

This thesis proposes and implements a new method of implementing Linux kernel drivers in userspace applications – userspace drivers. Instead of proposing fixed interface for accessing kernel functionality from userspace application and using dedicated kernel module to implement functionality not accessible via the interface, proposed method allows applications to define custom interfaces to the kernel. The interface is defined in special bytecode that is loaded into the kernel. The bytecode also provides functions that can be called from the kernel and that work even in atomic context (interrupt handlers, etc.). The bytecode is architecture and kernel configuration independent. The proposed method makes it possible to develop kernel drivers implemented in userspace applications without the need of dedicated kernel module.