The Software Engineering Observatory Portal

Mívian Ferreira[2], Bruno Sousa[2], Kecia Ferreira[1], and Mariza Bigonha[2]

[1]CEFET-MG, Dept. of Computing, Minas Gerais (Brazil)

[2]UFMG, Dept. of Computer Science, Minas Gerais (Brazil)

What is Software Engineering?

Discipline of Computer Science

Term was coined by Margareth Hamilton in the 1960’s

Concerns with all aspects of software production, involving specification, development, test, maintenance, and management

52 years of Software Engineering

A massive quantity of techniques, methods, and tools have been proposed

test
bugs
software architecture
mining software repositories
dynamic analysis
software evolution
measure/metric
requirements
collaboration
specifications
software product line
education
change
cloud
security
visualization
concurrency
web service
mobile
android
performance
debugging
design
tool
traceability
modeling
feature
constraints
software process
social networks
open source software
software maintenance
software engineering
refactoring
static analysis
software quality
program comprehension
model driven
machine learning
domain specific
end user software engineering
human factors
source code
agile
runtime
case study
program analysis
automated repair
model checking
behavior analysis
dynamic symbolic execution
parallel programming
object oriented software
formal verification
data mining
eclipse
documentation
reliability
developer
abstraction
search based
human computer interaction
grounded theory
spreadsheet
recommendation
integration
verification and validation
symbolic execution
bad smells
coverage
experimental software engineering
global software development
context aware
deep learning
self adaptation
distributed systems
pull request
flaky tests
annotation
evolution
experience
expertise
sampling
clustering
natural language processing
information retrieval
formal methods
scrum
scalability
data structure
program synthesis
semantic
reuse
api
crash
safety
theory
effort
sat
resources
vulnerability
impact
energy
usability
risk
patterns
sketch
github
types
clones
java
survey
uaf
wsdl
sql
nlp
csrf
mozilla
cpcomo
nyd
ide
app
n gram
feks
css
sos
rest
ajax
git
svn
ufo
xml
uml
tags
task
tactics
slicing
build
cots
successors
alloy
night
beta
peer review
wrapper
abstract syntax
version control
problem identification
graphical user interface
reverse engineer
code analysis
data quality
commit
review
program slicing
neuron coverage
newcomer
database
boa
overwork
linux
framework
usage
cscw
stack overflow
lda
algorithms
software development
software design
decision analysis
code search
exploit
cs1
service oriented architecture
code completion
interface
transparency
concept location
log analysis
sustainability
cryptography
python
systematic literature review
gerrit
contract
management
customer
context aware
evidence style
blogs
big data
use case
winwin
snote
recall
ideas
cost
wiki
africa
libed
xpath
coding
c++11
mylyn
ajdt
php
shoe
roles
ldir
variability
accuracy
ic3
sslvls
simulink
jmc
hpe
sml
aspects
project
tws
patch
context
productivity
unix
environment
dynamic
classification
configuration
topic model
modularity
privacy
code review
crowdsourcing
temporal logic
domain
oracle
intellij
teams
awareness
dynamic
lazy
jml
data structure
scalability
flaky tests
pull request
evolution
human computer interaction
experience
spreadsheet
expertise
sampling
clustering
natural language processing
information retrieval
formal methods
scrum
scalability
data structure
program synthesis
semantic
reuse
api
software evolution
measure/metric
requirements
collaboration
specifications
software product line
education
change
cloud
security
visualization
concurrency
web service
mobile
android
performance
debugging
design
tool
traceability
modeling
feature
constraints
software process
social networks
open source software
software maintenance
software engineering
refactoring
static analysis
software quality
program comprehension
model driven
machine learning
domain specific
end user software engineering
human factors
source code
agile
runtime
case study
program analysis
automated repair
model checking
behavior analysis
dynamic symbolic execution
parallel programming
object oriented software
formal verification
data mining
eclipse
documentation
reliability
developer
abstraction
search based

How to understand Software Engineering Evolution?

Aims to **compile the body of knowledge** of Software Engineering

The portal allows analyzing software engineering evolution and researchers' contributions through **data visualization**

Building the Database

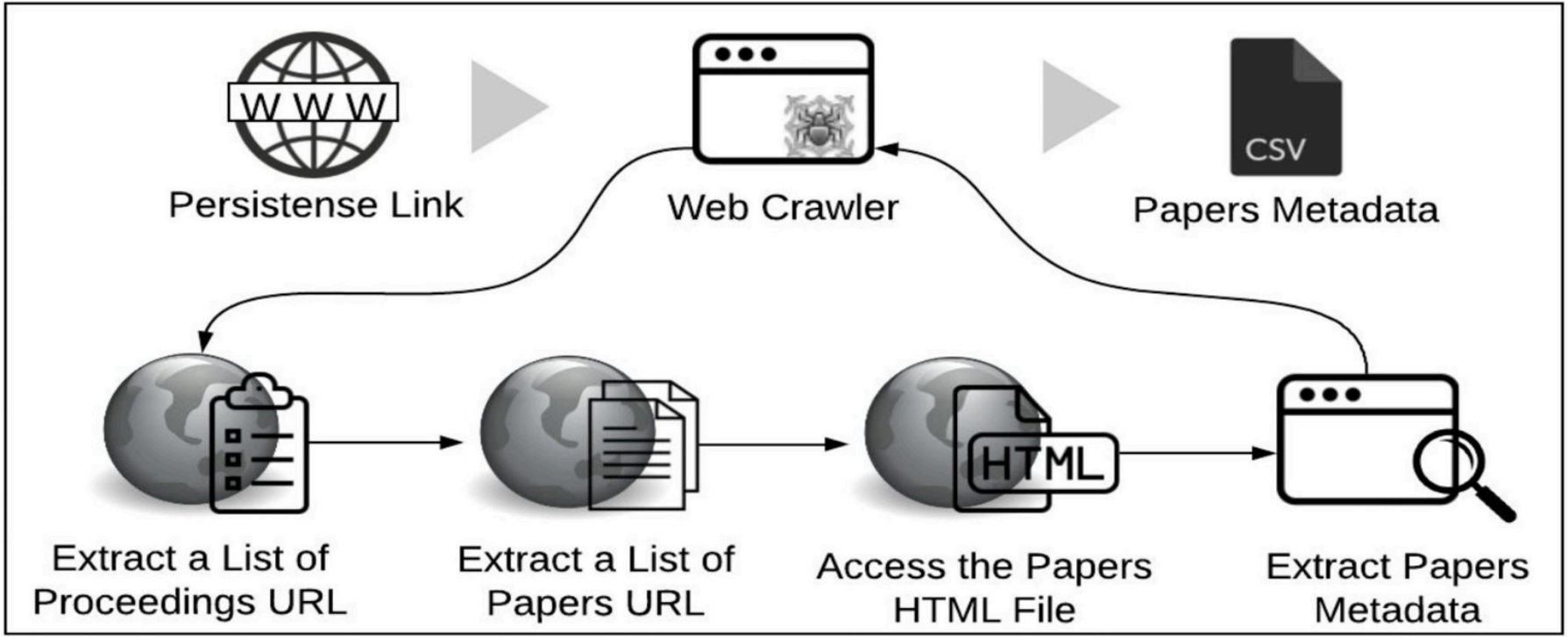
Retrieved metadata of the two main venues of Software Engineering:

ICSE - International Conference on Software Engineering (1988-2018)

TSE - IEEE Transactions on Software Engineering (1975-2018)

Documents available at the IEEE Xplorer Digital Library

Building the Database



The Papers’ Metadata

- Title
- Authors’ names
- Authors’ affiliations
- Year of publication
- Number of citations
- Keywords
- Publication venue (ICSE or TSE)

Clearing the Database

Inclusion Criteria:

- full articles
- short papers
- research in progress
- works available in electronic format

Exclusion Criteria:

- tutorials
- panels
- lectures, and keynote talks
- call for workshops and symposium
- proceedings and round tables
- presentation of sessions and tracks

Manual Preprocessing

Keywords Standardization:

- identify synonyms
- Example: bug, error, failure

Papers Classification:

- procedure or technique, qualitative or descriptive, model, empirical model, analytic model, tool, specific solution, empirical study, and report

Authors' names disambiguations:

- identify authors with different forms of names
- Example: Prem Devanbu, P. Devanbu, P. K. Devanbu

The Software Engineering Observatory Portal

Features

50 Years of Software Engineering

mivanferreira.github.io/docs/TheSoftwareEngineeringObservatoryPortal/word_cloud/ICSE.html

50 Years of Software Engineering

Home Authors Papers Categories Keywords Word Cloud

2010 2016 FILTER

Conclusion

The Software Engineering Observatory Portal allows analyzing the **evolution of Software Engineering** in terms of authorship and subjects investigated in the area

The portal is based on **Data Visualization**

So far, the data used in the portal are from the two main venues of software engineering research: **ICSE and TSE**

Future Work

We are working on the automatization of the data preprocessing to keep the portal up to date

With such automatization, we will Include data from other venues in the portal

The portal may be extended to other disciplines of Computer Science

The same method may be applied in other fields

Questions or Suggestions?

Feel free to contact any of us:

Mívian Ferreira - mivian.ferreira@dcc.ufmg.br

Bruno L. Sousa - bruno.luan.sousa@dcc.ufmg.br

Kecia A. M. Ferreira - kecia@cefetmg.br

Mariza Bigonha - mariza@dcc.ufmg.br