

Generisches Testsystem für Betriebssysteme

Technische Informatik, Betriebssysteme / Betreuer: Prof. Roger Weber, Roland Eggli*, Peter Huber*
Projektpartner: *Triadem Solutions AG, Biel / Experte: Daniel Kühni

Ziel dieser Arbeit war es, ein generisches Testsystem zu entwerfen, welches plattformübergreifende Funktions- und Performancemessungen durchführt. Das Testsystem soll einen Vergleich unterschiedlicher Betriebssysteme und Hardware-Plattformen ermöglichen. Dabei stellt das Testsystem dem Nutzer eine Architektur zur Verfügung, welche ihm ermöglicht, seine eigenen individuellen Funktions- und Performancemessungen zu integrieren und durchzuführen. Die generierten Messresultate werden zentral verwaltet und dienen zur automatischen Erstellung von Testberichten.

Ausgangslage

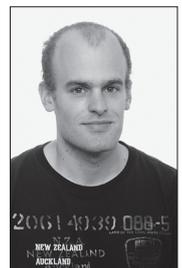
Ein Betriebssystem (OS – Operating System) ist für das Verwalten der System Ressourcen verantwortlich und bietet dem Programmierer eine Basis für die Entwicklung eigener Software. Auf dem Markt gibt es eine Vielzahl von Betriebssystemen mit den unterschiedlichsten Anwendungsgebieten. Da so viele verschiedene OS auf dem Markt sind, ist es oft schwierig, ein geeignetes Betriebssystem für sein Projekt auszuwählen. Als Entscheidungsgrundlage können hier Funktions- und Performancemessungen helfen. Bei diesen Messungen können die verschiedenen OS jedoch schlecht miteinander verglichen werden, da diese Zeiten Prozessor/Architektur abhängig sind. Des Weiteren ist das Durchführen von Funktions- und Performancemessungen zeitin-

tensiv und es gibt keine einheitliche Basis, welche die Vorbereitung und Durchführung dieser Messungen erleichtert. Aus diesen Gründen wurde im Rahmen dieser Arbeit ein generisches Testsystem entwickelt, mit welchem Funktions- und Performancemessungen auf unterschiedlichen Betriebssystemen und Prozessoren/Architekturen durchgeführt werden können.

Aufbau

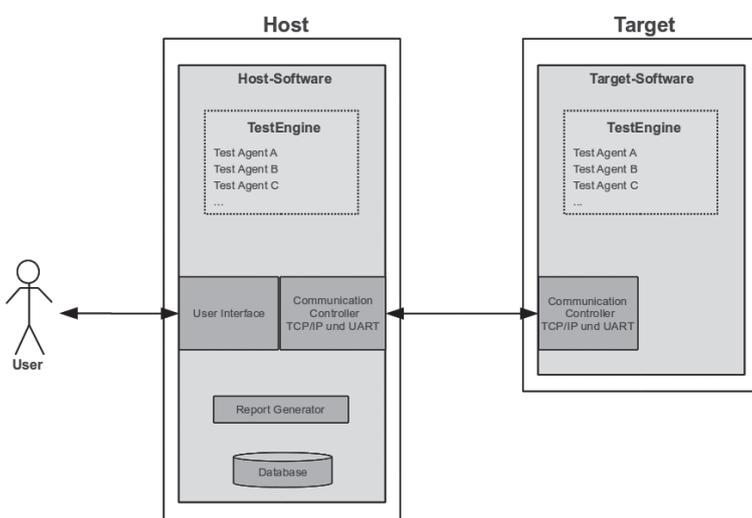
Das Testsystem besteht im Wesentlichen aus zwei Komponenten: Dem Host und dem Target. Über die grafische Oberfläche kann der User mit dem Host kommunizieren. Innerhalb der Testengine werden auf dem Host die unterschiedlichen Test Agents implementiert. Dabei entspricht ein Test Agent einer spezifischen Funktions- oder Performance-

messung. Test Agents können durch den User konfiguriert und gestartet werden. Die anfallenden Daten einer Messung werden in einer Datenbank zentral verwaltet und können falls gewünscht zu einem Testbericht zusammengestellt werden. Über TCP/IP oder eine serielle Schnittstelle kann der Host mit der zweiten Komponente, dem Target, im Testsystem kommunizieren. Auf dem Target werden nun die einzelnen Funktions- und Performancemessungen durchgeführt. Wird eine Messung durch den User gestartet, wird innerhalb des Host der entsprechende Test Agent aktiviert. Der Test Agent startet anschliessend über die Schnittstelle die entsprechende Funktions- und Performancemessung auf dem Target und speichert die ankommenden Messdaten ab.



Tim Wacher

tim.wacher@bluemail.ch



Zusammenfassung

Als Ergebnis steht dem Nutzer nun ein System zur Verfügung, welches er mit individuellen Test Agents ergänzen kann. Weiter wurde die Target Software auf zwei Betriebssysteme, QNX und GNU/Linux, portiert. Beide Betriebssysteme nutzen ein Beagleboard xM als Target-Hardware. Auf dem Host können Testsequenzen gestartet und die anfallenden Daten in der Datenbank zentral verwaltet werden.