

Smart Home & Building



Konzeption und Implementierung einer Build-Umgebung inkl. Continuous Build Server

Anforderung des Kunden

Für eine neue Produktlinie des Kunden sollte eine Build-Umgebung für Software-Komponenten und Applikationen erstellt werden. Die Build-Umgebung sollte so konzipiert sein, dass diese möglichst einfach in ein Continuous Build System integriert werden kann. Neben der Erstellung der Firmware für verschiedene Plattformen (Prototyp, Ziel-Hardware, Testumgebung), war die Unterstützung automatisierter Tests auf simulierter Hardware gefordert.

Außerdem sollten als weitere qualitätssichernde Maßnahme ein toolgestützter Review Prozess für den Quellcode implementiert werden. Dabei sollten mehrere bestehende Software Repositories eingebunden werden.



Lösung comlet

Es wurde eine konfigurierbare Make-/CMake-Umgebung erstellt, mit der die ausführbaren Softwarekomponenten, Applikationen und Tests erstellt werden. Zusätzlich wurden automatisierte Nightly Builds und Testruns auf einem Jenkins Server implementiert. Ziel war es unter anderem auch, dass der Jenkins Server stets aktuelle Entwickler-SDKs und die Firmware-Packages für die Zielplattform zur Verfügung stellt. Mit Google Test erstellte Unit Tests werden automatisiert in einer qemu-Umgebung ausgeführt. Bei Problemen werden die relevanten Personen automatisch benachrichtigt.

Zur Steigerung und Überwachung der Source Code-Qualität wurden zwei git-Repositories zur Durchführung webbasierter Reviews, in einen Review Board Server eingebunden.

Verwendete Technologien

Windriver Linux, Jenkins, make, CMake, bash, git, Review Board, Google Test, qemu

