SEG3101 (Fall 2009)

Requirements Management

Gunter Mussbacher, University of Ottawa

Based on material from:

Kotonya & Sommerville, Z. Zhang, IBM and Telelogic,
S. Somé 2008, and D. Amyot 2008-2009

Table of Contents

- Introduction to Requirements Management
- Traceability
- Baselines
- Change Management
- Requirements Management Tools

- A factor present in every successful project and absent in every unsuccessful project is sufficient attention to requirements.[1]

[1] Suzanne & James Robertson, “Requirements-Led Project Management”, Addison-Wesley, 2004

THE PROJECT REQUIREMENTS ARE FORMING IN MY MIND
NOW THEY'RE CHANGING... CHANGING... CHANGING... CHANGING... OKAY. NO, WAIT... CHANGING... CHANGING... DONE.
NATURALLY, I WON'T BE SHARING ANY OF THESE THOUGHTS WITH ENGINEERING.
I BUDGETED FOR SOME GOONS TO BEAT IT OUT OF YOU.
S. Adams
E-Mail: SCOTTADAMS@AOL.COM
© 1994 United Feature Syndicate, Inc.
E-Mail: SCOTTADAMS@AOL.COM
©1994 United Feature Syndicate

Introduction to Requirements Management

Why Do Requirements Change?

- Change in software development: as inevitable as difficult to control!
  - Better understanding: new requirements become apparent
  - Everything else is changing…
    - Business
    - Context
    - Technologies
    - Markets
    - …
- Possible responses to change
  - Add, modify, or remove requirements

Some Problems Due to Changing Requirements

- Requirements changing towards the end of development without any impact assessment
- Unmatched/outdated requirements specifications causing confusion and unnecessary rework
- Time spent coding, writing test cases or documentation for requirements that no longer exist

Requirements Management

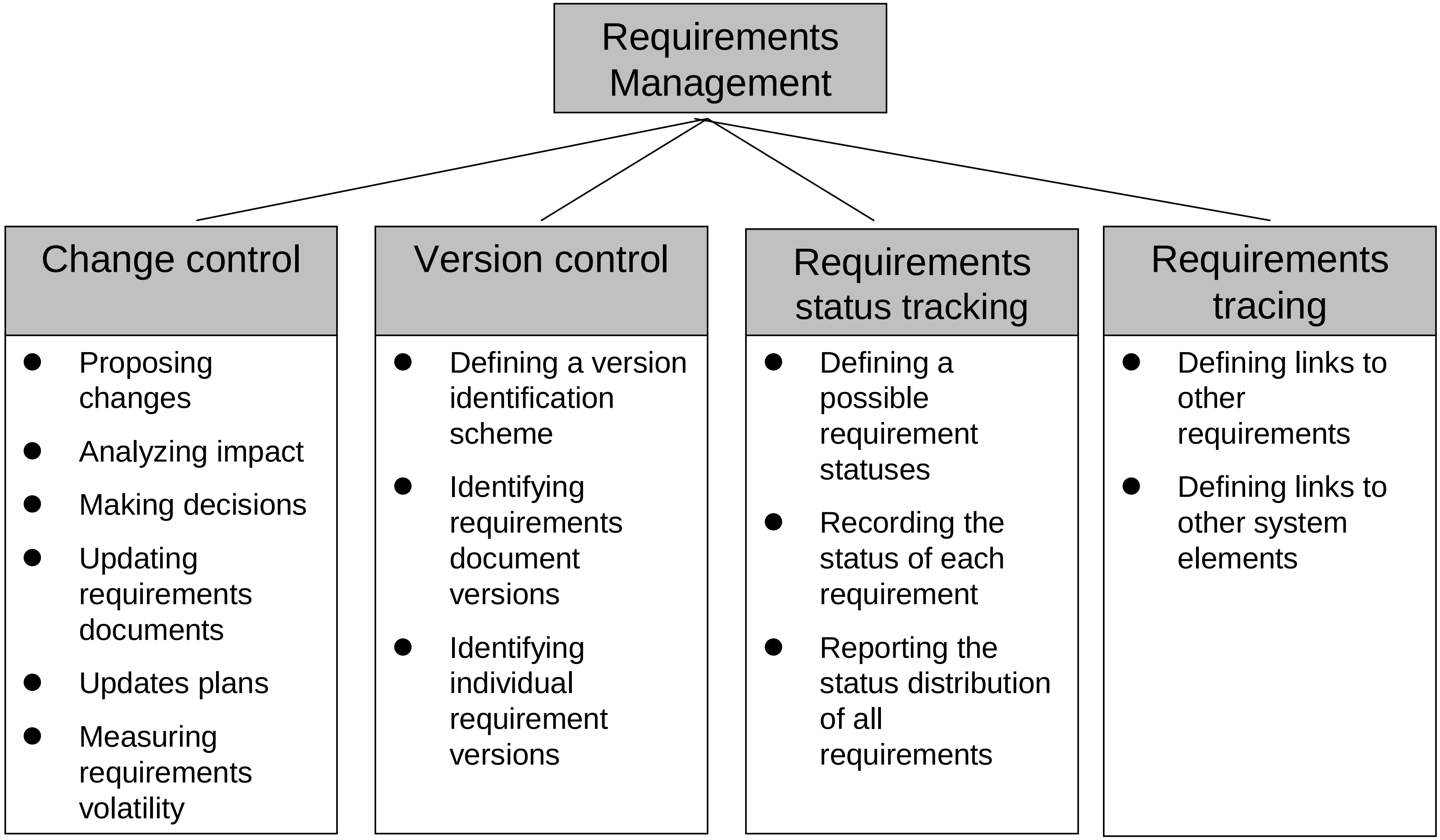
- A systematic approach to eliciting, organizing, and documenting the requirement of the system, and a process that establishes and maintains agreement between the customer and the project team on the changing requirements of the system.[1]

[1] Leffingwell & Widrig 1999, p.16

Requirements Management Activities (1)

- Requirements management includes all activities intended to maintain the integrity and accuracy of expected requirements
  - Manage changes to agreed requirements
  - Manage changes to baseline (increments)
  - Keep project plans synchronized with requirements
  - Control versions of individual requirements and versions of requirements documents
  - Manage relationships between requirements
  - Managing the dependencies between the requirements document and other documents produced in the systems engineering process
  - Track requirements status

Requirements Management Activities (2)



Requirements Management

Change control

- Proposing changes
- Analyzing impact
- Making decisions
- Updating requirements documents
- Updates plans
- Measuring requirements volatility

Version control

- Defining a version identification scheme
- Identifying requirements document versions
- Identifying individual requirement versions

Requirements status tracking

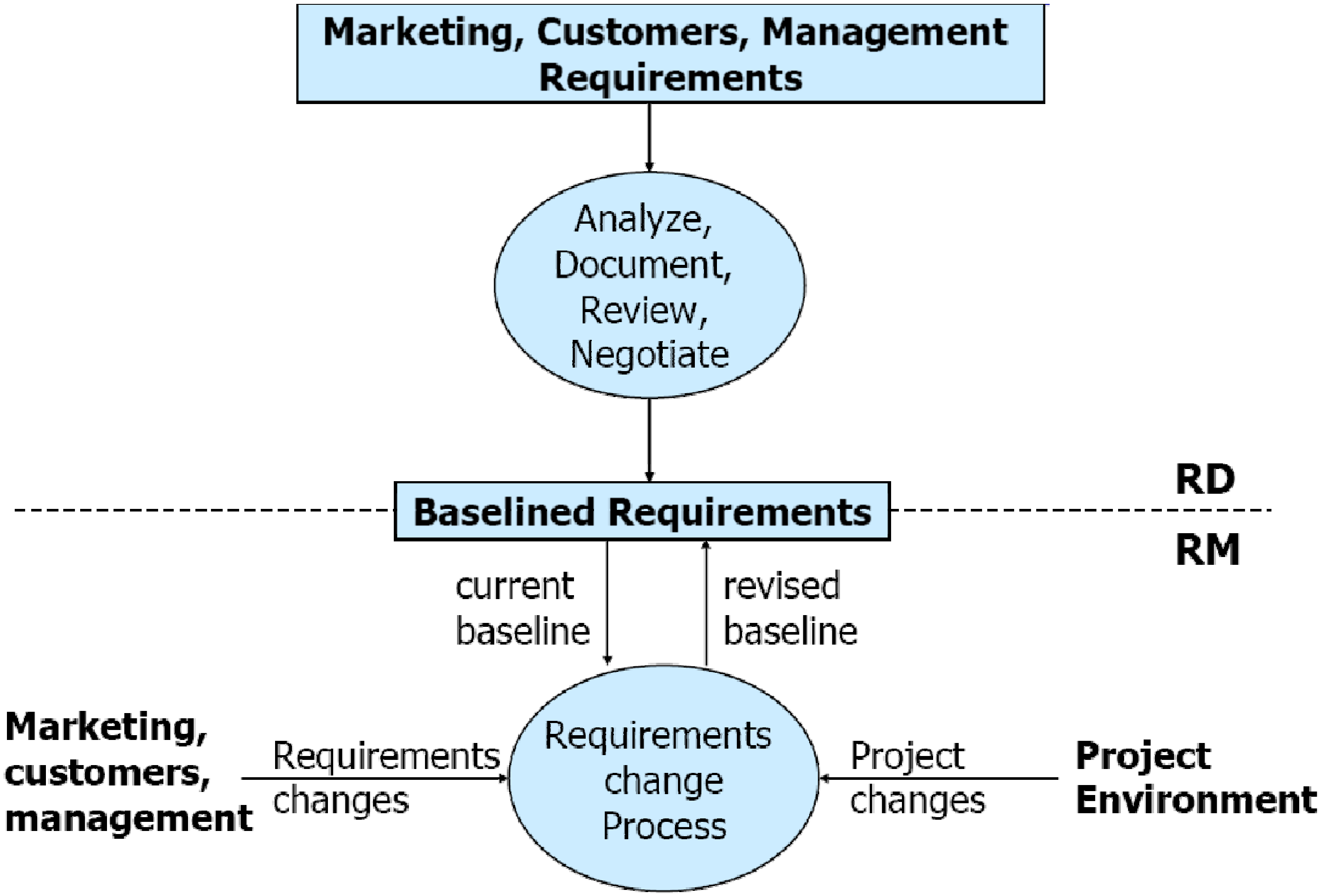
- Defining a possible requirement statuses
- Recording the status of each requirement
- Reporting the status distribution of all requirements

Requirements tracing

- Defining links to other requirements
- Defining links to other system elements

Source: Wiegers, 1999

Requirements Development (RD) and Management (RM)


Marketing, Customers, Management Requirements
Analyze, Document, Review, Negotiate
Baselined Requirements
RD
RM
current baseline
revised baseline
Marketing, customers, management
Requirements changes
Requirements change Process
Project changes
Project Environment

Source: Wiegers, 1999

From Management to Tools

- Changes lead to a need for management
- There is no management without:
  - Traceability
  - Baselines enabling comparisons
- From a practical point of view, there is no traceability or management without appropriate tools

In theory, practice and theory are similar…
But in practice they are different ☺

Requirements Change Factors (1)

- Requirements errors, conflicts, and inconsistencies
  - May be detected at any phase (when requirements are analyzed, specified, validated, or implemented)
- Evolving customer/user knowledge of the system
  - When the requirements are developed, customers/users simultaneously develop a better understanding of what they really need
- Technical, schedule, or cost problems
  - Difficult to plan and know everything in advance
  - We may have to revisit the list of requirements and adapt it to the current situation

Requirements Change Factors (2)

- Changing customer priorities, new needs
  - May be caused by a change in the system environment (technological, business, political...), i.e., the context
  - Business and strategic goals may change
  - May be caused by the arrival of a new competitor
  - Laws and regulations may change
  - Collaborating systems may change
  - May also be caused by technology changes in the enterprise (migration to a new operating system, DBMS…)
  - May be caused by organizational changes (organizational structure, business processes, employees…)

Requirements Volatility

- Requirements continuously change
  - While the requirements are being elicited, analyzed, specified, and validated and after the system has gone into service
- Some requirements are usually more subject to change than others
  - Stable requirements are concerned with the essence of a system and its application domain
    - Derived from the client’s principal business activities or the domain model
    - They change more slowly than volatile requirements
    - E.g., a hospital will always have doctors, nurses, patients…
  - Volatile requirements are specific to the instantiation of the system in a particular environment for a particular customer at a particular time
    - E.g., in a hospital, we can think of requirements related to the policies of the government health system

Types of Volatile Requirements

- Mutable requirements
  - These are requirements which change because of changes to the environment in which the system is operating
- Emergent requirements
  - These are requirements which cannot be completely defined when the system is specified but which emerge as the system is designed and implemented
- Consequential requirements
  - These are requirements which are based on assumptions about how the system will be used
  - Once the system is in place, some of these assumptions will be wrong
- Compatibility requirements
  - These are requirements which depend on other equipment, technology, or processes

Expectations of Requirements Management (1)

- Identification of individual requirements
- Traceability from highest level requirements to implementation
  - Established via links through a requirements database
  - Links between requirements and design models, tests, code…
  - Coverage and consistency analysis
  - What are the traceability policies? What types of links? From where? To where?
- Impact assessments of proposed changes
  - Analysis tools let you see which other requirements (and other linked artifacts) will be affected by a change

Expectations of Requirements Management (2)

- Controlled access to current project information
  - A shared database ensures that all users are working with current data (consistency, parallel access)
  - A central repository allows all users to see the information that they need to see (visibility)
- Change control
  - Change proposal system implements controlled process for managing change
  - How do we collect, document, and address changes?
- Deployment of required tool support
  - To help manage requirements change

Identification of Requirements

- It is essential for requirements management that every requirement has a unique identification
  - The most common approach is requirements numbering based on chapter/section in the requirements document
- There are several problems with this approach
  - Numbers cannot be unambiguously assigned until the document is complete
  - Assigning chapter/section numbers is an implicit classification of the requirements ➔ may mislead readers of the document into thinking that the most important relationships are with requirements in the same section

Requirements Identification Techniques

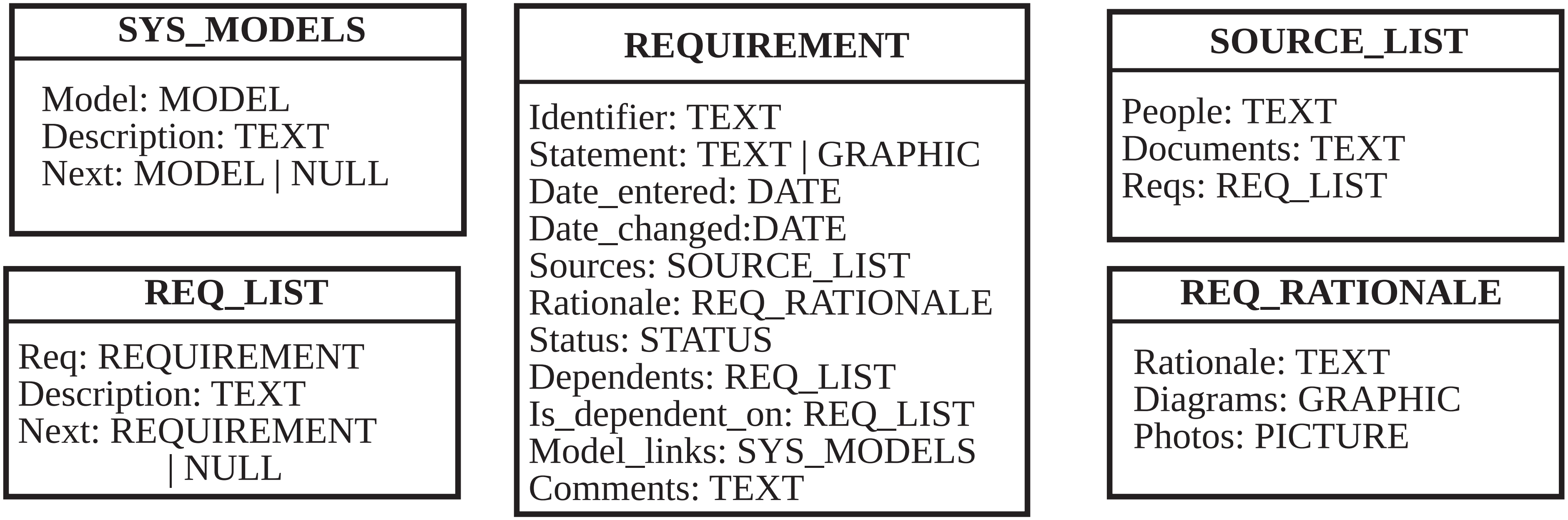
- Dynamic renumbering
  - Some word processing systems allow for automatic renumbering of paragraphs and the inclusion of cross references
  - As you reorganise your document and add new requirements, the system keeps track of the cross references and automatically renumbers your requirements depending on its chapter, section, and position within the section
- Database record identification
  - When a requirement is identified, it is entered in a requirements database and a database record identifier is assigned which is then used for all subsequent references to the requirement
- Symbolic identification
  - Requirements can be identified by giving them a symbolic name which is associated with the requirement itself (e.g., SEC1, SEC2, SEC3… may be used for requirements which relate to system security)

BTW, Requirements Have Attributes!

- Apart from an identifier, requirements should have attributes that establish context and background, and go beyond the requirement description
- For filtering, analysis, metrics…
  - Creation date, Last update, Author, Stakeholders (Owners / Source)
  - Version number
  - Status, Priority, Importance, Stability
  - Rationale, Comments
  - Acceptance criteria
  - Subsystem / Product release number
  - …
- The more complex the project, the richer the attributes…
- Many attributes are predefined in RM tools, others are defined by requirements engineers as required by the project

Requirements Attributes

- Classes and attributes of a requirements management database



SYS_MODELS

Model: MODEL
Description: TEXT
Next: MODEL | NULL

REQ_LIST

Req: REQUIREMENT
Description: TEXT
Next: REQUIREMENT | NULL

REQUIREMENT

Identifier: TEXT
Statement: TEXT | GRAPHIC
Date_entered: DATE
Date_changed:DATE
Sources: SOURCE_LIST
Rationale: REQ_RATIONALE
Status: STATUS
Dependents: REQ_LIST
Is_dependent_on: REQ_LIST
Model_links: SYS_MODELS
Comments: TEXT

SOURCE_LIST

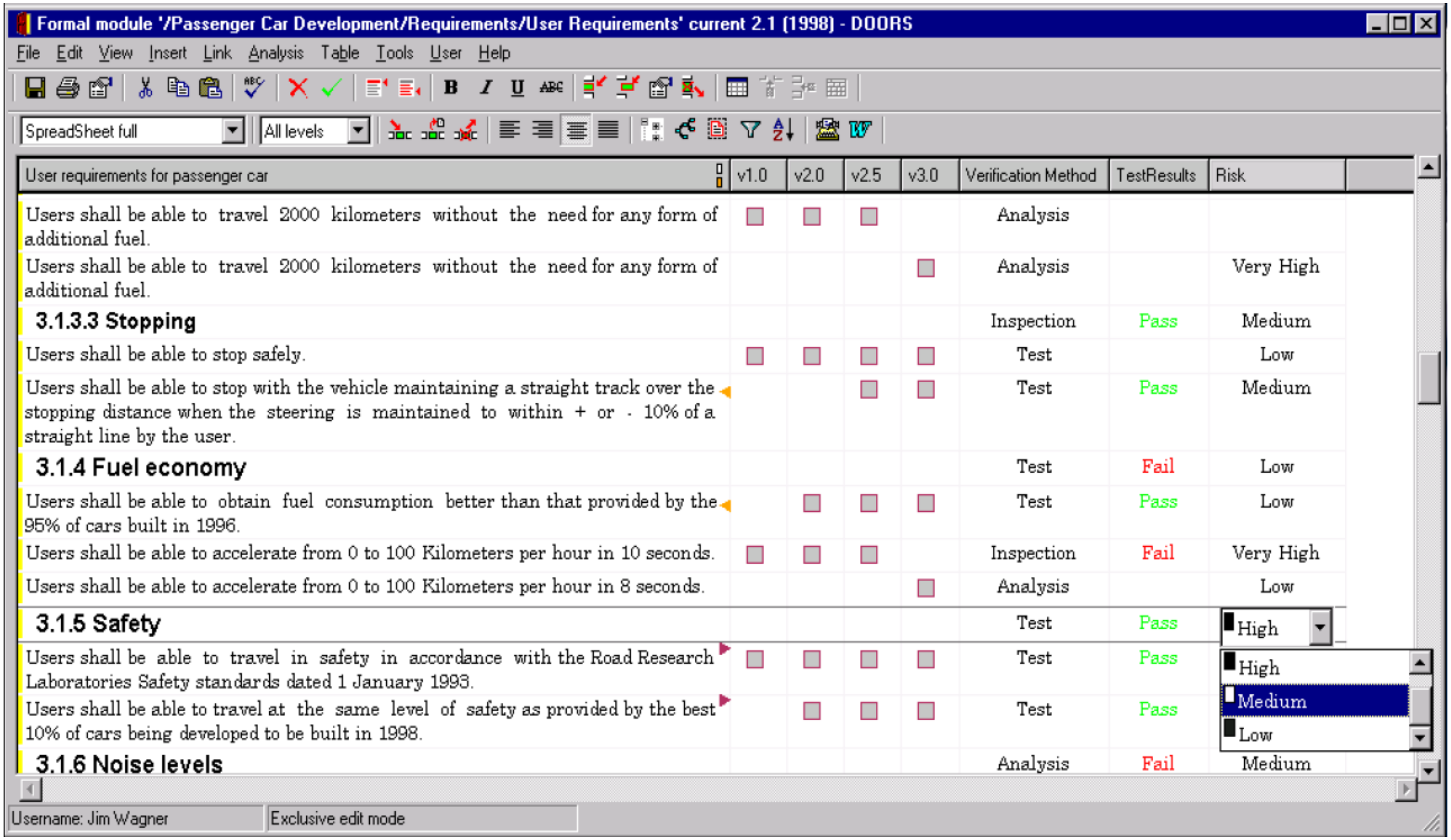
People: TEXT
Documents: TEXT
Reqs: REQ_LIST

REQ_RATIONALE

Rationale: TEXT
Diagrams: GRAPHIC
Photos: PICTURE

- Select only the necessary attributes!

DOORS – Objects and Attributes



Formal module '/Passenger Car Development/Requirements/User Requirements' current 2.1 (1998) - DOORS

User requirements for passenger car	v1.0	v2.0	v2.5	v3.0	Verification Method	TestResults	Risk
Users shall be able to travel 2000 kilometers without the need for any form of additional fuel.	□	□	□		Analysis		
Users shall be able to travel 2000 kilometers without the need for any form of additional fuel.				□	Analysis		Very High
3.1.3.3 Stopping					Inspection	Pass	Medium
Users shall be able to stop safely.	□	□	□	□	Test		Low
Users shall be able to stop with the vehicle maintaining a straight track over the stopping distance when the steering is maintained to within + or - 10% of a straight line by the user.			□	□	Test	Pass	Medium
3.1.4 Fuel economy					Test	Fail	Low
Users shall be able to obtain fuel consumption better than that provided by the 95% of cars built in 1996.		□	□	□	Test	Pass	Low
Users shall be able to accelerate from 0 to 100 Kilometers per hour in 10 seconds.	□	□	□		Inspection	Fail	Very High
Users shall be able to accelerate from 0 to 100 Kilometers per hour in 8 seconds.				□	Analysis		Low
3.1.5 Safety					Test	Pass	High
Users shall be able to travel in safety in accordance with the Road Research Laboratories Safety standards dated 1 January 1993.	□	□	□	□	Test	Pass	High Medium Low
Users shall be able to travel at the same level of safety as provided by the best 10% of cars being developed to be built in 1998.		□	□	□	Test	Pass	
3.1.6 Noise levels					Analysis	Fail	Medium

Username: Jim Wagner

Exclusive edit mode

Requirements Statuses

- Help manage the requirement lifecycle
  - Their number and nature depend on the process in place
- Example of a set of statuses:
  - Proposed: by some stakeholder
  - Approved: part of baseline, committed to implement
  - Rejected: after evaluation
  - Implemented: designed and implemented
  - Verified: Relevant tests have passed
  - Deleted: Removed from list
- RM includes amongst its tasks the tracking of the status of all requirements during the project

Version Control

- Another essential aspect of requirements management
  - Every version of a requirement needs to be uniquely identified
  - The last version of a requirement must be available to all team members
  - Changes need to be documented and clearly communicated
  - A version identifier must be updated with every change to the requirement
- Requirements documents should include
  - A revision history: changes, dates, by whom, why...
  - Standard markers for revisions (e.g., strikethrough or underlined text, coloring, line markers…)
- Version control tool may be used
  - To store and manage the revision history
  - To store justifications (to add, modify, delete, reject a requirement)

Traceability

Traceability?

- "Can I ask you some questions?"
- "By all means."
- "Okay. Well, for starters I'll have who, what, when and where and then wither, whence and wherefore for a follow-up, and then one bit side-order of why."

Source: Zaphod Beeblebrox & Zarniwoop, The Hitchhiker's Guide to the Galaxy

Traceability Quotes (1)

- Requirements traceability refers to the ability to describe and follow the life of a requirement, in both forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of ongoing refinement and iteration in any of these phases)”.[1]
- A software requirements specification is traceable if the origin of each of its requirements is clear and if it facilitates the referencing of each requirement in future development or enhancement documentation.[2]
- Traceability gives essential assistance in understanding the relationships that exist within and across software requirements, design, and implementation.[3]
- A link or relationship defined between entities.[4]

[1] Gotel & Finkelstein, 1994; [2] IEEE Standard 830-1998; [3] Palmer, 2000; [4] Watkins and Neal, 1994

Traceability Quotes (2)

- Traceability is often mandated by contracts and standards.[1]
  - E.g., military and aerospace
- One cannot manage what cannot be traced.[2]
- Traceability information helps assess the impact of changes to requirements, connecting these requirements as well as requirements for other representations of the system.[3]
- Traceability is a property of a system description technique that allows changes in one of the three system descriptions – requirements, specifications, implementation – to be traced to the corresponding portions of the other descriptions. The correspondence should be maintained through the lifetime of the system.[4]

[1-2] Watkins and Neal, 1994; [3] Kotonya and Sommerville, 1998; [4] Greenspan, McGowan, 1978

Importance of Traceability (1)

- Requirements cannot be managed effectively without requirements traceability
  - A requirement is traceable if you can discover who suggested the requirement, why the requirement exists, what requirements are related to it, and how that requirement relates to other information such as systems designs, implementations and user documentation

Importance of Traceability (2)

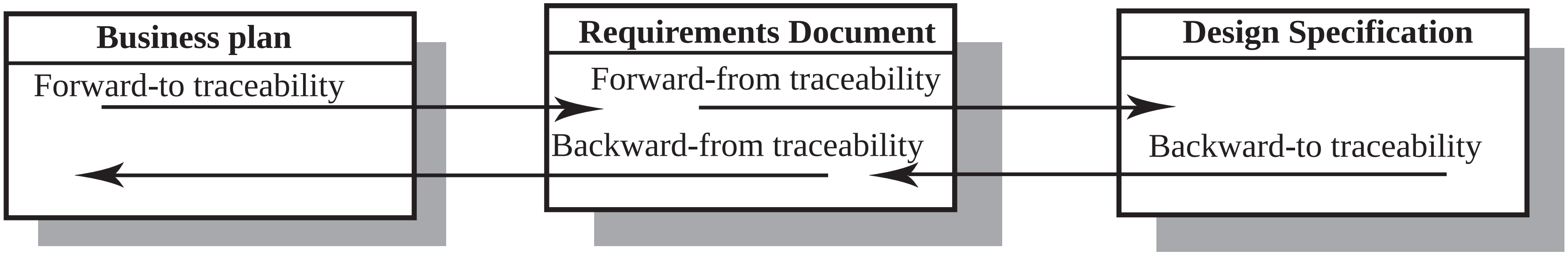
- Benefits of traceability
  - Prevents losing knowledge
  - Supports the verification process (certification, localization of defects)
  - Impact analysis
  - Change control
  - Process monitoring (e.g., missing links indicate completion level)
  - Improved software quality (make changes correctly and completely)
  - Reengineering (define traceability links is a way to record reverse engineering knowledge)
  - Reuse (by identifying what goes with a requirement: design, code…)
  - Risk reduction (e.g., if a team member with key knowledge leaves)

Traceability Difficulties

- Various stakeholders require different information
- Huge amount of requirements traceability information must be tracked and maintained
- Manual creation of links is **very** demanding
  - Likely the most annoying problem
- Specialized tools must be used
- Integrating heterogeneous models/information from/to different sources (requirements, design, tests, code, documentation, rationales…) is not trivial
- Requires organizational commitment (with an understanding of the potential benefits)

Backward and Forward Traceability (1)

- Backward traceability
  - To previous stages of development
  - Depends upon each element explicitly referencing its source in earlier documents
- Forward traceability
  - To all documents spawned by a document
  - Depends upon each element in the document having a unique name or reference number



Source of figure: Kotonya and Sommerville

Backward and Forward Traceability (2)

- Top to bottom from requirements' point of view
  - Forward-to traceability
    - Links other documents (which may have preceded the requirements document) to relevant requirements
    - Help validation
    - Help evaluate which requirements are affected by changes to users' needs
  - Forward-from traceability
    - Links requirements to the design and implementation components
    - Help assure that all requirements have been satisfied

Backward and Forward Traceability (3)

- Bottom to top from requirements' point of view
  - Backward-to traceability
    - Links design and implementation components back to requirements
    - Help determine why each item is designed/implemented
  - Backward-from traceability
    - Links requirements to their sources in other documents or people
    - Help validation
    - Help evaluate how changes to requirements impact stakeholders needs

Types of Traceability (1)

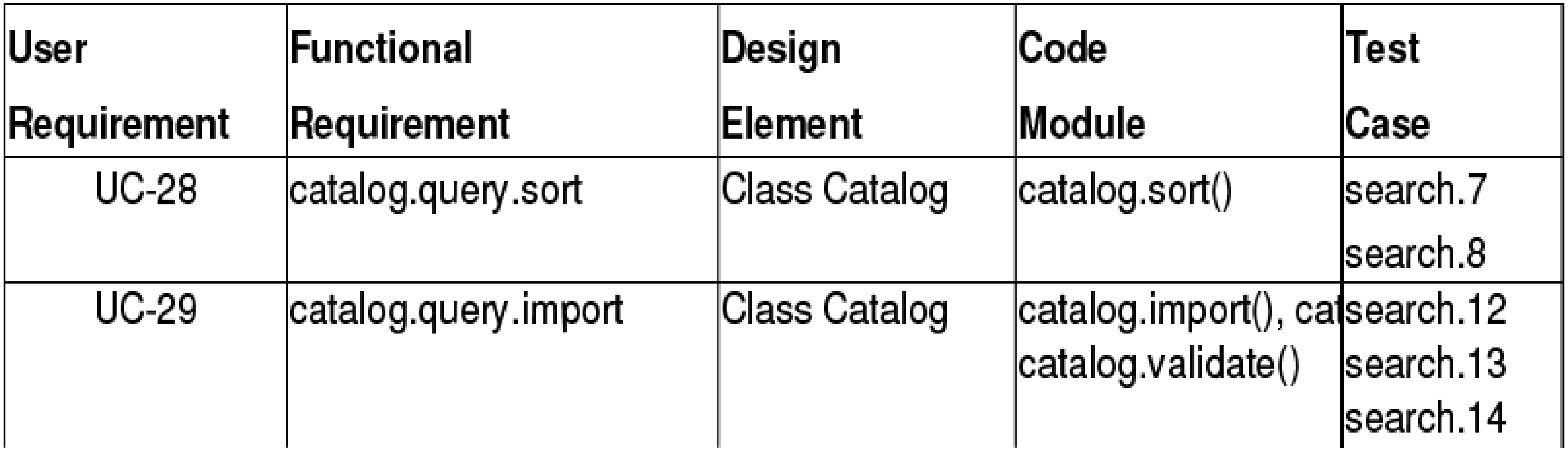
- Requirements – source traceability
  - Links requirements with a person or document
- Requirements – rationale traceability
- Requirements – requirements traceability
  - Links requirements with other requirements which are, in some way, dependent on them
- Requirements – architecture traceability
  - Links requirements with the subsystems where these requirements are implemented (particularly important where subsystems are being developed by different subcontractors)
- Requirements – design traceability
  - Links requirements with specific hardware or software components in the system which are used to implement the requirement

Types of Traceability (2)

- Requirements – interface traceability
  - Links requirements with the interfaces of external systems which are used in the provision of the requirements
- Requirements – feature traceability
- Requirements – tests traceability
  - Links requirements with test cases verifying them (used to verify that the requirement is implemented)
- Requirements – code traceability
  - Generally not directly established, but can be inferred

Representation – Traceability Table

- Show the relationships between requirements or between requirements and other artifacts
- Table can be set up to show links between several different elements
- Backward and forward traceability
- Difficult to capture different types of links



User Requirement	Functional Requirement	Design Element	Code Module	Test Case
UC-28	catalog.query.sort	Class Catalog	catalog.sort()	search.7 search.8
UC-29	catalog.query.import	Class Catalog	catalog.import(), cat catalog.validate()	search.12 search.13 search.14

Representation – Traceability Matrix

- Define links between pairs of elements
  - E.g., requirements to requirement, use case to requirement, requirement to test case…
- Can be used to defined relationships between pairs
  - E.g., specifies/is specified by, depends on, is parent of, constrains…
- More amenable to automation than traceability table

Depends-on

	R1	R2	R3	R4	R5	R6
R1			*	*		
R2					*	*
R3				*	*	
R4		*				
R5						*
R6						

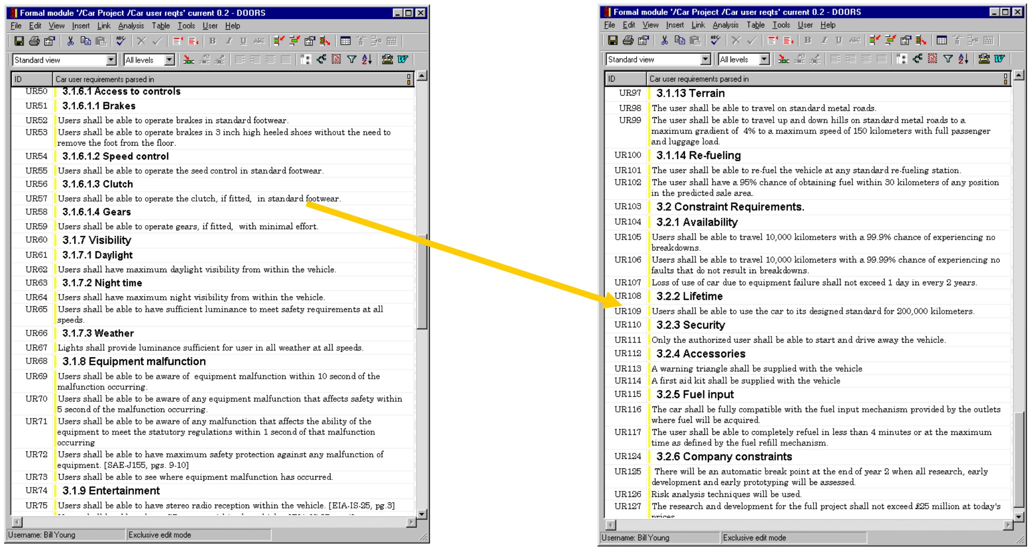
Representation – Traceability List

- Traceability matrices become more of a problem when there are hundreds or thousands of requirements as the matrices become large and are sparsely populated
- A simplified form of a traceability matrix may be used where, along with each requirement description, one or more lists of the identifiers of related requirements are maintained

Requirement	Depends-on
R1	R3, R4
R2	R5, R6
R3	R4, R5
R4	R2
R5	R6

Example – DOORS Links

- A relationship between two objects in the DOORS database is established using a link
  - One source object and one destination object
- Links can be followed in either direction
- Possible to have many links between the same two objects
  - Links also have types and attributes!



Formal module '/Car Project /Car user reqts' current 0.2 - DOORS

ID	Car user requirements parsed in
UR50	**3.1.6.1 Access to controls**
UR51	**3.1.6.1.1 Brakes**
UR52	Users shall be able to operate brakes in standard footwear.
UR53	Users shall be able to operate brakes in 3 inch high heeled shoes without the need to remove the foot from the floor.
UR54	**3.1.6.1.2 Speed control**
UR55	Users shall be able to operate the seed control in standard footwear.
UR56	**3.1.6.1.3 Clutch**
UR57	Users shall be able to operate the clutch, if fitted, in standard footwear.
UR58	**3.1.6.1.4 Gears**
UR59	Users shall be able to operate gears, if fitted, with minimal effort.
UR60	**3.1.7 Visibility**
UR61	**3.1.7.1 Daylight**
UR62	Users shall have maximum daylight visibility from within the vehicle.
UR63	**3.1.7.2 Night time**
UR64	Users shall have maximum night visibility from within the vehicle.
UR65	Users shall be able to have sufficient luminance to meet safety requirements at all speeds.
UR66	**3.1.7.3 Weather**
UR67	Lights shall provide luminance sufficient for user in all weather at all speeds.
UR68	**3.1.8 Equipment malfunction**
UR69	Users shall be able to be aware of equipment malfunction within 10 second of the malfunction occurring.
UR70	Users shall be able to be aware of any equipment malfunction that affects safety within 5 second of the malfunction occurring.
UR71	Users shall be able to be aware of any malfunction that affects the ability of the equipment to meet the statutory regulations within 1 second of that malfunction occurring
UR72	Users shall be able to have maximum safety protection against any malfunction of equipment. [SAE-J155, pgs. 9-10]
UR73	Users shall be able to see where equipment malfunction has occurred.
UR74	**3.1.9 Entertainment**
UR75	Users shall be able to have stereo radio reception within the vehicle. [EIA-IS-25, pg.3]

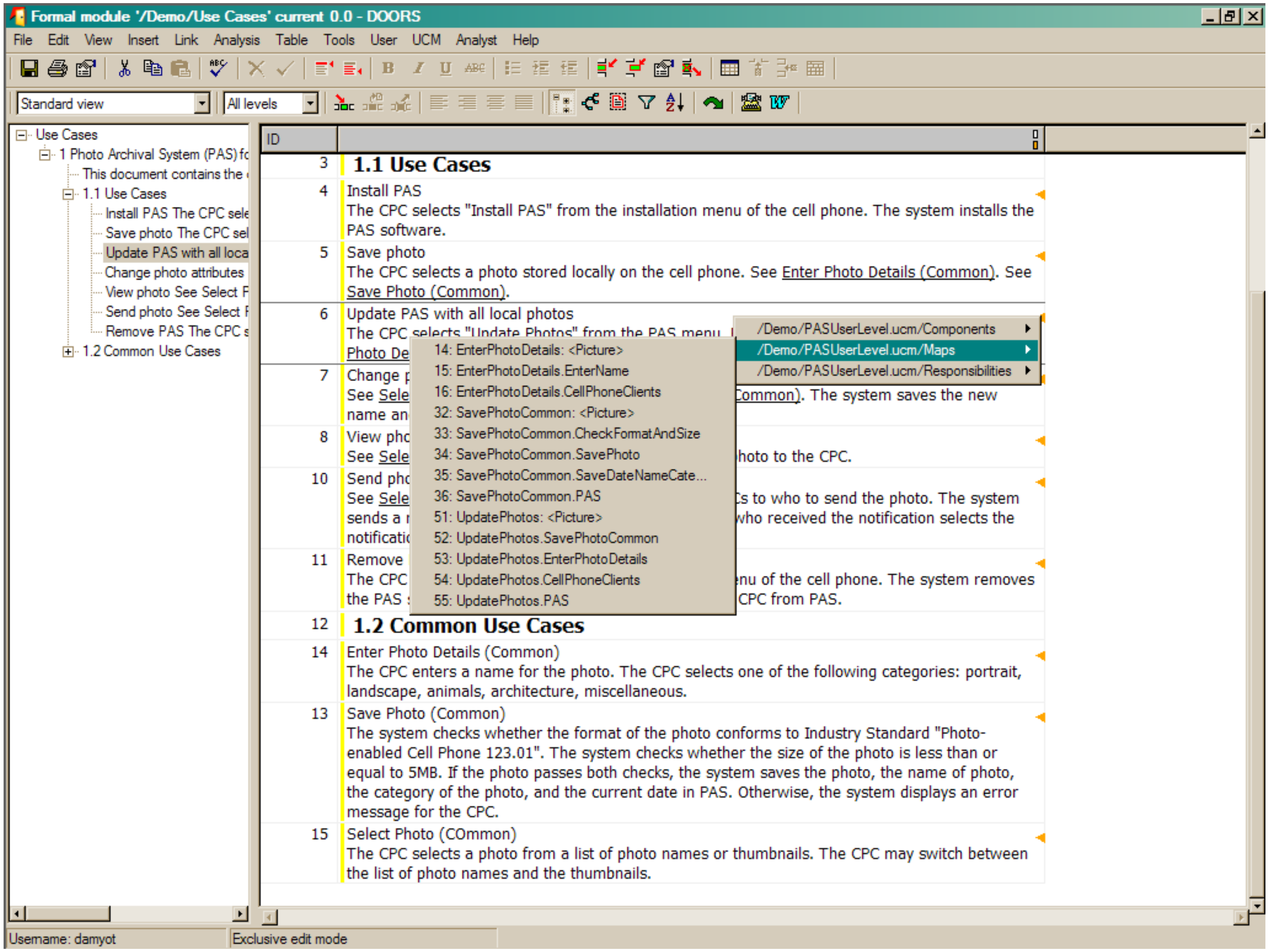
Username: Bill Young | Exclusive edit mode

Formal module '/Car Project /Car user reqts' current 0.2 - DOORS

ID	Car user requirements parsed in
UR97	**3.1.13 Terrain**
UR98	The user shall be able to travel on standard metal roads.
UR99	The user shall be able to travel up and down hills on standard metal roads to a maximum gradient of 4% to a maximum speed of 150 kilometers with full passenger and luggage load.
UR100	**3.1.14 Re-fueling**
UR101	The user shall be able to re-fuel the vehicle at any standard re-fueling station.
UR102	The user shall have a 95% chance of obtaining fuel within 30 kilometers of any position in the predicted sale area.
UR103	**3.2 Constraint Requirements.**
UR104	**3.2.1 Availability**
UR105	Users shall be able to travel 10,000 kilometers with a 99.9% chance of experiencing no breakdowns.
UR106	Users shall be able to travel 10,000 kilometers with a 99.99% chance of experiencing no faults that do not result in breakdowns.
UR107	Loss of use of car due to equipment failure shall not exceed 1 day in every 2 years.
UR108	**3.2.2 Lifetime**
UR109	Users shall be able to use the car to its designed standard for 200,000 kilometers.
UR110	**3.2.3 Security**
UR111	Only the authorized user shall be able to start and drive away the vehicle.
UR112	**3.2.4 Accessories**
UR113	A warning triangle shall be supplied with the vehicle
UR114	A first aid kit shall be supplied with the vehicle
UR115	**3.2.5 Fuel input**
UR116	The car shall be fully compatible with the fuel input mechanism provided by the outlets where fuel will be acquired.
UR117	The user shall be able to completely refuel in less than 4 minutes or at the maximum time as defined by the fuel refill mechanism.
UR124	**3.2.6 Company constraints**
UR125	There will be an automatic break point at the end of year 2 when all research, early development and early prototyping will be assessed.
UR126	Risk analysis techniques will be used.
UR127	The research and development for the full project shall not exceed £25 million at today's prices

Username: Bill Young | Exclusive edit mode

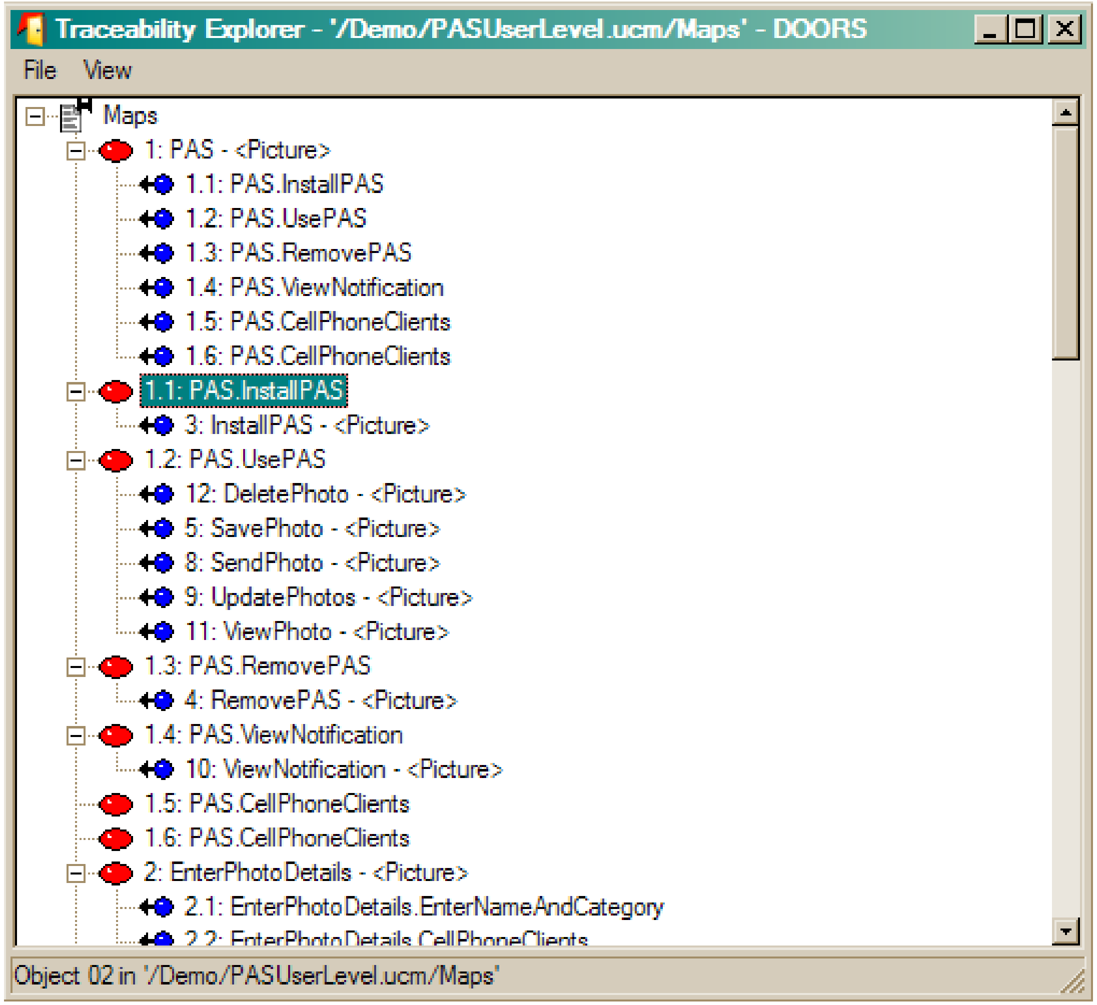
DOORS – Creating and Accessing Links


Formal module '/Demo/Use Cases' current 0.0 - DOORS
File Edit View Insert Link Analysis Table Tools User UCM Analyst Help
Standard view
All levels
ID
/Demo/PASUserLevel.ucm/Components
/Demo/PASUserLevel.ucm/Maps
/Demo/PASUserLevel.ucm/Responsibilities
14: EnterPhotoDetails: <Picture>
15: EnterPhotoDetails.EnterName
16: EnterPhotoDetails.CellPhoneClients
32: SavePhotoCommon: <Picture>
33: SavePhotoCommon.CheckFormatAndSize
34: SavePhotoCommon.SavePhoto
35: SavePhotoCommon.SaveDateNameCate...
36: SavePhotoCommon.PAS
51: UpdatePhotos: <Picture>
52: UpdatePhotos.SavePhotoCommon
53: UpdatePhotos.EnterPhotoDetails
54: UpdatePhotos.CellPhoneClients
55: UpdatePhotos.PAS
Username: damyot
Exclusive edit mode

- Use Cases
  - 1 Photo Archival System (PAS) fc
    - This document contains the
    - 1.1 Use Cases
      - Install PAS The CPC sele
      - Save photo The CPC sel
      - Update PAS with all loca
      - Change photo attributes
      - View photo See Select F
      - Send photo See Select F
      - Remove PAS The CPC s
    - 1.2 Common Use Cases

ID	
3	**1.1 Use Cases**
4	Install PAS The CPC selects "Install PAS" from the installation menu of the cell phone. The system installs the PAS software.
5	Save photo The CPC selects a photo stored locally on the cell phone. See Enter Photo Details (Common). See Save Photo (Common).
6	Update PAS with all local photos The CPC selects "Update Photos" from the PAS menu. l Photo De
7	Change p See Sele ommon). The system saves the new name an
8	View pho See Sele hoto to the CPC.
10	Send pho See Sele :s to who to send the photo. The system sends a r who received the notification selects the notificati
11	Remove The CPC nu of the cell phone. The system removes the PAS CPC from PAS.
12	**1.2 Common Use Cases**
14	Enter Photo Details (Common) The CPC enters a name for the photo. The CPC selects one of the following categories: portrait, landscape, animals, architecture, miscellaneous.
13	Save Photo (Common) The system checks whether the format of the photo conforms to Industry Standard "Photo-enabled Cell Phone 123.01". The system checks whether the size of the photo is less than or equal to 5MB. If the photo passes both checks, the system saves the photo, the name of photo, the category of the photo, and the current date in PAS. Otherwise, the system displays an error message for the CPC.
15	Select Photo (COmmon) The CPC selects a photo from a list of photo names or thumbnails. The CPC may switch between the list of photo names and the thumbnails.

DOORS – Exploring Traceability Links

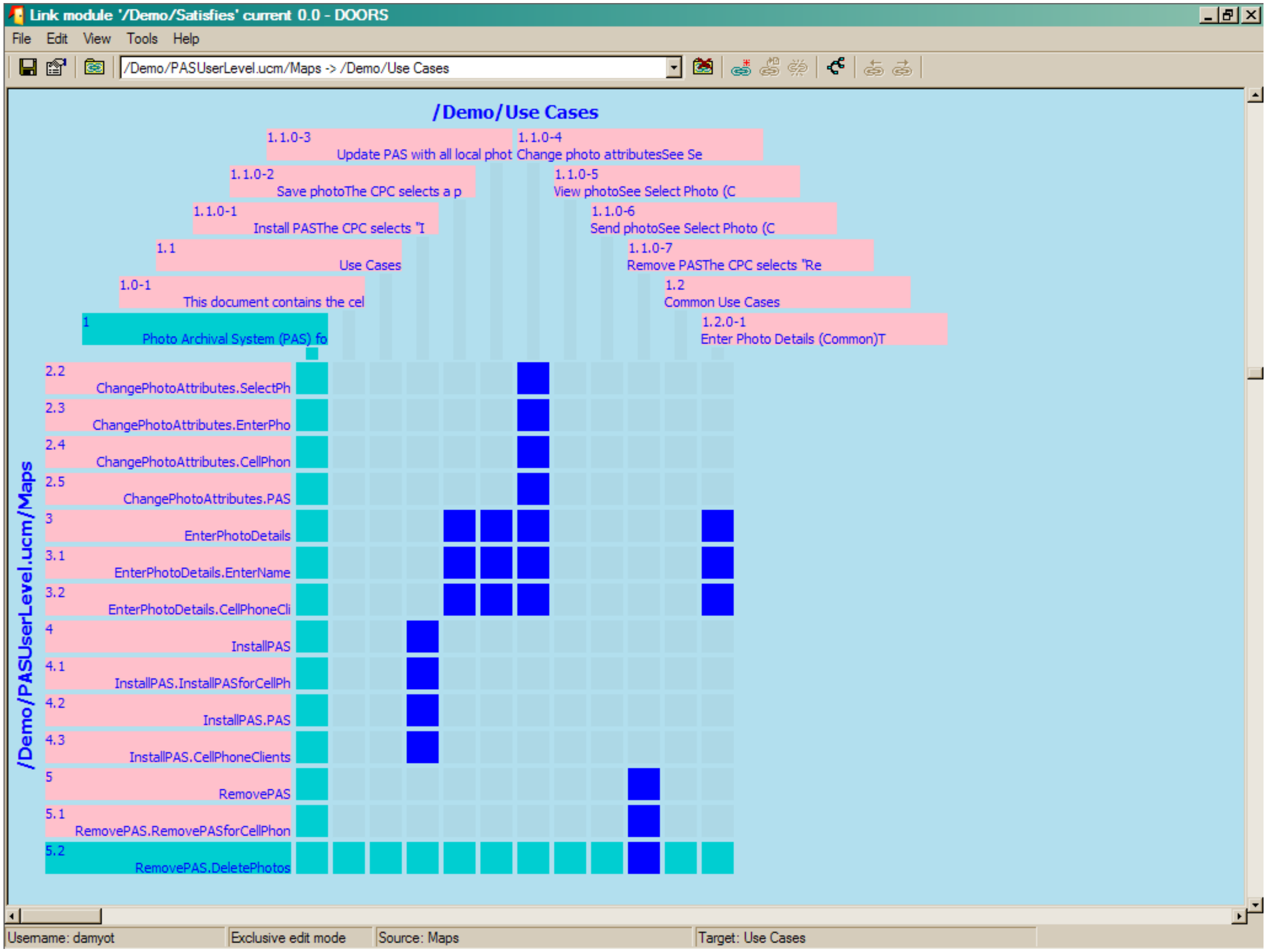


Traceability Explorer - '/Demo/PASUserLevel.ucm/Maps' - DOORS

```
Maps
    1: PAS - <Picture>
        1.1: PAS.InstallPAS
        1.2: PAS.UsePAS
        1.3: PAS.RemovePAS
        1.4: PAS.ViewNotification
        1.5: PAS.CellPhoneClients
        1.6: PAS.CellPhoneClients
    1.1: PAS.InstallPAS
        3: InstallPAS - <Picture>
    1.2: PAS.UsePAS
        12: DeletePhoto - <Picture>
        5: SavePhoto - <Picture>
        8: SendPhoto - <Picture>
        9: UpdatePhotos - <Picture>
        11: ViewPhoto - <Picture>
    1.3: PAS.RemovePAS
        4: RemovePAS - <Picture>
    1.4: PAS.ViewNotification
        10: ViewNotification - <Picture>
    1.5: PAS.CellPhoneClients
    1.6: PAS.CellPhoneClients
    2: EnterPhotoDetails - <Picture>
        2.1: EnterPhotoDetails.EnterNameAndCategory
        2.2: EnterPhotoDetails.CellPhoneClients
```

Object 02 in '/Demo/PASUserLevel.ucm/Maps'

DOORS – Link Matrix View



Link module '/Demo/Satisfies' current 0.0 - DOORS
File Edit View Tools Help

/Demo/PASUserLevel.ucm/Maps -> /Demo/Use Cases

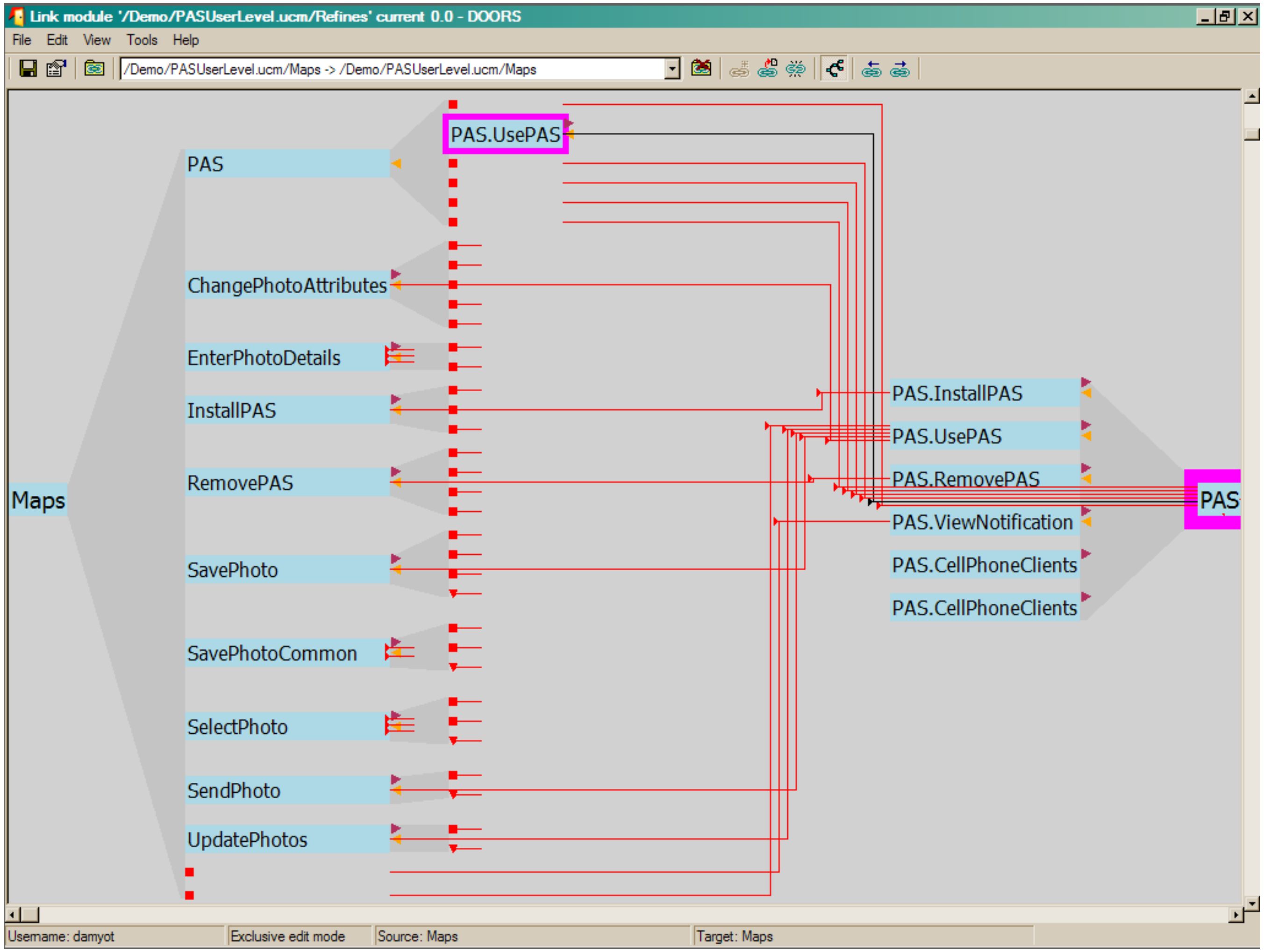
/Demo/Use Cases

/Demo/PASUserLevel.ucm/Maps

	1 Photo Archival System (PAS) fo	1.0-1 This document contains the cel	1.1 Use Cases	1.1.0-1 Install PASThe CPC selects "I	1.1.0-2 Save photoThe CPC selects a p	1.1.0-3 Update PAS with all local phot	1.1.0-4 Change photo attributesSee Se	1.1.0-5 View photoSee Select Photo (C	1.1.0-6 Send photoSee Select Photo (C	1.1.0-7 Remove PASThe CPC selects "Re	1.2 Common Use Cases	1.2.0-1 Enter Photo Details (Common)T
2.2 ChangePhotoAttributes.SelectPh							■					
2.3 ChangePhotoAttributes.EnterPho							■					
2.4 ChangePhotoAttributes.CellPhon							■					
2.5 ChangePhotoAttributes.PAS							■					
3 EnterPhotoDetails					■	■	■					■
3.1 EnterPhotoDetails.EnterName					■	■	■					■
3.2 EnterPhotoDetails.CellPhoneCli					■	■	■					■
4 InstallPAS				■								
4.1 InstallPAS.InstallPASforCellPh				■								
4.2 InstallPAS.PAS				■								
4.3 InstallPAS.CellPhoneClients				■								
5 RemovePAS										■		
5.1 RemovePAS.RemovePASforCellPhon										■		
5.2 RemovePAS.DeletePhotos										■		

Username: damyot | Exclusive edit mode | Source: Maps | Target: Use Cases

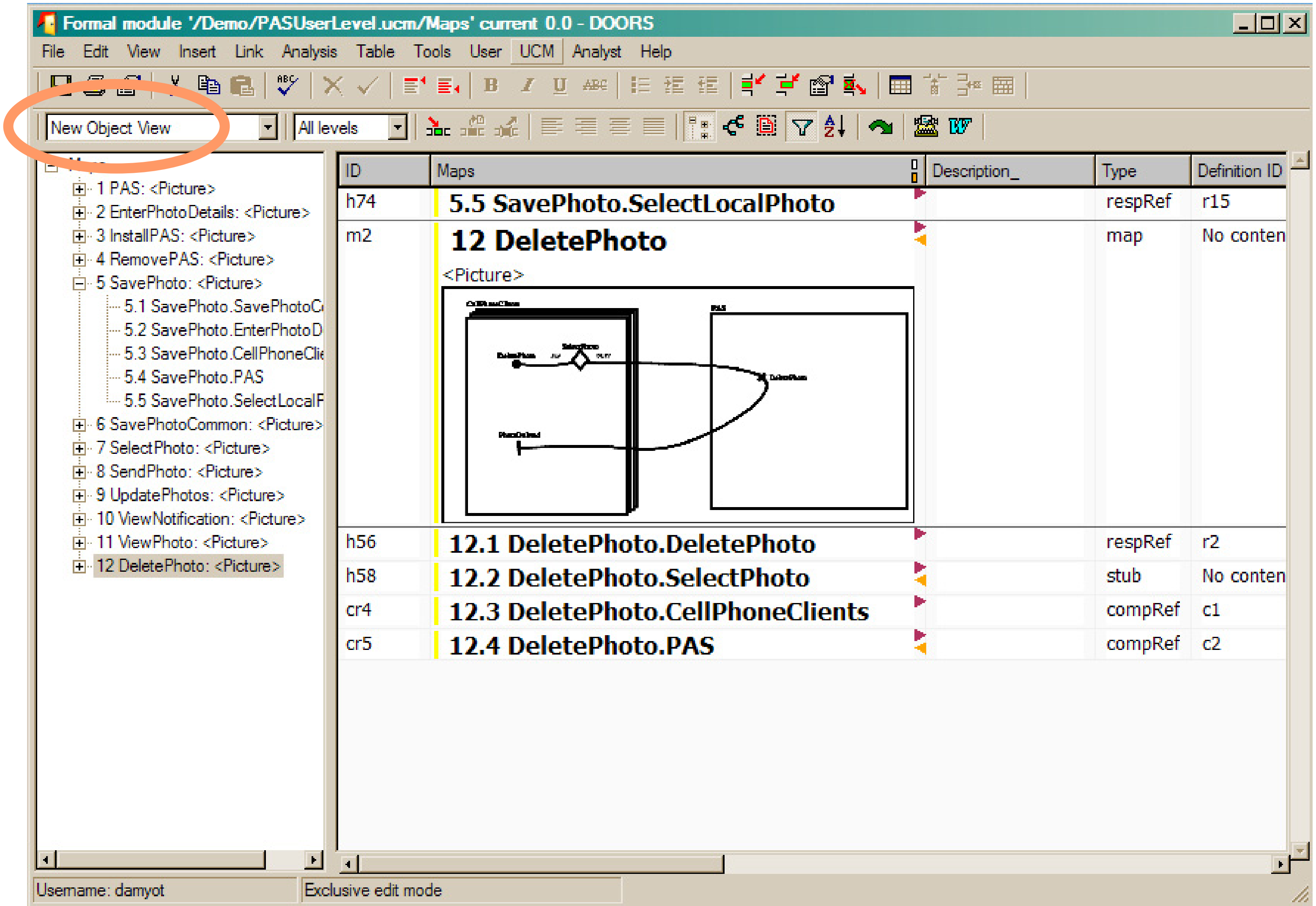
DOORS – Hierarchical Link View


Link module '/Demo/PASUserLevel.ucm/Refines' current 0.0 - DOORS

/Demo/PASUserLevel.ucm/Maps -> /Demo/PASUserLevel.ucm/Maps

Username: damyot | Exclusive edit mode | Source: Maps | Target: Maps

DOORS – Filtering View (Query on Attributes)



Formal module '/Demo/PASUserLevel.ucm/Maps' current 0.0 - DOORS

File Edit View Insert Link Analysis Table Tools User UCM Analyst Help
New Object View
All levels

- Maps
  - 1 PAS: <Picture>
  - 2 EnterPhotoDetails: <Picture>
  - 3 InstallPAS: <Picture>
  - 4 RemovePAS: <Picture>
  - 5 SavePhoto: <Picture>
    - 5.1 SavePhoto.SavePhotoC
    - 5.2 SavePhoto.EnterPhotoD
    - 5.3 SavePhoto.CellPhoneClie
    - 5.4 SavePhoto.PAS
    - 5.5 SavePhoto.SelectLocalF
  - 6 SavePhotoCommon: <Picture>
  - 7 SelectPhoto: <Picture>
  - 8 SendPhoto: <Picture>
  - 9 UpdatePhotos: <Picture>
  - 10 ViewNotification: <Picture>
  - 11 ViewPhoto: <Picture>
  - 12 DeletePhoto: <Picture>

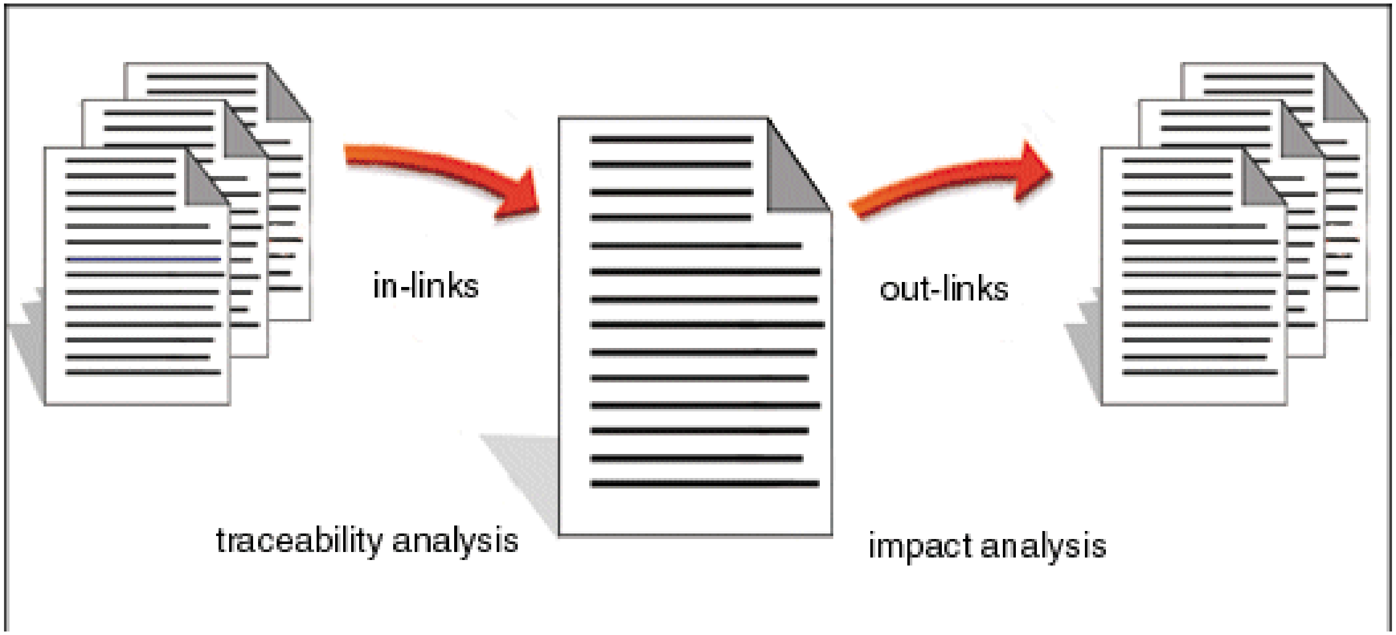
ID	Maps	Description_	Type	Definition ID
h74	**5.5 SavePhoto.SelectLocalPhoto**		respRef	r15
m2	**12 DeletePhoto** <Picture>		map	No conten
h56	**12.1 DeletePhoto.DeletePhoto**		respRef	r2
h58	**12.2 DeletePhoto.SelectPhoto**		stub	No conten
cr4	**12.3 DeletePhoto.CellPhoneClients**		compRef	c1
cr5	**12.4 DeletePhoto.PAS**		compRef	c2

Username: damyot

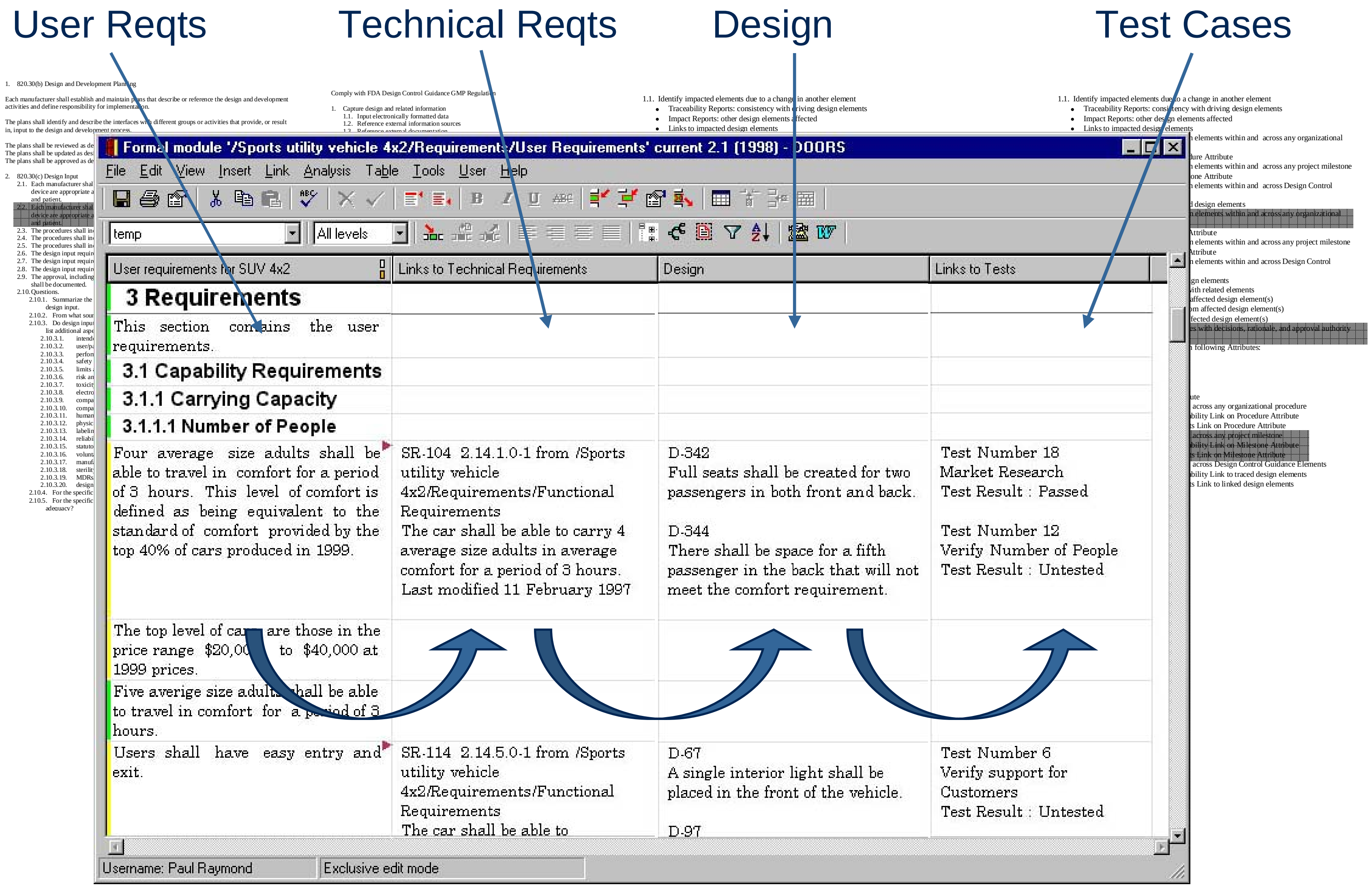
Exclusive edit mode

DOORS – Types of Analysis

- Impact Analysis
  - Analyze out-links (which objects in other modules are affected when this module is changed)
- Traceability Analysis
  - Analyze in-links (changes in these objects will affect this module)
- May involve multiple levels of objects/documents



DOORS – Multi-Module Traceability



User Reqts

Technical Reqts

Design

Test Cases

1. 820.30(b) Design and Development Planning

Each manufacturer shall establish and maintain plans that describe or reference the design and development activities and define responsibility for implementation.

The plans shall identify and describe the interfaces with different groups or activities that provide, or result in, input to the design and development process.

The plans shall be reviewed as de
The plans shall be updated as desi
The plans shall be approved as de

2. 820.30(c) Design Input
   2.1. Each manufacturer shal device are appropriate a and patient.
   2.2. Each manufacturer shal device are appropriate a and patient.
   2.3. The procedures shall in
   2.4. The procedures shall in
   2.5. The procedures shall in
   2.6. The design input requir
   2.7. The design input requir
   2.8. The design input requir
   2.9. The approval, including shall be documented.
   2.10. Questions.
      2.10.1. Summarize the design input.
      2.10.2. From what sour
      2.10.3. Do design input list additional aspe
         2.10.3.1. intend
         2.10.3.2. user/pa
         2.10.3.3. perforn
         2.10.3.4. safety
         2.10.3.5. limits
         2.10.3.6. risk an
         2.10.3.7. toxicity
         2.10.3.8. electro
         2.10.3.9. compa
         2.10.3.10. compa
         2.10.3.11. human
         2.10.3.12. physic
         2.10.3.13. labelin
         2.10.3.14. reliabil
         2.10.3.15. statuto
         2.10.3.16. volunta
         2.10.3.17. manufa
         2.10.3.18. sterility
         2.10.3.19. MDRs
         2.10.3.20. design
      2.10.4. For the specific
      2.10.5. For the specific adequacy?

Comply with FDA Design Control Guidance GMP Regulation

1. Capture design and related information
   1.1. Input electronically formatted data
   1.2. Reference external information sources
   1.3. Reference external documentation

1.1. Identify impacted elements due to a change in another element
- Traceability Reports: consistency with driving design elements
- Impact Reports: other design elements affected
- Links to impacted design elements

1.1. Identify impacted elements due to a change in another element
- Traceability Reports: consistency with driving design elements
- Impact Reports: other design elements affected
- Links to impacted design elements

n elements within and across any organizational
dure Attribute
n elements within and across any project milestone
one Attribute
n elements within and across Design Control
d design elements
n elements within and across any organizational
Attribute
n elements within and across any project milestone
Attribute
n elements within and across Design Control
gn elements
with related elements
affected design element(s)
om affected design element(s)
fected design element(s)
es with decisions, rationale, and approval authority
n following Attributes:
ute
across any organizational procedure
bility Link on Procedure Attribute
ts Link on Procedure Attribute
across any project milestone
bility Link on Milestone Attribute
ts Link on Milestone Attribute
across Design Control Guidance Elements
bility Link to traced design elements
ts Link to linked design elements

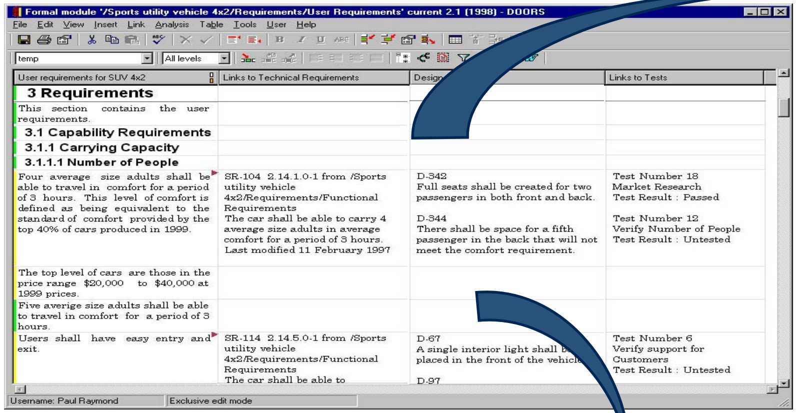
Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' current 2.1 (1998) - DOORS

User requirements for SUV 4x2	Links to Technical Requirements	Design	Links to Tests
3 Requirements			
This section contains the user requirements.			
3.1 Capability Requirements			
3.1.1 Carrying Capacity			
3.1.1.1 Number of People			
Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 40% of cars produced in 1999.	SR-104 2.14.1.0-1 from /Sports utility vehicle 4x2/Requirements/Functional Requirements The car shall be able to carry 4 average size adults in average comfort for a period of 3 hours. Last modified 11 February 1997	D-342 Full seats shall be created for two passengers in both front and back. D-344 There shall be space for a fifth passenger in the back that will not meet the comfort requirement.	Test Number 18 Market Research Test Result : Passed Test Number 12 Verify Number of People Test Result : Untested
The top level of cars are those in the price range $20,000 to $40,000 at 1999 prices.			
Five averige size adults shall be able to travel in comfort for a period of 3 hours.			
Users shall have easy entry and exit.	SR-114 2.14.5.0-1 from /Sports utility vehicle 4x2/Requirements/Functional Requirements The car shall be able to	D-67 A single interior light shall be placed in the front of the vehicle. D-97	Test Number 6 Verify support for Customers Test Result : Untested

Username: Paul Raymond | Exclusive edit mode

DOORS – Traceability and Software Artefacts

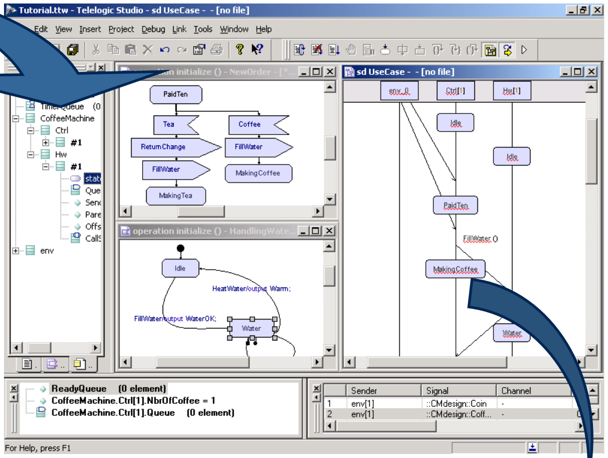
DOORS:
Requirements Management & Traceability


Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' current 2.1 (1998) - DOORS
File Edit View Insert Link Analysis Table Tools User Help
temp
All levels

User requirements for SUV 4x2	Links to Technical Requirements	Design	Links to Tests
3 Requirements			
This section contains the user requirements.			
3.1 Capability Requirements			
3.1.1 Carrying Capacity			
3.1.1.1 Number of People			
Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 40% of cars produced in 1999.	SR-104 2.14.1.0-1 from /Sports utility vehicle 4x2/Requirements/Functional Requirements The car shall be able to carry 4 average size adults in average comfort for a period of 3 hours. Last modified 11 February 1997	D-342 Full seats shall be created for two passengers in both front and back. D-344 There shall be space for a fifth passenger in the back that will not meet the comfort requirement.	Test Number 18 Market Research Test Result : Passed Test Number 12 Verify Number of People Test Result : Untested
The top level of cars are those in the price range $20,000 to $40,000 at 1999 prices.			
Five averige size adults shall be able to travel in comfort for a period of 3 hours.			
Users shall have easy entry and exit.	SR-114 2.14.5.0-1 from /Sports utility vehicle 4x2/Requirements/Functional Requirements The car shall be able to	D-67 A single interior light shall be placed in the front of the vehicle D-97	Test Number 6 Verify support for Customers Test Result : Untested

Username: Paul Raymond | Exclusive edit mode

TAU/Architect & TAU/Developer:
System Modeling & Code Generation

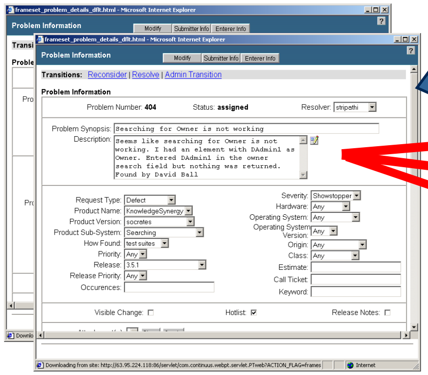


ReadyQueue (0 element)
CoffeeMachine.Ctrl[1].NbrOfCoffee = 1
CoffeeMachine.Ctrl[1].Queue (0 element)

	Sender	Signal	Channel
1	env[1]	::CMdesign::Coin	-
2	env[1]	::CMdesign::Coff...	-

For Help, press F1

SYNERGY/Change:
Work Orders


frameset_problem_details_dflt.html - Microsoft Internet Explorer
Problem Information
Modify
Submitter Info
Enterer Info

Transitions: Reconsider | Resolve | Admin Transition

Problem Information

Problem Number: **404** Status: **assigned** Resolver: stripathi

Problem Synopsis: Searching for Owner is not working

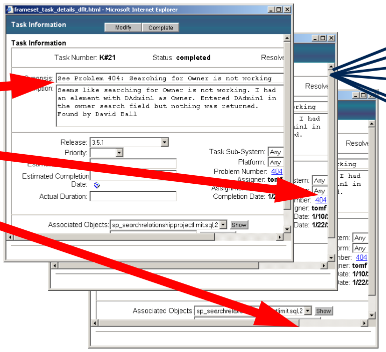
Description: Seems like searching for Owner is not working. I had an element with DAdmin1 as Owner. Entered DAdmin1 in the owner search field but nothing was returned. Found by David Ball

Request Type:	Defect	Severity:	Showstopper
Product Name:	KnowledgeSynergy	Hardware:	Any
Product Version:	socrates	Operating System:	Any
Product Sub-System:	Searching	Operating System Version:	Any
How Found:	test suites	Origin:	Any
Priority:	Any	Class:	Any
Release:	3.5.1	Estimate:	
Release Priority:	Any	Call Ticket:	
Occurences:		Keyword:	

Visible Change: □ Hotlist: ☑ Release Notes: □

Downloading from site: http://63.95.224.118:86/servlet/com.continuus.webpt.servlet.PTweb?ACTION_FLAG=frames Internet

SYNERGY/CM:
Engineering Tasks


frameset_task_details_dflt.html - Microsoft Internet Explorer
Task Information
Modify
Complete

Task Information

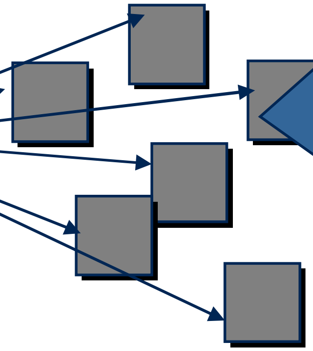
Task Number: **K#21** Status: **completed** Resolve

Synopsis: See Problem 404: Searching for Owner is not working

Description: Seems like searching for Owner is not working. I had an element with DAdmin1 as Owner. Entered DAdmin1 in the owner search field but nothing was returned. Found by David Ball

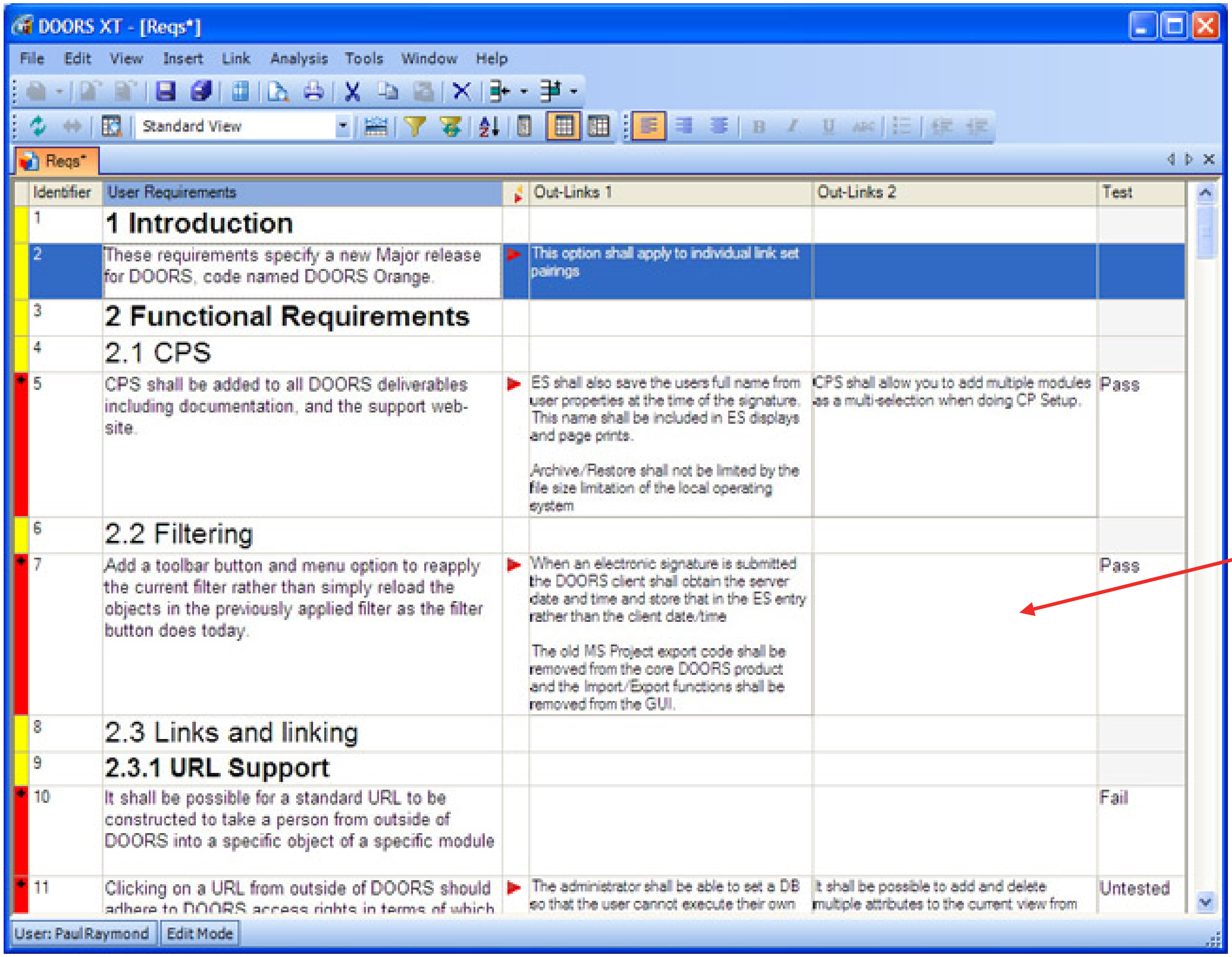
Release:	3.5.1	Task Sub-System:	Any
Priority:		Platform:	Any
		Problem Number:	404
Estimated Completion Date:		Assigner:	tomf
Actual Duration:		Completion Date:	1/2

Associated Objects: sp_searchrelationshipprojectlimit.sql,2 Show



ActiveCM:
Controlled Code Modules

DOORS – Analysis with Wizard

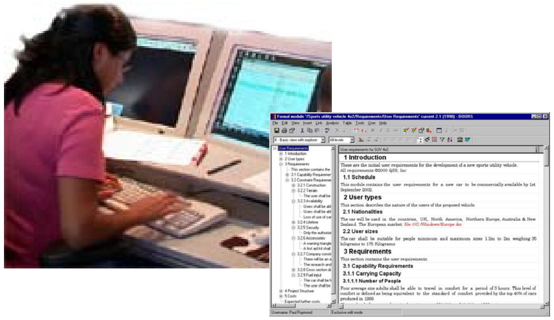


Identifier	User Requirements	Out-Links 1	Out-Links 2	Test
1	1 Introduction			
2	These requirements specify a new Major release for DOORS, code named DOORS Orange.	This option shall apply to individual link set pairings		
3	2 Functional Requirements			
4	2.1 CPS			
5	CPS shall be added to all DOORS deliverables including documentation, and the support web-site.	ES shall also save the users full name from user properties at the time of the signature. This name shall be included in ES displays and page prints. Archive/Restore shall not be limited by the file size limitation of the local operating system	CPS shall allow you to add multiple modules as a multi-selection when doing CP Setup.	Pass
6	2.2 Filtering			
7	Add a toolbar button and menu option to reapply the current filter rather than simply reload the objects in the previously applied filter as the filter button does today.	When an electronic signature is submitted the DOORS client shall obtain the server date and time and store that in the ES entry rather than the client date/time The old MS Project export code shall be removed from the core DOORS product and the Import/Export functions shall be removed from the GUI.		Pass
8	2.3 Links and linking			
9	2.3.1 URL Support			
10	It shall be possible for a standard URL to be constructed to take a person from outside of DOORS into a specific object of a specific module			Fail
11	Clicking on a URL from outside of DOORS should adhere to DOORS access rights in terms of which	The administrator shall be able to set a DB so that the user cannot execute their own	It shall be possible to add and delete multiple attributes to the current view from	Untested

Orphans indicate missing links

What are Suspect Links?

If documents are linked …



Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' current 2.1 (1998) - DOORS

User requirements for SUV 4x2

1 Introduction

These are the initial user requirements for the development of a new sports utility vehicle. All requirements ©2000 QSS, Inc

1.1 Schedule

This module contains the user requirements for a new car to be commercially available by 1st September 2002.

2 User types

This section describes the nature of the users of the proposed vehicle.

2.1 Nationalities

The car will be used in the countries, UK, North America, Northern Europe, Australia & New Zealand. The European market: file:///C:/Windows/Europe.doc

2.2 User sizes

The car shall be suitable for people minimum and maximum sizes 1.2m to 2m weighing 35 kilograms to 175 Kilograms.

3 Requirements

This section contains the user requirements.

3.1 Capability Requirements

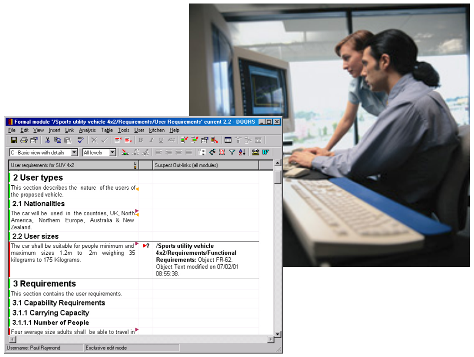
3.1.1 Carrying Capacity

3.1.1.1 Number of People

Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 40% of cars produced in 1999.

Username: Paul Raymond | Exclusive edit mode

▶?


File Edit View Insert Link Analysis Table Tools User kitchen Help
C - Basic view with details
All levels

Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' current 2.2 - DOORS

User requirements for SUV 4x2		Suspect Out-links (all modules)
2 User types		
This section describes the nature of the users of the proposed vehicle.		
2.1 Nationalities		
The car will be used in the countries, UK, North America, Northern Europe, Australia & New Zealand.		
2.2 User sizes		
The car shall be suitable for people minimum and maximum sizes 1.2m to 2m weighing 35 kilograms to 175 Kilograms.	▶?	**/Sports utility vehicle 4x2/Requirements/Functional Requirements:** Object FR-62. Object Text modified on 07/02/01 08:55:38.
3 Requirements		
This section contains the user requirements.		
3.1 Capability Requirements		
3.1.1 Carrying Capacity		
3.1.1.1 Number of People		
Four average size adults shall be able to travel in		

Username: Paul Raymond | Exclusive edit mode

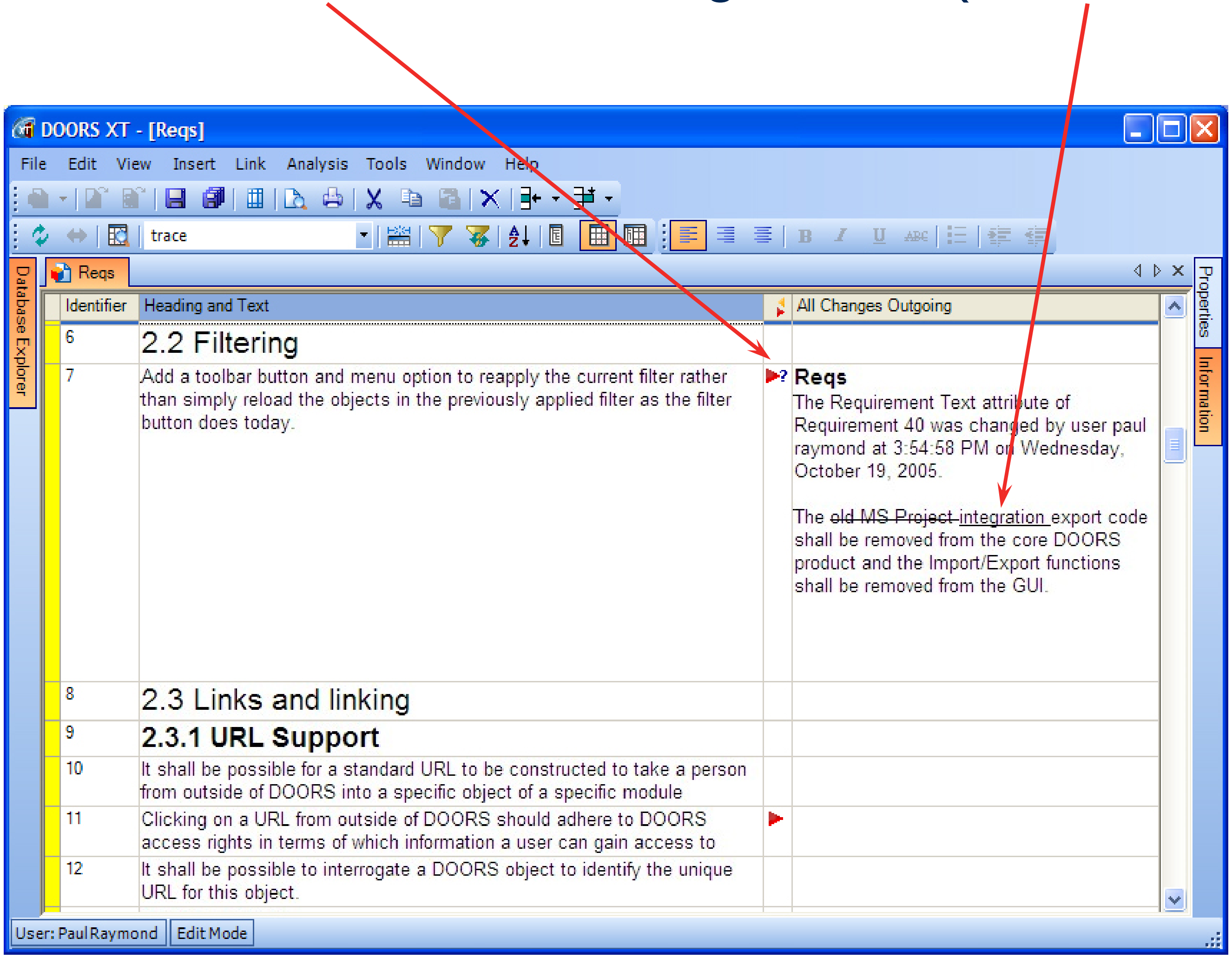
… a change by this user here…

… shows up as a warning flag to this user here.

- Proactively know when changes effect your requirements
- Suspect link indicates that element may have been affected by a change
- Help ensure ripple effects of changes are considered

Suspect Links

- Visible as indicators or with change notes (added/deleted)



Identifier	Heading and Text		All Changes Outgoing
6	2.2 Filtering		
7	Add a toolbar button and menu option to reapply the current filter rather than simply reload the objects in the previously applied filter as the filter button does today.	▶?	**Reqs** The Requirement Text attribute of Requirement 40 was changed by user paul raymond at 3:54:58 PM on Wednesday, October 19, 2005. The ~~old MS Project~~ integration export code shall be removed from the core DOORS product and the Import/Export functions shall be removed from the GUI.
8	2.3 Links and linking		
9	**2.3.1 URL Support**		
10	It shall be possible for a standard URL to be constructed to take a person from outside of DOORS into a specific object of a specific module		
11	Clicking on a URL from outside of DOORS should adhere to DOORS access rights in terms of which information a user can gain access to	▶	
12	It shall be possible to interrogate a DOORS object to identify the unique URL for this object.		

Traceability Planning

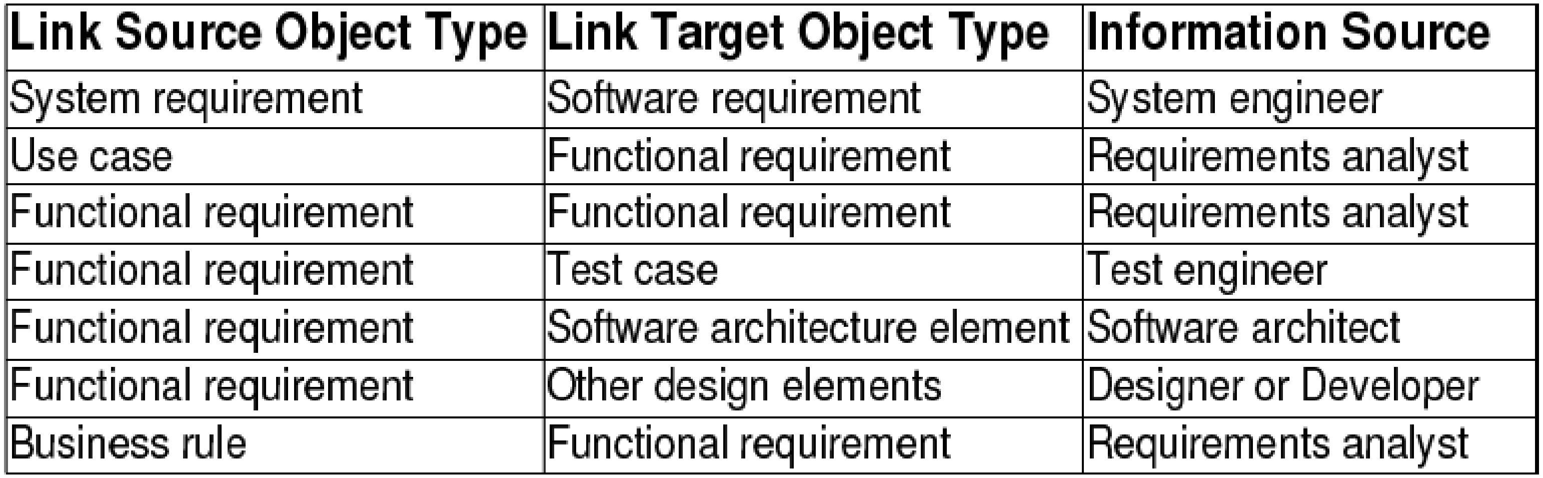
- Planning traceability? Yes, just like any other project!
  - Who are the stakeholders?
  - What are the needs (analysis, reports…)?
    - Useful, measurable, feasible objectives
  - Definition of links and attributes
    - Can some be inferred automatically?
  - Process (who collects and when to collect traceability information)
    - Roles, access
    - Data/link input and updates
    - Exceptional situations (e.g., lack of time)
  - Representations, queries, tools
  - …
  - Traceability policies define what and how traceability information should be maintained

Factors to Consider during Planning (1)

- Number of requirements
  - The greater the number of requirements, the more the need for formal traceability policies
- Expected system lifetime
  - More comprehensive traceability policies should be defined for systems which have a long lifetime
- Maturity level of organization
  - Detailed traceability policies are more likely to be implemented and used properly in a cost-effective way in organizations which have a higher level of process maturity
- Size of project and team
  - The larger the project or team, the greater the need for formal traceability policies

Factors to Consider during Planning (2)

- Type of system
  - Critical systems such as hard real-time control systems or safety-critical systems need more comprehensive traceability policies than non-critical systems
- Additional constraints from customer
  - E.g., compliance to military standard
- Traceability links should be defined by whoever has the appropriate information available



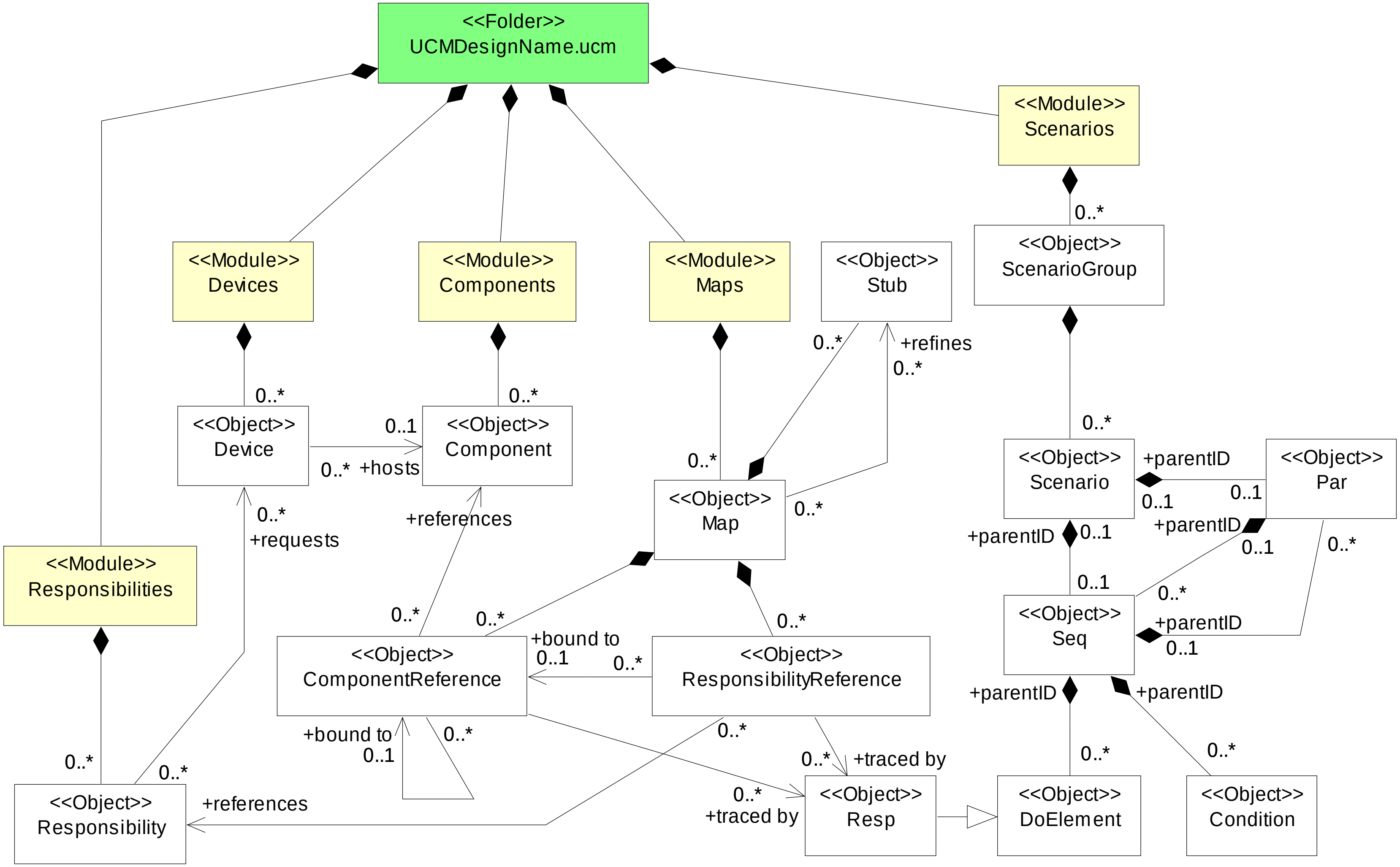
Link Source Object Type	Link Target Object Type	Information Source
System requirement	Software requirement	System engineer
Use case	Functional requirement	Requirements analyst
Functional requirement	Functional requirement	Requirements analyst
Functional requirement	Test case	Test engineer
Functional requirement	Software architecture element	Software architect
Functional requirement	Other design elements	Designer or Developer
Business rule	Functional requirement	Requirements analyst

Modeling Traceability

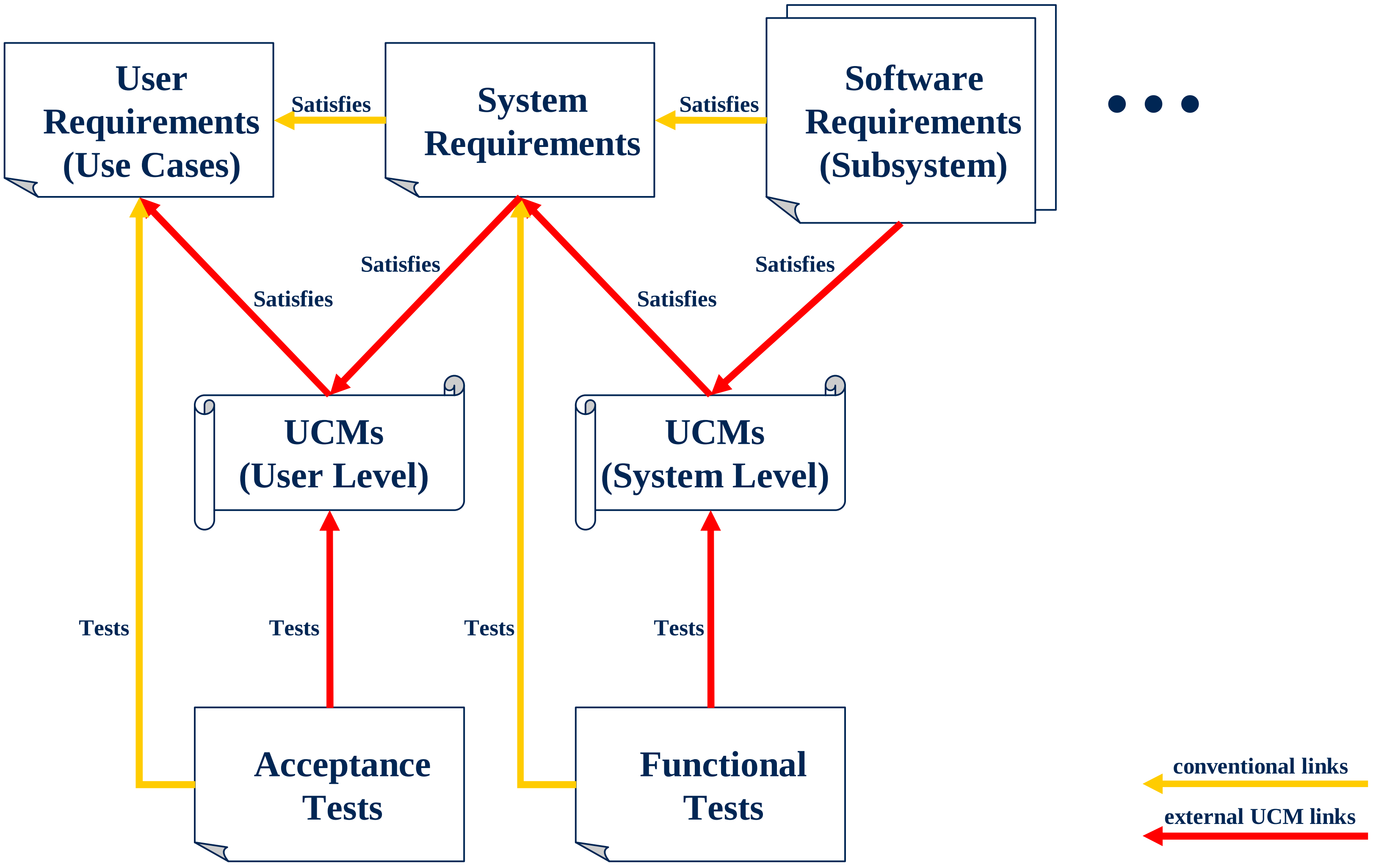
- The types of links to use (and their attributes) must be defined for different types of requirements
  - It is a design problem!
- May be modeled with a UML class diagram (metamodel)
  - Object types (classes)
  - Object attributes (attributes)
  - Structure of folders, modules, objects
    - Stereotypes, composition…
  - Link types (associations)
    - Satisfies, tests, refines, contains, contributes to, threatens, justifies…
  - Link attributes (association classes)
  - …

Example – UCM Models Imported in DOORS

- Associations describe internal links

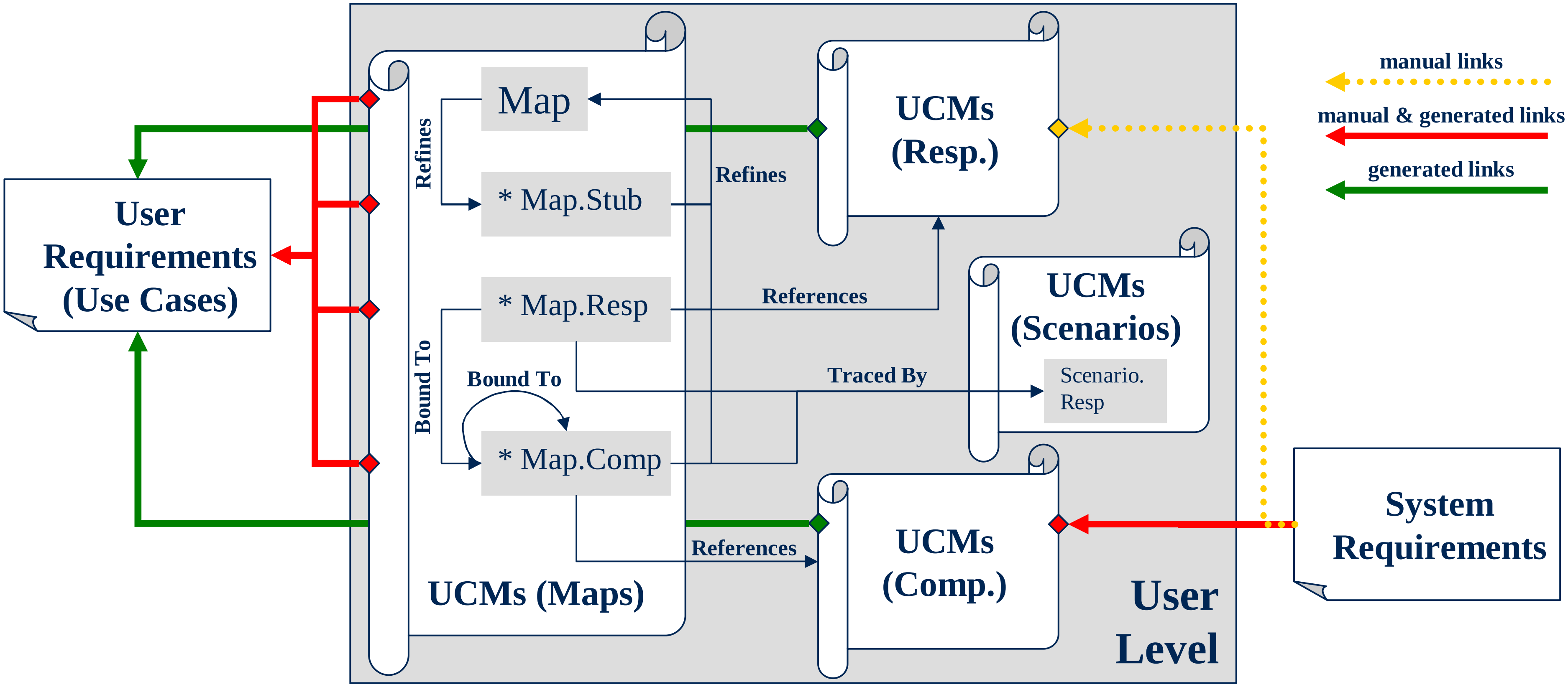


Example – UCM External Links in DOORS



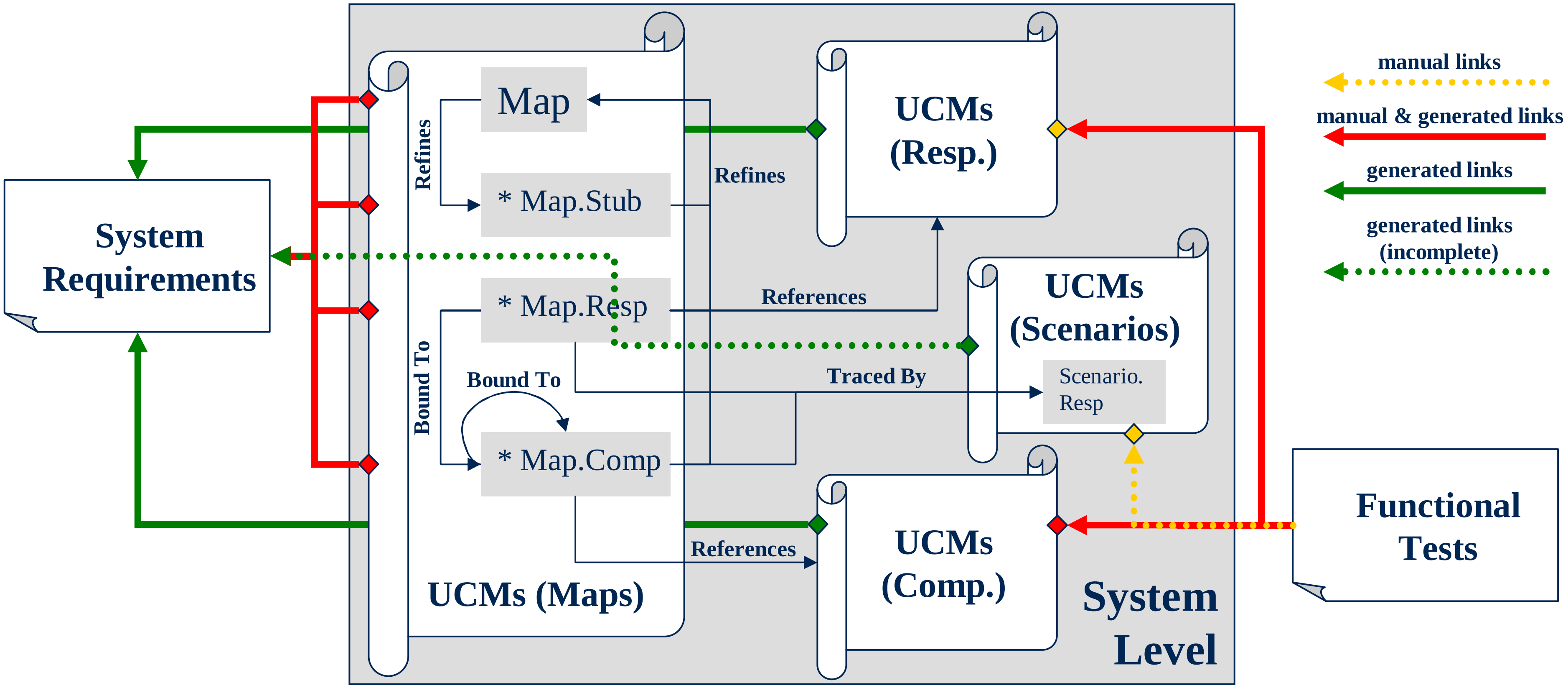
Example – Automatic Link Generation (1)

- Important to minimize the manual effort for link creation



- From system requirements to user-level UCMs to user reqs.

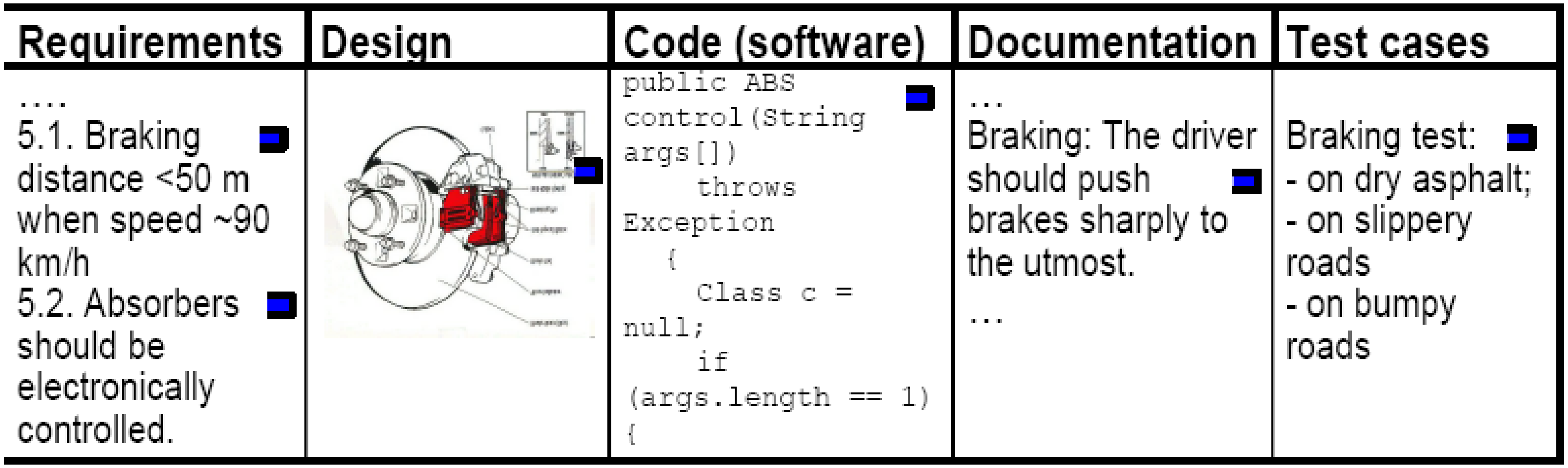
Example – Automatic Link Generation (2)



- From tests to system-level UCMs to system requirements

Types of Traceability Links

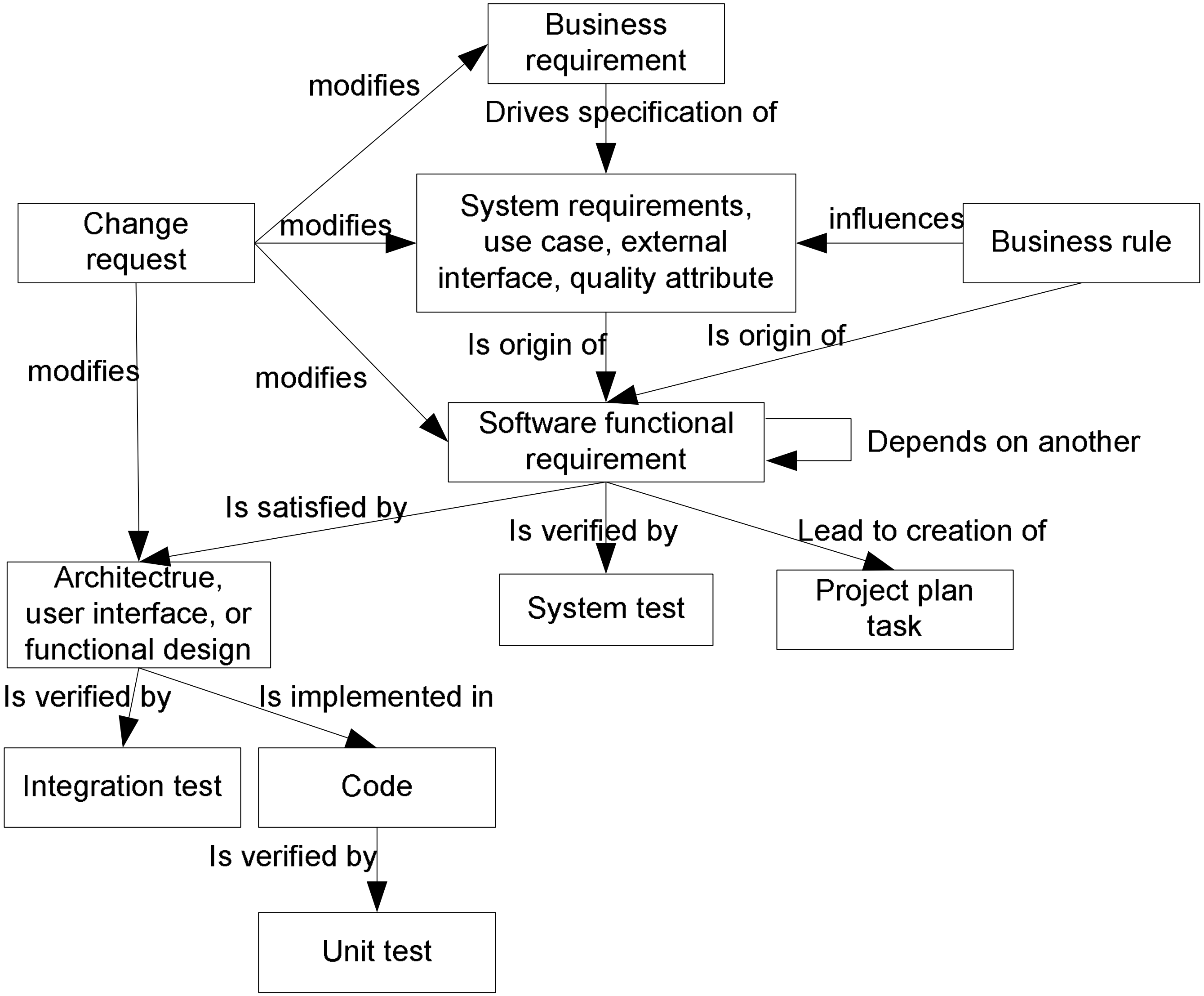
- Note the types of links in the previous examples, as well as the types of objects they relate
  - Satisfies, Tests
  - Refines, References, Contains...
- Others could be created
  - Contributes, Contradicts, Justifies, Depends on...



Requirements	Design	Code (software)	Documentation	Test cases

….
5.1. Braking distance <50 m when speed ~90 km/h
5.2. Absorbers should be electronically controlled.
		public ABS control(String args[]) throws Exception
		{
		Class c = null;
		if (args.length == 1)
		{
			…
			Braking: The driver should push brakes sharply to the utmost.
			…
				Braking test:
				- on dry asphalt;
				- on slippery roads
				- on bumpy roads

Other Example of Traceability Links



Cardinality of Traceability Links

- Depending on the traceability information, the cardinality of traceability links may be
  - One-to-one
    - E.g., one design element to one code module
  - One-to-many
    - E.g., one functional requirement verified by multiple test cases
  - Many-to-many
    - E.g., a use case may lead to multiple functional requirement, and a functional requirement may be common to several use cases

Advice for DOORS Links

- Direction of links
  - From the most concrete to the most abstract
  - To avoid access rights issues
  - To make use of the integrated analysis routines of DOORS

- Link Modules
  - One module per type of link
  - NEVER use default module (should not even be offered)
  - To avoid maintenance problems
  - Specific types facilitate analysis and filtering

Baselines

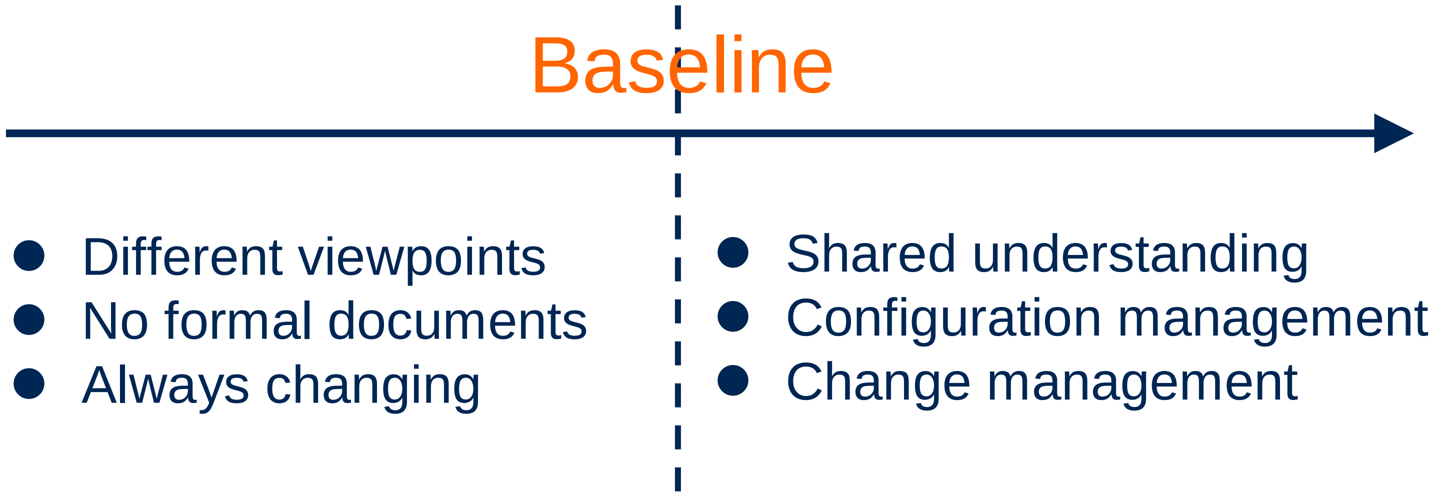
Baseline

- Non-modifiable (read-only) version of a document
  - Describes a moment in time
  - May include multiple documents at the same time
- Enables document comparison and management
- Comes with a change history for the document
  - Information on objects, attributes, and links created, deleted, or edited since the creation of the baseline
  - Often also contains information on user sessions (when the document was opened, by whom…)
- Requires access control

- It is advisable to establish a baseline for a new document that is imported into the document management system
  - In order not to lose any changes

Baseline for Requirements

- Represents the set of functional and non-functional requirements that the development team has committed to implement in a specific release
- Before going into the baseline, the requirements should be reviewed and approved by stakeholders
- Once in the baseline, all changes should follow a defined change control process



Baseline

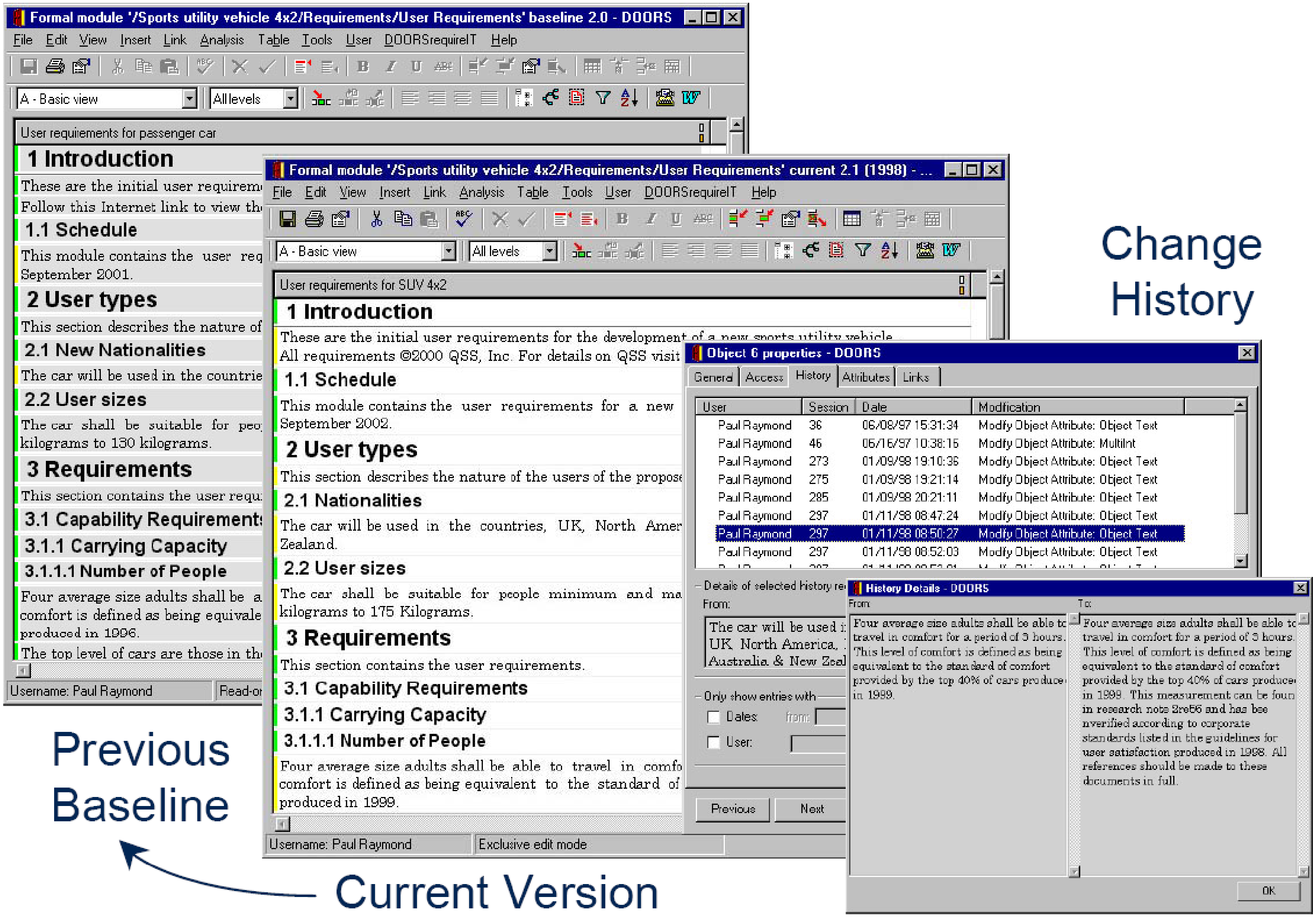
- Different viewpoints
- No formal documents
- Always changing

- Shared understanding
- Configuration management
- Change management

Baseline Usage

- Baselines may be
  - Created
    - Complete image of requirements state at a given time
  - Deleted
  - Visualized
    - Possibility to go back
  - Compared
    - To see changes since a certain time
  - Copied
  - Signed
    - For authorization, contract

DOORS – Baseline Compare



Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' baseline 2.0 - DOORS

File Edit View Insert Link Analysis Table Tools User DOORSrequireIT Help
A - Basic view
All levels

User requiements for passenger car

1 Introduction

These are the initial user requirem

Follow this Internet link to view th

1.1 Schedule

This module contains the user req September 2001.

2 User types

This section describes the nature of

2.1 New Nationalities

The car will be used in the countrie

2.2 User sizes

The car shall be suitable for peo kilograms to 130 kilograms.

3 Requirements

This section contains the user requ

3.1 Capability Requirement

3.1.1 Carrying Capacity

3.1.1.1 Number of People

Four average size adults shall be a comfort is defined as being equivale produced in 1996.

The top level of cars are those in th

Username: Paul Raymond | Read-or

Formal module '/Sports utility vehicle 4x2/Requirements/User Requirements' current 2.1 (1998) - ...

File Edit View Insert Link Analysis Table Tools User DOORSrequireIT Help
A - Basic view
All levels

User requirements for SUV 4x2

1 Introduction

These are the initial user requirements for the development of a new sports utility vehicle

All requirements ©2000 QSS, Inc. For details on QSS visit

1.1 Schedule

This module contains the user requirements for a new September 2002.

2 User types

This section describes the nature of the users of the propose

2.1 Nationalities

The car will be used in the countries, UK, North Amer Zealand.

2.2 User sizes

The car shall be suitable for people minimum and ma kilograms to 175 Kilograms.

3 Requirements

This section contains the user requirements.

3.1 Capability Requirements

3.1.1 Carrying Capacity

3.1.1.1 Number of People

Four average size adults shall be able to travel in comfo comfort is defined as being equivalent to the standard of produced in 1999.

Username: Paul Raymond | Exclusive edit mode

Object 6 properties - DOORS

User	Session	Date	Modification
Paul Raymond	36	06/08/97 15:31:34	Modify Object Attribute: Object Text
Paul Raymond	46	06/16/97 10:38:16	Modify Object Attribute: MultiInt
Paul Raymond	273	01/09/98 19:10:36	Modify Object Attribute: Object Text
Paul Raymond	275	01/09/98 19:21:14	Modify Object Attribute: Object Text
Paul Raymond	285	01/09/98 20:21:11	Modify Object Attribute: Object Text
Paul Raymond	297	01/11/98 08:47:24	Modify Object Attribute: Object Text
Paul Raymond	297	01/11/98 08:50:27	Modify Object Attribute: Object Text
Paul Raymond	297	01/11/98 08:52:03	Modify Object Attribute: Object Text

Details of selected history re

From:

The car will be used i UK, North America, Australia & New Zeal

Only show entries with

□ Dates: from:

□ User:

History Details - DOORS

From:

Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 40% of cars produce in 1999.

To:

Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 40% of cars produce in 1999. This measurement can be foun in research note 2re56 and has bee nverified according to corporate standards listed in the guidelines for user satisfaction produced in 1998. All references should be made to these documents in full.

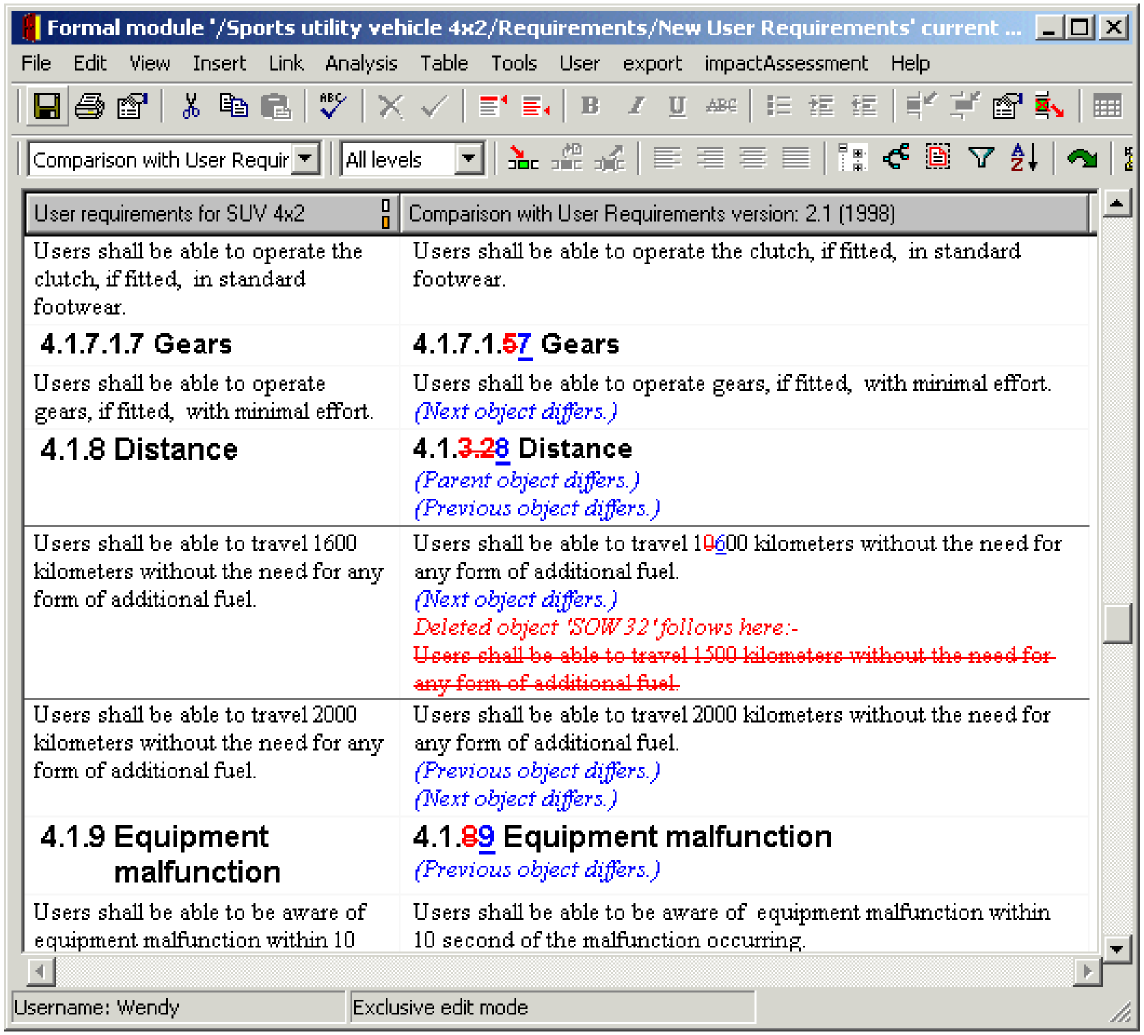
Change History

Previous Baseline

Current Version

DOORS – Module Compare

- Change analysis between versions



Formal module '/Sports utility vehicle 4x2/Requirements/New User Requirements' current ...

Comparison with User Requir
All levels

User requirements for SUV 4x2	Comparison with User Requirements version: 2.1 (1998)
Users shall be able to operate the clutch, if fitted, in standard footwear.	Users shall be able to operate the clutch, if fitted, in standard footwear.
4.1.7.1.7 Gears	**4.1.7.1.~~5~~7 Gears**
Users shall be able to operate gears, if fitted, with minimal effort.	Users shall be able to operate gears, if fitted, with minimal effort. *(Next object differs.)*
4.1.8 Distance	**4.1.~~3.2~~8 Distance** *(Parent object differs.)* *(Previous object differs.)*
Users shall be able to travel 1600 kilometers without the need for any form of additional fuel.	Users shall be able to travel 1~~0~~600 kilometers without the need for any form of additional fuel. *(Next object differs.)* *Deleted object 'SOW 32' follows here:-* ~~Users shall be able to travel 1500 kilometers without the need for any form of additional fuel.~~
Users shall be able to travel 2000 kilometers without the need for any form of additional fuel.	Users shall be able to travel 2000 kilometers without the need for any form of additional fuel. *(Previous object differs.)* *(Next object differs.)*
4.1.9 Equipment malfunction	**4.1.~~8~~9 Equipment malfunction** *(Previous object differs.)*
Users shall be able to be aware of equipment malfunction within 10	Users shall be able to be aware of equipment malfunction within 10 second of the malfunction occurring.

Username: Wendy
Exclusive edit mode

Change Management

Change Management (1)

- The more things change…

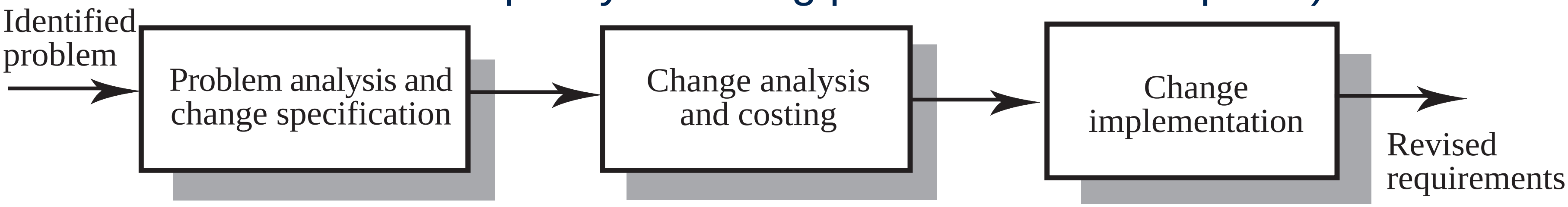
- If you see change not as an enemy, but as a welcome friend, you will secure the most valuable prize of all – the future…

Change Management (2)

- Concerned with the procedures, processes, and standards which are used to manage changes to a system requirements
- Change management policies may cover
  - The change request process and the information required to process each change request
  - The process used to analyse the impact and costs of change and the associated traceability information
  - The membership of the body that formally considers change requests
  - Software support (if any) for the change control process
- A change request may have a status as well as requirements
  - E.g., proposed, rejected, accepted, included...

Change Management Process

- Some requirements problem is identified
  - Could come from an analysis of the requirements, new customer needs, or operational problems with the system
  - The requirements are analysed using problem information and requirements changes are proposed
- The proposed changes are analysed
  - How many requirements (and, if necessary, system components) are affected? Roughly how much would cost, in both time and money?
- The change is implemented
  - A set of amendments to the requirements document or a new document version is produced (of course this should be validated with whatever normal quality checking procedures are in place)


Identified problem
Problem analysis and change specification
Change analysis and costing
Change implementation
Revised requirements

Change Request Form

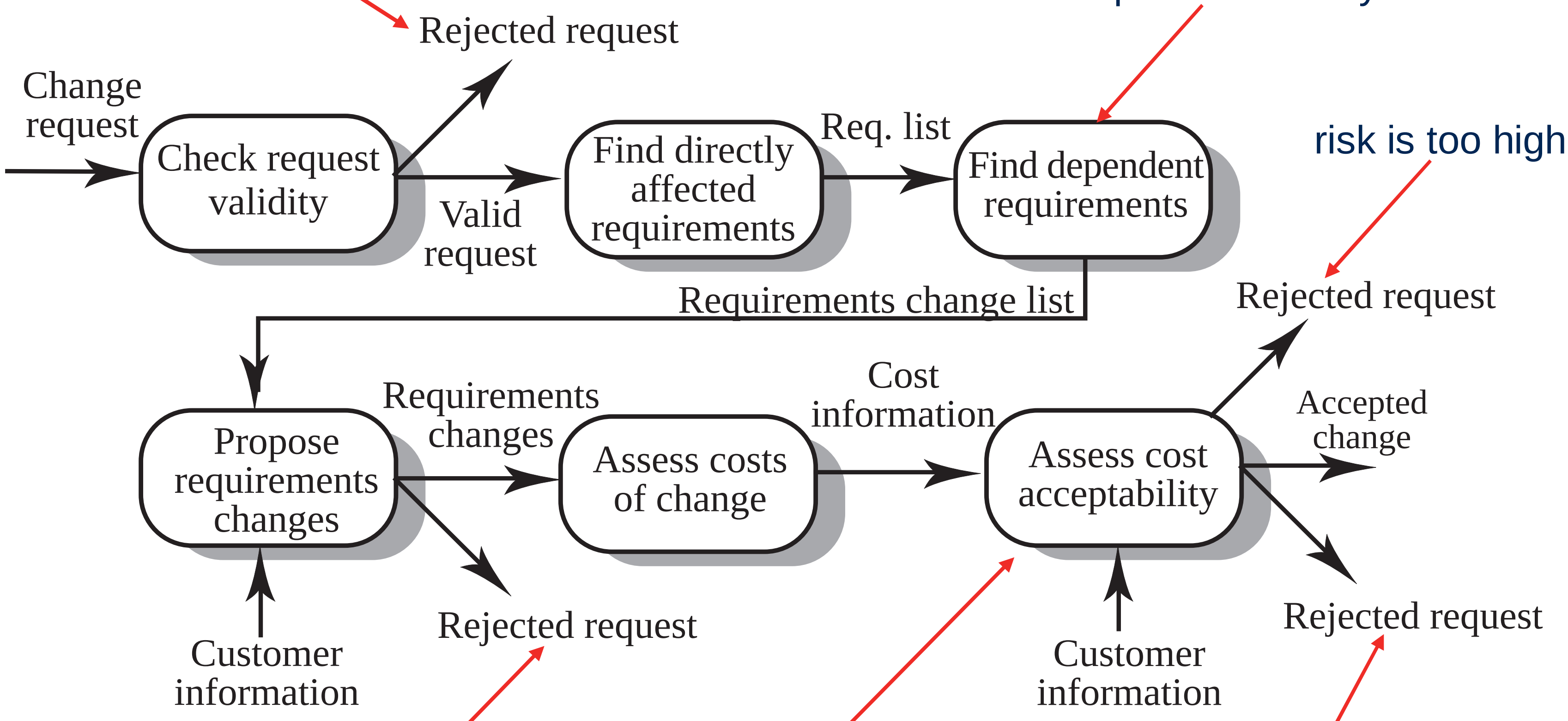
- Proposed changes are usually recorded on a change request form which is then passed to all of the people involved in the analysis of the change
- Change request forms may include
  - Date, Customer, Requester, Product including version
  - Description of change request including rationale
  - Fields to document the change analysis
  - Signature fields
  - Status
  - Comments

Change Analysis and Costing – Example

customers may misunderstand requirements and their context and suggest unnecessary changes

with the help of traceability information

risk is too high



Change request

Check request validity

Rejected request

Valid request

Find directly affected requirements

Req. list

Find dependent requirements

Requirements change list

Propose requirements changes

Customer information

Requirements changes

Rejected request

Assess costs of change

Cost information

Assess cost acceptability

Rejected request

Accepted change

Rejected request

Customer information

negotiations with customers

consequential changes may be unacceptable to user/customer

cost/time required for the implementation of change is too high/long

Source of figure: Kotonya and Sommerville

Different Management Aspects

- Change Management
  - How does a customer submit change requests?
  - How is this request being monitored, prioritized, and implemented?
- Configuration Management
  - Versioning, labelling, and tracking code and other components during the development cycle of software
- Release Management
  - Defines how and when different hardware and software will be made available together as a product

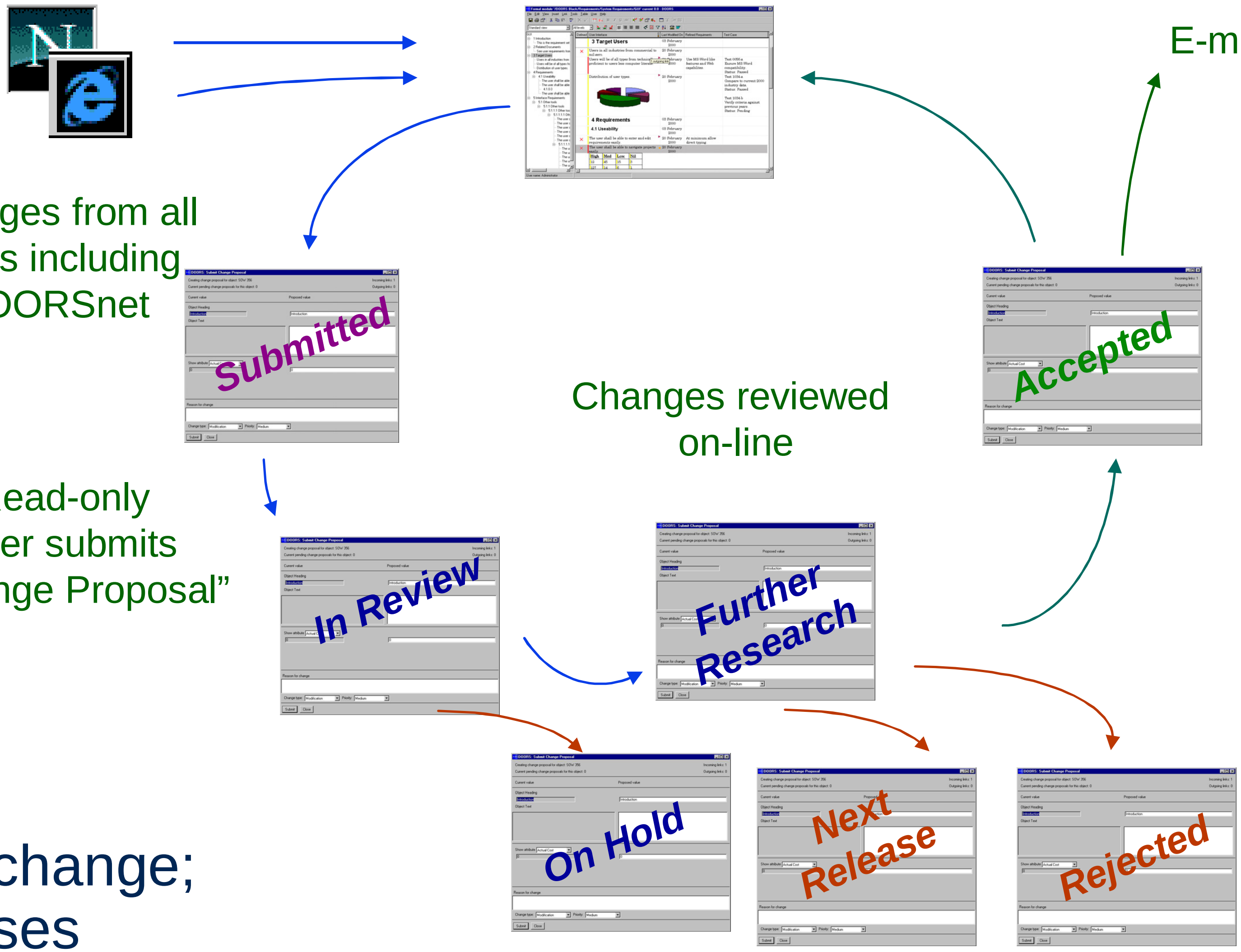
Tool Support for Change Management

- May be provided through requirements management tools or through configuration management tools
- Tool facilities may include
  - Electronic change request forms which are filled in by different participants in the process
  - A database to store and manage requests
  - A change model which may be instantiated so that people responsible for one stage of the process know who is responsible for the next process activity
  - Electronic transfer of forms between people with different responsibilities and electronic mail notification when activities have been completed
  - Electronic signatures
  - Discussion forums
  - In some cases, direct links to a requirements database

Example – DOORS Change Proposal System (1)

- The Change Proposal System (CPS) allows people to access DOORS modules and to propose changes (without immediately changing the modules)
- This allows for feedback and the application of changes in a controlled manner
- An administrator controls the visibility of data as well as who is allowed to propose change requests
- DOORS can also be integrated with SYNERGY
  - Version/change management system

Example – DOORS Change Proposal System (2)



Changes from all users including DOORSnet

Read-only user submits “Change Proposal”

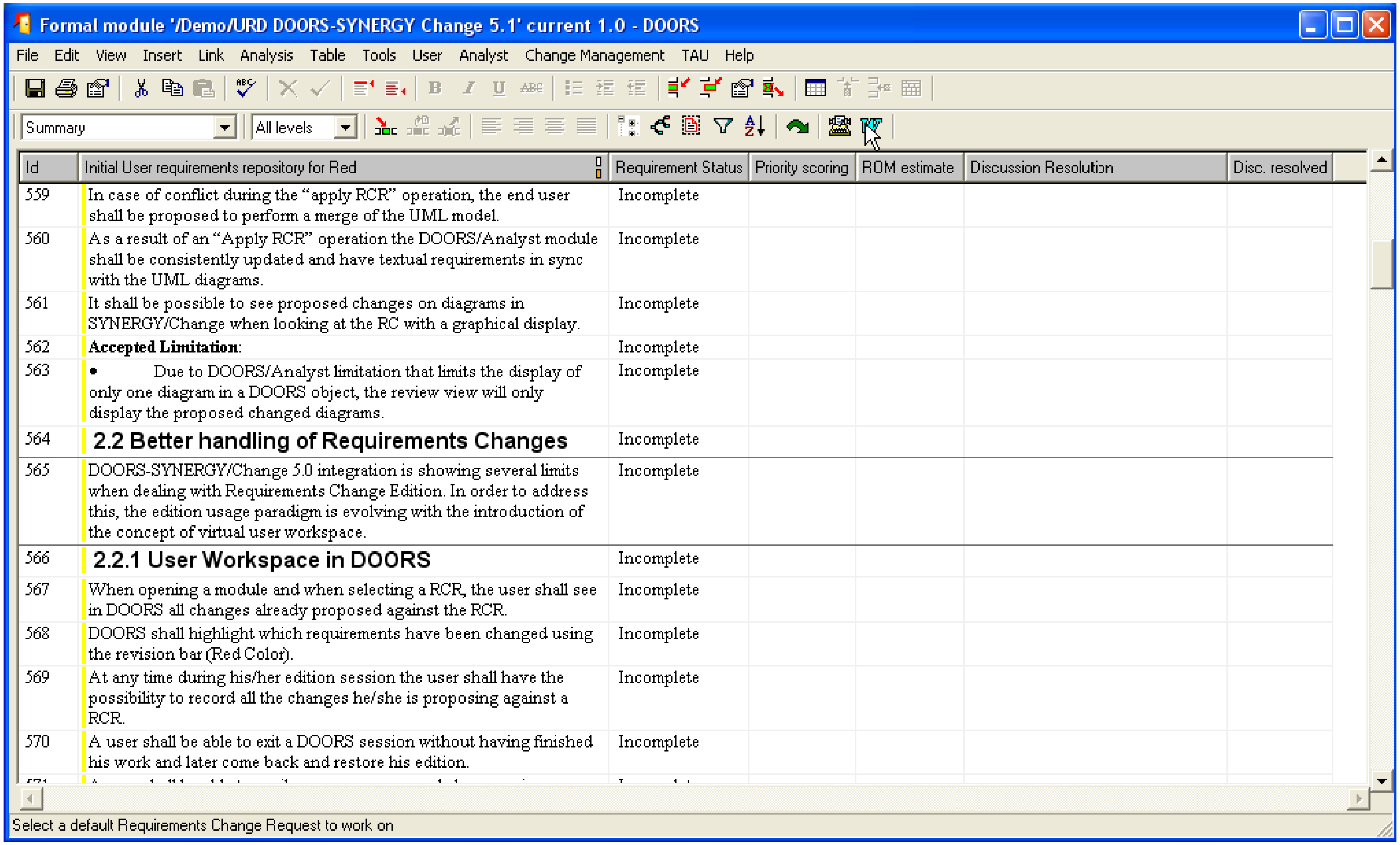
Changes reviewed on-line

E-mail

- Manage change; no surprises

Change Management with DOORS/SYNERGY (1)

- Standard DOORS module

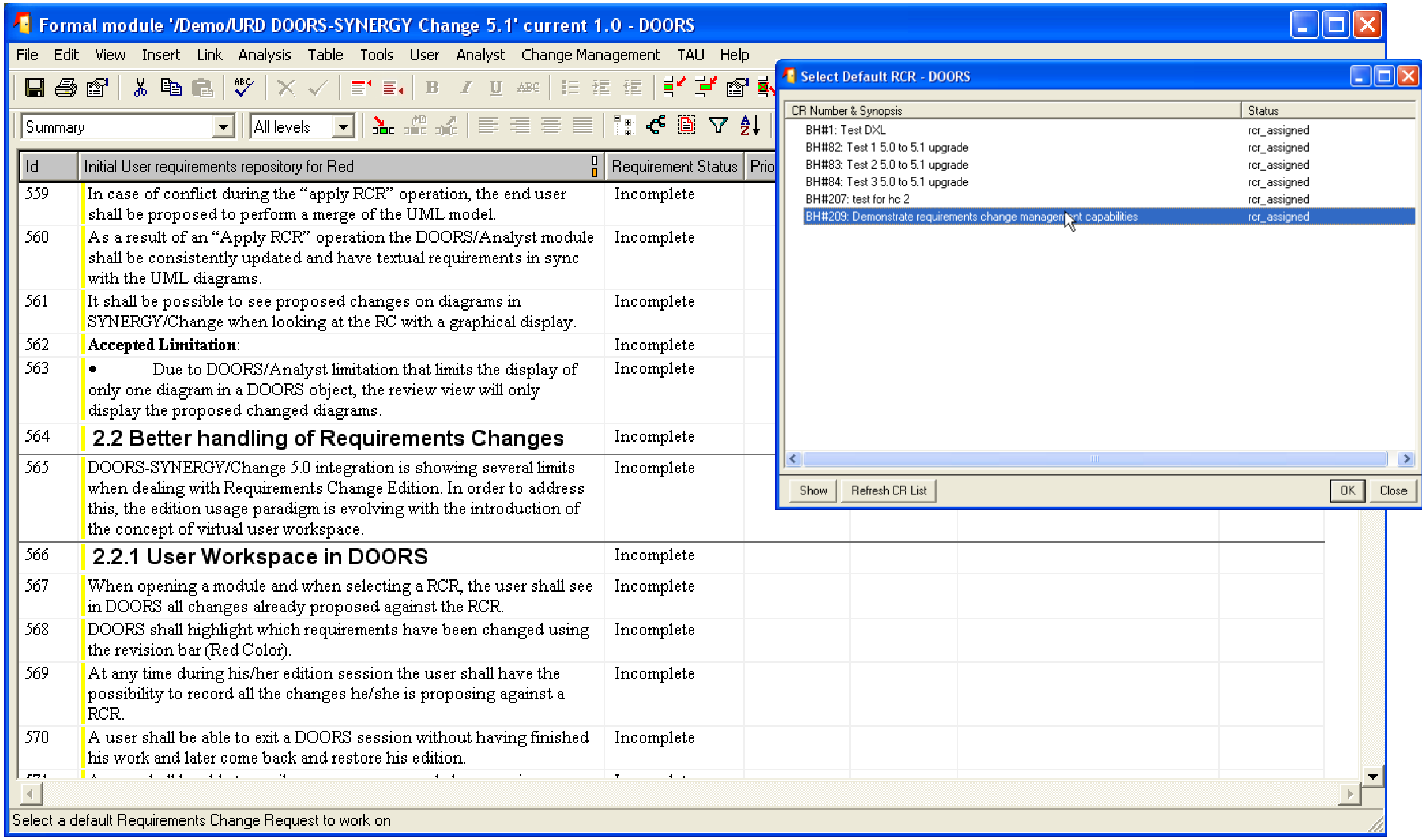

Formal module '/Demo/URD DOORS-SYNERGY Change 5.1' current 1.0 - DOORS
File Edit View Insert Link Analysis Table Tools User Analyst Change Management TAU Help
Summary
All levels

Id	Initial User requirements repository for Red	Requirement Status	Priority scoring	ROM estimate	Discussion Resolution	Disc. resolved
559	In case of conflict during the "apply RCR" operation, the end user shall be proposed to perform a merge of the UML model.	Incomplete				
560	As a result of an "Apply RCR" operation the DOORS/Analyst module shall be consistently updated and have textual requirements in sync with the UML diagrams.	Incomplete				
561	It shall be possible to see proposed changes on diagrams in SYNERGY/Change when looking at the RC with a graphical display.	Incomplete				
562	**Accepted Limitation**:	Incomplete				
563	• Due to DOORS/Analyst limitation that limits the display of only one diagram in a DOORS object, the review view will only display the proposed changed diagrams.	Incomplete				
564	**2.2 Better handling of Requirements Changes**	Incomplete				
565	DOORS-SYNERGY/Change 5.0 integration is showing several limits when dealing with Requirements Change Edition. In order to address this, the edition usage paradigm is evolving with the introduction of the concept of virtual user workspace.	Incomplete				
566	**2.2.1 User Workspace in DOORS**	Incomplete				
567	When opening a module and when selecting a RCR, the user shall see in DOORS all changes already proposed against the RCR.	Incomplete				
568	DOORS shall highlight which requirements have been changed using the revision bar (Red Color).	Incomplete				
569	At any time during his/her edition session the user shall have the possibility to record all the changes he/she is proposing against a RCR.	Incomplete				
570	A user shall be able to exit a DOORS session without having finished his work and later come back and restore his edition.	Incomplete				

Select a default Requirements Change Request to work on

Change Management with DOORS/SYNERGY (2)

- Select a SYNERGY change request



Formal module '/Demo/URD DOORS-SYNERGY Change 5.1' current 1.0 - DOORS

Id	Initial User requirements repository for Red	Requirement Status	Prio
559	In case of conflict during the "apply RCR" operation, the end user shall be proposed to perform a merge of the UML model.	Incomplete	
560	As a result of an "Apply RCR" operation the DOORS/Analyst module shall be consistently updated and have textual requirements in sync with the UML diagrams.	Incomplete	
561	It shall be possible to see proposed changes on diagrams in SYNERGY/Change when looking at the RC with a graphical display.	Incomplete	
562	**Accepted Limitation**:	Incomplete	
563	• Due to DOORS/Analyst limitation that limits the display of only one diagram in a DOORS object, the review view will only display the proposed changed diagrams.	Incomplete	
564	**2.2 Better handling of Requirements Changes**	Incomplete	
565	DOORS-SYNERGY/Change 5.0 integration is showing several limits when dealing with Requirements Change Edition. In order to address this, the edition usage paradigm is evolving with the introduction of the concept of virtual user workspace.	Incomplete	
566	**2.2.1 User Workspace in DOORS**	Incomplete	
567	When opening a module and when selecting a RCR, the user shall see in DOORS all changes already proposed against the RCR.	Incomplete	
568	DOORS shall highlight which requirements have been changed using the revision bar (Red Color).	Incomplete	
569	At any time during his/her edition session the user shall have the possibility to record all the changes he/she is proposing against a RCR.	Incomplete	
570	A user shall be able to exit a DOORS session without having finished his work and later come back and restore his edition.	Incomplete	

Select a default Requirements Change Request to work on

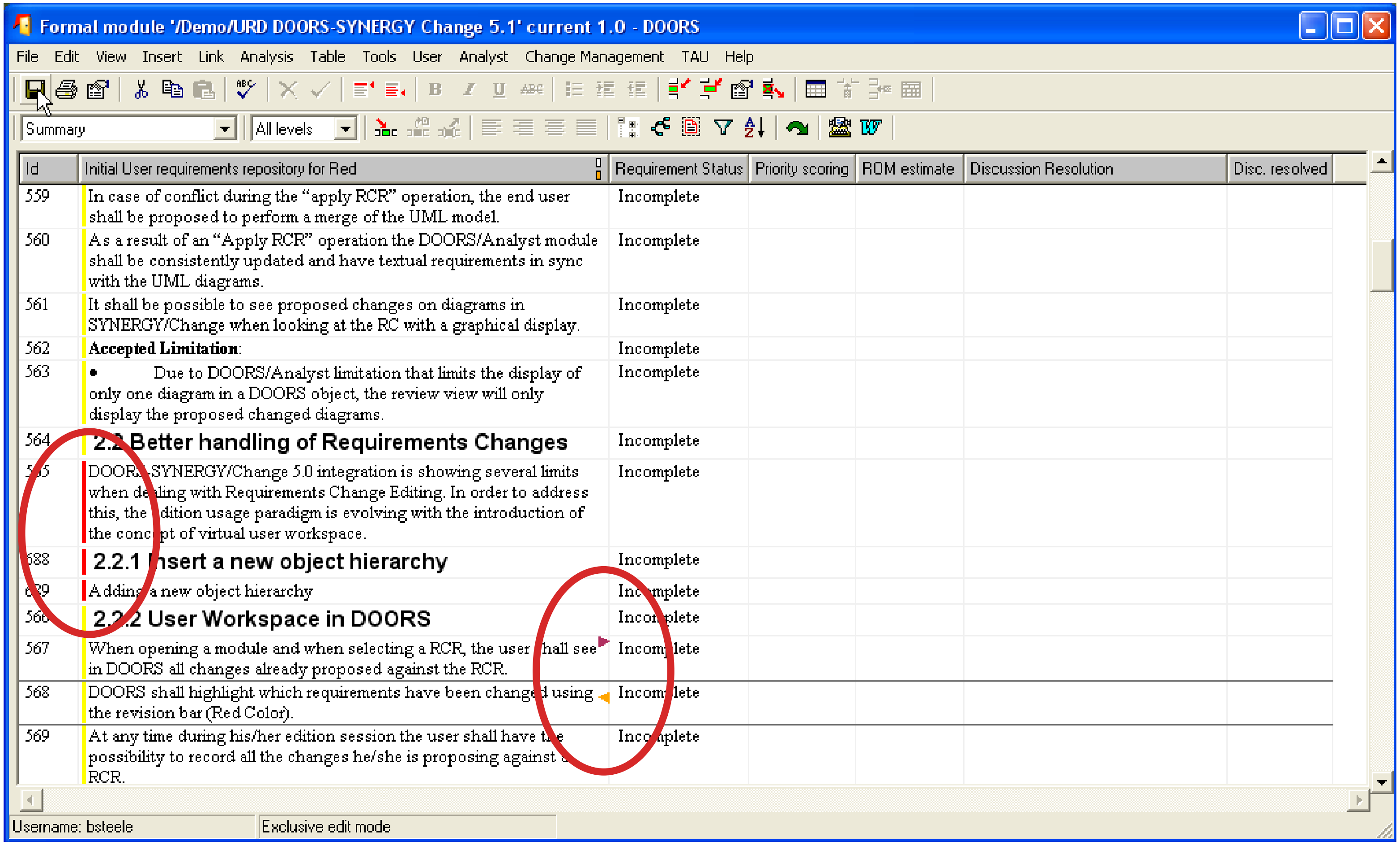
Select Default RCR - DOORS

CR Number & Synopsis	Status
BH#1: Test DXL	rcr_assigned
BH#82: Test 1 5.0 to 5.1 upgrade	rcr_assigned
BH#83: Test 2 5.0 to 5.1 upgrade	rcr_assigned
BH#84: Test 3 5.0 to 5.1 upgrade	rcr_assigned
BH#207: test for hc 2	rcr_assigned
BH#209: Demonstrate requirements change management capabilities	rcr_assigned

File Edit View Insert Link Analysis Table Tools User Analyst Change Management TAU Help
Summary
All levels
Show
Refresh CR List
OK
Close

Change Management with DOORS/SYNERGY (3)

- Perform appropriate changes

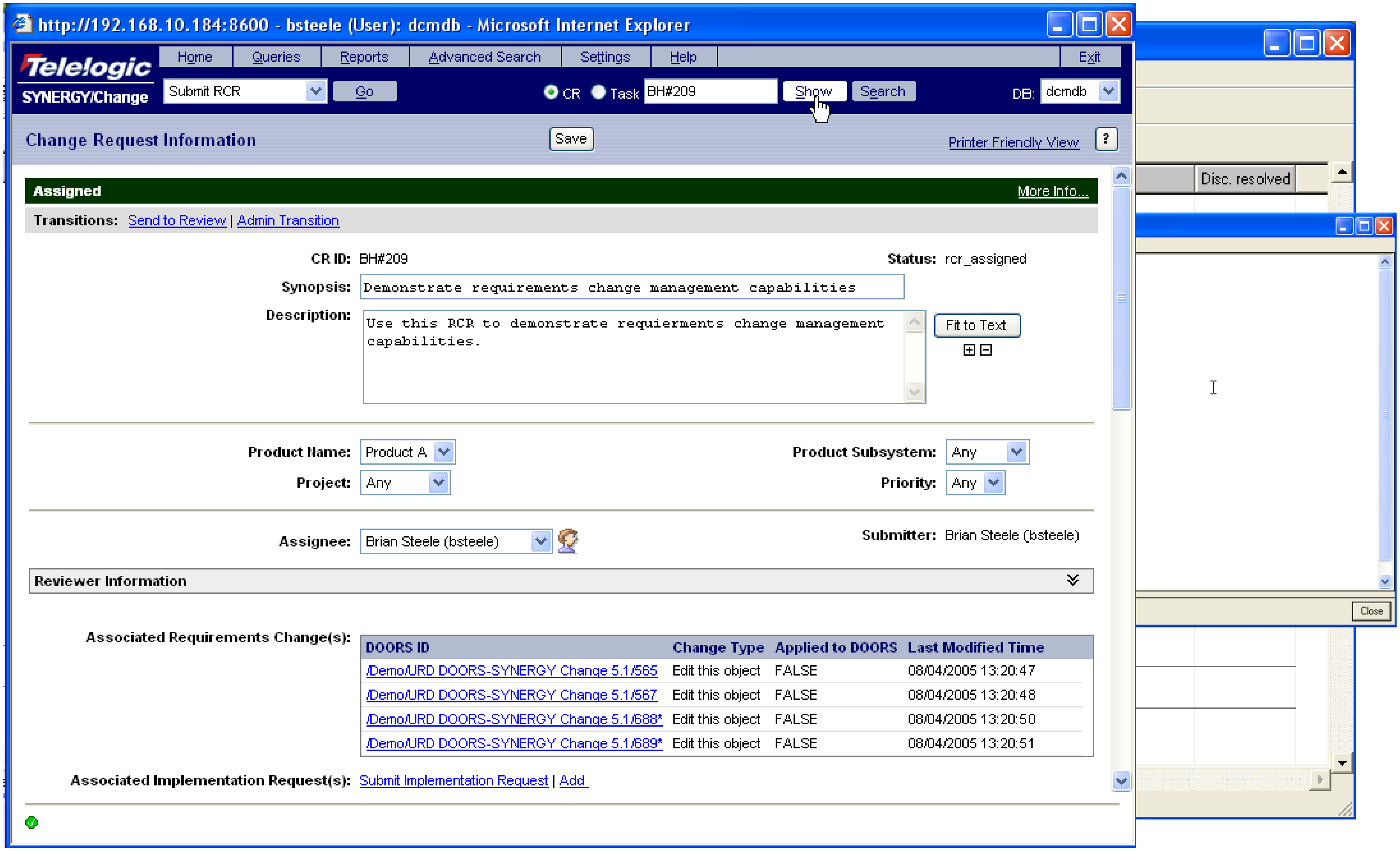

Formal module '/Demo/URD DOORS-SYNERGY Change 5.1' current 1.0 - DOORS
File Edit View Insert Link Analysis Table Tools User Analyst Change Management TAU Help
Summary
All levels

Id	Initial User requirements repository for Red	Requirement Status	Priority scoring	ROM estimate	Discussion Resolution	Disc. resolved
559	In case of conflict during the “apply RCR” operation, the end user shall be proposed to perform a merge of the UML model.	Incomplete				
560	As a result of an “Apply RCR” operation the DOORS/Analyst module shall be consistently updated and have textual requirements in sync with the UML diagrams.	Incomplete				
561	It shall be possible to see proposed changes on diagrams in SYNERGY/Change when looking at the RC with a graphical display.	Incomplete				
562	**Accepted Limitation**:	Incomplete				
563	• Due to DOORS/Analyst limitation that limits the display of only one diagram in a DOORS object, the review view will only display the proposed changed diagrams.	Incomplete				
564	**2.2 Better handling of Requirements Changes**	Incomplete				
565	DOORS-SYNERGY/Change 5.0 integration is showing several limits when dealing with Requirements Change Editing. In order to address this, the edition usage paradigm is evolving with the introduction of the concept of virtual user workspace.	Incomplete				
688	**2.2.1 Insert a new object hierarchy**	Incomplete				
689	Adding a new object hierarchy	Incomplete				
566	**2.2.2 User Workspace in DOORS**	Incomplete				
567	When opening a module and when selecting a RCR, the user shall see in DOORS all changes already proposed against the RCR.	Incomplete				
568	DOORS shall highlight which requirements have been changed using the revision bar (Red Color).	Incomplete				
569	At any time during his/her edition session the user shall have the possibility to record all the changes he/she is proposing against a RCR.	Incomplete				

Username: bsteele
Exclusive edit mode

Change Management with DOORS/SYNERGY (4)

- Changes managed by SYNERGY/Change



Submit RCR

BH#209

DB: dcmdb

Change Request Information

Assigned

Transitions: Send to Review | Admin Transition

CR ID: BH#209

Status: rcr_assigned

Synopsis: Demonstrate requirements change management capabilities

Description: Use this RCR to demonstrate requierments change management capabilities.

Product Name: Product A

Product Subsystem: Any

Project: Any

Priority: Any

Assignee: Brian Steele (bsteele)

Submitter: Brian Steele (bsteele)

Reviewer Information

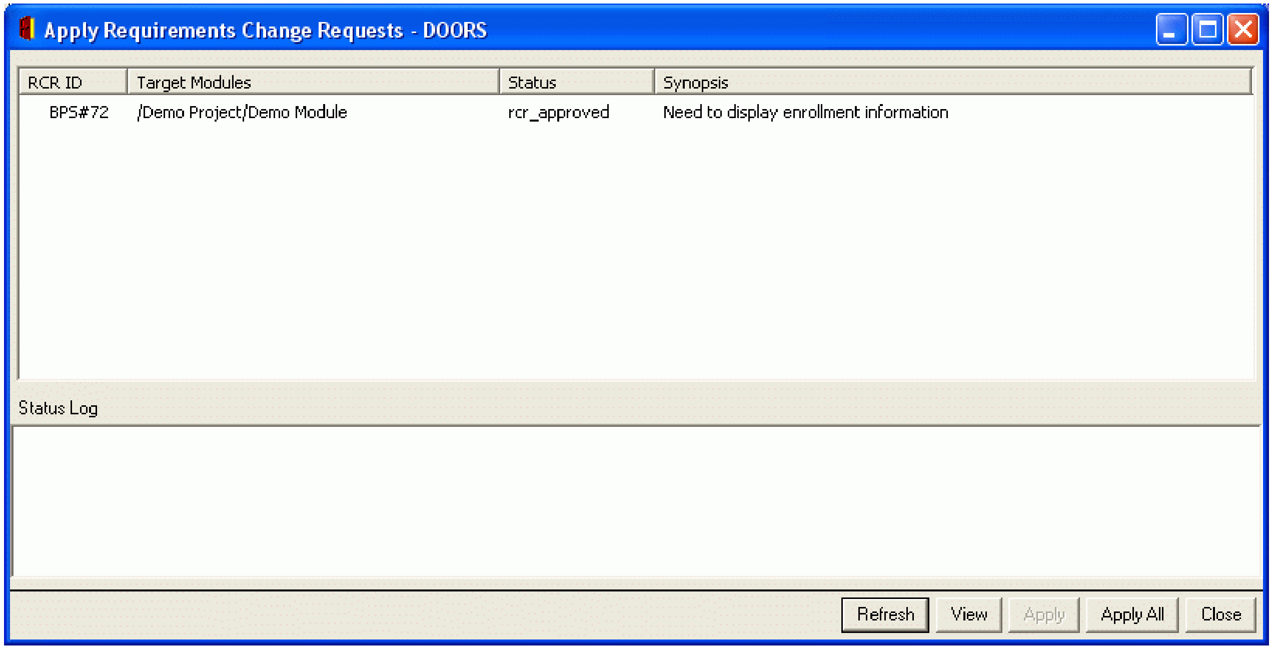
Associated Requirements Change(s):

DOORS ID	Change Type	Applied to DOORS	Last Modified Time
/Demo/URD DOORS-SYNERGY Change 5.1/565	Edit this object	FALSE	08/04/2005 13:20:47
/Demo/URD DOORS-SYNERGY Change 5.1/567	Edit this object	FALSE	08/04/2005 13:20:48
/Demo/URD DOORS-SYNERGY Change 5.1/688*	Edit this object	FALSE	08/04/2005 13:20:50
/Demo/URD DOORS-SYNERGY Change 5.1/689*	Edit this object	FALSE	08/04/2005 13:20:51

Associated Implementation Request(s): Submit Implementation Request | Add

Change Management with DOORS/SYNERGY (5)

- Once approved, the change request can be applied to DOORS


Apply Requirements Change Requests - DOORS

RCR ID	Target Modules	Status	Synopsis
BPS#72	/Demo Project/Demo Module	rcr_approved	Need to display enrollment information

Requirements Management Tools

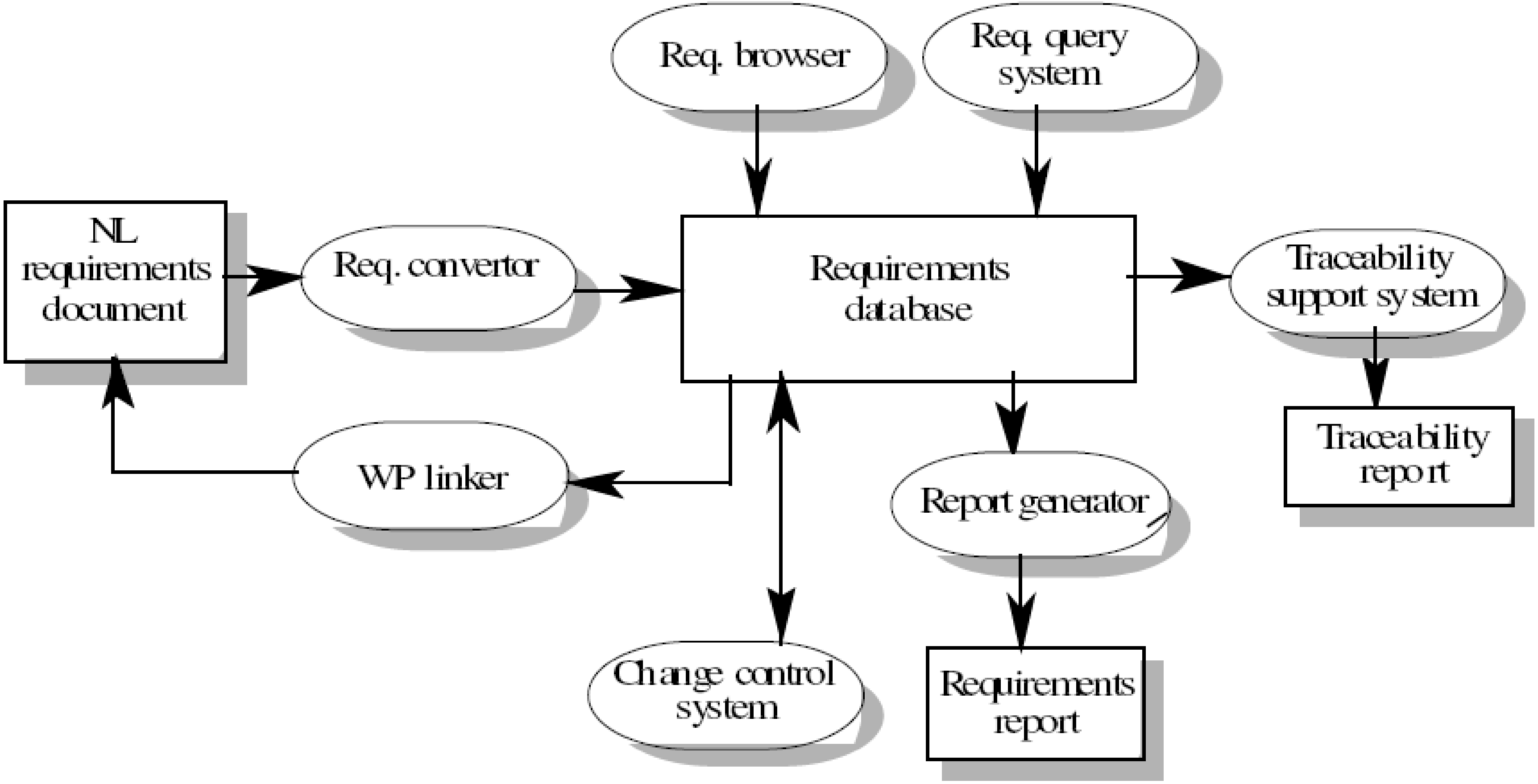
What Kind of Tool Do We Need?

- Different companies will use different tools, which may or may not be tailored to the requirements management task
  - Word processor (Microsoft Word with templates…)
  - Spreadsheet (Microsoft Excel…)
  - Industrial-strength, commercial RM tools
    - IBM/Telelogic DOORS, IBM Requisite Pro, Borland CaliberRM…
  - Internal tools
    - GenSpec (Hydro-Quebec)…
  - Open source RM tools
    - OSRMT: http://sourceforge.net/projects/osrmt
  - Bug tracking tools (free or not)
    - Bugzilla…
  - Collaboration tools (free or not)
    - TWiki…

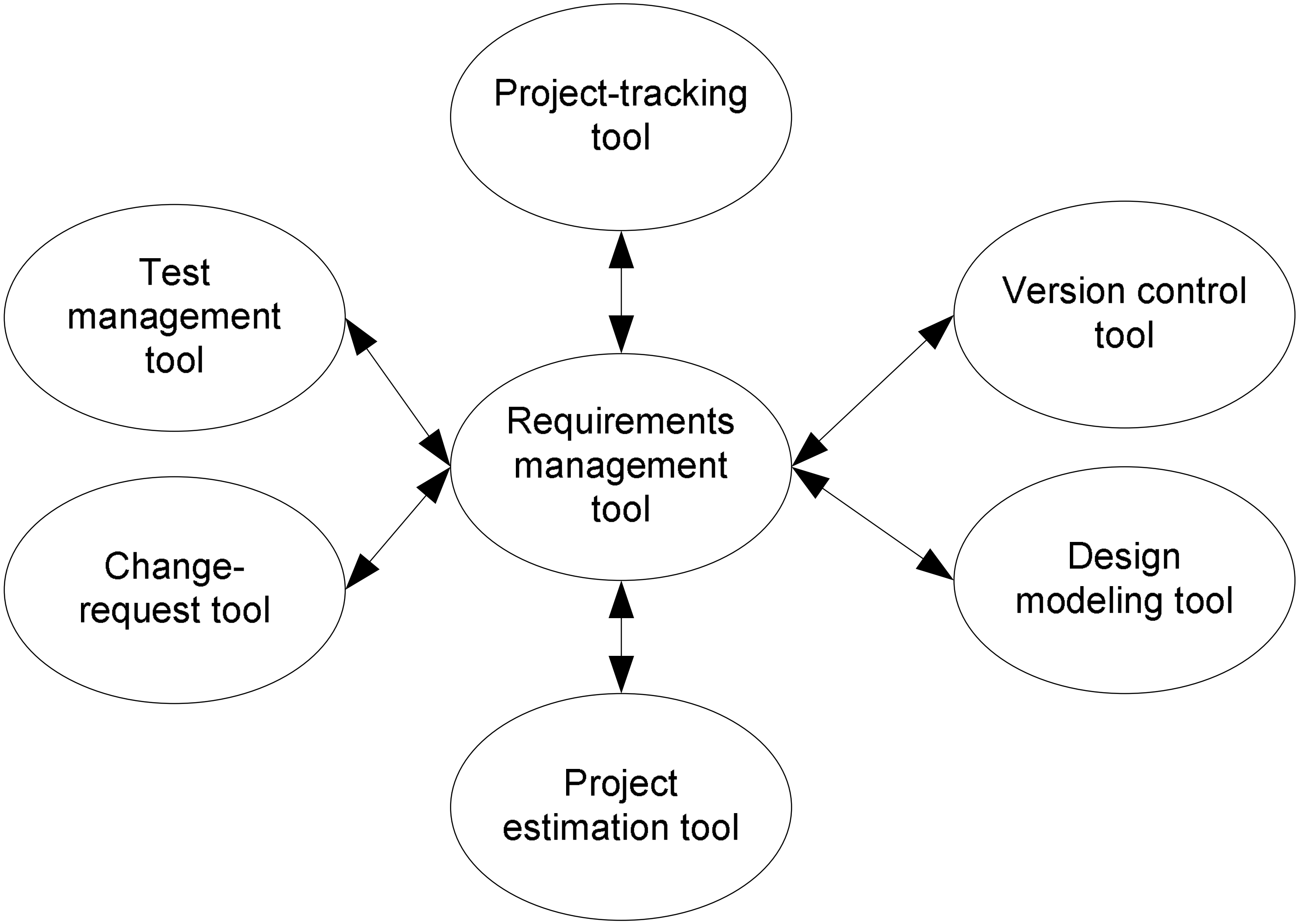
What Should We Look For in a Tool?

- Types/attributes for requirements and links
- Specifications and models
- Version and change management
- Database repository
- Traceability
- Analysis (impact, completeness, style, differences…)
- Automatic inspection of requirements (according to rules)
- Visualization and reports
- Requirements document generation
- Monitoring of requirements statuses
- Access control
- Import/export
- Communication with stakeholders
- Scripting language (for automation)
- Reuse of requirements, models, projects
- …

RM Tool Architecture – Example



Requirements Management Implies Integration!



Approaches – Document or Database? (1)

- Requirements have to be stored in such a way that they can be accessed easily and related to other requirements
- Document (e.g., Word)
  - Easy to use, easy to access, simple training
  - Requirements are all stored in the requirements document
  - It is easy to produce the final requirements document
  - But: Traceability? Status reports? Granularity of requirements? Search and navigation facilities? Change management? Version control? Analysis? Simultaneous access control?...

Approaches – Document or Database? (2)

- Database (e.g., DOORS)
  - Good for management, controlled access, links, analysis, reports
  - Good query and navigation facilities
  - Support for change and version management
  - But: hard (and costly) to configure, manage, and use; link between the database and the requirements document must be maintained (final requirements document must be generated)
- Ideally: Target the benefits of both
  - E.g., DOORS and RequisitePro offer integrations with Word (import/export) as well as document-oriented views (for the “look and feel”…)

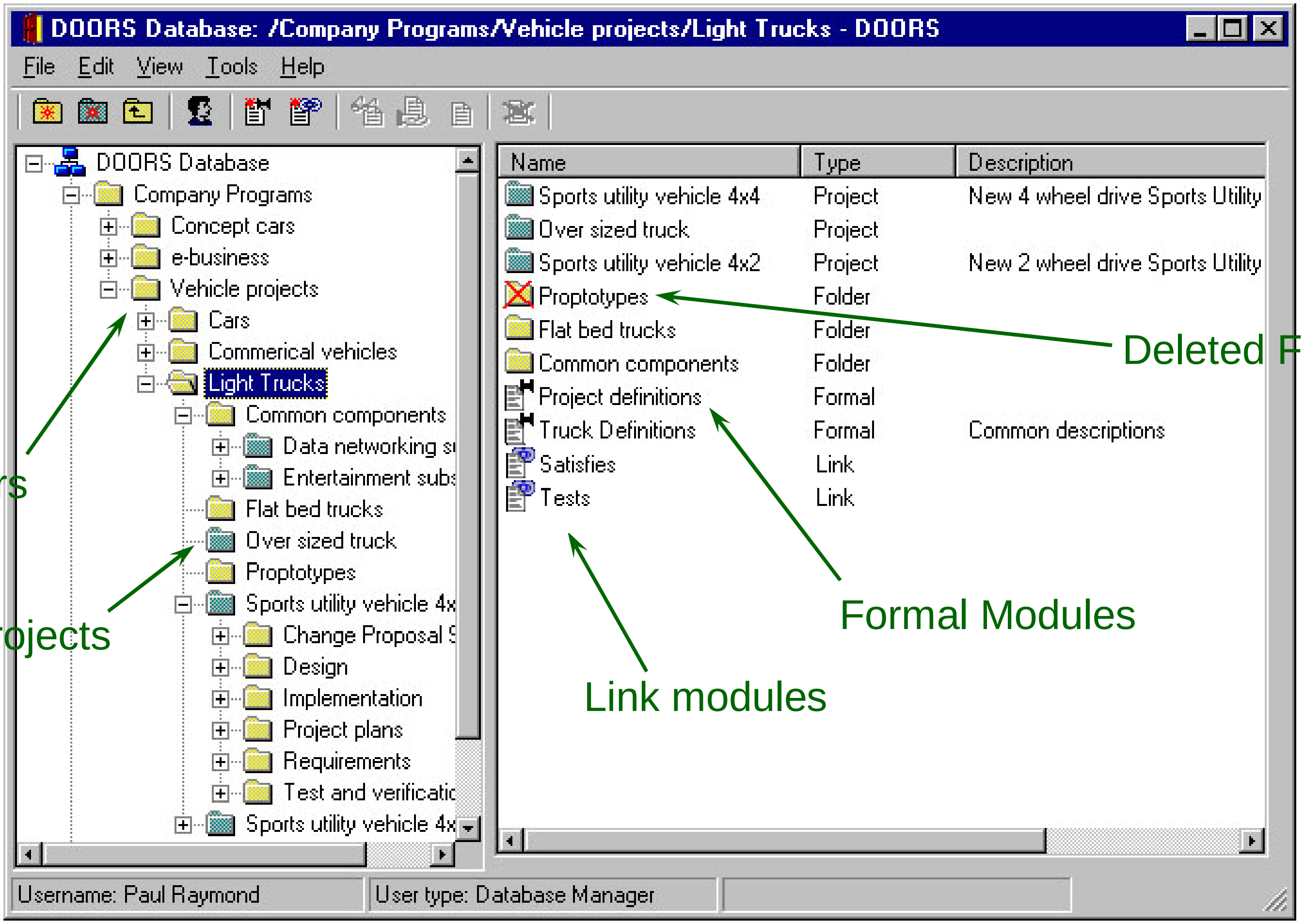
How About Evolving the Process Itself?

- Evolution of requirements types
  - Adding / modifying / deleting
    - Attributes
    - Link types
    - Requirements status
    - …
- Evolution of change management
  - Adding / modifying / deleting
    - Attributes
    - Lifecycle status
    - Forms
    - …

Thinking About Getting a RM Tool?

- The list of potential criteria and the list of products can be rather long…
  - See the INCOSE study: http://www.incose.org/ProductsPubs/Products/rmsurvey.aspx
    - About 25 tools and 80 criteria, with explanations
- Which are relevant to you, in your context (project, organization…)?
  - Need a goal model to define criteria and for assessment!

DOORS – Database View



DOORS Database: /Company Programs/Vehicle projects/Light Trucks - DOORS

- DOORS Database
  - Company Programs
    - Concept cars
    - e-business
    - Vehicle projects
      - Cars
      - Commerical vehicles
      - Light Trucks
        - Common components
          - Data networking s
          - Entertainment subs
        - Flat bed trucks
        - Over sized truck
        - Proptotypes
        - Sports utility vehicle 4x
          - Change Proposal S
          - Design
          - Implementation
          - Project plans
          - Requirements
          - Test and verificatic
        - Sports utility vehicle 4x

Name	Type	Description
Sports utility vehicle 4x4	Project	New 4 wheel drive Sports Utility
Over sized truck	Project	
Sports utility vehicle 4x2	Project	New 2 wheel drive Sports Utility
Proptotypes	Folder	
Flat bed trucks	Folder	
Common components	Folder	
Project definitions	Formal	
Truck Definitions	Formal	Common descriptions
Satisfies	Link	
Tests	Link	

Username: Paul Raymond

User type: Database Manager

Folders

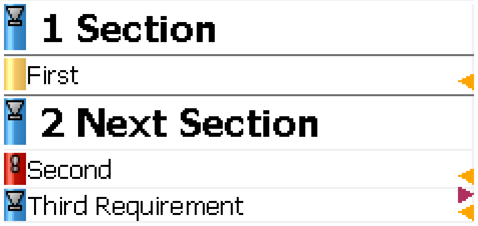
Projects

Deleted Folder

Formal Modules

Link modules

DOORS – Displayed Information



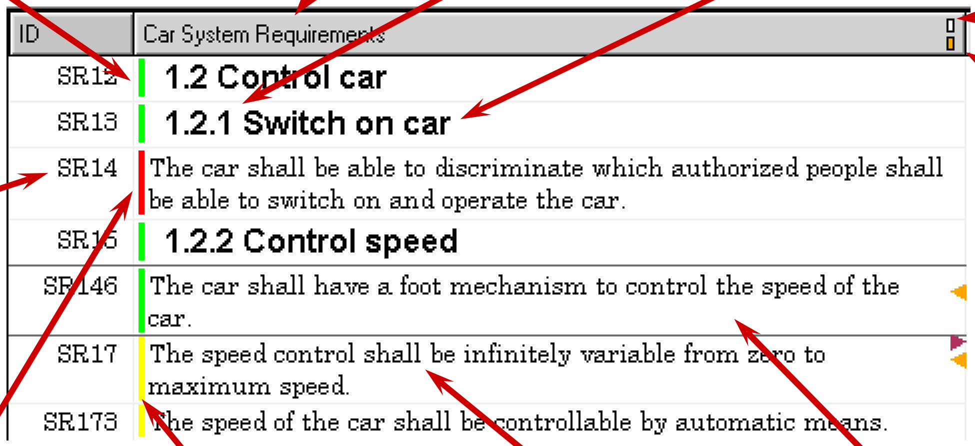
1 Section

First

2 Next Section

Second

Third Requirement



ID	Car System Requirements
SR12	1.2 Control car
SR13	1.2.1 Switch on car
SR14	The car shall be able to discriminate which authorized people shall be able to switch on and operate the car.
SR15	1.2.2 Control speed
SR146	The car shall have a foot mechanism to control the speed of the car.
SR17	The speed control shall be infinitely variable from zero to maximum speed.
SR173	The speed of the car shall be controllable by automatic means.

"No change since baseline" change-bar (green / blue)

Column Heading

Object or Section Number

Object Heading

Use in Graphical view

Use as DataTip

Object Identifier

Link Indicator

Object Text

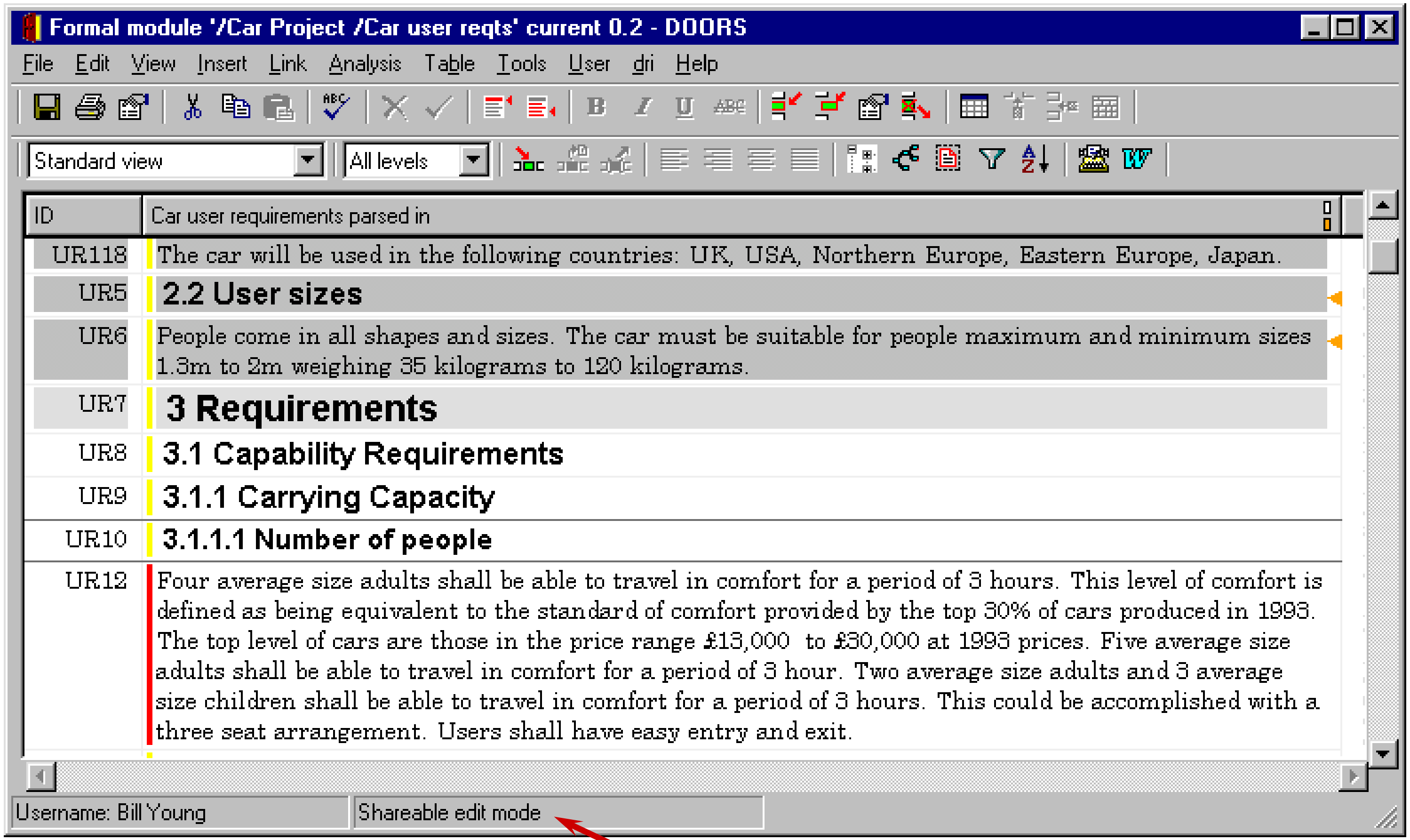
Current Object

"Changed this session" change-bar, unsaved (red)

"Changed since baseline" change-bar, saved (yellow)

DOORS – Multi-User Editing

- Make required edits, and unlock to allow others access



Formal module '/Car Project /Car user reqts' current 0.2 - DOORS
File Edit View Insert Link Analysis Table Tools User dri Help
Standard view
All levels

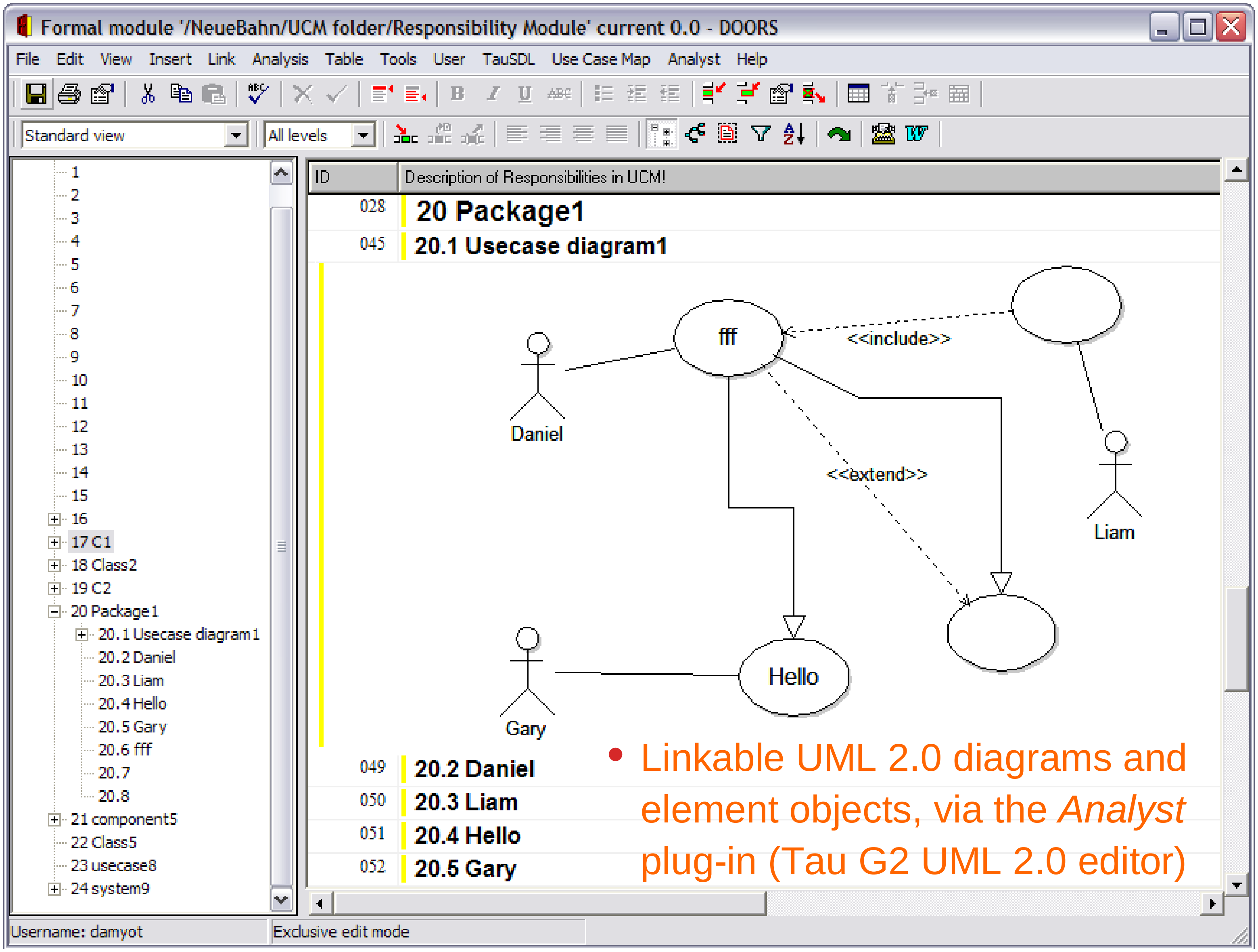
ID	Car user requirements parsed in
UR118	The car will be used in the following countries: UK, USA, Northern Europe, Eastern Europe, Japan.
UR5	2.2 User sizes
UR6	People come in all shapes and sizes. The car must be suitable for people maximum and minimum sizes 1.3m to 2m weighing 35 kilograms to 120 kilograms.
UR7	3 Requirements
UR8	3.1 Capability Requirements
UR9	3.1.1 Carrying Capacity
UR10	3.1.1.1 Number of people
UR12	Four average size adults shall be able to travel in comfort for a period of 3 hours. This level of comfort is defined as being equivalent to the standard of comfort provided by the top 30% of cars produced in 1993. The top level of cars are those in the price range £13,000 to £30,000 at 1993 prices. Five average size adults shall be able to travel in comfort for a period of 3 hour. Two average size adults and 3 average size children shall be able to travel in comfort for a period of 3 hours. This could be accomplished with a three seat arrangement. Users shall have easy entry and exit.

Username: Bill Young

Shareable edit mode

Mode

DOORS – Integration with UML 2.0



Formal module '/NeueBahn/UCM folder/Responsibility Module' current 0.0 - DOORS

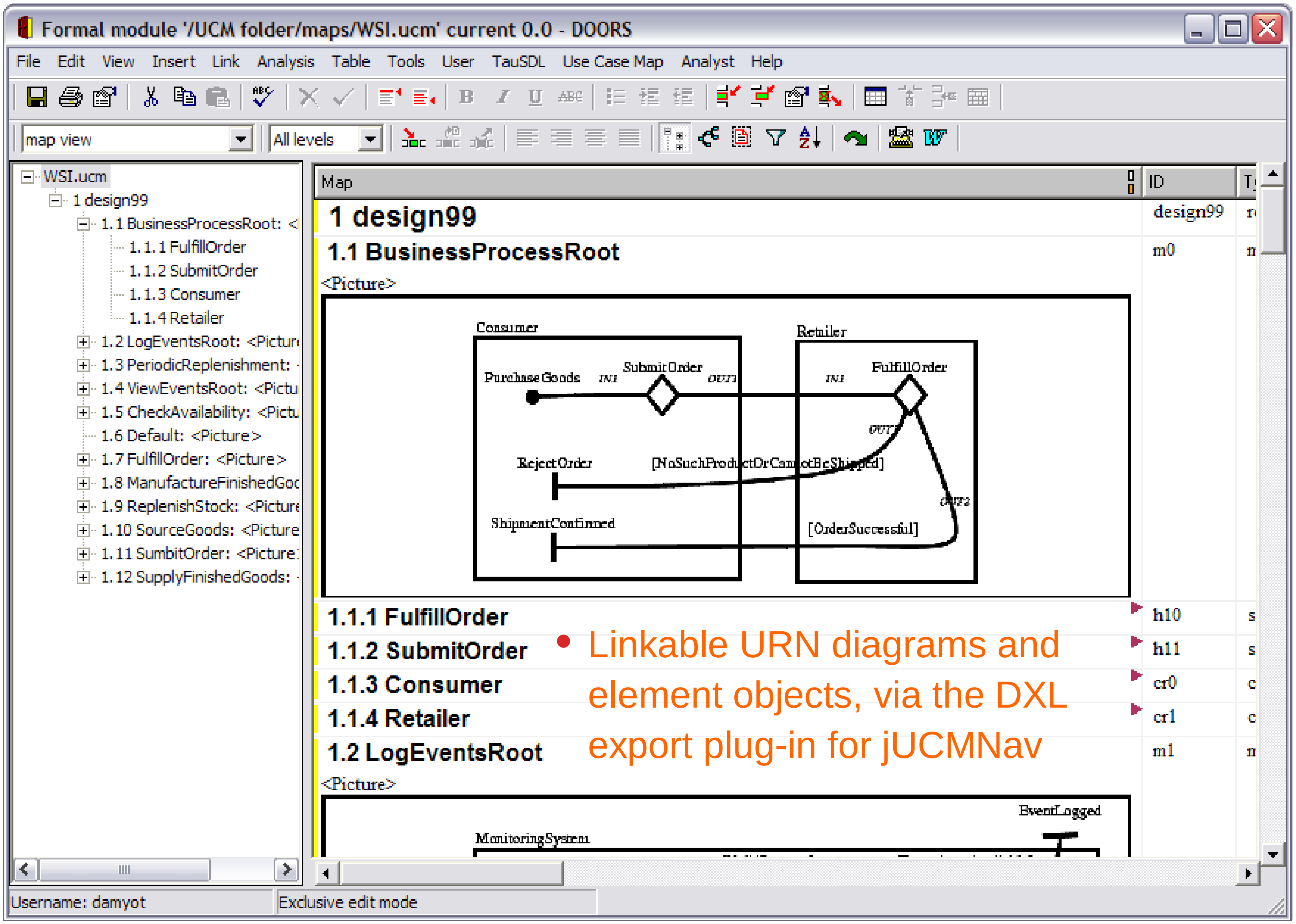
ID	Description of Responsibilities in UCM!
028	20 Package1
045	20.1 Usecase diagram1
049	20.2 Daniel
050	20.3 Liam
051	20.4 Hello
052	20.5 Gary

Username: damyot

Exclusive edit mode

- Linkable UML 2.0 diagrams and element objects, via the *Analyst* plug-in (Tau G2 UML 2.0 editor)

DOORS – Integration with URN



Formal module '/UCM folder/maps/WSI.ucm' current 0.0 - DOORS

- WSI.ucm
  - 1 design99
    - 1.1 BusinessProcessRoot: <
      - 1.1.1 FulfillOrder
      - 1.1.2 SubmitOrder
      - 1.1.3 Consumer
      - 1.1.4 Retailer
    - 1.2 LogEventsRoot: <Pictur
    - 1.3 PeriodicReplenishment:
    - 1.4 ViewEventsRoot: <Pictu
    - 1.5 CheckAvailability: <Pictu
    - 1.6 Default: <Picture>
    - 1.7 FulfillOrder: <Picture>
    - 1.8 ManufactureFinishedGoo
    - 1.9 ReplenishStock: <Picture
    - 1.10 SourceGoods: <Picture
    - 1.11 SumbitOrder: <Picture
    - 1.12 SupplyFinishedGoods:

Map	ID	T
1 design99	design99	r
1.1 BusinessProcessRoot <Picture>	m0	n
1.1.1 FulfillOrder	h10	s
1.1.2 SubmitOrder	h11	s
1.1.3 Consumer	cr0	c
1.1.4 Retailer	cr1	c
1.2 LogEventsRoot <Picture>	m1	n

Username: damyot

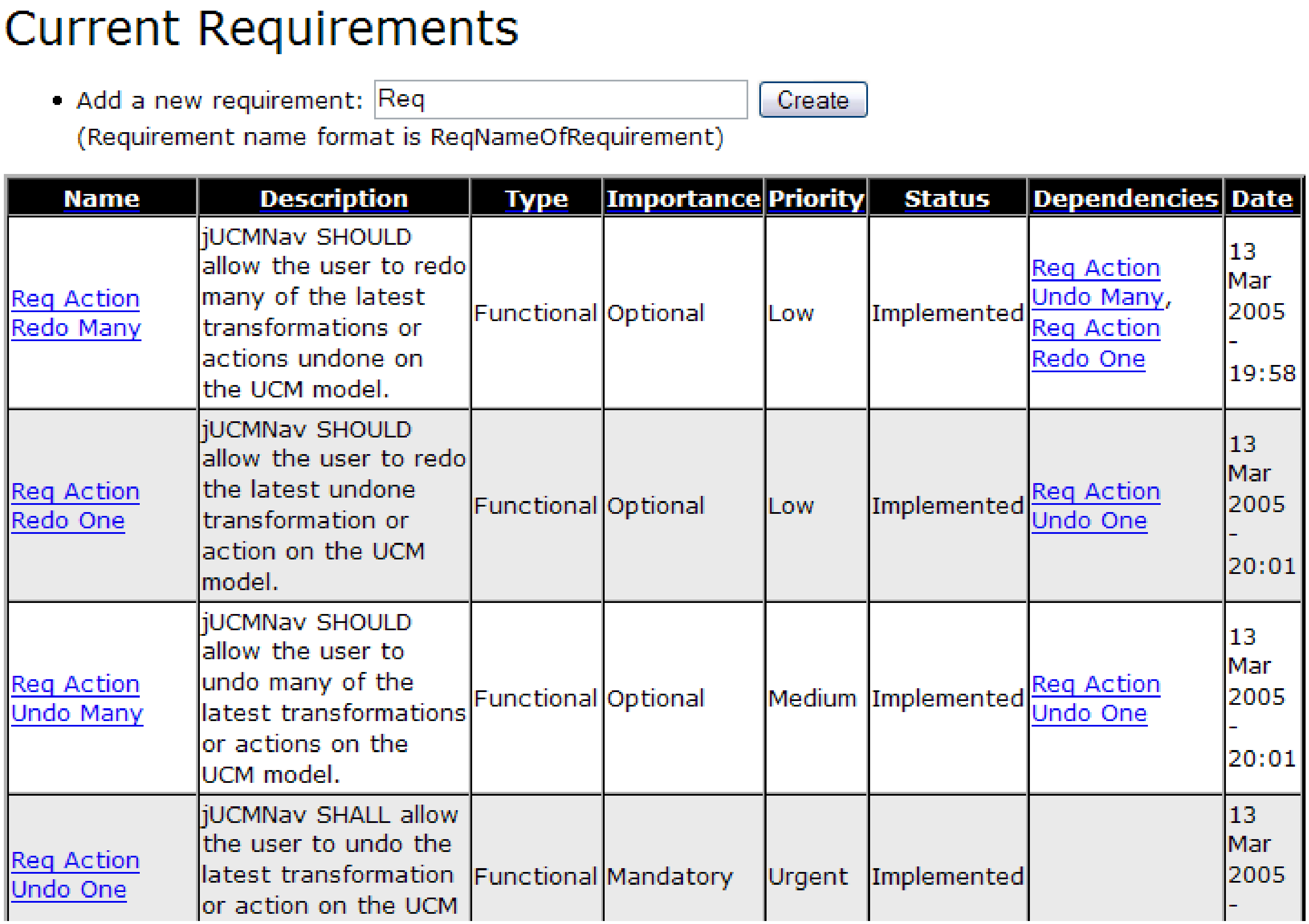
Exclusive edit mode

- Linkable URN diagrams and element objects, via the DXL export plug-in for jUCMNav

TWiki Overview

- A generic Wiki tool (TWiki.org)
  - Promotes collaboration
  - Database-driven
  - Access and version control
  - Forms and queries
  - State-based workflows (processes)
  - Text and graphics
  - Lightweight, extensible (plug-in architecture)
- Example of Forms and Queries
  - Requirements: http://cserg0.site.uottawa.ca/twiki/bin/view/ProjetSEG/UCMNavRequirements
  - Library: http://cserg0.site.uottawa.ca/twiki/bin/view/UCM/UCMVirtualLibrary
  - Use Cases: http://cserg0.site.uottawa.ca/seg/bin/view/CSI4900/UseCases

TWiki for Requirements Management

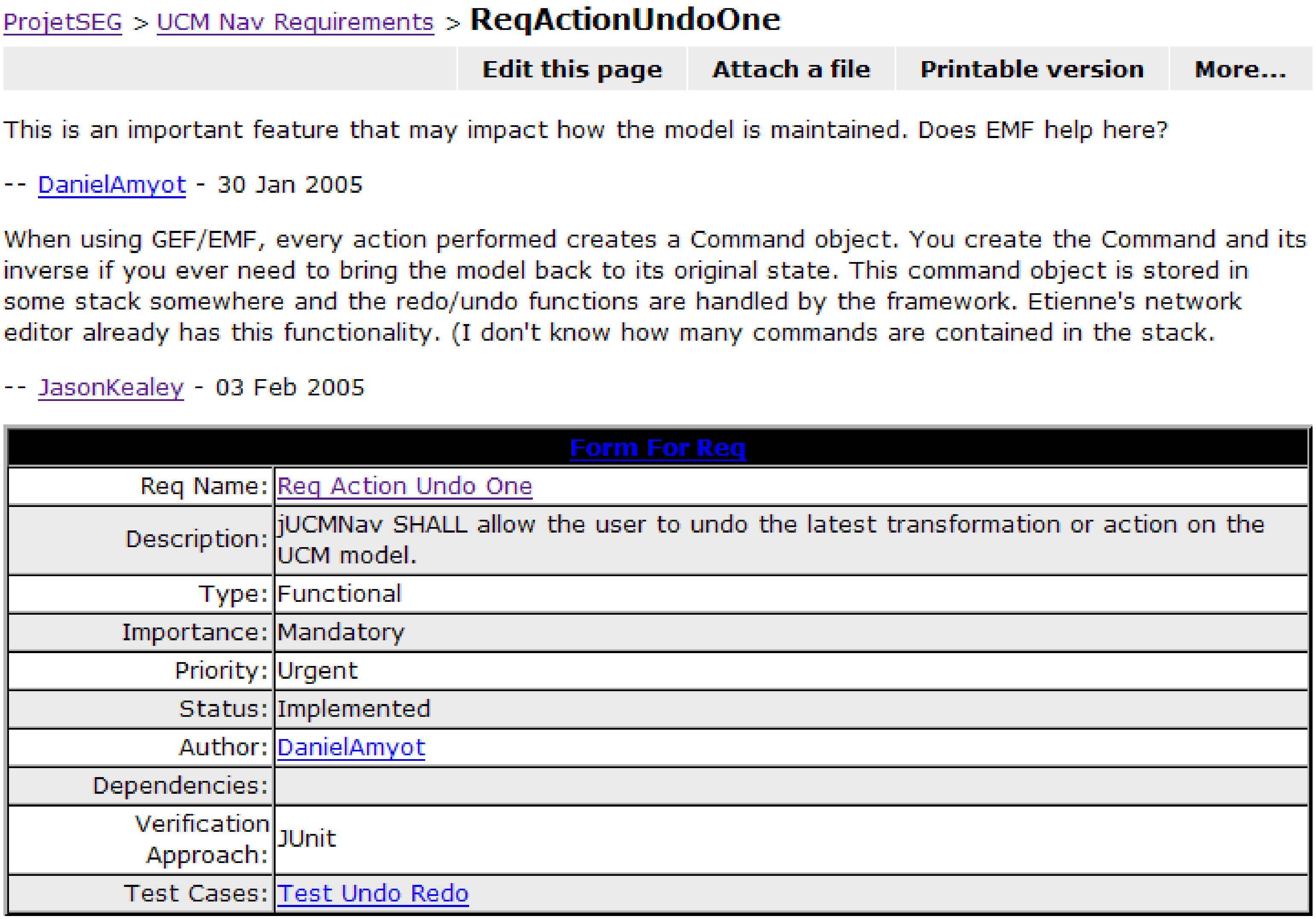


Current Requirements

- Add a new requirement: Req Create
  (Requirement name format is ReqNameOfRequirement)

Name	Description	Type	Importance	Priority	Status	Dependencies	Date
Req Action Redo Many	jUCMNav SHOULD allow the user to redo many of the latest transformations or actions undone on the UCM model.	Functional	Optional	Low	Implemented	Req Action Undo Many, Req Action Redo One	13 Mar 2005 - 19:58
Req Action Redo One	jUCMNav SHOULD allow the user to redo the latest undone transformation or action on the UCM model.	Functional	Optional	Low	Implemented	Req Action Undo One	13 Mar 2005 - 20:01
Req Action Undo Many	jUCMNav SHOULD allow the user to undo many of the latest transformations or actions on the UCM model.	Functional	Optional	Medium	Implemented	Req Action Undo One	13 Mar 2005 - 20:01
Req Action Undo One	jUCMNav SHALL allow the user to undo the latest transformation or action on the UCM	Functional	Mandatory	Urgent	Implemented		13 Mar 2005 -

Twiki – Requirement Example



ProjetSEG > UCM Nav Requirements > **ReqActionUndoOne**

This is an important feature that may impact how the model is maintained. Does EMF help here?

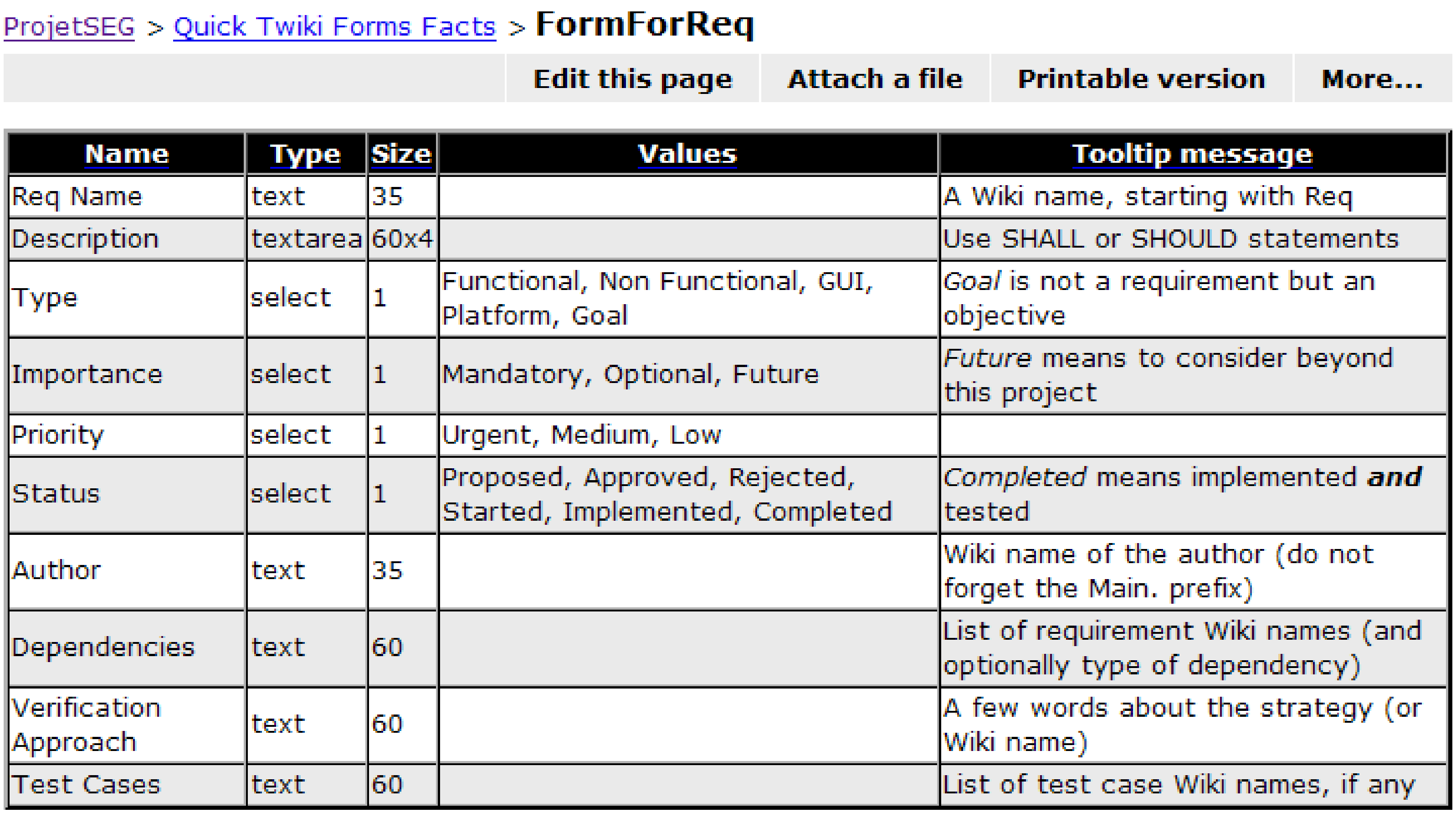
-- DanielAmyot - 30 Jan 2005

When using GEF/EMF, every action performed creates a Command object. You create the Command and its inverse if you ever need to bring the model back to its original state. This command object is stored in some stack somewhere and the redo/undo functions are handled by the framework. Etienne's network editor already has this functionality. (I don't know how many commands are contained in the stack.

-- JasonKealey - 03 Feb 2005

Form For Req	
Req Name:	Req Action Undo One
Description:	jUCMNav SHALL allow the user to undo the latest transformation or action on the UCM model.
Type:	Functional
Importance:	Mandatory
Priority:	Urgent
Status:	Implemented
Author:	DanielAmyot
Dependencies:	
Verification Approach:	JUnit
Test Cases:	Test Undo Redo

TWiki – Requirement Form Example



ProjetSEG > Quick Twiki Forms Facts > **FormForReq**

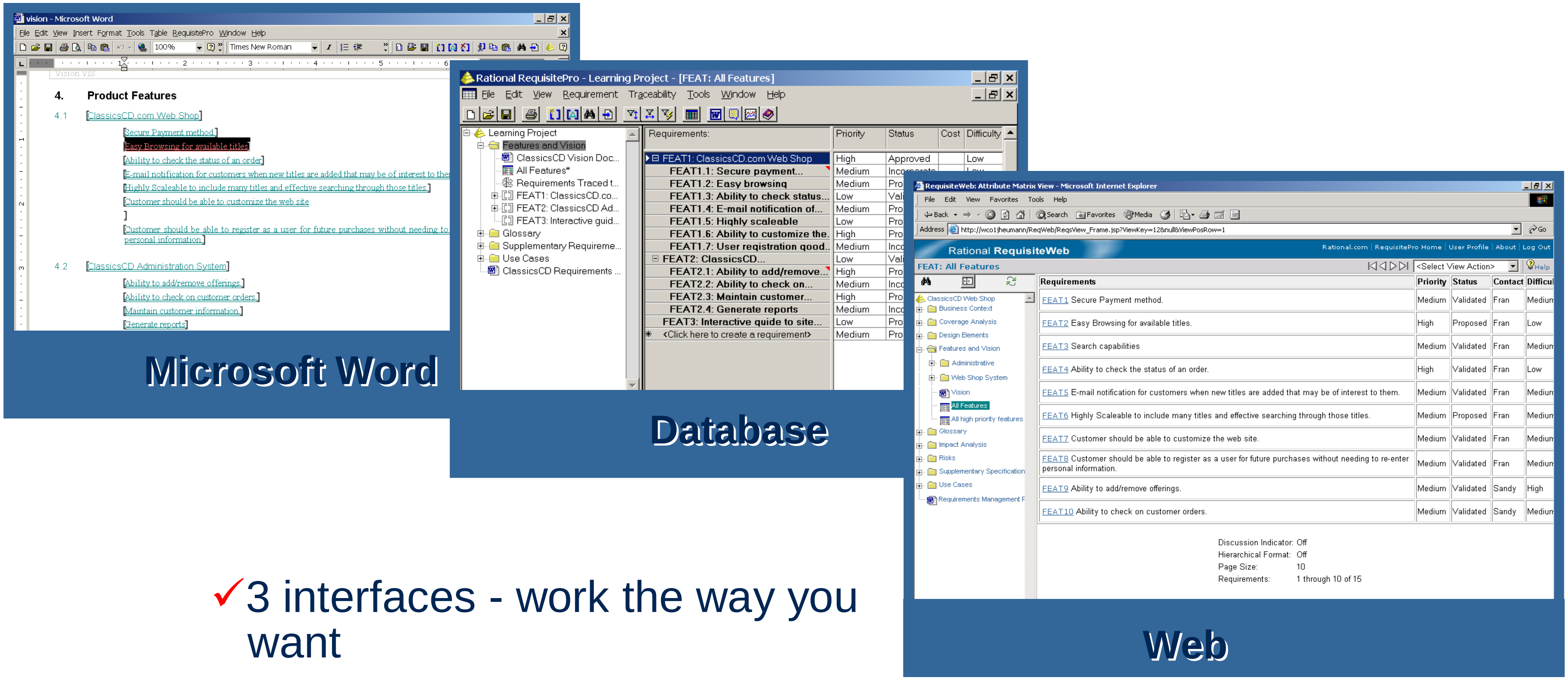
Name	Type	Size	Values	Tooltip message
Req Name	text	35		A Wiki name, starting with Req
Description	textarea	60x4		Use SHALL or SHOULD statements
Type	select	1	Functional, Non Functional, GUI, Platform, Goal	*Goal* is not a requirement but an objective
Importance	select	1	Mandatory, Optional, Future	*Future* means to consider beyond this project
Priority	select	1	Urgent, Medium, Low	
Status	select	1	Proposed, Approved, Rejected, Started, Implemented, Completed	*Completed* means implemented ***and*** tested
Author	text	35		Wiki name of the author (do not forget the Main. prefix)
Dependencies	text	60		List of requirement Wiki names (and optionally type of dependency)
Verification Approach	text	60		A few words about the strategy (or Wiki name)
Test Cases	text	60		List of test case Wiki names, if any

Using TWiki…

- We have:
  - Requirement types description with configurable statuses & attributes
  - Bidirectional links (WikiWords)
  - Configurable requests, filtering, reports
  - Access control and version management (showing differences)
  - Change management (again with forms, process, etc.)
  - Discussions, attachment of documents/images
  - Export (HTML)
  - Scripting language (Perl)
- But do we really have:
  - Graphical view of traceability?
  - Editable tables (à la Excel/Word)?
  - Baselines? Tool integration? Imports? Analysis?

IBM Requisite Pro

✓Keep your team on track



vision - Microsoft Word

File Edit View Insert Format Tools Table RequisitePro Window Help
100%
Times New Roman
Vision.VIS

4. **Product Features**

4.1 [ClassicsCD.com Web Shop]

[Secure Payment method.]

[Easy Browsing for available titles]

[Ability to check the status of an order]

[E-mail notification for customers when new titles are added that may be of interest to ther

[Highly Scaleable to include many titles and effective searching through those titles.]

[Customer should be able to customize the web site

]

[Customer should be able to register as a user for future purchases without needing to
personal information.]

4.2 [ClassicsCD Administration System]

[Ability to add/remove offerings.]

[Ability to check on customer orders.]

[Maintain customer information.]

[Generate reports]

Microsoft Word

Rational RequisitePro - Learning Project - [FEAT: All Features]

Requirements:	Priority	Status	Cost	Difficulty
FEAT1: ClassicsCD.com Web Shop	High	Approved		Low
FEAT1.1: Secure payment...	Medium	Inco		
FEAT1.2: Easy browsing	Medium	Pro		
FEAT1.3: Ability to check status...	Low	Vali		
FEAT1.4: E-mail notification of...	Medium	Pro		
FEAT1.5: Highly scaleable	Low	Pro		
FEAT1.6: Ability to customize the.	High	Pro		
FEAT1.7: User registration good..	Medium	Inco		
FEAT2: ClassicsCD...	Low	Vali		
FEAT2.1: Ability to add/remove...	High	Pro		
FEAT2.2: Ability to check on...	Medium	Inco		
FEAT2.3: Maintain customer...	High	Pro		
FEAT2.4: Generate reports	Medium	Inco		
FEAT3: Interactive guide to site...	Low	Pro		
<Click here to create a requirement>	Medium	Pro		

Database

RequisiteWeb: Attribute Matrix View - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://wco1jheumann/ReqWeb/ReqsView_Frame.jsp?ViewKey=12&null&ViewPosRow=1
Go
Rational RequisiteWeb
Rational.com | RequisitePro Home | User Profile | About | Log Out
FEAT: All Features
<Select View Action>
Help
ClassicsCD Web Shop
Business Context
Coverage Analysis
Design Elements
Features and Vision
Administrative
Web Shop System
Vision
All Features
All high priority features
Glossary
Impact Analysis
Risks
Supplementary Specification
Use Cases
Requirements Management P

Requirements	Priority	Status	Contact	Difficul
FEAT1 Secure Payment method.	Medium	Validated	Fran	Medium
FEAT2 Easy Browsing for available titles.	High	Proposed	Fran	Low
FEAT3 Search capabilities	Medium	Validated	Fran	Medium
FEAT4 Ability to check the status of an order.	High	Validated	Fran	Low
FEAT5 E-mail notification for customers when new titles are added that may be of interest to them.	Medium	Validated	Fran	Medium
FEAT6 Highly Scaleable to include many titles and effective searching through those titles.	Medium	Proposed	Fran	Medium
FEAT7 Customer should be able to customize the web site.	Medium	Validated	Fran	Medium
FEAT8 Customer should be able to register as a user for future purchases without needing to re-enter personal information.	Medium	Validated	Fran	Medium
FEAT9 Ability to add/remove offerings.	Medium	Validated	Sandy	High
FEAT10 Ability to check on customer orders.	Medium	Validated	Sandy	Medium

Discussion Indicator: Off
Hierarchical Format: Off
Page Size: 10
Requirements: 1 through 10 of 15

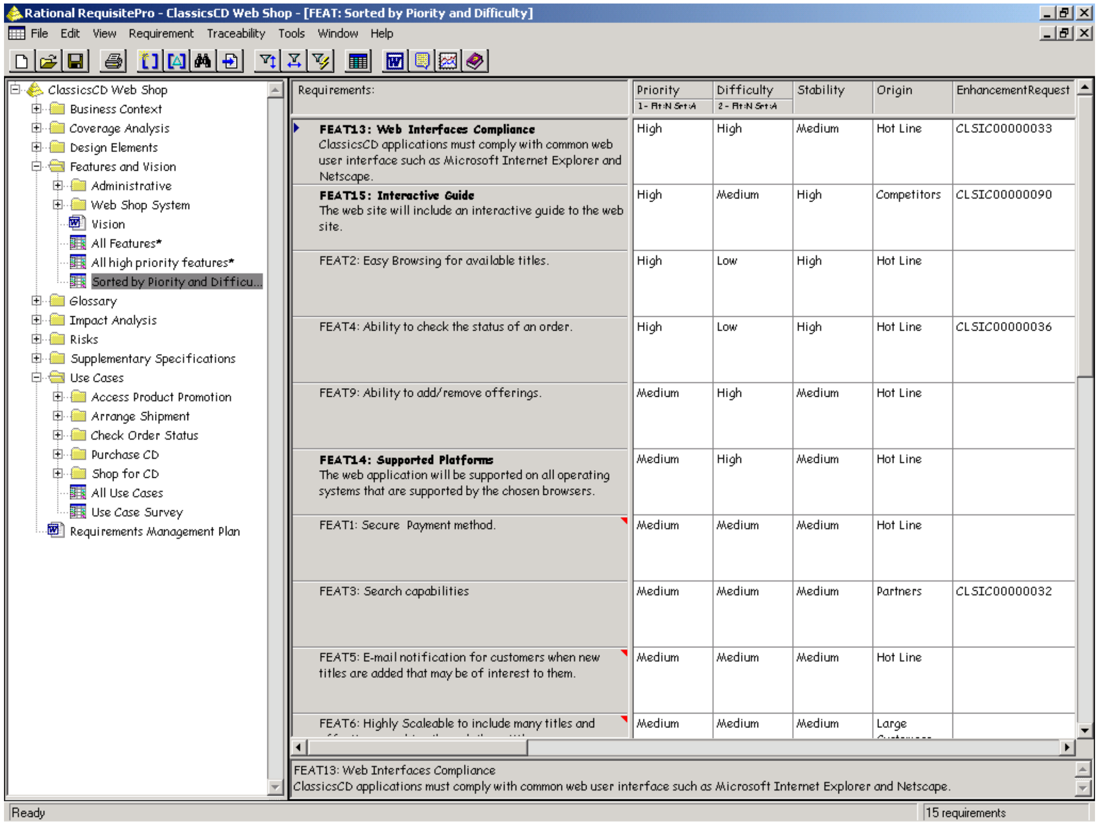
Web

✓3 interfaces - work the way you want

✓Document centric or database centric - your choice

IBM Requisite Pro – Types, Attributes, and Views

- ✓ User defined requirement types
- ✓ User defined attributes
- ✓ User defined filters (views)
- ✓ Saved views



Rational RequisitePro - ClassicsCD Web Shop - [FEAT: Sorted by Piority and Difficulty]

- ClassicsCD Web Shop
  - Business Context
  - Coverage Analysis
  - Design Elements
  - Features and Vision
    - Administrative
    - Web Shop System
    - Vision
    - All Features*
    - All high priority features*
    - Sorted by Piority and Difficu...
  - Glossary
  - Impact Analysis
  - Risks
  - Supplementary Specifications
  - Use Cases
    - Access Product Promotion
    - Arrange Shipment
    - Check Order Status
    - Purchase CD
    - Shop for CD
    - All Use Cases
    - Use Case Survey
  - Requirements Management Plan

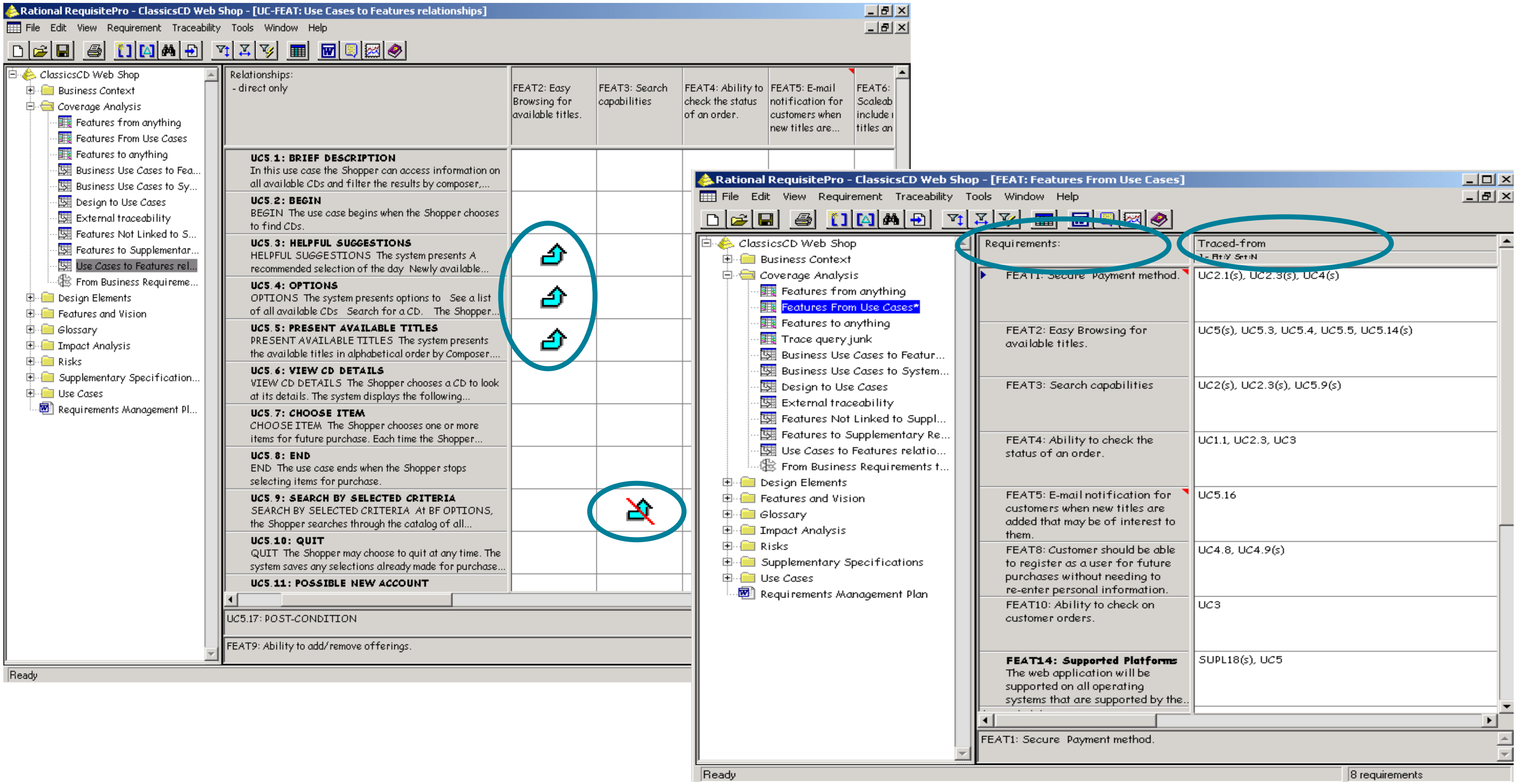
Requirements:	Priority 1- Rt:N Srt:A	Difficulty 2 - Rt:N Srt:A	Stability	Origin	EnhancementRequest
FEAT13: Web Interfaces Compliance ClassicsCD applications must comply with common web user interface such as Microsoft Internet Explorer and Netscape.	High	High	Medium	Hot Line	CLSIC00000033
FEAT15: Interactive Guide The web site will include an interactive guide to the web site.	High	Medium	High	Competitors	CLSIC00000090
FEAT2: Easy Browsing for available titles.	High	Low	High	Hot Line	
FEAT4: Ability to check the status of an order.	High	Low	High	Hot Line	CLSIC00000036
FEAT9: Ability to add/remove offerings.	Medium	High	Medium	Hot Line	
FEAT14: Supported Platforms The web application will be supported on all operating systems that are supported by the chosen browsers.	Medium	High	Medium	Hot Line	
FEAT1: Secure Payment method.	Medium	Medium	Medium	Hot Line	
FEAT3: Search capabilities	Medium	Medium	Medium	Partners	CLSIC00000032
FEAT5: E-mail notification for customers when new titles are added that may be of interest to them.	Medium	Medium	Medium	Hot Line	
FEAT6: Highly Scaleable to include many titles and	Medium	Medium	Medium	Large	

FEAT13: Web Interfaces Compliance
ClassicsCD applications must comply with common web user interface such as Microsoft Internet Explorer and Netscape.

Ready

15 requirements

IBM Requisite Pro – Traceability


File Edit View Requirement Traceability Tools Window Help
ClassicsCD Web Shop
Business Context
Coverage Analysis
Features from anything
Features From Use Cases
Features to anything
Business Use Cases to Fea...
Business Use Cases to Sy...
Design to Use Cases
External traceability
Features Not Linked to S...
Features to Supplementar...
Use Cases to Features rel...
From Business Requireme...
Design Elements
Features and Vision
Glossary
Impact Analysis
Risks
Supplementary Specification...
Use Cases
Requirements Management Pl...
Ready

Rational RequisitePro - ClassicsCD Web Shop - [UC-FEAT: Use Cases to Features relationships]

Relationships: - direct only	FEAT2: Easy Browsing for available titles.	FEAT3: Search capabilities	FEAT4: Ability to check the status of an order.	FEAT5: E-mail notification for customers when new titles are...	FEAT6: Scaleab include i titles an
UC5.1: BRIEF DESCRIPTION In this use case the Shopper can access information on all available CDs and filter the results by composer,...					
UC5.2: BEGIN BEGIN The use case begins when the Shopper chooses to find CDs.					
UC5.3: HELPFUL SUGGESTIONS HELPFUL SUGGESTIONS The system presents A recommended selection of the day Newly available...	↑				
UC5.4: OPTIONS OPTIONS The system presents options to See a list of all available CDs Search for a CD. The Shopper...	↑				
UC5.5: PRESENT AVAILABLE TITLES PRESENT AVAILABLE TITLES The system presents the available titles in alphabetical order by Composer....	↑				
UC5.6: VIEW CD DETAILS VIEW CD DETAILS The Shopper chooses a CD to look at its details. The system displays the following...					
UC5.7: CHOOSE ITEM CHOOSE ITEM The Shopper chooses one or more items for future purchase. Each time the Shopper...					
UC5.8: END END The use case ends when the Shopper stops selecting items for purchase.					
UC5.9: SEARCH BY SELECTED CRITERIA SEARCH BY SELECTED CRITERIA At BF OPTIONS, the Shopper searches through the catalog of all...		~~↑~~			
UC5.10: QUIT QUIT The Shopper may choose to quit at any time. The system saves any selections already made for purchase...					
UC5.11: POSSIBLE NEW ACCOUNT					

UC5.17: POST-CONDITION

FEAT9: Ability to add/remove offerings.

Rational RequisitePro - ClassicsCD Web Shop - [FEAT: Features From Use Cases]

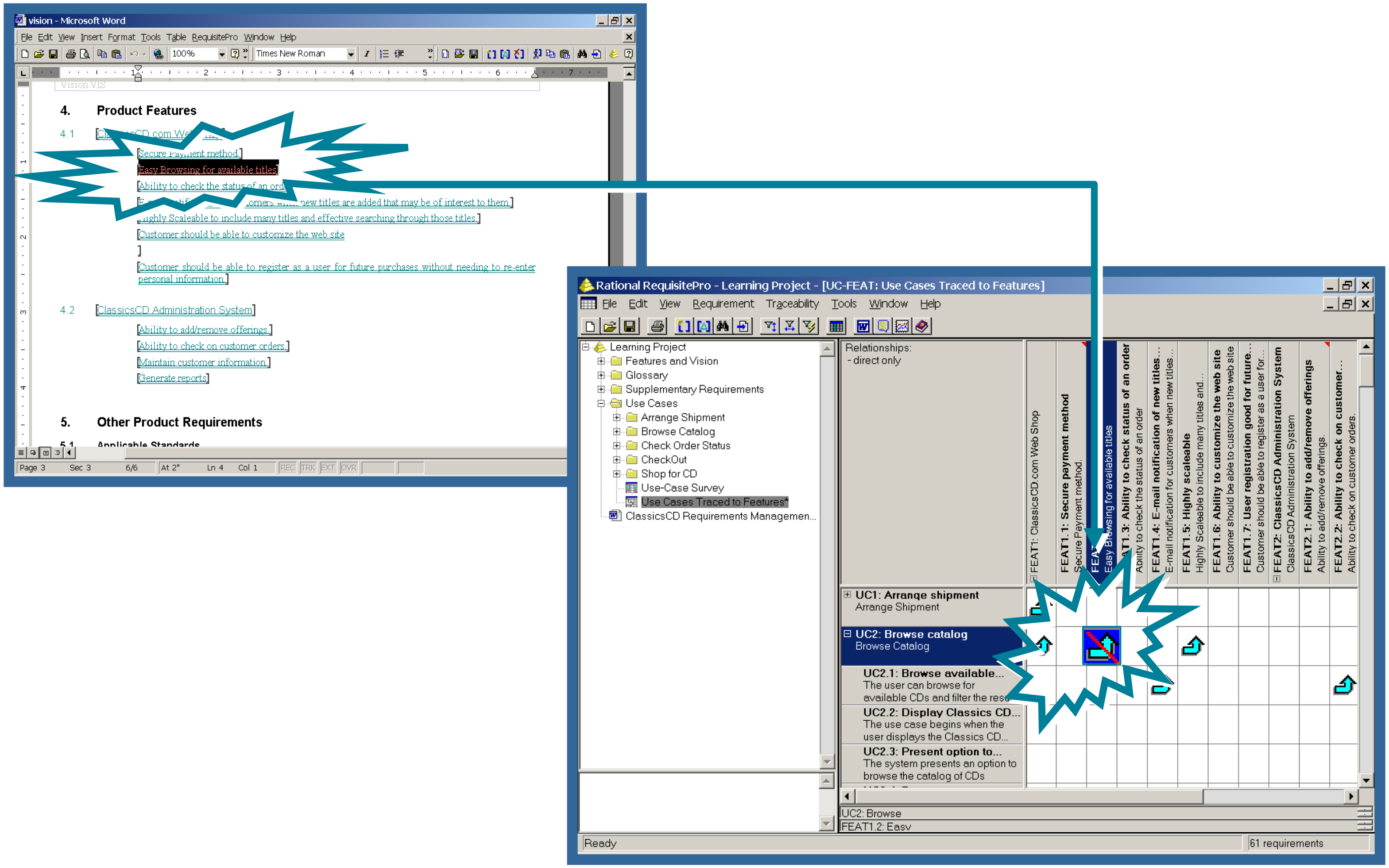
File Edit View Requirement Traceability Tools Window Help
ClassicsCD Web Shop
Business Context
Coverage Analysis
Features from anything
Features From Use Cases*
Features to anything
Trace query junk
Business Use Cases to Featur...
Business Use Cases to System...
Design to Use Cases
External traceability
Features Not Linked to Suppl...
Features to Supplementary Re...
Use Cases to Features relatio...
From Business Requirements t...
Design Elements
Features and Vision
Glossary
Impact Analysis
Risks
Supplementary Specifications
Use Cases
Requirements Management Plan
Ready

Requirements:	Traced-from 1- Flt:Y Srt:N
FEAT1: Secure Payment method.	UC2.1(s), UC2.3(s), UC4(s)
FEAT2: Easy Browsing for available titles.	UC5(s), UC5.3, UC5.4, UC5.5, UC5.14(s)
FEAT3: Search capabilities	UC2(s), UC2.3(s), UC5.9(s)
FEAT4: Ability to check the status of an order.	UC1.1, UC2.3, UC3
FEAT5: E-mail notification for customers when new titles are added that may be of interest to them.	UC5.16
FEAT8: Customer should be able to register as a user for future purchases without needing to re-enter personal information.	UC4.8, UC4.9(s)
FEAT10: Ability to check on customer orders.	UC3
FEAT14: Supported Platforms The web application will be supported on all operating systems that are supported by the..	SUPL18(s), UC5

FEAT1: Secure Payment method.

8 requirements

IBM Requisite Pro – Change Management


vision - Microsoft Word
File Edit View Insert Format Tools Table RequisitePro Window Help
100%
Times New Roman
Vision.VIS
Page 3 Sec 3 6/6 At 2" Ln 4 Col 1 REC TRK EXT OVR
Rational RequisitePro - Learning Project - [UC-FEAT: Use Cases Traced to Features]
File Edit View Requirement Traceability Tools Window Help
Learning Project
Features and Vision
Glossary
Supplementary Requirements
Use Cases
Arrange Shipment
Browse Catalog
Check Order Status
CheckOut
Shop for CD
Use-Case Survey
Use Cases Traced to Features*
ClassicsCD Requirements Managemen...
Relationships:
- direct only
UC2: Browse
FEAT1.2: Easy
Ready
61 requirements

4. Product Features

4.1 [ClassicsCD.com Web Shop]

[Secure Payment method.]

[Easy Browsing for available titles]

[Ability to check the status of an order]

[E-mail notification for customers when new titles are added that may be of interest to them.]

[Highly Scaleable to include many titles and effective searching through those titles.]

[Customer should be able to customize the web site
]

[Customer should be able to register as a user for future purchases without needing to re-enter personal information.]

4.2 [ClassicsCD Administration System]

[Ability to add/remove offerings.]

[Ability to check on customer orders.]

[Maintain customer information.]

[Generate reports]

5. Other Product Requirements

5.1 Applicable Standards

FEAT1: ClassicsCD.com Web Shop

FEAT1.1: Secure payment method
Secure Payment method.

FEAT1.2: Easy Browsing for available titles

FEAT1.3: Ability to check status of an order
Ability to check the status of an order

FEAT1.4: E-mail notification of new titles...
E-mail notification for customers when new titles...

FEAT1.5: Highly scaleable
Highly Scaleable to include many titles and...

FEAT1.6: Ability to customize the web site
Customer should be able to customize the web site

FEAT1.7: User registration good for future...
Customer should be able to register as a user for...

FEAT2: ClassicsCD Administration System
ClassicsCD Administration System

FEAT2.1: Ability to add/remove offerings
Ability to add/remove offerings.

FEAT2.2: Ability to check on customer...
Ability to check on customer orders.

UC1: Arrange shipment
Arrange Shipment

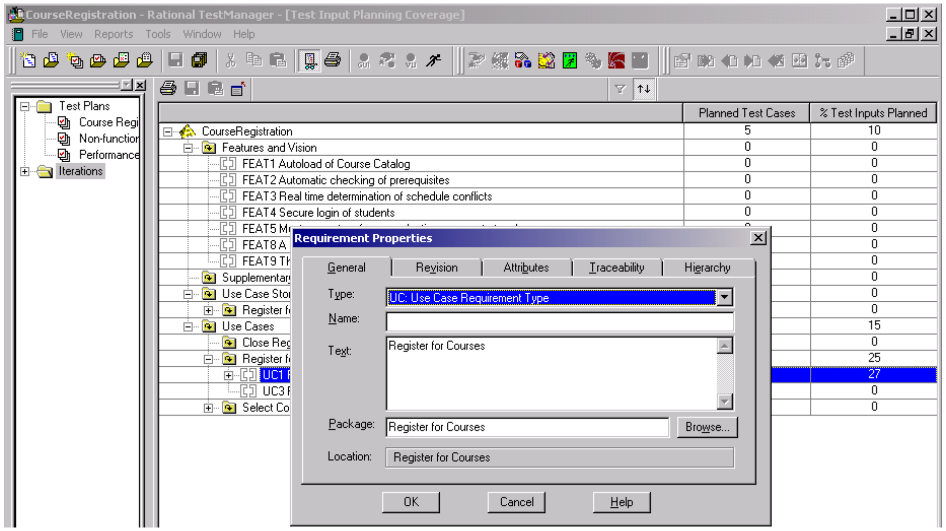
UC2: Browse catalog
Browse Catalog

UC2.1: Browse available...
The user can browse for available CDs and filter the res...

UC2.2: Display Classics CD...
The use case begins when the user displays the Classics CD...

UC2.3: Present option to...
The system presents an option to browse the catalog of CDs

IBM Requisite Pro – Integration



	Planned Test Cases	% Test Inputs Planned
CourseRegistration	5	10
Features and Vision	0	0
FEAT1 Autoload of Course Catalog	0	0
FEAT2 Automatic checking of prerequisites	0	0
FEAT3 Real time determination of schedule conflicts	0	0
FEAT4 Secure login of students	0	0
FEAT5 M		0
FEAT8 A		0
FEAT9 Th		0
Supplementary		0
Use Case Stor		0
Register f		0
Use Cases		15
Close Reg		0
Register f		25
UC1 F		27
UC3 F		0
Select Co		0

Requirement Properties

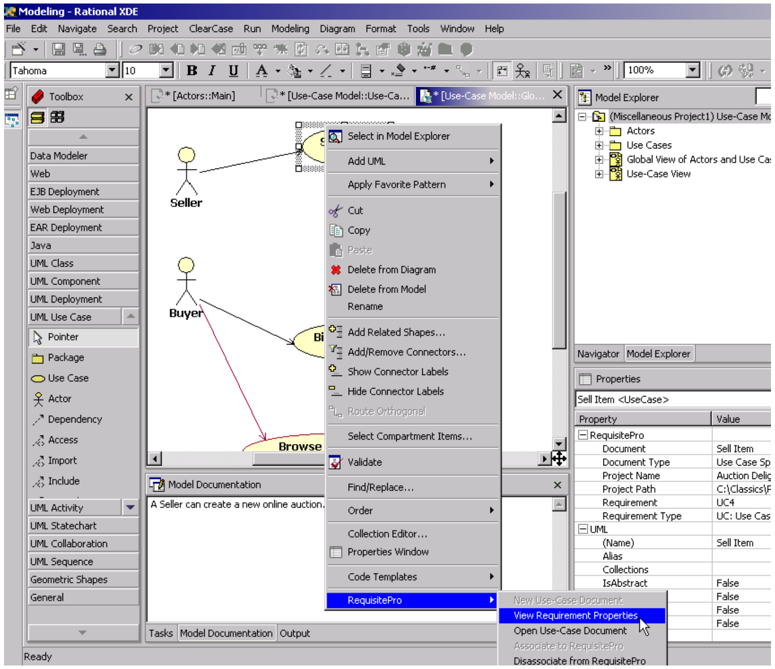
Type: UC: Use Case Requirement Type

Name:

Text: Register for Courses

Package: Register for Courses

Location: Register for Courses



Model Documentation: A Seller can create a new online auction.

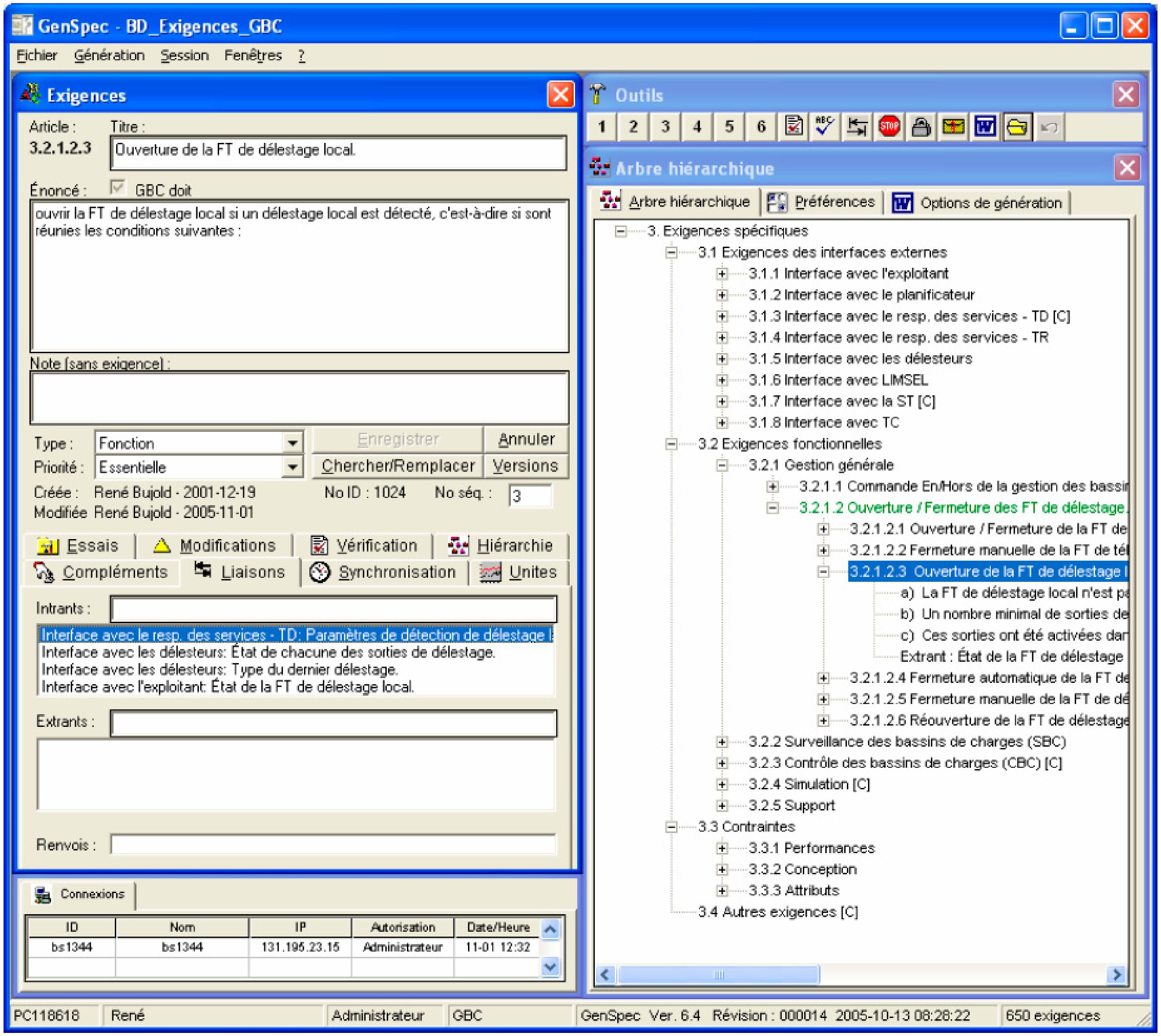
Sell Item <UseCase>

Property	Value
RequisitePro	
Document	Sell Item
Document Type	Use Case Sp
Project Name	Auction Delic
Project Path	C:\Classics\F
Requirement	UC4
Requirement Type	UC: Use Cas
UML	
(Name)	Sell Item
Alias	
Collections	
IsAbstract	False
	False
	False
	False

- ✓ IBM Rational TestManager
- ✓ Testers view current state of requirements from their tool

- ✓ IBM Rational XDE and IBM Rational Rose, Rational Software Architect and Rational Software Modeler
- ✓ Developers view current state of requirements from their tool

Genspec



GenSpec - BD_Exigences_GBC
Fichier Génération Session Fenêtres ?
Exigences
Outils
1 2 3 4 5 6
Arbre hiérarchique
Arbre hiérarchique Préférences Options de génération
Enregistrer Annuler
Chercher/Remplacer Versions
Essais Modifications Vérification Hiérarchie
Compléments Liaisons Synchronisation Unites
Connexions

Article : 3.2.1.2.3

Titre : Ouverture de la FT de délestage local.

Énoncé : ☑ GBC doit

ouvrir la FT de délestage local si un délestage local est détecté, c'est-à-dire si sont réunies les conditions suivantes :

Note (sans exigence) :

Type : Fonction

Priorité : Essentielle

Créée : René Bujold - 2001-12-19

Modifiée René Bujold - 2005-11-01

No ID : 1024 No séq. : 3

Intrants :

Interface avec le resp. des services - TD: Paramètres de détection de délestage l
Interface avec les délesteurs: État de chacune des sorties de délestage.
Interface avec les délesteurs: Type du dernier délestage.
Interface avec l'exploitant: État de la FT de délestage local.

Extrants :

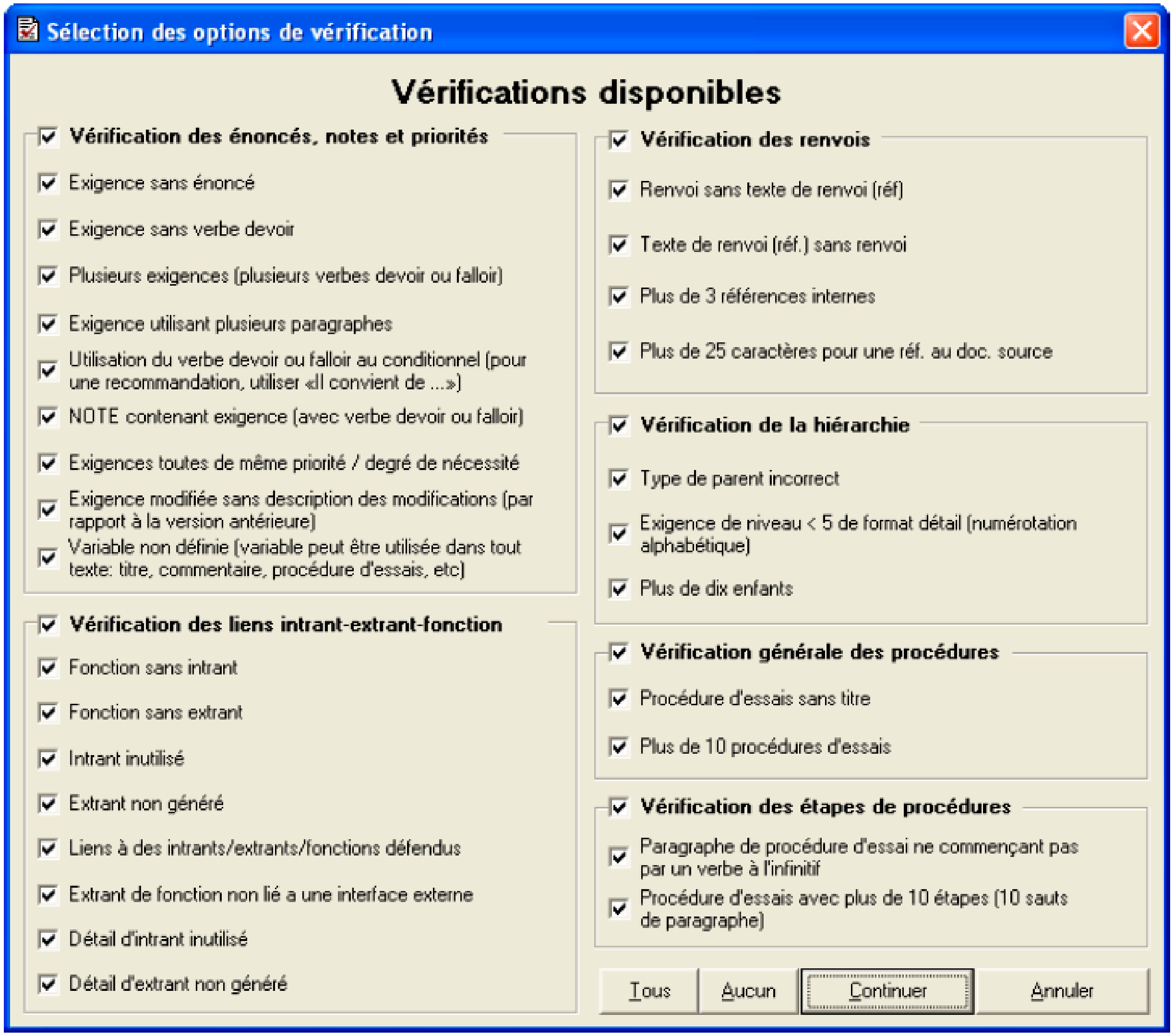
Renvois :

ID	Nom	IP	Autorisation	Date/Heure
bs1344	bs1344	131.195.23.15	Administrateur	11-01 12:32

PC118618 | René | Administrateur | GBC | GenSpec Ver. 6.4 Révision : 000014 2005-10-13 08:28:22 | 650 exigences

- 3. Exigences spécifiques
  - 3.1 Exigences des interfaces externes
    - 3.1.1 Interface avec l'exploitant
    - 3.1.2 Interface avec le planificateur
    - 3.1.3 Interface avec le resp. des services - TD [C]
    - 3.1.4 Interface avec le resp. des services - TR
    - 3.1.5 Interface avec les délesteurs
    - 3.1.6 Interface avec LIMSEL
    - 3.1.7 Interface avec la ST [C]
    - 3.1.8 Interface avec TC
  - 3.2 Exigences fonctionnelles
    - 3.2.1 Gestion générale
      - 3.2.1.1 Commande En/Hors de la gestion des bassir
      - 3.2.1.2 Ouverture / Fermeture des FT de délestage.
        - 3.2.1.2.1 Ouverture / Fermeture de la FT de
        - 3.2.1.2.2 Fermeture manuelle de la FT de tél
        - 3.2.1.2.3 Ouverture de la FT de délestage l
          - a) La FT de délestage local n'est pa
          - b) Un nombre minimal de sorties de
          - c) Ces sorties ont été activées dar
          - Extrant : État de la FT de délestage
        - 3.2.1.2.4 Fermeture automatique de la FT de
        - 3.2.1.2.5 Fermeture manuelle de la FT de dé
        - 3.2.1.2.6 Réouverture de la FT de délestage
    - 3.2.2 Surveillance des bassins de charges (SBC)
    - 3.2.3 Contrôle des bassins de charges (CBC) [C]
    - 3.2.4 Simulation [C]
    - 3.2.5 Support
  - 3.3 Contraintes
    - 3.3.1 Performances
    - 3.3.2 Conception
    - 3.3.3 Attributs
  - 3.4 Autres exigences [C]

Genspec – Automated Inspection of Specification



Vérifications disponibles

☑ **Vérification des énoncés, notes et priorités**

- ☑ Exigence sans énoncé
- ☑ Exigence sans verbe devoir
- ☑ Plusieurs exigences (plusieurs verbes devoir ou falloir)
- ☑ Exigence utilisant plusieurs paragraphes
- ☑ Utilisation du verbe devoir ou falloir au conditionnel (pour une recommandation, utiliser «Il convient de …»)
- ☑ NOTE contenant exigence (avec verbe devoir ou falloir)
- ☑ Exigences toutes de même priorité / degré de nécessité
- ☑ Exigence modifiée sans description des modifications (par rapport à la version antérieure)
- ☑ Variable non définie (variable peut être utilisée dans tout texte: titre, commentaire, procédure d'essais, etc)

☑ **Vérification des liens intrant-extrant-fonction**

- ☑ Fonction sans intrant
- ☑ Fonction sans extrant
- ☑ Intrant inutilisé
- ☑ Extrant non généré
- ☑ Liens à des intrants/extrants/fonctions défendus
- ☑ Extrant de fonction non lié a une interface externe
- ☑ Détail d'intrant inutilisé
- ☑ Détail d'extrant non généré

☑ **Vérification des renvois**

- ☑ Renvoi sans texte de renvoi (réf)
- ☑ Texte de renvoi (réf.) sans renvoi
- ☑ Plus de 3 références internes
- ☑ Plus de 25 caractères pour une réf. au doc. source

☑ **Vérification de la hiérarchie**

- ☑ Type de parent incorrect
- ☑ Exigence de niveau < 5 de format détail (numérotation alphabétique)
- ☑ Plus de dix enfants

☑ **Vérification générale des procédures**

- ☑ Procédure d'essais sans titre
- ☑ Plus de 10 procédures d'essais

☑ **Vérification des étapes de procédures**

- ☑ Paragraphe de procédure d'essai ne commençant pas par un verbe à l'infinitif
- ☑ Procédure d'essais avec plus de 10 étapes (10 sauts de paragraphe)