Software Technology for Self-Adaptive Systems

Emerging cyber-physical systems[1] increasingly change and improve the way we live, work, and communicate. It is the software of these systems that, to a large degree, controls their behavior and determines their usefulness to us. While software is a convenient way for humans to "tell" machines how to behave, the engineering of software-intensive systems is far from a mature discipline. A software system is developed to meet a set of requirements in a given environment that includes other (cyber-physical) systems. Any system must conform to its environment and requirements, so developers must design the system accordingly. The complex nature of both environment and requirements present a challenge to software developers. When both environment and requirements are uncertain at development time, the challenge becomes almost unmanageable.

Yet, in an ever-changing world, **uncertainty** at design time is rather the rule than the exception. The uncertainty is caused by inherent complexity and continuous change in the design preconditions such as requirements and environments of systems. Information can be incomplete, erroneous, or even unknown when decisions have to be made. Software engineers must cope with this uncertainty, and rely on non-deterministic information and probabilities. In the event that good enough design decisions were made at development time, the continuous change of environment conditions and requirements will most likely invalidate these later during development or after the system is deployed [1].

1 Synergy project's purpose and State-of-the-Art

1.1 Purpose

Uncertainty of system requirements and environment can cause expensive, underperforming, and misbehaving systems. **Self-Adaptive Systems** mitigate this risk [2] since they are able to change with the environment and requirements, and still remain reliable.

Uncertainty about requirements and environment can be handled if design decisions are delayed to or (continuously) reconsidered at runtime, when they are certain. So, in self-adaptive systems, some design activities become run-time activities. The responsibility for these activities shifts from software engineers to software. This may assure system requirements under uncertainty [3] but it blurs the traditional boundary between development-time and run-time of systems and invalidates classic software technology that supports development and operation [4]. Moreover, it adds a level of complexity, software engineers cannot just design software; they need to design "software-changing software".

To engineer software for self-adaptive systems is expensive and the quality of these systems vary because their development is more complex than traditional software since it cannot rely of well-established software technology [5][6]. The **purpose** of the suggested synergy project is to **increase the engineering efficiency of self-adaptive systems**. The development, maintenance and operation of such systems should be manageable and cost-efficient, and the systems should be safe, robust, high-performing, etc. To address these issues, we need to adapt proven software technology to the engineering of self-adaptive systems.

[1] A cyber-physical system is a system of collaborating computational elements sensing the environment and/or controlling physical entities, e.g., smart power grids, telecommunication systems with smart phones, and automated production cells.

1.2 State-of-the-art

There is a large body of research that suggest software engineering approaches to develop self-adaptive systems, including taxonomies, system level architecture patterns, development and operation processes and their interaction, etc.

Modern software development processes account for change to mitigate risks. For example, modern agile development interleaves requirement specification with software development in order to adapt to requirement. While agile development processes cope with changes in requirements, human developers bear the consequences, so these changes may still be expensive and result in low quality software.

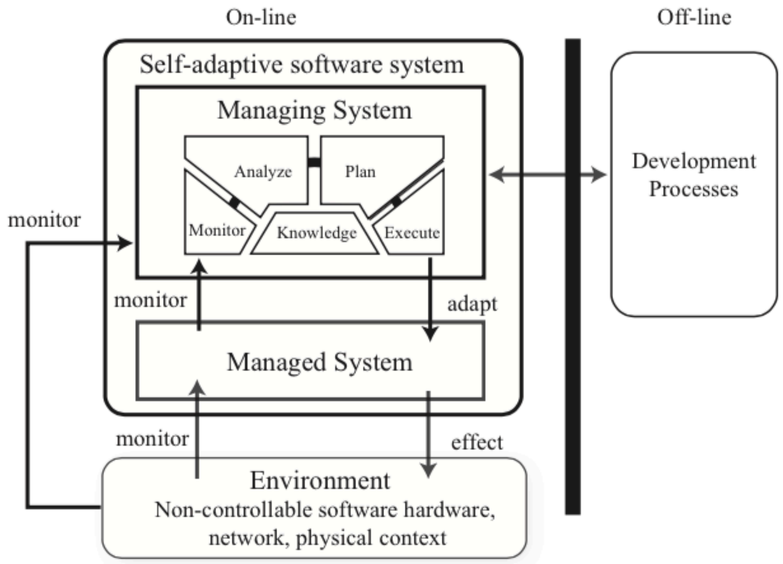

Figure 1: MAPE-K System Level Reference Architecture

The basic principles of self-adaptation were introduced in the pioneering work of Oreizy et al. [7]. IBM's MAPE-K model [8] that describes the primary functions to realize self-adaptation: Monitor, Analyze, Plan, Execute, and Knowledge has been particularly influencing (cf. Figure 1). System level architecture patterns such as MAPE-K, help system architects to structure their design. We use it as taxonomy to highlight similarities between the three application projects that study cyber-physical systems in our synergy project. Each self-adaptive system consists of two major components that interact with an ever-changing environment: the *Managed System*, implements the core functionality by monitoring and affecting its environment while the *Managing System* is responsible for the self-adaptation by monitoring both environment and Managed System and by adapting the latter. The Managing System consists of five interacting components that *monitor* and *analyze* change, and *plan* and *execute* adaptation. It gains *knowledge* about expected changes and successful adaptation that can be used for future analysis and planning activities. While this pattern provides a top-level system architecture, the design of the components and their interaction are still complex and unsupported.

We acknowledge the success of individual self-adaptive software systems, e.g., the seminal work on Rainbow [9] demonstrates how the MAPE-K model can be realized.

Adaptation typically employs multiple models that may reside at different tiers [10]. Control-theoretic approaches for resource-aware adaptation are applied to the software adaptation [11][12]. To provide assurance when dealing with uncertainties is a challenge. Quantitative run-time verification is a promising approach to provide such assurances [13]. Automatic software performance tuning (auto-tuning) addresses the need for performance-portability of software across various computing environments and contexts. Self-adapting algorithms for linear algebra packages are discussed in [14]. An approach to adaptive mapping of parallel programs to resources is presented in [15].

While our own efforts study the principles of self-adaptation [16][17], our main focus is to consolidating the know-how into reusable software technology assets for building self-adaptive systems, such as design and implementation patterns [18][19], algorithms and data structures [20][21], platforms and infrastructures [22][23], architectural patterns [24][25], and engineering processes [26].

2 Synergy project's core question

2.1 Core question for synergy

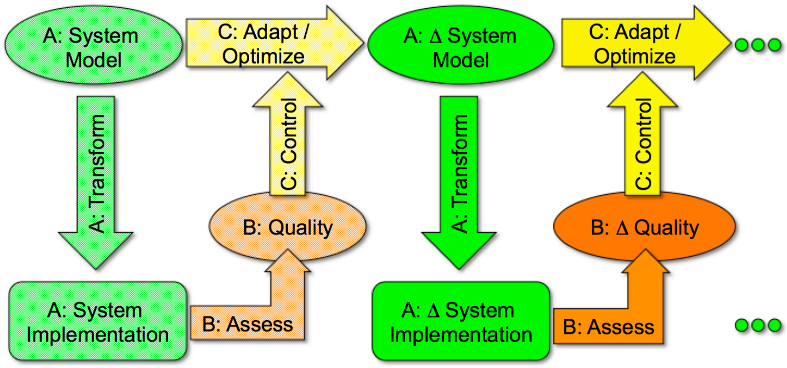

Figure 2: Essential TAPPAS supporting self-adaptation process.

To increase engineering efficiency when developing software to manage systems with uncertain requirements that run in uncertain environments, software engineers need reusable and consolidated know-how. Software technology provides such reusable knowledge for developing classic software systems, including **T**heories and Models, **A**rchitecture, **P**rocesses and methods, **P**latforms, frameworks, libraries, infrastructures, **A**lgorithms and data structures, and **S**ervices and Tools. We refer to these artifacts as **TAPPAS**. The state-of-the-art contributes such TAPPAS but there is a need to consolidate, validate, and integrate these into a sound engineering approach.

What is a consolidated, validated, and integrated engineering approach to self-adaptive systems that mitigates uncertainty?

An engineering approach for self-adaptive systems and TAPPAS that supports it should address and integrate the following aspects:

A. Models of system configurations that describe the current self-adaptive system and an automated process to derive the implementation of that system from its models.
B. Methods and infrastructures to quantify the quality of the current and the target system states.
C. Algorithms to optimize the current system to a target system that maximize quality.

These essential TAPPAS are integrated as depicted by Figure 2.

We aim to develop a consolidated, validated, and integrated engineering approach for self-adaptive systems in synergy with innovations in industry and academic research (vertical synergy, cf. Section 2.2) and the three application projects (horizontal synergy, cf. Section 2.3).

2.2 Relevance to industry partners (vertical synergy)

Together with our industry partners, we study three cyber-physical systems with uncertainties in the environment and requirements: smart power grids, telecommunication systems, and control cabinets of automated productions cells.

There are major uncertainties when engineering **smart power grids**, most prominently the varying production and the consumption of electricity. If engineers cannot manage this uncertainty, society risks energy losses during transmission, insufficient supply of electricity, or even power outages.

There is also uncertainty in the configurations of **telecommunication systems** for mobile and fixed network operators, including equipment (hardware), potentially not yet known services (software), and their manuals (documentation). This leads to the risk that not all system configurations can be quality assured, in turn, which may cause inconsistent and low quality configurations.

Finally there is uncertainty in requirements of **control cabinets** of automated productions cells, which results in cabinets being manufactured, i.e., assembled manually. This leads to the risk that the requested quantity of cabinets cannot be built in time at a reasonable price.

We conduct the **Synergy Project** based on three challenging **Application Projects** for self-adaptive systems:

A. **Model-Driven Engineering (MDE) applied to Smart Power Grids** with Vattenfall as customer and Hughes Power Systems and IBM as technology providers,
B. **Automated Quality assurance applied to Telecommunication Systems** of software and documentation with Ericsson as customer, Sigma Technology and Softwerk as technology providers, and
C. **Optimizations applied to Control Cabinets** for production automation with Yaskawa as customer and EPLAN as technology provider.

Researchers at Linnaeus University (LNU) will define and develop TAPPAS, in co-production with our industry partners. These are *Technology providers* who will use, test, benchmark, and validate the TAPPAS, and their *Customers* who own and directly benefit from self-adaptive systems in their application domains. The answer to our research question is of high relevance to both technology provides and customers.

Technology providers are interested in novel TAPPAS to use in their own services and products as well as their development. While we answer the research question, they can develop, use, and validate TAPPAS to build self-adaptive application systems. They can integrate them in their services and products for developing and maintaining self-adaptive systems. They can quantitatively evaluate the savings in development efficiency and gains in system quality, and relate these to potential income from customers.

Customers are interested in running self-adaptive systems (prototypes). While we answer the research question they can test the feasibility of innovative self-adaptive systems in their application context. Moreover, they can quantitatively evaluate the development costs and relate them to the business value of these systems.

The marketing of the technology providers as cutting edge innovators who participate in and drive scientific research projects in collaboration with academia is one of the key benefits for our partners beyond the results. The project setup enables such marketing towards a key customer within the project but dissemination of results and spreading the words about this research will reach other potential customers.

Moreover, the project trains promising young engineers and junior researchers to understand problems and find solutions in the respective application domains. This plays an important role in the technology providers' strategy for internal competence development and external recruitment.

2.3 Contribution of sub-projects to the core question (horizontal synergy)

While all three application projects develop self-adaptive systems there are differences in their contributions to TAPPAS. Table 1 shows the focus of the individual application projects. Notice that all projects either contribute to or require different TAPPAS. This both demands and enables synergy between the application projects.

The row Architecture (MAPE-K) of Table 1 shows how each individual application project will refine the MAPE-K reference architecture (cf. Figure 1). Table 2 shows the focus of each individual application project regarding the refinement and reuse of MAPE-K components. Note that all projects either refine or reuse different MAPE-K components, while the focus of the contribution is different for the individual projects.

TAPPAS	A – Model-Driven Eng.	B – Quality Assurance	C – Optimization
Theories, models	Use: structural and behavioral models	Define: quality models	Use: planning and optimization theories
Architecture (MAPE-K)	**Knowledge** accumulation	**Analyze** change	**Plan** adaptation
Processes and methods	**Define: MDE process**	Define: Quality assurance process	Use: existing design process of customer
Platforms, frameworks, infrastructures	Use: platform for monitoring environment	**Define: Structure, behavior and usage monitoring platform**	Use: existing platform of tech provider
Algorithms, data structures	Use: self-adaptation	Use: self-adaptation	Define: modified and new planning and optimization algorithms
Services and tools	Define: Modeling ad development tools	Define: Quality monitoring service	**Define: prototype for 3D placement of modules**

Table 1: TAPPAS Focus and Synergy of the individual Application Projects. "Define" means that an application project (columns) contributes to the respective TAPPAS kind (rows) while "Use" means that the application project requires this TAPPAS. Bold face denotes the main contributions.

MAPE-K	A – Model-Driven Eng.	B – Quality Assurance	C – Optimizations
Monitoring change	Define: probes and sensors	Define: static, behavioral and usage monitoring	Use: monitoring of the assembly robots and changes in 2D schematics of a cabinet
Analyzing change	Define: analysis of environment and system	**Define: quantifiable quality state and trend**	Define: analyze changes in 2D schematics of cabinets
Plan adaptation	Use: re-configuration plan	Use: configuration specifications and plans	**Define: construct a 3D layout of a cabinet from previously unseen 2D schematics**
Execute adaptation	Define: safe and reliable reconfigurations	Use: configuration tools	Use: adapt to changes in 2D schematics, previous plans to new requirements
Knowledge accumulation	**Define: Models at runtime**	Define: Quality norms and distributions	Use: knowledge about quality of layouts

Table 2: MAPE-K Focus and Synergy of the individual application projects. Bold face denotes the main contributions.

3 Objectives

We distinguish between research objectives (RO) and industry innovation objectives (IO) for the synergy project and each of the application projects.

3.1 Objectives of the synergy project

The research objective of the synergy project is to derive a consolidated, validated, and integrated engineering approach for self-adaptive systems including approaches to:

RO-S1: Model a system configuration and automatically derive the actual system.
RO-S2: Quantify the current and target quality of such systems (adaptation goal).
RO-S3: Optimize a current system to a target system with maximum quality (adaptation).

These objectives are achieved by generalizing the results from application projects A, B, and C, respectively. Together, these contribute to our industry innovation objective on synergy level:

IO-S1: TAPPAS that synergistically support our engineering approach and that software engineers can reuse efficiently and effectively.

3.2 Objectives of the application projects

3.2.1 MDE applied to Smart power grids

RO-A1: Investigate how MDE techniques can be used as the basis for the engineering of self-adaptive systems.

RO-A2: Provide a framework based on MDE that allow engineers to consider adaptation issues and reuse proven solutions while designing, implementing, validating and operating software systems.

IO-A1: Identify relevant models in smart power grids including load status and prediction models, network configuration models, and software configuration models.

IO-A2: Develop generator technology that transforms load model instances into configurations.

IO-A3: Develop monitoring technology that derives model instances and deploys actual network and software configurations.

3.2.2 Automated Quality Assurance applied to Telecommunication systems

RO-B1: Investigate to what extent a combination of static and dynamic quality assessments can be used to assure quality in self-adaptive systems of software and technical documentation.

RO-B2: Investigate how to define flexible and meaningful quality models.

IO-B1: Investigate how quality assessment translates to new system representations (code, documentation).

IO-B2: Improve the definition, use, and reuse and adaptation of quality models.

IO-B3: Improve the impact of quality assessment on systems engineering and operations.

3.2.3 Search based optimizations applied to Control cabinets

RO-C1: Improve methods for search-based optimization of multi-target constrained optimization problems.

IO-C1: Develop a method and a tool that based on a 2D schematic of a control cabinet determines the optimal location of components in the 3D cabinet model.

3.3 Relationship between the application and synergy project objectives

Table 1, Table 2, and the research and industry innovation objectives of each application project show the focus on each of the essential TAPPAS. Application project A focuses on self-adaptive system modeling (related to RO-S1), B on the quantification of quality of such a system (related to RO-S2), which is necessary for C that focus on search-based optimization (related to RO-S3).

4 Organization, strategy and risk analysis

4.1 Organization of the Synergy and Application Projects

4.1.1 Management, Steering committee, Administration

Figure 3 shows the organizational chart of the synergy project, the three application projects, and their management. An Executive Board manages the project. It consists of the PIs of the synergy project (Danny Weyns, Welf Löwe, Jesper Anderson,) and the three PIs of the application projects (Mauro Caporuscio – A, Morgan Ericsson – B, Sabri Pllana – C).

Managing director Welf **Löwe** coordinates the operations of the project. He is (was) the PI of successful research projects with a turnover of altogether ~25 MSEK (including two Knowledge Foundation funded projects). He is founder, research coordinator, and chairman of the board of the Information Engineering Center[2] (IEC), a cluster with a turnover of altogether ~6.7 MSEK in the

[2] http://lnu.se/iec

last 4 years. He is also the co-founder and former CEO of ARiSA AB (today Softwerk AB), a spin-off of a Knowledge Foundation project, with an annul turnover of ~5 MSEK.

Danny **Weyns** coordinates a Marie Curie project on the foundations of decentralized self-adaptation, a Swedish Research Council project on assurances in decentralized self-adaptive systems, and a Vinnova project on self-adaptive systems utilizing these to support everyday life of elderly people. Jesper **Andersson** is the head of department of Computer Science. The Ministry of Education and Research appointed him to the LNU organizational committee. Morgan **Ericsson** managed the Software Engineering and Management program at the University of Gothenburg with 300 international students. His role included operational responsibility for the students, their environment, and an education manager, and an improvement budget of ~650KSEK over two years. He has also managed large project courses (150-200 students per course) with industry involvement (typically 2-5 companies per course). Sabri **Pllana** coordinated the FP7 EU project PEPPHER (with a total funding of more than 30 million SEK) that comprised nine organizations from six European countries.

Besides the PIs with extensive experience in academic management, we integrate a younger colleague, Mauro **Caporuscio**, into the project management. He has lots of research project experience, though not yet in management positions. If needed, he can refer to senior researchers in his application projects; Arianit **Kurti** (PI of a Knowledge Foundation Expert competence for innovation project and managing director of the Interactive Institute Swedish ICT in Norrköping) and Welf Löwe.

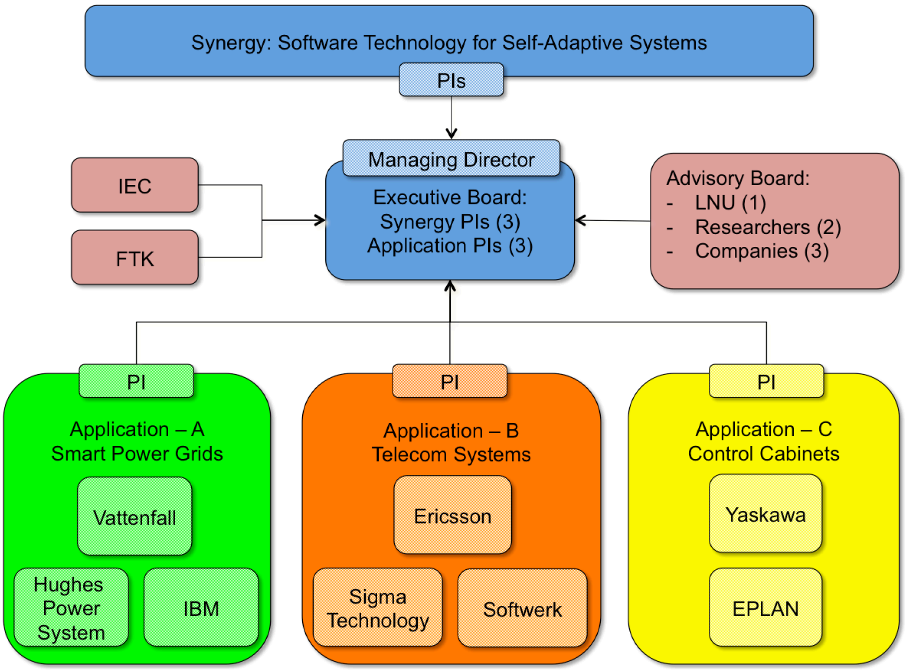

Figure 3: Overall Project Organization.

The Executive Board is supported by an Advisory Board that consists of the Dean of the Faculty of Technology (FTK) at LNU, two external inter-nationally renowned researchers in self-adaptive systems, and three industry representatives.

FTK supports the project administration, aiding with economy administration. IEC supports the project's cooperation with industry and society beyond the industry application project partners. IEC is a cluster of academia, industry and society focused on information and communication technologies (ICT). Almost 200 member companies and organizations are currently involved in IEC and three PIs of the synergy project lead IEC Interest Groups.

4.1.2 Scientific leadership

The synergy project is collaboratively led by three PIs who are also responsible for the integration of the scientific results of the application projects.

Professor Danny **Weyns**' main research interest include formalisms and design models to realize and assure self-adaptation for different quality goals, particularly in how to exploit the design models at runtime to provide evidence for the required adaptation goals and handle changing requirements. He is very well cited (h-index 30) and published in top venues, incl. IEEE TSE, ACM TOSEM and TAAS, and co-edited about 15 volumes in the domain of adaptive systems. He is a member of the editorial board of the Journal of Autonomous Agents and Multi-agent systems.

Professor Welf **Löwe**'s main research interests include software and information analysis, composition and optimization. He is well-cited (h-index 20) and published in top venues, incl. IEEE TSE, IST, and PC. He initiated the Nordic Conference on Web Services conference series (ESOCC today) and is member of the IFIP Working Group 2.4 Software Implementation Technology.

Associate Professor Jesper **Andersson**'s main research interest include the engineering of self-adaptive software systems, more specifically theory, methods, and techniques for engineering such systems, in particular for strategic reuse. He is well-cited (h-index 19) and published in top venues, incl. ACM TAAS and J. Syst. Software. He co-edited a special issue on state-of-the-art and research challenges in self-adaptive systems.

A PI scientifically leads each application projects; the scientific competences of the PIs in their respective application domain are highlighted in Sections 5.1.3, 5.2.3, and 5.3.3, respectively.

4.2 Organization and scientific management of the research environment

The Department of Computer Science at LNU conducts research on the engineering of self-adaptive software systems. The group participates in several national and international research projects, in close collaboration with industrial partners. It is a principal partner in IEC.

The research group consists of 3 full professors, 11 assistant and associate professors, and 2 post-docs. The group has a background in compilers and optimization, software engineering and software architecture, artificial intelligence and multi-agent systems, and information and software visualization. It is internationally renowned and an active member in the international research community in self-adaptive software systems. In recent years members of the group have organized several scientific events and participated actively in others. The group regularly publishes its results in renowned international journals and at other high-quality venues. The international research network is large and includes collaboration with SEI/CMU, Hasso-Plattner Institut, Karlsruhe Institute of Technology, RUG-Gronningen, University of L'Aquila, and York University.

4.3 Synergy application's alignment with the HEI's strategy

4.3.1 Alignment with the Strategy of the Department of Computer Science 2015-2020

The proposed synergy project will empower the research and research-driven education at the department of Computer Science. In 2013, the researchers at our department decided to center their research on the common topic of software technology for self-adaptive systems. In recent years we received considerable funding for projects in this field. A synergy project would strengthen the collaboration among our projects and researchers. It would also pave way for becoming a nationally and internationally recognized research center. Moreover, it serves as a template for future projects, helping to shape and develop IEC towards an Institute for Applied Research & Innovation. Finally, it is a stepping-stone to a successful Knowledge Foundation profile proposal planned for 2018/19.

4.3.2 Alignment with the Strategy of LNU 2015-2020

LNU's strategy 2015—2020[3] aims to "A creative knowledge environment in the spirit of Linnaeus" which is characterized by challenging educational programs, prominent research, societal driving force, and global values.

Challenging educational programs: Driven by the needs of the local (IT) industry, LNU seeks the rights to offer a Master of Science in Engineering (Civilingenjör) program. In 2019/20, the Computer Science department plans to submit a proposal for such a program with a focus on software technology in general and on the engineering of self-adaptive systems in particular.

Prominent research: The synergy project contributes to the state-of-the-art in self-adaptive software systems. Additionally, the application projects contribute to the state-of-the-art in their respective application domain. High-quality results will be published in renowned journals (TSE, TOSEM, TAAS) and conference series (ICSE, ASE, SEAMS) in the fields of Software Engineering and Computer Science. TAPPAS constitute reusable assets to keep the engineering complexity of self-adaptive systems manageable as validated by demonstrator systems. It will contribute to a renewal of the research focus in self-adaptive software systems as it brings it closer to innovation. At the same time, the demonstrators are sources of inspiration for new important research questions, which will be addressed in future research projects (e.g., on EU level within the ECHORD++ program[4]).

Societal driving force: The IT sector is import to the Linnaeus region (Kronoberg and Kalmar regions). Växjö municipality has the second highest IT job density[5] in Sweden and the IT industry will announce about 500 open IT positions over the next five years. Demanding and attractive IT education programs together with a nationally and internationally recognized research center will keep and attract the best (from students to senior experts) to this region.

Global value: The domain of self-adaptive systems is recognized as a top research priority in the EU Framework Program for Research and Innovation "Horizon 2020"[6]. At a recent EU event on "Cyber-Physical Systems: Uplifting Europe's innovation capacity" Khalil Rouhana[7], the director for "Components & Systems" at DG CONNECT of the European Commission, foresaw a high potential for economic growth and employment capacities of 250,000 jobs within this domain.

4.4 Co-financing

The synergy and the application projects are funded by LNU (8.5%) and by our innovation partners from industry (49.3%). We apply for funding to the Knowledge Foundation to cover the remaining 42,2%.

LNU covers the overhead costs of its employees in the project that are not covered by the Knowledge Foundation. Co-funding by our industry partners is detailed in the budget motivation sections of the application projects, 5.1.3, 5.2.3, and 5.3.3, respectively.

4.5 Risk analysis

Risk	Probability/ Impact	Risk management action
Technology provider	Low/	Redundancy of technology providers in A and B. Use open

[3] http://lnu.se/polopoly_fs/1.111406!A_journey_into_the_future_2015-2020.pdf
[4] http://echord.eu
[5] Fraction of population in IT related jobs.
[6] http://ec.europa.eu/research/horizon2020/index_en.cfm
[7] http://ec.europa.eu/digital-agenda/en/khalil-rouhana

drops out	Medium	source technology. Recruit alternative technology providers using IEC.
Customer drops out	Low/ Medium	Use remaining customers to demo/apply the results. Recruit alternative customer of technology provider.
Researcher drops out	Medium/ Low	Redundancy of researchers in synergy and application projects.
Key person of management drops out	Medium/ Low	Redundancy of management resources in synergy and application projects.
Application project fails in finding novel TAPPAS/MAPE-K.	Low/ Low	Use state-of-the-art technology for the respective essential TAPPAS and MAPE-K components.

Table 3: Risk Analysis and Management

Risks are continuously assessed and, if needed, management actions are taken, cf. Table 3. Risks that are specific to the application projects are analyzed in Sections 5.1.3, 5.2.3, and 5.3.3.

5 Application Projects

5.1 Project A – Model-driven Engineering applied to Smart Power Grids (MoDES)

5.1.1 Purpose, State-of-the-Art, and Research Question

Purpose: Smart environments are envisioned as *"small worlds where different kinds of smart device are continuously working to make inhabitants' lives more comfortable"* [A1]. They represent a promising application domain that creates new appealing opportunities for sustainable development and business. An example is a *Smart Power Grid* (SPG) that aims at producing and managing sustainable electric energy provision. Energy production and provision is becoming more and more complex, and it is dynamically changing. In fact, nowadays energy providers can configure power grids to achieve different capacities, depending on both *consumer* demands and *producer* supplies. To guarantee the provision of stable and sustainable energy in an unknown, ever-changing environment, the *SPG* should self-adapt to functional and extra-functional uncertainty inherent to the operation context. The *SPG* must be able to (1) *monitor* the environment, e.g., the consumption/production, (2) *analyze* the data and detect anomalies, e.g., energy loss, (3) *plan* the needed reconfiguration actions, e.g., switching distribution path, and (4) *execute* the reconfiguration actions at run time.

The lack of engineering best practices for developing and operating *SPGs* makes it difficult for the industry to fully benefit from such a rich business sector. In fact, full comprehension of continuously self-adapting systems combined with the uncertainty of the context in which they operate cannot be anticipated at development time. This means that engineering such systems with incomplete knowledge of available resources is challenging. The incomplete functional knowledge inherent to self-adaptive systems (i) and the incomplete knowledge about changing non-functional requirements (ii) raise a set of challenges. This requires changing radically the way software is designed, implemented, validated, and operated.

An *SPG* should support both *adaptive* and *evolutionary* behavior, and on-line and off-line activities. Concerning *adaptation*, the *SPG* should be able to monitor itself and the environment, detect significant changes, decide how to react, and enact such decisions. *SPG adaptation* deals with run-time (Running System) uncertainty: (*i*) at *Managed System* level, because of emerging internal situations, e.g., energy delivery disruptions, interruptions and fluctuation, and (*ii*) at *Environment* level, because of external situations, e.g., dynamic changes in the environmental conditions. On the other hand, *SPG* should support its *evolution* towards satisfying new requirements to meet the changing needs of stakeholders, and even the changing stakeholders. *SPG evolution* deals with design-time (Development Process) uncertainties: (*i*) at business level, because of emerging new markets opportunity, (*ii*) at design and implementation level, because of emerging new requirements, and (*iii*) at infrastructure level, because of emerging new technologies.

This drives towards the need of defining *Processes and Methods* (cf. Table 1, TAPPAS) allowing engineers to consider all the above challenges throughout all phases of the software life cycle, from requirement elicitation to operation. Specifically, the MoDES project aims at providing a Development Framework (DF) leveraging on the Model-Driven Engineering (MDE) paradigm that refers to the systematic use of models as primary artifacts for the engineering of systems. Therefore, models will be exploited at all stages of the software system development process, from requirement elicitation to design and development, to run time. Departing from traditional software engineering best practices, where models are used off-line for reasoning about the "future" behavior of the system under development, models are now kept alive at run time, and used to continuously assess the "current" behavior of the system under execution.

To this end, the MoDES project will investigate *Theories and Models* (cf. Table 1, TAPPAS) to achieve run-time analysis: from run-time monitoring to (incremental) model checking. Because of that and since systems reconfigure according to changes in the execution context, their run-time analysis must be carried out on the fly in a timely and efficient manner. It is well known that the computational complexity of analysis techniques is a key challenge.

State-of-the-Art: Model-Driven Engineering (MDE) is a software development paradigm that aims to use (software) models as the main development artifact instead of code [A2, A3, A4]. One approach to reach this goal is to use a specification language adapted to a specific problem domain (domain-specific language, DSL) and then to apply automated transformations to generate models of lower abstraction levels, which finally get transformed into code. These techniques also allow specifying Quality of Service (QoS) and Quality of Experience (QoE) constraints, generating deployment descriptors, and managing the deployment and wiring of services during runtime.

Exploiting models at run time is not new per se, and different approaches have been proposed to extend MDE applicability to the run time [A5]. In fact, MDE capabilities (e.g., model transformation and code generation) are considered viable means to enable on-line system monitoring, model analysis and adaptation [A6]. For instance, in [A7] model-based analysis is exploited to manage QoS level of software systems at run time, whereas in [A9] architectural models are used to achieve both run-time adaptation and evolution.

In addition, a scenario-based approach captures software architecture evolution [A10]. This approach provides a framework for modeling various types of architectural views, such as re-engineering, analyzing and comparing architectures [A10]. In fact, the proposed model for software architecture evolvability analysis enables better understanding of systems' ability to easily accommodate changes [A11]. Several sub-characteristics of software evolvability and corresponding measuring attributes are identified (such as analyzability, architectural integrity, changeability, extensibility) [A11]. A comparison framework based on the concept of software open points provides a flexible approach to decentralized software evolution [A12]. Whereas, an open architecture approach and its model provides the software system with the ability to evolve over time in terms of new services, devices and subsystems attached to it, in order to adapt to different usage contexts in heterogeneous environments [A13].

Research Question: How can we design and implement a dependable and self-adaptive SPG? By generalizing from this industrial challenge, the MoDES project contributes to the Synergy project by addressing the following research question:

How can we develop systems that can benefit from smart environments by being at the same time self-adaptive and dependable?

5.1.2 Objectives, Expected Results and Benefits

The main objective with respect to the synergy project is twofold. Regarding TAPPAS, MoDES will contribute to *Processes and Methods* by defining Model-Driven Engineering methods for self-adaptive systems. Regarding the MAPE-K reference architecture, MoDES will contribute to *Knowledge* accumulation by defining models at run-time.

MoDES aims to provide a DF that employs the MDE paradigm and exploits models at all stages of the development process, from *requirement elicitation* to *run-time*.

RO1: Investigate how Model-Driven Engineering techniques can be used as the basis for the engineering of self-adaptive systems.

1. To what extent can run-time models be automatically derived from design-time models?
2. To what extent can run-time models be timely fed and analyzed on-line?

RO2: Provide a framework based on MDE allowing engineers to consider adaptation issues and reuse proven solutions while designing, implementing, validating and operating software systems.

1. To what extent are traditional software engineering development processes/practices well suited for self-adaptive systems?
2. To what extent can we assure/enforce dependability properties?

At industry level, the practical benefit of MoDES is the development of an SPG demonstrator system that self-adapts according to the environmental dynamics.

IO1: Identify relevant models in smart power grids including load status and prediction models, network configuration models, software configuration models.

1. How can we make use of prediction models that includes load status for network and software configuration?
2. How can Vattenfall create business value by using such prediction models?

IO2: Develop generator technology that transforms load model instances into configurations. How to transform

1. How can we transform load model instances into configuration?
2. How can Hughes Power Systems create business value by using these load model instances?

IO3: Develop monitoring and control technology that derives model instances and deploys actual network and software configurations.

1. How can we develop a monitoring technology that helps derive load (load model instances)?
2. How can we develop control technology changes to the network configuration that is optimum for the actual load?
3. How can IBM make use of the monitoring technology to extend its own Smarter Planet Platform?

To find engineers with competence in electrical engineering and computer science is not easy. The management of competence development at Hughes Power Systems will benefit from this interdisciplinary project.

Hughes Power Systems will be able to demonstrate and quantify power savings to Vattenfall as a potential customer. To win Vattenfall as a customer and demonstrate high research and innovation competence from collaboration with LNU and IBM will open up further business opportunities for Hughes Power System.

Last but not least, to develop technology that saves energy resources, to quantify the savings, and to eventually implement the technology for a customer such as Vattenfall with its impact on the energy market in Sweden and Europe is of highest societal benefit.

5.1.3 Implementation

Project Organization: *Researcher*: LNU provides research expertise in Distributed systems, Model-Driven Software Engineering, Software architecture, Self-adaptive systems.

Technology Provider: Hughes Power Systems (HPS) specializes in development and manufacture of voltage switchgear and transformer products for the electrical utility industry. IBM Sweden Smarter Planet (IBM) is technology provider for environmental monitoring.

Customer: Vattenfall Services Nordic (VS) is electric utility that aims to dynamically reconfigure power grid infrastructures to guarantee the provision of stable and sustainable energy in an ever-changing environment.

Workflow and Project Plan: The work is divided into work packages (WP) specific to the research objectives of this project and synergy packages (SP) for achieving the objects of the synergy project. The WPs and SPs are described in detail below and scheduled according to Table 1 of the document "Overall Project Plan". There are two additional general activities that stretch over the whole project period. The first aims to ensure compliance of project activities with the project plan and mitigating the risk by proactive monitoring all the activities. To this end, MoDES will adopt an iterative and incremental approach, and will rely on frequent meetings and workshops, within a single research activity (and among different activities) to present consolidated results, discuss open issues and possible alternatives, and find shared solutions. The second aims to distributing MoDES results to the widest possible audience. MoDES will target the usual academic channels: papers will be submitted to the most important international conferences and to high-reputation international journals.

WP-A1 [responsible VS, involved all]: To understand how MoDES should deal with *SPGs* and to elicit first-class concepts to be accounted within the project implementation, during this activity we will conduct an accurate *Domain Analysis* and *Requirements Elicitation*, as well as we will build the MoDES Conceptual Model.

WP-A2 [LNU, all]: During the Research activity we will investigate 1) model-based *Processes and Methods* for dealing with the development of self-adaptive systems and 2) model-based techniques for operating "knowledge accumulation".

WP-A3 [IBM and HPS, all]: This activity will develop 1) the *Smart Power Grid* demonstrator specified in **WP-A1**, and 2) a set of tools implementing methods and techniques defined in **WP-A2**. *Smart Power Grid* prototype and *Toolset* will integrate optimization heuristics for finding the minimum loss network configuration for a given network load situation. They will be released at Month 18.

WP-A4 [HPS, all]: This activity aims at integrating the different tools developed in **WP-A3** into the Development Framework. Central to this activity is experimental testing in a real world test facility for smart power grids. Candidate final version of Development Framework will be released at Month 42.

SP-A1 [LNU, all]: The first iteration is dedicated to validate the tools developed in **WP-A3,** and to identify TAPPAS candidates to be released at Month 24. The second iteration will be dedicated to validate the rebuilt TAPPAS identified during the first iteration.

Members of the team have the required expertise in the project-relevant domains: Model Driven Engineering (Mauro Caporuscio, Welf Löwe), Self-Adaptive Software systems (Mauro Caporuscio, Arianit Kurti, Bahtijar Vogel), Smart Grids (Welf Löwe, Rune Gustavsson), Power Grids (Hans Ottosson, Kent Olssons, Mauro Caporuscio, Bahtijar Vogel), Smarter Planet (Lars Wiigh, Mauro Caporuscio, Bahtijar Vogel).

Person	Affiliation	Percentage
Mauro Caporuscio (PI)	Linnaeus University (org. number 202100-6271)	30%
Bahtijar Vogel	Linnaeus University (202100-6271)	30%
Arianit Kurti	Linnaeus University (202100-6271)	10%
Welf Löwe	Linnaeus University (202100-6271)	10%
Rune Gustavsson	Linnaeus University (202100-6271)	10%
Hans Ottosson	Hughes Power System (556926-5068)	47%
Kent Olssons	Vattenfall Services Nordic AB (556417-0859)	10%
Lars Wiigh	IBM Sweden AB (556026-6883)	10%

Table 4: Members of the project team A.

Budget Motivation: Network configuration optimization techniques (finally integrated in a Development Framework) are tested in a real world test facility outside Växjö in Väckelsång, Tingsryd. It allows integrating multiple Small Scale Embedded Production Units in various configurations of micro grids. The facility consists of a 12kV indoor power grid including cable subnetworks of different capacity and substations 12-0.4kV, and is powered by a gas turbine power plant.

The in-kind contribution of Hughes Power Systems will include: (1) operation of and access to the power grid system during the whole project duration, and (2) time for R&D personnel involved in the project. The in-kind contribution of Vattenfall includes time for R&D personnel involved in the project. The in-kind contribution of IBM will include: (1) Access to a set of IBM tools during the whole project duration, (2) training session to learn about the IBM Smarter Planet, and (3) time for R&D personnel involved in the project. Through the Smarter Planet initiative we get access to a set of IBM tools, including: Intelligent Operation Center for Big-data aggregation and Analysis, Watson for Cognitive Computing, the BlueMix IDE.

	Item	Contribution [SEK]
Hughes Power System	Operational costs of gas turbine	1 500 000
	Test facility deprecation (over 4 years)	125 000
	R&D time (800h/year, 800 SEK/h)	2 560 000
	Hughes Power System total	**4 185 000**
Vattenfall Services	R&D time (94h/year, 800 SEK/h)	300 000
	Vattenfall Services total	**300 000**
IBM Sweden	Operational costs (4 years)	1 200 000
	Training and tutorials	250 000
	R&D time (172 h/year, 800 SEK/h)	550 000
	IBM total	**2 000 000**
	TOTAL Contribution	**6 485 000**

Table 5: Contributions of industrial partners of project A.

Hughes Power Systems will provide MoDES with access to their generator and power grid infrastructures using so-called small scale embedded production units in the test facility in Väckelsång. To assess research/industrial innovation relevance and the business potential, MoDES will be validated through a Smart Power Grid use case experiments. We need a set of computing devices (sensors and mobile devices) for integration to the test facility. MoDES will deploy a set of sensors to monitor both the Smart Power Grid infrastructure and the environment. Raw data collected by sensors will be stored, aggregated, and analyzed on-the-fly, and finally presented to a big screen on the *Monitoring Center* and to *Expert on the field* (i.e., on their mobile devices). Therefore, MoDES will integrate the *Smarter Planet* platform with a set of tools (software/hardware) specifically addressing the Smart Power Grids issues. For that reason, we need to purchase equipment such as: Microcontrollers (e.g., Raspberry, Arduino 3, Galileo) and Sensors Packs, Server for Big-Data Processing, and Mobile computing devices for on-the-field monitoring.

For adapting and optimizing the grid network configuration under operation, we secured the services of Professor Leslie Falkingham. He will assist us in questions regarding vacuum switchgear, arc physics, and discharges in vacuum, which are central to actually deploying a reconfiguration, with 50 h/year.

Travel cost for traveling to conferences and project meetings amount to 350 KSEK. Additional equipment worth 100 KSEK, including sensors, computing units, etc., is needed to conduct the experiments in the test bed. For the dissemination in Open Access Journals, for organizing events, seminar and tutorials, we apply for additional 50 KSEK.

Risk Analysis: *Technological risks*. Advances in theoretical work can be slower than the actual evolution of computing and communication technologies. This problem is typical for engineering research in rapidly evolving areas where the need for sound design methods is challenged by continuous changes in technology. Risks can be mitigated by adopting an iterative and incremental approach for the development of MoDES, and by a thorough evaluation of the obtained results against the advances in the technological context at the end of each iteration. To this extent, MoDES exploits two iteration loops, and will release an open-source solution at the end of each iteration.

Reality risks. Often research solutions outperform the state-of-the-art in limited experimental settings, but do not scale to realistic contexts. MoDES aims at delivering research that is realistically usable and can be convincingly demonstrated. MoDES will produce research open-source prototypes (not products) demonstrating the innovation, soundness, and feasibility of the proposed solutions. MoDES aims at mitigating this risks by exploiting the iterative and incremental approach, and evaluating, at the end of each iteration, the productions against the *Smart Power Grid*. This evaluation allows for the identification of further requirements for the next phase, and the identification of corrective actions to keep the project on the track of “realistic” results and solutions.

5.2 Project B – Automated Quality Assurance applied to Telecommunication systems and documentations

The Darwin Information Typing Architecture (DITA) is an XML-based, end-to-end architecture for authoring, producing, and delivering technical information[8]. This architecture consists of a set of design principles for creating “information-typed” components at a topic level and for using these components in delivery modes such as online help and product support portals on the Web. The concept of a topic in DITA is very similar to that of an object or a class, and all the benefits of object-orientation apply, such as encapsulation, polymorphism, and message passing. There is a great interest for DITA in the information engineering industry due the increased possibility for reuse, which can lower cost and effort, and easier translation across different mediums.

From a quality assurance perspective, DITA’s similarity to object-orientation strengthens the idea that software quality assessment principles can be applied to information as well [B21]. A manual becomes a collection of (specialized) topics, and quality indicators such as size, coupling, and depth of inheritance become trivial to translate. The quality of a documentation (or a program) can be considered the sum of the quality of the topics (or classes).

[8] OASIS standard: http://docs.oasis-open.org/dita/v1.2/os/spec/DITA1.2-spec.html

However the topics (or program components) are used in many different and changing configurations; the configuration space is immense and dynamically changing. It is not feasible to check, test and assure quality of all static configurations of topics (and program components).

Do the configurations of topics and components impact the quality of a system of hardware, software and documentation components? If the quality in-use is considered, i.e., how well the system (hardware, software, documentation) solves the problem at hand, then it certainly does. In these cases, we need to automate quality assessment of system configurations. Once this is possible, we can integrate it in self-adaptive systems that reconfigure themselves to check if their quality meets the requirements and even use quality as an optimization goal when we choose the best-fit reconfiguration.

5.2.1 Purpose, State-of-the-Art, and Research Question

Purpose: The DITA case may seem as an engineering task, but as our discussion shows, it is actually a self-adaptive system if we consider it during use. The user controls the adaptation based on changing needs. From a research perspective, systems (including technical documentation) that can adapt to usage scenarios present an interesting challenge. Currently, we can statically assess quality of both software and technical information, and we can assess traditional technical documentation in use. This project aims to extend state-of-the-art to assess software in use and investigate how well the combination of static and dynamic quality assessment works when systems can adapt; is it feasible to statically assess all possible combinations or can we statically assess the components (classes, topics) and dynamically assess the configuration?

Quality models are a means to connect quality goals with quality indicators and metrics observable in the objects under assessment. They take a lot of time and effort to define, and there are many sources of uncertainty, such as how well a specific metric represents a quality indicator, what effect metrics aggregation has on the quality score, and so on. On an abstract level, a quality model consists of metrics that aggregate and integrate to form new metrics, and so on. This "metric orientation" can be considered similar to topic and object orientation. Given the uncertainty it might make sense to consider a quality model itself as a self-adaptive system, where we can learn and adjust the quality model to the actual use (and desired quality goals). The uncertainty not only makes the quality models difficult to define, it also makes the result difficult to communicate and understand. Visualizations do help, but our experience show that these are not sufficient; there is a need to reduce the uncertainty in the models.

From the perspective of TAPPAS and the Synergy project, we aim to provide a quality assurance process that includes the definition of quality models, and reusable tools to assess and communicate quality that can be used for self-adaptive systems.

State-of-the-Art: This project seeks to advance the state-of-the-art of automated quality assessment of systems that consist of software and documentation, specifically self-adaptive systems. Automated quality assessment involves metrics that measure aspects of the system under assessment and models that relate these measures to determine how well quality goals are fulfilled. There exist a range of frameworks to describe software and documentation quality, from research efforts, e.g., [B11, B17, B12] to international standards such as ISO/IEC 25010:2011. These effort focus on what quality is, but generally not how to measure it automatically. Automated measure of software quality, metrics, have been studied by, e.g., [B8, B4, B1], and validated by, e.g., [B2, B6]. We have extended these efforts to the information quality of technical documentation [B18–21].[9]

[9] Knowledge Foundation project Effective and Efficient Information Quality Assessment, 3/2012—2/2015.

There are, to our knowledge no specific quality models or quality assessment tools for self-adaptive systems. Monitoring is an important part of such systems, so there exist a large body of work on how to effectively and efficiently monitor run-time aspects of them, e.g., [B14]. [B16] presents a framework for properties that are important to adaptation and how these map to qualities.

The effects of uncertainty on quality assessment has been noted and studied to some extent. [B10] finds that the same metrics are defined and measured in significantly different ways by different tools and [B9] finds that different quality models with the same purpose (e.g., to measure maintainability) result in significantly different assessments. [B5] shows that the data normalization used within a model will affect the quality assessment. It is well known that certain metrics, such as lines of code, have a non-normal distribution [B7] and there are efforts that rely on equality indices [B15] or non-parametric statistics [B13] to avoid this problem. It is not clear how the accuracy of the assessment is affected by these efforts.

Research Question: We seek to expand the state-of-the-art of automated software and information quality assessment, specifically applied to self-adaptive systems.

How can we automatically assess quality of systems that are dynamically configured and reconfigured from software and documentation components?

5.2.2 Objectives, Expected Results and Benefits

The overall objective of the project is to contribute to the efficient use and definition of effective quality models. We want quality models to be sound and accurate, as well as easy to use and understand. Metrics-based quality assessment is used in industry, but metrics are often used in isolation or as part of existing quality models.

We define general research and industry objectives from the project purpose. These objectives are divided into research questions and industry questions. The former are mainly of benefit to the research community, while the latter is beneficial to our industry partners, and the community.

RO1: Investigate to what extent can a combination of static and dynamic quality assessment be used to assess self-adaptive software systems and technical information.

1. To what extent can static quality assessment be used to assess self-adaptive software systems and technical information?
2. To what extent can dynamic quality assessment replace the need for static assessment of configurations?

RO2: Investigate how to define flexible but sound quality models

1. How general are quality models? To what extent must the use change before the quality model needs to change to reflect this?
3. How robust are quality models? To what extent can the structure change before the precision changes?
4. What difficulties exist when we combine static and dynamic metrics?
5. How can a probability-based quality model be used to assess and communicate quality?

IO1: Investigate how quality assessment translates to new system representations (code, documentation).

1. Provide strong support for DITA.
2. How well do our current quality assessment tools and quality model apply to DITA?
3. What quality factors support portability to DITA (and further on)?

IO2: Improve the definition, use, re-use, and adaptation of quality models.

1. How can the initial quality model definition become more efficient?

2. How can we adapt the quality model to new quality goals?

IO-B4: IO3: Improve the impact of quality assessment on systems engineering and operations.

1. How can we trace a quality issue to its root cause?
2. How can we support measures removing the root cause (and not the indicators)?
3. How can we evaluate the benefits of improved quality?

The project aims to improve how we reason about and assess software and information quality. Even a small contribution to the state-of-the-art of these fields in industry will have an impact on the involved companies and society at large. We have already observed a shift in how our partners approach and think about information quality during our previous collaboration, and we expect this process to continue. Our two previous Knowledge Foundation projects have resulted in tools and methods that are used by industry, and we also expect this project to continue that tradition.

The results and any improvements to existing tools and methods will improve the competiveness of the two technology provides, Softwerk and Sigma Technology. Ericsson will benefit from better quality, and improved efficiency and effectiveness of the quality assessments, with respect to both software and systems.

5.2.3 Implementation

Project Organization: The project involves four partners: *researchers* from LNU, *technology providers* Sigma Technology and Softwerk, and Ericsson as a *customer*. LNU conducts research on software and information quality assessment and provides expertise on processes and methods. Sigma Technology provides content management tools and domain expertise on documentation and software. Softwerk provide quality monitoring tools and domain expertise on software and information quality assessment in practice. Ericsson is a product owner of software and documentation, and provides domain expertise on telecom systems.

Workflow and Project Plan: The project uses a spiral model, where theoretical insights drive method and tool development. Each innovation cycle starts from a set of ideas and validated solutions. It results in a set of issues, questions, and a plan on how to address these. The cycles follow the traditional scientific method, where a model is created and used to formulate hypotheses. The hypotheses are validated through experiments, which in turn are used to refine the model. Each cycle pushes the knowledge boundary towards the envisioned goals. Based on experience from previous projects, we expect these cycles to be short, at most a month per cycle.

We take an agile approach to project management. The overall project structure follows the four phases of the synergy project. Each phase contains a number of activities that mix theory and practice, and are expected to deliver extensions to our current tool and method as well as theoretical contributions. The only dependencies that exist in the project are between phases; so all activities within a phase can run in parallel. Each activity has an estimated effort relative to the total effort of the phase (in percent).

The first phase focuses on how our current quality assessment approach can be adapted to self-adaptive systems of software and technical documentation. We rely on two cases relevant to the telecommunications domain, a self-adaptive software system and a documentation that follows the DITA structure. The goal of the first phase is to achieve the same performance as our methods and tools have on traditional software systems and technical documentation. Phase 1 is divided into activities that investigate the following:

WP-B1 Static quality assessment for reconfigurable systems (20%).

WP-B2 Dynamic quality assessment/metrics for software systems (20%).

WP-B3 Quality models that combine static and dynamic metrics (25%).

WP-B4 How should the assessment results be communicated and interpreted (30%)?

WP-B5 Tool-support for DITA (5%).

There are no explicit deliverables from the activities; they are designed to bring value to all project partners. For example, Activity WP-B1 will investigate how to statically assess multiple models and configurations of software and documentation, and provide a quality assessment for all of the possible configurations. Once we have developed a method to achieve this, the tool will be extended and the results and experience will be disseminated. So, the deliverables from WP-B1 is an updated tool and a conference publication. In general, we expect that activities result in workshop and conference publications/presentations, and phases in journal publications.

The second phase focuses on the integration of our quality assessment tools and methods into the tools and workflow from the other two projects and to define quality models that are suitable for their needs. We also expect this joint tool, Synergy TAPPAS, to be deployed at our industry partners and used in production to some extent.

SP-B1 Integration of our monitoring framework into Synergy TAPPAS (60%).

SP-B2 Define quality models for Projects A and C (40%).

The third phase focuses on advancing the state-of-the-art beyond our current approach and tools. The quality model and tools will be reformulated to use probabilities rather than normalized values that are aggregated using weighted averages. We will apply this approach to the two cases from phase one, as well as the systems investigated in our previous project to determine how effective this approach is compared the previous state-of-the-art. We will use all three projects to assess the approach from Phase 1 and to evaluate the joint system from Phase 2

WP-B6 Reformulate our quality model to probabilities (35%).

WP-B7 How should the assessment results be communicated and interpreted under the new model (15%)?

WP-B8 Extends tools (10%).

WP-B9 How effective is the new approach compared to the previous in terms of accuracy (20%)?

WP-B10 How effective is the new approach compared to the previous in terms of communication and interpretation (20%)?

The fourth phase focuses on improvements to the joint tool based on results from Phase 3. If the probability-based approach is successful, it will be integrated into the tool parallel to the approach from Phase 1.

SP-B3 Integration of the extended monitoring framework into Synergy TAPPAS and refinement based on experience from Phase 3 (100%).

The WPs in phases 1 and 3 provide TAPPAS. There is some overlap between what specific TAPPAS each activity focus on, but main focus is as follows: WP-B1, WP-B2, and WP-B8 focus on platforms for quality analysis and monitoring, WP-B4 and WP-B6 focus on theories and models, and WP-B4 and WP-B7 focus on the quality assurance process. Given that WP-B6 and WP-B7 propose a new formalism, we added WP-B9 and WP-B10 to explicitly evaluate this approach. In all other cases, the evaluation is part of the activity and the following SP-activities.

The research staff consists of Morgan Ericsson, principle investigator, Welf Löwe, Anna Wingkvist, and Maria Ulan. Professor Welf Löwe founded the research group on software and information quality at LNU, and Anna Wingkvist and Morgan Ericsson helped further information quality as a research field. The PhD student Maria Ulan is a recent addition to the team to help strengthen the research on foundations of quality models.

Sigma Technologies is represented by Johan Thornadtsson and Robin Long, who are both domain experts on information engineering and content-management systems. The industry representatives will be part of the operation of the project and serve as access points to the respective companies. Softwerk is represented by Rüdiger Lincke, who is an expert on software quality assessment. Ericsson is represented by Frans Frejdestedt, Richard Lundberg and Anna-Karin Hulth, who are domain experts on their respective systems. Anna-Karin Hulth is employed by Sigma Technology, but on contract at Ericsson and will serve as an expert on its systems.

The respective degree of activities of the research team is given in the Table 6.

Person	Affiliation	Percentage
Morgan Ericsson (PI)	Linnaeus University (org. number 202100-6271)	25%
Welf Löwe	Linnaeus University (202100-6271)	10%
Anna Wingkvist	Linnaeus University (202100-6271)	10%
Maria Ulan	Linnaeus University (202100-6271)	80%
Johan Thornadtsson	Sigma Technology (556000-9366)	12%
Robin Long	Sigma Technology (556000-9366)	12%
Rüdiger Lincke	Softwerk (556714-0503)	5%
Frans Frejdestedt	Ericsson (556016-0680)	6%
Anna-Karin Hulth	Ericsson (556016-0680)	6%
Richard Lundberg	Ericsson (556016-0680)	6%

Table 6: Members of the project team B.

Based on our previous experience with similar projects, we find it best if all partners are involved in all activities to some extent. All activities will be lead by the principle investigator (Morgan Ericsson) and tasks within them will be allocated to the other researchers or industry partners. The research group will have weekly meetings and the industry partners will be part of the weekly meeting (at least) once per month.

Ericsson and Sigma Technologies are product owners and domain experts. In phases 1 and 3, they are responsible for access to software and documentation as well as to continue to work on the definition of quality. They should also contribute to the discussion on how to communicate and interpret the results, as well as conduct internal studies on how these tools can be used. Softwerk provide access to the quality assessment tools and support. In Phase 1 Softwerk will also help to implement support for DITA in their tools. In Phase 2, Ericsson and Sigma Technology will deploy Synergy TAPPAS in a limited production environment. They will also provide advice to the other industry partners when we define quality models for their systems. In Phase 3, Ericsson and Sigma Technology are also expected to use Synergy TAPPAS and help evaluate it.

All researchers will be involved in all activities, but Welf Löwe will mainly work on WP-B1, WP-B2, WP-B3, SP-B1, WP-B8, and SP-B3. Anna Wingkvist will mainly work on WP-B4, SP-B2, WP-B9, and WP-B10. Maria Ulan will mainly work on WP-B3, SP-B1, WP-B6, WP-B7, WP-B9, WP-B10, and SP-B1.

Budget Motivation: All industry partners cover their salary costs for participating in the activities. Additionally, Softwerk and Sigma Technologies provide access to their Quality Monitor and their content-management system (DocFactory), respectively.

	Item	Contribution [SEK]
Sigma Technology	Johan (salary 200 h/year, 800 SEK/h)	640 000
	Robin (200 h/year, 500 SEK/h)	400 000
	DocFactory (licenses)	1 920 000
	Sigma Technology total	**2 960 000**
Softwerk	Rüdiger (salary, 78h/year, 800 SEK/h)	250 000
	Quality Monitor (license, maintenance)	275 000

	Softwerk total	**525 000**
Ericsson	Frans (100 h/year, 800 SEK/h)	320 000
	Richard (100 h/year, 800 SEK/h)	320 000
	Anna-Karin (100 h/year, 500 SEK/h)	200 000
	Ericsson total	520 000
	TOTAL Contribution	**4 325 000**

Table 7: Contributions of industrial partners of project B.

We budget for 2-3 (international) conference trips per year, and bi-monthly (national) visits with our partners.

Risk Analysis: We have an established collaboration with a majority of the project partners from our previous Knowledge foundation project on Effective and Efficient Information Quality Assessment and other activities such as organizing Teknikinformation i Centrum 2014[10], educational collaborations, etc. So, we have already established that we can work together as a team. However, in a large organization such as Ericsson, it can be challenging to establish internal communication between the software development, technical documentation, and operations departments. We want this communication to get a holistic view on system quality and mitigate this risk by disseminating results from our previous project to other units within Ericsson, e.g., those responsible software development and software quality prior to the synergy project. We involved these units in our project meetings and presented results locally at the respective units as well as globally at the Ericsson developer conference. This way, we created awareness of a holistic quality assessment approach and assured us of their willingness to support our approach. Moreover, we are in contact with researchers from Gothenburg University and Ericsson SW Research to synchronize our efforts with their research activities on software quality assurance.

5.3 Project C – Optimizations applied to Automated Assembly of Customized Control Cabinets

5.3.1 Purpose, State-of-the-Art, and Research Question

Purpose: The vision of the project is an adaptive and automated assembly of customized control cabinets using industrial robots. Many companies currently assemble cabinets manually, from reading and interpreting the CAD-drawing to the actual assembly of modules and cables. This process could be largely automatized with adaptive robots and software systems. Ideally, the robot will be able to identify the type of cabinet that is to be assembled, and by taking as input assembly instructions in a robot-understandable format, it will adapt to the task of assembling that cabinet.

Uncertainty in the type of a customized control cabinet that a customer may order causes common engineering approaches for production of large quantities of units to not be applicable and as a result, the planning and production process is often done manually. Our project will handle this issue by automatic transformation of the two-dimensional CAD-drawing of a customized control cabinet into a form that is amenable to automatic assembly, resulting in a flexible robot system.

We will focus on one of the technical issues that must be solved towards the realization of adaptive automatic assembly of electrical cabinets. Electrical CAD-drawings for customized cabinets are made in a CAD program (such as EPLAN). In EPLAN engineers can create 2D schematics of which components should be in the cabinet and how they should be connected to other components, manually place components in a virtual 3D model of the cabinet and automatically generate wire routes. It is also possible to export the project to assembly instructions that can be used by industrial robots to assemble the cabinet. The problem is that based on a 2D schematic the engineer has to manually place each component in the virtual 3D model of the cabinet. To this end, the

[10] http://www.boti.se/anmalan-till-tic-2014/

purpose of the project is to develop a method and a tool that based on a 2D schematic of a control cabinet, determines the optimal location of components in the 3D cabinet model.

The conceptual contribution will be advancements in the field of automated optimization in general. We will investigate general optimization techniques, e.g., non-deterministic and search- and learning-based optimization, with multiple optimization goals and constraints. We will parallelize these techniques to achieve the necessary optimization efficiency and consequently reduce the time-to-solution. The results will likely be applicable to other application domains as well including those addressed in projects A and B.

State-of-the-Art: To find an optimal placement of components within a cabinet is a type of problem commonly referred to as a *Constraint Satisfaction Problem* (CSP) [C1]. A CSP is defined by a set of variables (3D position of each individual module), a set of constraints (minimum distance between modules, port locations, etc.), and a solution (configuration plan of the cabinet). A possible solution shall not violate any constraints, and the optimal solution is the best solution in the set of possible solutions according to the optimization parameters (spacing, heat dissipation, wiring length, etc.). Constraint based placement of objects have previously been used in various application domains. Sanchez et al. proposed a constraint-based system for placing furniture that optimized the solution using a genetic algorithm [C2]. Osman, Georgy and Ibrahim used a similar approach where they developed a system that optimized the layout planning of a construction site using *genetic algorithms* [C3]. Yu et al. developed a system for optimizing the layout of furniture in an indoor virtual scene using a constraint-based approach with optimization using *simulated annealing* [C4]. Fisher et al. used a slightly different approach for optimizing furniture layout based on probabilistic model learning with *Bayesian networks* [C5]. Petcu and Faltings propose a method for distributing constraint optimization using a multi-agent based approach [C6]. Togelius et al. proposed a multi-objective evolutionary optimization method for generating maps for strategy games [C7, C8].

In contrast to the related work that focuses on 3D arrangement of furniture, we will address the optimized automatic 3D arrangement of control cabinet components considering not only physical dimensions but also electrical properties of components. To reduce the time needed for optimizing a layout configuration of a control cabinet we will study the application of parallel processing to constraint optimization on modern many-core architectures (such as the Intel Xeon Phi co-processor [C9]).

Our project will advance the field of constraint-based optimization using different optimization approaches and implementation techniques. The results will likely by applicable to other application domains besides 3D component layout in control cabinets.

Research Question: The scientific objective is to study and develop an approach for multi-objective constraint optimization of a control cabinet considering various aspects including physical dimensions and port locations of particular modules, interference, heat dissipation, and physical dimensions of the custom cabinet. To demonstrate usefulness of our approach we will develop a demonstration tool using the API of EPLAN.

This will require that we develop new algorithms for optimizing placement of modules in a 3D layout. The tool must monitor and adapt to changes in requirements (new customized cabinets not seen before), and the algorithms must adapt a known configuration plan accordingly. The tool should also support online learning for incremental improvement of the component layout of a certain type of control cabinet. This gives rise to our research question:

Based on a 2D schematic of a control cabinet, how we can determine the optimal location of components in the 3D cabinet model?

5.3.2 Objectives, Expected Results and Benefits

Objective RO-C1: The scientific benefit of the project will be development of new and/or improvements of existing search-based optimization algorithms to allow for automatic 3D placement of components, while at the same time take into consideration constraints of the limited space and electrical features of individual components. The algorithms will be able to adapt to changes in the requirements to create layouts of previously unknown schematics, and support incremental improvement of layouts. Search-based optimization algorithms have been used in various areas such as staff scheduling, service oriented architectures, computer game AI and much more. Our project aims to provide advancements in this field, which can benefit a wide range of application areas.

Objective IO-C1: The goal is to develop a system that based on a 2D schematic of a customized control cabinet, determines the optimal location of components in the 3D cabinet model. A standalone demonstration tool will be implemented that uses the built in API of EPLAN to get details about the schematic and each individual component, and proposes a configuration plan of the cabinet to the user. The demonstration tool will be a proof-of-concept, and could be included as a module in the future versions of EPLAN software. The placement module of the tool can likely be adapted to other applications such as furniture placement, construction site planning and placement of other types of modules/components.

Effects and Benefits: The results of our project will in particular benefit our industrial partners: Yaskawa Nordic AB and EPLAN Software & Service AB. Yaskawa is currently using the software of EPLAN for two-dimensional CAD-drawings, the further steps of production of control cabinets are largely completed manually. Our project will contribute to a better integration of EPLAN software in the Yaskawa and create of new business opportunities by offering advanced integrated solutions comprising EPLAN software and Yaskawa robots that will likely increase the production rate of control cabinets. The value for EPLAN is further improvements to the automatic assembly capabilities of the EPLAN software.

Dissemination of Scientific Results: We will disseminate scientific results of our project at appropriate international conferences (such as, SASO, GECCO, Euro-Par) and journals (such as, Springer journal on Genetic Programming and Evolvable Machines, ACM Transactions on Parallel Computing) in the form of peer-review papers. We plan to publish one conference paper per year and one journal paper every second year.

5.3.3 Implementation

Project Organization: Industrial partners involved in this project are Yaskawa Nordic AB and EPLAN Software & Service AB. Yaskawa has a manufacturing facility for its control cabinets in Torsås where they produce about 4000 cabinets yearly. EPLAN Software & Service AB is a major provider of software-based engineering solutions for mechatronics with over 45,000 customers and 110,000 licenses of their CAD system worldwide. The CAD system supports going from a manually drawn 2D schematic to the design and assembly of control cabinets.

Yaskawa will provide access to an industrial robot system, and know-how with respect to industrial robot-systems. EPLAN will provide access to CAD software tools for design of control cabinets, and know-how related to EPLAN CAD tools. LNU will study and develop a constraint-based optimization method and the corresponding tool-support (using EPLAN API) for optimized automatic 3D arrangement of components within a control cabinet.

Workflow and Project Plan: The project will be divided into the following phases: Analysis, Implementation, Evaluation, and Synergy integration. We will adopt an iterative software-engineering process with two iterations, cf. Table 1 of the document "Overall Project Plan". The first prototype will be available after 18 months; after the completion of evaluation of the first prototype at project month 24, the second iteration will be conducted.

WP-C1 [*Analysis* I]: Arranging electrical components within limited physical space (such as, a control cabinet) has to take into consideration general rules, requirements and constraints on each individual component as well as the cabinet as a whole. Together with our industrial partners Yaskawa and EPLAN we will create a specification of the rules, requirements and constraints our system must follow.

The EPLAN software is used to design customized control cabinets, starting from a 2D schematics to a complete 3D layout in a virtual cabinet with position of each module and wiring of power cables and connectors. EPLAN is a complex software system with rich functionality. In this phase we will attend training sessions at the industrial partner EPLAN to learn all necessary details about the software to carry out the project. This will also include learning about the built-in API that will be used to read the 2D schematics of a control cabinet.

In this phase a thorough review of available constraint optimization algorithms for placement of objects will be carried out. A set of possible candidate algorithms that suits our problem will be selected and evaluated if they can, with or without modifications, be used in the system developed in the project.

WP-C2 [*Implementation* I]: A software module for connecting to the EPLAN software API will be developed. We will investigate if all necessary details about placement and individual components for generating an optimal 3D layout are available through the API. If some knowledge cannot be obtained from the API we will investigate alternative options for retrieving the missing knowledge.

A constraint-based optimization algorithm that uses heuristics for generating a 3D layout of a cabinet from 2D schematics will be designed, developed and implemented. The algorithm has to take into consideration individual constraints of each component, general constraints of the cabinet as well as optimize for spacing, heat dissipation and interference.

Internally in the EPLAN software project the placement of components and wiring is stored in a Microsoft Excel file. The EPLAN software reads the Excel file to show a virtual 3D layout of the cabinet. In this phase we will develop a module that generates an Excel file for a 3D layout from a 2D schematics of a cabinet. The Excel file has to follow the interface and rules of the EPLAN software to make sure it can be used correctly in the software.

WP-C3 [*Evaluation* I]: The generated Excel files from several 2D schematics of customized cabinets will be evaluated if they can be read and interpreted correctly by the EPLAN software.

Experts from our industrial partners will evaluate the generated 3D layouts of several customized cabinets to see if the placement of components is correct and optimal.

SP-C1 [*Synergy* I]: Integration of our optimization framework into TAPPAS, and adaption to the power optimization problem domain of Project A.

WP-C4 [*Analysis* II]: The EPLAN software can generate instructions in a robot understandable format for automatic assembly of control cabinets. In this phase we will analyze what is needed in terms of hardware and software to construct a robot working cell for automatic assembly of control cabinets at our industrial partner Yaskawa. Furthermore, we will study parallelization strategies for our optimization algorithm.

WP-C5 [*Implementation* II]: In this phase, we will implement a parallel version of our optimization algorithm and we will construct the robot working cell for automatic assembly using Yaskawa robots at the Yaskawa factory. The goal is to install the individual modules in the cabinet using robots. It is also possible to do automatic wiring, but it requires specialized robots not part of the Yaskawa product catalogue and thus will not be part of this project.

WP-6 [*Evaluation* II]: In this phase we will evaluate together with our industrial partners how well the constructed robot working cell works. The evaluation will take into consideration what the working cell is capable of, limitations, and performance- and security aspects.

SP-C2 [*Synergy* II]: Integration of our improved optimization framework into TAPPAS, and adaption to the power optimization problem domain of Project A.

The members of the team have the required expertise in the project-relevant domains: CAD software (Fredrik Landh), robotics (Robert Bickö, Johan Hagelbäck), artificial intelligence (Johan Hagelbäck), optimization (Johan Hagelbäck and Sabri Pllana), parallel computing (Sabri Pllana), large-scale project coordination (Sabri Pllana). The Ph.D. student at LNU will develop under the supervision of Sabri Pllana and Johan Hagelbäck a sequential and a parallel version of a search-based optimization algorithm for 3D arrangement of components.

Person	Affiliation	Percentage
Sabri Pllana (PI)	Linnaeus University (org. number 202100-6271)	20%
Johan Hagelbäck	Linnaeus University (202100-6271)	30%
Ph.D. student	Linnaeus University (202100-6271)	50%
Robert Bickö	Yaskawa Nordic AB 556020-9990)	10%
Fredrik Landh	EPLAN Software & Service AB (556588-8723)	10%

Table 8: Members of the project team C.

Budget Motivation: To carry out the project the in-kind contribution of Yaskawa Nordic AB must include: (1) access to an industrial robot system, (2), training sessions for LNU and EPLAN for understanding how automatic assembly of cabinets can be incorporated in the Yaskawa construction processes, and (3) time for R&D personnel to successfully integrate automatic assembly in the Yaskawa production line.

The in-kind contribution of EPLAN Software & Service AB must include: (1) licenses for the EPLAN software during the whole project duration, (2) training sessions for LNU and Yaskawa personnel to learn about the EPLAN software modules for automatic assembly of control cabinets, and (3) time for R&D personnel involved in the project to support the usage of the EPLAN software.

	Item	Contribution [SEK]
YASKAWA	Industrial robot system (4 years access)	640 000
	R&D time	320 000
	Training (three persons)	180 000
	YASKAWA total	**1 140 000**
EPLAN	Electric P8 Professional (4 years; three licenses)	401 000
	ProPanel (4 years; three licenses)	123 000
	R&D time	320 000
	Training for P8 Professional (three persons)	84 000
	Training for ProPanel (three persons)	30 000
	EPLAN total	**958 000**
	TOTAL Contribution	**2 098 000**

Table 9: Contributions of industrial partners of project C.

Risk Analysis: We have addressed management risks in Section 4.5 and therefore we focus here on project-specific technical risks. The API included in the EPLAN software might not contain enough functionality to get all necessary information from the 2D schematic or individual components needed to create an optimal 3D layout. If this occurs, we will provide our recommendations for extensions of the API in EPLAN Software.

6 Suggested Reviewers

- Uwe Assmann, TU Dresden, Germany (Synergy).
- Wolfgang Reif, University of Augsburg, Germany (Synergy)
- Leonardo Mostarda, Middlesex University, UK (Project A)
- Kari Systä, Tampere University of Technology, Finland (Project A)
- Markus Helfert, Dublin City University, Ireland (Project B)
- Miroslaw Staron, University of Gothenburg, Sweden (Project B)
- Krzysztof Kuchcinski, Lund University, Sweden (Project C)
- Håkan Grahn, BTH, Sweden (Project C)

References

Publications with co-authors who are team members of the proposed project are indicated by **bold** font.

[1] T. Bures, D. **Weyns**, M. Klein, Rodolfo E. Haber, 1st International Workshop on Software Engineering for Smart Cyber-Physical Systems, International Conference on Software Engineering, Florence 2015.

[2] N. Esfahani, E. Kouroshfar, and S. Malek. Taming Uncertainty in Self-Adaptive Software." In proceedings of the 8th joint meeting of the European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering, 2011

[3] D. **Weyns**, N. Bencomo, R. Calinescu, J. Camara, et al. Perpetual assurances in self-adaptive systems. In Assurances for Self-Adaptive Systems, Dagstuhl Seminar 13511, 2014.

[4] L. Baresi, C. Ghezzi: The disappearing boundary between development-time and run-time. In: Proc. of the FSE/SDP workshop on Future of software engineering research (FoSER'10). pp. 17--22. ACM, New York (2010)

[5] B. Cheng et al., *Software Engineering for Self-Adaptive Systems: A Research Roadmap*, Lecture Notes in Computer Science vol. 5525, Springer 2009

[6] R. de Lemos, et al., *Software Engineering for Self-Adaptive Systems: A second Research Roadmap*, Lecture Notes in Computer Science vol. 7475, Springer 2013

[7] P. Oreizy, N. Medvidovic, R. N. Taylor: Architecture-Based Runtime Software Evolution. ICSE 1998: 177-186

[8] J. O. Kephart, D. M. Chess: The Vision of Autonomic Computing. IEEE Computer 36(1): 41-50, 2003

[9] D. Garlan, S. Cheng, A. Huang, B. Schmerl and P. Steenkiste. Rainbow: Architecture-Based Self Adaptation with Reusable Infrastructure. In IEEE Computer, Vol. 37(10), October 2004.

[10] N. D'Ippolito, V. Braberman, J. Kramer, J. Magee, D. Sykes, and S. Uchitel, Hope for the Best, Prepare for the Worst: Multi-tier Control for Adaptive Systems, ICSE 2014

[11] J. L. *Hellerstein*, Y. Diao, S. Parekh, and D. M. Tilbury. Feedback Control of Computing Systems. John Wiley & Sons, 2004.

[12] H. Hoffmann, M. Maggio, M. D. Santambrogio, A. Leva, and A. Agarwal, SEEC: A Framework for Self-aware Computing, MIT CSAIL Technical Report, MIT-CSAIL-TR-2010-049, October 2010.

[13] R. Calinescu, C. Ghezzi, M. Z. Kwiatkowska, R. Mirandola: Self-adaptive software needs quantitative verification at runtime. Comm. ACM 55(9): 69-77 (2012)

[14] J. Dongarra, V. Eijkhout: Self-Adapting Numerical Software and Automatic Tuning of Heuristics. International Conference on Computational Science 2003: 759-770

[15] M. Emani, Z. Wang, M. O'Boyle: Smart, adaptive mapping of parallelism in the presence of external workload. CGO 2013: 1-10

[16] J. **Andersson**, R. de Lemos, S. Malek, and D. **Weyns**, *Modeling Dimensions of Self-Adaptive Software Systems*, Software Engineering for Self-Adaptive Systems, Lecture Notes in Computer Science, vol. 5525, Springer 2009

[17] D. **Weyns**, S. Malek, and J. **Andersson**, *FORMS: Unifying Reference Model for Formal Specification of Distributed Self-Adaptive Systems*, ACM Transactions on Autonomous and Adaptive Systems, TAAS, 7(1), 2012

[18] C. W. Kessler, W. **Löwe**, *Optimized composition of performance-aware parallel components*. Concurrency and Computation: Practice and Experience 24(5): 481-498, April 2012

[19] S. **Pllana**, S. Benkner, E. Mehofer, L. Natvig, and F. Xhafa. *Agent-supported Programming of Multi-core Computing Systems*. In the book Complex Intelligent Systems and Their Applications. Published by Springer in the book series Springer Optimization and Its Applications, Vol. 41, 2010.

[20] E. Österlund, W. **Löwe**: *Dynamically transforming data structures*. ASE 2013: 410-420.

[21] E. Österlund, W. **Löwe**: Concurrent transformation components using contention context sensors. ASE 2014: 223-234

[22] S. Benkner, S. **Pllana**, J. L. Träff, P. Tsigas, U. et al., *PEPPHER: Efficient and Productive Usage of Hybrid Computing Systems*, IEEE Micro, vol. 31, no. 5, pp. 28-41, Sep./Oct. 2011

[23] R. Haesevoets, D. **Weyns**, T. Holvoet, *Architecture-Centric Support for Adaptive Service Collaborations*, ACM Transactions on Software Engineering and Methodology (TOSEM) Feb 2014.

[24] D. **Weyns**, B. Schmerl, V. Grassi, S. Malek, R. Mirandola, C. Prehofer, J. Wuttke, J. **Andersson**, H. Giese, and K. Goschka, *On Patterns for Decentralized Control in Self-Adaptive Systems*, Software Engineering for Self-Adaptive Systems II. Lecture Notes in Computer Science Volume 7475, 2013, pp 76-107.

[25] D. Gil de la Iglesia and D. **Weyns**, MAPE-K Formal Templates to Rigorously Design Behaviors for Self-Adaptive Systems, ACM Transactions on Autonomous and Adaptive Systems, TAAS accepted 2015.

[26] J. **Andersson**, L. Baresi, N. Bencomo, R. de Lemos, A. Gorla, P. Inverardi and T. Vogel, *Software Engineering Processes for Self-Adaptive Systems*, Software Engineering for Self-Adaptive Systems II Lecture Notes in Computer Science vo. 7475, Springer 2013.

[A1] D. Cook, S. Das. Smart Environments: Technology, Protocols and Applications. Wiley-Interscience. (2005) ISBN 0-471-54448-5.

[A2] B. Selic, "The Pragmatics of Model-driven Development", In IEEE Software, vol. 20, no. 5, pp. 19–25, 2003.

[A3] R. France, B. Rumpe, "Model-driven Development of Complex Software: A Research Roadmap", In Proc. of FOSE 2007, pp. 37–54, IEEE, 2007.

[A4] T. Stahl, M. Völter. "Model-Driven Software Development", Wiley, 2006.

[A5] G. S. Blair, N. Bencomo, and R. B. France. Models@runtime. IEEE Computer, 42(10), 2009.

[A6] H. J. Goldsby and B. H. Cheng. Automatically generating behavioral models of adaptive systems to address uncertainty. In Proc. of the 11th international conference on Model Driven Engineering Languages and Systems (MoDELS), 2008.

[A7] **M. Caporuscio**, A. D. Marco, and P. Inverardi, "Model-based system reconfiguration for dynamic performance management". Journal of Systems and Software, vol. 80, iss. 4, pp. 455-473, 2007.

[A8] D. Garlan, B. Schmerl, and J. Chang. Using gauges for architecture-based monitoring and adaptation. In Proc. of the Working Conference on Complex and Dynamic Systems Architecture, 2001.

[A9] **M. Caporuscio**, M. Funaro, and C. Ghezzi. "Architectural issues of adaptive pervasive systems". In Engels, G., Lewerentz, C., Schäfer, W., Schürr, A., and Westfechtel, B., Eds., Graph Transformations and Model-Driven Engineering, volume 5765 of LNCS. 2010.

[A10] C. H. Lung, S. Bot, K. Kalaichelvan, R. Kazman, An approach to software architecture analysis for evolution and reusability, Conference of the Centre for Advanced Studies on Collaborative Research (CASCON), 1997.

[A11] H. P. Breivold. Software Architecture Evolution through Evolvability Analysis. PhD thesis, Mälardalen University, School of Innovation, Design and Engineering, 2011.

[A12] P. Oreizy. Open architecture software: a flexible approach to decentralized software evolution. Phd thesis, University of California, Irvine, 2000.

[A13] **B. Vogel**, **A. Kurti**, T. Mikkonen, M. Milrad. Towards an Open Architecture Model for Web and Mobile Software: Characteristics and Validity Properties. In Proceedings of the 38th Annual International Computers, Software & Applications Conference (COMPSAC'2014), IEEE, Västerås, Sweden, July 21-25, 2014.

[A14] M. Autili, **M. Caporuscio**, V. Issarny, and L. Berardinelli, "Model-driven engineering of middleware-based ubiquitous services", Software & Systems Modeling, vol. 13, iss. 2, pp. 481-511, 2014.

[B1] F. B. Abreu, "The MOOD metrics set," in *Proceedings eCOOP workshop on metrics*, 1995.

[B2] V. R. Basili, "The role of experimentation in software engineering: Past, current, and future," in *ICSE '96: Proceedings of the 18th international conference on software engineering*, 1996, pp. 442–449.

[B3] G. Baxter, M. Frean, J. Noble, M. Rickerby, H. Smith, M. Visser, H. Melton, and E. Tempero, "Understanding the shape of java software," *SIGPLAN Not.*, vol. 41, no. 10, pp. 397–412, Oct. 2006.

[B4] S. R. Chidamber and C. F. Kemerer, "A Metrics Suite for Object-Oriented Design," *IEEE Transactions on Software Engineering*, vol. 20, no. 6, pp. 476–493, 1994.

[B5] **M. Ericsson, W. Lowe,** T. Olsson, D. Toll, and **A. Wingkvist**, "A study of the effect of data normalization on software and information quality assessment," in *20th Asia-Pacific Software Engineering Conference*, 2013, vol. 2, pp. 55–60.

[B6] R. Harrison, S. J. Counsell, and R. V. Nithi, "An Investigation into the Applicability and Validity of Object-Oriented Design Metrics," *Empirical Software Engineering*, vol. 3, no. 3, pp. 255–273, 1998.

[B7] D. E. Knuth, "An empirical study of fORTRAN programs," *Software - Practice and Experience*, vol. 1, no. 2, pp. 105–133, 1971.

[B8] W. Li and S. Henry, "Maintenance metrics for the object oriented paradigm," in *IEEE proceedings of the first international software metrics symposium*, 1993, pp. 52–60.

[B9] **R. Lincke, W. Löwe,** and T. Gutzmann, "Software Quality Prediction Models Compared," in *10th International Conference on Quality Software*, 2010.

[B10] **R. Lincke,** J. Lundberg, and **W. Löwe,** "Comparing software metrics tools," in *Int. Symp. Software Testing and Analysis*, 2008, pp. 131–142.

[B11] J. A. McCall, P. G. Richards, and G. F. Walters, "Factors in Software Quality," NTIS, NTIS Springfield, VA, Volume I, 1977.

[B12] S. McConnell, *Code Complete, Second edition*, Microsoft Press, 2004.

[B13] K. Mordal-Manet, F. Balmas, S. Denier, S. Ducasse, H. Wertz, J. Laval, F. Bellingard, and P. Vaillergues, "The SQUALE model; a practice-based industrial quality model," in *International Conference on Software Maintenance*, 2009, pp. 531–534.

[B14] A. Ramirez, B. Cheng, and P. McKinley, "Adaptive monitoring of software requirements," in *1st International Workshop on Requirements@Run.Time*, 2010, pp. 41–50.

[B15] B. Vasilescu, A. Serebrenik, and M. van den Brand, "You can't control the unfamiliar: A study on the relations between aggregation techniques for software metrics," in *27th International Conference on Software Maintenance (ICSM), 2011*, 2011, pp. 313–322.

[B16] N. M. Villegas, H. A. Müller, G. Tamura, L. Duchien, and R. Casallas, "A framework for evaluating quality-driven self-adaptive software systems," in *6th International Symposium on Software Engineering for Adaptive and Self-managing Systems*, 2011, pp. 80–89.

[B17] R. Y. Wang and D. M. Strong, "Beyond accuracy: What data quality means to data consumers," *J. Manage. Inf. Syst.*, vol. 12, no. 4, pp. 5–33, 1996.

[B18] **A. Wingkvist, M. Ericsson, and W. Löwe,** "A software infrastructure for information quality assessment," in *16th international conference on information quality*, 2011.

[B19] **A. Wingkvist, M. Ericsson, and W. Löwe,** "Analysis and visualization of information quality of technical documentation," *The Electronic Journal of Information Systems Evaluation*, vol. 14, no. 1, pp. 150–159, 2011.

[B20] **A. Wingkvist, M. Ericsson, and W. Löwe**, "Making sense of technical information quality — a software-based approach," *Journal of Software Technology: Analyzing and Measuring Information Quality*, vol. 14, no. 3, pp. 12–18, 2011.

[B21] **A. Wingkvist, M. Ericsson, R. Lincke, and W. Löwe,** "A metrics-based approach to technical documentation quality," in *7th Int. Conf. Quality of Information and Communications Technology,* 2010.

[C1] S. Russell, P. Norvig, *Artificial Intelligence – A modern Approach*. 3rd ed. Pearson, 2009.

[C2] S. Sanchez, O. Le Roux, H. Luga, and V. Gaildrat. "Constraint-Based 3D-Object Layout using a Genetic Algorithm", proceedings of *Sixth International Conference on Computer Graphics and Artificial Intelligence*, 2003.

[C3] H. M. Osman, M. E. Georgy, and M. E. Ibrahim, "A hybrid CAD-based construction site layout planning system using genetic algorithms", *Automation in Construction*, vol. 12, issue 6, 2003.

[C4] L-F. Yu, S-K Yeung, C-K Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher, "Make it Home: Automatic Optimization of Furniture Arrangement", proceedings of *ACM SIGGRPAH 2011*, vol. 30, issue 4, 3022.

[C5] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, "Example-based Synthesis of 3D Object Arrangements", proceedings of *ACM SIGGRAPH Asia*. vol. 31, issue 6, 2012.

[C6] A. Petcu, and B. Faltings. "A Scalable Method for Multiagent Constraint Optimization", proceedings of the *19th International joint conference on Artificial Intelligence*, 2005.

[C7] J. Togelius, M. Preuss, N. Beume, S. Wessing, **J. Hagelbäck**, G. N. Yannakakis. Multiobjective Exploration of the StarCraft Map Space, proceedings of 2010 IEEE Conference on Computational Intelligence and Games (CIG), 2010.

[C8] J. Togelius, M. Preuss, N. Beume, S. Wessing, **J. Hagelbäck**, G. N. Yannakakis, C. Grappiolo, Controllable procedural map generation via multiobjective evolution", Journal of Genetic Programming and Evolvable Machines", vol. 14, issue 2, pages 245-277, 2013.

[C9] J. Dokulil, E. Bajrovic, S. Benkner, **S. Pllana**, M. Sandrieser, B. Bachmayer, High-level Support for Hybrid Parallel Execution of C++ Applications Targeting Intel® Xeon Phi™ Coprocessors, Procedia Computer Science, Volume 18, Elsevier 2013, Pages 2508-2511, ICCS 2013.