

Requirements Verification

Preparing for Verification Activities

Leyna Cotran

leynacotran@gmail.com

IN4MTX 113

March 3, 2011

Today's Topics

- Preparing for Verification Activities
 - Inspections and Reviews (Ch. 5)
 - Queries using a Requirements Database
 - Verification through Formal Checks (Ch. 5)
- Verification Activities
 - Definitions of V&V
 - Why do we care?
 - Verification Methods
 - The VCRM

**Congrats. You have completed 2
use case assignments.**

... but who checked it?

3

Why Have Inspections and Reviews?

- Validate that the RD:
 - Reflects stakeholder needs
 - Explains the system accurately
 - Get feedback
 - Progress
 - Customer feedback
 - Subject matter experts that can catch your mistakes
- End goal: have an accurate, complete, and consolidated RD

4

Requirements Inspection Process

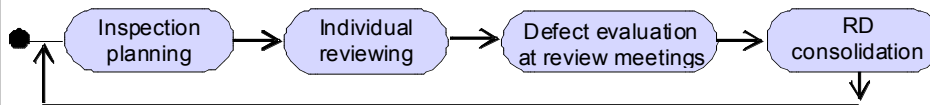


Figure 5.1 – Requirements inspection, review, and consolidation

5

Requirements Inspection Process

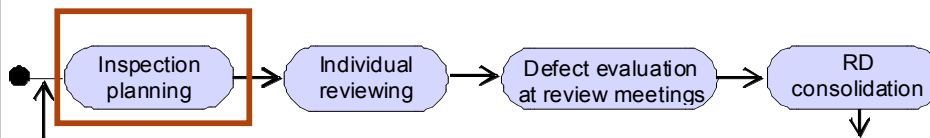


Figure 5.1 – Requirements inspection, review, and consolidation

6

Inspection Planning

- *Different terms:* Walkthroughs, Colleague Reviews, Peer Reviews
 - Subtle differences involving scope and format of these reviews
 - Which one to use?
- Planning involves the basics: schedule, scope, have your document ready to review, decide who is getting invited, find a conference room, etc

7

Some Guidance to Inspection Planning

- Time it well
- Limit the number of people attending: key experts and stakeholders only (max: 7, min: 3)
- Don't invite any manager
- Give your reviewers enough time
- Get your comments ahead of time
- Customers are tricky

8

Requirements Inspection Process

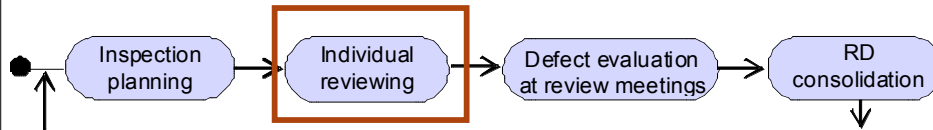


Figure 5.1 – Requirements inspection, review, and consolidation

9

Individual Reviewing

- Free Form/Free Style
 - No direction given
 - Find what you can
- Checklist-based
 - Specifics you want your reviewers to provide feedback: format, readability, clarity, consistency (defects), language and semantics
- Process-based
 - Assign roles
 - Seek different perspectives from specific disciplines: safety, design, test, quality

10

Example: Defect-based Checklist

- | | |
|---|---------------------------------------|
| • Omission: | Was something greatly missed? |
| • Contradiction: requirements/concepts? | Consistent with other |
| • Inadequacy: | Did this meet the stakeholders needs? |
| • Ambiguity: | Too many interpretations? |
| • Immeasurability: verifiable? | Are these requirements |
| • Noise: relevant? | Are these statements |
| • Over specification any value to verify? | Do these requirements add |
| • Unfeasibility | Is this possible? |
| • Unintelligibility here? | Why is this statement/requirement |
| • Poor Structuring | Bad wording? |
| • Forward Reference document? | Is the concept defined later in the |
| • Remorse definition? | Has the concept been used before |
| • Propagated Changes | Would a change here propagate |

11

Some Guidance to Individual Reviewing

- Ask yourself: what are you seeking?
 - Technical accuracy?
 - Clarity in wording?
 - ... then ask your reviewers for the same.
- Providing direction will yield the best results: go with checklist-based or process-based reviews.

12

Requirements Inspection Process

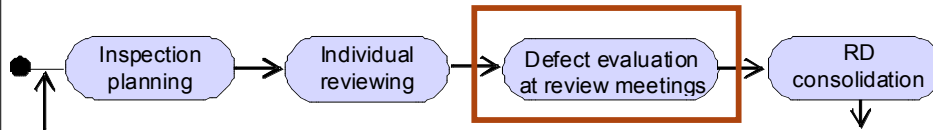


Figure 5.1 – Requirements inspection, review, and consolidation

13

Defect Evaluation at Review Meetings

- Have the inspection review meeting, collect comments.
- Tips:
 - Excel is very powerful / matrix comments, resolution, action items
 - Document the problem ... analyze later.

14

Defect Evaluation at Review Meetings

an example

D	Section#	ReqID	Description	Type of Finding	Major/Minor	Actioner	Actionee	ECD	Status
1	3.1.1	100	Add "upon user selection" for selectability of configuration	CS	Minor	Bruce H.	L. Cotran	9/4/2008	Closed
2	3.2.3	122	requirement is redundant, remove	CL	Minor	Dave C	L. Cotran	9/12/2008	Closed
3	3.1.2.2.1	164	When you dark subtract, make sure not to set subzero values to zero, set to two. Fix wording.	CL	Major	Bruce H	L. Cotran	9/12/2008	Closed
4	3.8.8	202	Ensure that the dark subtraction includes negative values. Missing.	CS	Major	Bruce H	L. Cotran	9/15/2008	Closed
5	3.9	206	Make default wafer front configuration zero	CL	Minor	Josh L	L. Cotran	9/17/2008	Closed
6	3.12	233	limits are +/- 2Hz, not 3Hz	CL	Minor	Dace C	L. Cotran	9/17/2008	Closed
7	3.13.2	242	Strike - redundant to req_096	CS	Minor	Bruce H	L. Cotran	9/17/2008	Closed

15

Requirements Inspection Process

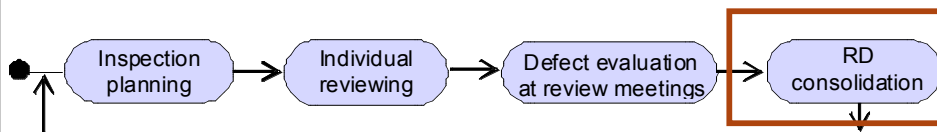


Figure 5.1 – Requirements inspection, review, and consolidation

16

RD Consolidation

- Consolidate comments
- Tip: decide your next action
 - Resolve conflicting comments
 - Defer if the conflict gets out of hand
 - Disagreeing with a comment
 - Reconcile comments with your updated RD

17

Requirements Inspection Process

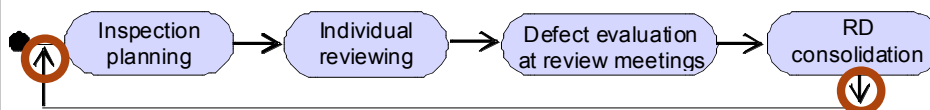
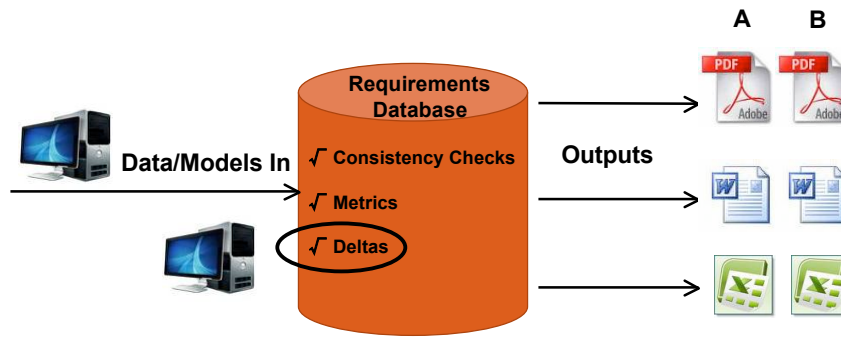


Figure 5.1 – Requirements inspection, review, and consolidation

18

Queries on a Database



19

Queries on a Database



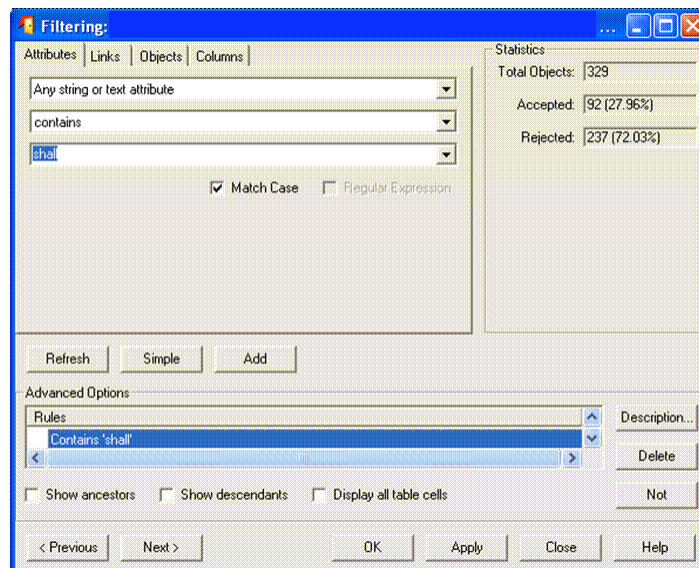
- Full Inspection Review of Version A
- Delta Inspection Review of Version B

20

Queries on a Database

- Queries can be made to determine consistency in wording and assigned entities
- Queries for metrics:
 - # of requirements: Volatile requirements comparison (Version A vs Version B)
 - # of specific requirements: i.e. how many requirements related to interfaces? Safety requirements?
 - Traceability: finding orphaned requirements, childless parents
- Deltas from one baseline of an RD to the next

21



22

Verification through Formal Checks

- Language Checks
- Dedicated Consistency and Completeness Checks
- Model Checking
- Theorem Proving

23

Verification with Model Checking

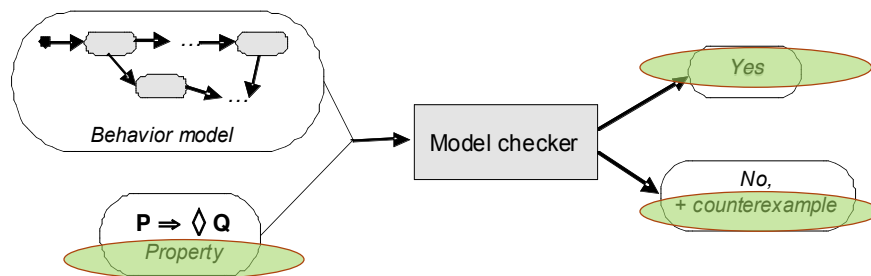


Figure 5.4 – Model checking

24

Proof by Counterexample

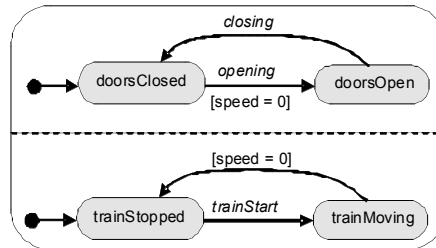


Figure 5.5 – A faulty SM model for the behavior of a controller of train doors and movements

init:	(doorsClosed, trainStopped)
start:	(doorsClosed, trainMoving)
[speed = 0]:	(doorsClosed, trainStopped)
opening:	(doorsOpen,
trainStopped)	
start:	(doorsOpen, trainMoving)

Missing from →
DoorsState =
'closed'

Requirements Verification Activities

Definitions of V&V

- Verification
 - The process of determining whether or not the products of a given phase of the software development cycle fulfill the requirements established during the previous software phase.
- Validation
 - The process of evaluating software at the end of the software development process to ensure compliance with software requirements.

These definitions are taken from : Verifying and validating software requirements and design specifications.
Boehm, BW IEEE Software, Vol. 1, no. 1, pp. 75-88. 1984

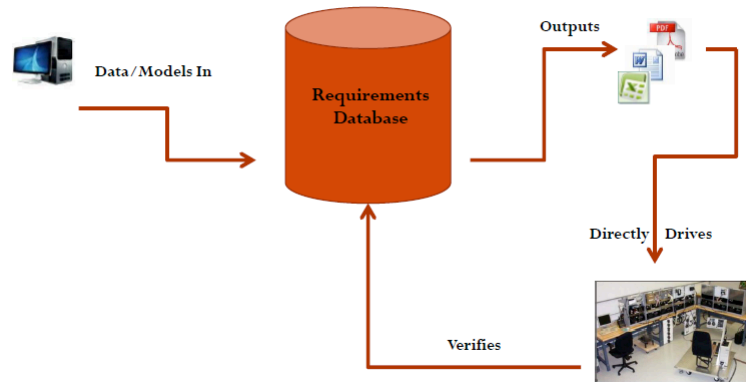
27

Basically...

- Verification: “Am I building the product right?”
- Validation: “Am I building the right product?”

28

Why do we care about verification?



29

Requirements Inspection Process

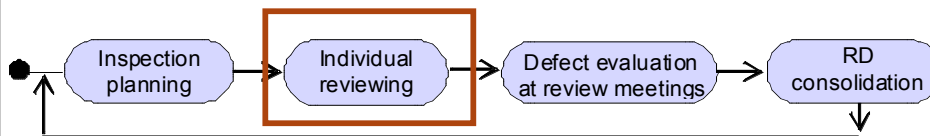


Figure 5.1 – Requirements inspection, review, and consolidation

Your testers are key stakeholders

30

Verification Methods

- Industry accepted methods are:
 - Test
 - Analysis
 - Demonstration
 - Inspection
 - S
 - Any guesses to the “S” in T.A.D.I.S?

31

The VCRM

- Verification Cross Reference Matrix
- Can be combined with the Requirements Traceability Matrix
- Supplemental to an RD
 - Its an explanation of how you will verify requirements

32