

# Scientific Literature (@ TUM)

*Dr. Benedikt Hauptmann*

*Dr. Henning Femmer*

With material from

*Dr. Florian Deußenböck, and*

*Prof. Dr. Stefan Wagner*

## Goals and Content of this talk

1. Methodology: Searching for literature in a scientific field
2. Evaluation: Indicators for quality of scientific papers
3. Technical Aspects (@TUM): Getting a paper for a citation

## Why should we care about literature?

- Part of your job.

### Goals of literature reviews:

- Understand the state of science
- Identify currently open questions
- Show relevance
- Define commonalities and differences with other work (and explain why)
- Place your work in the area of research
- Give evidence for your assumptions
- ...

Why?



## Which formats exist?

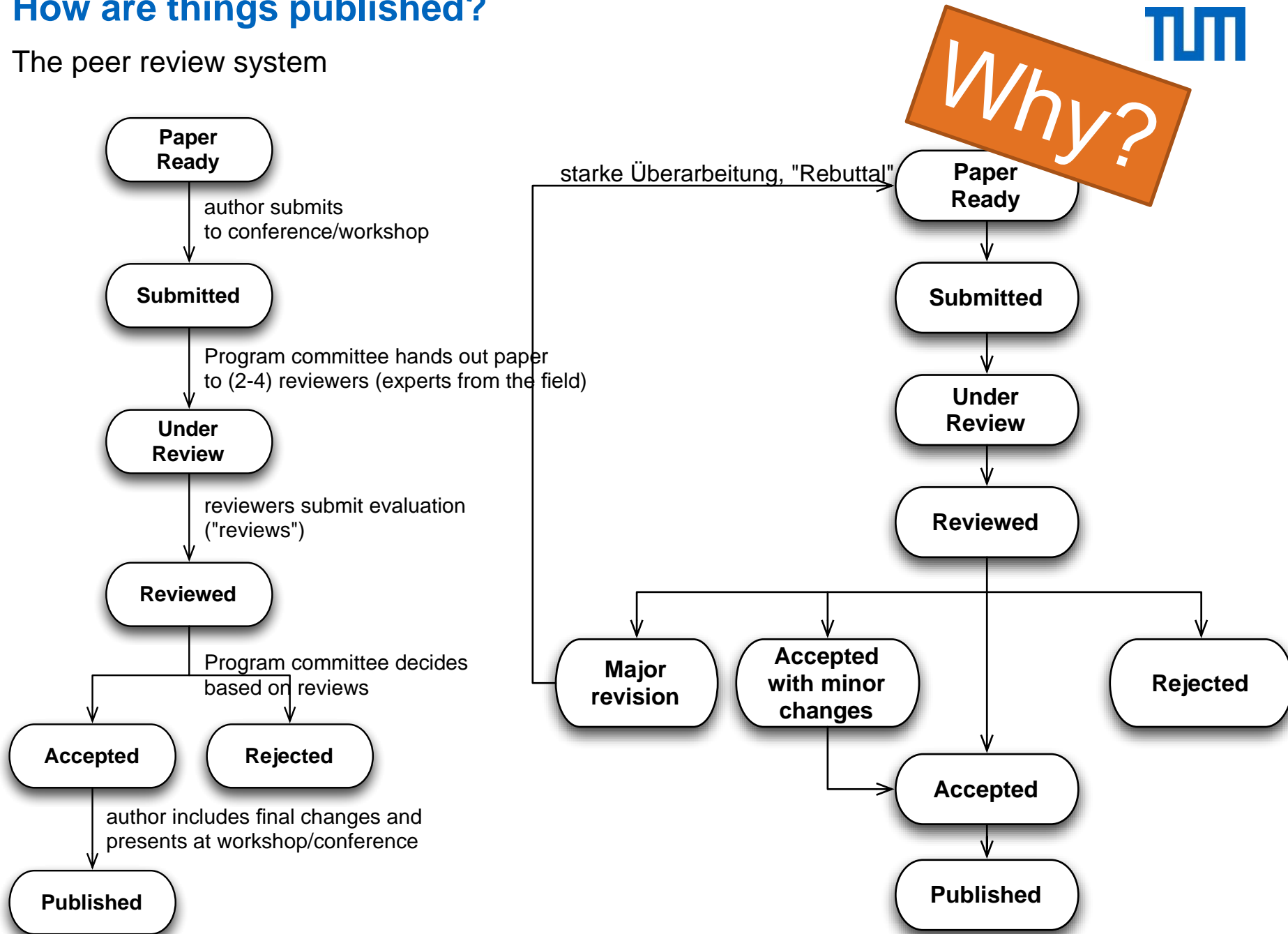
**Different Formats  
-> Different Quality**

- **Book**
  - Usually single-author
  - 100 – 1000 pages
- **Book chapter**
  - 20 – 50 pages
- **Journal article/paper**
  - 10 – 30 pages
- **Proceedings article/paper (conference)**
  - 3 – 15 pages
- **Workshop article/paper**
  - 3 – 15 pages
- **Technical reports**
- **Thesis (Dissertation/Master's Thesis/Bachelor's Thesis)**
- **Blogs**
- ...



# How are things published?

The peer review system



# Publication formats and peer reviewing

- **Book**
  - *Mostly single author, no peer review*
- **Book chapter**
  - *Peer review (several round trips)*
- **Journal article**
  - *Peer review (several round trips)*
- **Proceedings article/paper (conference)**
  - *Peer review*
- **Workshop article/paper**
  - *Peer review*
- **Technical Reports**
  - *No peer review*
- **Thesis (Dissertation/Master's Thesis/Bachelor's Thesis)**
  - *?*
- **Web pages / blogs**
  - *No peer review*
- ...



# Example: Reviews for a workshop



## ----- REVIEW 2 -----

PAPER: 4

TITLE: Who is the Advocate? Stakeholders for Sustainability

AUTHORS: Birgit Penzenstadler, Henning Femmer and Debra Richardson

## ----- REVIEW -----

The paper attempts to devise a systematic process for identifying advocates of sustainability. The process leverages on well-established/classical requirements engineering processes for stakeholders identification.

- + The paper is well-written and presented. The examples are useful.
- + The use of the requirements techniques to identifying sustainability stakeholders looks to be plausible.
- Combining/contrasting the outcomes the four techniques look to be an expensive exercise. Furthermore, sustainability analysis may also require the identification of an equally important slice of stakeholders, who are the devils' advocates for sustainability. This could be important for promoting behavioral change, analyzing obstacles for the sustainability agenda, understanding their incentives, quantifying the risks to adoption and working on solutions for promoting acceptance.
- I would also expect more justification of the exercise in terms of long term benefits and influences on various software artifacts. The motivation is bit weak to justify the effort.



## ----- REVIEW 1 -----

PAPER: 8

TITLE: Detecting Inconsistencies in Wrappers - A Case Study

AUTHORS: Henning Femmer, Dharmalingam Ganesan, Mika

= Overall opinion =

It's a pleasure to read such a self-contained paper that answers pretty much every question the reader raises while reading. It's quick and provides both detailed background theory and insightful

## ----- REVIEW 2 -----

= Paper overview =

The paper addresses the problem of detecting inconsistencies in software abstractions underlying implementations.

The authors propose a solution for detecting the equivalence of several programs and the modification of parameters. The solution is neither complete nor

The solution proposed is based on static analysis and learning. The static analysis is based on a given implementation of the wrappers, which are "extractors" driving the differences in return values. The solution also integrates a phase. Finally, the methodology is to keep the "best" data

The methodology is to keep the "best" data code composed of 10 OS revealed 84 issues, or failures. Each type of the industrial team is given.

= Detailed feedback =

- abstract - typo: "instead of to the" -> "instead of the"

- p2 - typo: "an SAL" -> "a SAL"

- p3 - typo: "an analysts" -> "an analyst"

- p3: "The tool detects differences in the function pair and highlights the important ones to the user." -> at this stage we don't understand how the "important ones" are defined. It is also unclear at this stage that the training set starts empty (all functions unclassified) and is incrementally defined/increased. I would suggest to be more explicit when describing step (1) and (3).

- p3 - fig 2: the "very light-red background" is very hard to see; I would strongly recommend to find a better way to highlight.

- p3 - fig 2: why are "uint32" and "free" highlighted although they are identical in both implementations?

- p5: "304 pairwise comparisons": can't thing be optimized here? Eg: if f1 and f2 are equivalent and f2 and f3 too, we don't really need to check for f1 and f3, do we?

- p7- fig 6: I would suggest to repeat (at least in the label) which OS which side corresponds to (as in "VxWorks on the left and RTEMS on the right")

to another language?

pleasant to read, with clearly stated assumptions and commented, tail, with very few errors used, and

of 2 technologies; it is integrated, and it is free of false positives

industrial team, from using the tools methodology is applied, with promising

the target language is of the methodology which could it be applied



## Example

p1 l58

4 dimensions vs. views

Btw., why is the technical dimension important? For me it is simply a subset of the economic 'dimension'. Keeping a system sustainable from a technical perspective makes it more profitable from the economic point of view. If it is not 'technically sustainable', it is also not in the economically way.

Reviewer: 1

p1 l 44

example of green software code are mobile apps

Recommendation

Why are they green?

Comments:

#It is esse

+ 4 more pages  
+ “please see the  
attached file for more  
detailed notes”

First some genera

Sorry to be hard c

not want to demot

To me, it looks to

categories to clas

I wound it confusi

I would evaluate it

might be my fault

I. Your aim is to re

Mine, too, I consid

Therefore, the cor

When I think back

paper. Then I rem

the contribution of

p2 l4

conflicts be

?- Are reb

?- Arent th

How do th

Software?

p2 l10

What is th

GSW?

I think the sentence should be reformulated to avoid misunderstandings.

----Sidebar section

Sec 2 Related Work and its Quality

p2 l22

Attention in SW Engineering only recently.

- depends on whether SW E is something different than the production of software

SW that is now classifiyd as greening (Green by) has been produced since many years in environmental sciences, and there has been environmental informatics for a while. There is the conference series of EnviroInfo (Environmental Informatics) (27th this year). I had a lectures how to design environmental software in 1994. Early (german) names are e.g. R Grützner, B Page, V. Wohlgemuth. At least since 2003 there is a Journal on Environmental Informatics (methodologies, applications, and policy considerations, the needs for environmental systems analysis, the challenges of environmental systems modeling, and the impacts of environmental informatics are discussed etc., btw., IF 3.619, not that I like Impact Factors).

One more genera

Is green software

I have the feeling

enough knowledge

Use with care...

## How good is a scientific publication?

Easier to judge:

1. Where was it published (venue)?
  - Peer-reviewed venue
  - Impact factor of venue
  - Acceptance rate of venue
2. How was it received in literature?
  - Number of citations

The screenshot shows the Google Scholar interface. At the top, the Google Scholar logo is on the left, and a search bar contains the text 'do code clones matter' with a magnifying glass icon on the right. Below the search bar, the word 'Articles' is displayed with a blue diamond icon, followed by the text 'About 65,700 results (0.09 sec)'. On the left side, there is a filter menu with the following options: 'Any time' (selected), 'Since 2017', 'Since 2016', 'Since 2013', and 'Custom range...'. The main search result is titled 'Do code clones matter?' in blue. Below the title, the authors 'E Juergens, F Deissenboeck...' are listed, followed by the year '2009' and the source 'ICSE 2009 ... , 2009 - ieeexplore.ieee.org'. The abstract text reads: 'Abstract: Code cloning is not only assumed to inflate maintenance costs but also considered defect-prone as inconsistent changes to code duplicates can lead to unexpected behavior. Consequently, the identification of duplicated code, clone detection, has been a very active'. At the bottom of the result, there are icons for a star, a bookmark, and a citation count, followed by the text 'Cited by 347', 'Related articles', and 'All 33 versions'.

Google Scholar

do code clones matter

Articles

About 65,700 results (0.09 sec)

Any time

Since 2017

Since 2016

Since 2013

Custom range...

**Do code clones matter?**

E Juergens, F Deissenboeck... - ... , 2009. ICSE 2009 ... , 2009 - ieeexplore.ieee.org

Abstract: Code cloning is not only assumed to inflate maintenance costs but also considered defect-prone as inconsistent changes to code duplicates can lead to unexpected behavior. Consequently, the identification of duplicated code, clone detection, has been a very active

☆ 347 Cited by 347 Related articles All 33 versions

## Engineering venues\*

Guess:  
Acceptance Rate

Top General SE Conferences	<u>ICSE</u>	<u>FSE/ESEC</u>	<u>ASE</u>	<u>SPLASH/OOPSLA</u>	<u>ECOOP</u>	<u>ISSTA</u>	<u>FASE</u>
2013	<u>85/461(18%)</u>	<u>51/251(20%)</u>	<u>?</u>	<u>?</u>	<u>?</u>	<u>32/124(26%)</u>	<u>26/112(23%)</u>
2012	<u>87/408(21%)</u>	<u>34/201(17%)</u>	<u>?</u>	<u>?</u>	<u>?</u>	<u>31/108(29%)</u>	<u>33/134(25%)</u>
2011	<u>62/441(14%)</u>	<u>34/203(17%)</u>	<u>?</u>	<u>?</u>	<u>?</u>	<u>35/121(29%)</u>	<u>29/99(29%)</u>
2010	<u>52/380(14%)</u>	<u>34/169(20%)</u>	<u>37/252(15%)</u>	<u>61/166(37%)</u>	<u>?</u>	<u>24/105(23%)</u>	<u>24/96(25%)</u>
2009	<u>50/405(12%)</u>	<u>32+7/217(15%)</u>	<u>34+31/191(18%)</u>	<u>?(28%)</u>	<u>?(23%)</u>	<u>25/93(27%)</u>	<u>30/124(24%)</u>
			<u>38+33/222(17%)</u>	<u>25/144(17%)</u>	<u>25/117(21%)</u>		

- Journal of Software Engineering
- Wiley Journal of Software Engineering
- Springer Empirical Software Engineering

### Conferences (Proceedings)

- International Conference on Software Engineering (ICSE)
- Foundations of Software Engineering (FSE)
- International Conference on Automated Software Engineering (ASE)
- OO Programming, Systems, Languages and Applications (OOPSLA)
- International Symposium on Software Testing and Analysis (ISSTA)
- International Conference on Software Maintenance (ICSM)

- <http://web.engr.illinois.edu/~taoxie/seconferences.htm>

\*in no particular order

# Hands-on

# Where can we find papers?

## Publishers Websites:

- ACM Digital Library
- IEEE Xplore
- Springer Link
- Elsevier
- TUM library

## Direct sources

- Authors homepages

## Meta sources

- scholar.google.com
- Research Gate

## We use mostly Google Scholar!

- Pro
  - All results in one place
  - Direct meta-information (citations)
  - Sometimes direct link to PDF
  - Author graphs
- Con
  - No quality filter
  - Few search filter options

### Henning Femmer

MSc. with honours

E-mail  
femmer@in.tum.de

Address  
Technische Universität München  
- Institut für Informatik  
- Lehrstuhl für Software & Systems Engineering  
Boltzmannstr. 3  
85748 Garching  
Germany

Phone +49 (89) 289-17080

Fax +49 (89) 289-17307

Room [0.11.064](#)



### NEWS

- Are you a master  
Then [this](#) might b
- Consider submit  
[RE Conference!](#)

### Fields of interest

- Requirements E
- Sustainable Soft
- Empirical Softwa
- Agile Software E

### Teaching

- SS 14  
◦ Teaching A

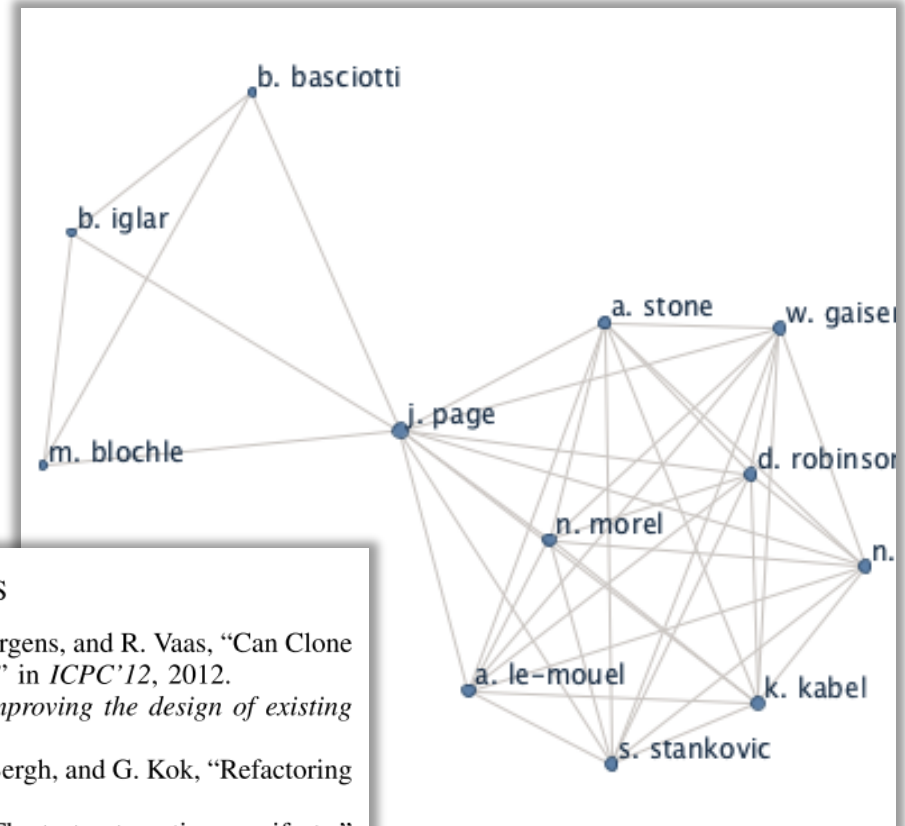
### Publications

- [Rapid Requirements Checks with Requirements Smells: Two Case Studies](#)  
Henning Femmer, Daniel Méndez Fernandez, Elmar Juergens, Michael Klose, Ilona Zimmer,  
2014 1st International Workshop on Rapid and Continuous Software Engineering (RCOSE at ICSE)
- [In Quest for Requirements Engineering Oracles: Dependent Variables and Measurement](#)  
Daniel Méndez Fernández, Jakob Mund, Henning Femmer and Antonio Vetro'  
2014 18th International Conference on Evaluation and Assessment in Software Engineering (EASE)
- [Systematic Mapping Study on Software Engineering for Sustainability \(SE4S\)](#)  
Birgit Penzenstadler, Ankita Raturi, Debra Richardson, Coral Calero, Henning Femmer and X  
2014 18th International Conference on Evaluation and Assessment in Software Engineering (EASE)
- [Reviewing Natural Language Requirements with Requirements Smells – A Research Paper](#)  
Henning Femmer  
2013 11th International Doctoral Symposium on Empirical Software Engineering (IDoESE'13)
- [RE@21: Time to Sustain!](#)  
Birgit Penzenstadler, Henning Femmer  
2013 Second International Workshop on Requirements Engineering for Sustainable Systems (RE4S)
- [Detecting Inconsistencies in Wrappers: A Case Study](#)  
Henning Femmer, Dharmalingam Ganesan, Mikael Lindvall, David McComas  
2013 35th International Conference on Software Engineering (ICSE'13), Track: Software Eng
- [Who Is the Advocate? Stakeholders for Sustainability](#)  
Birgit Penzenstadler, Henning Femmer, Debra Richardson  
2013 2nd International Workshop on Green and Sustainable Software (GREENS at ICSE'13)
- [A Generic Model for Sustainability with Process- and Product-specific Instances](#)  
Birgit Penzenstadler, Henning Femmer  
2013 1st Intl. Workshop on Green In Software Engineering, Green By Software Engineering (G



## Search strategies

1. Manually searching for keywords
2. Searching through author's pages
3. (Literature-) Snowballing
4. Systematic strategies
  - Systematic Mapping Studies
  - Systematic Literature Reviews



### REFERENCES

- [1] B. Hauptmann, M. Junker, S. Eder, E. Juergens, and R. Vaas, "Can Clone Detection Support Test Comprehension?" in *ICPC'12*, 2012.
- [2] M. Fowler and K. Beck, *Refactoring: improving the design of existing code*. Addison-Wesley, 1999.
- [3] A. van Deursen, L. Moonen, A. van den Bergh, and G. Kok, "Refactoring Test Code," in *XP'01*, 2001.
- [4] G. Meszaros, S. Smith, and J. Andrea, "The test automation manifesto," in *XP'03*, 2003.
- [5] G. Meszaros, *xUnit Test Patterns: Refactoring Test Code*. Addison-Wesley, 2007.
- [6] M. Abbes, F. Khomh, Y.-G. Gue andhe andneuc, and G. Antoniol, "An empirical study of the impact of two antipatterns, blob and spaghetti code, on program comprehension," in *CSMR'11*, 2011.
- [7] F. Khomh, M. Di Penta, and Y.-G. Gueheneuc, "An exploratory study of the impact of code smells on software change-proneness," in *WCRE'09*, 2009.
- [8] A. van Deursen and M. Leon, "The video store revisited thoughts on refactoring and testing" in *XP'02*, 2002.

## Goals and Content

1. Methodology: Searching for literature in a scientific field
2. Evaluation: Indicators for quality of scientific papers
3. Technical Aspects (@TUM): Getting a paper for a citation



## More details

- B. Kitchenham and S. Charters, “Guidelines for performing Systematic Literature Reviews in Software Engineering,” 2007.
- S. Keshav, “How to Read a Paper”, 2013  
<http://blizzard.cs.uwaterloo.ca/keshav/home/Papers/data/07/paper-reading.pdf>
- Kent Beck: How to get a paper accepted at OOPSLA:  
<http://plg.uwaterloo.ca/~migod/research/beckOOPSLA.html>
- A. Zeller and T. Zimmermann, “Failure is a Four-Letter Word – A Parody in Empirical Research”  
and the corresponding presentation:  
<https://www.youtube.com/watch?v=NM3CIIbuVoM>
- Zugang zu Wissenschaftlichen Publikationen für Mitarbeiter und Studierende der TUM  
<https://www.lrz.de/services/netzdienste/proxy/zeitschriftenzugang/>
- Paper-Verwaltung:  
<http://www.mendeley.com>  
<http://jabref.sourceforge.net>  
...