



Testing in Agile Environments

In practice

Erik Boelen

QA Consult Services

Erik.boelen@qaconsult.eu

www.qaconsult.eu

What will I talk about?

- | Agile Testing for me
- | Am I an Agile Tester?
- | How to put in practice?

Nowadays...

I am an Agile Tester

We don't use
documentation!

We work very
chaotic!

TRENDY

**We scrum at least once a
day!**

Terms and Definitions

Intentions are good

The truth is not in the words

Positive influence on the end result

**Agile Testing is not about using
terms and definitions**

What is Agile for me?

- | Providing a continuous stream of value to your customers
 - Increase the rate of feedback
 - Reduce the waste
- | Not about
 - Compressing the schedule
 - Tossing out the documentation
 - Coding up to the last minute

It is all about maximum value for minimal cost

What is Agile to the world?

Agile Manifesto on Software Development

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

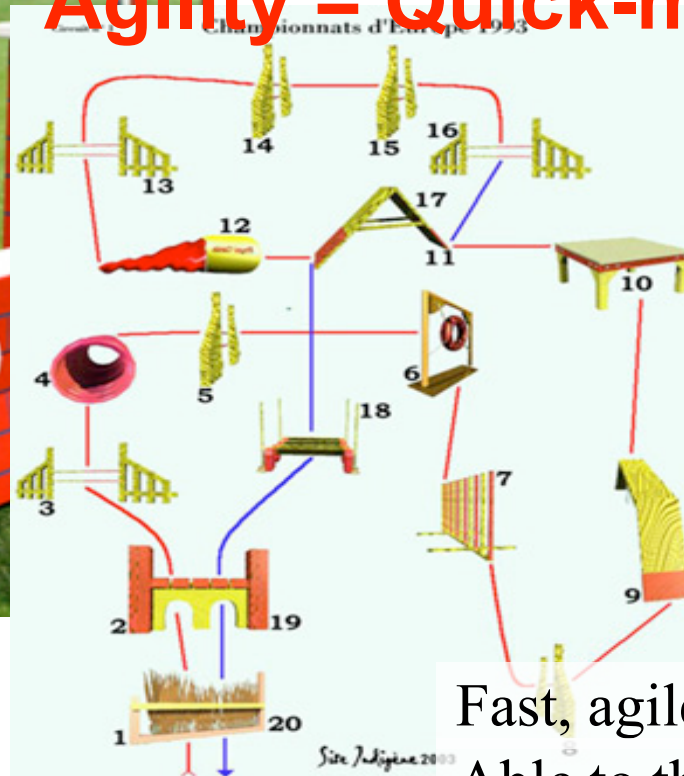
A Dog's Life

How to explain Agile?

Agility = Quick-moving & Nimble



AGILITY



Fast, agile and light in movement
Able to think quickly and cleverly

Who is an Agile Tester?

How to determine if you're an Agile Tester?

How did I set up the requirements?

Experience

Opinion of peer software testers

Experts on the topic

I love my customer

The Customer is responsible for ensuring the correct application gets built by:

- Providing the vision that drives the product
- Steering the project through numerous specific decisions about the sequence of work based on an understanding of the overall business benefit relative to the cost of each piece of work. This is an ongoing process that accounts for changes to the business over time
- Describing the requirements clearly and succinctly, at just the right time, and in just the right level of detail

The customer role is also responsible for assessing that the application was built right by:

- Defining acceptance criteria / tests that drive product development
- Testing the delivered product

I love my customer

It helps to be connected to:

- the test object
- the environment
- The solution

If not, you can become uninspired

Team work can be influenced by the weakest link

I work in a very structured way

Agile =

- Chaotic
- Hectic
- Ever changing

From an outsiders point of view

Structure is required

Absolutely necessary

- Very tight version control
- Detailed planning

I adore unconventional test tools

‘Boost your testing super powers’ – James Bach

- Turn your transistor radio into a test tool

Agile environment allows using unconventional tools

... if agreed upon

Lots of free tools available to assist testing activities

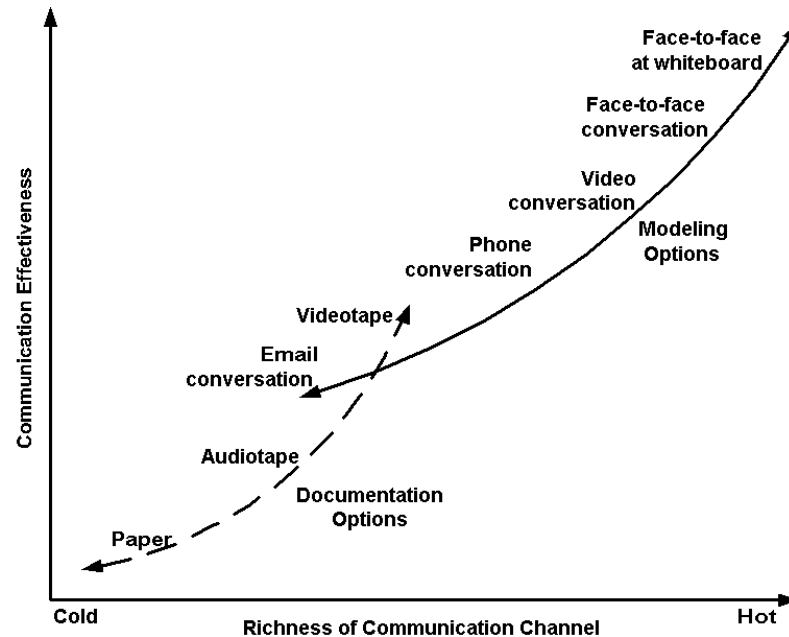
- CSV Editors for data uploading
- Mind Mapping tools
- Screen Cams
- Monitoring tools
- Performance testing tools
- Test Management tools

Customized tools using Python or Ruby

I communicate from an open position

I communicate ...

- Communication is crucial
- Providing solutions over writing documents describing the solution



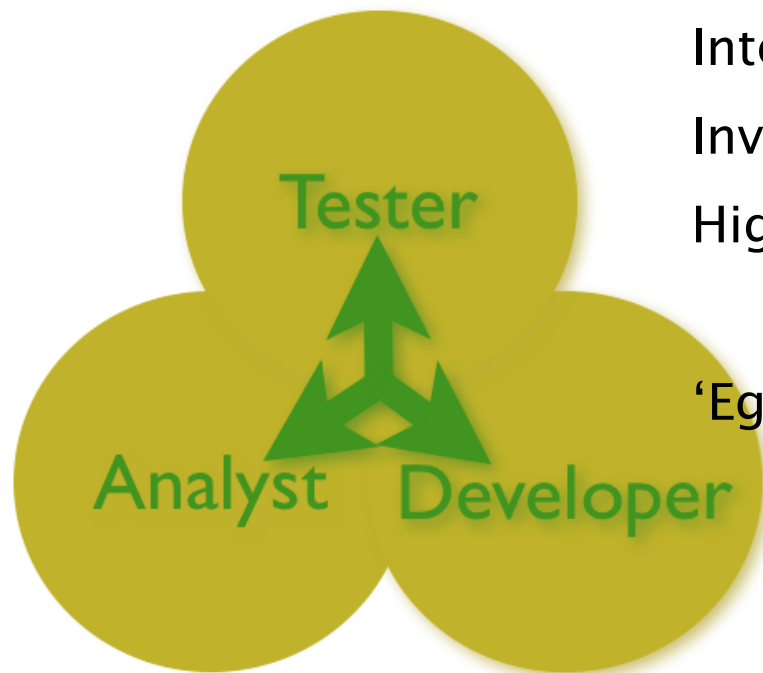
Original Diagram Copyright 2002 Alistair Cockburn, Modified Version Scott Ambler 2002

Forms of communication
in agile environments

Alistair Cockburn

I communicate from an open position

... from an open position



Interaction between different roles

Involvement in other areas of expertise

High level of honesty

‘Egoless programming’

I am a VIP in my project

Just like the others

Individuals and interactions over processes and tools

- ‘I felt appreciated as a tester! Wow! That was new to me!’

Elisabeth Hendrickson on Agile Testing

Each individual is part of the team

- Every contribution is equally appreciated
- Similar reward and recognition

But, on an individual level

Independent of role played in the project

I don't spend time covering my ass

Reasons for setting up piles of documentation

- To pass on information
- Get agreements on the content
- Build a database of permanent documentation

Ass Coverage

Collaboration overrules points of control

Examples of useful documents

- Features to Test
 - Test Strategy
 - Test Planning
 - Main risks to be mitigated
- Problem reporting
 - Detailed description
 - Including test steps executed

I know my boundaries

Management issue in agile environments

- Stay away from the border of unstructured and uncontrolled project management

Content boundaries

- Many releases of small parts of the solution
 - Increments or iterations
- Content of test activities is also limited to this increment or iteration
- Crossing that border causes overhead to others

I know my boundaries

Timing boundaries

- Time Boxing
- Detailed planning
- Close follow up of time requirements
- Work according to schedule
 - Schedule may be adapted through group decision

I am a tester

My favorite item

Agile tester is tester adapted to agile environments

Comes naturally to testers

Exploratory testing

Put in practice

Work iteratively in a
non-iterative
environment

Apply exploratory testing
in a test preparation
phase

Combine these two ...



... and gain quality

Background Info

The Assignment

Governmental environment
Throughput time of 4 months
11 technicians

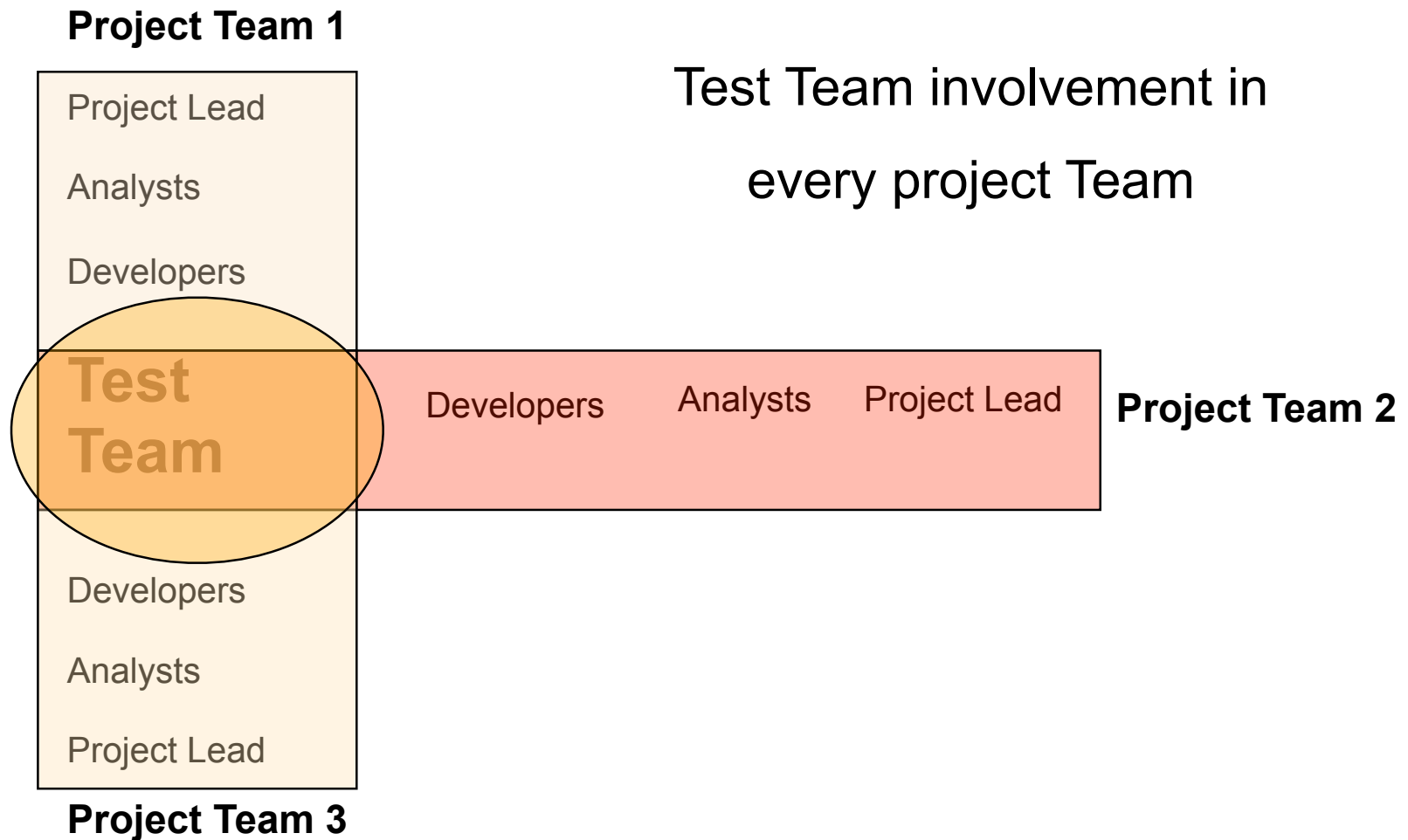
The Goal

High quality system
All user specifications met

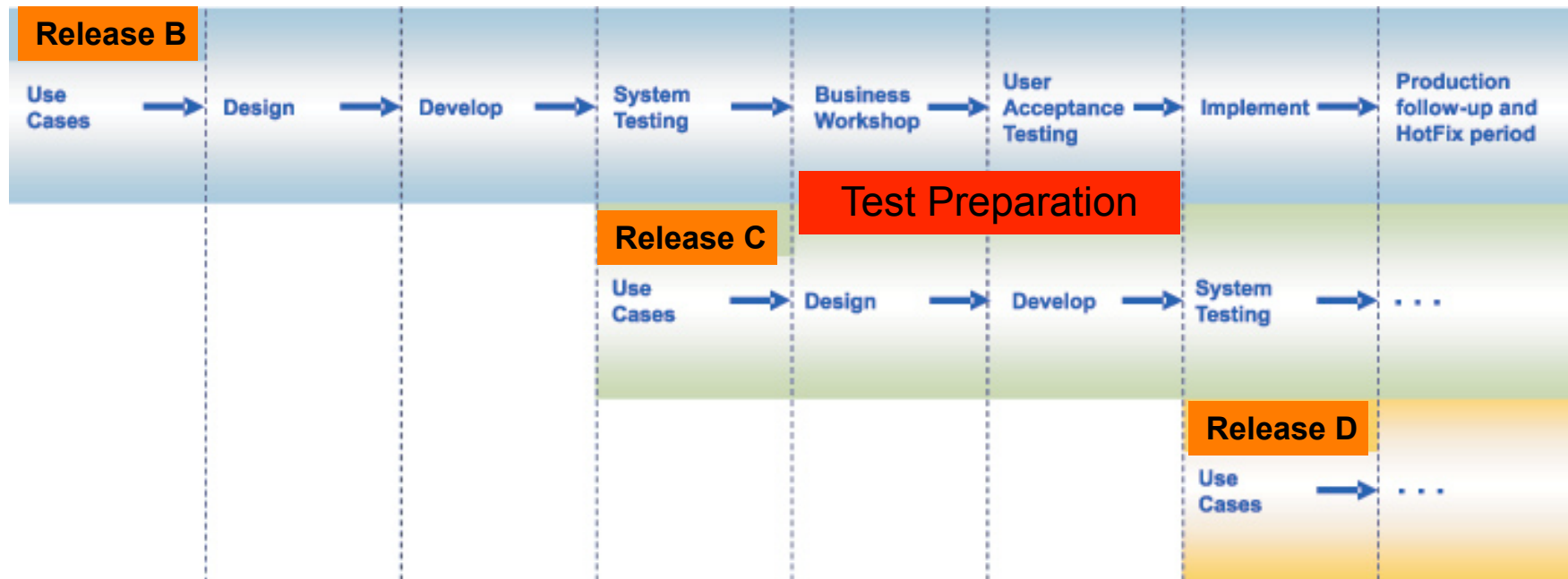
The Test Team

Small, only three persons
Involved in every release for different project teams

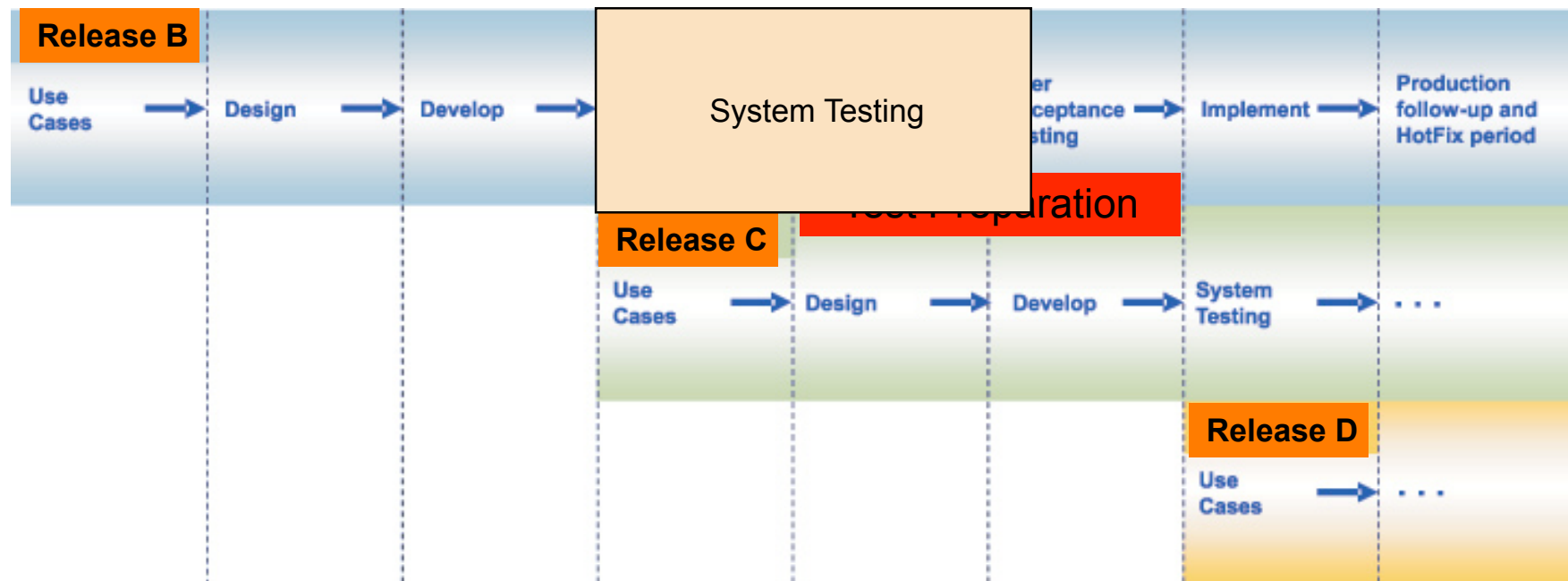
Background Info



The Process



The Process – Where it went wrong

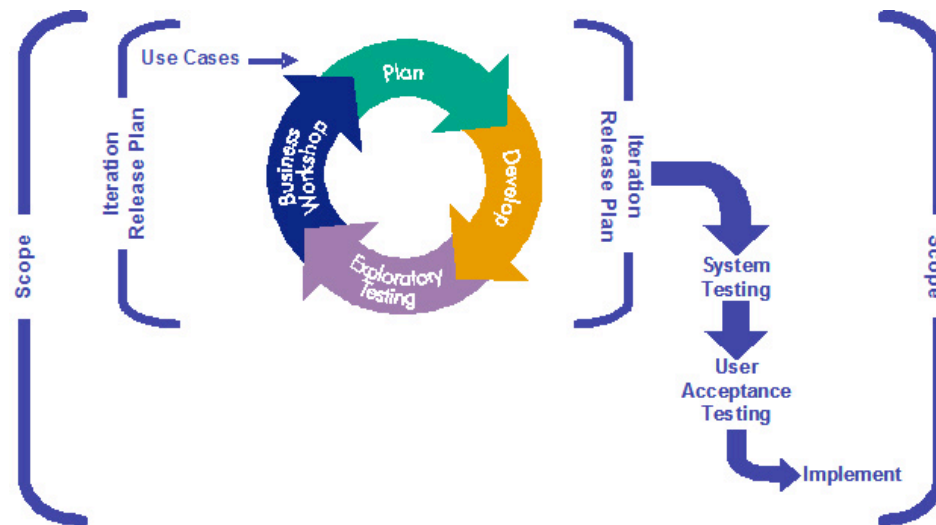


System Test phase too long

Test Team Involvement in every release

No Preparation time for Release C

The Solution

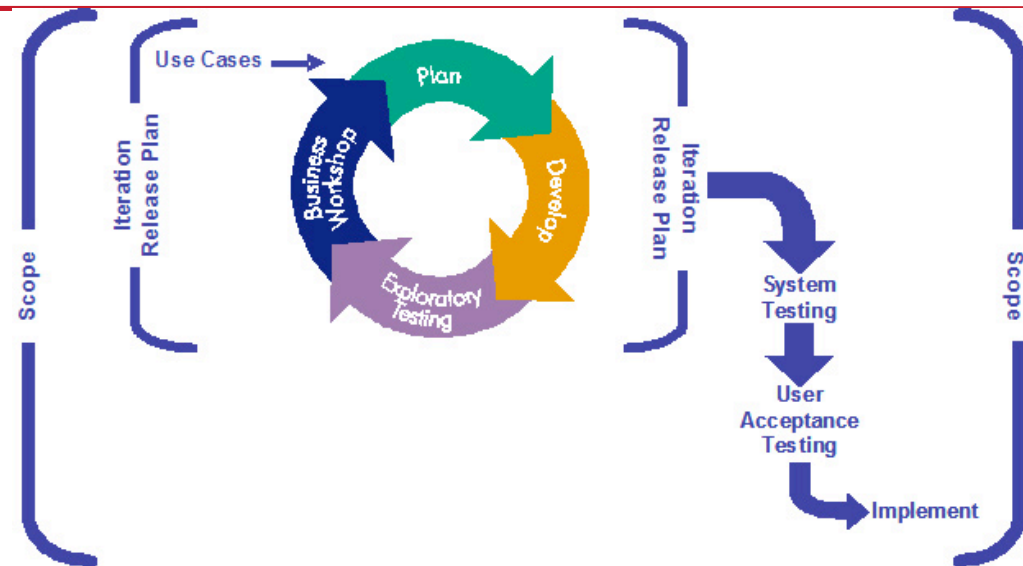


- | Young team of developers
- | “Iterations”, “Agile”, “XP” were buzz-words
- | Project Lead had experience in Iterative processes

- | Towards management, we “played our testing card”
 - What if we don’t change our way of working?
 - No test preparation
 - Less efficiency during test execution
 - More issues in Go Live system
 - Less thrust from Business users and sponsors

The Solution

- | Iterative
 - | Plan
 - | Development
 - | Exploratory Testing
 - | Business Workshops

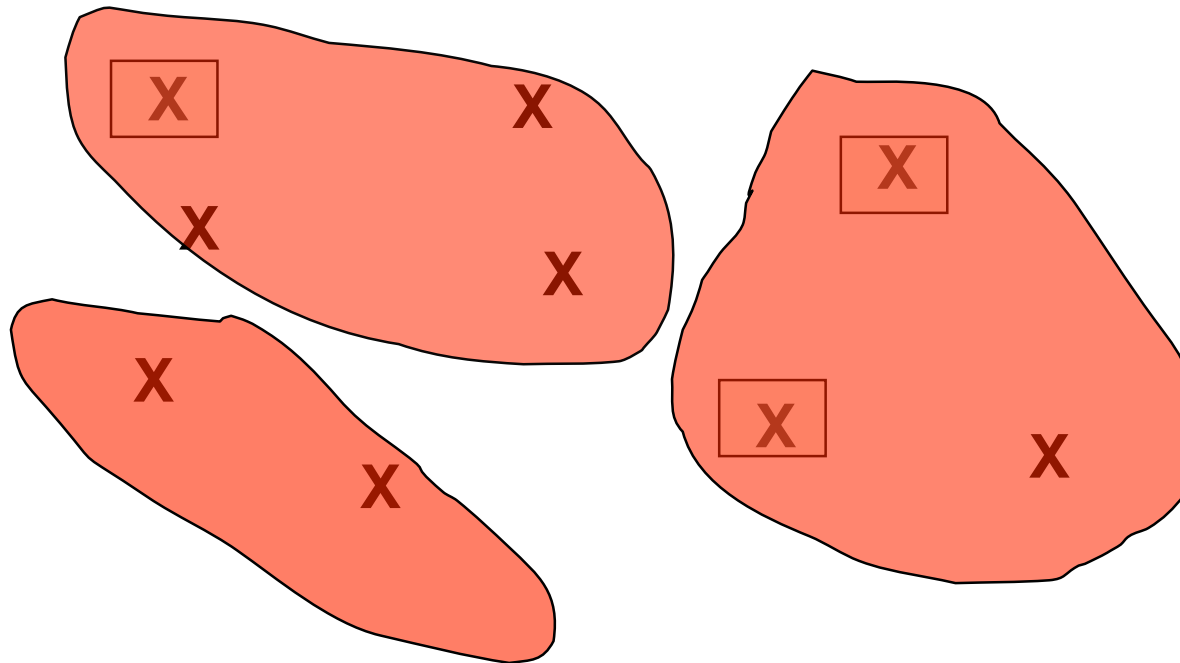


- | Non-iterative
 - | Scope Definition
 - | Use Cases
 - | System & Integration Testing
 - | Acceptance Testing
 - | Implementation

The Solution

Determine the iteration

P

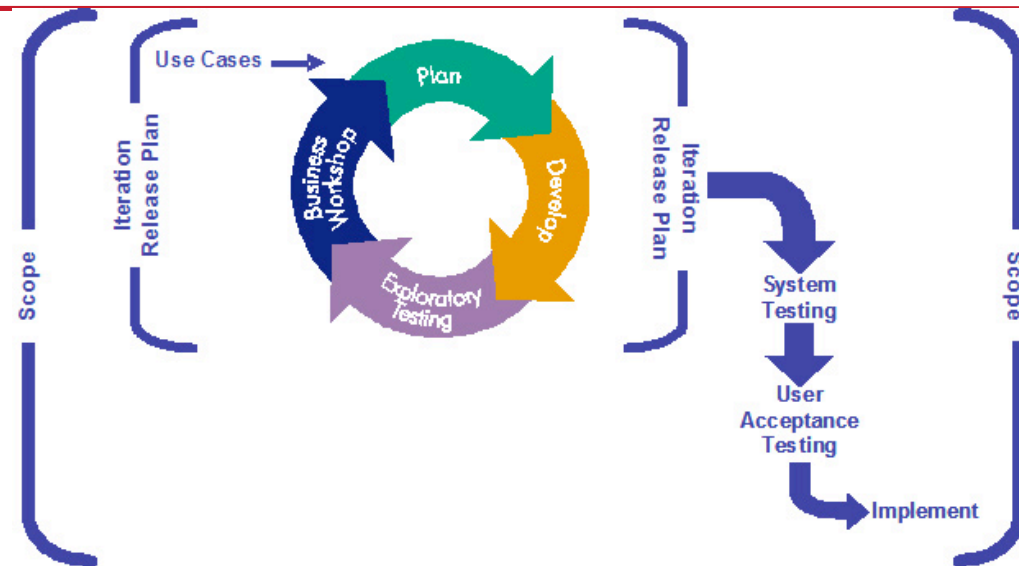


Developers choice

Testers choice

The Solution

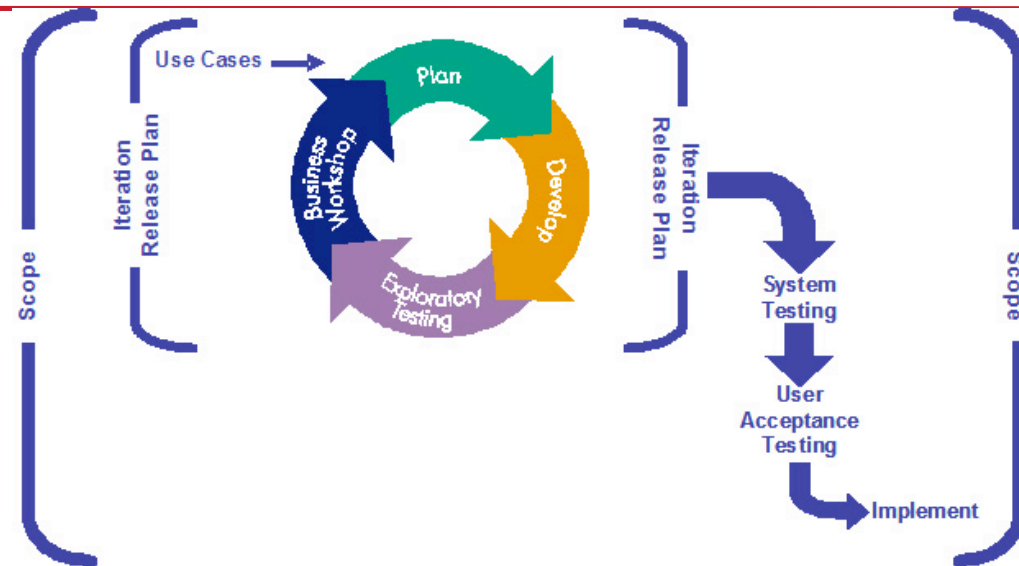
Develop + Exploratory Testing



- | Explore the part of the application already developed
- | Set up test cases based on
 - | documentation
 - | the prototype
 - | the section of the application already built
- | Register any issues found during the exploration phase

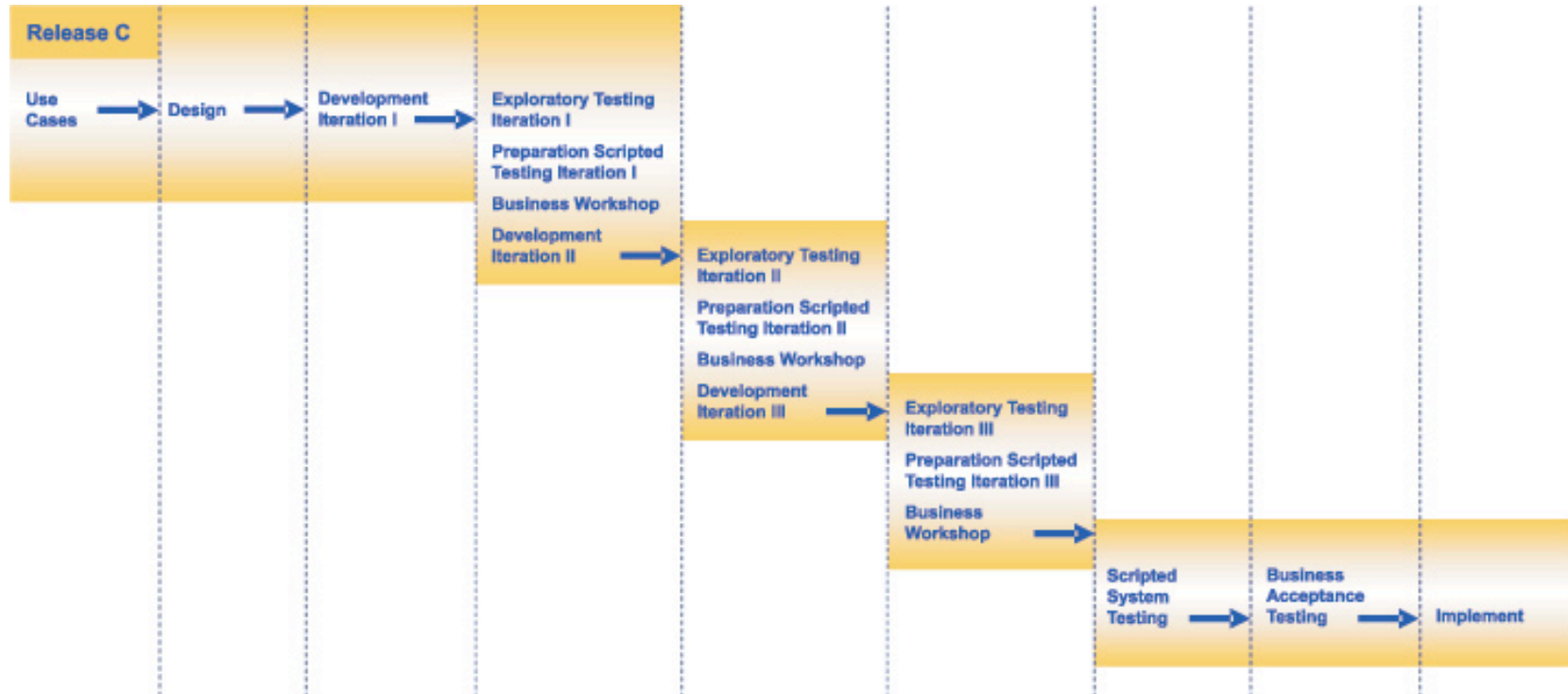
The Solution

Business Workshop



- | Guided walkthrough of the part already developed
- | Involvement of Business Users and sponsors in early stage
- | Business Users got acquainted with the application
- | UAT test scenarios could be set up during development

The Solution – Work Breakdown



Results

- | Overall results
 - Better application knowledge
 - Positive enhancements in team cooperation
 - Detect major issues before the actual testing period
 - we improved quality and gained time
- | The exploratory testing iterations forced us to
 - explore the application in greater detail
 - ask more questions of analysts
 - communicate more extensively with business users
 - our application knowledge increased significantly
 - enhanced application knowledge will be used and further improved in future releases by the test engineers.

Results cont'd

| **System testing**

- Planned for five weeks
- Exploratory testing iterations
- Actual throughput time was two weeks

| **User acceptance testing**

- Planned for three weeks
- Users were well-informed about the application thanks to the business workshops
- Users had already developed ideas about what needed to be tested during this phase
- We gained thrust of Business Users and sponsors
- Users end-to-end test cases covered the entire application
- Actual throughput time was two weeks

Conclusion

- | Approach can be applied in different situations
- | Involvement of all different parties is crucial
- | Use this approach to increase team cooperation

Take your team in an iterative direction

BUT

- | Make sure that you're up for the job
- | Not everybody is an agile tester
- | It's definitively worthwhile to try