



IBM Software

UK Innovate 2010

The Rational Software Conference

Smarter software for a smarter planet.



IBM Software

UK Innovate2010

The Rational Software Conference

Using DOORS for Iterative Development

From the Perspective of the
2009 DOORS Web Access 1.3 Release

Martin Henderson

Smarter software for a smarter planet.



Agenda

Introduction

Setting the Scene

IBM's Commitment to Quality

IBM's Integrated Product Development Process

Functional and Quality Objectives

Summary of our Approach

Artefacts and Relationships

User Stories

What Happened During A Typical Iteration

Results



Introduction

IBM Rational DOORS is the market leading Requirements Management solution for Systems and complex IT applications.

Multiple award winner for Best Product for Requirements Driven Development (Yphise).

ISO 9001 compliant development.

It helps you to reduce costs, increase efficiency and improve quality by enabling you to optimize requirements communication, collaboration and by promoting compliance and verification.

Comprehensive access controls, easy-to-use document oriented views, full lifecycle traceability, baselines, history, FDA compliant electronic signature, etc.

It adapts easily to address changing requirements and management needs - throughout your organization and across your supply chain.

DOORS Web Access helps promote collaboration and requirements use in your development process by providing access to view, edit and discuss requirements via a zero-footprint web application.

Fast, easy and scalable deployment— it expands visibility of requirements and relationships to all (distributed) stakeholders.



Setting the Scene

DOORS and DOORS Web Access were being developed using a traditional Waterfall model.

Typical releases were between 9-12 months in duration, of which 3 months was a dedicated “back-end” test phase.

All requirements were agreed up front with Product Management with the intention that development would not start until functional content was reviewed and signed off by all relevant stakeholders.

However the act of review and signature was a long, drawn out and often frustrating process - and inevitably development was pushed further and further into the lifecycle.

Opportunities to test and remediate work prior to functional freeze became limited and change was very difficult to manage.

There was a recognized need for a rethink in order to secure the ongoing success of a highly strategic and visible product.



IBM's Commitment to Quality

IBM has an overriding worldwide commitment to the quality of the products, solutions and services we provide to our customers. Quality is recognized as a fundamental component of the value customers receive from IBM.

IBM is committed to the goals of achieving total customer satisfaction, delivering superior products, solutions, and services, and exceeding customer requirements.

Recognizing that the marketplace is the driving force behind everything we do, IBM implements effective business processes that support value creation for our customers and our stakeholders.

IBM leaders are responsible for establishing objectives and using measurements to drive continual improvement in quality and in customer satisfaction.

All IBM employees are expected to contribute to continual improvement as an integral part of our Quality Management System.

As a minimum each product must be superior to it's predecessor.



IBM's Integrated Product Development Process

IPD is a management system designed to optimize the development and delivery of successful products and offerings.

It consists of five phases with a defined set of deliverables and periodic checkpoints that are predicated on industry best-practices and fact-based decision making.

Helps develop offerings and then manage them from concept to end-of-life.

The cornerstone of IPD is team-based management involving the representation and active participation of all relevant functions.

Introduced in 1996 during a period of significant cultural change.

Significant business results:

- Reduced time-to-market.
- Lower development expense to revenue ratio.
- Positive effect on corporate profits.
- Higher percentage of winning offerings.
- Significant expense and cost saving from parts reduction and re-use.
- Dramatic reduction in expenses due to abandoned effort.



IBM's Integrated Product Development Process

From the team's perspective it essentially came down to 2 documents.

A Decision Checkpoint that stated the business value proposition and (more importantly) the functional objectives.

A Quality Plan that documented the improvement commitments and the intended approach / actions to achieve them.

Achievement of the Quality Plan is documented with the Quality Certification, which is ultimately approved by the product's Development Vice-President.



DOORS Web Access 1.3 Functional Objectives

Extend edit capability by allowing users the ability to create, copy, move, delete and undelete objects in existing DOORS modules.

Extend traceability by allowing users the ability to be able to create and delete DOORS Links and External Links.

Allow users the ability to perform module level "goto" operations using heading paragraph numbers.

Add change bar indicators to current view so that users can easily determine those objects that have been altered since the previous baseline.

Improve Enterprise scalability by increasing the number of supported concurrent users.

Support for DOORS / Rational Requirements Composer integration.

Support for "click-and-accept" licensing.

Support for Internet Explorer 8.

Japanese and Simplified Chinese translations.



DOORS Web Access 1.3 Quality Objectives

Retain ISO 9001 compliance.

Lower existing defect debt.

No introduced S1 (Showstopper) or S2 (High) defects.

Overall 80% fix-rate for all introduced defects.

High levels of planned, attempted and successful FVT / SVT.

Consumability, Serviceability and CUPRIMDSAQ.

Extension of Unit Test framework for increased coverage.

Automatic execution of Unit Tests during overnight builds.

Review of client by User Experience and Visual Design teams

Publicly available evaluation server.



Summary of our Approach

High level functional and quality objectives agreed up front.

A streamlined requirements process which remains ISO 9001 compliant.

4 week iterations.

User Stories.

Daily Scrums.

Separate Development and Test teams with a focus on parity and collaboration.

Self-hosting of DOORS Web Access against main requirements repository.

Regular show-and-tell presentations of new functionality to Product Management, Support, Test, etc.

Formal iteration handovers / walkthroughs between Development and Test.

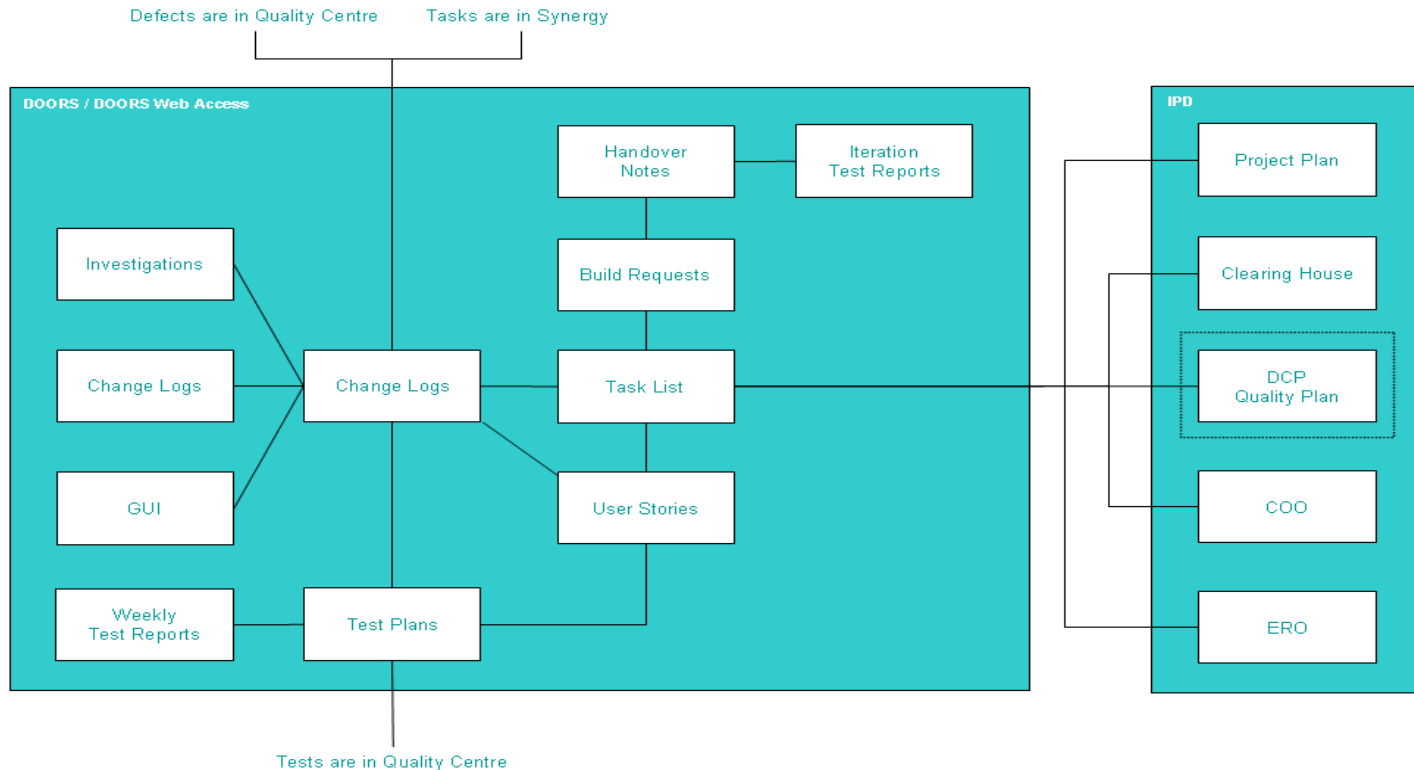
Allowances made in project plan for dedicated effort towards improving unit tests.

Overnight builds.

Automated testing.



Artefacts and Relationships



User Stories

A user story is a system requirement formulated as one or more sentences in the everyday or business language of the user.

The user stories should be written by the customers for a software project and are their main instrument to influence the development of the software.

Each user story is limited to ensure that it does not grow too large.

User stories are a quick way of handling customer requirements without having to elaborate vast formalised requirement documents and without performing overloaded administrative tasks related to maintaining them.

The intention with the user story is to be able to respond faster and with less overhead to rapidly changing real-world requirements.

Please follow this [link](#) for a good article by Mike Cohn that discusses differences between user stories, use cases and traditional requirements.



What Happened

Development stated objectives at the start of each iteration.

User Stories were used to represent those objectives.

New tests were created to reflect the User Stories.

Designs and GUI mock-ups created and reviewed as necessary.

Focus was on increased collaboration (i.e. scrums and show-and-tells).

Important to keep Task List and other collateral accurate and up to date.

At the end of each iteration Development present a walk-through of the deliverables.

At the end of each iteration Test present a walk-through of how those deliverables will be verified.

Entry / Exit criteria agreed and signed off.

Test then perform the actual testing.

Development move onto planning the next Iteration (introduced defects triaged and included as necessary).



Results

All stated functional and quality objectives met.

Regular demonstrable versions of the system available to all stakeholders.

New unit test frameworks created and existing unit tests strengthened. All automated.

A significantly lower introduced defect count than previous functionally comparable releases.

A ~76% fix ratio for all introduced defects.

A ~35% reduction in existing defect backlog.

100% of all Severity 1 and Severity 2 defects addressed.

15 high priority customer defects addressed.

A ~200% increase in enterprise scalability via formal support for concurrent users.

100% FVT attempted and ~97% completed.

100% SVT attempted and ~99% completed.



Questions



Thank
YOU



Appendix A - Consumability

Consumability is a customer-centric term that describes our customers' experience with IBM products.

Consumable products are products that have the right design, installation, configuration, fundamental quality, information, integration, maintenance, and upgrade capabilities.

Highly consumable products enable customers to achieve fast time to value, get their solutions up and running quickly on our software, and deal simply and effectively with product maintenance throughout the product lifecycle.

Assessed via surveys (both internal and customer based) that help development teams appraise how effectively a product can be used by IBM clients in their business and IT environments.



Appendix B - Serviceability

This is a measure of the amount of effort it takes to identify the root cause of a software problem and resolve it.

The measure includes the end to end experience from the time a customer perceives they have a problem with our products, solutions or services to the time they believe it has been successfully resolved for them.

IBM strives to provide our clients with products, solutions, services and support which enable them to resolve their problems quickly with minimum effect on their business.

Sometimes perceived as part of Consumability. *Consumability = Ease of Use + Ease of Corrective Action.*

There are two perspectives to Serviceability:

- External Diagnostic Information for Customers.
- Internal Diagnostic Information for L2 / L3 Support.



Appendix C - CUPRIMDSAO

Capability	Having the necessary functions and features to perform as expected by the customer.
Usability	The ease of use of an offering, that is, the attributes that allow customers to interact effectively with that offering. This includes, but is not limited to, navigation and interaction, consistency, graphics and images, installation, and ease of upgrade.
Performance	The response time or speed with which an offering executes its functions.
Reliability	The impact of failures, malfunctions, errors, and other defect-related problems encountered by customers.
Installability	Ease of installation, initialization, and migration of an offering.
Maintainability	The ease of maintaining the offering, including applying fixes or migrating to a new version or release of the offering.
Documentation	The information provided with the offering, such as printed manuals, online help, online soft copy books, online wizards, and information on CD-ROM. Customer satisfaction with "Documentation" can be measured by the accuracy and completeness of the information, and the time it takes to find information.
Serviceability	The impact of fix management on customers. This includes, but is not limited to, problem determination, first failure data capture, and fix delivery and installation.
Availability	The percentage of time that a system is active or available for servicing requests.
Overall	Customer's overall satisfaction with the new release or product.





IBM Software

UK Innovate 2010

The Rational Software Conference

Smarter software for a smarter planet.

