Formal verification of behavior of a component application requires a suitable specification language. It is necessary that the specification language captures all important aspects of the future implementation with respect to desired properties. Behavior Protocols have been proven to be a suitable component behavior specification platform if one is interested in absence of communication errors. In this thesis, we (1) propose a new specification language based on Behavior Protocols and (2) address the issue of insufficient performance of BPChecker—a proprietary tool for verification of absence of communication errors in Behavior Protocols. Motivated by issues raised during specification of a real-life-sized case study aiming at providing wireless Internet access at airports, we extended the original Behavior Protocols with support for method parameters, local variables, synchronization of more than two components, and specification of variable-controlled loops. To address the second issue, we propose a method for verification of Behavior Protocols via their transformation to Promela—the input language of the Spin model checker.