Pattern-based Formalization of Safety Requirements

The composing process of safety requirements is a very important step in the field of software development, especially with the fault tolerant systems. This is a complex task, and it requires specific knowledge and experience to write correct and consistent requirements.

The requirements defined in a natural language mostly imprecise and easy to misunderstand. However if we use a mathematical formalism (automatons, logical languages, etc.) the expression can be complicated, so it is difficult to understand, write and make modifications.

The goal of the solution described in this paper is to make this workflow easier besides that the precision is not decreasing. The experiences show that the safety requirements typically based on patterns. On this basis we can develop a method which helps to write complex requirements by composing and parameterizing the collected patterns.

For this it was necessary to build a formalism to define the patterns, create a method to parameterize them and write the set of rules of the composing process as theoretical research. The formalism supports the temporal logic requirements, context behavioural and time dependent expressions. For the complex requirements composed from the patterns, we defined a transformation to a precise, formal language. After this, the formalised requirements can be used to check plans or to synthesize monitors for run-time verification.

The tool created to implement this method contains the collection of the most often used patterns, and provides the possibility to create custom patterns and add it to the collection. Based on the defined formalism and set of rules it grants the graphical representation and parameterization of the patterns, and the graphical composing process. The tool fits into the Eclipse environment in order to make it easy to learn and easy to use in the development process.