Politechnika Wrocławska

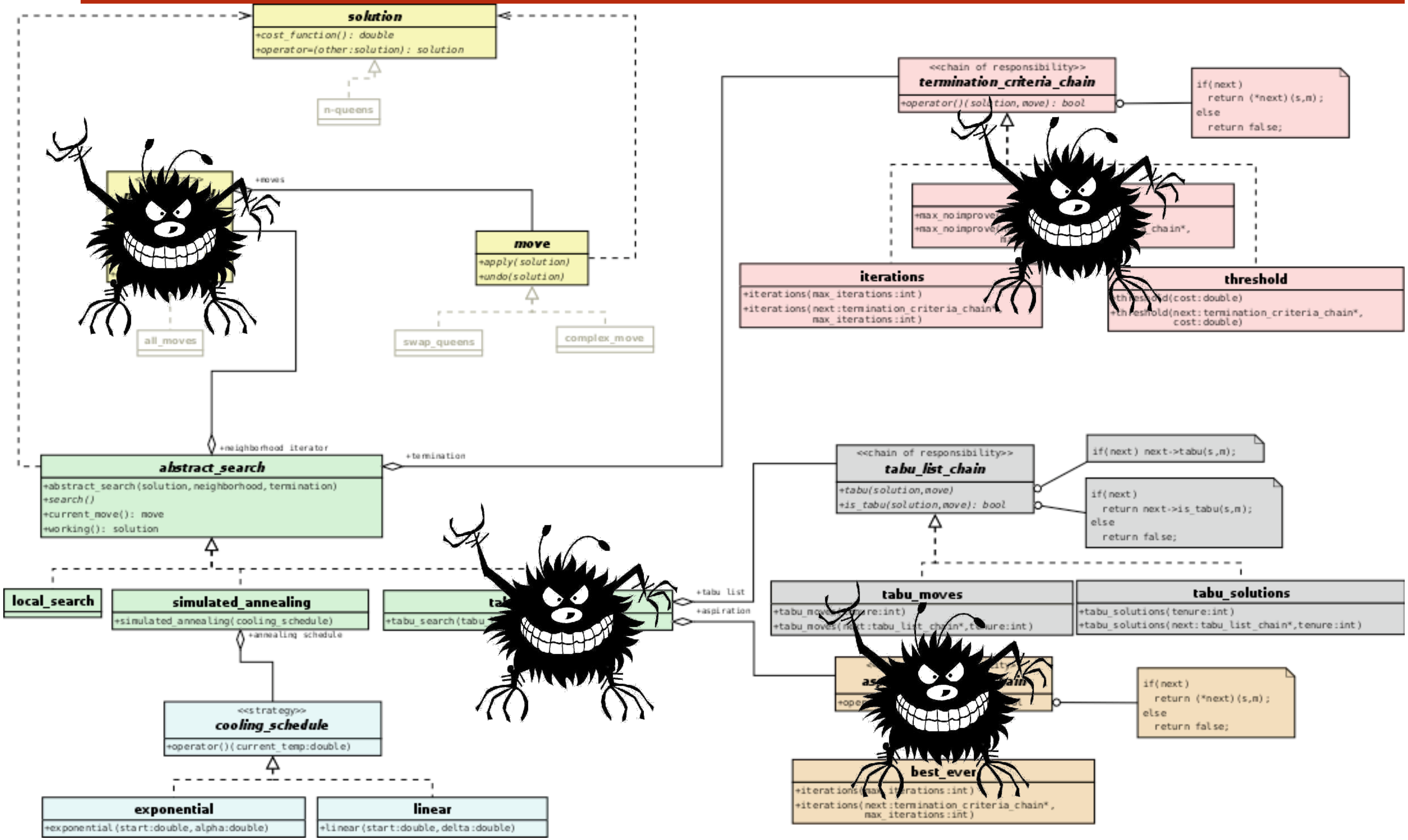
Towards identifying software project clusters with regard to defect prediction

Marian Jureczko, Wrocław University of Technology
Lech Madeyski, Wrocław University of Technology

Agenda

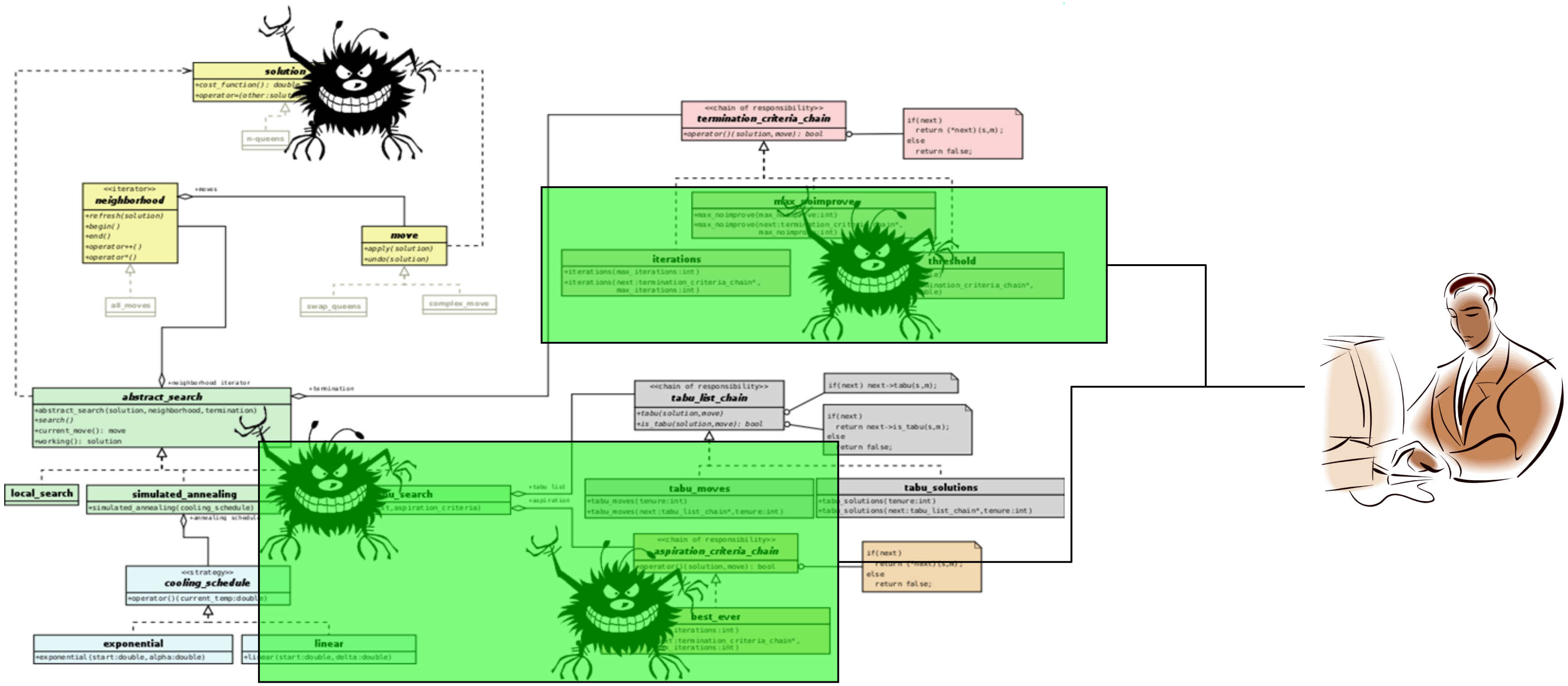
- Introduction
- Data acquisition
- Study design
- Results
- Conclusions

Introduction



Motivation - Why defect prediction?

20% of classes contain 80% of defects



We can use the software metrics to predict error prone classes and therefore prioritize and optimize tests.

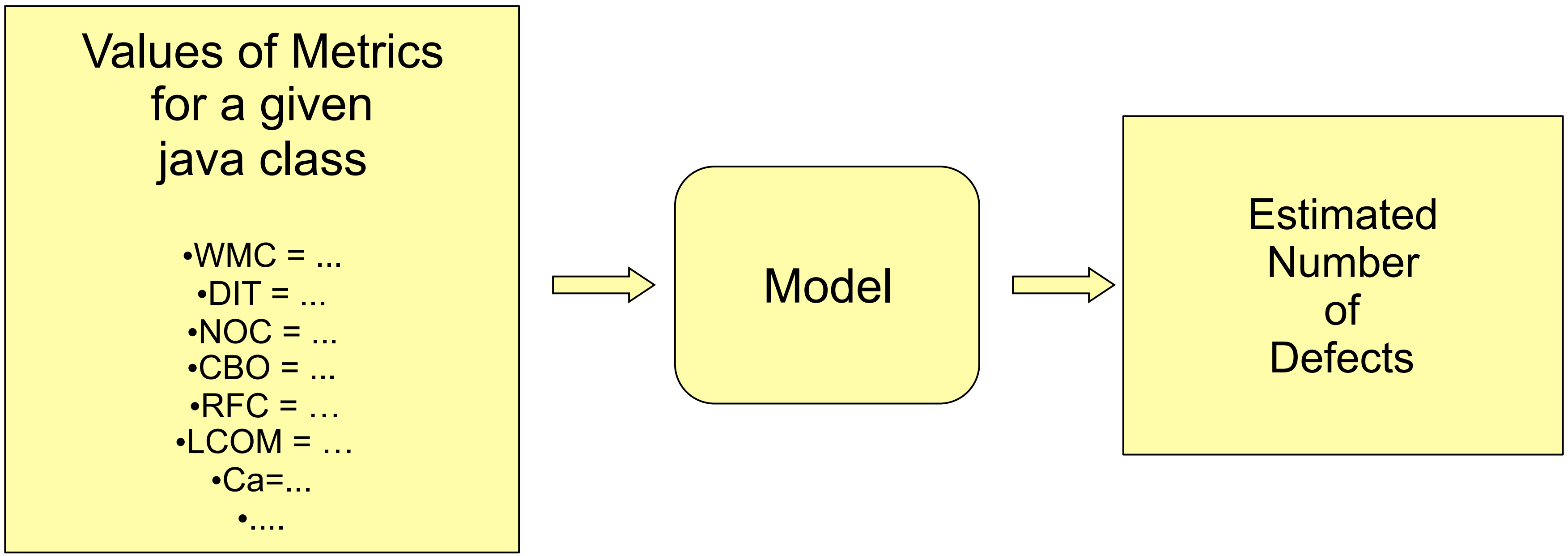
Motivation - Why clustering projects?

- Defect prediction is sometime impossible because lack of training data:
  - It may be the first release of a project
  - The company or the project may be to small to afford collecting training data

- With well defined project clusters the cross-project defect prediction will be possible

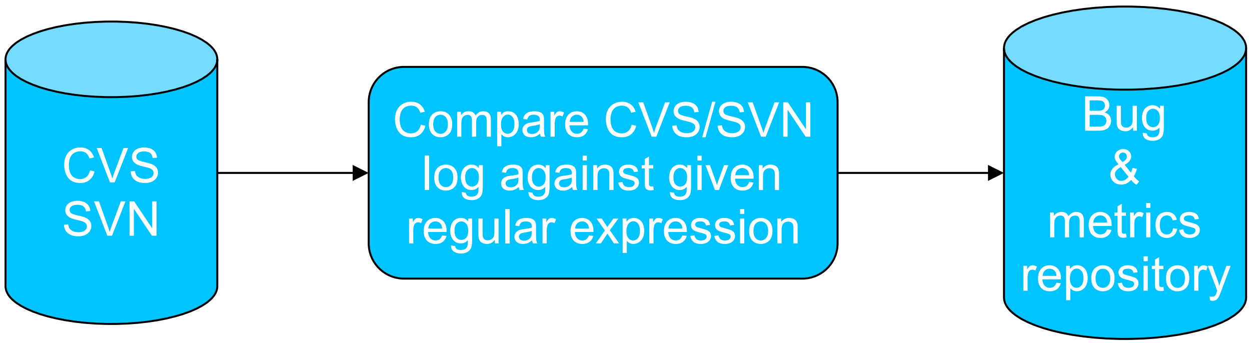
Definitions

- Defect
  - Interpreted as a defect in the investigated project
  - Commented in the version control system (CVS or SVN)
- Defect prediction model



Data acquisition

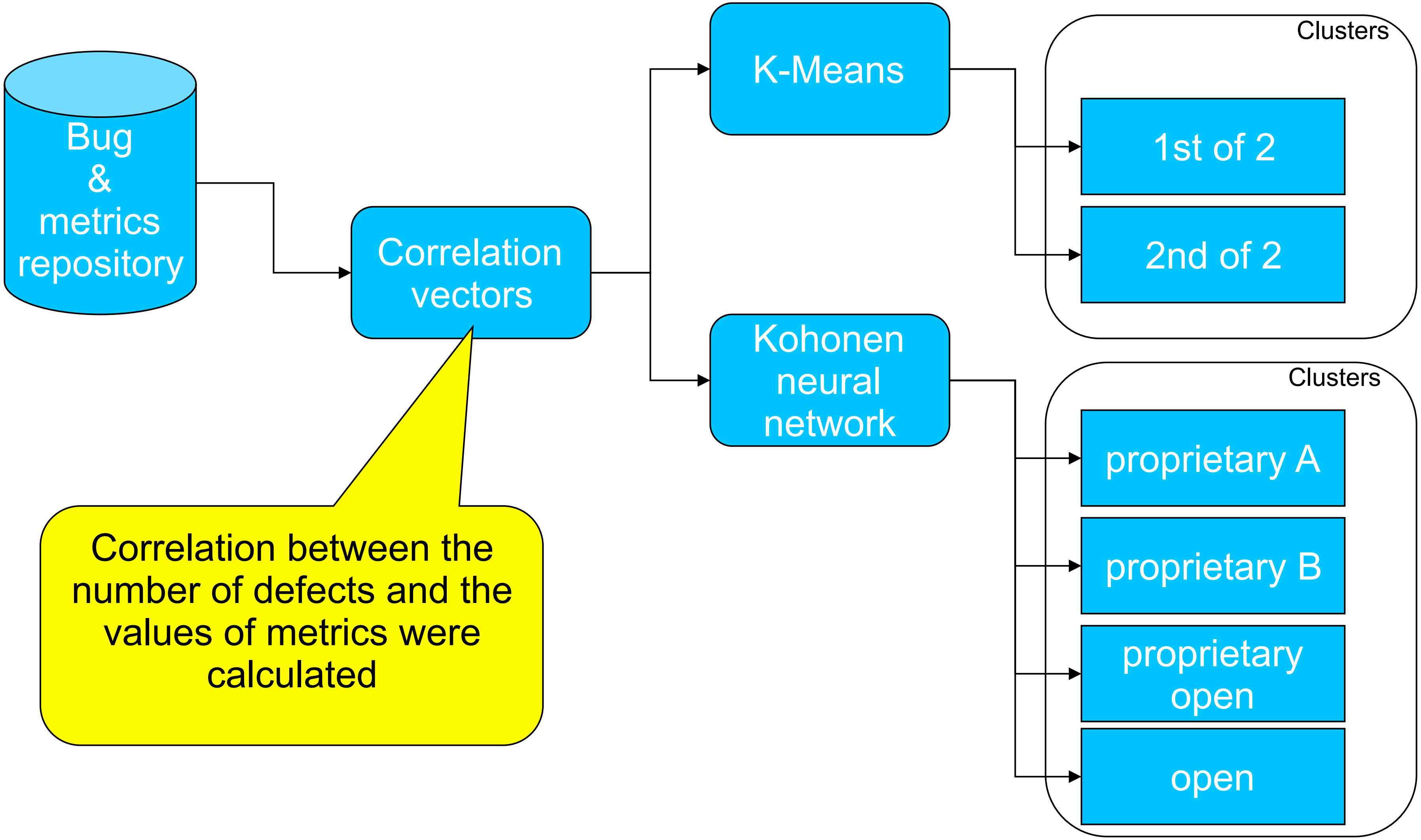
- 19 different metrics were calculated with the CKJM tool (http://gromit.iiar.pwr.wroc.pl/p_inf/ckjm)
  - Chidamber & Kemerer metrics suite
  - QMOOD metrics suite
  - Tang, Kao and Chen’s metrics (C&K quality oriented extension)
  - Cyclomatic Complexity, LCOM3, Ca, Ce and LOC
- Defects were collected with BugInfo (http://kenai.com/projects/buginfo)



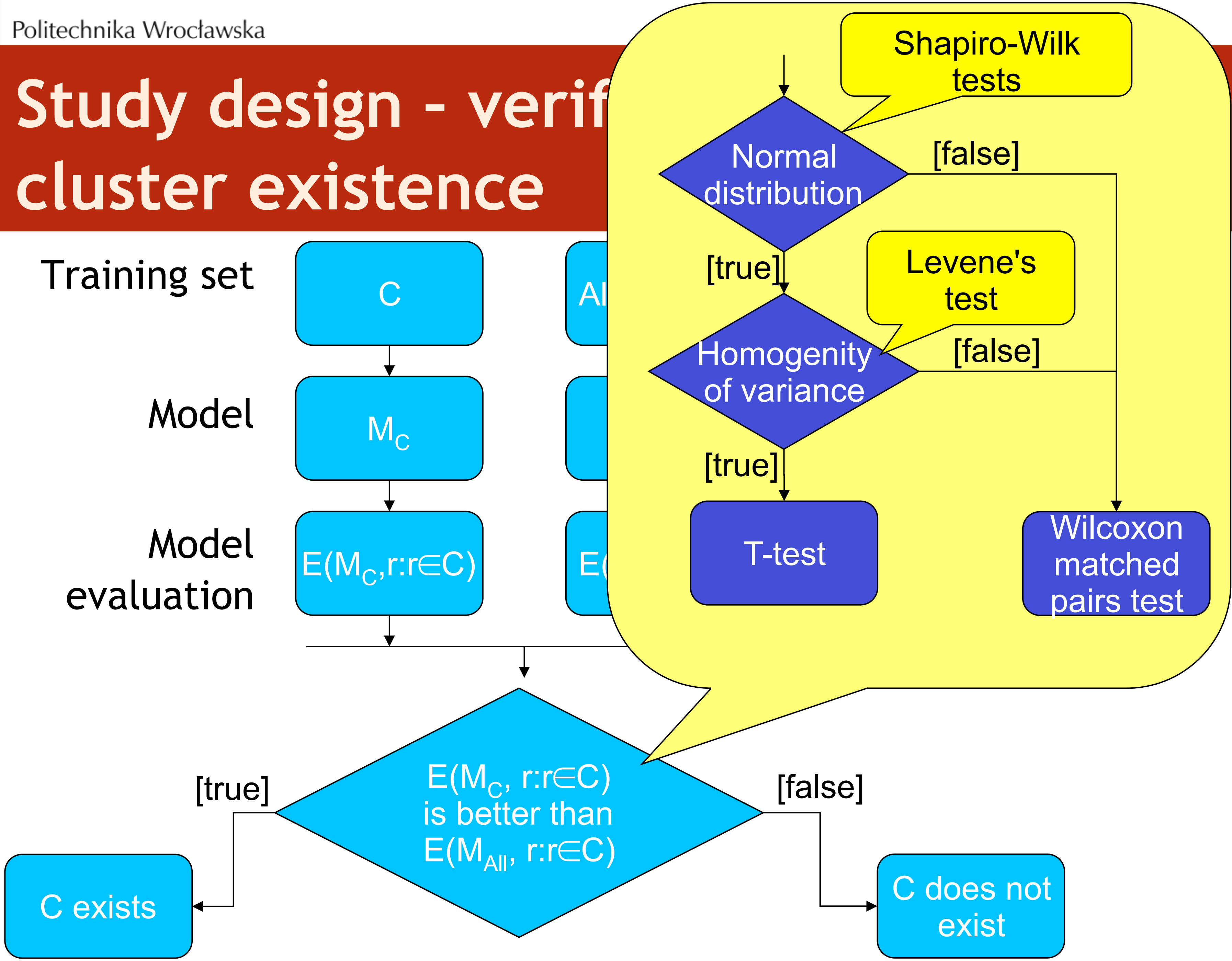
Data acquisition

- 92 versions of 38 projects were analysed
  - 6 proprietary projects *(5 custom build solutions from insurance domain, 1 quality assurance tool)*
  - 17 academic projects
  - 15 open-source projects (*Apache Ant*, *Apache Camel*, *Ckjm*, *Apache Forrest*, *Apache Ivy*, *JEdit*, *Apache Log4j*, *Apache Lucene*, *PBeans*, *Apache POI*, *Apache Synapse*, *Apache Tomcat*, *Apache Velocity*, *Apache Xalan-Java*, *Apache Xerces*)
- Metrics Repository (http://purl.org/MarianJureczko/MetricsRepo)

Study design - clustering



Study design - verif cluster existence



Results

Cluster	Is the cluster model better?	P value (statistical test)
1st of 2	YES	0.954
2nd of 2	NO	-
proprietary A	NO	-
proprietary B	YES	0.035
proprietary / open	YES	0.005
open-source	NO	-

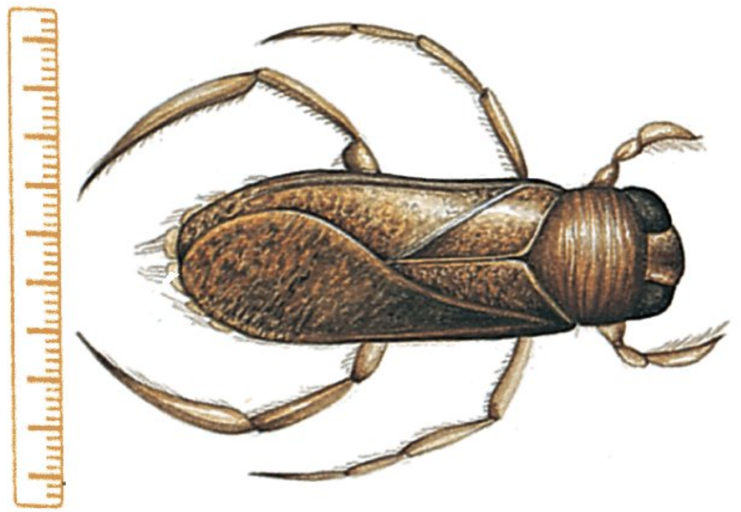
Results

- Cluster 'Proprietary B'
  - custom build solutions;
  - heavy weight, plan driven development process;
  - already installed in the customer environment;
  - insurance domain;
  - manual tests;
  - similar development period;
  - use database;
  - proprietary – the same company.

- Cluster 'proprietary / open'
  - text processing domain;
  - SVN and Jira or Bugzilla used;
  - medium size international team;
  - automatization in the testing process;
  - do not use database

Conclusions

- 92 releases of 38 proprietary, open-source and academic projects were analysed
- 2 methods of clustering were applied
- 6 clusters were identified and the existence of 2 of them were proven



Politechnika Wrocławska

Thank You
for Your attention