

# Towards Trace-Any: Interactive and Transitive Recovery of Traceability Links

Hiro - Hironori Washizaki

Waseda University / National Institute of Informatics

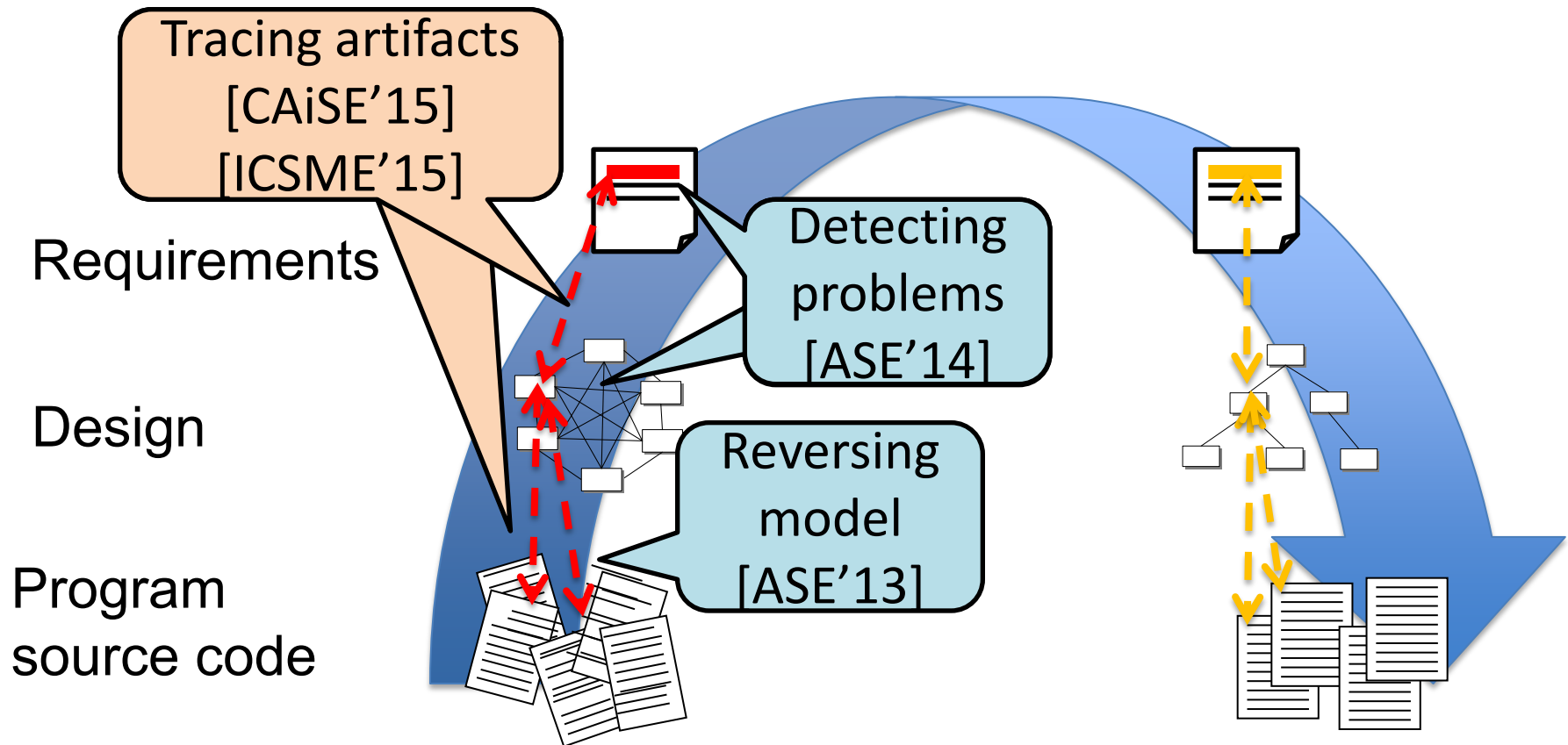
Twitter: @Hiro\_Washi [washizaki@waseda.jp](mailto:washizaki@waseda.jp)

<http://www.washi.cs.waseda.ac.jp/>



# Reverse engineering for maintenance

- Often the only reliable information is embedded in code
- Analysis process to identify elements and create target's representations in another or at a higher level of abstraction



[ASE'13] Automated Verification of Pattern-based Interaction Invariants in Ajax Applications

[ASE'14] Validating Ajax Applications Using a Delay-Based Mutation Technique

[CAiSE'15] Interactive Recovery of Requirements Traceability Links Using User Feedback and Configuration Management Logs

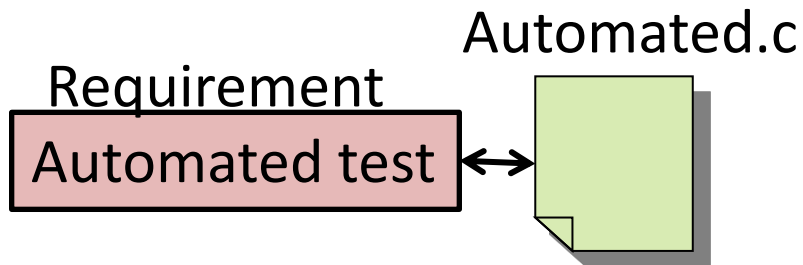
[ICSME'15] Recovering Transitive Traceability Links among Software Artifacts

# Facts of traceability links

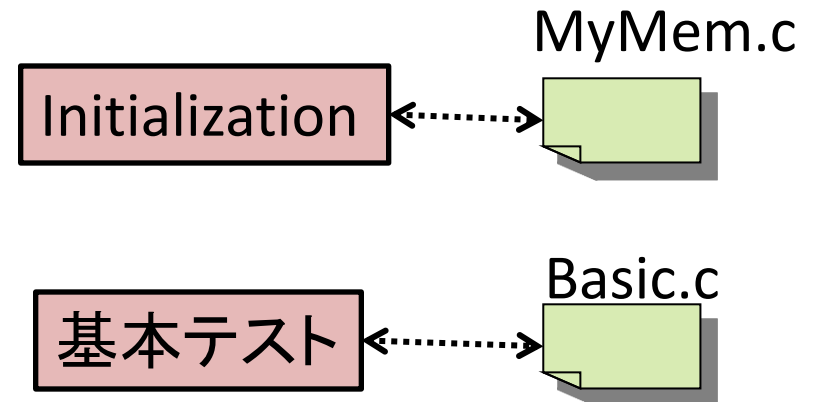
- Relationships among software artifacts
- Key to ensure consistency among artifacts [Antoniol'00]
- Rarely established explicitly [Lucia'12]
- Most of existing recovery techniques employing IR approaches such as TF/IDF and Vector Space Model (VSM)

=> **1. Interactive** and **2. Transitive Recovery**

OK



NG



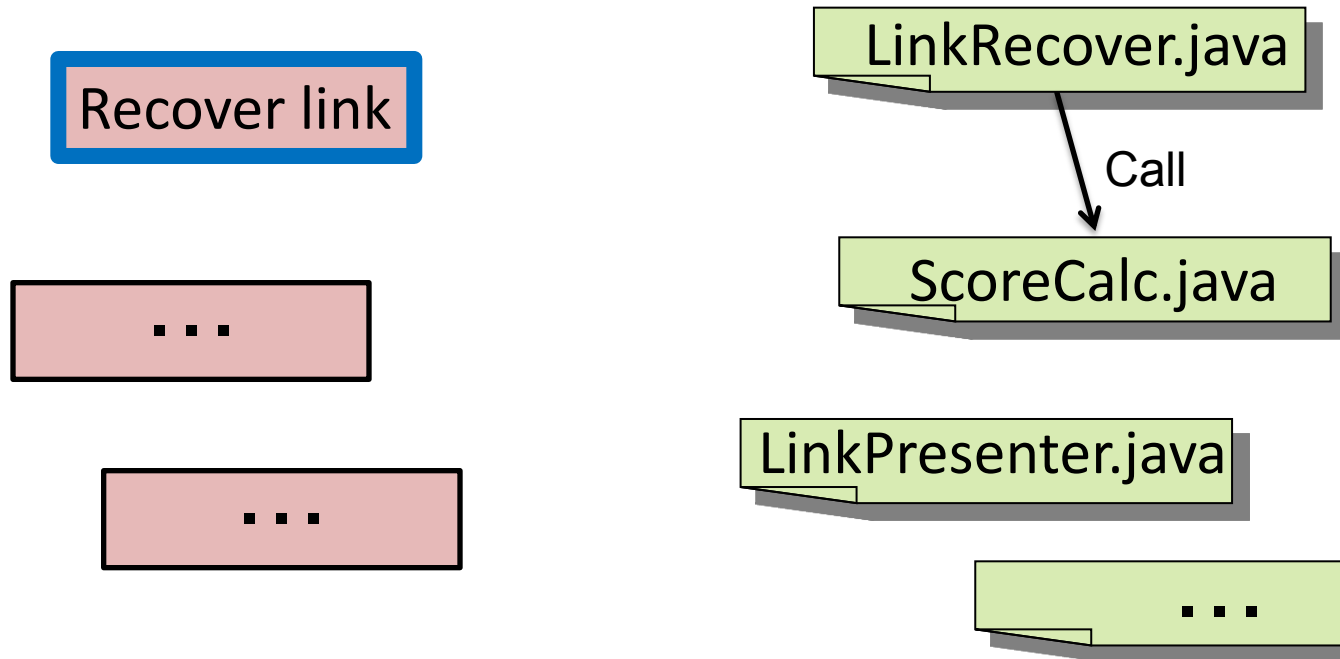
[Antoniol'00] G. Antoniol, B. Caprile, A. Potrich and P. Tonella, "Design-Code Traceability for Object-Oriented Systems," Annals of Software Engineering, vol. 9, no. 1-4, pp. 35-58, 2000

[Lucia'12] A. De Lucia, et al., "Information Retrieval Methods for Automated Traceability Recovery," in Software and Systems Traceability, 1st ed., J. Cleland-Huang, O. Gotel and A. Zisman, Ed. New York: Springer, p. 71-98, 2012.

# 1. Interactive recovery [CAiSE'15]



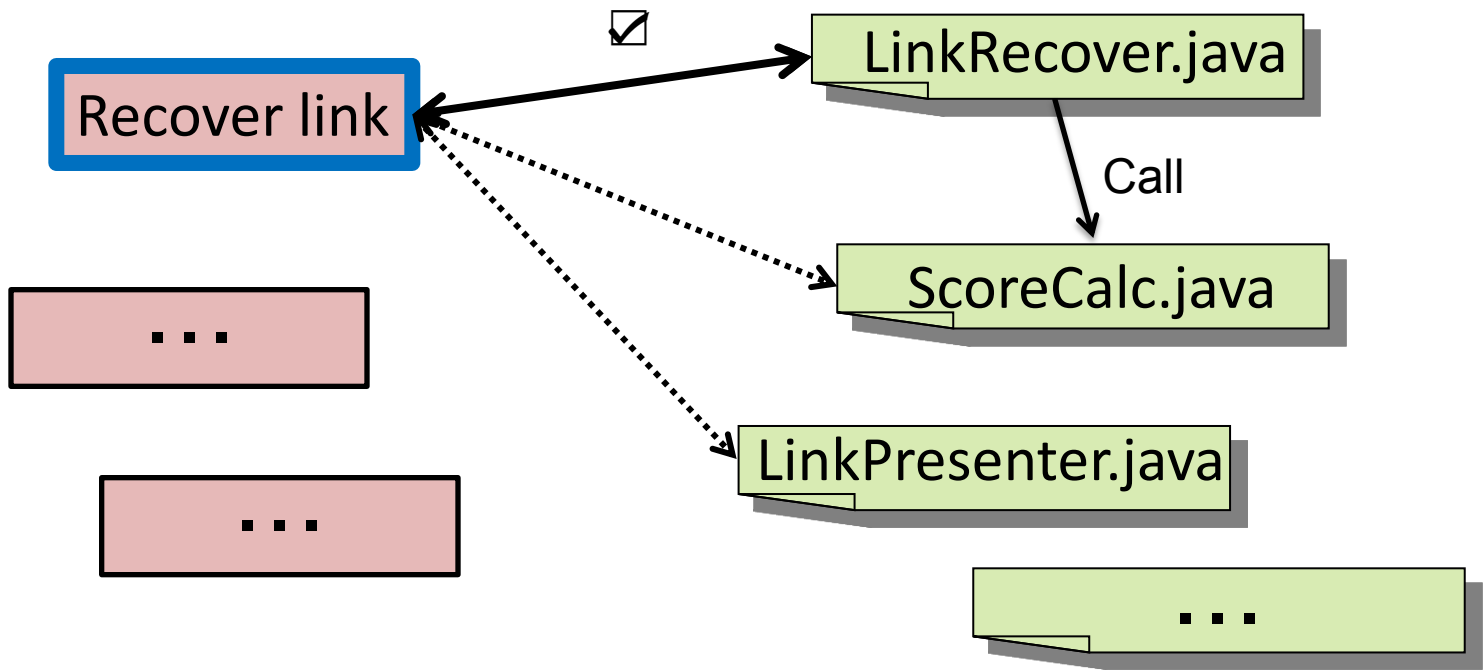
Rank	Requirement	Code file	Score	Recommended	Correct
1	Recover link				
2	Recover link				
3	Recover link				
...	...				



# 1. Interactive recovery [CAiSE'15]



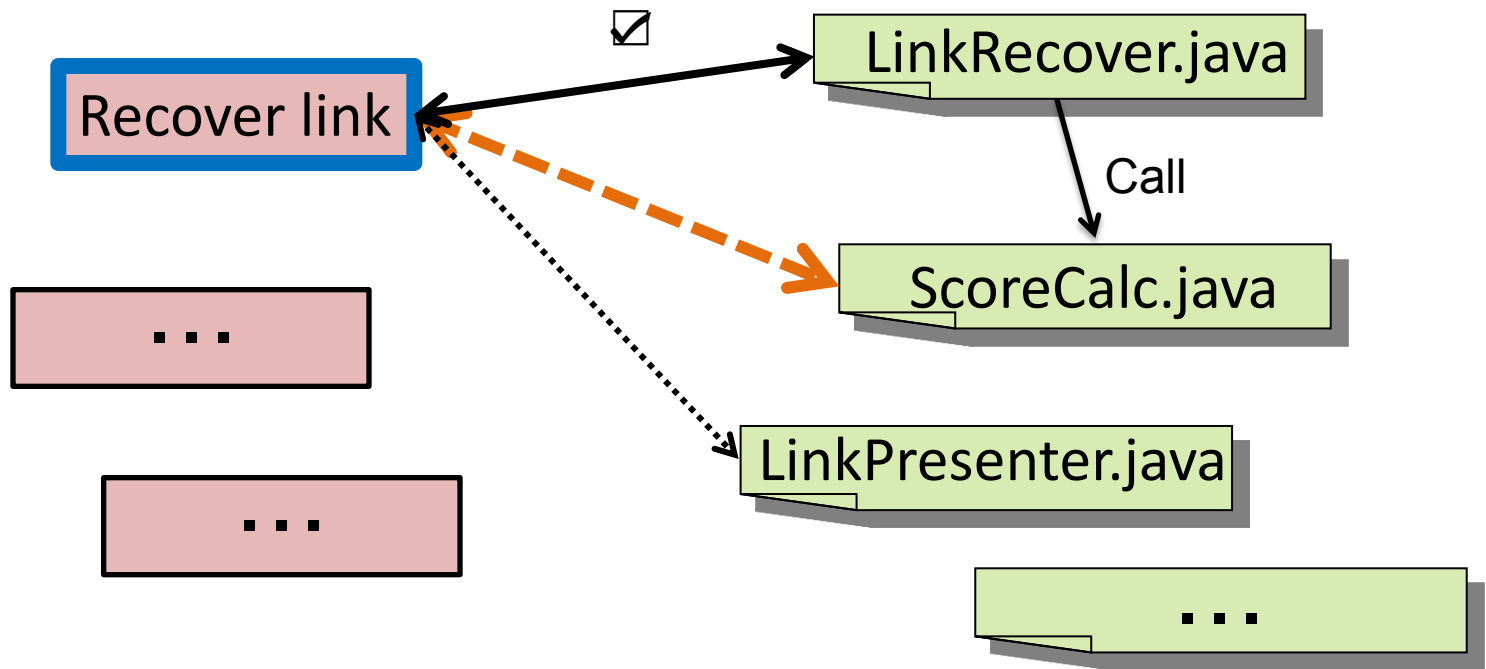
Rank	Requirement	Code file	Score	Recommended	Correct
1	Recover link	LinkRecover.java	0.98	not yet	<input checked="" type="checkbox"/>
2	Recover link	LinkPresenter.java	0.65	not yet	
3	Recover link	ScoreCalc.java	0.30	not yet	
...	...	...	...		



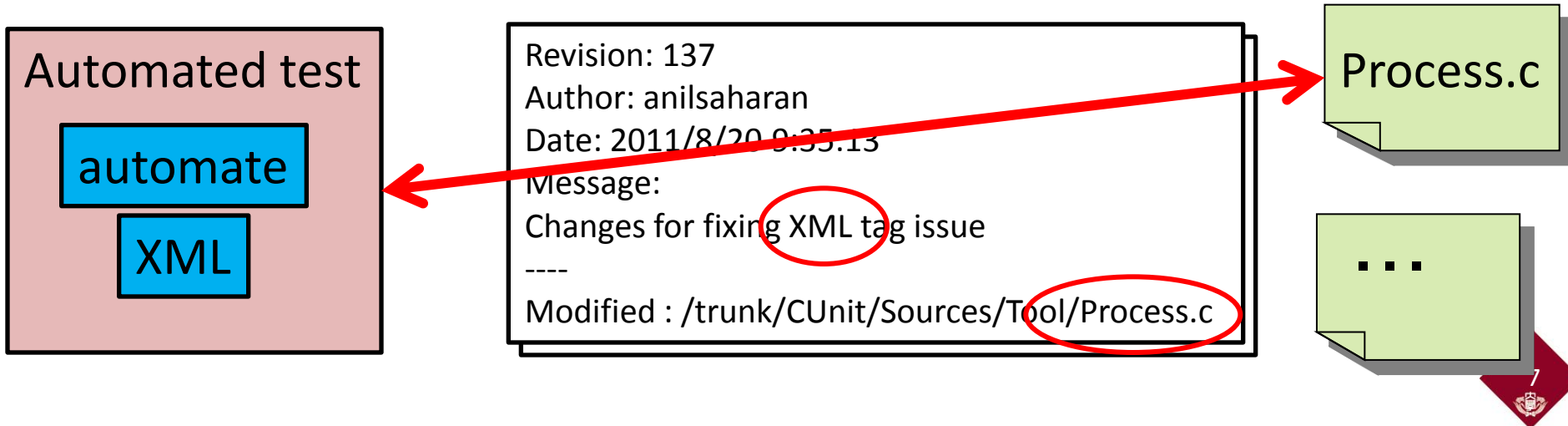
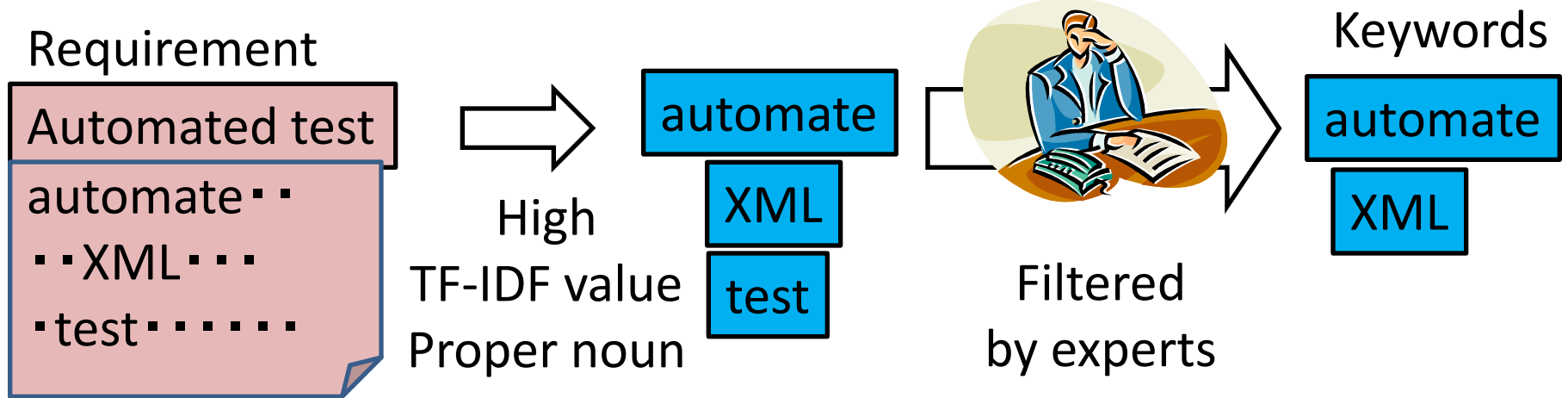
# 1. Interactive recovery [CAiSE'15]



Rank	Requirement	Code file	Score	Recommended	Correct
1	Recover link	LinkRecover.java	0.98	not yet	<input checked="" type="checkbox"/>
2	Recover link	ScoreCalc.java	0.30	Recommended	
3	Recover link	LinkPresenter.java	0.65	not yet	
...	...	...	...		

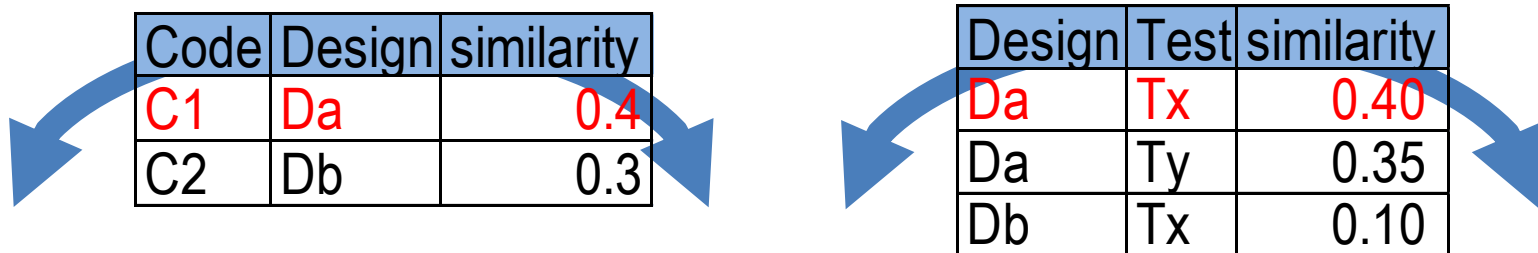


# 2. Configuration management log-based recovery



# 2. Transitive recovery [ICSME'15]

- Vector Space Model (VSM) for basic recovery



```
<content>Class GUIPrenotaVisita Date:16/09/2003
Version: 0 01 000
Description Make the mask on the management
of booking a visit
Attributes
Name Access Description
Private btnAnnulla button that allows you to
return implemented by the mask GUIPaziente
Private btnConfermaTipo button that allows you
to confirm type of access to reservation
Private btnPrenota button that allows
you to confirm Reservations
Methods
Signature notificaSuccesso () Public Access
Description Its task is to notify the success
of
Signature show () Public Access
Description View mask
Signature visualizzaDataVisi Public Access
```

<content>reservation for a visit after initial  
 The application of class GUIPrenotaVisita handler  
 displays the mask on the reservation to visit the  
 Patient must be subject (for instance  
 GUIPrenotaVisita. On this Operator select the  
 form and confirms the type of access and control  
 returns to the instance of which  
 GUIPrenotaVisita handler delegating the task to  
 insert a new one application of PrenotazioniManager  
 This feature is was described by the collaboration  
 diagram of Fig 3 17 Also because? the book a  
 Outpatient service is the heart of it is thought  
 to comment on that functionality with un'activity  
 diagram (Fig 3 18) near the  
 iterations th  
 system itself

<content>Test case reservation  
 a visit Date:  
 C41 control preceded by 20/06/2003  
 another visit check  
 Version: 0 02 000  
 Use Case Meets the request  
 for a reservation  
 UcPreVis service by Outpatient  
 patient  
 Media Priorities  
 Set up The last reservation is  
 06/10/2003 hours 12 00  
 Visit control  
 Description test  
 Input type selected visit: Visit  
 check  
 The Oracle system enables

Code explanation (C1)

Design (Da)

Test case (Tx)

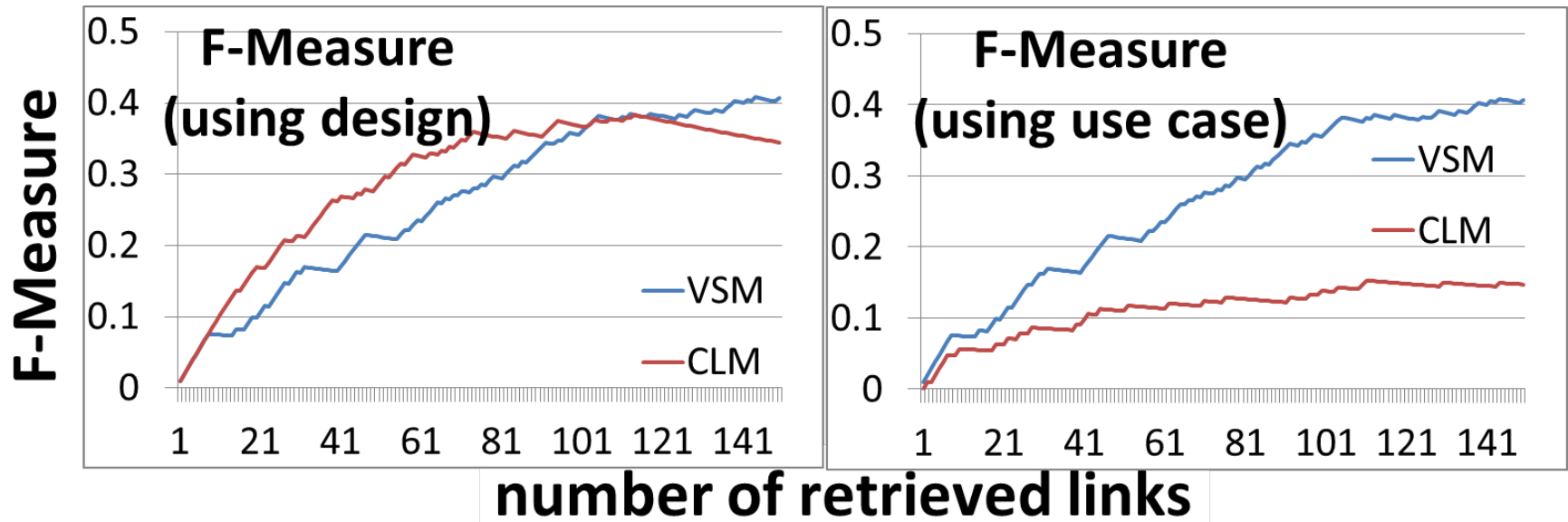
Code	Design	Test	score
C1	Da	Tx	0.16
C1	Da	Ty	0.14
C1	Db	Tx	0.03





## 2. Experiment: EasyClinic

- Target link: 47 Java code  $\Leftrightarrow$  63 test docs
- Supportive artifacts: 20 design docs or 30 use cases
- Effectiveness varies with supportive artifacts



# Open questions

- 0. More sophisticated NLP/IR approaches for basic recovery and unstructured artifacts (such as LSA, LDA, island parsing, ...)?
- 1. Can developers always evaluate correctness of links? What if incorrect?
- 2. What kind of supportive artifacts are useful for recovery/maintenance of other (missing) links?

# Related work and open questions

- Related work and possible extension
  - Relevance feedback [Lucia'06], multi-faceted Interactive exploration [Wanget'13]
  - Recovery based on structural relationships in program [Kagdi'07][CSIE'09][Ghabiand'12]
- Open questions
  - 0. Sophisticated NLP/IR approaches for basic recovery and unstructured artifacts (such as LSA, LDA, island parsing, ...)?
  - 1. Can developers always evaluate correctness of links?
  - 2. What kind of supportive artifacts are useful for recovery/maintenance of other (missing) links?

[Lucia'06] Andrea De Lucia, Rocco Oliveto, and Paola Sgueglia, Incremental Approach and User Feedbacks: a Silver Bullet for Traceability Recovery? , ICSM2006

[Wanget'13] J Wanget al., Improving Feature Location Practice with Multi-faceted Interactive Exploration, ICSE2013

[Kagdi'07] H. Kagdi, J. Maletic, and B. Sharif, "Mining Software Repositories for Traceability Links," Proc. 15th IEEE Int'l Conf. Program Comprehension, pp. 145-154, June, 2007.

[Kassab'09] M. Kassab, O. Ormandjieva, and M. Daneva, "A metamodel for tracing non-functional requirements," WRI World Congress on Computer Science and Information Engineering (CSIE'09), vol.7, pp.687-694, March, 2009.

[Ghabiand'12] A. Ghabiand A. Egyed., Code Patterns for Automatically Validating Requirements-to-Code Traces, ASE2012