EMPAT — Evolvable Modularity Patterns: From Inconvenient Truth to Evolvable Engineering

Special Session along with PATTERNS 2018, February 18 - 22, 2018 - Barcelona, Spain
http://www.iaria.org/conferences2017/PATTERNS17.html

Herwig T. Mannaert
University of Antwerp
Antwerp, Belgium

***Abstract*—Modularity, or the division of systems into interrelated and hierarchical subsystems, has been considered to be a powerful concept in many domains for many decades. While modular artifacts are believed to have the potential to exhibit several beneficial characteristics, including evolvability, the actual realization of this evolvability — or flexibility over time — remains challenging. Actually, systems in almost all domains, both physical artefacts and administrative and/or information systems, face severe evolvability issues, which are closely related to some of the most pressing challenges of our time. The availability of design rules and patterns, allowing the design of systems that exhibit higher levels of evolvability than currently known, could lead to huge benefits in terms of productivity, sustainability, and innovation. In the PATTERNS 2018 conference in Barcelona, Spain, a second special session on *Evolvable Modularity Patterns* is included. Three papers explore architectural patterns that could contribute to attaining evolvable modular structures, and therefore to the benefits that such architectures could entail.**

Keywords–Modularity; Design Science; Evolvability; Architectural Patterns.

I. Introduction

In his seminal work *The Sciences of the Artificial* [1], Herbert Simon already stated in 1969 that *the architecture of the artificial is hierarchy*: systems consist of interrelated subsystems, hierarchic in structure, until some lowest level of elementary subsystem. Today, the fact that a system is subdivided into a set of interacting subsystems, is generally referred to as modularity. This concept of modularity is a cornerstone of engineering and design, and has proven to be a very powerful technique in many domains, such as computer science, mechanical and electrical design, product engineering, and even organizational sciences [2].

Modular artifacts are considered desirable due to several potential benefits which are attributed to it. For instance, designing a product in a modular way is expected to lower the complexity, as the design can be decomposed into a set of smaller (less complex) problems. Another major benefit expected from modularity, is the reuse of modular components. Indeed, various subsystems or modules could be reused in other systems. Modularity is also associated with increased flexibility, and/or evolvability. In a modular artifact, one particular part (module) of the system can be substituted for another version of it, without having to build up the artifact again from scratch. This kind of plug-and-play behavior also allows for variation (using the same set of available module versions, different aggregations or variants can be made available) and evolvability (over time, an artifact can evolve from one variant to another), and is deemed very powerful.

However, achieving these modular benefits is not straightforward, and clearly related to the design and architecture of the various modules or interrelated subsystems. It is generally accepted that the coupling, the dependencies and interactions, between the modules in a system should be studied and minimized. After a generic analysis of these interactions by Herbert Simon [1], Dave Parnas presented in 1972 a specific analysis — targeted at software engineering — of the criteria to be used in decomposing systems into modules [3], where he formulated his *double dictum*, of low coupling and high cohesion. In their seminal work, Carlyss Baldwin and Kim Clark analyzed in 2000 the modularity conundrum [4], and presented a set of design rules, in order to truly leverage the power and value of modularity.

Nevertheless, after all these years, it still remains often unclear how the benefits and value of modularity should be realized in specific situations. In particular, several features in modular structures are *cross-cutting* (e.g., security in a software application) in the sense that they are required across the whole modular structure (e.g., every data entity should be securely stored). As adaptations in such cross-cutting concerns can create large ripple-effects in the system (i.e., a change in one module implies a change in another module and so on), the evolvability — the ability to change and evolve — of these systems is clearly not obvious. Therefore, we may conclude that the design of the modular architecture of a system remains a very complex and high-dimensional problem.

II. An Inconvenient Truth

In today's competitive business environment, companies need to be able to adapt to changing customer or regulatory requirements, competitors, suppliers, substitute products or services, and newcomers to the market [5]. It is clear that lack of evolvability in the modular design, for instance due to the ripple effects related to adaptations in cross-cutting concerns as described above, may hamper the adaptability of various systems, and therefore the ability of companies to react to a changing environment.

Already in the nineteen-seventies, Manny Lehman identified this lack of adaptability and evolvability in large software systems, and formulated his *law of increasing complexity* [6], stating: *As an evolving program is continually changed, its complexity, reflecting deteriorating structure, increases unless work is done to maintain or reduce it*. Even today, it seems that current technologies and tools do not enable software engineers to provide the levels of evolvability required in the ever faster changing world we currently live in [7]. It is widely believed that most corporate IT departments spend the majority of their resources (even between 70% and 90%) maintaining existing information systems, posing a serious threat to the *productivity* of information technology.

This lack of evolvability in man-made artefacts, hampering the ability of systems and organizations to react to changing environments, is not limited to software systems. It was for instance demonstrated in [8], that companies obliged to report financial data in multiple GAAP, face huge issues and ripple effects when forced to comply with new financial regulations or accounting standards. In [9], it was shown that in today's flexible education landscape, universities face huge transitional ripple effects when performing small changes to a study program. All these examples show that these evolvability issues do not only *threaten the productivity*, but also *hamper the ability to react* to changing environments, and therefore *limit the possibilities to innovate*.

These evolvability issues are not limited to administrative processes either, but surface just as well in the design and engineering of physical artefacts. Most people know that extending a house with one or two additional rooms, may lead to lots of complications, in particular due to changes in the various utility conduits (water, heating, electricity, etcetera), being the cross-cutting concerns of the construction. In order to reduce carbon emission levels for instance, older cars are currently being banned from our inner cities. It would be nice if the possibility existed to replace the engines of those cars with a cleaner and more modern engine, instead of directing the entire car to the junkyard. And in [10], an example of an 18 year old racing bike is presented, where the necessary replacement of a gear handle entails the additional replacement of the gear blocks, cables, and chains, leading to the inevitable conclusion that the racing bike ends up in the junkyard. These examples of evolvability issues in physical artefacts, clearly indicate that lack of evolvability *seriously hampers the scalability and sustainability* of man-made artefacts.

Therefore, we may conclude that evolvability issues are a widespread and fundamental phenomenon in the current state of engineering and design, both in technology and administrative processes. As these issues are seldomly expressed open and upfront, we might even call it an inconvenient truth.

III. Imagine Evolvable Modularity

We have explained in the previous section that issues with respect to the evolvability of systems, are closely related to some of the most pressing challenges our society faces: productivity, adaptability, scalability, and sustainability of man-made artefacts. This implies that we should strive to provide designers and engineers with better and more specific guidance to design and create systems with higher levels of evolvability than currently available. Developing design and engineering guidance and knowledge belongs scientifically to *design science research*, as introduced by Richard Buckminster Fuller [11], and endorsed by Herbert Simon [1].

The research field within design science aiming to facilitate the design and engineering of systems exhibiting high levels of evolvability, may lead to generic and/or domain-specific rules, that need to be obeyed in order to design evolvable systems. As a large part of design knowledge is currently consolidated in so-called design patterns, the research should probably entail generic and/or domain-specific architectural patterns as well. Examples of such generic patterns that exhibit various degrees of evolvability, have for instance been proposed in [10]. The discipline as a whole, striving to enhance the body of knowledge supporting the design of evolvable artefacts and processes, could be identified as *evolvable modularity*.

The potential of such a discipline seems extremely promising, if only because it is related to some of the most pressing challenges our society faces, like scalability and sustainability of man-made artefacts. In order to show this, let us simply imagine that such a thorough scientific basis exists for evolvable modularity, allowing us to redefine the discipline of engineering, and to be capable of *evolvable engineering*.

In a world capable of evolvable engineering, many desirable evolutions would suddenly appear to be at our fingertips. Houses could be enlarged or reduced in size, without tearing down walls or facing serious issues with utility conduits. Electricity distribution could be optimized without impacting all electrical devices in the world. Entire buildings could be given another purpose by simply rearranging them, instead of demolishing the entire construction. Older cars could be made cleaner and less polluting by replacing (parts of) their engines, instead of bringing them to the junkyard. Aging racing bikes could be repaired by replacing specific parts instead of throwing them away. Innovative services could be defined and offered in an instant, without worrying about the tedious changes to the underlying information systems. Information systems in general could be maintained and upgraded without numerous ripple effects, currently burdening the operations and budgets of IT departments. University faculties could introduce innovative courses and adapt the study programs, without fearing the avalanche of transition effects, and the numerous changes in their information systems. Improved explanations of technicalities could be inserted and automatically distributed, without requiring manual changes in all books, white papers, and presentations containing those details. Rules for auditing and financial reporting by corporations could be modified, without entailing huge changes in all accounting and/or information systems around the globe.

The sheer magnitude and almost utopian nature of the possibilities, point to the importance of evolvable modularity in general, and the corresponding research in particular. In a special session on *Evolvable Modularity Patterns*, held as part of the PATTERNS 2017 conference in Athens, Greece, four papers were presented that explore architectural patterns in several domains to attain higher levels of evolvability. In this second EMPAT session, part of the PATTERNS 2018 conference in Barcelona, Spain, three papers are pressented. Marek Suchànek et al. propose an initial conceptualization for the evolvable representation and management of documents. In a case study based approach, Peter De Bruyn et al. attempt to chart the various dimensions of variability in evolving software applications. A third paper by Herwig Mannaert et al. studies the evolvability of a specific code generation pattern.

REFERENCES

[1] H. Simon, The Sciences of the Artificial. MIT Press, 1969.

[2] D. Campagnolo and A. Camuffo, "The Concept of Modularity in Management Studies: A Literature Review." International Journal of Management Reviews, vol. 12, no. 3, 2010, pp. 259–283.

[3] D. Parnas, "On the criteria to be used in decomposing systems into modules," Communications of the ACM, vol. 15, no. 12, 1972, pp. 1053–1058.

[4] C. Y. Baldwin and K. B. Clark, Design Rules: The Power of Modularity. Cambridge, MA, USA: MIT Press, 2000.

[5] M. E. Porter, "Strategy and the Internet." Harvard Business Review, vol. 79, no. 3, 2001, pp. 62–78, 164.

[6] M. M. Lehman, "Programs, life cycles, and laws of software evolution," Proceedings of the IEEE, vol. 68, no. 9, sep 1980, pp. 1060–1076.

[7] H. Mannaert and J. Verelst, Normalized Systems: Re-creating Information Technology Based on Laws for Software Evolvability. Koppa, 2009.

[8] E. Vanhoof, "Evolvable accounting information systems: applying design science methodology and normalized systems theory to tackle combinatorial effects of multiple gaap," Ph.D. dissertation, University of Antwerp, 2016.

[9] G. Oorts, H. Mannaert, P. De Bruyn, and I. Franquet, On the evolvable and traceable design of (under)graduate education programs. Springer International Publishing, 2016, vol. 252.

[10] H. Mannaert, J. Verelst, and P. De Bruyn, Normalized Systems Theory: From Foundations for Evolvable Software Toward a General Theory for Evolvable Design. Koppa, 2016.

[11] R. B. Fuller, A Comprehensive Anticipatory Design Science. Royal Architectural Institute of Canada, 1957.