Formal automated software measurement plan.

Sarah DAHAB

Telecom SudParis, CNRS UMR 5157, Univ. Paris-Saclay, France

Supervisor: Prof. Stephane MAAG

Plan

- **Context**
- **Problematics**
- **Motivations & Objectives**
- **State of Art**
- **Our approach**
- **Conclusion & Perspectives**

Context

- **Software measurement**
  - More complex systems
  - Current measurement process is no longer adapted :
    - Sequential
    - Static
    - Lots of data to manage and analyze
  - ➔ Heavy measurement process
  - High demand for adapted measurement

➔ **Improvement of software measurement process**

Problematics

- **Lack of formal basis in SW measurement**
  - Formal metric definition & Measurement plan
- **Heavier measurement management load**
  - Static measurements plan
    - Planed at the beginning
  - Sequential analysis
  - Resources and Time consumption/cost
  - Useless metrics (at some times)
- **Huge amount of data to analyze**
  - More properties to be evaluated (system complexity)
  - Need of great measurement coverage
  - Difficulties to find the properties of interest (cross information…)
- **Expert-dependent**

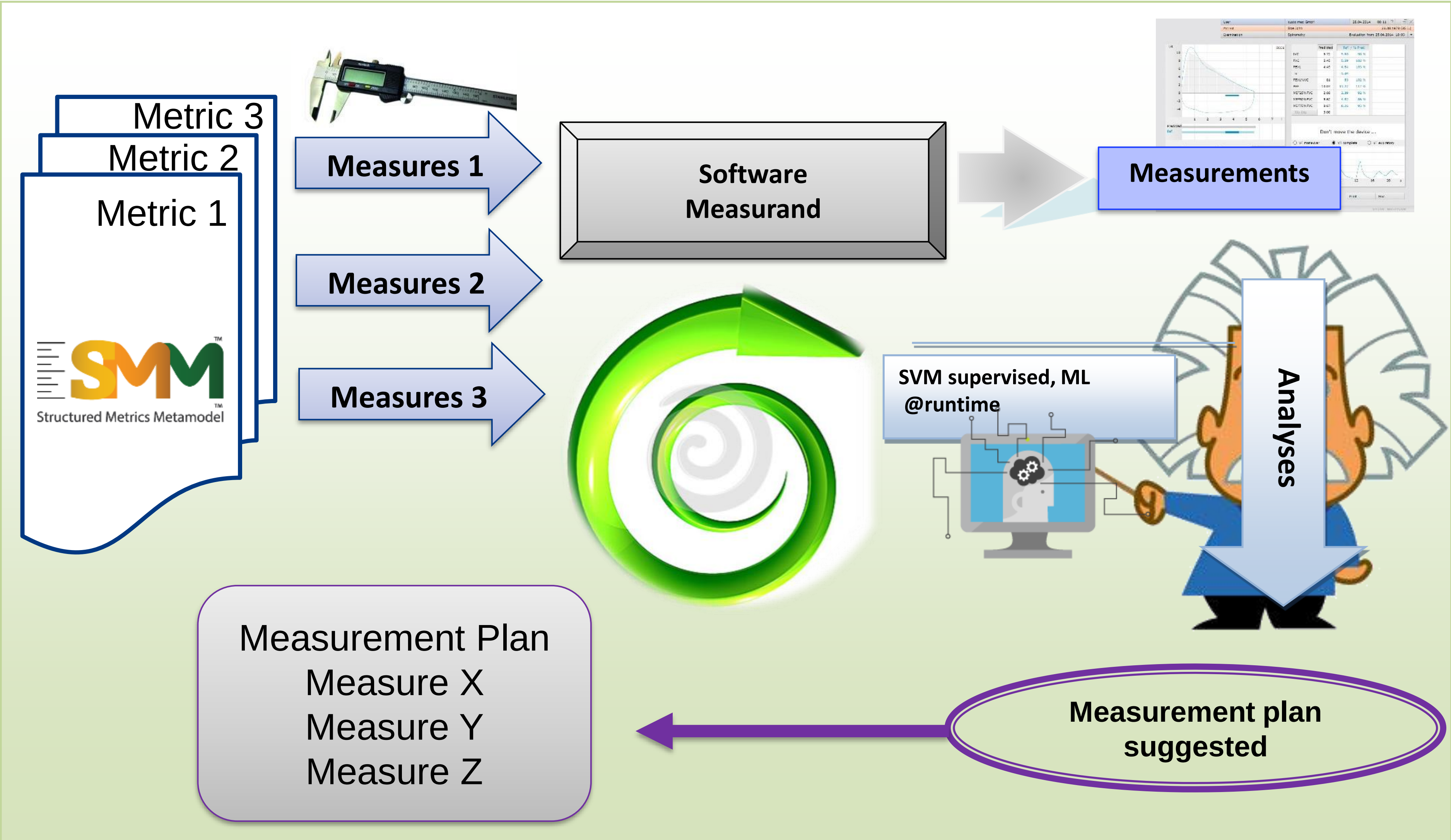
Motivations & Objectives

Increase the measurement process quality, efficiency and reliability

- ➢Improve the metrics sustainability, use and interoperability
- ➢Reduce the software measurement management costs
- ➢Ensure an accurate evaluation continuously

- **Standard-driven Metrics**
  - Standard specification
  - SMM of OMG
- **Automated analysis**
  - Machine learning – measurements result classifications
  - @Runtime
- **Metrics suggestion**
  - Measurement plan flexible
  - Accurate measurement @Runtime
  - Wide coverage continuously

Our objectives



State of Art

- **Software evaluation model**
  - ISO/IEC 25000: 2011 as software quality models

  ➔the implementation and management is left to the charge of the project manager.
  - Tahir, T., Rasool, G., & Gencel, C. (2016). A systematic literature review on software measurement programs. *Information and Software Technology*, *73*, 101-121.

  ➔ Lack of measurement formal models

- **Software measurement knowledge**
  - Ramesh, M. R., & Reddy, C. S. (2016). Difficulties in Software Cost Estimation: A Survey. *International Journal of Scientific Engineering and Technology, Volume*, (5), 10-13.
  - Kitchenham, B. (2010). What's up with software metrics?–A preliminary mapping study. *Journal of systems and software*, *83*(1), 37-51.

  ➔Difficult to measure the cost of a measurement plan

- **Machine Learning approach for SW defect prediction**
  - P. Deep Singh and A. Chug, "Software defect prediction analysis using machine learning algorithms," *2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence*, Noida, 2017, pp. 775-781.

➔Not used for the suggestion

- **Prioritize security inspection and testing efforts**
  - Shin, Y., Meneely, A., Williams, L., & Osborne, J. A. (2011). Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities. *IEEE Transactions on Software Engineering*, *37*(6), 772-787.

➔Only use for security and static selection

… And many others works on software measurement

Measurement Definition

- **Measurable space [1]**
  - Element to be measured X
  - The set of properties measurable A
  - $(X, A) | A \in X$
- **Measure space**
  - Measurable space
  - Associated function
  - $(X, A, f) | A \in X, f: A \rightarrow Z$
- **Software metric**
  - Software property
  - Software scope
  - Software measure

[1] Yingxu Wang, "The measurement theory for software engineering," CCECE 2003 - Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No.03CH37436), 2003, pp. 1321-1324 vol.2.

Formal-driven Metrics Specification through Modeling

- **Structured Metric Meta-model (SMM)** [1]
  - OMG standard specification
  - Meta-model to formally specify
    - software measurement
    - Software metric
  - → Standard interchange format, measurement documentation

- **Modelio** [2] MODELIOSOFT
  - Open source modeling tool
  - Based on UML

[1] http://www.omg.org/spec/SMM/1.1/

[2] https://www.modelio.org/

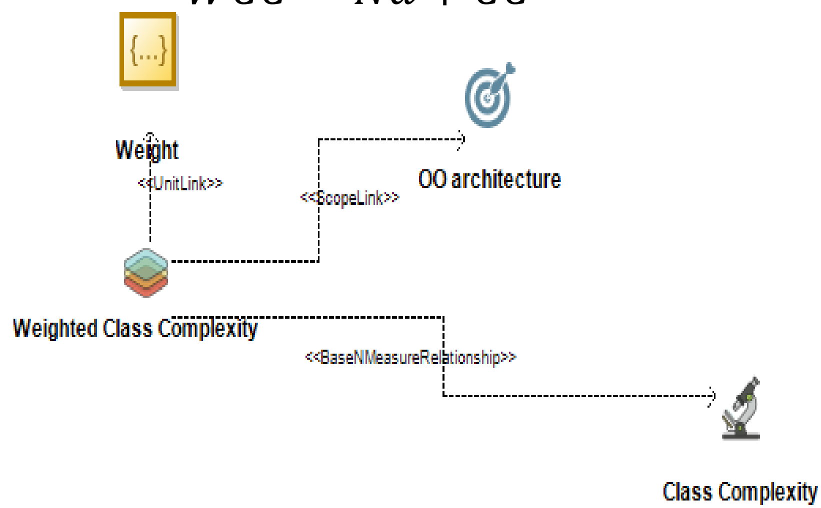
The formal Weighted Class Complexity Metric modeled in SMM with Modelio [1]

$$WCC = Na + \sum_{p=1}^{s} MCp$$

Na : number of attribute

MCp: method complexity of CC

$$WCC = Na + CC$$



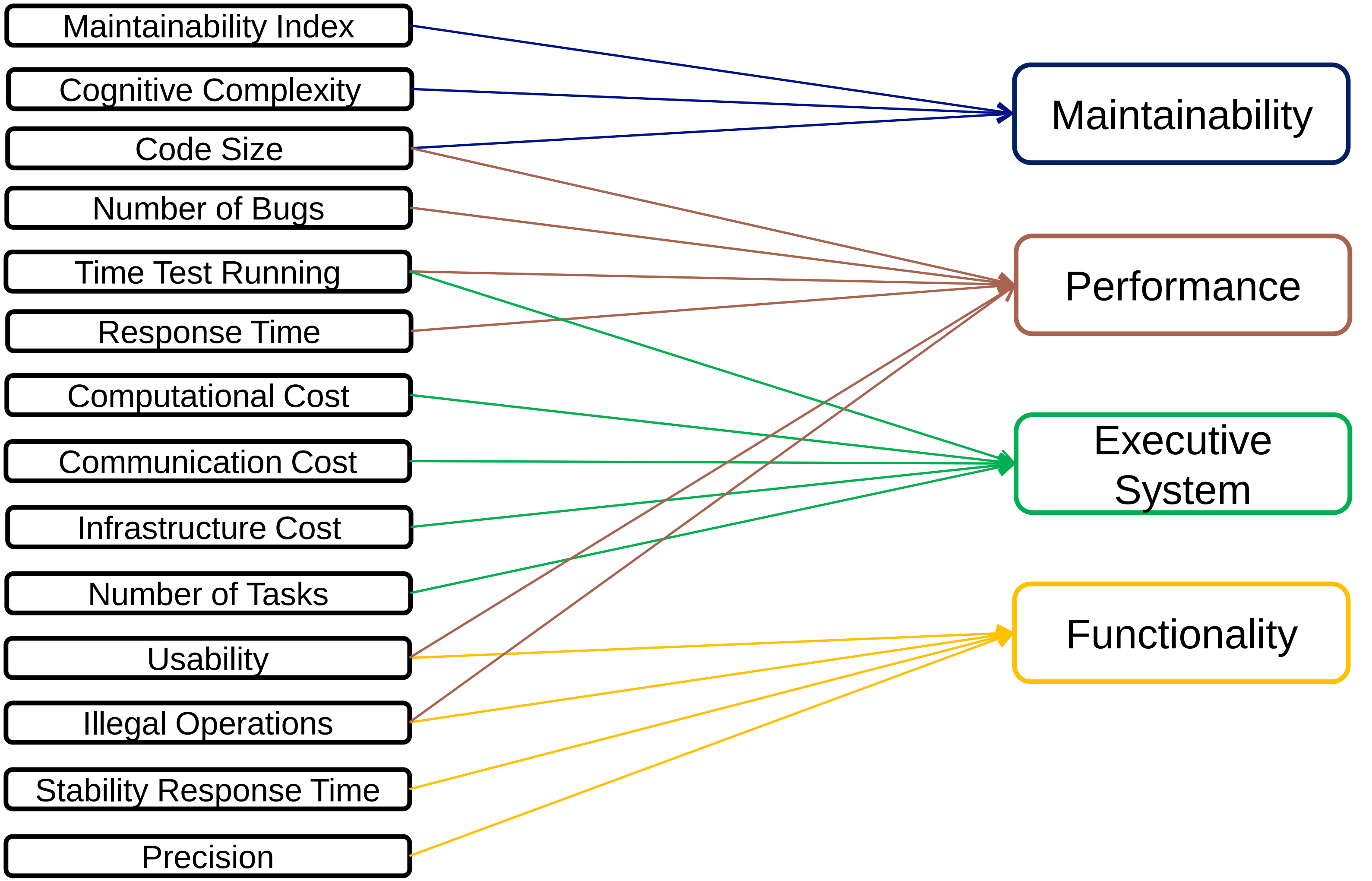
Our approach : A Learning Metrics Suggester

- **Our Set of Measures & properties associated**
  - Relevant indicators on software properties state
  - ISO/IEC standard 25000

- **Learning-based analysis**
  - Based on Machine Learning technique
  - Classification of measures result vectors
  - Mandatory Features Selection

- **Analysis-based measurement plan suggestion**
  - Metrics suggestion according to the analysis

Our Set of Measures as Software Properties Indicators



Learning-based analysis

- **Support Vector Machine (SVM)** [1]
  - Supervised learning technique
  - Classify a data sample
  - Through a linear hyperplane
  - From a training data set

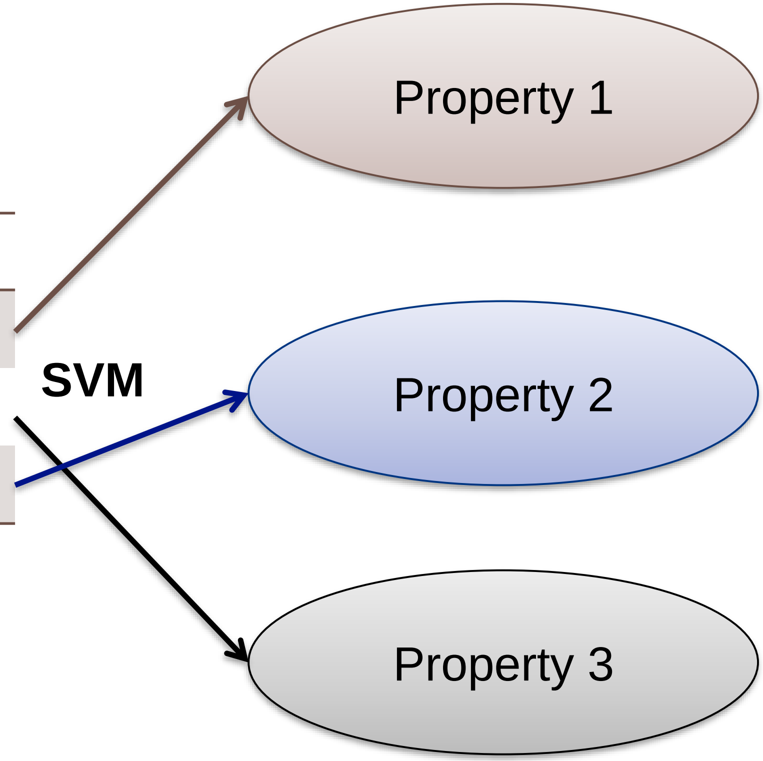
  ➔ Automating & Supports big data

➔ **Effective and expert-independent measurements analysis**

[1] Zhou X., Zhang X., Wang B., 2016, Online Support Vector Machine: A Survey. In: Kim J., Geem Z. (eds) Harmony Search Algorithm. Advances in Intelligent Systems and Computing, vol 382. Springer, Berlin, Heidelberg

Automated analysis : classification

Measure 1	Measure 2	Measure 3	Measure 4
0,45	0,98	0,5	0,69
0,12	0,70	0,81	0,56
0,90	0,23	0,75	0,15



Each line corresponds of a set of different measures result. Each line (set) is classify in one property by the SVM algorithm.

Dynamic Mandatory Features Selection

- **Subset of features with good predictive power**
  - RFE : Recursive Feature Elimination [1]
  - Based on the classification
  - ➔ Dynamic mandatory metrics selection

➔Keep the minimal overall information

[1] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in Science and Information Conference (SAI), 2014. IEEE, 2014, pp. 372–378.

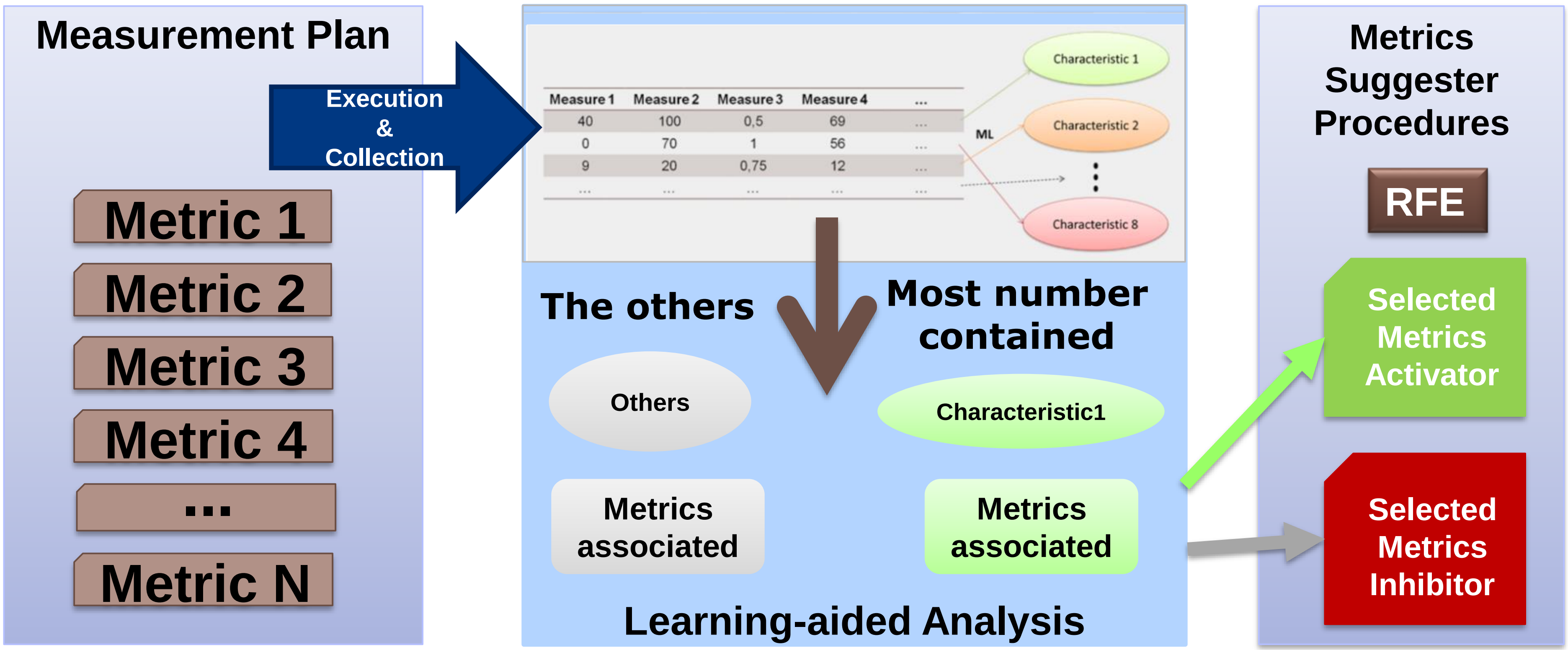
Measurement Plan Suggestion

- **Measurements analysis**
  - Our Set of Measures classification related to its values
  - Using of RFE algorithm to select mandatory features
  - ➔ Property of interest & Mandatory features

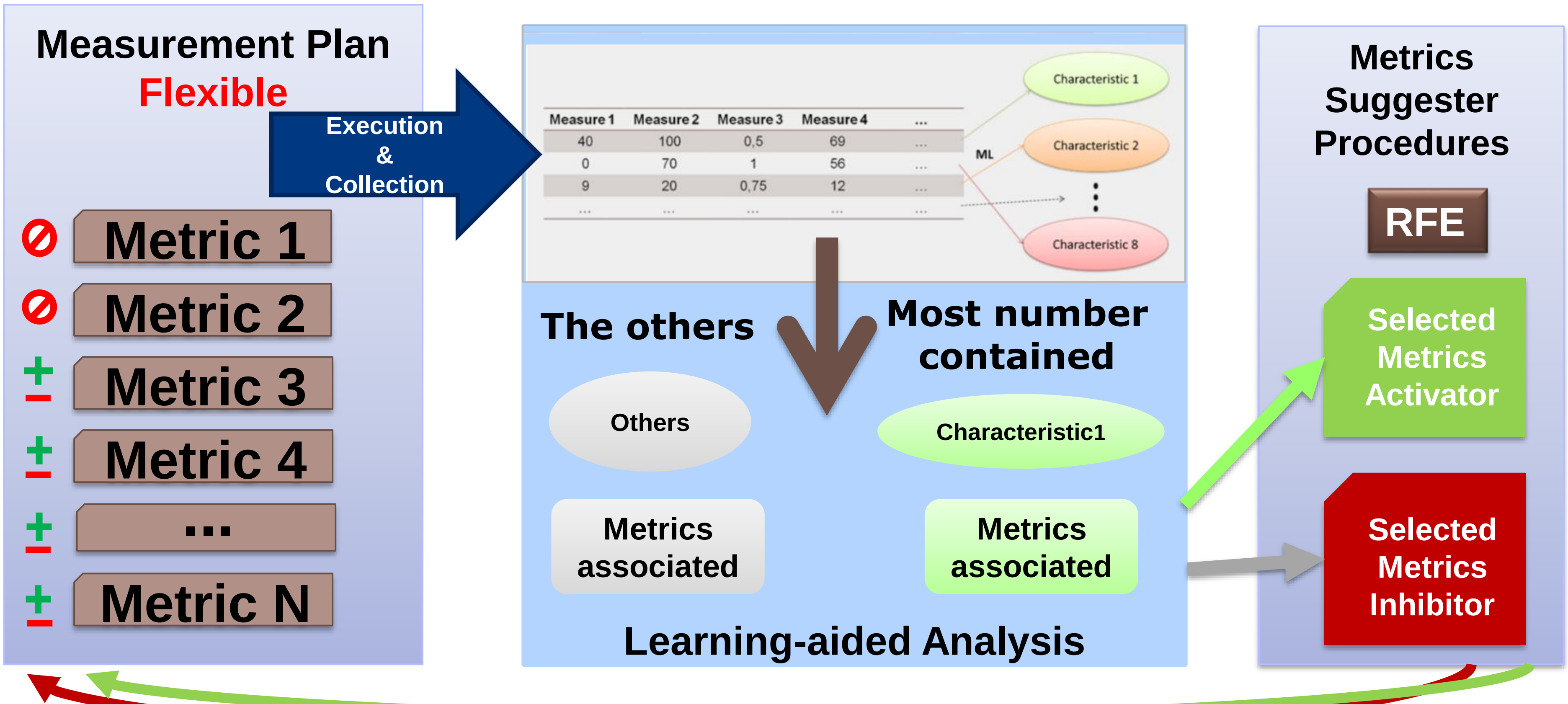
- **Efficient measures suggestion**
  - Set of metrics associated to the property of interest
  - ➔ Specific measurement suggestion
  - Selected mandatory features
  - ➔ keeping information on others property

➔ **Flexible measurement plan**

Metrics Suggester Framework



Metrics Suggester Framework



Metrics for flexible measurement plan

Experiments :

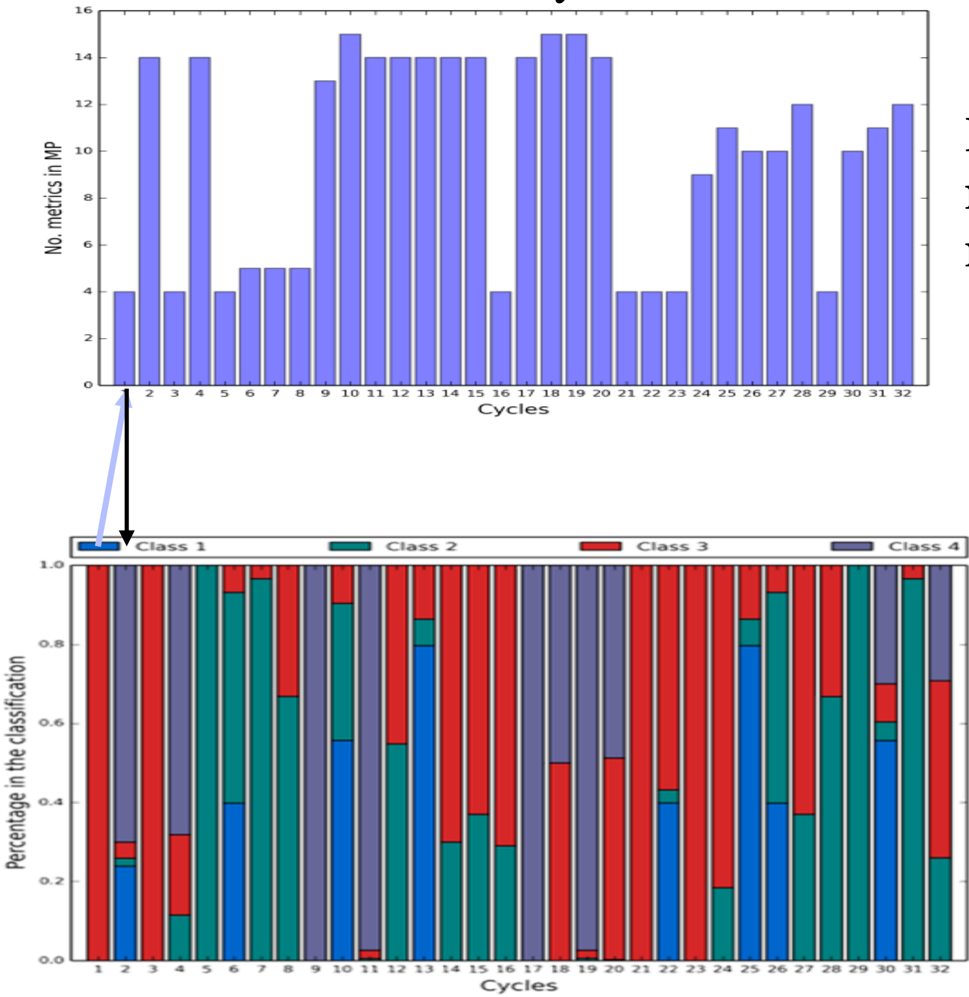
- **Case study**
  - European project MEASURE
  - OO platform in use
  - 15 metrics and 4 properties as measurement basis

- **16,000,000 vectors / 32 subsets of 500,000 vectors**
  - As measurements generation
  - 4 classes
  - 15 features

- **Scenarios**
  - 5 metrics as initial MP defined by the expert
    - Maintainability Index
    - Response Time
    - Running Time
    - Usability
    - Computational Cost
  - Class of interest : the one with the most predicted instances

Results

Number of metrics of the suggested plan in each cycle



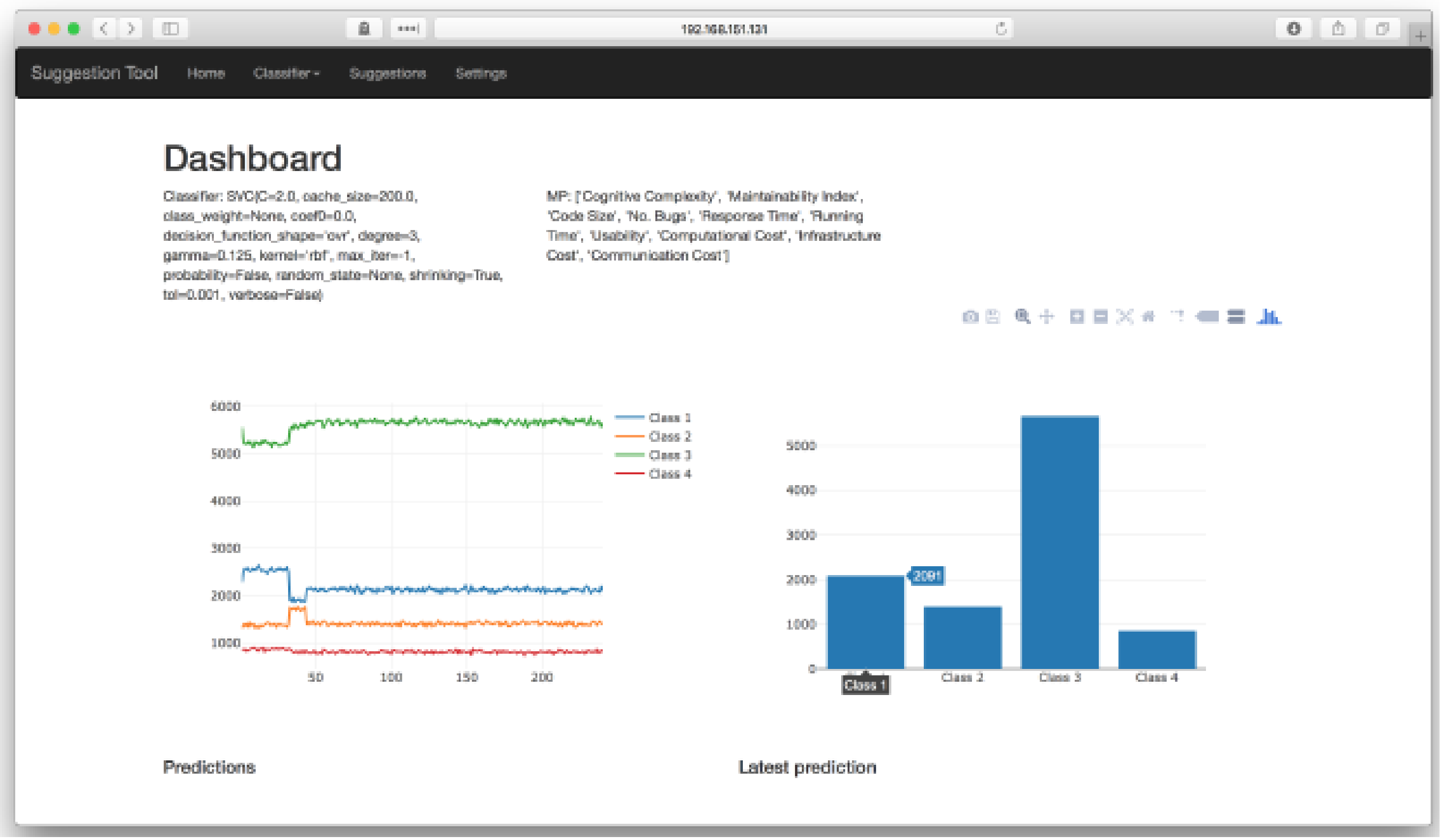
Each column represents the number of metrics suggested by the classification result.

Suggestion

Classification

Each column represents the classification result of the suggested metrics by the last classification. The classification is done between the four classes distinguished by colors.

Metrics Suggester Visualization



Conclusion & Perspectives

Conclusion:

- Expert-independent analysis
  - Automated & Big data analysis at runtime
- Flexible measurement process at runtime
- Safe measurement coverage continuously

- Open questions: Relevant/efficient suggestions? Time and cost reduced? Usefulness to industrial experts (compared to fixe MP)?

Perspectives :

- Industrial integration
- Measurement gathering intervals modified at runtime
  - Variable measurement cycles
- Exploration of other ML techniques
- Relevant justification

Bibliography

- Requirements and Evaluation (SQuaRE) -- System and software quality models, March, 2011

- I.H. Laradji et al., Software defect prediction using ensemble learning on selected features. Inf. and Soft. Technology, 58, 2015

- Manjula.C.M. Prasad, et al., A Study on Software Metrics based Software Defect Prediction using Data Mining and Machine Learning Techniques, Int. J. of Datab. Th. and App., 8(3), 2015

- J. Hentschel, A. Schmietendorf, and R. R. Dumke, "Big data benefits for the software measurement community," in 2016 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), Oct 2016, pp. 108–114.

- Y. Shin, A. Meneely, L. Williams, and J. A. Osborne, "Evaluating Complexity , Code Churn , and Developer Activity Metrics as Indicators of Software Vulnerabilities," vol. 37, no. 6, pp. 772–787, 2011.

Thank you for your attention