

Tool support for change impact analysis using formalization of requirements relations

by: W.Spijkerman

supervisors: A. Göknil

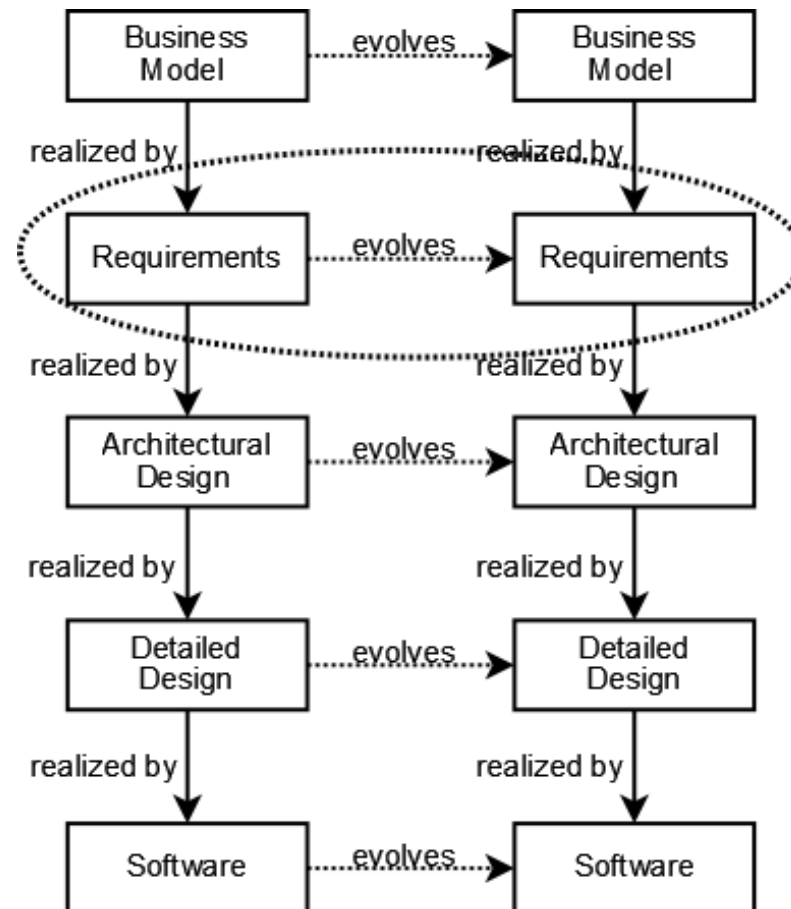
I. Kurtev

K. van den Berg

Outline

- § Scope
- § Problem
 - § Illustration of the issue
 - § Poor change impact analysis support
- § Approach
 - § Building blocks used
 - § Steps taken
- § Tool support
 - § Demonstration
 - § Future features

Scope



Outline

- § Scope
- § Problem
 - § Illustration of the issue
 - § Poor change impact analysis support
- § Approach
 - § Building blocks used
 - § Steps taken
- § Tool support
 - § Demonstration
 - § Future features

Impact explosion problem (1)

R97	The system shall allow only the administration to manage courses
R98	The system shall allow only the administration to create new courses
R99	The system shall allow only the administration to delete courses
R100	The system shall allow only the administration to update static course information
R101	The system shall allow only the administration to appoint (principal) lecturers to courses
R102	The system shall allow only the administration to specify the minimum number of students for a course. If there are too little subscriptions in a semester, that course will not be given during that semester

Impact explosion problem (2)

Change



R97	The system shall allow only the administration to manage courses
R98	The system shall allow only the administration to create new courses
R99	The system shall allow only the administration to delete courses
R100	The system shall allow only the administration to update static course information
R101	The system shall allow only the administration to appoint (principal) lecturers to courses
R102	The system shall allow only the administration to specify the minimum number of students for a course. If there are too little subscriptions in a semester, that course will not be given during that semester

Impact explosion problem (3)

Change



R97	The system shall allow only the administration to manage courses
???	R98 The system shall allow only the administration to create new courses
???	R99 The system shall allow only the administration to delete courses
???	R100 The system shall allow only the administration to update static course information
???	R101 The system shall allow only the administration to appoint (principal) lecturers to courses
???	R102 The system shall allow only the administration to specify the minimum number of students for a course. If there are too little subscriptions in a semester, that course will not be given during that semester

Problem

- § Performing change impact analysis results in imprecise results
 - § Impact explosion
 - § Many false positives
- § Current requirements management tools
 - § Have informal requirements relations
 - § Do not support (formalized) types of change

Goals

- § Providing formalized change impact rules
- § Providing tool support
- § Intension of providing *better* tool support
 - § Less false positives
 - § Hopefully a smaller impact explosion

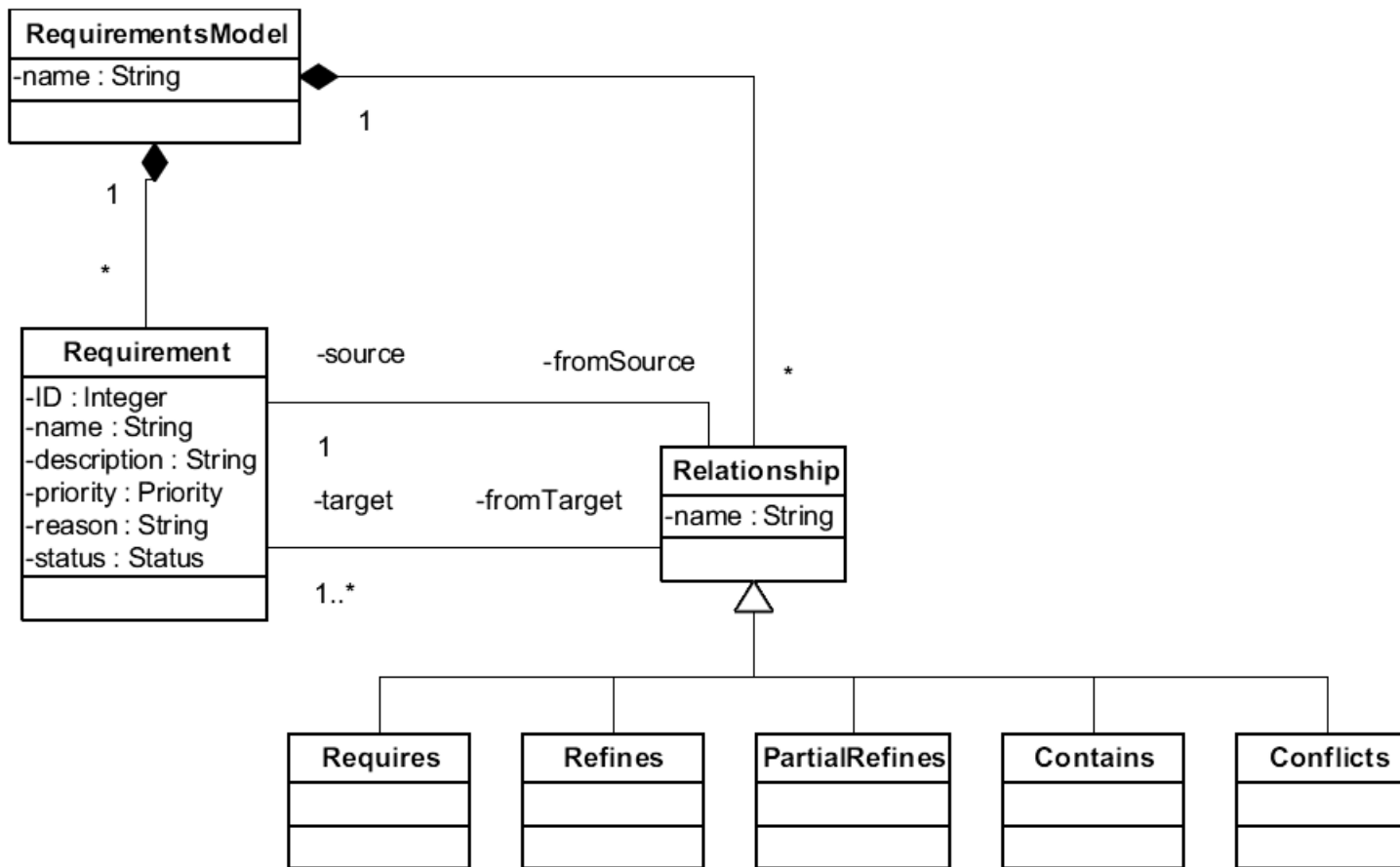
Outline

- § Scope
- § Problem
 - § Illustration of the issue
 - § Poor change impact analysis support
- § Approach
 - § Building blocks used
 - § Steps taken
- § Tool support
 - § Demonstration
 - § Future features

Approach

- § Use available formalization of requirements and relations
- § Determine requirements data model
- § Derive classification of change and formalization of change types
- § Determine decision trees
- § Use rationale of change

Approach – Requirements meta model



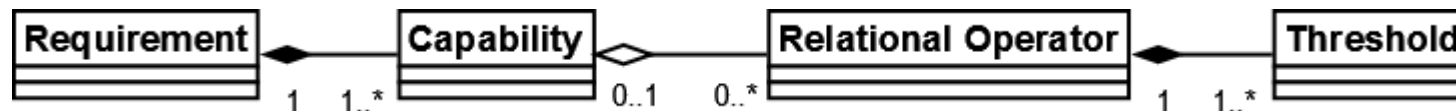
Approach – Requirements relations

- § Requirements relations description
 - § Contains: related through a part-whole hierarchy
 - § Requires: only fulfilled when related required requirement is fulfilled
 - § Refines: derived from the other by adding more details to it
 - § Partially refines: derived from the other by **only** capturing a part of the requirement, but this part in more detail
 - § Conflicts: Fulfillment of one excludes the fulfillment of the other

- § Formalized in First Order Logic

Approach – Requirements data model

§ Textual requirement primitives (Wasson)



§ Formalized requirements



Approach – Classification of change

- § Add/delete/update requirements relation
- § Add requirement
- § Delete requirement
- § Update requirement
 - § Add property to requirement
 - § Add constraint to requirement
 - § Remove property from requirement
 - § Remove constraint from requirement
 - § Change property in requirement
 - § Change constraint in requirement

Approach – Terminology

- § Rationale of change:
 - § Domain change: originates from environment that the stakeholders want to be modeled, and possibly implemented
 - § Refactoring: change in representation of requirements, which is not a result from a change in the environment
- § Proposed change
 - § Change captured in the model, but not yet applied
- § Propagated proposed change
 - § Impact caused by a (propagated) proposed change

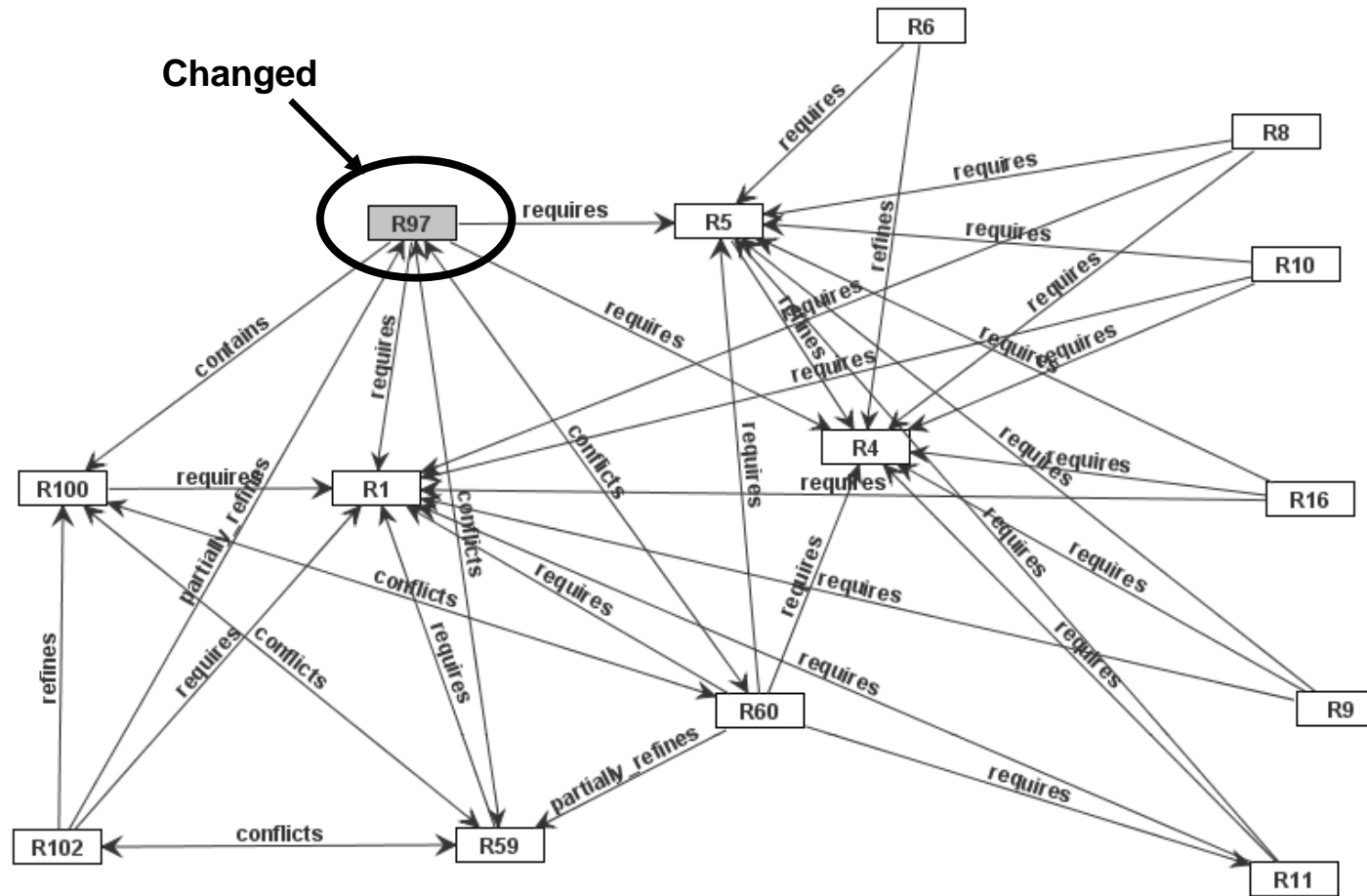
Example – Changing a requirement (1)

- § **R97** The system shall allow only the administration to manage courses.

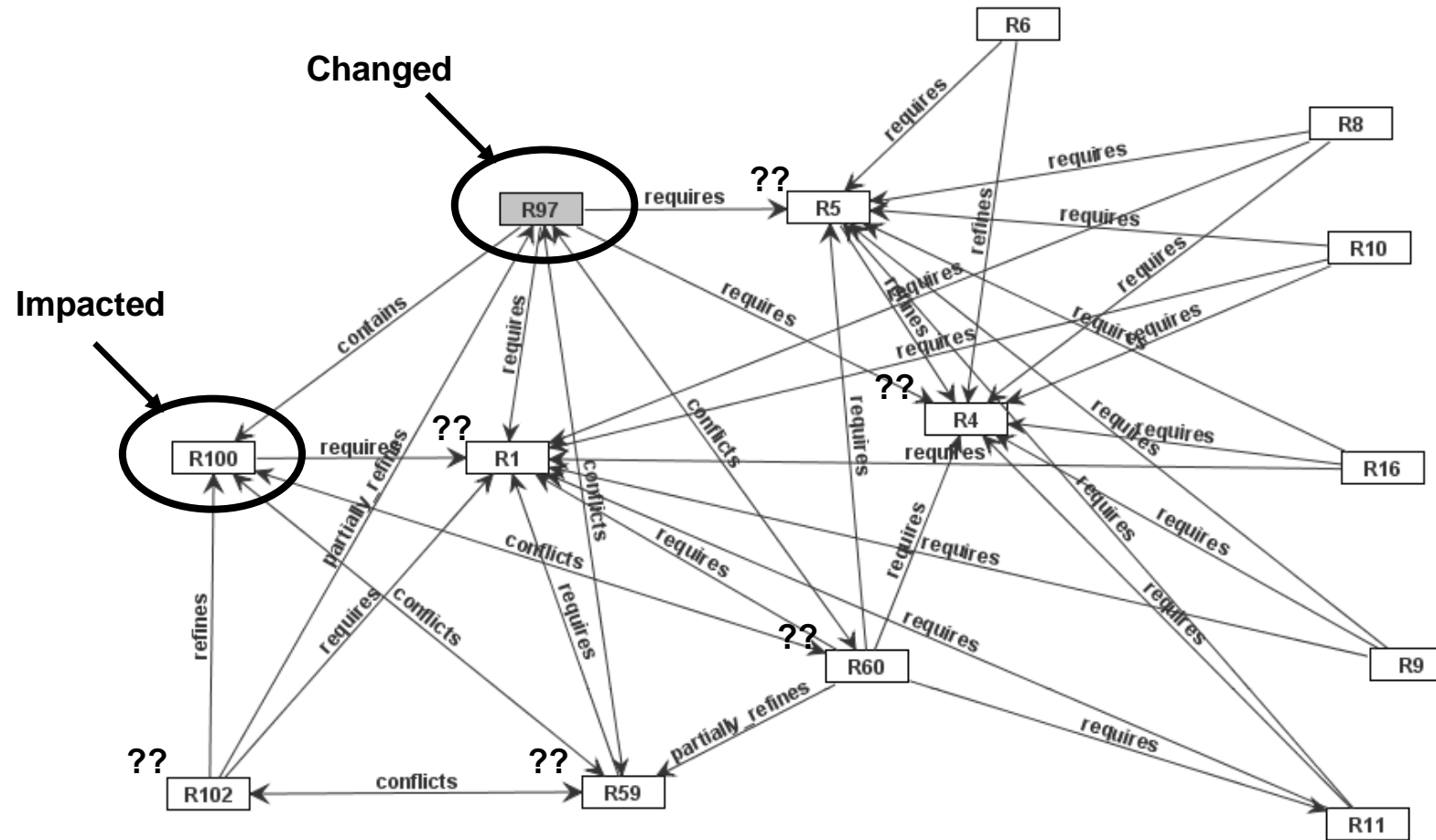
- § Textual primitives (Wasson):
 - § Capability provided: Functionality of managing courses
 - § Limitation: Only allowed to the administration
 - § Relational operator: By permission

- § (Domain) change such that the `allow only the administration' limitation is removed, is of type `remove constraint'

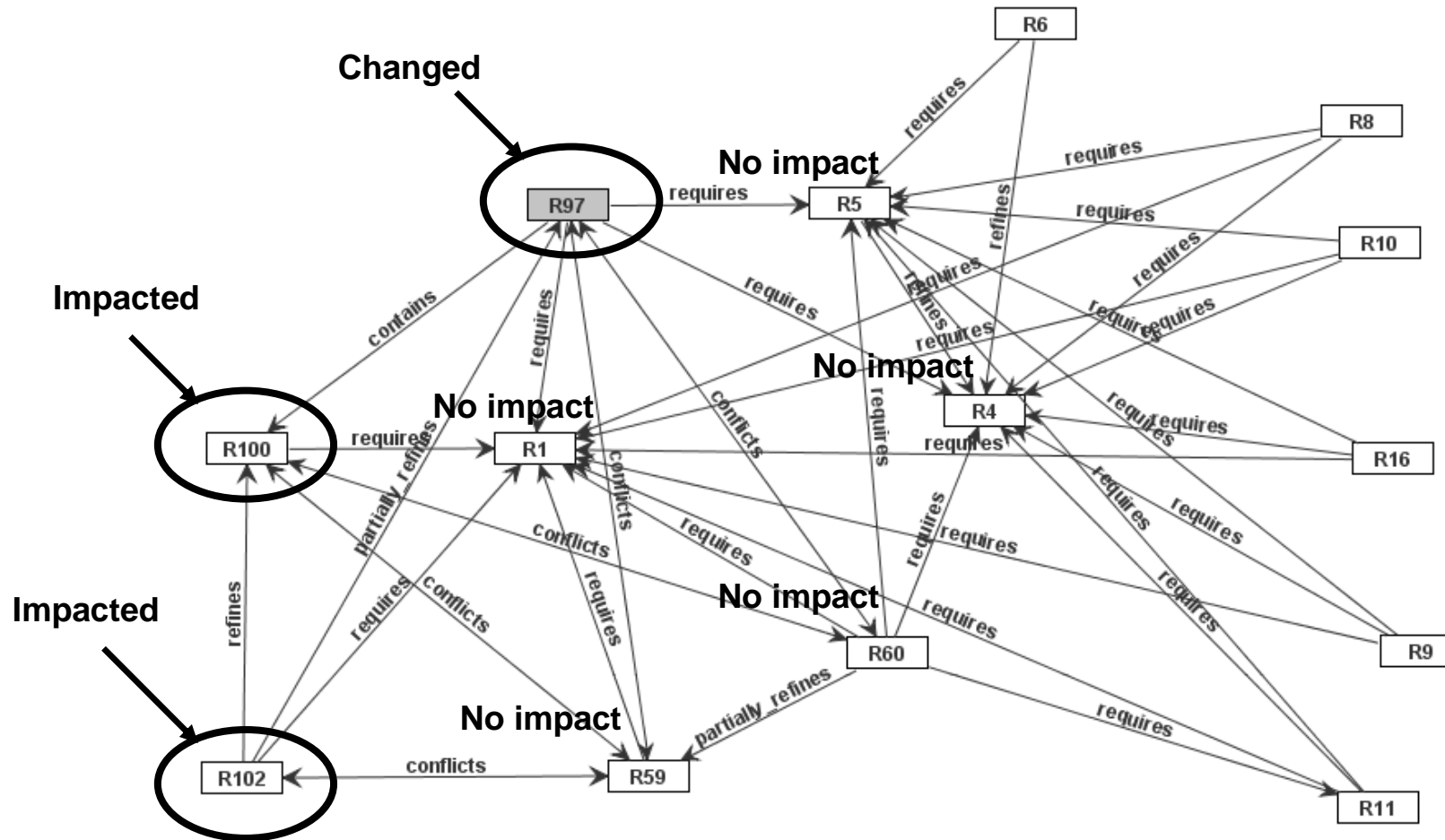
Example – Changing a requirement (3)



Example – Changing a requirement (5)

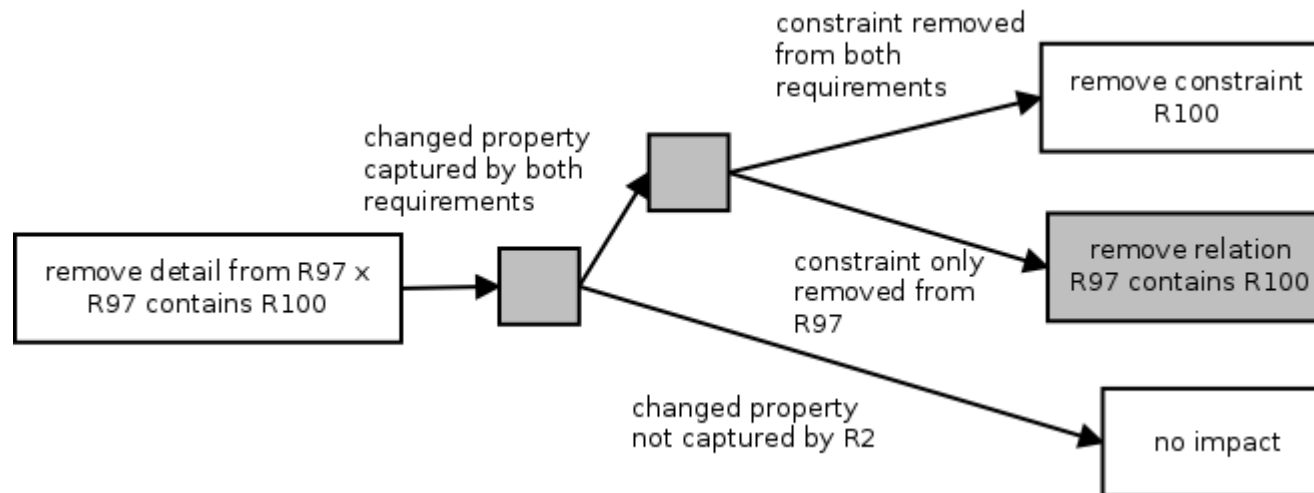


Example – Changing a requirement (6)



Decision tree for change impact analysis

- § An investigated requirement may have multiple propagation possibilities
- § Choices of which requirement to investigate next and what propagation are captured in decision trees



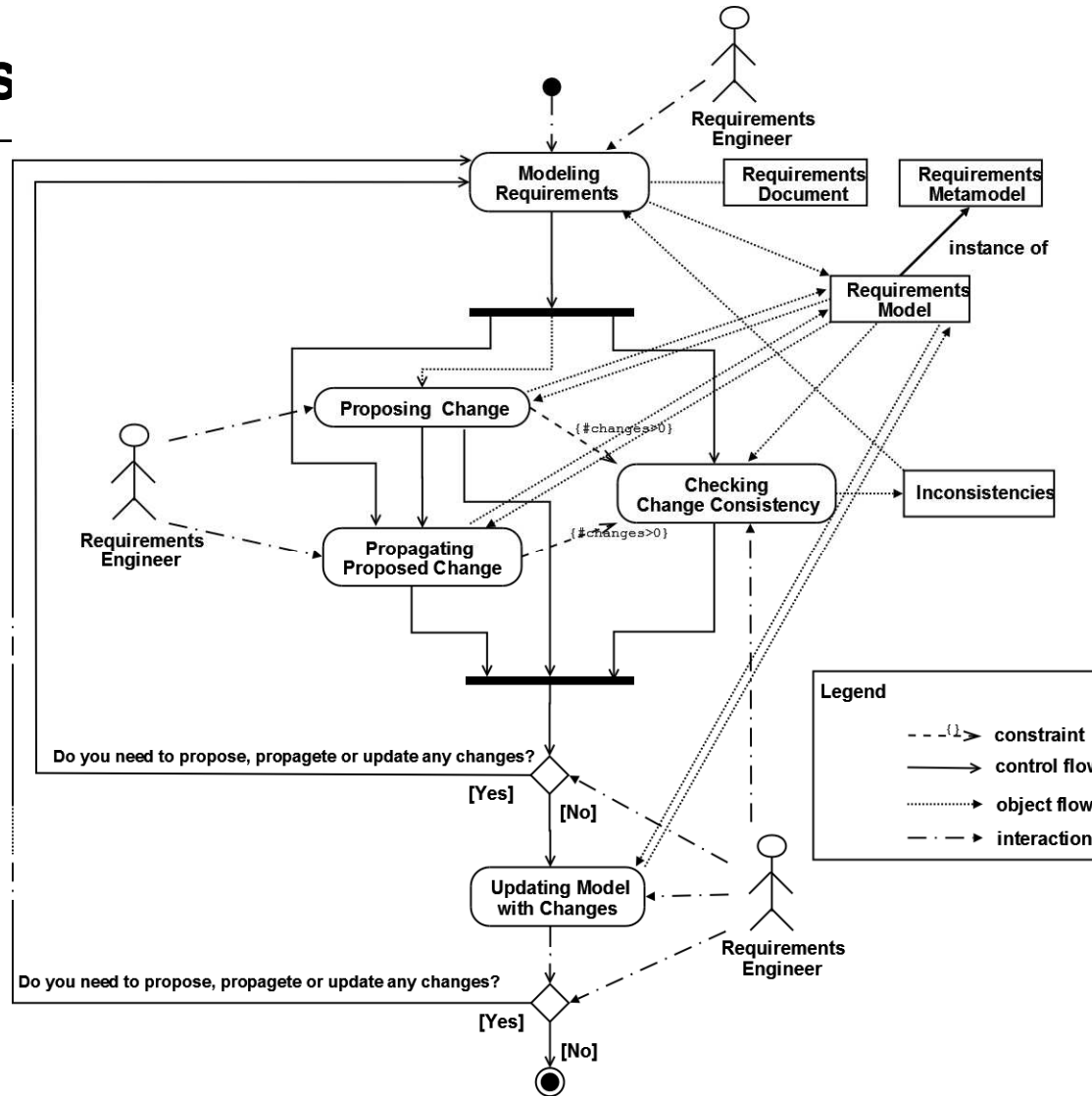
Approach – Change impact cases for domain changes (1)

Change type	R1 contains R2	R1 refines R2	R1 partrefines R2	R1 traces to R2
Add property to R1	No impact Add property to R2	Add property to R2 Delete relation	Delete relation	No impact Change R2 Remove relation
Remove property from R1	No impact Delete R2 Remove property from R2	Remove property from R2 Remove property from R2 & Remove relation	Remove property from R2	No impact Change R2 Remove relation
Change property In R1	No impact Change property In R2	Change property in R2 No impact	Change property in R2 No impact	No impact Change R2 Remove relation

Approach – Change impact cases for domain changes (2)

Change type	R1 contains R2	R1 refines R2	R1 partrefines R2	R1 traces to R2
Add property to R1	No impact Add property to R2	Add property to R2 Delete relation	Delete relation	No impact Change R2 Remove relation
Remove property from R1	No impact Delete R2 Remove property from R2	Remove property from R2 Remove property from R2 & Remove relation	Remove property from R2	No impact Change R2 Remove relation
Change property In R1	No impact Change property In R2	Change property in R2 No impact	Change property in R2 No impact	No impact Change R2 Remove relation

Modeling proces



Outline

- § Scope
- § Problem
 - § Illustration of the issue
 - § Poor change impact analysis support
- § Approach
 - § Building blocks used
 - § Steps taken
- § Tool support
 - § Demonstration
 - § Future features

Tool support

§ TRIC

- § Based on formalization of requirements relations
- § Supports inferencing of requirements relations
- § Supports consistency checking of requirement model

§ Extend TRIC

- § To support proposing changes
- § To support change impact analysis over these proposed changes
- § To support consistency checking of proposed (propagated) changes
- § To support applying proposed changes to the requirements model

Tool demonstration

1. Requirements model in TRIC
2. Example of a single proposed change to a single requirement
3. (Possible) propagation matrix
4. Propagation and determining new candidates
5. Resolving multiple paths to the same target requirement

Tool support – Features

- § Current support
 - § Propagation of proposed change *consisting of single change*
 - § Starting change set consists of a *single changed requirement*
 - § Detecting multiple paths from same source to requirement
- § Future support
 - § Propagation of proposed changes consisting of *multiple changes*
 - § Starting change set consisting of *multiple changed requirements*
 - § Indication of overlapping propagations from different changes
 - § Applying proposed change(s) to requirements model
 - § Representation of decision trees

Summary

- § We provided formalized change impact analysis rules
- § Tool is still under development
- § Is the tool support *better*?
 - § The change propagation is more precise due to change types
 - § In theory less false positives
 - § Have to do measurements to determine how impact explosion compares

- § Features intended to be implemented by august