

# The Ptidej Team in a Nutshell

Yann-Gaël Guéhéneuc

MoSART

2009/01/16

# History

- Established circa 2004
- Alumni
  - Simon, post-doc. (1)
  - Naouel, Ph.D. (1)
  - Paul, Saliha, Gerardo, Jean-Yves, Norddin, Olivier, Khashayar, Janice, Rabih, Samah, Yousra, M.Sc. (11)
  - 52 B.Sc.



# As of Today

## ■ Members

- Gerardo, associate (1)
- Ferdaous, Salima, Foutse, Abdou, Usman, Amine, Ph.D. students (6)
- Marwen, Francois, Mathieu, Julien, Wei, M.Sc. Students (5)
- Diane, B.Sc. Trainee (1)

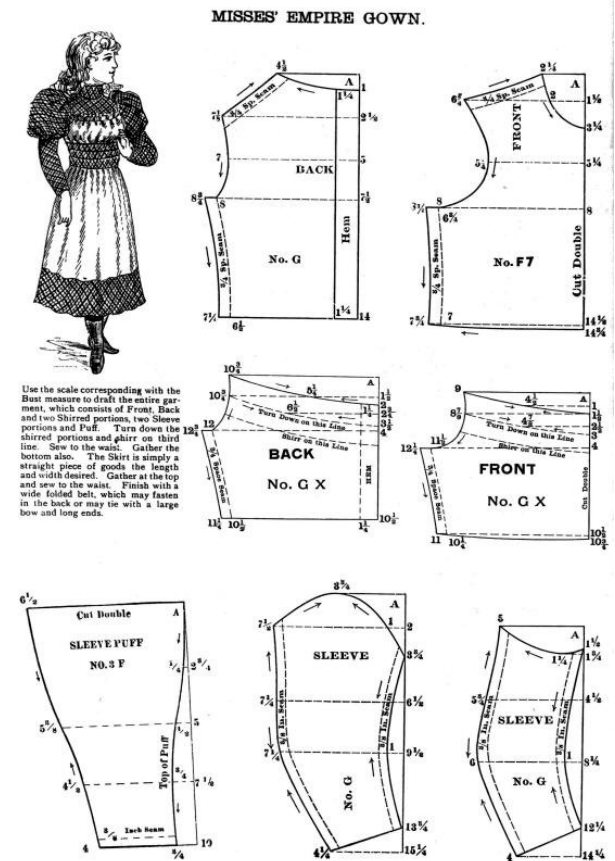
## ■ Publications

- 8 journal articles/chapters
- 57 conference papers



# Mission Statement

- To develop theories, methods, and tools, to evaluate and to improve the quality of object-oriented programs by promoting the use of idioms, design patterns, and architectural patterns



# Themes and Roadmap



## ■ Design defects

- Operational descriptions
- Automated generation of detection algorithms
- Automated generation of suggestions of correction
- Automated application of corrections
- Evolution and relations with external information

## ■ Design patterns (structural)

- Operational descriptions
- Automated detection (using explanation-based constraint programming)
- Automated detection (using efficient bio-informatic algorithms)
- Quality characteristics
  - Quantitative
  - Empirical (machine learning and visualisation)

## ■ Feature identification

- Epidemiological metaphor

## ■ Design patterns (behavioural, creational)

- Reverse-engineering of sequence diagrams
- Operational descriptions
- Automated detection (using explanation-based constraint programming)

## ■ Evolution Patterns

- Co-changing files
- Traceability of refactorings and other transformations
- Framework-Clients Evolution
- Architecture as a kind of DNA

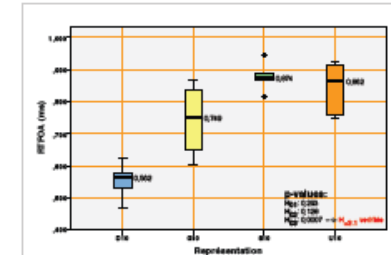
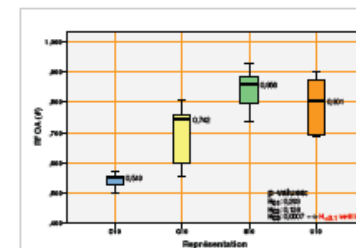
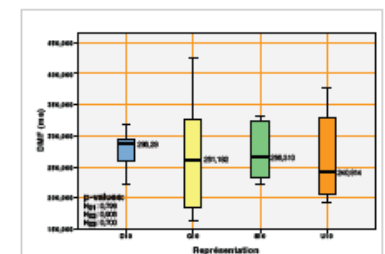
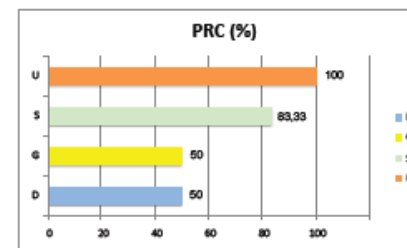
## ■ Understanding program comprehension

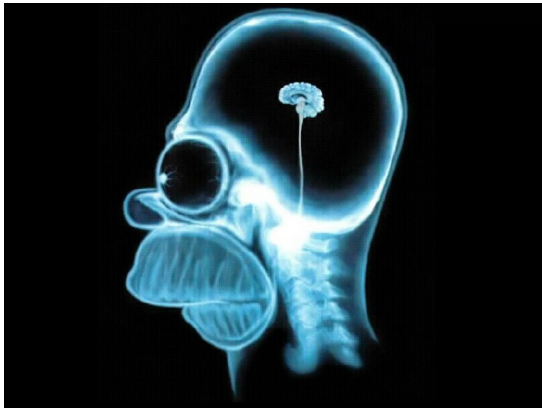
- Theories of program comprehension with design patterns
- Experiments on program comprehension and patterns using eye-trackers
- Advanced 3D visualisation techniques

# (Very) Recent Results

- Relations between defects and bugs
- Suggestion of modifications to clients following framework evolution
- Impact of playing roles in some design patterns on metrics

- Impact of visualisation techniques on comprehension





# Open Questions

- How to merge several scenario diagrams to obtain a sequence diagram?
- How to model software evolution flexibly and easily?
  - Transformation traceability
  - Meta-modeling