

Enhancing IBM Requisite Pro with IR-based Traceability Recovery Features

Andrea De Lucia, Raffaele Landi, Rocco Oliveto, Genny Tortora
Department of Mathematics and Informatics, University of Salerno

84084, Fisciano (SA), Italy

adelucia@unisa.it, raffaele.landi81@gmail.com, {roliveto,
tortora}@unisa.it

Outline

- Background
 - Traceability recovery
 - Guidelines for the design of traceability recovery tools derived from empirical studies
 - Motivation: facilitate the technology transfer
- ReqTracer Pro: traceability recovery in IBM Requisite Pro
 - Architecture and functionalities
- Conclusion and future work

Background

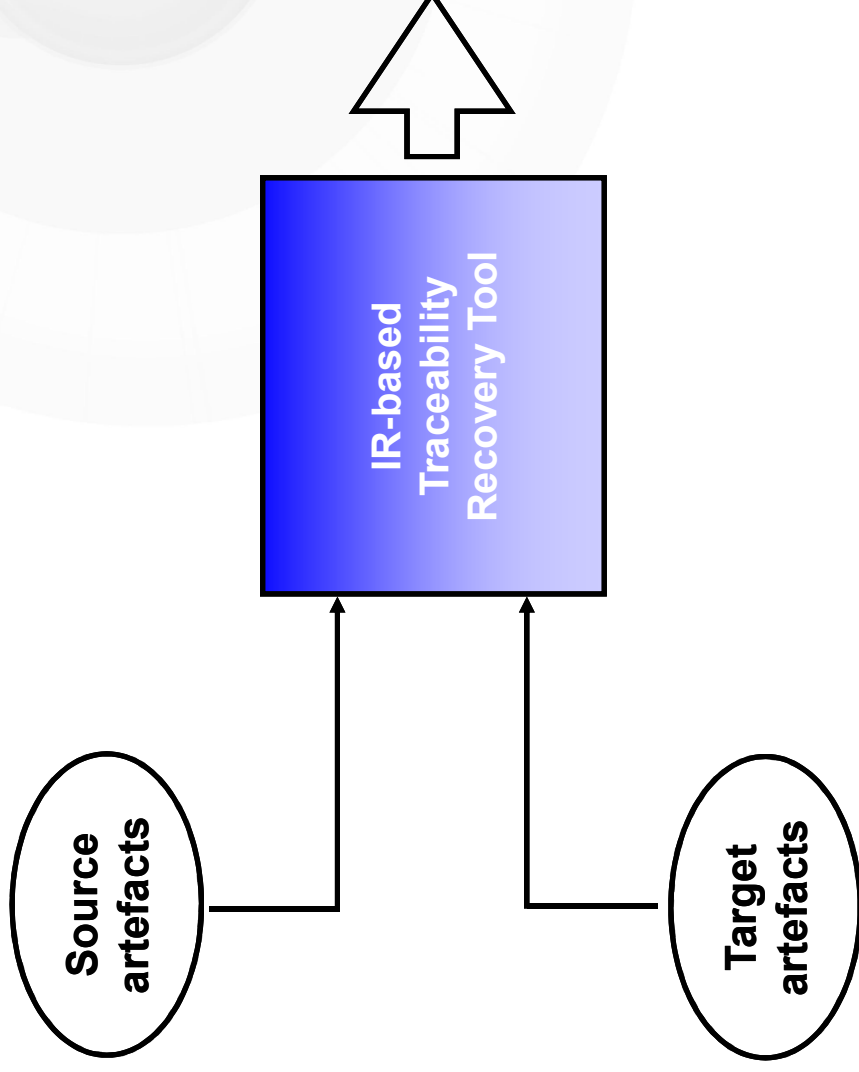
Enhancing IBM Requisite Pro with IR-based Traceability Recovery Features

by Andrea De Lucia, Raffaele Landi, Rocco Oliveto, Genny Tortora

Context

- Traceability...
 - the ability to describe and follow the artefact life-cycle
 - Example: a use case is implemented by one or more classes that are tested by a set of test cases
- Maintaining traceability links during software evolution
 - Tedious and error prone task
 - Often this information becomes out of date or it is completely absent
- Need for automatic support
 - Several methods have been proposed
 - Promising results achieved by IR-based tool

IR-based traceability recovery



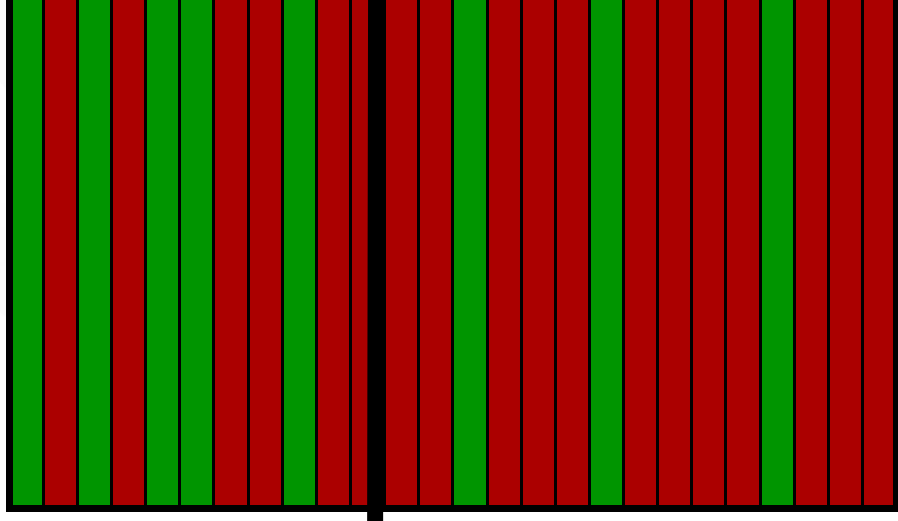
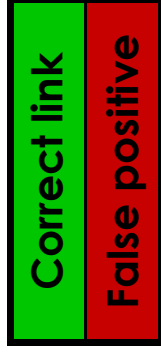
**Ranked list
Candidate links**

Source_1	Target_2	95.4%
Source_3	Target_4	92.1%
Source_1	Target_1	85.6%
Source_2	Target_2	83.2%
Source_3	Target_3	81.2%
Source_1	Target_3	79.0%
Source_3	Target_2	77.5%
Source_2	Target_4	64.3%
Source_2	Target_3	53.2%
Source_3	Target_1	43.9%
Source_2	Target_1	38.7%
Source_1	Target_4	23.6%

The Precision/Recall problem

- Two metrics to measure the performances of IR-based tools
 - $\text{Recall} = (|\text{correct} \cap \text{retrieved}|) / |\text{correct}|$
 - $\text{Precision} = (|\text{correct} \cap \text{retrieved}|) / |\text{retrieved}|$
 - Low precision \Rightarrow high number of false positives to discard
- Recovering all correct links is in general impractical!
 - Necessary to use a very low threshold
 - Low threshold \Rightarrow High number of links retrieved \Rightarrow Low precision
 - High effort to discard too many false positives
- Results from a case study show that...
 - about 50,000 false positives have to be discarded by the software engineer in order to trace 361 links among about 200 artefacts

Density of correct links and false positives



CUT

POINT

IDEAL

THRESHOLD

- Automatic tracing should be combined with manual tracing!

Incremental traceability recovery

Source_1	Target_2	95.4%
Source_3	Target_4	92.1%
Source_1	Target_1	85.6%
Source_2	Target_2	83.2%
Source_3	Target_3	81.2%
Source_1	Target_3	79.0%
Source_3	Target_2	77.5%
Source_2	Target_4	72.3%
Source_2	Target_3	53.2%
Source_3	Target_1	43.9%
Source_2	Target_1	38.7%
Source_1	Target_4	23.6%

90%

80%

70%



Software Engineering Lab
UNIVERSITÀ DEL SALENTO

Correct link
False positive

Link classification

Source_1	Target_3	79.0%
Source_3	Target_2	77.5%
Source_2	Target_4	72.3%

The software engineer decides to stop the process as the effort to discard false positives is becoming too high. Probably he does not retrieve all correct links!

Motivations

- Lacks of IR-based traceability recovery tools
 - In general, stand-alone tools
- A traceability recovery tool is needed
 - Case studies and controlled experiments were carried out
 - The tool drastically reduce the time to complete a traceability recovery task and reduce tracing errors
 - Generally subjects traced about 60-70% of the links with the tool support
- Facilitate the technology transfer of traceability recovery tools
 - Its usefulness should be investigated through empirical user studies
 - It should be integrated within a commercial and widely used CASE tool

ReqTracer Pro Traceability recovery in IBM Requisite Pro

Enhancing IBM Requisite Pro with IR-based Traceability Recovery Features

by Andrea De Lucia, Raffaele Landi, Rocco Oliveto, Genny Tortora

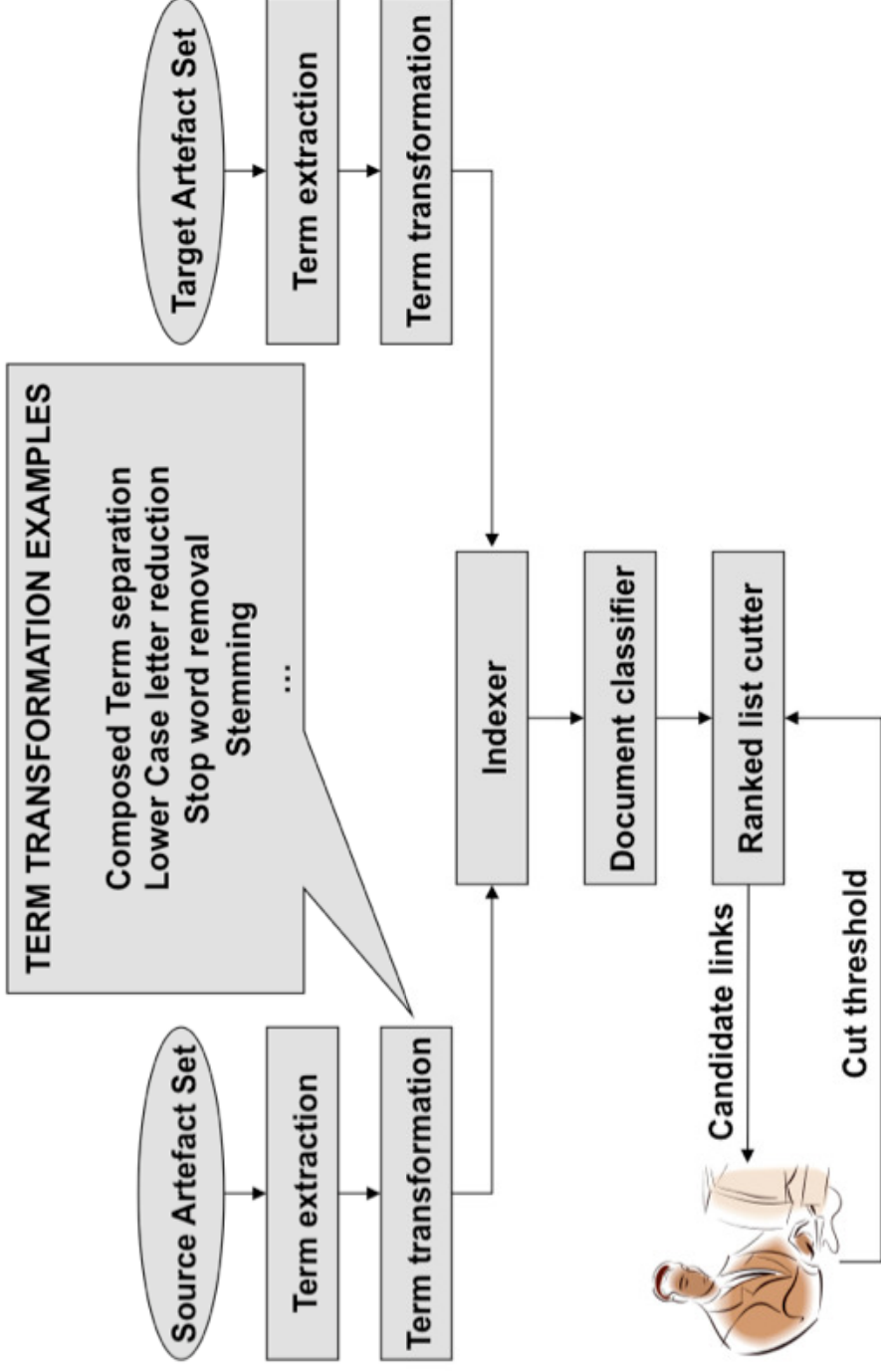
IBM Requisite Pro and Modeler

Software Engineering Lab
UNIVERSITY OF SALERNO

- IBM Requisite Pro
 - Requirements management tool
 - It provides support for traceability, but
 - Traceability information has to be manually managed
- IBM Modeler
 - A collaborative platform for visual modeling and design
 - It integrates several products of IBM Rational Suite, including Requisite Pro
 - It is built on top of the Eclipse platform
- Why such a choice...
 - Traceability links are stored in Requisite Pro
 - IBM Modeler provides a high level of extensibility
 - It also provides API allowing the communication with Requisite Pro
 - A plug-in for Modeler (Eclipse) has the same UI of Requisite Pro

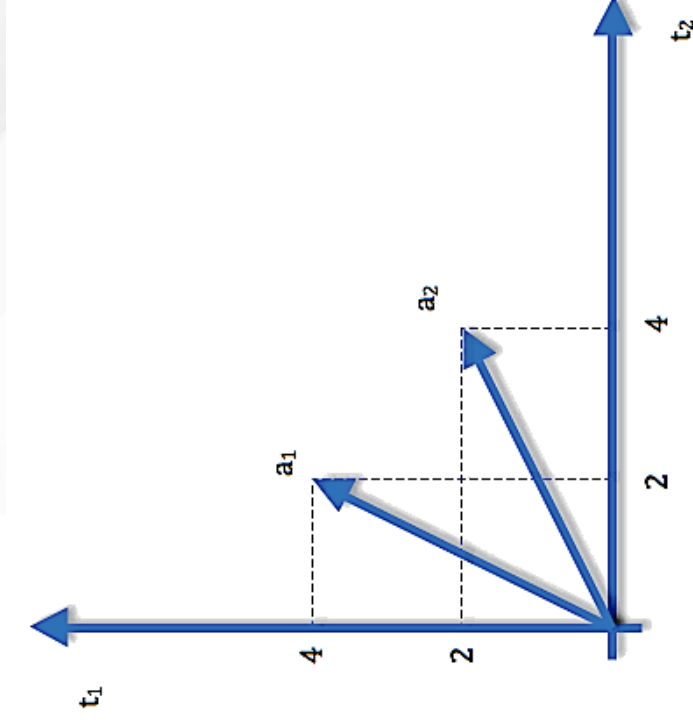
ReqTracer Pro: recovery process

ering Lab
SALERNO



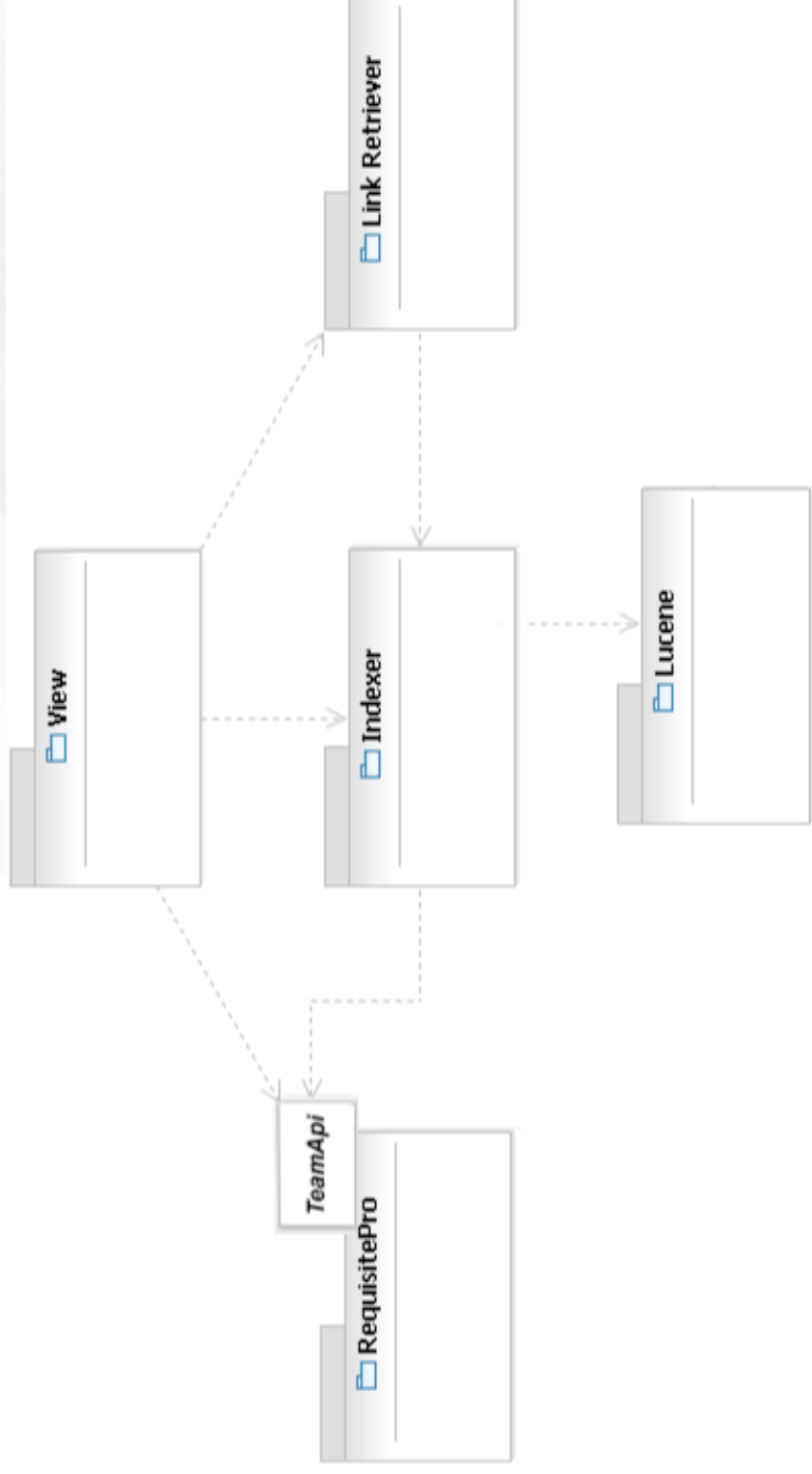
ReqTracer Pro: classifier

- A traceability recovery tool based on Vector Space Model
 - VSM represents artefacts as vector of terms (extracted from the artefacts)
 - The similarity between two artefacts is represented by the cosine of the angle between the two artefact vectors

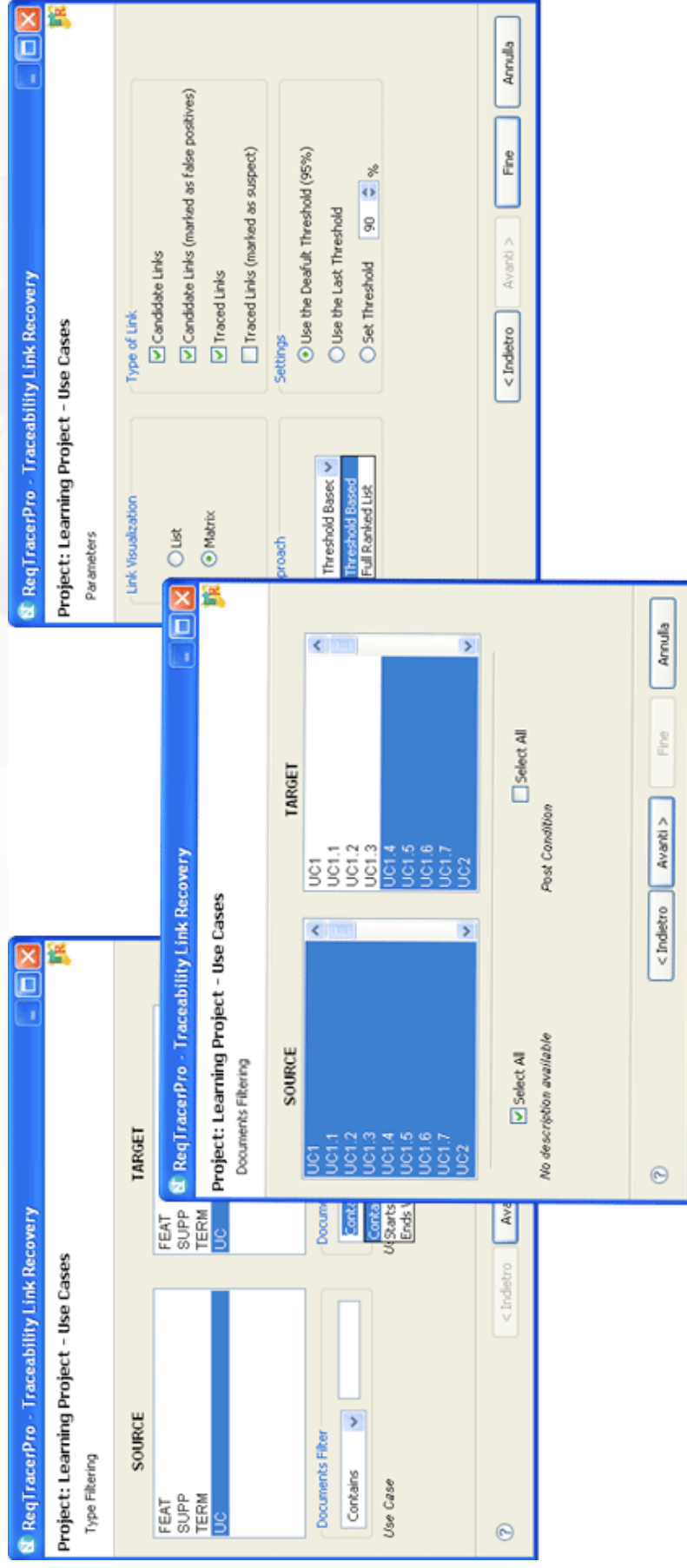


ReqTracer Pro: architecture

Software Engineering Lab
UNIVERSITY OF SALERNO



ReqTracer Pro: traceability recovery



ReqTracer Pro: link analysis

- A list or a matrix of candidate links is provided

Recovery Bar

Similarity Threshold: 35 %

Indexing Statistics: Index data: 2008/9/16 Documents: 68 Time (ms): 312 Reqs type: 4

Traceability Statistics: SubSession N°: 4 Traced Links: 4 False Positives: 2

Source Id	Source Name	Target Id	Target Name	Similarity	Trace Link	False Positive	Direction
317	UC3.8	316	UC3.7	75.0%	<input type="checkbox"/>	<input type="checkbox"/>	from
318	UC3.9	319	UC3.10	44.0%	<input type="checkbox"/>	<input type="checkbox"/>	from
324	UC2.5	327	UC2.4	41.0%	<input type="checkbox"/>	<input type="checkbox"/>	from
310	UC3.1	309	UC3	41.0%	<input type="checkbox"/>	<input type="checkbox"/>	from
326	UC2.7	320	UC2	38.0%	<input type="checkbox"/>	<input type="checkbox"/>	from

Type of Links

Similarity Threshold: 30 %

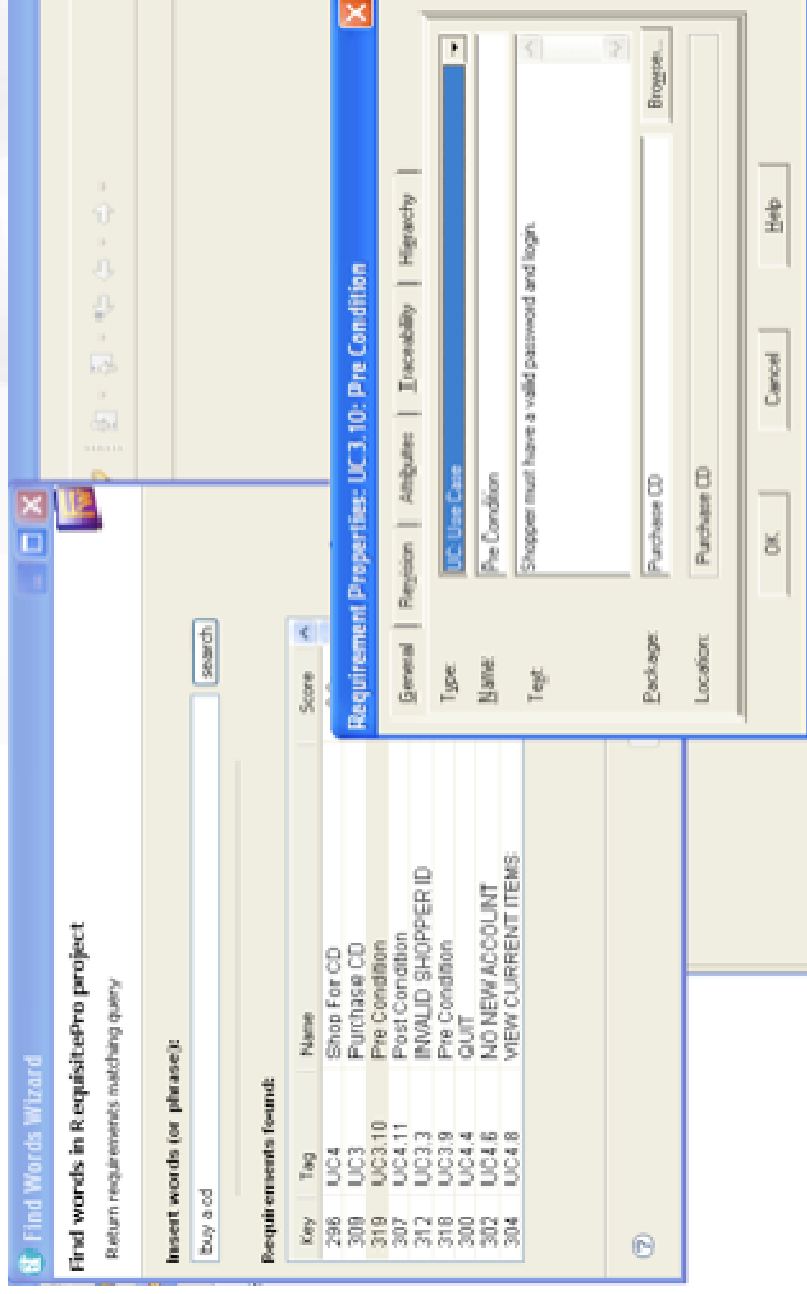
Indexing Statistics: Index data: 2008/9/16 Documents: 68 Time (ms): 312 Reqs type: 4

Traceability Statistics: SubSession N°: 4 Traced Links: 4 False Positives: 2

Type of Links	UC2 Check Or...	UC2.1 BriefD...	UC2.2 Basic FL...	UC2.3 TRACK...	UC2.4 USER A...	UC2.5 INVAL...	UC2.6 OPTION...	UC2.7 Pre Con...	UC3 Purchase...
UC1.6 ORDE...									
UC1.7 SUCC...									
UC2 Check ...									
UC2.1 Brief ...									
UC2.2 Basic ...									
UC2.3 TRAC...									
UC2.4 USER ...									
UC2.5 INVAL...									
UC2.6 OPTIO...									
UC2.7 Pre C...									
UC3 Purchas...									
UC3.1 Brief ...									
UC3.2 Basic ...									

ReqTracer Pro: search engine

- A search engine is also integrated in Requisite Pro



Conclusion and future work

Enhancing IBM Requisite Pro with IR-based Traceability Recovery Features

by Andrea De Lucia, Raffaele Landi, Rocco Oliveto, Genny Tortora

Conclusion

- ReqTracer Pro
 - An IR-based traceability recovery tool integrated in Requisite Pro
- The design of the tool is based from empirical studies
- Why Requisite Pro?
 - Facilitate the technology transfer
- Future work
 - Further experimentation
 - Improvement of the IR engine

Thank you!

Questions and/or comments

Andrea De Lucia, Raffaele Landi, Rocco Oliveto, Genny Tortora
Department of Mathematics and Informatics, University of Salerno
84084, Fisciano (SA), Italy

adelucia@unisa.it, raffaele.landi81@gmail.com, {roliveto, tortora}@unisa.it