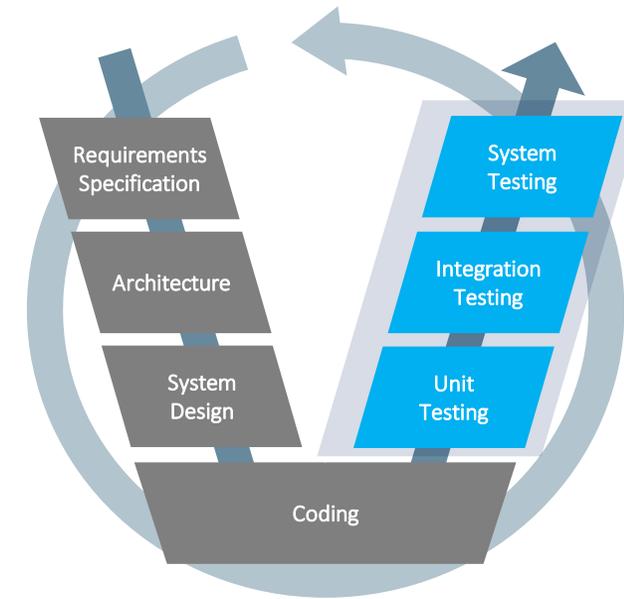
Real-time KPIs

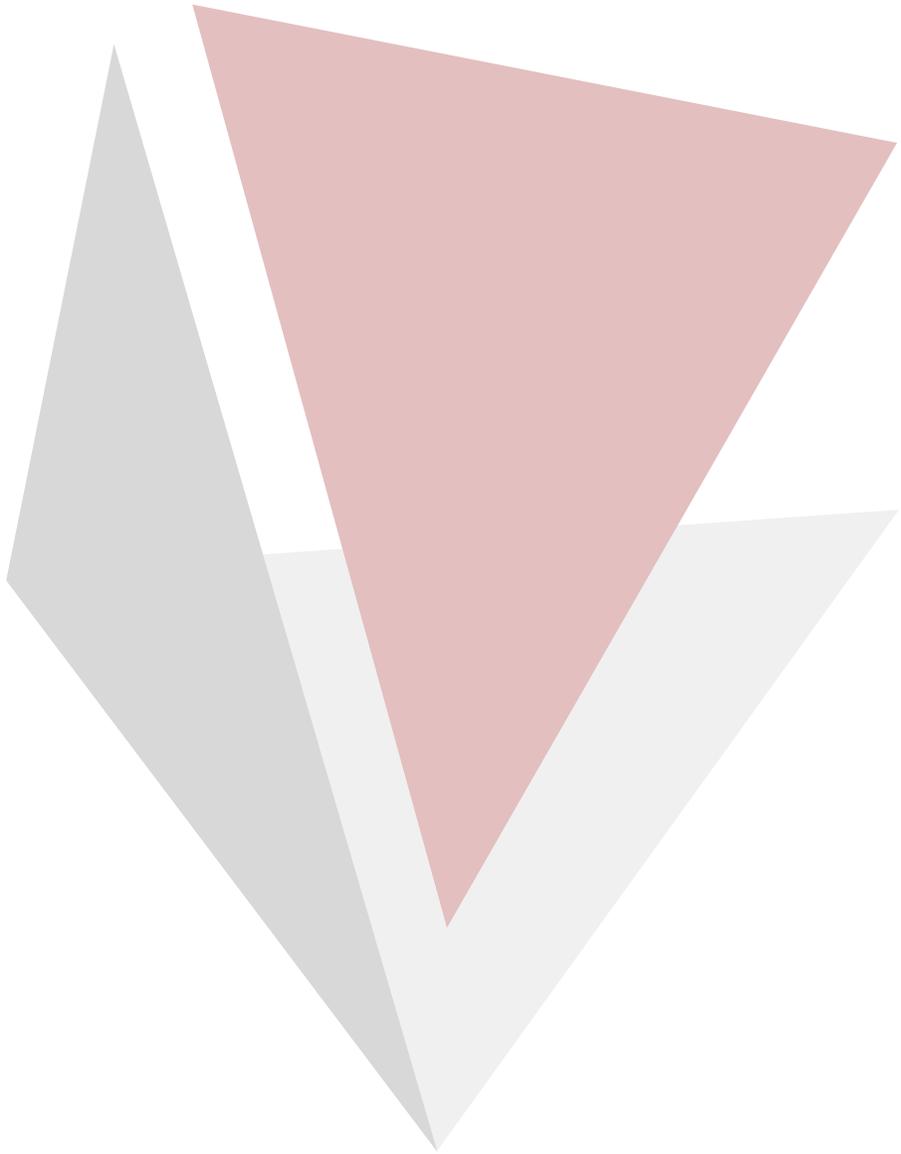
- Technical Debt
 - Code Complexity
 - Testing Completeness
 - Comment Density
- Test Case Quality
 - Test quantity
 - Test failures
 - Tests without requirements
 - Requirements without tests

Functions 902	Statements 23,160	Statement Coverage 32%
Avg. Complexity / Function 9.82	Static Analysis Issues 5,820	Comment Density 66%

Leverage Existing Testing Infrastructure

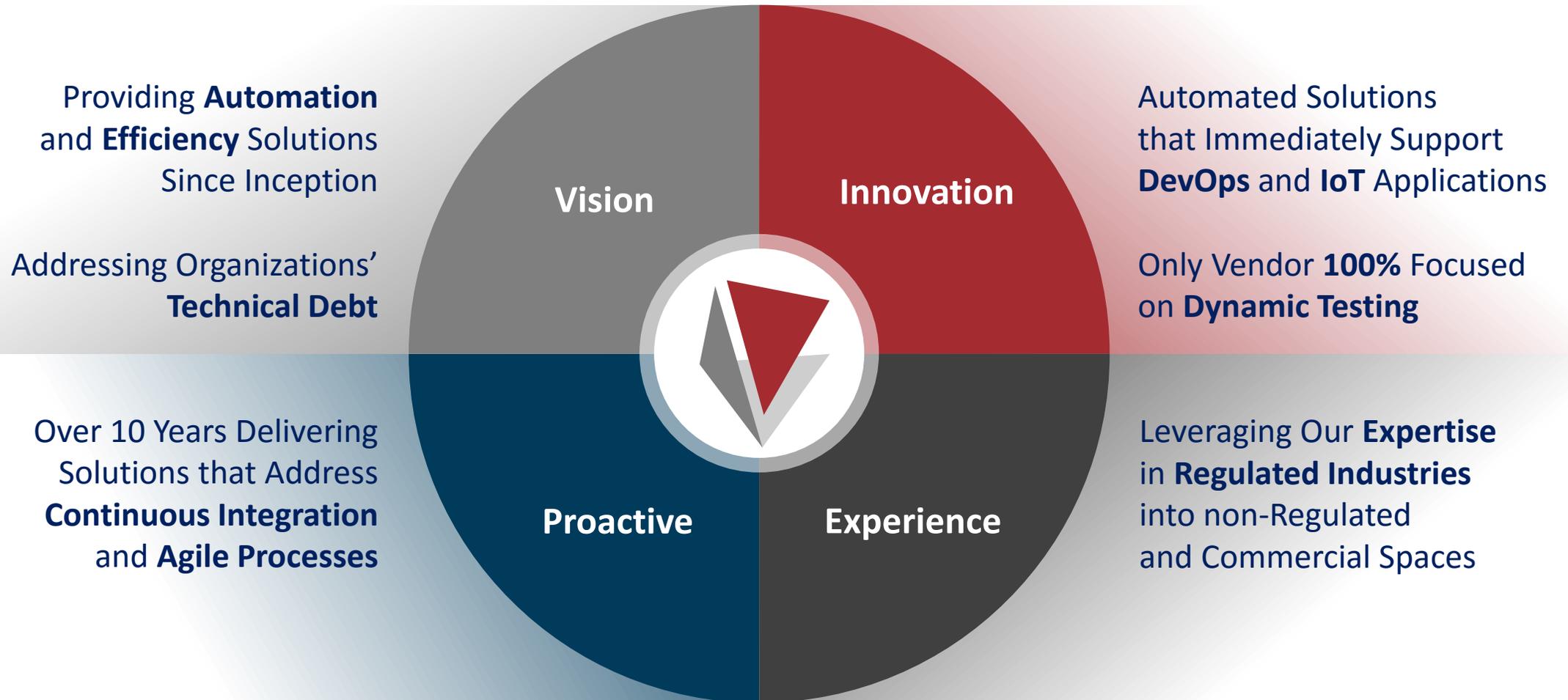
- Use VectorCAST/Cover to measure completeness of existing tests from 3rd party tools
- Import tests from 3rd party tools into VectorCAST
 - Reuse existing test data
 - Move to VectorCAST with minimal risk without loss of investment
- Leverage test data in existing CSV files



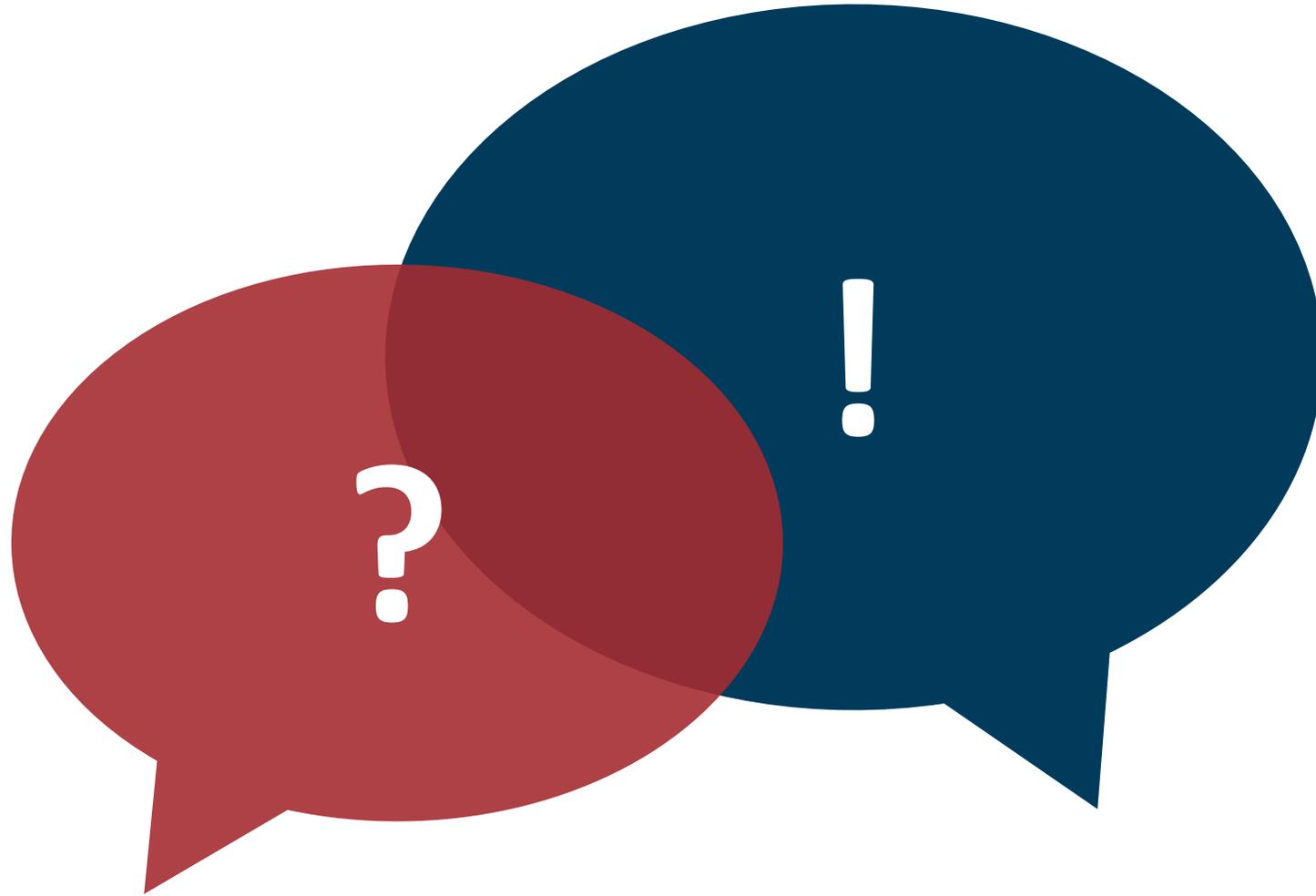


Summary

Vector Software Advantage



Questions and Answers



Connect With Us



Proven Test Solutions for Reliable Software

Americas

1351 South County Trail
East Greenwich, RI 02818
United States of America

sales@vectorcast.com
+1 401 398 7185

Europe, Middle East, and Africa

8 Duncannon Street
London WC2N 4JF
United Kingdom

sales@vectorcast.com
+44 203 603 0120

St. Töniser Str. 2a
47906 Kempen
Germany

sales@vectorcast.com
+49 2152 8088 808

Asia-Pacific and Russia

222-0033 Wise Next 3F
2-5-14 Shin-Yokohama, Kouhoku-ku
Yokohama, Kanagawa, Japan

sales@apac.vectorcast.com
China: +86 108 418 4600
India: +91 802 658 3300
Japan: +81 4528 5938 7
Singapore: +65 3158 2718

vectorcast.com



Additional Slides Here

