

# *Verification and Validation*

*What They Are and Why Both Matter*



**RBCS**

**TIME TESTED.  
TESTING IMPROVED.**

[www.RBCS-US.com](http://www.RBCS-US.com)



# Introduction

- ✦ Some people use *verification* and *validation* interchangeably
- ✦ Some people have made up new names for these well-established concepts
- ✦ However, there are significant differences, and using new names just confuses people unnecessarily
- ✦ In the cover painting, the woman is verifying the weight of the jewelry with a scale, but validating the beauty of the jewelry with her eyes
- ✦ Both verification and validation are part of testing, and both are essential to good testing
- ✦ A good balance of both is necessary – in testing and beyond – to successfully deliver a quality product
- ✦ Let's explore verification and validation, in the literature and by examples, and look at why both are essential to proper software testing...



# *Working Definitions to Get Started*

- ⊕ Verification
  - ⊕ Checking conformance to specifications (cf. Crosby's definition of quality)
  - ⊕ Did we build the thing right?
- ⊕ Validation
  - ⊕ Checking fitness for use (cf. Juran's definition of quality)
  - ⊕ Did we build the right thing?
- ⊕ If you're checking whether you built what you said you'd build, you're verifying
- ⊕ If you're checking whether you built what you should have built, you're validating
- ⊕ Verification and validation can each include positive and negative tests (and other checks)



# *What's with those ISTQB Definitions?*

- ⊗ ISTQB definitions derive from ISO 9000 standard
  - ⊗ Verification: *Confirmation by examination and through provision of objective evidence that specified requirements have been fulfilled*
  - ⊗ Validation: *Confirmation by examination and through provision of objective evidence that the requirements for a specific intended use or application have been fulfilled*
- ⊗ Notice *common* and different elements
- ⊗ The differences reflect the working definitions (i.e., conformance to specifications versus fitness for use)
- ⊗ Either can be done by:
  - ⊗ Direct examination; e.g., reviews, static analysis, dynamic tests
  - ⊗ Indirect examination; i.e., by auditing another organization's review, analysis, and test activities and results



# *Examples, Value, and Blending*

- ❖ Verification: developing a set of automated tests based on the acceptance criteria of a user story
- ❖ Validation: using a defect taxonomy to check for failures common to a particular application
- ❖ Validation is necessary because no set of specifications can perfectly reflect all aspects of a software product
- ❖ Verification is necessary because no set of validators (e.g., testers, users, etc.) can perfectly reflect all possible current and future uses
- ❖ Validation and verification are overlapping activities
  - ❖ Better specifications means more overlap between them
  - ❖ Worse specifications means more importance in validation
- ❖ Many testing and other quality activities include elements of both, so strictly classifying an activity often fails



## *V&V and Test Types*

- ❖ Static analysis: basically verification (if rules are seen as specifications)
- ❖ Reviews: a mix of both, but push as much verification as possible into static analysis
- ❖ White-box: usually verification
- ❖ Black-box (func. and non-func.): both, with heavy validation where specifications are missing
- ❖ Experience-based: usually validation
- ❖ Defect-based: usually validation
- ❖ Dynamic analysis: can be used to support both



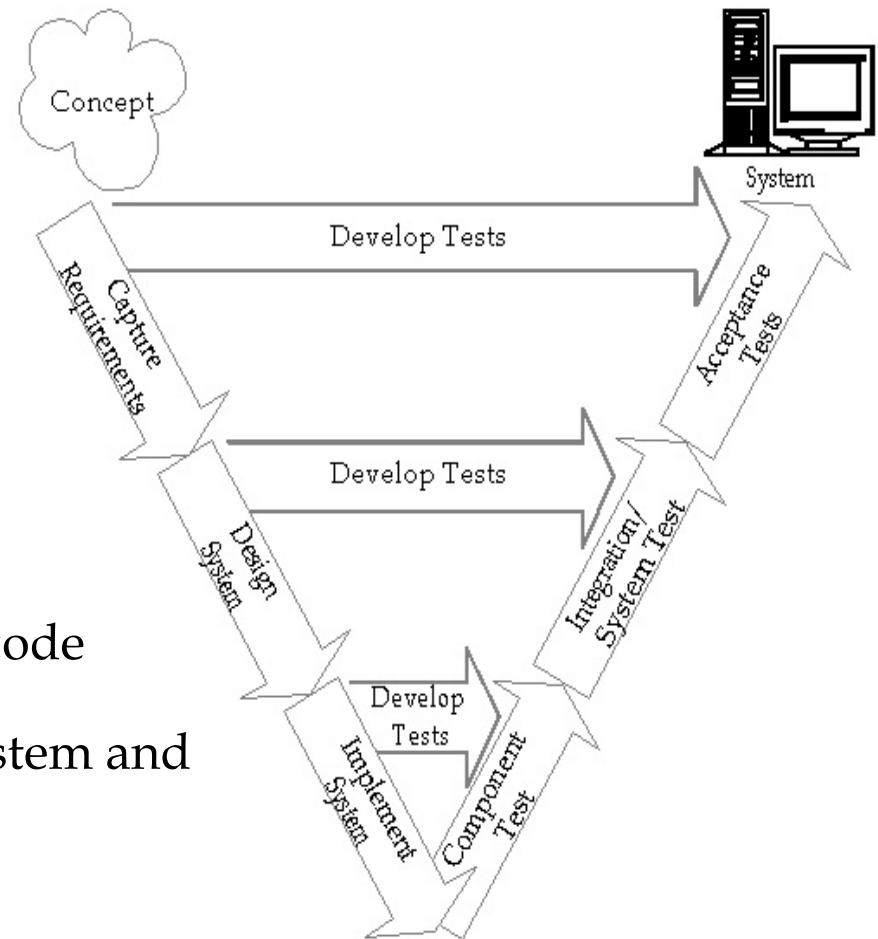
# Examples of V&V in the V Model

## Verification activities

- ❑ Requirements, design, and code reviews
- ❑ Static analysis
- ❑ Horizontal and vertical traceability checking
- ❑ Phase exit reviews
- ❑ Dynamic tests at all levels
- ❑ External audits

## Validation activities

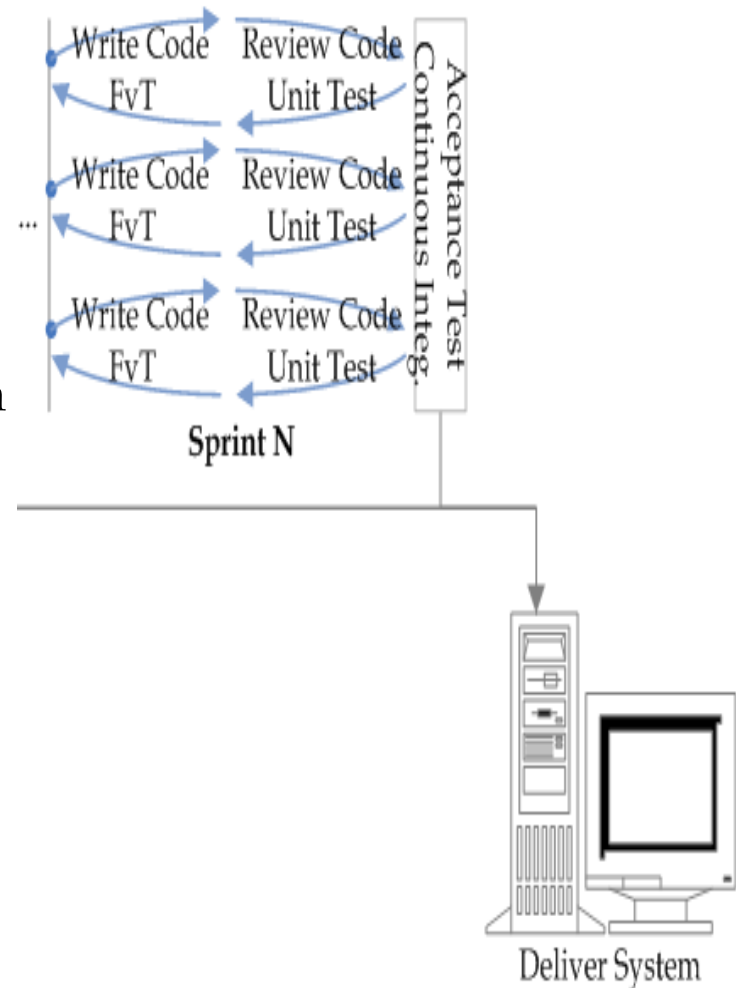
- ❑ Requirements, design, and code reviews
- ❑ Dynamic tests (mostly at system and acceptance level)
- ❑ Phase exit reviews
- ❑ Project status reviews





# Examples of V&V in Agile Model

- ❖ Verification activities
  - ❖ User story, design, and code reviews
  - ❖ Static analysis
  - ❖ Traceability checking
  - ❖ Dynamic tests at all levels (esp. unit tests and feature verification tests)
  - ❖ Continuous integration tests
- ❖ Validation activities
  - ❖ User story, design, and code reviews
  - ❖ Dynamic tests (mostly at system and acceptance level)
  - ❖ Product owner demos

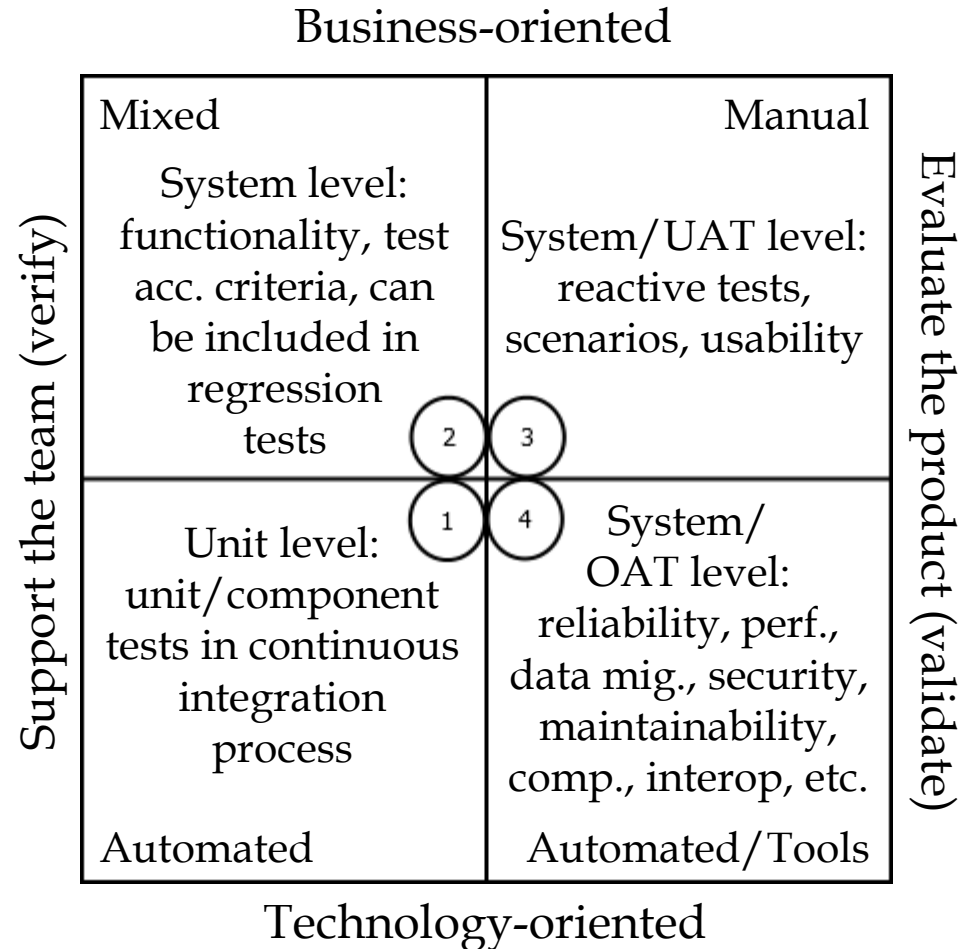






# V&V and Dynamic Test Quadrants

- ❖ The test quadrants associate test levels, types, goals, and focus
  - ❖ Covers important dynamic types/levels
  - ❖ Differentiates, describes types
- ❖ A better way to categorize tests
- ❖ Still imperfect and overlapping





# V&V and Test Pyramids

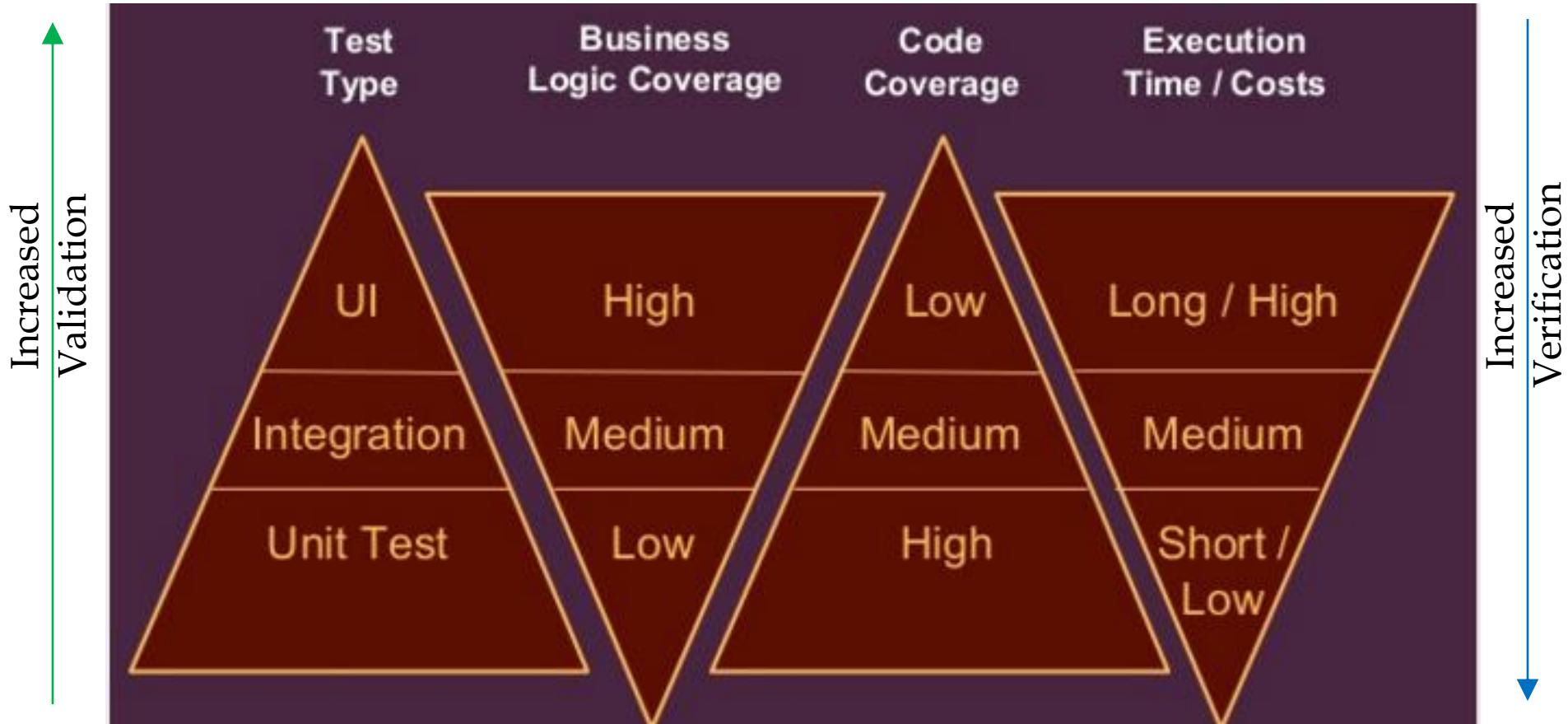


Figure courtesy Kristian Karl @kristiankarl



# V&V and Test Strategies

Strategy	Verification	Validation
Analytical	Usually, especially requirements based	Depends on the test basis analyzed
Model-based	Yes, if model is part of or derived from specifications	Yes, if model is based on observed or predicted usage
Methodical	Only when checklist is part of a specification	Yes, if checklists reflects user expectations/needs
Process/standard-compliant	Both: e.g., Agile, CMMi, FDA, FAA,	
Reactive	No	Yes
Consultative (directed)	Mix depends on direction given	
Regression-averse	Yes	No



# Conclusions

- ⊕ Don't make up new names (or promulgate made up new names) for concepts that are already well-established
- ⊕ Verification and validation are:
  - ⊞ Different in their goals and focus
  - ⊞ Partially overlapping
  - ⊞ Automatable, to varying extents with different tools
  - ⊞ Present in varying degrees across many test strategies and test types
  - ⊞ Relevant regardless of lifecycle
  - ⊞ Both essential to good testing
- ⊕ Know how to apply both in your testing efforts



# *To Contact RBCS*

For over 20 years, RBCS has delivered consulting, outsourcing, and training services to clients, helping them with software and hardware testing. Employing the industry's most experienced and recognized consultants, RBCS advises its clients, trains their employees, conducts product testing, builds and improves testing groups, and hires testing staff for hundreds of clients worldwide. Ranging from Fortune 20 companies to start-ups, RBCS clients save time and money through improved product development, decreased tech support calls, improved corporate reputation and more. To learn more about RBCS, visit [www.rbc-us.com](http://www.rbc-us.com).

Address: RBCS, Inc.  
31520 Beck Road  
Bulverde, TX 78163-3911  
USA

Phone: +1 (830) 438-4830

E-mail: [info@rbc-us.com](mailto:info@rbc-us.com)

Web: [www.rbc-us.com](http://www.rbc-us.com)

Twitter: @RBCS, @LaikaTestDog

Facebook: RBCS, Inc

[www.rbc-us.com](http://www.rbc-us.com)