

Software model checking is a process of checking for properties of a software application and thus assuring the software reliability. It is still necessary to divide the software application into pieces that are checked separately; the whole application yields an enormous state space that is impossible to traverse in a reasonable time. Therefore, the use of components is a straightforward approach for dividing the entire application. Furthermore, as model checker usually works with a close code only, a suitable component environment need to be provided for each component. Behavior protocols are a method for component behavior specification. They allow for checking for components' behavior compatibility and compliance, used at the design time of an application to find possible architectural component misplacements. They are also suitable to be compared with the behavior of the component implementation. The goal of the thesis is to provide a tool for comparing a primitive component behavior with its specification. Furthermore, a component environment generator using the component frame protocol will be implemented that would enable for checking for violation of the component behavior specification. The Java PathFinder model checker and the Fractal component model are to be used as a model checking platform.