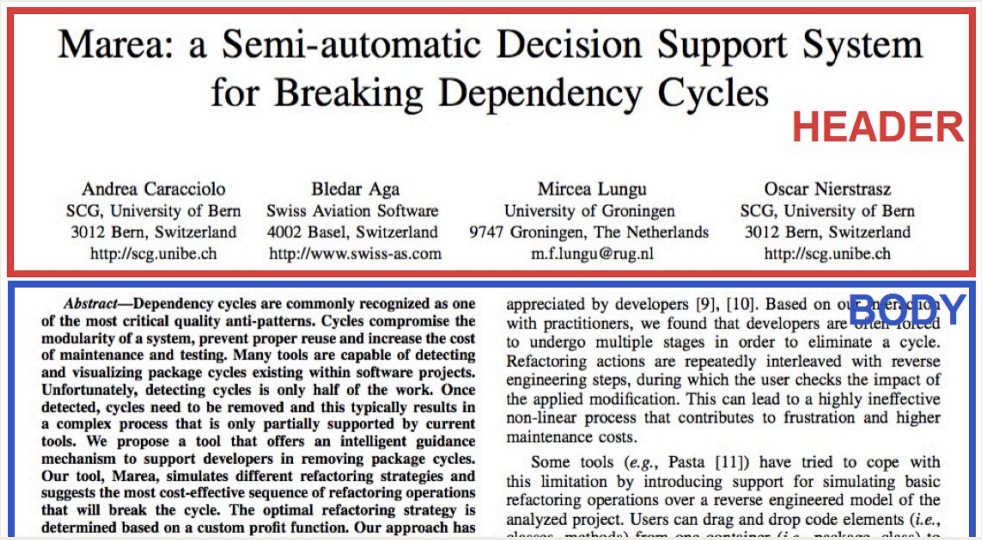
VISUALLY MINING SCIENTIFIC COMMUNITIES

BSc project by Silas Berger, supervised by Leonel Merino
E-Mail: silas.berger@students.unibe.ch

What problem we want to solve

- Valuable information is hidden in amounts of papers too large for humans to read
- Visualizations augments human understanding of data



Marea: a Semi-automatic Decision Support System for Breaking Dependency Cycles

HEADER

Andrea Caracciolo
SCG, University of Bern
3012 Bern, Switzerland
http://scg.unibe.ch

Bledar Aga
Swiss Aviation Software
4002 Basel, Switzerland
http://www.swiss-as.com

Mircea Lungu
University of Groningen
9747 Groningen, The Netherlands
m.f.lungu@rug.nl

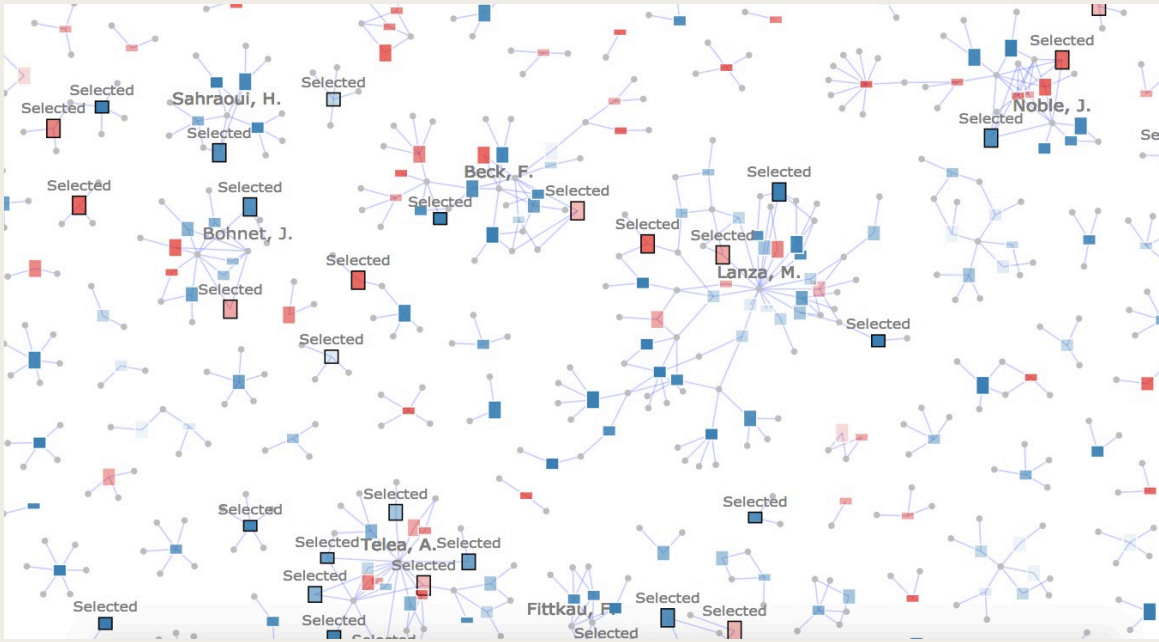
Oscar Nierstrasz
SCG, University of Bern
3012 Bern, Switzerland
http://scg.unibe.ch

BODY

***Abstract*—Dependency cycles are commonly recognized as one of the most critical quality anti-patterns. Cycles compromise the modularity of a system, prevent proper reuse and increase the cost of maintenance and testing. Many tools are capable of detecting and visualizing package cycles existing within software projects. Unfortunately, detecting cycles is only half of the work. Once detected, cycles need to be removed and this typically results in a complex process that is only partially supported by current tools. We propose a tool that offers an intelligent guidance mechanism to support developers in removing package cycles. Our tool, Marea, simulates different refactoring strategies and suggests the most cost-effective sequence of refactoring operations that will break the cycle. The optimal refactoring strategy is determined based on a custom profit function. Our approach has**

appreciated by developers [9], [10]. Based on our interaction with practitioners, we found that developers are often forced to undergo multiple stages in order to eliminate a cycle. Refactoring actions are repeatedly interleaved with reverse engineering steps, during which the user checks the impact of the applied modification. This can lead to a highly ineffective non-linear process that contributes to frustration and higher maintenance costs.

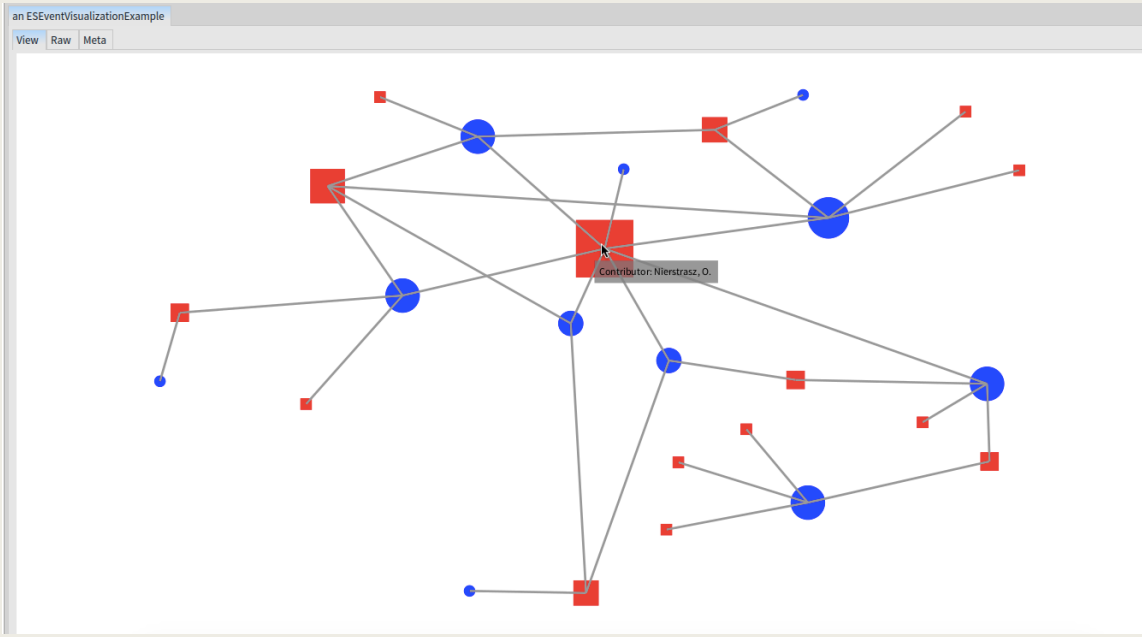
Some tools (*e.g.*, Pasta [11]) have tried to cope with this limitation by introducing support for simulating basic refactoring operations over a reverse engineered model of the analyzed project. Users can drag and drop code elements (*i.e.*,



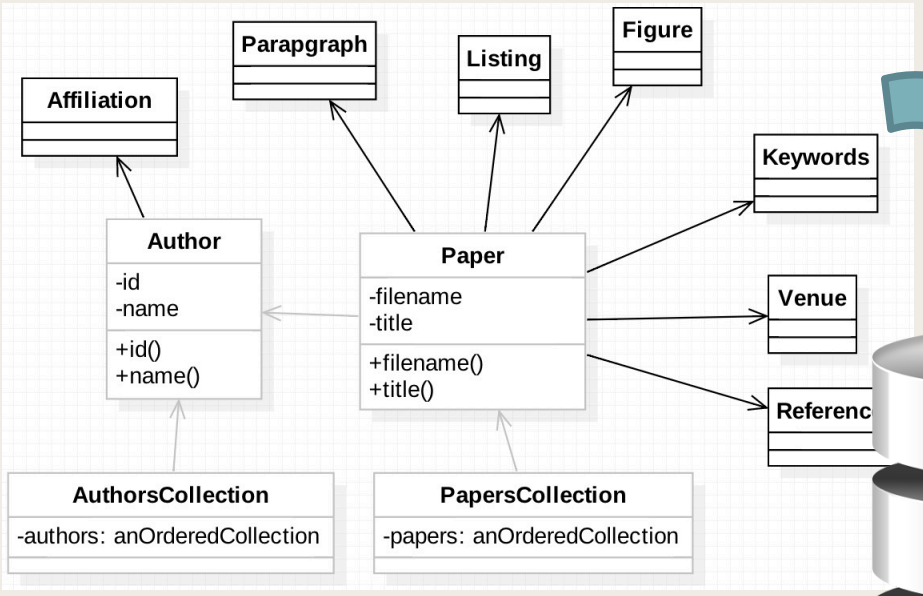
What we are building on (EggShell)

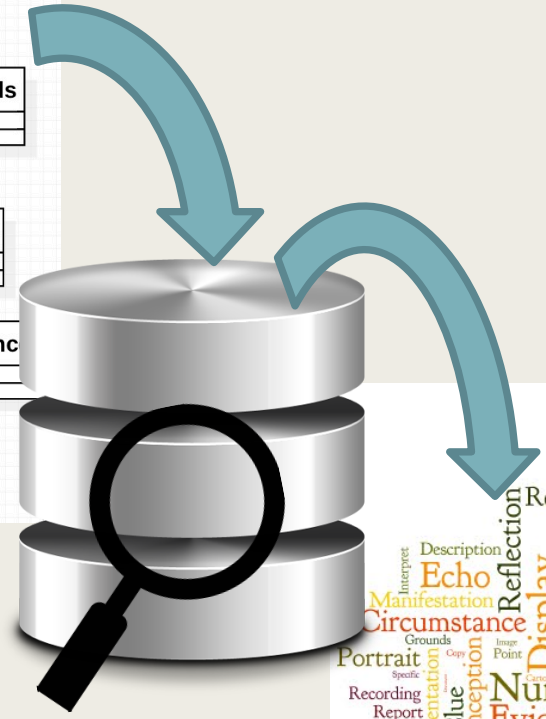
- Can import PDFs as XML extract title and contributors and model retrieved data
- First visualization exists, but focus is on data extraction and (visual) assessment of precision

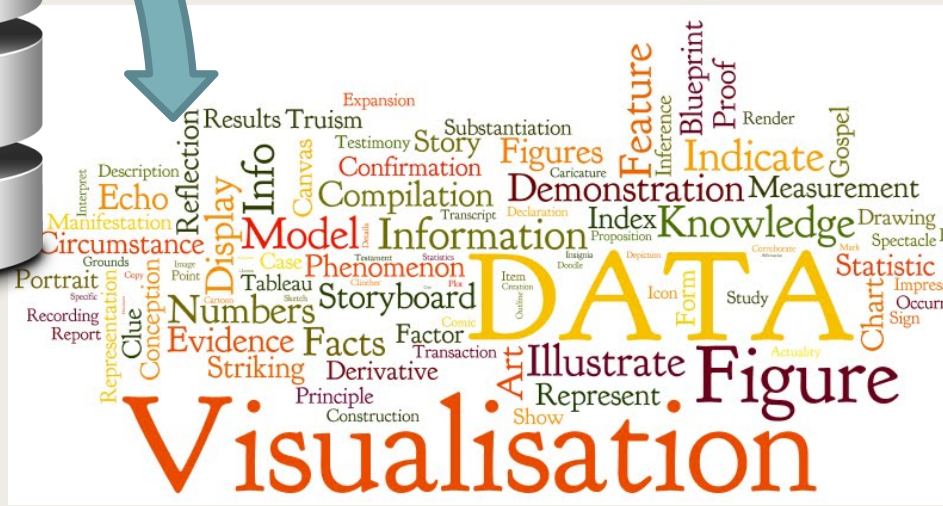
```
model := EggShell modelPapersInFolder: 'examplePI
    using: #ESXMLPipeline.
visualization := ESEventVisualizationExample new
visualization setModel: model.
visualization open.
```



Project outlook



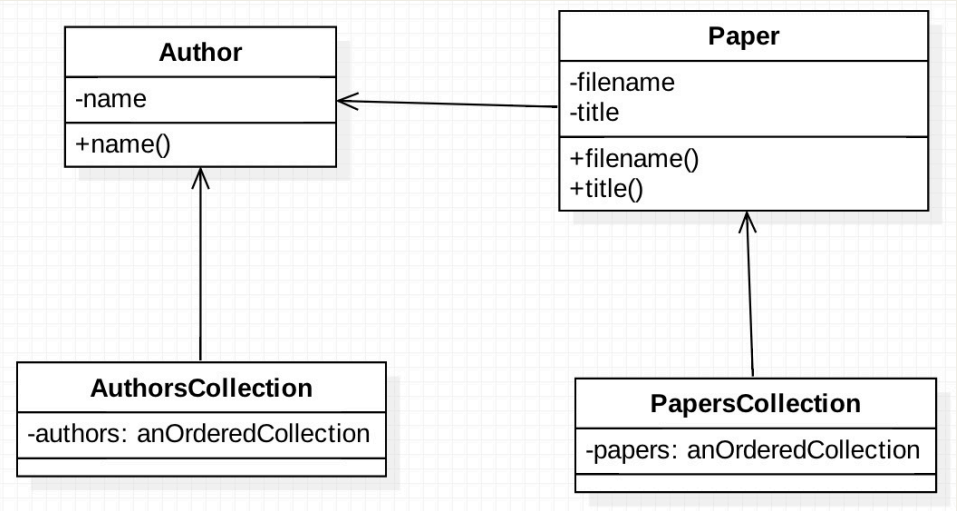




Database query image: graphicsfuel.com/commons.wikimedia.org, word cloud: visualdatablog.blogspot.ch

Current work

- Query language (OCL-like approach)
- Data model expansion
- Work out sample queries to assess quality of data model and query language




```
ExtendedEggShell venue: #VISSOFT lookFor: #Pharo during: (2000 to: 2006).

(ExtendedEggShell papers select: [ :paper | paper authors includes: 'L. Merino']) flatCollect: [ :paper | paper authors ].

(ExtendedEggShell venue: #VISSOFT allAuthors) collect: [ :author | author affiliation ] sortByOccurrences: #DESC.

(ExtendedEggShell papers) lookFor: ('Roassal' 'communities' 'agile') collect: [ :paper | paper title].
```

VISUALLY MINING SCIENTIFIC COMMUNITIES

- Based on EggShell
- Adapting the data model of scientific communities
- Language for querying the model
- Visualization of results that encourage exploration