Reverse Engineering of Data Models from Legacy Spreadsheets-Based Systems: An Industrial Case Study

discussion paper

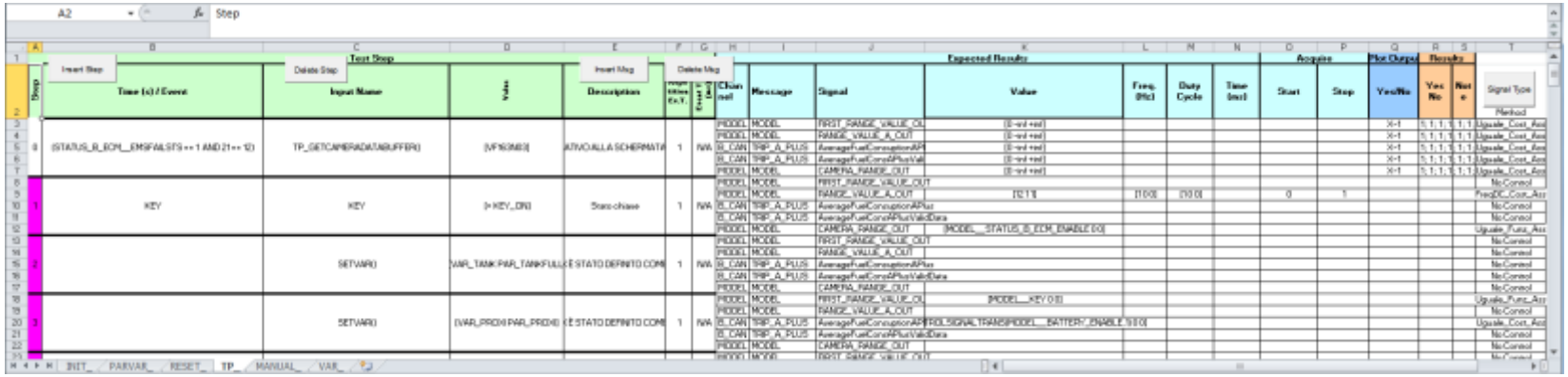
Domenico Amalfitano
Anna Rita Fasolino
Valerio Maggio
Porfirio Tramontana
Vincenzo De Simone

Spreadsheet Based Information System Issues

- Spreadsheets are designed only for computing purposes and commercial applications but …
- … very often they are used as Information Systems
  - Very difficult to maintain
    - High rate of duplicated data between different sheets and files
- The first and more critical step of a migration process is the **Data Reengineering**

Case Study

- An automotive company collects the specification of the tests executed on the vehicles in form of **Test Patterns**
  - Test Patterns are implemented in Excel files following a common template
- We have 30,615 different Excel files with 2,700 data cells on average
  - There is a high rate of replication data
    - 50% of data cells recurred more than 100 times
- Excel Test Patterns represent the input of an automatic test generation process



Data Model Reverse Engineering

- Data Model Reverse Engineering is the first step of a more general migration process towards a Web MVC architecture
- An heuristic based approach to infer the Data Model was proposed.
- A set of 26 heuristics were considered.
  - 11 heuristics derived from the literature and were adapted to work in this specific context.

Data Model Reverse Engineering

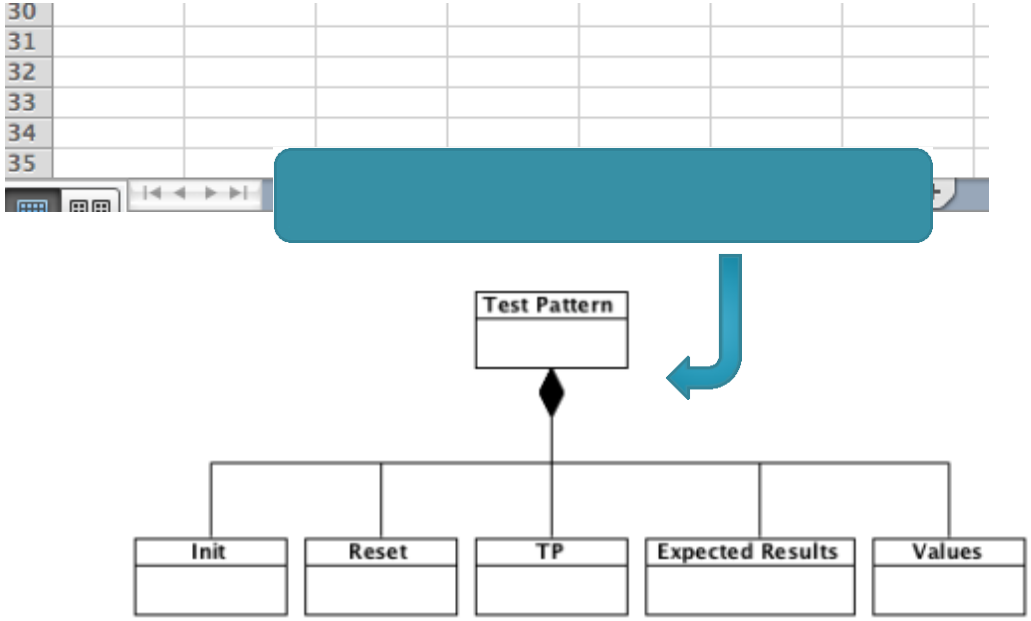
- Heuristics can be grouped in two main classes:
  - Structure based rules (SBRs)
  - Information based rules (IBRs)

Structure based rules (SBRs)

- SBRs analyze the structure and the properties of spreadsheets and their components, such as sheets, cells, cell headers, etc.
  - Used to abstract the set of candidate classes and their relationships;
  - Applied to a single Excel File.

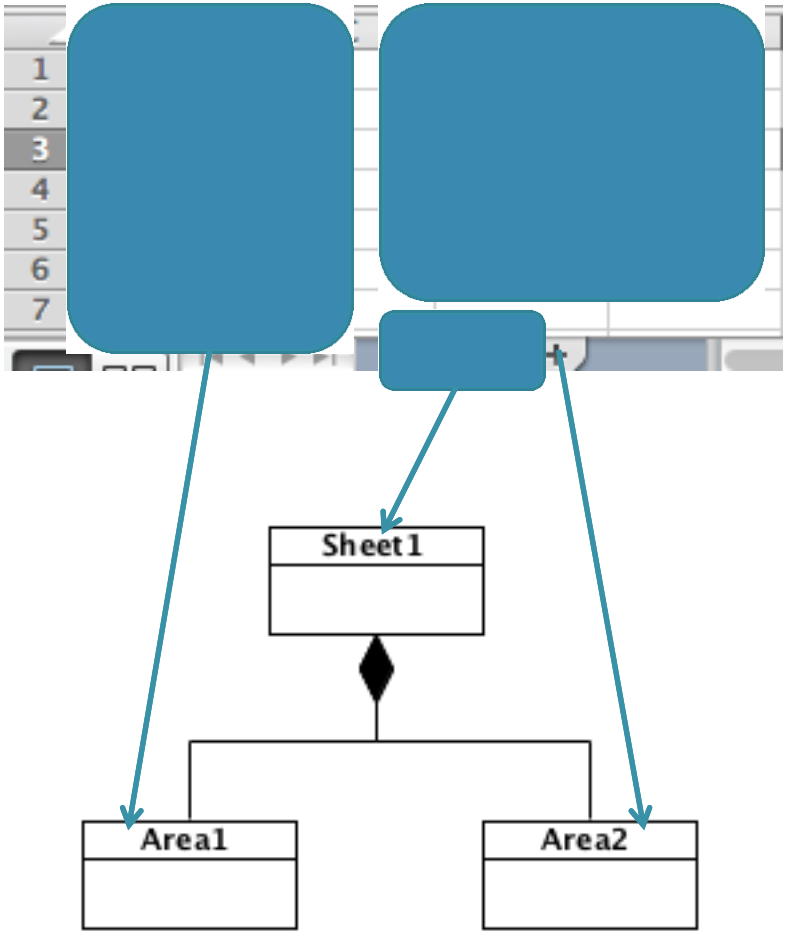
Example of SBR

Rule:
If the spreadsheet contains more than one sheet, **then** it is possible to associate the spreadsheet to a class C and each component sheet to a distinct class S_i, where C has a UML composition relationship with each S_i.



Example of SBR

Rule:
If a sheet S contains sets of consecutive non-empty cells (hereafter *non-empty cell area*) that are well delimited from each other by means of empty cells, **then** it is possible to associate each non-empty cell area to a single class C_i and the sheet S to a candidate class C_S, where S has a UML composition relationship with each C_i.

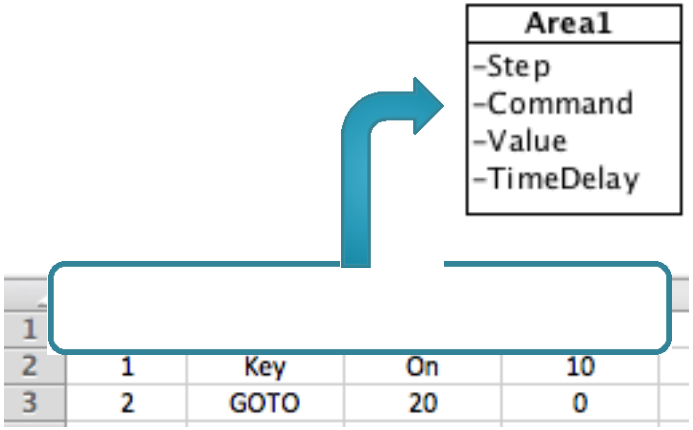


Information based rules (IBRs)

- IBRs analyze the informative content of the cells by looking for repeated data, synonyms, and cells containing well-defined data structures such as array strings, integer matrixes, etc.
  - Used to infer the attributes of classes, the relationships between classes and their cardinalities.
  - Applied to all the Excel Files

Example of IBR

Rule:
If the header cells of the columns that discriminated the extraction of a given class A assume the same textual content in all the spreadsheets, **then** these values may be considered attributes of that class.



Process Execution and Results

- Selected groups of rules were iteratively applied to the spreadsheets.
- Sets of candidate classes and relationships were automatically proposed.
- The data model made by 18 classes, 27 relationships, and 95 attributes was reconstructed at the end of the process.
- Candidates were submitted to domain experts who chose to accept, to refine or to reject them.
  - Experts accepted 75% of candidates inferred by means of SBRs and 33% of candidates inferred by IBRs