SSaaPP:
SpreadSheets as a Programming Paradigm

Project's Final Report

Rui Abreu, Tiago Alves, Orlando Belo, José C. Campos, Jácome Cunha, João Paulo Fernandes, Pedro Martins, Jorge Mendes, Hugo Pacheco, Christophe Peixoto, Rui Pereira, Alexandre Perez, Hugo Ribeiro, André Riboira, João Saraiva[*]**, André Silva, João Carlos Silva, Joost Visser**

[*] **Project's Principal Investigator**

`jas@di.uminho.pt`

TR-HASLab:02:2014
SSaaPP:
SpreadSheets as a Programming Paradigm

Project's Final Report
by Rui Abreu, Tiago Alves, Orlando Belo, José C. Campos, Jácome Cunha,
João Paulo Fernandes, Pedro Martins, Jorge Mendes, Hugo Pacheco,
Christophe Peixoto, Rui Pereira, Alexandre Perez, Hugo Ribeiro, André
Riboira, João Saraiva*, André Silva, João Carlos Silva, Joost Visser

* Project's Principal Investigator

Abstract

This technical report describes the research goals and results of the SpreadSheet as
a Programming Paradigm research project. This was a project funded by Fundação
para a Ciencia e Tecnologia – FCT: the Portuguese research foundation, under reference
`FCOMP-01-0124-FEDER-010048`, that ran from May 2010 till July 2013.

This report includes the complete document reporting the results achieved during
the project execution, which was submitted to FCT for evaluation on October 2013. It
describes the goals of the project, and the different research tasks presenting the deliverables of each of them. It also presents the management and result dissemination work
performed during the project's execution. The document includes also a self assessment of the achieved results, and a complete list of scientific publications describing the
contributions of the project. Finally, this document includes the FCT evaluation report.

1 Aims and Work Plan

Spreadsheet tools can be viewed as programming environments for non-professional programmers. These so-called "end-user" programmers vastly outnumber professional programmers. In fact, spreadsheets, when viewed as a programming language, are one of the largest programming languages and can be characterized as a particularly low-level one: there is no support for abstraction, testing, encapsulation, modular or structured programming. As a result, numerous studies have shown that existing spreadsheets contain errors at an alarmingly high rate. In fact, companies are being put at risk due to their failure to realize that the process of constructing spreadsheets requires the discipline of traditional programming. Spreadsheet applications are more vulnerable to poor design and to errors than conventional programs. This means that a greater degree of discipline is required in the process of spreadsheet development.

Recent advances in programming languages are providing their users with powerful mechanisms to structure and develop their programs. Indeed, the development of powerful type systems, modular systems, abstract models, support for generic programming, support for domain specific language embedding, lazy evaluation engines and incremental models of computations are changing the way we write our programs. Moreover, the development of techniques for refactoring, testing and slicing are also changing the way we construct, transform and maintain our software. Surprisingly, with the exception of the pioneering work of Erwig and Burnett, little work has been done on the foundations of spreadsheets by the programming language community. As a result, none of those developments have been incorporated in spreadsheets yet. In fact, spreadsheets still do not offer support for abstraction, encapsulation, or structured programming yet! For this reason, it is fair to state that spreadsheets could benefit from all recent developments in programming languages even more than programming languages themselves.

The overall objective of this project is to study and develop techniques to incorporate these recent advances in programming languages, in spreadsheets. Thus, we will study new type systems and modular systems, and formal models for spreadsheets. Furthermore, we will develop techniques to reason about existing spreadsheets. That is to say that we want to define techniques to extract the model of a spreadsheet, to define model based techniques for spreadsheets, to define a catalog of spreadsheet smells, to apply software evolution techniques to spreadsheets, and to define new incremental computational models for spreadsheets.

The SSaaPP project proposes to study the foundations of spreadsheets both from a software engineering and theoretical point of view. Furthermore, the research i be conducted in four axes: *analysis*, *transformation*, *evaluation* and *quality assessment of spreadsheets*. The project workplan/tasks directly follows the four axes which will conduct our research, namely, analysis (task 1), transformation (task 2), evaluation (task 3) and quality of spreadsheets (task 4). The project includes a specific task (task 5) devoted to the integration, benchmarking and documentation of the results produced in each of these axes.

This document is intended to describe in detail the development of the SSaaPP project. In section 2 we describe the contributions and deliverables of each of the tasks in the original plan. In Section 3 reports on management and dissemination activities. Finally, section 4 discusses what, in the proponent's opinion, has been achieved within SSaaPP and enumerates a few future research directions.

2 Work Plan Execution

2.1 Task T1: Analysis of Spreadsheets

The main goal of the analysis of spreadsheet task was to develop techniques to analyze spreadsheet data. This task was conducted in three main areas:

- the study of techniques to automatically infer a model representing the business logic of spreadsheet data.

- the use of software localization techniques to locate faults in spreadsheets

- the definition of a catalog of spreadsheet (bad) smells and the detection of such smells.

To infer the business logic expressed by spreadsheet data, we adapted data mining techniques to analise spreadsheet data so that a model defining the business logic of the spreadsheet data is automatically computed. As described in project proposal, in *Year 1* we developed techniques to infer classSheet models from spreadsheet data. We adapted database theory and applied data mining algorithms to reason about spreadsheet data. This was joint work with Prof. Martin Erwig, a consultant of the SSaaPP project. We published this work at VL/HCC 2010 (class A, according to the CORE conference ranking) [8]. In the following year, we developed the HaLeX tool, that was presented in the tool demo session at the next edition of the VL/HCC conference [14]. In order to extract the behaviour model of spreadsheet systems, we have also applied reverse engineering techniques and tools, namely the use of the GUISurfer tool, to infer models of interaction [38, 36, 37]. In this context, João Carlos Silva defended his PhD thesis [39] in 2010 (an expected deliverable of the project).

In 2010, Prof. Rui Maranhão joined our research group after receiving his PhD from Delft University on Software Fault Localization. Because spreadsheets are widely known to be error-prone, we invited Rui Maranhão to join the project's team, to work on spreadsheet fault localization. Thus, our work on fault localization started in the second year of the project, and we obtained results on at the end the *Year 2* [2] and in *Year 3* of the project [26, 1].

The work on spreadsheet fault localization opened another important research work: the detection of smells (but not faults) in spreadsheets. Following the seminal work of Fowler on bad smells on programs' source code, we proposed a MSc thesis to work on bad smells for spreadsheets. Hugo Ribeiro joined the team, supported by the planned BI grant, and developed a catalog of spreadsheets smells [11]. He defended his MSc thesis in 2011 [34], a deliverable expected in this task. We have implemented our catalog as the SmellSheet Detective tool, which was presented as a tool demo paper at VL/HCC 2012 [12].

Deliverables:

- 1 PhD thesis: [39]

- 1 MSc thesis [34]

- 10 papers

- 2 software tools: HaExcel [14], SmellSheet Detective [12].

2.2   Task T2: Transformation of Spreadsheets

The task on the transformation of spreadsheets aimed at use advanced regular programming language techniques to transform, refactor, and manipulate programs source code, in the context of spreadsheets. Having proposed on task 1 techniques to infer a model defining the business logic of spreadsheet data, we build on these results to transform spreadsheets such that,

- they provide a model-driven engineering approach to spreadsheet development.
- they support (type-safe) model evolution and instance co-evolution.
- they provide a powerful bidirectional engine for model/instance evolution/co-evolution.

The use of spreadsheet models to provide model-driven spreadsheet development, was first proposed by Engels and Erwig when they introduced the classSheet models. The proposed approach offered a limited form of model-driven engineering: because the classSheet model and the spreadsheet data (that is, the instance) were manipulated by different tools, the specific classSheet tool and the spreadsheet system, respectively, was impossible to have the two software artifacts synchronized. We follow a different approach: we developed an embedding of the domain specific visual modeling formalism, the classSheet model, within a general purpose spreadsheet system [21], which offer a full model-driven spreadsheet development environment to end-users. Indeed, both the spreadsheet model and the spreadsheet instance co-exist as different worksheets within the same spreadsheet, as as a result, we are able to co-evolve the instance after we evolve the model.

We have also defined two different techniques for the synchronization of spreadsheet models and instances. First, we use data refinement techniques that guarantee the type-safe evolution of classSheet models and the instance co-evolution [22]. Because, this approach only considers the model evolution and instance co-evolution, we developed bidirectional transformation techniques that allow for the instance evolution and the model co-evolution, as well [10]. A tool demo of this techniques was presented/published at ICSE 212 conference (class A*, according to the CORE conference ranking) [15]. Most of this work was done in the context of Jácome Cunha PhD thesis [6] who developed novel techniques for model-driven spreadsheet development, which were also published as a book [7]. Jorge Mendes in his MSc project, supported by the planned BI grant, continued this work and developed techniques for the evolution of model-driven spreadsheets [30]. Jorge presented his work on several (funded) PhD students events (with reviewing and formal publication), namely: ICSE 2012 Student Research Competition [28] and the VL/HCC 2012 Graduate Consortium [29]. Recently, we studied techniques to direct the evolution of model-driven spreadsheets, so that we can automatically evolve a source model into a known target model [35].

As a programming language/system paradigm, spreadsheets lack a query language to manipulate/filter its data, like, for example, SQL offer to databases. In his MSc project [33], Rui Pereira, supported by a BI grant, developed techniques for querying model-driven spreadsheets which offer a powerful query system for spreadsheets [13]. This query language was implemented in the QuerySheet tool [5].

Deliverables:

- 1 book
- 1 PhD thesis: [6]

- 2 MSc projects: [33], and [35]
- 9 papers
- 1 software tool: [5]

2.3   Task T3: Evaluation of Spreadsheets

The task on evaluation of spreadsheets at evaluating both the performance of spreadsheet systems and the productivity of spreadsheet end-users. This task was organized in two main areas:

- We studied how an efficient spreadsheet (incremental) engine could be model in a modern lazy, functional programming language
- we conducted several empirical studies to compare end users productivity when using standard spreadsheets versus model-driven spreadsheets.

Building a spreadsheet system is a complex and hard-working task: the developer needs to define/maintain not only a complex dependency graph to determine the re-evaluation order and propagate changes, but also a powerful incremental evaluation engine. Within the context of lazy functional languages, the developer does not have to handle such dependencies: the lazy machinery infers the dependencies and evaluation order at run-time. In fact, circular, lazy programs are an elegant and concise setting to express spreadsheet-like computations. Lazy programs, however, are not easily incrementalized: an important requirement in a spreadsheet setting. In [24] we presented techniques to transform a concise and elegant circular program into an equivalent strict one. The idea is that we are able to efficiently incrementalize the strict version of the circular one. Having defined techniques to strictify circular programs, we developed program calculational laws to derive such circular programs [31].

In order to evaluate spreadsheet users productivity using our model-driven spreadsheets, we performed several empirical studies with real end users. In [4] we conducted a large empirical study to compare in terms of performance and effectiveness of end users using regular and model-driven spreadsheets. At the ICSE workshop on *User Evaluation for Software Engineering Researchers* (USER 2012), we discussed an empirical study for our bidirectional evolution system [17]. Recently, and as part of the invited tutorial presented at the PhD Summer School on Domain Specific Languages, we performed an empirical study with the school participants. The results of this empirical study will appear in the lecture notes of the school [20].

Deliverables:

- 5 papers

2.4   Task T4: Quality of Spreadsheets

Spreadsheets have proven to be highly error-prone and very difficult to maintain and evolve without introducing serious flaws. For these reasons, new techniques are needed to help spreadsheet users evaluate and improve the quality of their spreadsheets. In task 1, we defined techniques to locate faults and (bad) smells in spreadsheets. In task 2, we defined techniques for (type-safe) model-driven spreadsheet evolution. In task 3, we conducted several empirical studies to assess end-users productivity. The goal of the quality of spreadsheet task is to define a quality model for spreadsheets.

Christophe Peixoto in his MSc project [32], funded by a BI grant, proposed a quality model for spreadsheets. This model is defined by introducing a set of domain specific metrics for spreadsheets which are used to measure concrete spreadsheet characteristics. Such characteristics are related to the general notions of software quality as defined on the ISO/IEC 9126 standard. To calibrate our quality model we use a large repository of real-world spreadsheet instances, namely the EUSES Spreadsheet corpus. This work was published in [18]. The use of software metrics and the calibration approach to assess software quality is proposed in Tiago Alves PhD thesis [3]. In [23] we defined a set of complexity metrics to assess the quality/complexity of classSheet models. Moreover, in [25], we present a pragmatic method for management of risks that arise due to spreadsheet use in large organizations. We combine peer-review, tool-assisted evaluation and other pre-existing approaches into a single organization-wide approach that reduces spreadsheet risk without overly restricting spreadsheet use.

Deliverables:

- 1 MSc thesis: [32]
- 1 PhD thesis: [3]
- 3 papers

2.5   Task T5: Spreadsheets Framework

Each of the previous tasks produced specific software prototypes, documentation and case studies. Moreover, different software prototypes were developed in different programming languages and software systems. For example, all the work on software evolution described on task 2 is implemented in the programming language Haskell. The embedding of the classSheet models in a spreadsheet system, is implemented as an addon to a widely, open source spreadsheet system: the libreOffice system. In this task, we developed techniques to integrate all the previous achievements into a more general and complete framework that includes libraries, tools, videos and webpages.

Jorge Mendes in his MSc thesis develop a bridge (in C++) between the libreOffice addon and the Haskell evolution library. Thus, he developed the MDSheet framework, that provides a model-driven spreadsheet development environment to end-users. This tool was used to perform the empirical studies described in Section 3. To conduct such studies, extensive documentation was produced so that end users not familiar with the environment could use it. A video showing a demo of the system is also available on the Internet (youtube). The MDSheet was presented at the ICSE'12 as a formal tool demo [16] and it is detailed describe in [30]. To make it possible to export classSheet models into other modeling settings, we defined a mapping from classSheet into UML+OCL [19], and we have also extended the classSheet model formalism [9]. Finally. we developed a framework for customizable software analysis, so that software systems like the MDSheet and the SmellSheet detective can be easily reused [27].

Deliverables:

- 1 MSc thesis: [30]
- 3 papers
- 1 software tool: MDSheet [16]

4

3 Management and Dissemination

3.1 Management

The management structure adopted for the SSaaPP follow the SSaaPP research proposal. The project was managed by the PI, who coordinated and monitored the progress of the overall project. The PI was responsible for organizing a weekly team meeting, that were scheduled for Friday afternoon. In each of the meetings, with some rare exceptions, one team member was asked to prepare 3 to 4 slides, so that he could present his work in a more organized way. The PI was responsible for scheduling such short/informal talks and to motivate team members to participate. These weekly meetings were attended by most of the team members, and were a very important moment to discuss new ideas and to share research results.

The PI was also responsible to coordinate the visits of our consultants. In fact, with the exceptions of Prof. Umut Acar and Prof. Eric Meijer, our consultants visited the team during the execution of the project. Prof. Martin Erwig attended the first workshop of the project, giving insightful comments about the project progress. He was also a member of the PhD committee of Jácome Cunha PhD thesis . Prof. Alberto Pardo visited our team twice (on *Year 1* and *Year 2*), and worked on the use of program calculation techniques to derive circular programs. Dr. Laura Beckwith visited the team on *Year 1* to help us conducting empirical end user studies. We have join publications with all these consultants. Prof. Umut Acar and Prof. Eric Meijer changed jobs (and countries) during the execution of the project, which made it difficult to visit our team. Because we did not spend the planned budget with such visits, we asked FCT to be able to re-assign such budget to an additional BI grant.

To manage all the research results achieved, the software prototypes produced and to disseminate our results, we used a collaborative (wiki) site. This site includes several subpages, namely:

- *Publications:* where all scientific papers and MSc and PhD thesis are available.

- *Software:* where all software prototypes are available for downloading

- *Activities:* where all activities of the project are described. For example, the visits of our consultants, the participation in conferences by team members, etc

- *Workshops:* we organized two SSaaPP workshops during the execution of the project. The program of those workshops are available here.

- *Empirical Studies:* the empirical studies we conducted produced several material that can be useful for the scientific community and that we refer in resulting scientific publications. This material is available in this sub-page.

3.2 Dissemination

The project scientific results were disseminated by 30 publications, and we have 5 papers under reviewing. Moreover, some of our scientific results were published at very prestigious events: ICSE, VL/HCC and ETAPS/FASE. As originally foreseen, the project let to 3 PhD thesis and 5 MSc thesis. In two cases the work of grant holders evolved into PhD projects that will be concluded after the project lifetime. Such are the cases of grant holders Jorge Mendes and Rui Pereira. Jorge Mendes is now doing a PhD thesis on extending the model-driven spreadsheet evolution to a multi-user environment. Rui

Pereira is starting the PhD thesis working on defining a bidirectional engine for his QuerySheet tool.

Moreover, the project promoted two SSaaPP workshops (see workshop programs in the projects webpage) where team members gave talks about their work in the project. These SSaaPP workshops were open to the research community.

In 2013 the PI was invited to give a tutorial on Spreadsheet Engineering at the PhD School on Domain Specific Languages, Cluj, Romania, July 2013. Springer Verlag already agreed to publish the final proceedings in LNCS Tutorial series in 2014.

The project produced several software prototypes. Such prototypes are available from the project webpage. To advertise our tools, we produced tool demo videos that are available in the Internet (youtube). The MDSheet tool video has more than 600 views.

Together with Pedro Faria, a BI grant holder, team members developed techniques to ensure confidentiality in spreadsheets in a multi-user environment. We are discussing with TecMinho the possibility of defining a patent on this work.

In 2013, an error in a spreadsheet was widely discussed in the media: a research paper used by the European Union politics to justify its economic policy had a spreadsheet error. That type of spreadsheet error does not occur in our model-driven engineering approach, described in task 1 and 2. As a consequence, we had the opportunity to present our work to several Portuguese newspapers (*Jornal de Negócios*, *Jornal Económico*, *Exame Informatica*, etc) and our team was on the news on national TV (TVI).

The PI served as chair of two international scientific meetings in areas of intervention of the SSaaPP project. Namely,

- *4th Summer School on Generative and Transformational Techniques in Software Engineering.* 3–9 July, 2011, Braga, Portugal. `http://gttse.wikidot.com/2011`

- *4th International Conference on Software Language Engineering.* 3–6 July, 2011, Braga, Portugal. `http://gttse.wikidot.com/2011`

The PI also served as editor of two special issues in the area of the project: Journal of Science of Computer Programnmning (special issue on the Brazilian Symposium on Programming Languages), Lecture Notes in Computer Science, Tutorial Series (lecture notes of the Summer School on Generative and Transformational Techniques in Software Engineering 2011).

4   Self Assessment

Globally the SSaaPP project was successful in meeting the proposed research aims and following the original work plan with almost no deviations. All scientific results and deliverables are listed above in Section 2. They have far exceeded the expected indicators.

In particular, it must be mentioned the project contribution with respect to

- Defining a model-driven spreadsheet development techniques, where a model specifies the business logic of the spreadsheet data. Moreover, the model guides end users in introducing correct data: only spreadsheet data updates that conform to the model are available.

- Defining advanced spreadsheet evolution, where end users can interact both with the spreadsheet model/instance, and having the automatic co-evolution of the instance/model.

- Defining techniques for fault and smell spreadsheet localization, and a quality model for spreadsheets.
- Defining program calculational techniques to express spreadsheets as circular, lazy programs.
- Performing empirical studies that show that our techniques do improve end user productivity.
- Providing several software prototypes, available as open source software to the scientific community, that offer model-driven engineering, querying, and smells detection in SpreadSheets

In terms of deliverables the project produced:

- 3 PhD thesis;
- 5 MSc thesis;
- 1 PhD School tutorial (invitation) with formal publication (to appear in 2014)
- 1 book
- 29 papers
- 1 papers in national conferences with peer-review;
- 4 software tools:
- 2 SSaaPP workshops;

5   FCT Evaluation Report

"A - Relatório Final APROVADO"

"Os objectivos científicos previstos foram plenamente atingidos. Os resultados evidenciam grande qualidade científica, nomeadamente ao nível das publicações em revistas internacionais com referee. O projecto contribuiu para a formação de jovens investigadores e para a projecção internacional da equipa envolvida."

"The project fully met the proposed scientific goals, with very good research outputs.

In fact, not only the goals were met, but also some more, unforeseen results were obtained, such as the development of a query language for spreadsheets developed according to the model of MDSheet. The ambitious goals in terms of pos-graduate education (3 PhDs and 5 MScs) were achieved. The number of expected research publications was clearly exceeded, some of them in very prestigious conferences. Some of the publications already got, in just a very short period, a good number of citations (half a dozen papers with over 20 citations each in less than two years). The publications listed in the final report are clearly relevant to the project, and its initial goals. The project also provided interesting and usable prototypical tools.

This is a good example of a very successful project."

References

[1] Rui Abreu, Jácome Cunha, João Paulo Fernandes, Pedro Martins, Alexandre Perez, and João Saraiva. Smelling faults in spreadsheets. In *Fundamental Approaches to Software Engineering (submitted)*, 2014.

[2] Rui Abreu, André Riboira, and Franz Wotawa. Constraint-based debugging of spreadsheets. In *XV Ibero-American Conference on Software Engineering (CibSE12)*, April 2012.

[3] Tiago Alves. *Benchmark-based Software Product Quality Evaluation*. PhD thesis, University of Minho, May 2012.

[4] Laura Beckwith, Jácome Cunha, João Paulo Fernandes, and João Saraiva. End-users productivity in model-based spreadsheets: An empirical study. In *Proceedings of the Third International Symposium on End-User Development*, IS-EUD '11, pages 282–288, 2011.

[5] Orlando Belo, Jácome Cunha, João Paulo Fernandes, Jorge Mendes, Rui Pereira, and João Saraiva. Querysheet: A bidirectional query environment for model-driven spreadsheets. In *Proceedings of the 2013 IEEE Symposium on Visual Languages and Human-Centric Computing*, VLHCC '13, Washington, DC, USA, 2013. IEEE Computer Society. to appear.

[6] Jácome Cunha. *Model-based Spreadsheet Engineering*. PhD thesis, University of Minho, 2010.

[7] Jácome Cunha. *Model-based Spreadsheet Engineering: Using Relational Models to Improve Spreadsheets*. LAP LAMBERT Academic Publishing, 2012.

[8] Jácome Cunha, Martin Erwig, and João Saraiva. Automatically inferring classsheet models from spreadsheets. In *VL/HCC'10: IEEE Symp. on Visual Languages and Human-Centric Computing*, pages 93–100. IEEE Computer Society, 2010.

[9] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. Extension and implementation of classsheet models. In *Proceedings of the 2012 IEEE Symposium on Visual Languages and Human-Centric Computing*, VLHCC '12, pages 19–22, Washington, DC, USA, 2012. IEEE Computer Society.

[10] Jácome Cunha, João P. Fernandes, Jorge Mendes, Hugo Pacheco, and João Saraiva. Bidirectional transformation of model-driven spreadsheets. In *ICMT '12*, volume 7307 of *LNCS*, pages 105–120. Springer, 2012.

[11] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva Hugo Ribeiro. Towards a Catalog of Spreadsheet Smells. In *The 12th International Conference on Computational Science and Its Applications*, volume 7336 of *ICCSA'12*, pages 202–216. LNCS, 2012.

[12] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, Pedro Martins, and João Saraiva. Smellsheet detective: A tool for detecting bad smells in spreadsheets. In *Proceedings of the 2012 IEEE Symposium on Visual Languages and Human-Centric Computing*, VLHCC '12, pages 243–244, Washington, DC, USA, 2012. IEEE Computer Society.

[13] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, Rui Pereira, and João Saraiva. Querying model-driven spreadsheets. In *Proceedings of the 2013 IEEE Symposium on Visual Languages and Human-Centric Computing*, VLHCC '13, Washington, DC, USA, 2013. IEEE Computer Society. to appear.

[14] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. Haexcel: a model-based spreadsheet evolution system. Poster/Tool Demo at the 2011 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2011), September 2011.

[15] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. A bidirectional model-driven spreadsheet environment. In *Software Engineering (ICSE), 2012 34th International Conference on*, pages 1443–1444, 2012.

[16] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. MDSheet: A framework for model-driven spreadsheet engineering. In *Proc. of the 34rd Int. Conf. on Software Engineering (ICSE)*, pages 1412–1415. ACM, 2012.

[17] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. Towards an Evaluation of Bidirectional Model-driven Spreadsheets. In *USER'12: User evaluation for Software Engineering Researchers*, pages 25–28. ACM, 2012.

[18] Jácome Cunha, João Paulo Fernandes, Christophe Peixoto, and João Saraiva. A quality model for spreadsheets. In *Proceedings of the 8th International Conference on the Quality of Information and Communications Technology, Quality in ICT Evolution Track*, pages 231–236, 2012.

[19] Jácome Cunha, João Paulo Fernandes, and João Saraiva. From Relational ClassSheets to UML+OCL. In *Proceedings of the Software Engineering Track at the 27th Annual ACM Symposium On Applied Computing (SAC 2012)*, pages 1151–1158. ACM, 2012.

[20] Jácome Cunha, João Paulo Fernandes, and João Saraiva. Spreadsheet engineering. In Viktória Zsók, Zoltán Horváth, and Rinus Plasmeijer, editors, *Summer Schhol on Domain Specific Languages, Central European Functional Programming School*, Lecture Notes in Computer Science, Berlin, Heidelberg. Springer-Verlag (to appear).

[21] Jácome Cunha, Jorge Mendes, João Paulo Fernandes, and João Saraiva. Embedding and evolution of spreadsheet models in spreadsheet systems. In *IEEE Symp. on Visual Lang. and Human-Centric Comp.*, pages 186–201. IEEE, 2011.

[22] Jácome Cunha, Joost Visser, Tiago Alves, and João Saraiva. Type-safe evolution of spreadsheets. In *Proc. of the 14th Int. Conf. on Fundamental Approaches to Software Engineering (FASE)*, pages 186–201. Springer-Verlag, 2011.

[23] Jácome Cunha, João Paulo Fernandes, Jorge Mendes, and João Saraiva. Complexity metrics for classsheet models. In Beniamino Murgante, Sanjay Misra, Maurizio Carlini, CarmeloM. Torre, Hong-Quang Nguyen, David Taniar, BernadyO. Apduhan, and Osvaldo Gervasi, editors, *Computational Science and Its Applications – ICCSA 2013*, volume 7972 of *Lecture Notes in Computer Science*, pages 459–474. Springer Berlin Heidelberg, 2013.

[24] João Paulo Fernandes, João Saraiva, Daniel Seidel, and Janis Voigtländer. Strictification of circular programs. In *Proceedings of the 20th ACM SIGPLAN workshop on Partial evaluation and program manipulation*, PEPM '11, pages 131–140, New York, NY, USA, 2011. ACM.

[25] Miguel A. Ferreira and Joost Visser. Governance of spreadsheets through spreadsheet change reviews. In *Proceedings of the European Spreadsheet Risks Interest Group*, EuSpRIG '12, July 2012.

[26] Birgit Hofer, André Riboira, Franz Wotawa, Rui Abreu, and Elisabeth Getzner. On the empirical evaluation of fault localization techniques for spreadsheets. In Vittorio Cortellessa and Dániel Varró, editors, *Fundamental Approaches to Software Engineering*, volume 7793 of *Lecture Notes in Computer Science*, pages 68–82. Springer Berlin Heidelberg, 2013.

[27] Pedro Martins, Nuno Carvalho, JoãoPaulo Fernandes, JoséJoão Almeida, and João Saraiva. A framework for modular and customizable software analysis. In Beniamino Murgante, Sanjay Misra, Maurizio Carlini, CarmeloM. Torre, Hong-Quang Nguyen, David Taniar, BernadyO. Apduhan, and Osvaldo Gervasi, editors, *Computational Science and Its Applications – ICCSA 2013*, volume 7972 of *Lecture Notes in Computer Science*, pages 443–458. Springer Berlin Heidelberg, 2013.

[28] Jorge Mendes. Coupled evolution of model-driven spreadsheets. In *Software Engineering (ICSE), 2012 34th International Conference on*, pages 1616–1618, 2012.

[29] Jorge Mendes. Model-driven spreadsheets in a multi-user environment. In *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium on*, pages 231–232, 2012.

[30] Jorge Cunha Mendes. Evolution of model-driven spreadsheets. Master's thesis, Minho University, 2012.

[31] Alberto Pardo, João Paulo Fernandes, and João Saraiva. Shortcut fusion rules for the derivation of circular and higher-order programs. *Higher-Order and Symbolic Computation*, 24(1-2):115–149, 2011.

[32] Christophe Peixoto. A quality model for spreadsheet. Master's thesis, Minho University, 2010.

[33] Rui Pereira. Querying for model-driven spreadsheets. Master's thesis, Minho University, 2013.

[34] Hugo Ribeiro. Smells in spreadsheets. Master's thesis, Minho University, 2010.

[35] André Silva. Directed evolution of model-driven spreadsheets. Master's thesis, Minho University, 2013.

[36] J. C. Silva, J. C. Campos, and J. A. Saraiva. Gui inspection from source code analysis. *Electronic Communications of the EASST*, 33: Foundations and Techniques for Open Source Software Certification 2010), 2010.

[37] J. C. Silva, C. E. Silva, J. C. Campos, and J. A. Saraiva. Gui behavior from source code analysis. In O. Mealha, J. Madeira, D. Tércio, and B.S. Santos, editors, *4a. Conferência Nacional em Interacção Humano-Computador (Interacção 2010)*, pages 81–88. GPCG, 2010.

[38] João Carlos Silva, Carlos Silva, Rui D. Gonçalo, João Saraiva, and José Creissac Campos. The guisurfer tool: towards a language independent approach to reverse engineering gui code. In *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems*, EICS '10, pages 181–186, New York, NY, USA, 2010. ACM.

[39] João Carlos Silva. *GUISurfer: A Generic Framework for Reverse Engineering of Graphical User Interfaces*. PhD thesis, University of Minho, 2010.