

# Quality in Web Time: the Art of the Possible

Quality Week Europe,  
November 11, 1998

Robert V. Binder  
RBSC Corporation  
[www.rbsc.com](http://www.rbsc.com)

# Component-Based Development

- ◆ Silver Bullet du jour
- ◆ Complex, least understood, newest typically most buggy.
- ◆ Components are not magic
- ◆ “Write Once, Debug Everywhere”
- ◆ Reliability matters to users

# Web-time Testing

- ◆ Weeks instead of months, etc.
- ◆ Market timing is critical
- ◆ Frequency-based testing increases reliability
- ◆ Testing by Fear, Uncertainty and Doubt (FUD) finds bugs
- ◆ Finite, limited test budget

# How to Balance?

- ◆ FUD-focus can miss high frequency
- ◆ Frequency-based can miss big bugs
- ◆ Weighted trade-off
- ◆ Allocate time per use-case

# Frequency Classification

- ◆ Per use-case
  - ◆ High: 60% of all usage
  - ◆ Medium: next 30% of all usage
  - ◆ Low: next 10% of all usage

# FUD Classification

- ◆ Per use-case, consider its components
  - ◆ Component stability/maturity?
  - ◆ Prior known-good usage?
  - ◆ Complexity of business rules?
  - ◆ Size/complexity of implementation?
  - ◆ Degree of code churn?

# FUD Factor Matrix

		<i>FUD</i>		
		High	Med	Low
<i>Frequency</i>	High	36%	18%	6%
	Med	18%	9%	3%
	Low	6%	3%	1%

# Test Effort Allocation

		High FUD	Med FUD	Low FUD
High Frequency	Total Hours	360	180	60
	Number of Use cases	12	5	11
	Test Hours per Use Case	30	36	5
Medium Frequency	Total Hours	180	90	30
	Number of Use cases	7	3	12
	Test Hours per Use Case	26	30	3
Low Frequency	Total Hours	60	30	10
	Number of Use cases	4	1	5
	Test Hours per Use Case	15	30	2

# Implications

- ◆ Plan testing as part of project
- ◆ Develop testable use-cases
- ◆ Supports Manual or Automated Testing
- ◆ Supports pure or mixed component-based development

# References

- ◆ Robert V. Binder. Use-cases, threads, and relations: the FREE Approach to system testing. *Object Magazine* 5(9):72-79, February 1996.
- ◆ -- Quality in Web Time: the Art of the Possible. *Component Strategies* 1(6), December 1998.