DrawNet++: a Flexible Framework for Building Dependability Models

G. Franceschinis, M. Gribaudo, M. Iacono, V. Vittorini, C. Bertoncello

1. DrawNet++ quick overview and context

The DrawNet++ project addresses the compositional construction of dependability models [3, 4]. Its main goals are to provide: a) a GUI to any graph-based formalism; b) a support to the design process of dependability models, according to concepts inspired by object orientation (OO); c) a user friendly front-end for different classes of analysis/simulation tools.

Such features enforce the rapid prototyping and the reuse of sub-models. The formalisms adopted to implement the (sub)models may differ, by allowing to study both compositional problems related to the usage of a single specification formalism and interoperability issues within a multi-formalism environment. A prototype post-processor has been implemented to interface a few analysis/simulation tools and retrieve/show performance results. By now, Parametric Fault Tree (PFT) [1] and Stochastic Well Formed Nets (SWN) [2], are supported. An XML description of the user defined models is generated by the tool and XSL styles are used to translate the XML representation into a syntax suitable for the specific analysis tool. We use the DrawNet++ framework to build compositional SWN models and models of highly redundant systems by combining PFT with SWN [1]: since state space analysis tools are needed to represent dependencies between components, repair, transient faults etc, we model them by SWN sub-models and compose such predefined blocks with a SWN model of the system obtained by automatically translating a PFT model. In this context we use the DrawNet++ a) to define and draw composed models (e.g. PFT and SWN blocks); b) to produce the syntax representation of the entire model and c) to present the results computed by the solvers.

2. DrawNet++ demonstration

The goal of this demonstration is to present (1) the definition of a graph based formalism, including compositionality issues; (2) some guidelines for a dependability model construction cycle from PFTs to SWNs; (3) the OO capabilities that can be exploited in the model design process.

The first point addresses the XML specification of a graph based formalism and the automatic creation of a GUI to edit and manipulate (sub)models (Fig.1). The second point deals with the usage of XSL style sheets and the post-processing of the models in order to operate the conversion from a PFT model into a SWN model; this will be demonstrated by means of a case study [1]. Finally, the third point is aimed to show the mechanisms to define classes and objects that represent sets of system component models, create hierarchies, implement information hiding capabilities, and specify template objects.

References