Predictability and Evolution in Resilient Systems

Ivica Crnkovic

School of Innovation, Design and Engineering, Mälardalen University, Västerås, Sweden
ivica.crnkovic@mdh.se

Abstract. This paper gives a short overview of the talk related to the challenges in software development of resilient systems. The challenges come of the resilience characteristic as such it a system emerging lifecycle property, neither directly measurable nor computable. While software is an essential part of a system, its analysis not enough for determining the system resilience. The talk will discuss about system resilience reasoning, its limitations, and possible approaches in the software design that include resilience analysis.

1 System Resilience, Robustness and Sustainability

In the last decade the importance of resilience as an intrinsic system attribute is arising in every domain of system and software development. Resilience is an attribute often related to robustness, and survivability (and by this dependability) from one side, and sustainability from other side [1]. Robustness and dependability are the important properties of many systems. Such systems are traditionally analyzed and designed in a top-down manner by defining the system boundaries, requirements and the system specifications with a goal to achieve predictable behavior in predictable (though not necessary known) situations. The main challenges and the main problems in these types of systems occur due to unforeseen events. For this reasons one important element of the quality assurance process is to ensure that such events do not happen, or the probability they can happen is minimized. The sustainability attribute on the other hand is the capacity to endure, related to resource constraints and maintenance of status quo of the external environment (i.e low negative impact in short and long terms). These characteristics are complementary and not necessary compatible, likely to require trade-off analysis to find a satisfactory balance. The resilience property should embodies both these properties the resilient systems should provide their services in a trustworthy and sustainable way even when the circumstances change; this may include drastic changes. More and more systems are required to perform in this way. The systems became more complex, more interactive, being integrated with another systems, and are continuously evolving. For this reason new characteristics of such systems become important; these systems must be adaptive and self-organizing. They are evolving, as the external environment is evolving.

Since software as an immanent part of most of the systems, the software development has to include the same concerns of the system resilience design, though the method can be specific for software. This talk will focus on two different software development aspects for resilient systems bottom-up and top-down approaches; a) on the software and system attributes composability, and b) software evolution analysis.

2 Compositions of System and Software Attributes Limitations

The modern paradigm of software development includes (re)use and possible adaptation of the existing software, either in a form of components, services, underlying platforms and infrastructure, or even complete software systems. This calls for design methods that use bottom-up approach, and composition of the existing components. The efficient development requires not only efficient mechanism for composition of functional properties, but also extrafunctional properties. In the concrete case, we can sate a question: Can we, and under which assumptions, reason about resilience composition? This talk will present different types of extrafuctional properties with respect to their compositions and composition assumptions [2, 3].

3 Software Evolvability Analysis and Resilience Property

A complementary approach to composability is a top-down approach analysis of a system in respect to resilience. Since resilience is a complex property, it depends on a number of subproperties that may be easier analyzed and in some cases measured. The question is which are these subproperties. In this talk we shall present a method and a model for analyzing software evolvability [4] , one of the subcharacteristics of resilience. The example, based on industrial case studies [5], shows a method for evolvability assessment of software architecture. The method includes identification of subcharacteristics of evolvability and the assessment process, based on quantitative and qualitative methods, The example can be used as a starting point for resilience assessment.

References

1. Fiksel, J.: Designing resilient, sustainable systems. Environmental Science and Technology **37**(23) (May 2003) pp. 5330–5339
2. Crnkovic, I., Sentilles, S., Vulgarakis, A., Chaudron, M.R.: A classification framework for software component models. IEEE Transactions on Software Engineering **In Press** (2010)
3. Crnkovic, I., Larsson, M., Preiss, O.: Concerning predictability in dependable component-based systems: Classification of quality attributes. Springer, LNCS 3549 (2005) pp. 257 278
4. Breivold, H.P., Crnkovic, I., Larsson, M.: A systematic review of software architecture evolution research, information and software technology. Information and Software Technology **In Press** (June 2011)
5. Pei-Breivold, H., Crnkovic, I., Eriksson, P.: Analyzing software evolvability. In: IEEE International Computer Software and Applications Conference (COMPSAC 2008). (July 2008)