Don’t Get Entangled in Your Web of Requirements

Alfredo Gutiérrez
Senior Manager, Events Program
IBM Software, Rational

fredgz@us.ibm.com

Innovate2010
The Rational Software Conference

Let’s build a smarter planet.

The premiere software and product delivery event.

Tuesday, August 24, 2010 – Singapore

Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

How well do you Write Requirements
Real World Advertisement Examples

- Dog for sale: eats anything and is fond of children.
- Dinner Special - Turkey $2.35; Chicken or Beef $2.25; Children $2.00.
- We do not tear your clothing with machinery. We do it carefully by hand.
- Wanted. Man to take care of cow that does not smoke or drink.

Requirements are Everywhere
Changing Landscape of Requirements

- **More integrated systems** rather than stand alone systems
  - More interdependence of components hence more interdependence of requirements
- **More stakeholders involved** in projects
  - More requirements sources
    - Business analysts, users, customers, marketing, regulators, architects, domain experts, legacy system experts, development, testing etc.
  - Stakeholders need to validate requirements in a form they can understand
    - Need to leverage diagrams, process flows, screen sketches, use cases etc. as well as textual requirements
    - Need to be able to incorporate these different forms into the requirements process
- **Improved technology** and tools enable us to build more complex systems
  - More complex requirements
  - But users still want solutions to be easy to use!!!

Requirements are Everywhere
Also Other Challenges for Project Teams

- **Shorter market cycle times**
  - Necessitating more project agility and more frequent requirement changes
- Drive to **reduce cost and schedule**
  - Focus on productivity and value add activities
- Teams and stakeholders are **more geographically distributed**
  - Need better communication and collaboration

Symptoms of requirements issues

Suggesting a need to invest in requirements practices

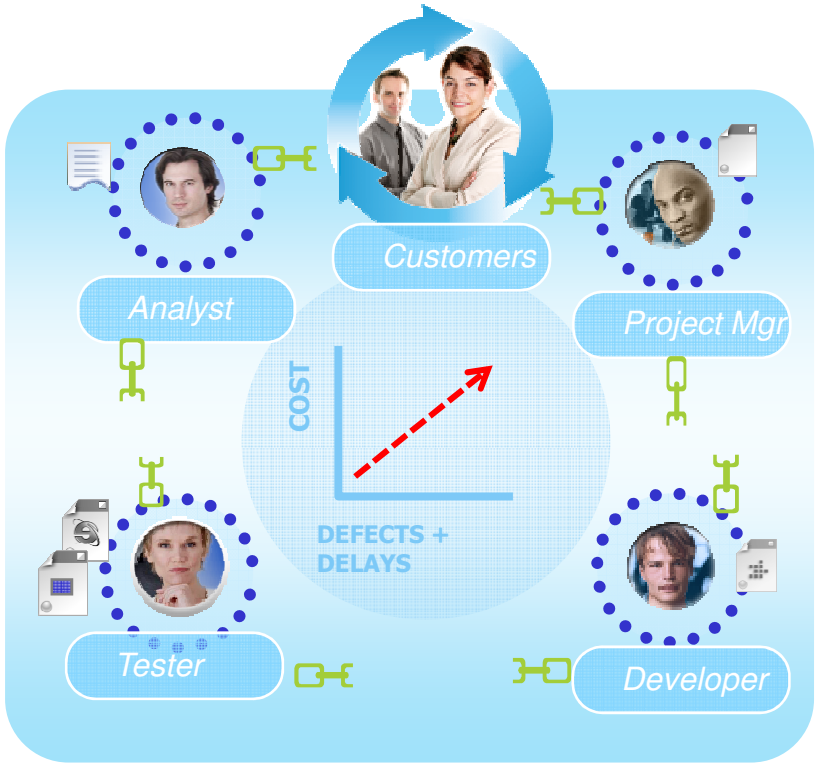
Too many project surprises, overruns and failures

Customers are not satisfied with the process or the results

Analysts, developers and testers find it difficult to work in tandem

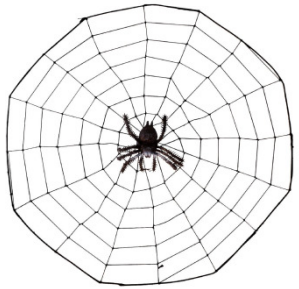
Team is bombarded by change but struggling to keep up

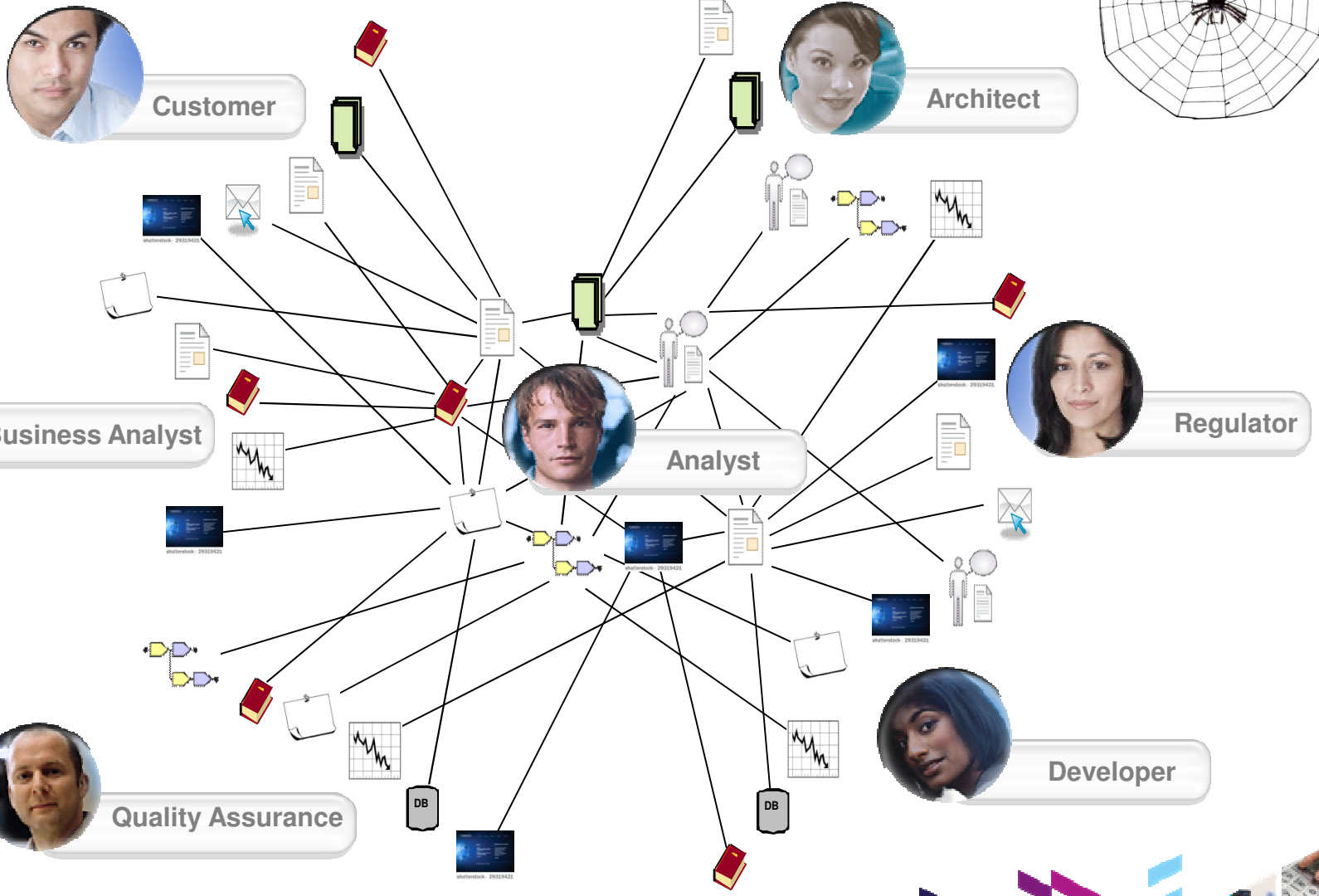
You aspire to greater agility but are unsure how to achieve it



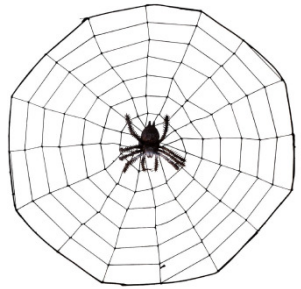
Requirements are Everywhere

Web of Requirement Interdependencies





Office Tools Exacerbate the Problem



Microsoft Word

Track Changes, Comments, Formatting nightmares, HUGE documents

Excel spreadsheets for requirement sets

Tabular format helps (at first) until the Tabs start to grow

Visio to model process, flows, screen designs

Slightest change requires many manual updates to the same component

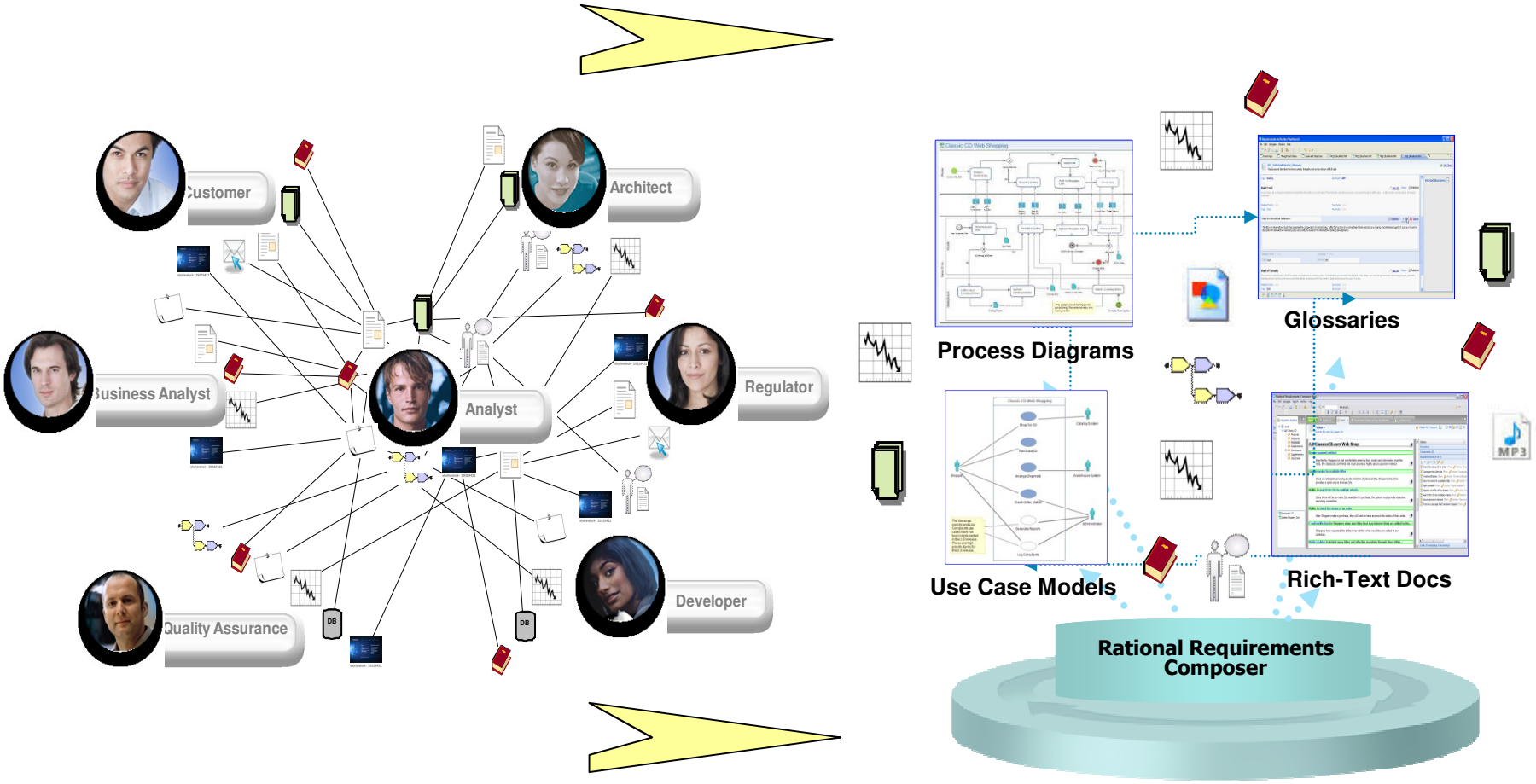
PowerPoint as communication vehicle

Dozens of presentations to crawl through for information

No way to establish relationships and link artifacts

Have to have endless meetings to put together the pieces

Tools are Adapting to Accommodate Needed Change



Challenge is – How to best leverage the tools?

Use of general tools for requirements poses challenges

Value in RDM

Difficult to address:

- ✘ Artifacts are not interrelated (are on islands)
- ✘ No team collaboration & workflow
- ✘ Ad-hoc procedures for relating and versioning artifacts
- ✘ Difficult to reuse requirements “downstream”
- ✘ Working with artifacts in well-defined groups
- ✘ Coordinated updates across related artifacts
- ✘ Generating requirements reports and documents

Office documents

Requirements can be captured in …

- Documents
- Drawings
- Spreadsheets
- Pictures

Collaborative tools

Requirements can be communicated via:

- Email
- Instant messaging
- Newsgroups
- Wikis and blogs
- Groupware
- Shared file system

Domain-specific tools help the team go to the "next level"

Value in RDM

✓ Secure repository provides central location

✓ Team collab. & workflow

✓ Progressively structure unstructured information

✓ Artifacts are interrelated and versioned

✓ Requirements are reused "downstream"

Office documents

Requirements can be captured in …

- Documents
- Drawings
- Spreadsheets
- Pictures

✓ Collections group related artifacts

✓ Baselines for scoping and comparison

✓ Views for coordinating updates across artifacts

✓ Purposeful documents and dashboards

Collaborative tools

Requirements can be communicated via:

- Email
- Instant messaging
- Newsgroups
- Wikis and blogs
- Groupware
- Shared file system

Requirements Domain-specific

Tools enabling high-productivity practices

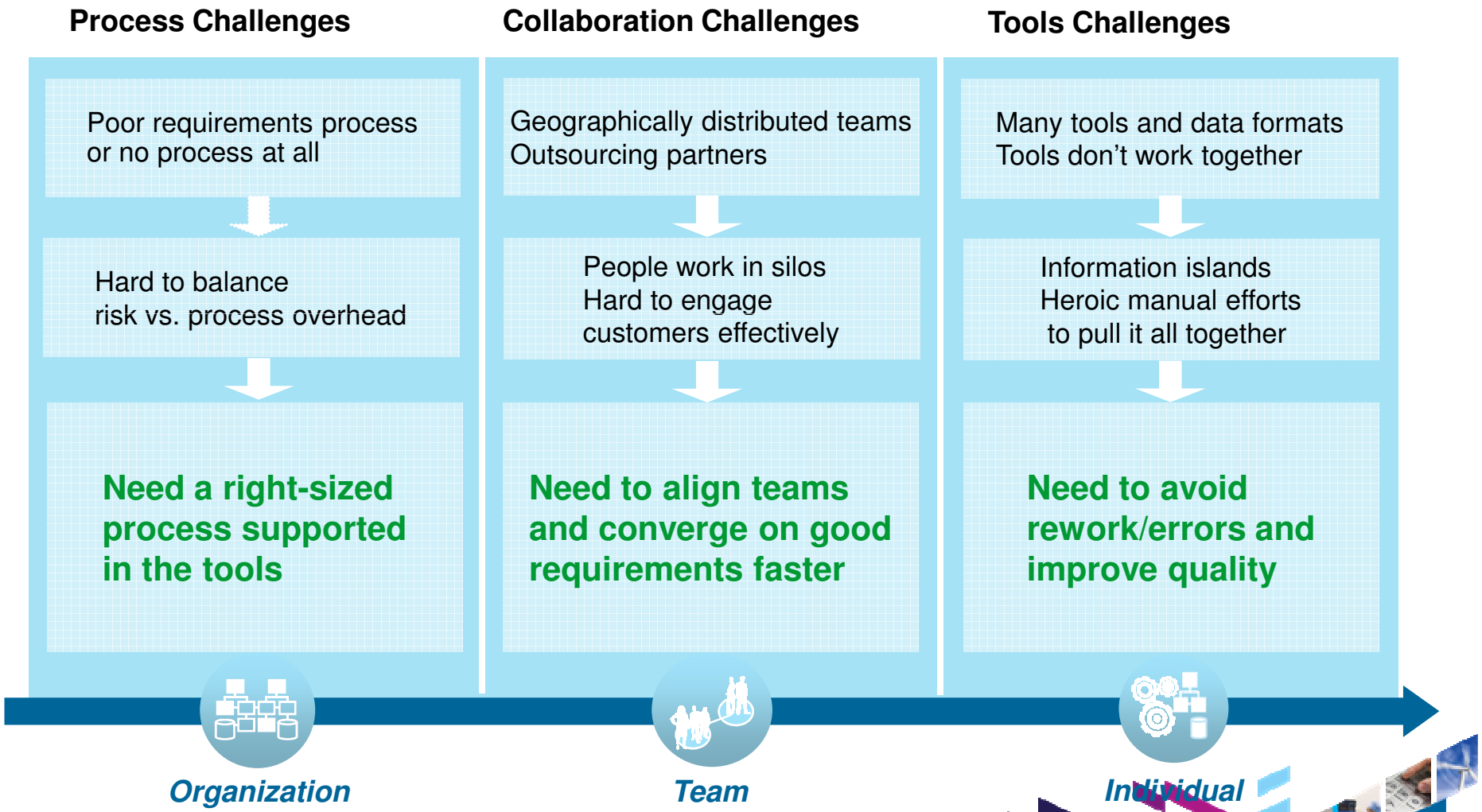
- Native support for various RD techniques
- Embrace office documents
- Collaboration in the requirements practices

Requirements across the lifecycle

- Relate requirements to project milestones and work items
- Testing verifies solution meets requirements
- Coverage and impact analysis, change mgmt

Good requirements *practices* are key to project success

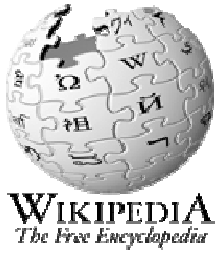
Address the big challenges



Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

What is Traceability?
General Definition



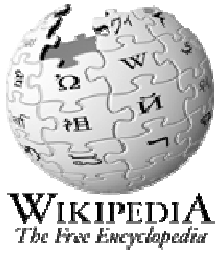
General definition of traceability is the:

> "Ability to chronologically interrelate the uniquely identifiable entities in a way that matters."

Typically in the requirements domain we are interested in structural relationships rather than temporal or chronological relationships

What is Traceability?

General Definition



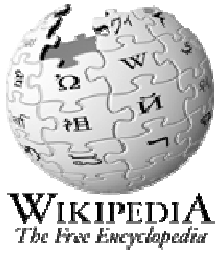
General definition of traceability is the:

"Ability to structurally interrelate the uniquely identifiable entities in a way that matters."

Typically in the requirements domain we are interested in structural relationships rather than temporal or chronological relationships

What is Traceability?

Requirements Specific Definitions



A much cited definition of requirements traceability:

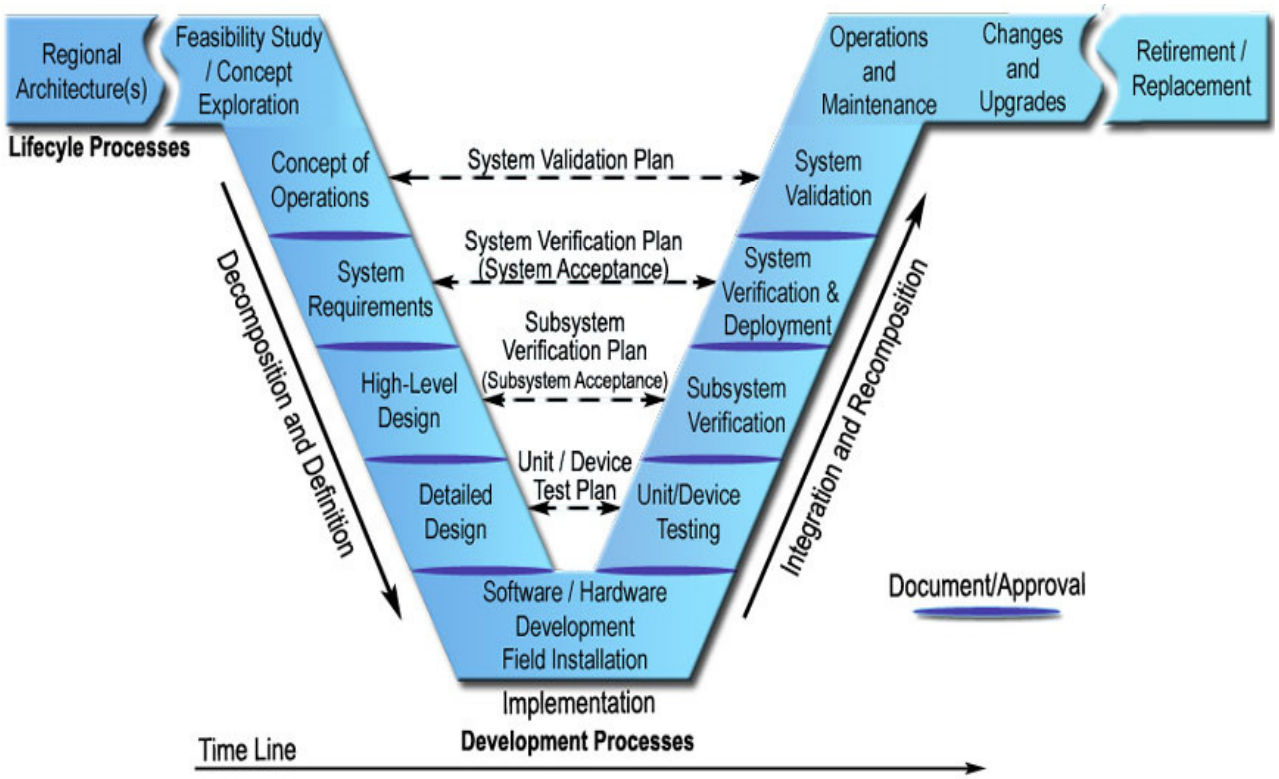
"Refers to the ability to describe and follow the life of a requirement, in both forwards and backwards direction."

Another definition of requirements traceability stressing relationships between many kinds of development artifacts:

"Refers to the ability to define, capture and follow the traces left by requirements on other elements of the software development environment and the trace left by those elements on requirements."

What is Traceability?

System Engineering "V" Model



What is Traceability?
Benefits

- **Process visibility** and auditing
  - Understand where a requirement came from, its importance, how it was implemented, and how it was tested.
- Build the **right system**
  - Verify that all stakeholder needs are implemented and adequately tested or validated.
  - Verify that there are no “extra” system behaviors that cannot be traced to a stakeholder requirement.
- Project **management and maintenance**
  - Understand requirements and project status
  - Understand the impact of changes and manage the implementation of changes
  - Keep the project team in sync

What is Traceability?
Challenges

- **Cost** of creating and maintaining traceability
  - Minimize traceability scope to achieve project goals
  - Careful with scope of manual traceability – more error prone, more time consuming and more expensive
  - Consider value based traceability (only trace high priority requirements)

- **Other considerations**
  - Instil discipline
  - Maintain integrity otherwise worthless
  - Consider long term value
  - Leverage tools

What is Traceability?
What about References

Consider traceability and referencing to be very different

- Traceability:
  - Focus on **structural relationships** – there is a definitive link between the entities i.e., derivation, dependance or impact
- References:
  - Provides **additional background information** but not a strong structural relationship
  - Helps the reader to better understand the context or to get background information

Agenda

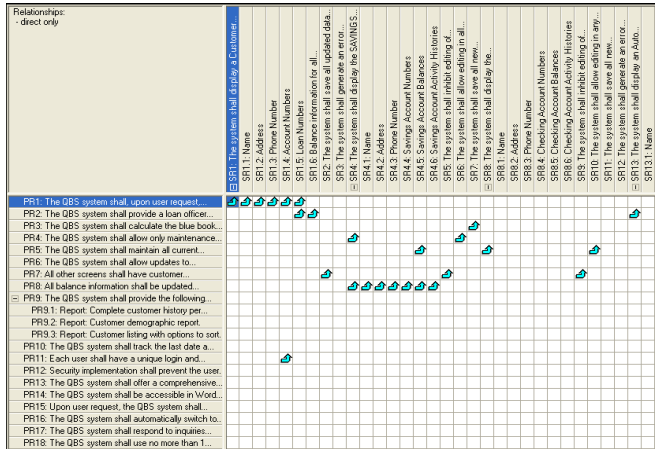
- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

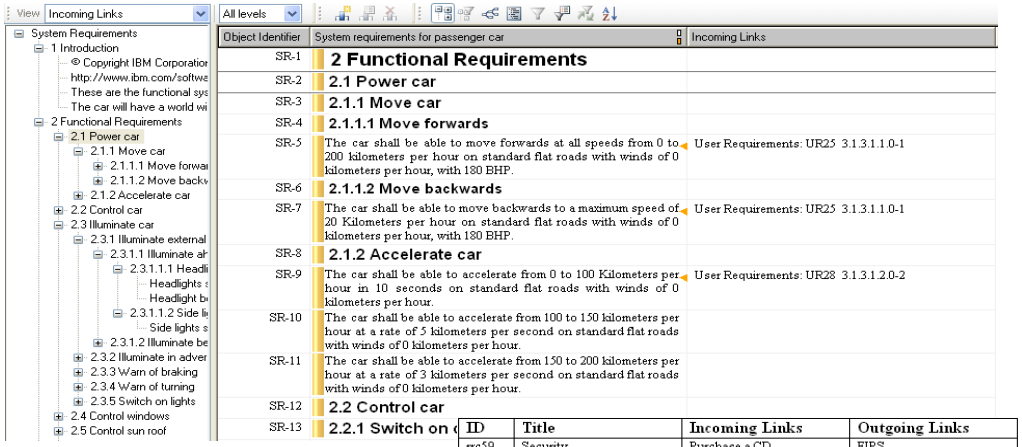
How well do you Write Requirements
Real World Advertisement Examples

- In a Los Angeles dance hall: "Good clean dancing every night but Sunday."
- In the window of a Kentucky appliance store: "Don't kill your wife. Let our washing machine do the dirty work."
- In a clothing store: "Wonderful bargains for men with 16 and 17 necks."
- Sign seen on an electricity pylon: DANGER! "To touch these wires will result in instant death. Anyone found doing so will be severely prosecuted."

How Do Tools Help?
Automation of Manual Actions

- Tools like Rational RequisitePro, Rational DOORS, Rational Publishing Engine and Rational Requirements Composer help by:
  - Eliminate manual errors
  - Aid changes or maintenance, e.g. suspect links, always latest view of information
  - Reporting on traceability relationships for completeness or missing relationships
  - Navigating quickly for information
  - Performing quick impact analyses





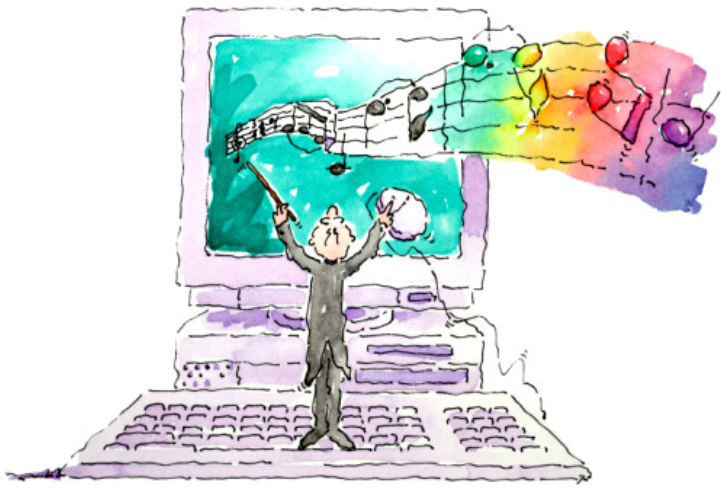
Name	Implemented By	Validated By	Type
Add a wide variety of squawkers to the system			Requirement
Add tiled squawkers as a alternative to tabbed in the GUI	23: Add tiled squawkers as a alternative to...		Requirement
Car squawker	15: Car squawker		Requirement
Cat squawker	12: Cat squawker		Requirement
Clock squawker	17: Clock squawker	Test Case 23: GUI - Clock	Requirement
Command line interface	16: Command line interface		Requirement
Dialog for selection of a single squawker	20: Dialog for selection of a single squawker	Test Case 10: GUI DefaultSquawker	Requirement
Dog squawker	11: Dog squawker	Test Case 11 - GUI Dog	Requirement
Enable declarative squawkers	25: Enable declarative squawkers		Requirement

ID	Title	Incoming Links	Outgoing Links
rrc59	Security	Purchase a CD Classic CD Supplementary Requirements	FIPS (http://www.census.gov/geo/www/fips/fips.html) Security Requirements for Cryptographic Modules
rrc62	Scalability	Classic CD Supplementary Requirements	
rrc65	Training	Classic CD Supplementary Requirements	
rrc39	View Order Details	Shop Online	
rrc44	View CD Details	Purchase a CD	Shop for CD.doc
rrc47	Shop For CD		Shop for CD.doc
rrc61	Check Order Status	Shop Online	

Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

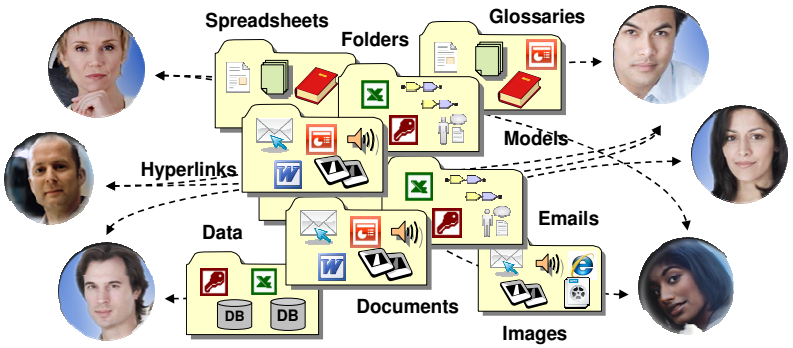
Rational Requirements Composer

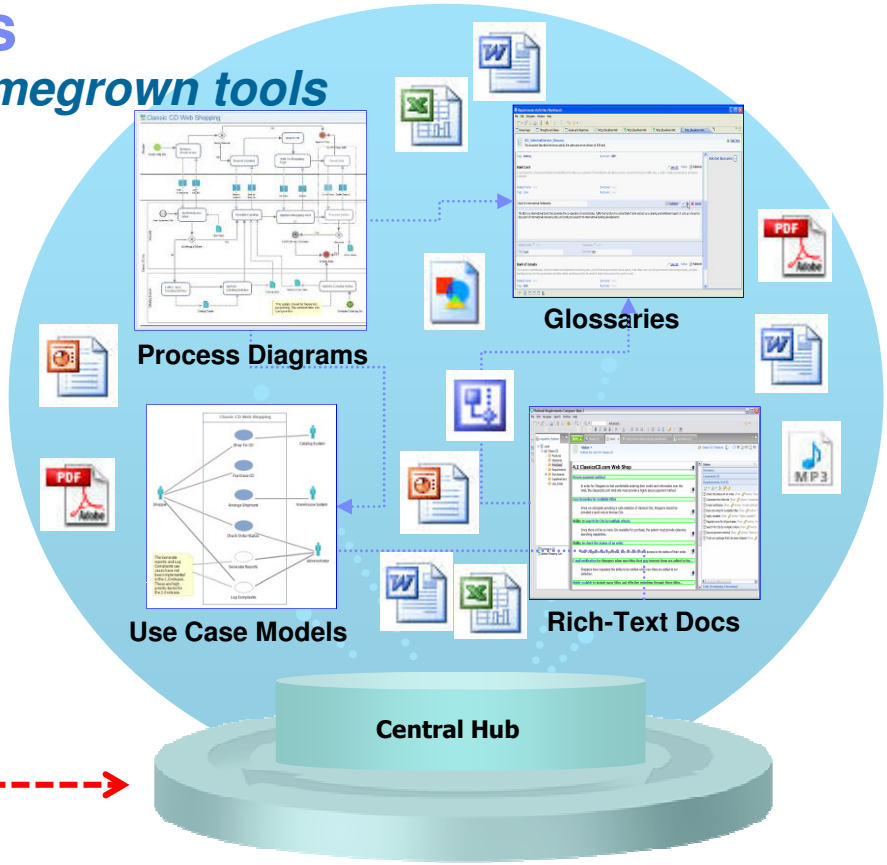


- Jazz – music to my ears!
- A bridge across the tool silos
- Effortless communication and collaboration across project teams
- All requirements artifacts can live in the same repository and be accessed via the same requirements solution
- A way to capture, connect, organize, and understand the complex web of requirements

Bridge your information islands

Embrace but move beyond Office and homegrown tools





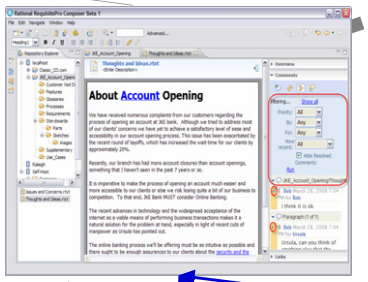
- There are **many kinds** of requirements artifacts
- Many tools, data formats and repositories create **information islands**
- A large **extended team** participates in the requirements "conversation"

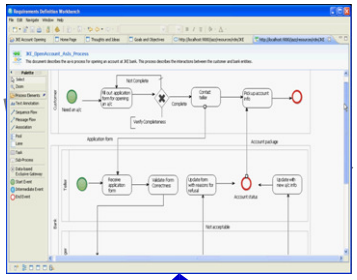
"The vast array of options linking artifacts sensibly with one another [in RRC] give it a definite benefit over using older, document-based approach to defining requirements ... This product is clearly a step up from our current methodology."
- Randy Haven, IBM Global Business Services

Rational Requirements Composer:

Capture, Connect and Make Sense of the Web Of Information

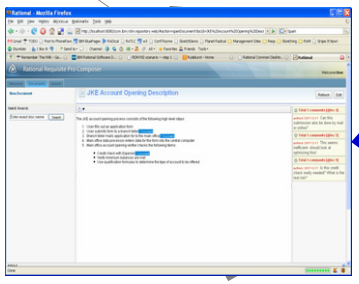
Use **Rich-text Documents** to capture structured and unstructured information (links, images, emails)

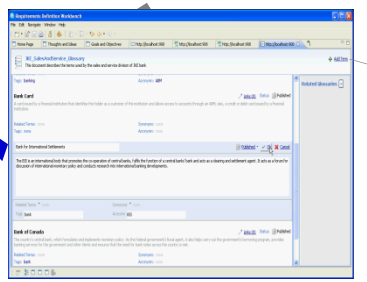




Capture the Current and Propose a Future State with **Business Process Diagrams**

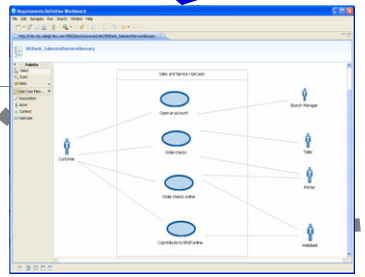
Collaborate in real-time using **Wiki-like discussions**, to quickly achieve sign-off.

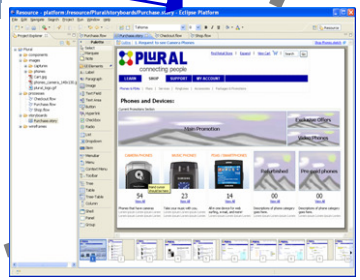




Remove Ambiguity on Business and Technology Terminology with **Shared Glossaries**

Build **Use Case Models** and Elaborate on the Processes, Actors and Activities

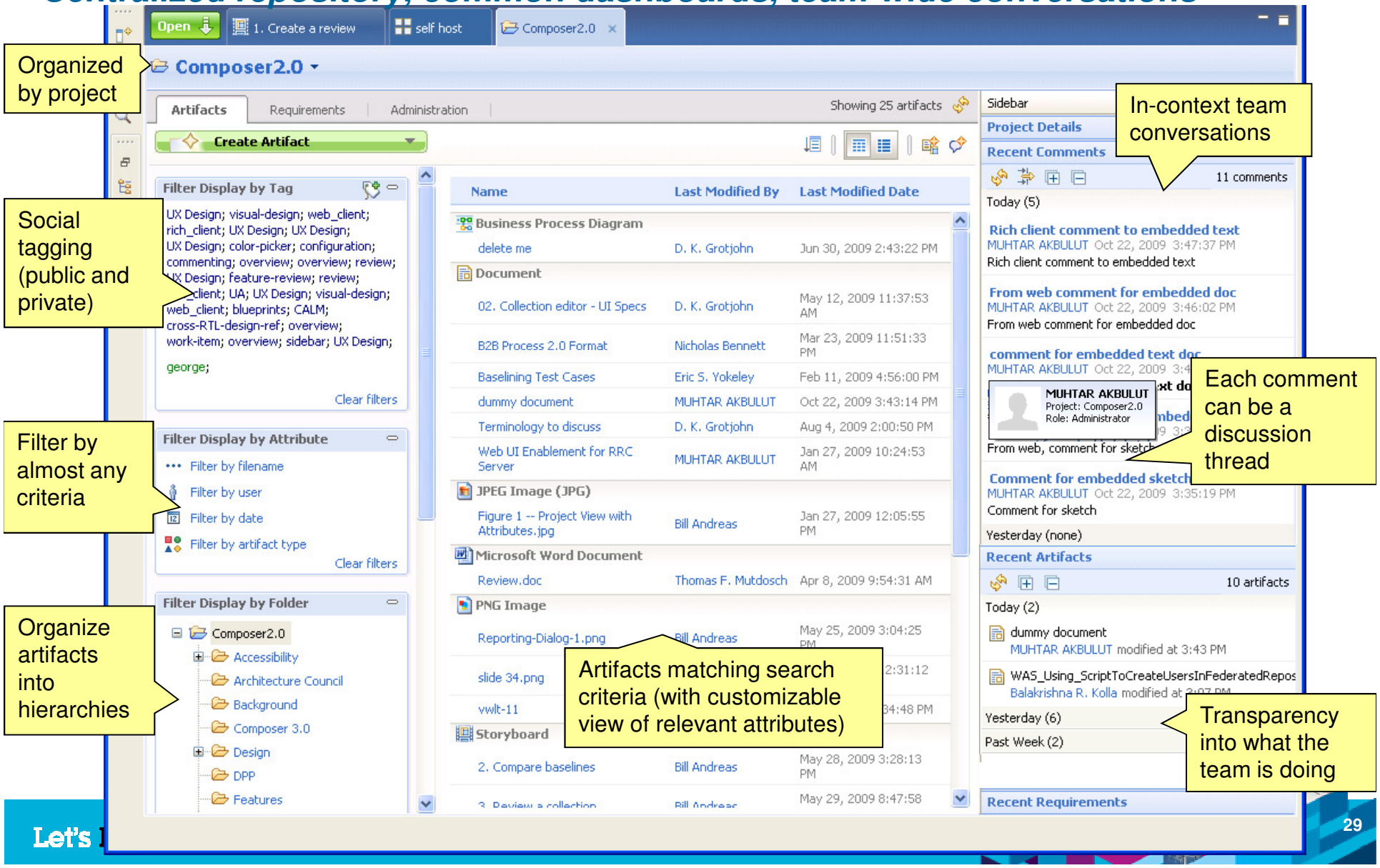




Visualize the User Experience with **User Interface Sketches and Storyboards**

Foster collaboration and team transparency

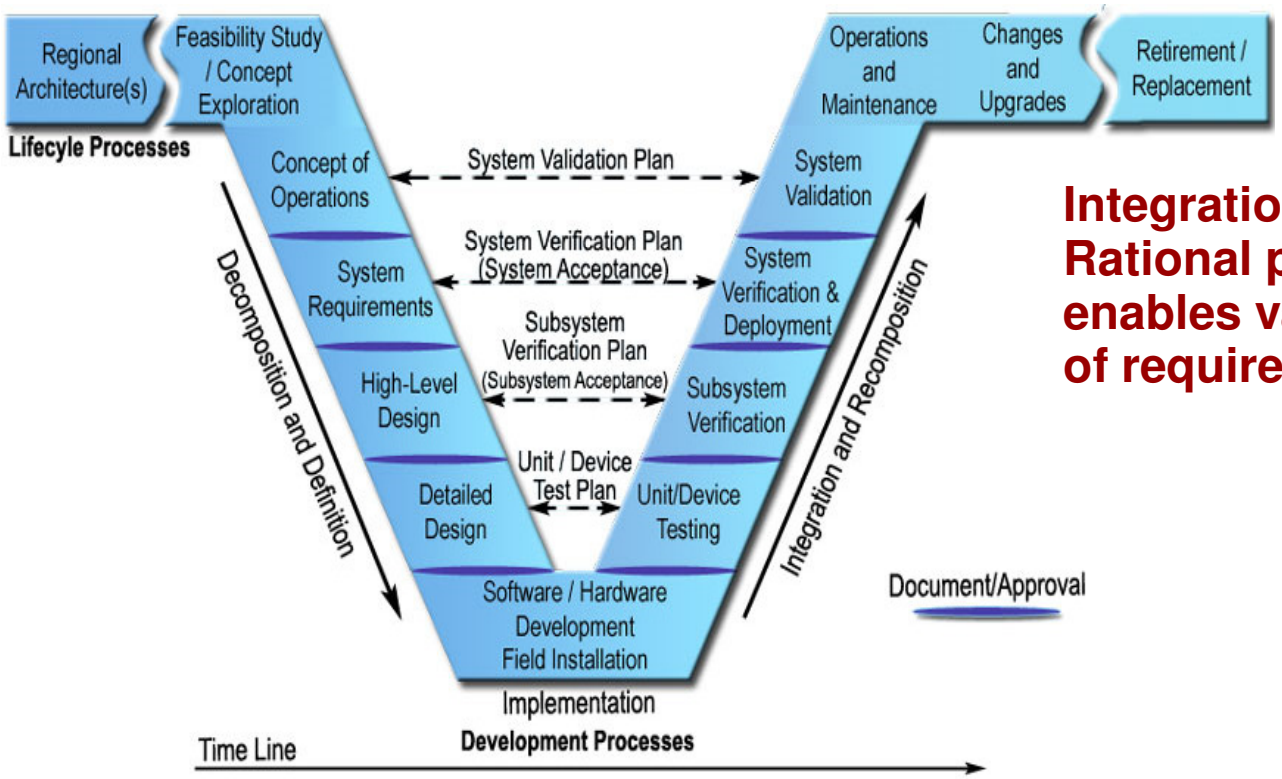
Centralized repository, common dashboards, team-wide conversations



Rational Requirements Composer

Traceability Focus

RRC focuses on early life cycle informal traceability



Integration with other Rational products enables validation of requirements

Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

How well do you Write Requirements?
Real World Advertisement Examples

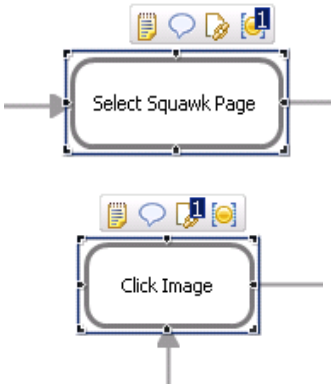
- In an office: After tea break staff should empty the teapot and stand upside down on the draining board.
- In a safari park: Elephants please stay in your car.
- Red Tape Holds Up New Bridge.
- Hospitals are Sued by 7 Foot Doctors.
- Local High School Dropouts Cut in Half.

Traceability Approaches
It Depends!

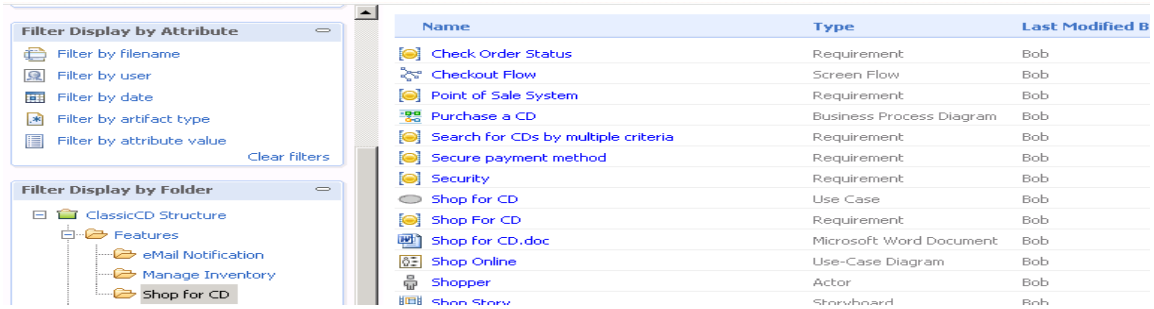
- Approach really **depends on where you start from**
  - New concept to the organization
  - New system within existing domain expertise
  - New functionality to existing system
  - Maintenance/bug fixes
- **Goal** of traceability
  - Compliance/audit
  - Regulations
  - Internal controls
  - Maintenance
- **Tools** available
  - None (manual)
  - Partial life-cycle coverage (one of Requirements Composer, DOORS or RequisitePro
  - Full life-cycle coverage (Requirements Composer with RequisitePro or DOORS and Quality Manager and Team Concert)

Rational Requirements Composer
Linking Capabilities

- Rational Requirements Composer linking/traceability includes:
  - Linking from any artifact to another artifact within the tool
  - Linking to external artifacts with a valid URL
  - Custom attributes allows references to be included as attributes
  - Collaborative / Application Life-cycle Management (C/ALM) integration with:
    - Work Items (stories) in Rational Team Concert
    - Test Cases in Rational Quality Manager
  - Freeform folder structure allows implicit linking

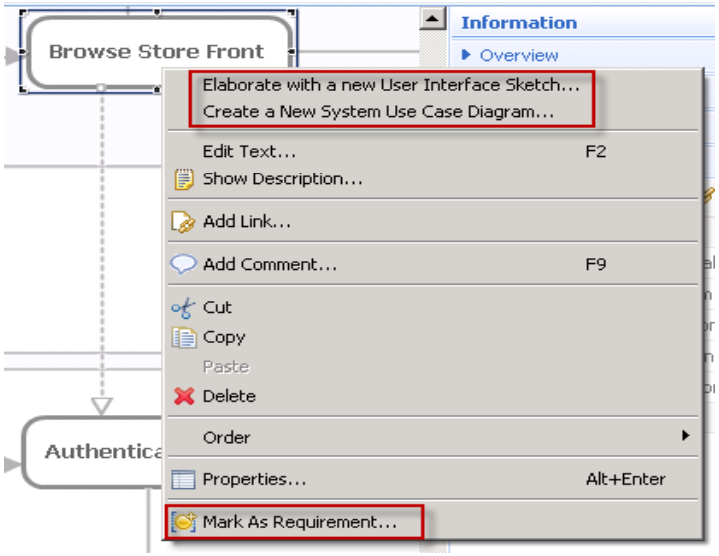


Name	Implemented By	Validated By
Add a wide variety of squawkers to the system		
Add tiled squawkers as a alternative to tabbed in the GUI	23: Add tiled squawkers as a alternative to...	
Car squawker	15: Car squawker	
Cat squawker	12: Cat squawker	
Clock squawker	17: Clock squawker	Test Case 23: GUI - Clock
Command line interface	16: Command line interface	
Dialog for selection of a single squawker	20: Dialog for selection of a single squawker	Test Case 10: GUI DefaultSquawker
Dog squawker	11: Dog squawker	Test Case 11 - GUI Dog
Enable declarative squawkers	25: Enable declarative squawkers	

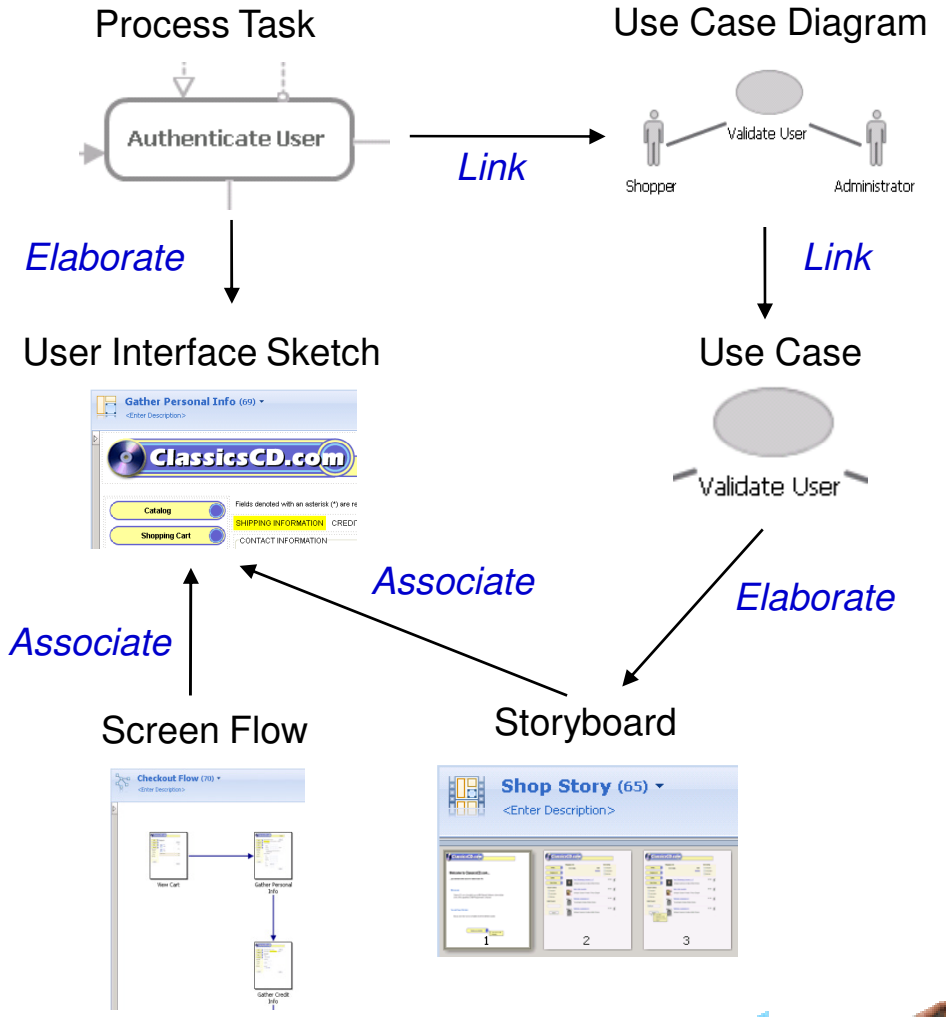


Rational Requirements Composer
Inherent Relationships

- Rational Requirements Composer inherent right click linking support



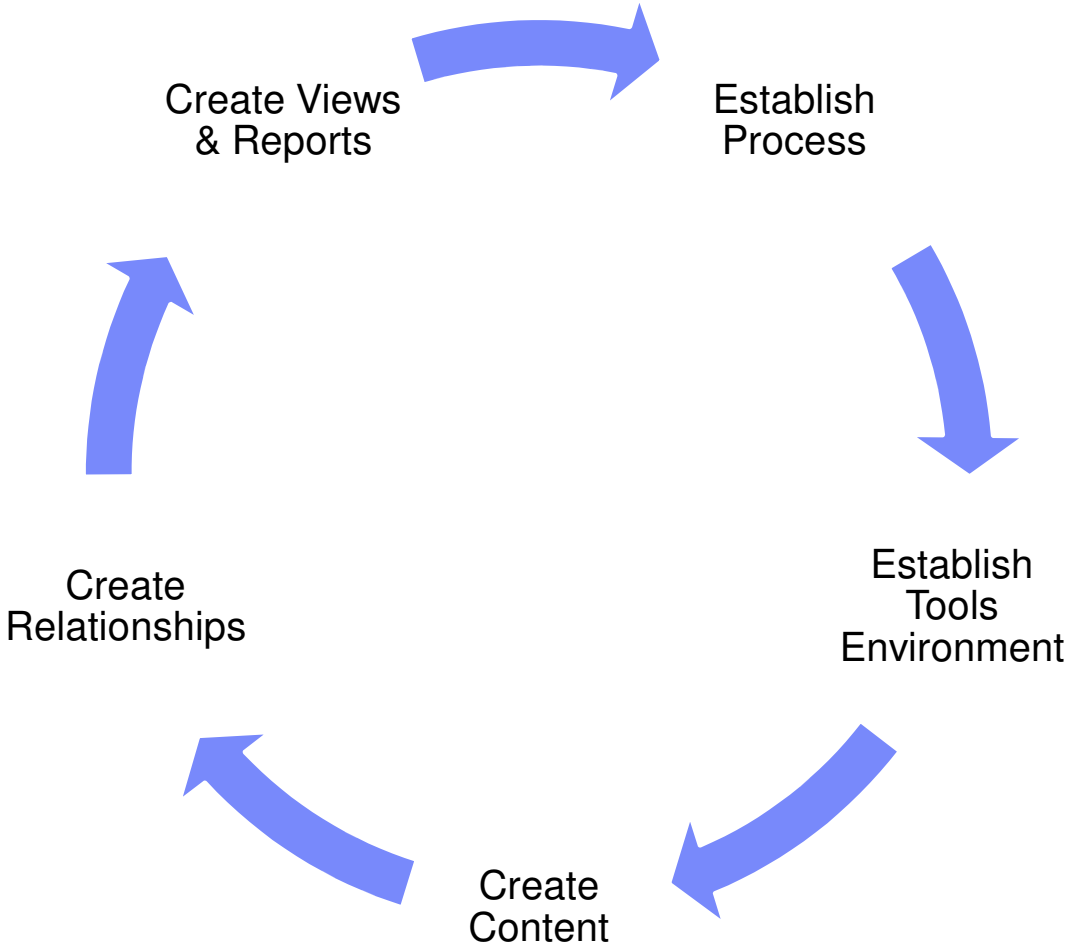
Requirements,
Internal Artifacts,
External Artifacts



Agenda

- Requirements are Everywhere
- What is Traceability?
- How do Tools Help?
- Rational Requirements Composer
- Traceability Approaches
- Best Practices

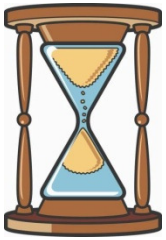
Moving from Files and Documents to ***Requirements***



Establish Process: 5 Steps to Agility

1. Timebox
2. Establish breadth
3. Prioritize
4. Go deeper
5. Repeat

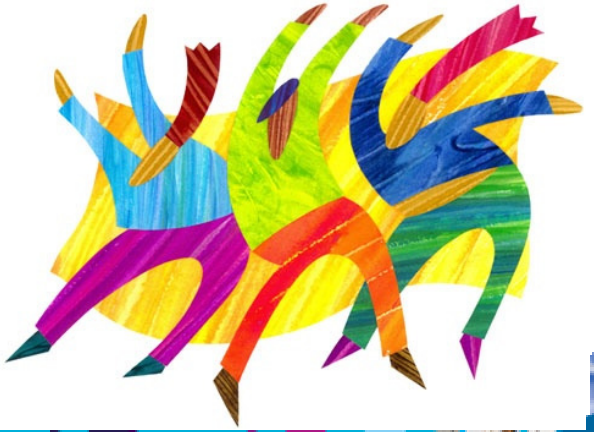
1. Timebox



- Establish direction, not perfection
  - Use what is known to develop requirements in phases
  - Establish a time-box for each phase
  - Keep each requirements cycle short and in a specific time frame
- Make assumptions, then move on
  - Be comfortable with assumptions for unknowns at current phase
  - Assumptions will either fall out of scope, or become requirements
- Don't try to capture all the detail up front
  - Doing so can lead to analysis paralysis

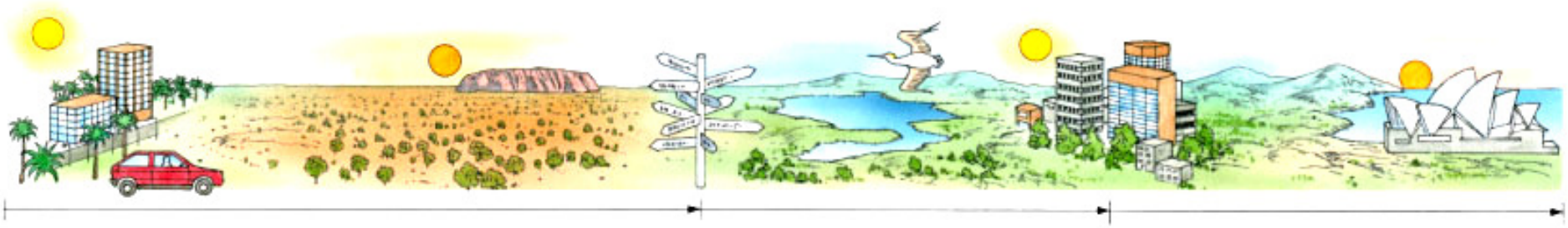
Create Content: Moving to a More Agile Approach

- Link documents, images, diagrams, and artifacts as they are collected and evolving
  - Create meaningful relationships across artifacts to paint the whole picture to the team
- Provide an accessible repository on the web for stakeholders to review and comment as needed
  - Invaluable to projects with GR component – less need for midnight calls!
- Resist the Giant Requirement Up Front (GRUF) temptation
  - Introduce a less rigid, more flexible requirements process
- Emphasis on the value of modeling
  - Fewer words, more pictures is always a good thing!



2. Establish Breadth

- Understand the 'lay of the land'
- Identifying the boundary: scope of automation
  - High-level requirements covering the *entire* breadth of the system
  - Staying within the time-box: whatever gets missed here can move to the next iteration
- Begin with a list of use cases and actors
  - Aim for a complete actor and Use Case set, start with outlines, details then get added incrementally

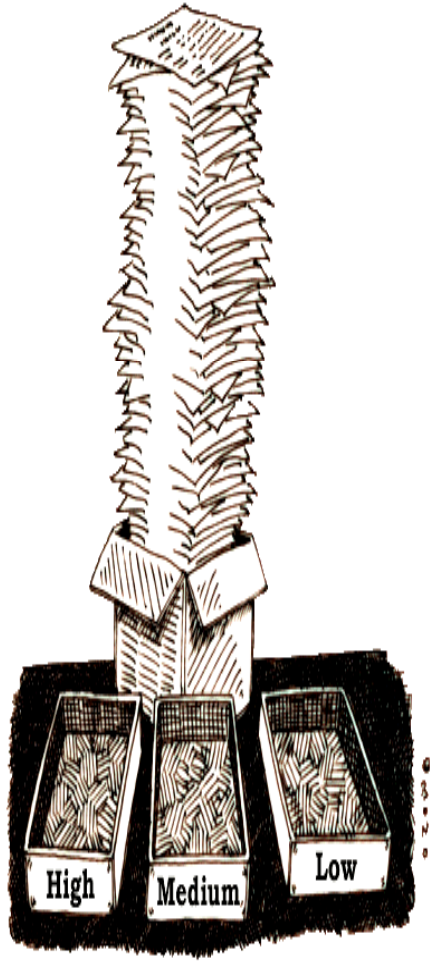


The Use Case Model in Composer

- ‘Automated’ tasks in business process model
  - Helps determine the scope of the system – black box for system use cases
- Aim is to ‘discover’ all use cases
  - Usually do not get it right the first time - need to combine or merge as needed later
- Sketch of the use case diagram
  - UML notation showing system boundary, actors & use case relationships
  - Initially start with a sketch, can turn into reusable components later
- Outline of use cases and actor descriptions
  - Use Case document template- provide brief description and identify major flows only

3. Prioritize

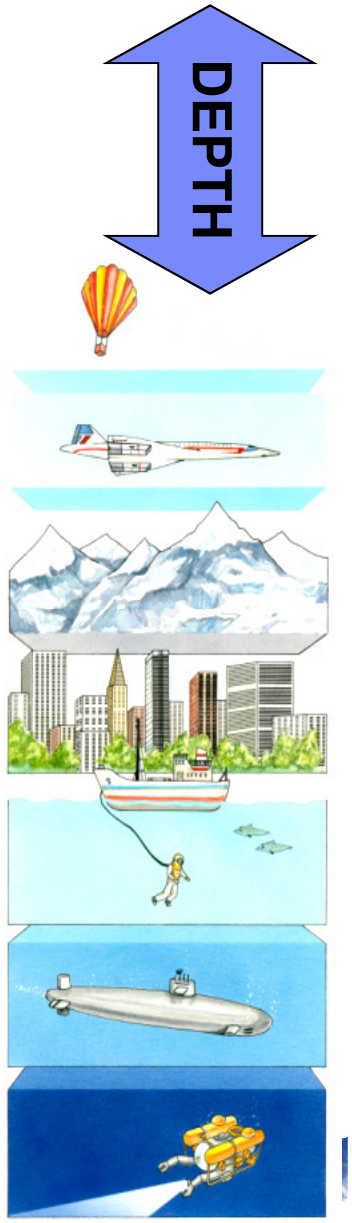
- According to RISK
  - Technical risk: Work with the architect to determine architectural risks
  - Business risk: Have discussions that challenge the customer
- According to BUSINESS VALUE
  - Which UCs and functions will deliver the biggest bang for the buck?
  - Pareto's Principle - **80/20 rule**
- Be ready to do it again! Will have to re-prioritize requirements in the next iteration…



Composer Can Help!

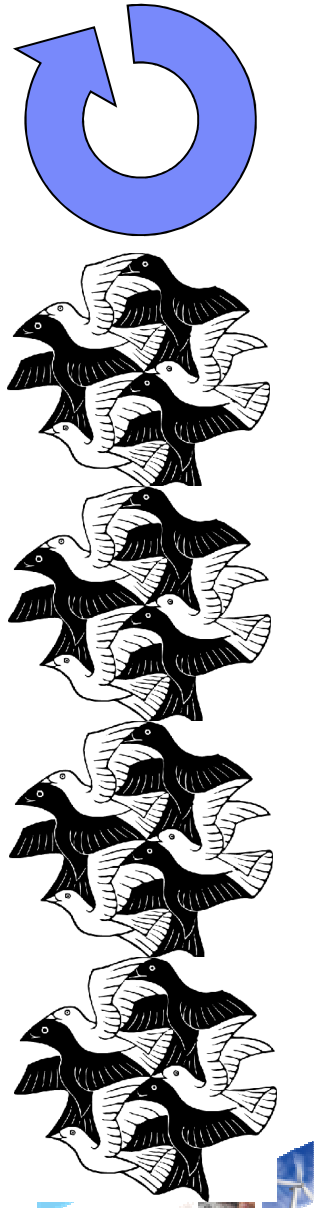
- Define and set attributes (priority, difficulty, risk)
  - These will not be lost and can later be imported to RequisitePro for management
- Commenting features to communicate and help achieve consensus
  - Each stakeholder may have their own idea/agenda – facilitate difficult discussions
- Business partners, stakeholders, development teams that do not have client access can take advantage of the web
  - Easy to use and navigate as needed – can send URL links to specific artifacts
- Define a strategy for how you will use attributes and tags consistently in RRC

4. Go Deeper



- Defining depth on the features / functions that are at the top of the list
  - The first few, highest priority UCs are addressed first – business rules & messages are referenced as well
- Moving from UC outlines to detailing all flows and major scenarios
  - Going through this process, it is not uncommon to discover that one UC should actually be two, or vice versa
- Scenarios are invaluable!
  - At the heart of the iterative approach, are the end to end user scenarios that can be coded and tested for that iteration…

5. Repeat



- The number of iterations will vary, but each iteration follows the four steps again
  - Timebox
  - Breadth
  - Prioritize
  - Depth
- With each iteration, emphasis naturally shifts from breadth to developing more depth (system detail)
  - The emphasis also begins to shift from definition to management of the existing requirement set
  - Managing change becomes a big component in later iterations

Requirements Identification and Creation

- As we repeat …consistency is key…
  - Requirements Definition Plan helps to plan upfront what will be 'outcome' from Composer
- Requirements created in Composer based on any artifact
  - Each requirement creates a rich text document
  - Can imbed or link other artifacts into the requirement (ie. Screenshot/wireframes)
- For large projects, **RequisitePro** needed for management
  - RequisitePro has many features to help with more complex reporting, traceability needs
- Early lifecycle adoption identified as key success criteria
  - Requirements definition happens during proposal phase on most services engagements
- Can trace to test cases in RQM and tasks in RTC


Questions

Jazz Million Seat March

Momentum is building…

Join the movement and try Jazz today!

http://www.jazz.net/sandbox

*Already **300,000+** Jazz seats in the marketplace*

Over *50,000* members active on Jazz.net

Community Feedback and Testimonials

*"Rational Team Concert has **excellent team collaboration** with automated project management dashboards that are transparent to everyone — not just managers. This immediate and automated feedback helps **keep teams on track and motivated** to achieve project goals."*

— Han Jie, Siemens

What Analysts are saying:

Liz Barnett, Ez-Insight: *"With the Jazz project, Rational has developed **breakthrough technology** and is **poised to set the standard for collaborative ALM**..."*

Julie Craig, EMA: *"Rational's differentiators are **difficult for competitors to equal,** and the new Jazz platform foundation may well turn out to be one of the best investments the Rational team has made."*

Innovate Around the World

Great Britain
Malaysia
Indonesia
Japan
Spain
China
Australia
Italy
Denmark
Netherlands
Korea
France
India
Singapore
Belgium
Philippines
Germany
Mexico
Taiwan
Thailand

Innovate – Around the World

http://www.ibm.com/software/rational/innovate/aroundtheworld/

Join us LIVE on September 14th

Innovate2010
Virtual Experience!

Let's build a smarter planet.

The Innovate Virtual Conference will deliver many of the same sessions as the live event. You'll be able to revisit the general sessions or listen to break outs you might have missed in Orlando. You can even download the presentation PDF to follow along and share with customers. You'll also be able to access many of the Innovate 2010 demos! Attendance is complimentary and includes 4 General Session videos, 6 Customer Experience Sessions & 6 Technical track sessions.

Register now for the Innovate Virtual Conference, where you can gain access to Innovate 2010 sessions and demos full of user expertise and key technical content that you can't get anywhere else now that Innovate is over.

REGISTER TODAY for the LIVE event: September 14, 2010!
http://events.unisfair.com/rt/ibm~innovate

IBM

www.ibm.com/software/rational

© Copyright IBM Corporation 2010. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.