Reference:	**M3- SARch**
Topic:	**Software Architecture Reconstruction of Cyber-Physical Production Systems Engineering Software**
Lecture-Type:	Practical Course, Bakk-/Master Thesis
Start:	As soon as possible
End:	To be defined
Industry partner:	High-performance manufacturing and engineering (Upper Austria)
Contact:	Stefan Biffl (stefan.biffl@tuwien.ac.at)

Background

In *cyber-physical production systems* (CPPS) engineering, experts coming from different disciplines use domain-specific artifacts and tools to build multidisciplinary engineering environments. The software architecture of such complex environments tends to erode over time unless effort is spent to analyze the actual software architecture and to reduce non-essential complexity.

Software architecture acts as a shared mental model of a system expressed at a high level of abstraction. By leaving details aside, this model plays a key role as a bridge between requirements and implementation. According to Garlan [1], software architecture plays an important role in at least six aspects of software development: *understanding*, *reuse*, *construction*, *evolution*, *analysis*, and *management*. To understand, document, and maintain large applications, it is thus important to know their architecture. If the actual systems architecture is not clear, software architecture reconstruction becomes necessary.

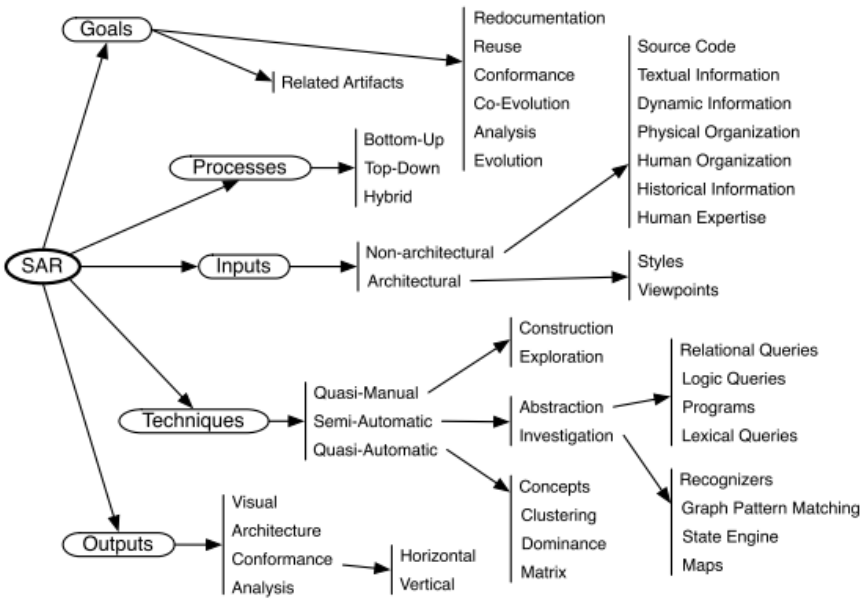

Figure 1: Categorization of Software Architecture Reconstruction approaches

Ducasse and Pollet [2] provide a process-oriented taxonomy on *Software Architecture Reconstruction* that evaluates available approaches and tools, and categorized these according to their *goals*, *processes*, *inputs*, *techniques* and *outputs* (see Figure 1).

The main ***goal of this work*** is to ***identify relevant architectural characteristics*** at an industrial company's software architecture (in an industry-oriented work). This precise architecture reconstruction will serve as the ***foundation for a migration project*** at the company to modernize their legacy system to a system fit for the needs of modern CPPS engineering.

Tasks

Based on the goal, specific tasks include:

- Basic literature review based on [2] and [3].
- Elicitation of software architecture and features from a legacy system and from domain experts
- Documentation of findings and challenges.
- Concepts for the mitigation of major software risks in the legacy system

Expertise and Skills Needed

Based on the selected tasks, required expertise and skills may vary. For this project, the following skill set is recommended:

- Software Engineering Skills
- Interest in software architecture
- Knowledge of or the willingness to learn Design Patterns and Clean Code/Architecture
- Willingness to explore and document software in an industry-oriented work
- Excellent social skills due to close teamwork with a company partner

References

[1] Garlan D.: "Software architecture: a roadmap". In Proceedings of ICSE, pages 91–101, 2000

[2] Ducasse S. and Pollet D.: "Software Architecture Reconstruction: A Process-Oriented Taxonomy", In: IEEE Transactions Software Engineering, pp.573-591, 2009

[3] van Deursen A., Hofmeister C., Koschke R., Moonen L. and Riva C.: "Symphony: view-driven software architecture reconstruction", In: Proceedings of the 4th Working IEEE/IFIP Conference on Software Architecture (WICSA 2004), pp. 122-132, 2004,