

Prototypimplementierung der "Real-Time Data Access" (RTDA) Spezifikation 2.0 für den Java-Hardwarezugriff am Beispiel einer "Controller Area Network" (CAN) Schnittstellenkarte

Studienarbeit von Alexander Roth (1.10.2005 - 31.12.2005)

Das Konzept der Plattformunabhängigkeit von Java-Applikationen verhindert in vielen Bereichen leider auch den hardwarenahen Zugriff auf Ressourcen und Systeme. Insbesondere spezielle, oft wechselnde Hardware, z.B. im Umfeld eingebetter Systeme, verlangt die direkte Anbindung und somit Treiberentwicklung in Java. Mit Kapselung nativer Bibliotheken über die "Java Native Interface" (JNI) Programm-Bibliothek wird dies derzeit oft gelöst, aber trotz standardisierter Schnittstelle sind diese Lösungen uneinheitlich und eben nicht mehr plattformunabhängig.

Mit der Spezifikation für "Real-Time Data Access" (RTDA) hat das J-Consortium schon vor Jahren begonnen, diesem Problem mit einer abstrakten Java-basierten Schnittstelle zu begegnen. Mit der Version 2.0 vom Dezember 2003 wurde der 1. Ansatz vereinfacht und die notwendigen Teile nativen Programmcodes wurden in Ein-/Ausgabe-Stellvertretern gekapselt und weiter reduziert.

Da das J-Consortium und die federführenden Firmen, wie z.B. Siemens, leider nie eine öffentlich verfügbare prototypische Implementierung dieser Spezifikation umgesetzt haben will der Forschungsbereich Software Engineering (SE) des FZI Karlsruhe mit der vorliegenden Studienarbeit diese Lücke schließen. Die Studienarbeit stellt die Grundsätze der aktuellen RTDA-Spezifikation vor und entwickelt eine prototypische Implementierung, die wesentliche Konzepte umsetzt. Die Ausarbeitung diskutiert notwendige Vereinfachungen an der Spezifikation, bleibt aber für die vollständige Umsetzung offen. Zur Verifikation des vorgestellten Prototypen wird die Anbindung einer CAN-Schnittstellenkarte über RTDA demonstriert und die erreichten Vorteile gegenüber auf C und Java/JNI-Wrapper basierenden Ansätzen verdeutlicht.