

Risk-Based Testing

What It Is and How You Can Benefit



RBCS

**TIME TESTED.
TESTING IMPROVED.**

www.RBCS-US.com



Risk-Based Testing: Setting the Course

- ✚ For any complex system, an infinite number of tests exist...
- ✚ ...but you don't have forever to test!
- ✚ So, out of the infinite cloud of possible tests, testers must select a finite number
- ✚ Measured as a percentage of what we *could* test, our coverage is always 0%
- ✚ So, we need to be very, very selective and smart about our test coverage



What is Risk-Based Testing?

- Risk: the possibility of a negative or undesirable event or outcome
- Quality risk: the possibility that the product or system might fail to deliver one or more of the key quality attributes, endangering our ability to achieve the quality outcomes we want
- Risk-based testing uses an analysis of quality risks to prioritize tests and allocate testing effort
- Risk-based testing involves key business *and* technical project stakeholders to align the focus and sequence of testing with what quality means
- Risk-based testing also means managing project risks, which are possible events or outcomes that endanger successful completion of the project



What Are the Benefits?

- Risk-based testing delivers the following benefits
 - Running the tests in risk order gives the highest likelihood of discovering defects in severity order (“find the scary stuff first”)
 - Allocating test effort based on risk is the most efficient way to minimize the residual quality risk upon release (“pick the right tests out of the infinite cloud of possible tests”)
 - Measuring test results based on risk will allow the organization to know the residual level of quality risk during test execution, and to make smart release decisions (“release when risk of delay balances risk of dissatisfaction”)
 - If schedule requires, dropping tests in reverse risk order reduces the test execution period with the least possible increase in quality risk (“give up tests you worry about the least”)
- All of these benefits allow the test team to operate more efficiently and in a targeted fashion, especially in time-constrained and/or resource-constrained situations



Risk-Based Testing: Increasing Test Work?

- ❖ No, if anything, the long-run effect is to decrease effort
- ❖ Risk-based testing leads to more efficient testing, reducing the total effort for testing
- ❖ After the initial quality risk analysis, only periodic updates and maintenance of traceability is required



Risk-Based Testing: A Case Study

- CA, an RBCS client, implemented risk-based testing with our help on a pilot project (subsequently followed by two additional pilots)
- Pilot effort consisted of six main activities, roughly sequential
 - Training key stakeholders on risk-based testing
 - Holding a quality risk analysis session
 - Analyzing and refining the quality risk analysis
 - Aligning the testing with the quality risks
 - Guiding the testing based on risks
 - Assessing benefits and lessons
- This pilot project was described in an article in the December 2008 issue of Better Software
- Let's look at what happened in detail...



Training

- Participants and stakeholders went through a one day training which covered:
 - Principles of and rationale for risk-based testing
 - Categories of quality risks
 - Analyzing quality risks
 - Aligning testing with risk levels
 - Documenting quality risks
 - Monitoring quality risks during test execution
 - Reporting risk-based test results
- Training consisted of presentation, discussion, and an exercise



Quality Risk Analysis Session

- Quality risk analysis session consisted of two sub-sessions
- First sub-session: participants identified quality risk items
 - Main quality risk categories written on three whiteboards
 - Participants wrote risk items on sticky notes and posted them under the appropriate category
 - Sub-session lasted three hours and identified over 100 risk items
 - By-products included 11 project risks and 3 other issues
- Second sub-session: participants assessed the risk items
 - Worked as a team to rate likelihood and impact of each risk item
 - Likelihood was easy, but impact was the harder rating
 - Identified and merged duplicate/overlapping risk items
- At the end of the session, we had 92 risk items, 40% rated
- After the session, the test manager worked with the other participants to finalize the ratings



Excerpt at End of Session

Risk ID Number	Quality Risk Category	Specific Quality Risk	Likelihood	Impact	RPN	Extent of Testing	Other Action Needed	PRS/DDS Tracing	Total Items	Total Rated	Percent Rated
1.000	Functionality	Failures that cause specific features not to work							24	19	79%
1.001		Unix System services compatibility issue	4	3	12						
1.002		MQ Agent not properly removed from product	3	5	15						
1.003		Data collection changes incorrect	3	2	6						
1.004		Incompatibility with previous Sysview	5	2	10						
1.005		New commands do not function.	TBD	TBD	##		[Jim: Might need to break this out for each new command?]				
1.006		Changed commands do not function	TBD	TBD	##		[Jim: Might need to break this out for each new command?]				
1.007		Data lib commands don't work	3	2	6		[Jim: Might need to break this out for each command?]				
1.008		Problems with exploitation of z/OS 1.10 functionality.	3	TBD	##		[Jim: Need to get list of 1.10 functions to determine impact.]				
1.009		Audit logs too vague	TBD	TBD	##						
1.010		System overview changes incorrect	4	5	20						
1.011		Parallel Sysplex compatibility issues	5	2	10						

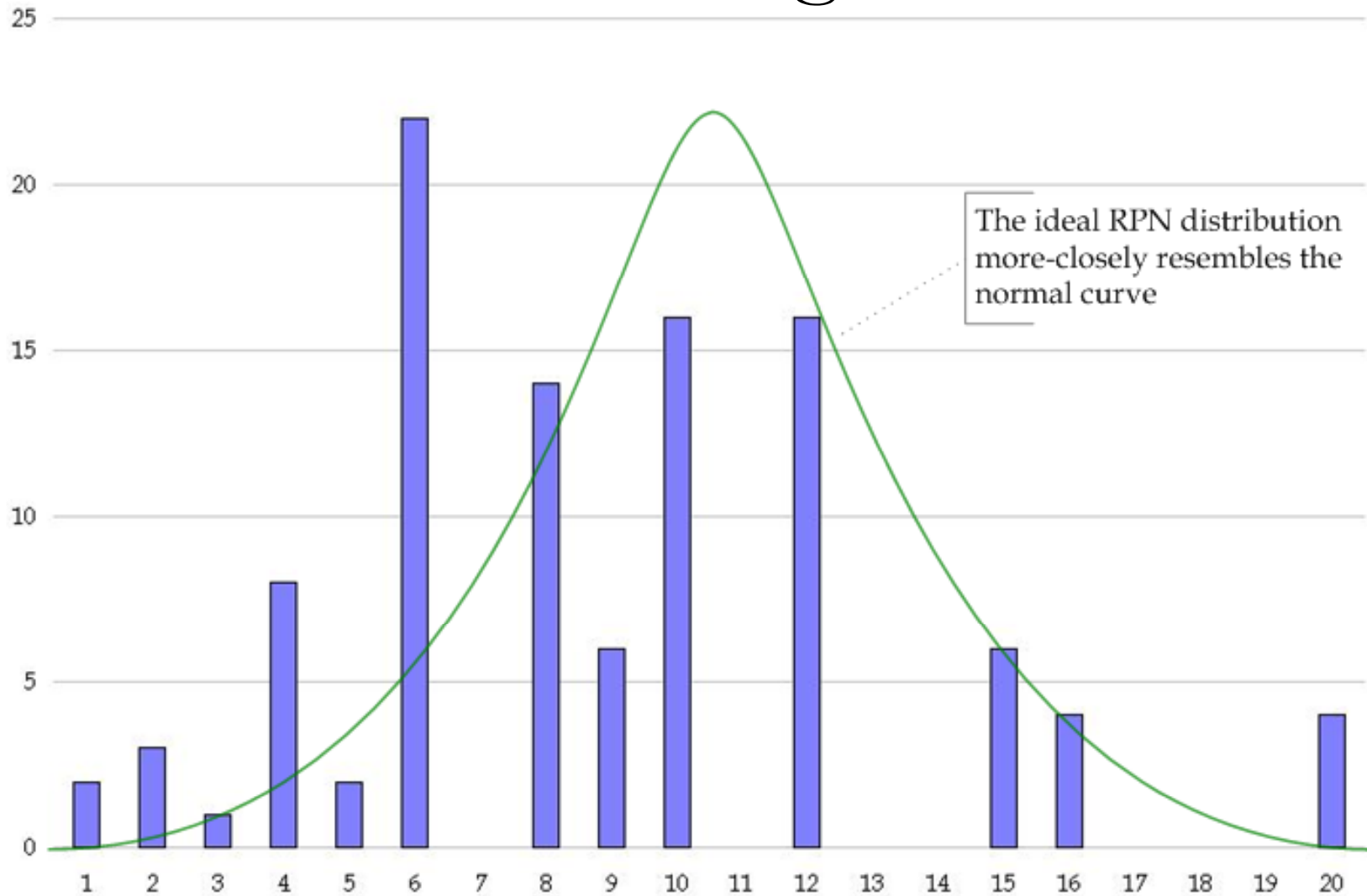


Analyzing and Refining the Assessments

- ✚ With ratings complete, the team calculated the risk priority number for each item
- ✚ With likelihood and impact rated on a five-point scale, risk priority numbers ranged from 1 to 25, with 1 the most risky and 25 the least risky
- ✚ A potential problem with quality risk analysis is a clumping of risk ratings
- ✚ Clumping commonly occurs for one or both of the following reasons
 - Teams consistently skew the impact of risk items; e.g., basing their ratings on worst-case outcomes
 - Teams use a scale with poorly defined distinctions
- ✚ You can check for clumping with a histogram...



RPN Histogram





Why the RPN Skewing?

Likelihood	Count	Impact	Count
1	5	1	10
2	9	2	52
3	25	3	32
4	39	4	8
5	26	5	2

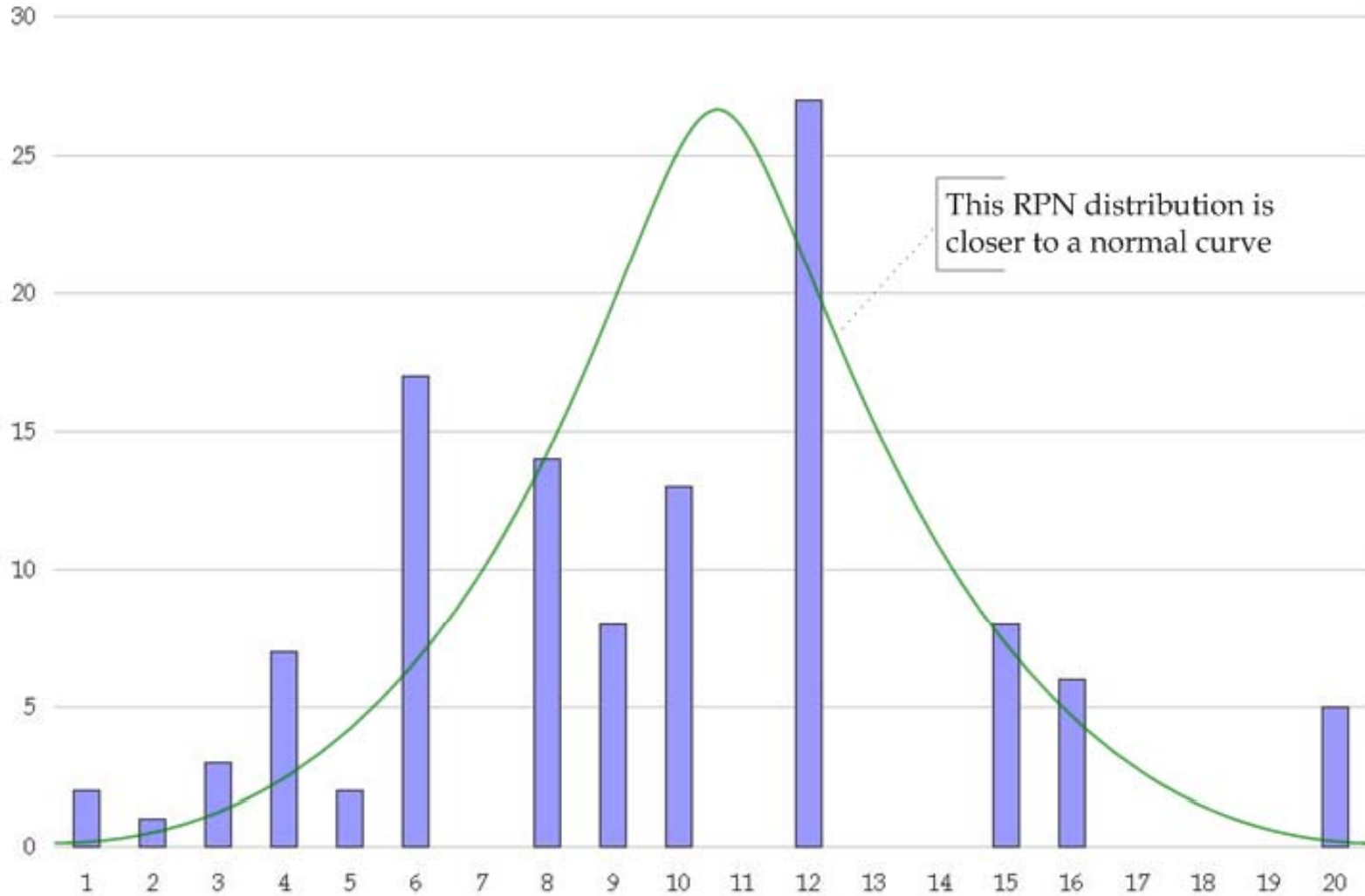
Looking at the left table, the likelihood ratings at first look skewed. But the product is mature and well-established, with a maintainable, solid code base and experienced developers.

The right table is problematic. Half of the impact ratings are 2, which skews the PRN rates.

To address this clumping, we adjusted the distinction between the impact ratings of two and three.

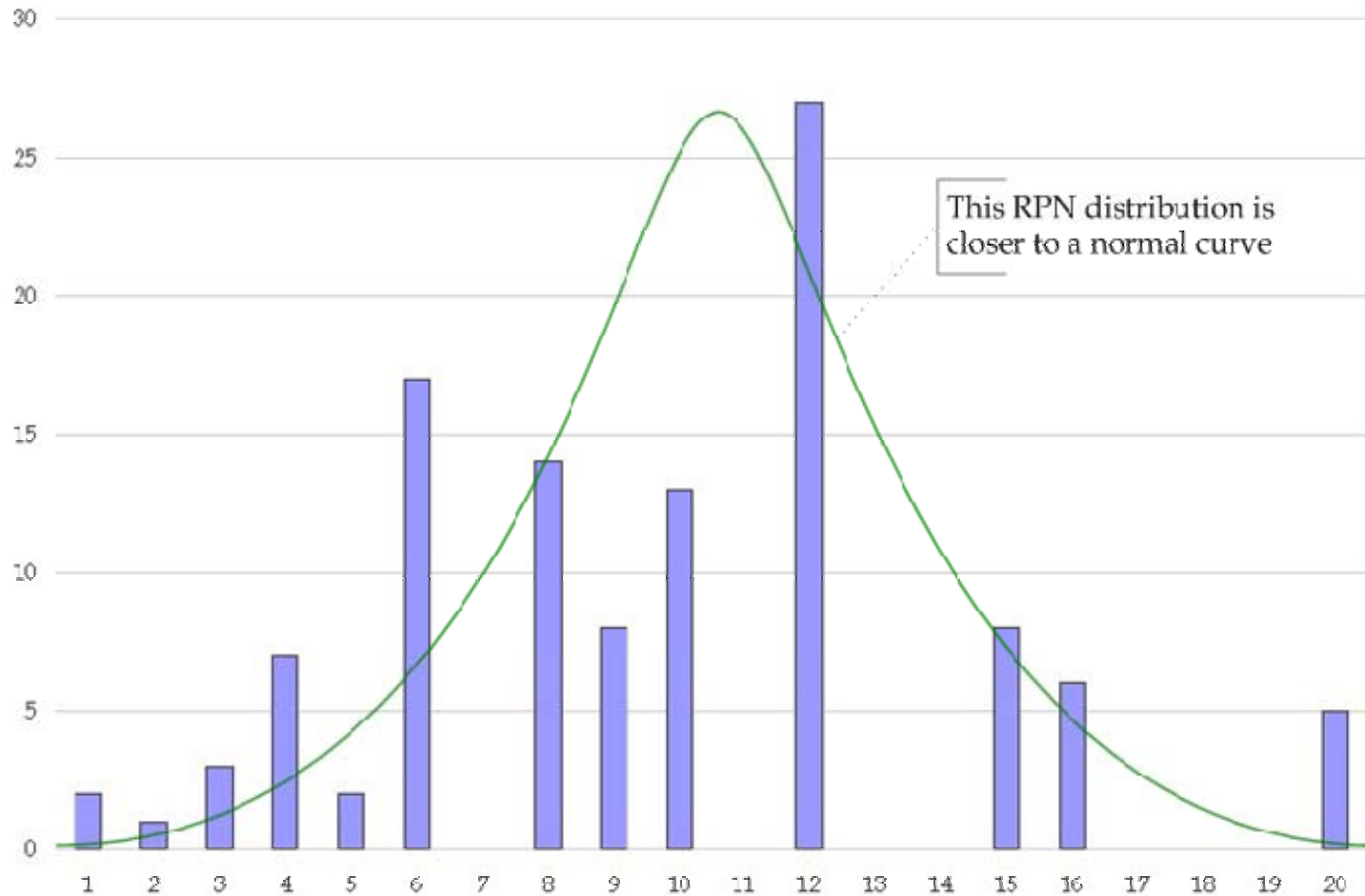


Adjusted PRN Histogram





Comparison of the Two RPN Distributions





Aligning Testing with Quality Risks

- With risk analysis complete, the test manager then aligned testing with risk
 - Map risk items to specifications (Product Requirements Specification and Detail Design Specification)
 - Map risk items to test cases
 - Evaluate and re-allocate testing effort based on the level of risk
 - Prioritize test cases based on risk levels of each risk item
- Effort allocation table customized for this application (next slide) so that all risk items receive at least cursory testing
- Mapping of risk items to specifications enables quick updates of risk items when specifications change



Effort Allocation

RPN Range	Extent of Testing	Comments
1 - 12	Extensive	Run a large number of tests that are both broad and deep, exercising combinations and variations of interesting conditions
13 - 16	Broad	Run a medium number of tests that exercise many different interesting conditions
17 - 20	Cursory	Run a small number of tests that sample the most interesting conditions
21 - 25	Opportunity	Leverage other tests or activities to run a test or two of an interesting condition, but only if it involves a very small investment of time and effort and only if the opportunity presents itself



Guiding the Project Based on Risk

- Based on finalized risk item RPN, effort, and test case/risk item traceability, the test manager prioritized the tests based on RPN
- This resulted in significant changes from the older, purely requirements-based approach
 - Previously, test tasks assigned based on tester expertise
 - This led to key player availability delays
 - With risk-based testing, the test manager avoided such bottlenecks
 - High-priority tests executed early, all low-priority tests executed later
- Risk-based testing also changed bug management
 - Bug severity rankings augmented with risk prioritization
 - Bug-fix efforts focused on high-risk problems



Conclusion: Benefits and Key Points

- The pilot project, as is typical, delivered benefits:
 - Intelligent effort allocation within constraints
 - Priority-order bug discovery, optimizing bug-fix time windows
 - Flexible handling of reduced time and resources
 - Optimization of quality within constraints by focused testing
- Key points:
 - Include business users and potentially even customers in analysis
 - Start risk analysis early in the lifecycle
- Bottom line: Risk-based testing allows teams to deliver increasingly sophisticated, complex products, within tightening constraints, prioritizing testing and balancing quality with other priorities



...*Contact RBCS*

For over a dozen years, RBCS has delivered services in consulting, outsourcing and training for software and hardware testing. Employing the industry's most experienced and recognized consultants, RBCS 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.

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

Phone: +1 (830) 438-4830

Fax: +1 (830) 438-4831

E-mail: info@rbc-us.com

Web: www.rbc-us.com