2018-00876 - PHD : Multi-Facet Actionable Analytics for Information System Rejuvenation M/F

Level of qualifications required: Graduate degree or equivalent
Function: PhD Position

About the research centre or Inria department

The Inria Lille - Nord Europe Research Center was founded in 2008 and employs a staff of 360, including 300 scientists working in sixteen research teams. Recognised for its outstanding contribution to the socio-economic development of the Nord - Pas-de-Calais Region, the Inria Lille - Nord Europe Research Centre undertakes research in the field of computer science in collaboration with a range of academic, institutional and industrial partners.

The strategy of the Center is to develop an internationally renowned centre of excellence with a significant impact on the City of Lille and its surrounding area. It works to achieve this by pursuing a range of ambitious research projects in such fields of computer science as the intelligence of data and adaptive software systems. Building on the synergies between research and industry, Inria is a major contributor to skills and technology transfer in the field of computer science.

Context

In the context of an industrial relationship with the CIM company we are proposing a PhD on software analysis and information systems. The context of the PhD is really nice for the candidate: The company is interested in hiring the PhD after the thesis if everything goes well. In addition an engineer will help building tools at the beginning of the project.

Assignment

Context and Challenges

Information Systems are one of the key software backbones of our society and economy. They manage key data of our lives and activities: Insurance, payroll, CRM, or human resource management system. Often they are the cornerstone of organisations and key enablers of revenue.

Reverse engineering, maintenance and evolution of software assets such as Information Systems has been identified by Deloitte as one of the 10 future breakthroughs in IT.

Old Languages. Very often, Information Systems’ lifetime spans decades. They survive technology hype. But the counter part is that they are developed in programming languages that seem old and out of fashion compared to modern technology. For example, half of the business of a large insurance group is programmed in a language that does not exist anymore according to Google.

Aging Software. Since information systems grew over a long period of time, the underlying software is aging. It frequently contains dead or duplicated code, obsolete documentation, lack of tests. Since original developers are often not longer part of the project, the overall knowledge of the application is scattered and incomplete.

Lack of Tools. Old languages lack modern tooling such as metrics, refactoring, test coverage, therefore it is difficult to extract exact information and control the evolution of an Information Systems. For example, performance analysis is often difficult to do because there are no off-the-shelf tools for old language.

Lack of Knowledge. It is often difficult to understand the flow of information and processes embedded in the software. Over the years, the systems had to interact with different technologies (REST, web services, ...) that may not even exist anymore. Yet this had an impact on the architecture of the system. Regularly past architecture decisions are lost and new changes unknowingly break basic assumptions or important invariants.

Changes at high risk. The lack of knowledge coupled to the fact that there is often no or limited test available, turn any change into a very risky task. Developers are then hampered to do more than bug fixes or immediate client requirements.

Main activities

Objectives

The goal of the PhD is to support the ‘Rejuvenation of Information Systems’. The experiments and validation of the results will happen in the context of the PowerBuilder Information System of the CIM company. To support the PhD, CIM is paying an expert engineer to build infrastructure (parser, meta-model) dedicated to PowerBuilder for the Moose open-source platform. The student will use and extend this infrastructure (software maps, quality assistant) for building new generation tools.

The student will work on the following challenges:

- Reverse engineering. Reverse engineering is not new. However, extracting key views that support decision making is complex since it depends on local context (business, process, framework constraints). Such contextual approach does not have a formal frame but it advocated by “Actionable Analytic”[1]. The student will work on how to support the reverse engineering of Information Systems taking into account their local context. This reverse engineering will integrate information from various different sources such as structural information, data flow between identified components, authors, bug reports, etc. This is a complex task because of the intrinsic complexity of the legacy and the local context.

- Actionable quality assessment. The student will develop domain and language specific quality assessment maps. The quality assessment will provide reports and maps about dead code, code duplication, specific metrics adapted to the language and the domain (form,
- **Run-time analysis and program charge.** CIM is planning to expand on a new market of large insurance companies. It is worried that its products may not scale up to the amount of data this will imply. There is a need to identify and understand optimization opportunities. How can we support understanding the run-time performances of an information system? This is a complex task because performance gains and losses are spread among different software layers (graphical interface, telecommunications, core application, database), and it is not clear where one should focus. In addition, instrumenting the legacy code and the other systems interacting with it (e.g., the database back-ends) is not straightforward.

### Identified Tasks

The student will work incrementally in 3 months “sprints” on the following tasks:

- Learn Powerbuilder, Moose and meta-modeling, literature review
- Identify contextual information (local/team patterns, frameworks constraints, processes).
- Define actionable metrics or queries.
- Build first actionable analysis (local anti-pattern identification) and related maps.
- Validate with development team.
- Build first run-time actionable analysis (database anti-pattern identification) and related maps.
- Identify concrete run-time bottlenecks.

### RMOD Supervisors

- Stéphane Ducasse (program understand, analyses, tooling), Nicolas Anquetil (code analysis, quality metrics, program transformation) et Anne Etien (tests, database, information systems).

### Advanced engineer

Guillaume Larcheveque


### Skills

Expected know how:

- Pharo [http://www.pharo.org](http://www.pharo.org)
- Moose [http://www.moosetechnology.org](http://www.moosetechnology.org)

### Metamodeling

Languages:

- English
- French is an advantage

### Relations

- Good communication
- Fast learner
- Good writing

### Additional skills

- Databases
- Program analysis
- Language semantics
- Software metrics
- Code quality

### Benefits package

- Subsidised catering service
- Partially-reimbursed public transport
- Paid leave
- Flexible working hours
- Sports facilities