

[Presentation](#)

[Bio](#)

[Return to Main Menu](#)

P R E S E N T A T I O N

WG1

Wednesday, Dec 6, 2000

Risk: The New language of e-Business Testing

Paul Gerrard

Risk – The New Language of E-Business Testing

Paul Gerrard

Systeme Evolutif Limited
9 Cavendish Place
London W1M 0BL, UK
Tel: +44 (0)20 7636 6060
email: paulg@evolutif.co.uk
http://www.evolutif.co.uk/



Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 1

Paul Gerrard

Systeme Evolutif are a software testing consultancy specialising in client/server testing, RAD, test process improvement and the selection and implementation of CAST Tools. Evolutif are founder members of the DSDM (Dynamic Systems Development Method) consortium, which was set up to develop a non-proprietary Rapid Application Development method. DSDM has been taken up across the industry by many forward-looking organisations.

Paul is the Technical Director and a principal consultant for Systeme Evolutif. He has conducted consultancy and training assignments in all aspects of Software Testing and Quality Assurance. Previously, he has worked as a developer, designer, project manager and consultant for small and large developments using 3 and 4GLs. Paul has engineering degrees from the Universities of Oxford and London, is Special Projects Secretary for the BCS SIG in Software Testing, a member of the BCS Software Component Test Standard Committee and Former Chair of the IS Examination Board (ISEB) Certification Board for a Tester Qualification whose aim is to establish a certification scheme for testing professionals and training organisations. He is a regular speaker at seminars and conferences in Europe and the US, and won the 'Best Presentation' award at EuroSTAR '95. He is also webmaster for the www.evolutif.co.uk web site.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 2

Caution!

You won't find this in
a text book...
...and it's not about
E-Business either.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 3

Agenda

- The E-Business Testing Challenge
- The Language of Risk
- How Much Testing is Enough?
- When Should We Stop Testing?
- When is the Product "Good Enough"?
- Risk and Benefits-Based Test Reporting
- Close.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 4

The E-Business Testing Challenge

- To assess E-Business risks
- To estimate overall test effort
- To gain consensus on the amount of testing
- To get budget to do do enough testing
- To implement and execute tests
- To provide information for a risk-based decision on release
...under pressure...

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 5

Pressure? What pressure?

- Compressed timescales (2-5 months)
 - "Current finance runs out in 4 weeks, we need the next round of Venture Capital..."
 - "Initial Public Offering on the way..."
 - "Prime-time multimedia ad campaign planned..."
- Risky project foundations
 - New technology
 - Inexperienced developers
 - Marketer-managed development
 - Lack of testing infrastructure
 - No-one on the project understands testing.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 6

What do the textbooks say?

- Rigorous, vigorous testing
 - Inspections and reviews
 - White-box test techniques and coverage analysis
 - Test design: thorough analysis of perfect requirements
 - Inflexible acceptance criteria that must be met
 - Risks are bad, testing addresses risk, so use the most rigorous techniques possible
- BUT
 - No allowance for poor or missing requirements
 - No compromise on quality
 - No limits on time, resources or technical infrastructure
 - No mention of benefits of early release.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 7

But this isn't new

- Testers have always worked in pressured environments
- Management don't appreciate testing
- Testers never get the budget required
- When development slips, testing gets squeezed
- We're at the wrong end of the project
- It's so unfair!

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 8

Perhaps we get what we deserve?

- Management don't appreciate testing
 - Do we ever explain testing in THEIR language?
- We never get the budget we ask for
 - Do we give them the assurance THEY require?
- When development slips, testing gets squeezed
 - Do managers really KNOW the risks they take?
- We're at the wrong end of the project
 - Perhaps we have NOTHING to say at the beginning?
- It's so unfair!
 - Welcome to the REAL world!

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 9

So what's new?

- The technology of course, but more importantly...
 - Everyone knows how unreliable the Web still is
 - Everyone knows *from personal experience* what can go wrong
- It's EASY to articulate the risk of failure of E-Business Applications.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 10

The Language of Risk

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 11

Risk: the language for E-Business testing

- We must talk to management and the stakeholders in THEIR language: Risks and Benefits
- They will make "better" decisions:
 - More test budget if the risks are severe
 - Deadlines are postponed when the situation requires it
 - Software releases where the risk of release is known
 - Risks and benefits of early release are visible
- The LANGUAGE of RISK
 - We **plan** in the **language of risk**
 - We **report progress** in the **language of risk**.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 12

Difficult questions for testers

- How much testing is enough?
- When (why?) should we stop testing?
- When is the product good enough for release?
- How good are we at testing anyway?

- ALL can be answered in the language of risk.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 13

How Much Testing is Enough? (planned tests)

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 14

“I need six testers for eight weeks...”

- The Project manager says
 - “Four testers for six weeks and that’s it!”
- The testers say
 - It’ll take longer than six weeks
 - It’ll cost more than the budget allocated
 - It’s too big/complicated/risky for us to do properly
 - “It’s just not enough”
- Was it ever possible to do “enough” testing in these circumstances?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 15

Testing is time delimited...always

- No upper limit to the amount of testing
- Even in the highest integrity environments, time and cost limit what we can do
- Testing is about doing the best job we can in the time available
- Testers should not get upset, if our estimates are cut down (or ignored)
- The benefits of early release may be worth the risk
- But who knows the status of the risks and benefits?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 16

Did any tester ever get enough time to test?

- No, of course not
- We need to separate the two responses:
 - the knee-jerk reaction (a back-covering exercise prior to the post-disaster “I told you so!”)
 - a rational evaluation of the risks and response to doing ‘too little’
- Often, just like cub-scouts, testers “promise to do their best” and don’t make waves.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 17

Risk-based test planning

- If every test aims to address a risk, tests can be prioritised by risk
- It’s always going to take too long so...
 - Some tests are going to be dropped
 - Some risks are going to be taken
- Proposal:
 - The tester is responsible for making the project aware of the risks being taken
 - Only if these risks are **VISIBLE**, will management ever reconsider.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 18

So how much testing is enough?

- Enough testing has been planned when the stakeholders (user/customer, project manager, support, developers) approve:
- TESTS IN SCOPE
 - They address risks of concern
- THE TESTS THAT ARE OUT OF SCOPE
 - Risk is low OR these tests would not give confidence
- The amount and rigour of testing is determined by CONSENSUS.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 19

When Should We Stop Testing?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 20

The test plan says...

- Acceptance criteria:
 - all tests run
 - all incidents raised have been resolved
 - all faults found have been fixed, re-tested
 - all regression tests run without failure
 - etc. etc.
- ...but if we always run out of time
 - ...what's the point of acceptance criteria?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 21

Running out of time

- The inevitable questions:
 - What are the risks of stopping testing now?
 - What are the benefits of releasing now?
- Only if we record the risks to be addressed when tests are planned can we say...
 - Which risks have been covered
 - Which risks have NOT been covered
 - Which benefits could be delivered
- The risks that are covered aren't the issue
- We need to understand the benefits to be delivered and the residual risks.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 22

When is the Product "Good Enough"?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 23

"Is the product good enough?"

- Testers are never comfortable with the amount of testing
- "Too little planned, too little completed, the product is imperfect"
- Should testers make recommendations?
 - "It's not ready" - dismissed as pessimists
 - "It looks OK" - they've taken someone else's responsibility.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 24

Compulsive behaviour

- Consultants, 'gurus', academics preach perfection through compulsive behaviour:
 - » product perfection through process maturity and continuous process improvement
 - » all bugs are bad, all bugs could be found, so use more and more rigorous/expensive techniques
 - » documentation is always worthwhile
 - » you can't manage what you can't count
 - » etc. etc..
- Process hypochondriacs can't resist.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 25

"Good Enough"

- James Bach* is main advocate of the 'Good Enough' view (also Yourdon and others)
- A reaction to compulsive formalism
 - if you aim at perfection, you'll never succeed
 - your users/customers/businesses live in the real world, why don't you?
 - compromise is inevitable, don't kid yourself
 - guilt and fear should not be part of the process.

* James Bach, "Good Enough Quality: Beyond the Buzzword", Computer, August 1997
Web site: www.satisfice.com

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 26

"Good Enough" means:

1. **X** has sufficient benefits
2. **X** has no critical problems
3. Benefits of **X** sufficiently outweigh problems
4. *In the present situation, and all things considered, improving X would cause more harm than good.*

All the above must apply

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 27

Contribution of testing to the release decision

- Have sufficient benefits been delivered?
 - Tests must *at least* demonstrate that features providing benefits are delivered completely
- Are there any critical problems?
 - Test records must show that any critical problems have been corrected, re-tested, regression tested
- Is our testing Good Enough?
 - Have we provided sufficient evidence to be confident in our assessment?

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 28

"Is the product good enough?" (2)

- "Good enough" is in the eye of the stakeholder
- Testers can only say:
 - "at the current time, our tests demonstrate that:
 - » the following features/benefits have been delivered
 - » the following risks have been addressed
 - » these are the outstanding risks of release..."
- Stakeholders and management, not the tester, can then make the decision.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 29

Risk and Benefits-Based Test Reporting

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 30

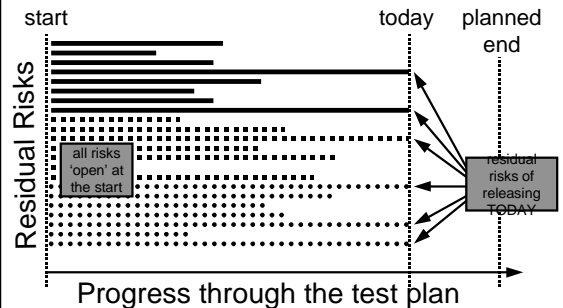
Proposed test objectives

- To provide evidence (and confidence) that
 - The benefits delivered are acceptably high
 - The risk of failure is acceptably low
- To report progress by identifying:
 - Risks addressed
 - Benefits delivered
- Stakeholders want to know about benefits delivered (and the benefits under threat)
- Project management want to know about the residual risks that block the benefits.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 31

Risk-based reporting



Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 32

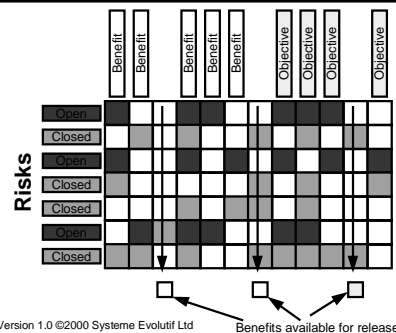
Benefits of risk-based test reporting

- Risk of release is known:
 - On the day you start and throughout the test phase
 - On the day before testing is squeezed
- Progress through the test plan brings positive results – risks are checked off, benefits available
- Pressure: to eliminate risks and for testers to provide evidence that risks are gone
- We assume the system does not work until we have evidence – “guilty until proven innocent”
- Reporting is in the language that management and stakeholders understand.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 33

Benefit (or objective) based test reporting



Version 1.0 ©2000 Systeme Evolutif Ltd

Benefits available for release

Slide 34

Benefits of benefit-based test reporting

- Risk(s) that block every benefit are known:
 - On the day you start and throughout the test phase
 - Before testing is squeezed
- Progress through the test plan brings positive results – benefits are delivered
- Pressure: to eliminate risks and for testers to provide evidence that benefits are delivered
- We assume that the system has no benefits to deliver until we have evidence
- Reporting is in the language that management and stakeholders understand.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 35

How good is our testing?

- Our testing is good if it provides:
 - Evidence of the benefits delivered
 - Evidence of the CURRENT risk of release
 - At an acceptable cost
 - In an acceptable timeframe
- Good testing is:
 - Knowing the status of benefits with confidence
 - Knowing the risk of release with confidence.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 36

Close

- If we use the language of risk, testers will be heard
- Test budgets can be based on a consensus view of product risks
- Good testing provides evidence of risks and benefits throughout the project
- Good testing provides enough evidence to allow stakeholders to make a balanced judgement
- Testers are often the best informed people on a project
- Let's communicate with a language that works.

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 37

Risk: The New Language of E-Business Testing

Thank-You!

www.evolutif.co.uk

Version 1.0 ©2000 Systeme Evolutif Ltd

Slide 38

Wednesday 6 December 2000

Keynote 1

Risk: The New language of e-Business Testing

Paul Gerrard

Paul is the Technical Director and a principal consultant for Systeme Evolutif. He has conducted consultancy and training assignments in all aspects of Software Testing and Quality Assurance. Previously, he has worked as a developer, designer, project manager and consultant for small and large developments using 3 and 4GLs. Paul has engineering degrees from the Universities of Oxford and London, is Special Projects Secretary for the BCS SIG in Software Testing, a member of the BCS Software Component Test Standard Committee and Former Chair of the IS Examination Board (ISEB) Certification Board for a Tester Qualification whose aim is to establish a certification scheme for testing professionals and training organisations. He is a regular speaker at seminars and conferences in Europe and the US, and won the 'Best Presentation' award at EuroSTAR '95. He is also webmaster for the www.evolutif.co.uk web site.

