Vorsorgeuntersuchung

Per Werkzeug automatisch überprüfen, ob Software die Code-Standards einhält

Jedes Kind lernt, dass Vorbeugen besser sei als Heilen. Nur in der Softwareentwicklung hat sich das noch nicht herumgesprochen: Statt ins Debugging sollten die Entwicklungsleiter besser in die Vermeidung von Fehlern investieren. Ein Appell für mehr Prävention. *Autor: Fergus Bolger*

Einstein sagte einmal: „Intellektuelle lösen Probleme, Genies aber vermeiden sie." Wer das beherzigen will, muss die heutigen Praktiken bei der Softwareentwicklung auf den Kopf stellen und den Ursachen von Programmierfehlern mehr Aufmerksamkeit widmen, statt sich auf das Finden von Fehlern in der Integrationsphase zu konzentrieren. Jede Organisation, in der Debugging-Zyklen den Großteil der Softwareengineering-Ressourcen verschlingen, dürfte dem zustimmen. Fehler vermeidet, wer vorab Standards für den Quellcode definiert und deren Einhaltung per CSE-Werkzeug sicherstellt (Coding Standard Enforcement – Überprüfung der Codestandard-Konformität).

Zum Glück gezwungen

Ein typisches Beispiel aus der Praxis der Firma PRQA (Programming Research) ist ein Unternehmen, das hunderte von Ingenieurstunden damit verbrachte, die Ursachen eines Speicherzugriffsfehlers zu ermitteln. Ein CSE-Audit vor Ort fand binnen einer halben Stunde die Ursache: inkonsistente Speicherzuweisungsmethoden in einem globalen Objekt, auf das verschiedene Systemmodule zugriffen. Der Fehler und sein zeitaufwändiges Debugging wären vermeidbar gewesen, wenn die Firma eine Reihe von Codierungsregeln befolgt hätte, die den Einsatz einer dynamischen Speicherzuweisung einschränken (ohne sie zu verbieten).

In einem anderen typischen Szenario sah es aus, als ob eine kürzlich erfolgte Umstellung auf 64-Bit-Hardware unberechenbares Systemverhalten verursacht hätte. Nach dem Einsatz von PRQA-Analysewerkzeugen zeigte sich, dass eine unerwünschte Konvertierung zwischen Pointer- und Integer-Datentypen die eigentliche Ursache war. Auch dieser Softwarefehler wäre mit Codierungsregeln vermeidbar gewesen.

Schwammiger Sprachstandard

Die ISO-Standards für C und C++ definieren nur etwa 70 Prozent der Programmiersprache ausreichend klar. Es gibt unzuverlässige Elemente beziehungsweise unsicheres Verhalten innerhalb des Codes. CSE schützt Softwareentwickler davor, Teile der Programmiersprache zu verwenden, die bekanntermaßen zu Problemen führen können. Dadurch lassen sich direkte Codierungsfehler, eine falsche Verwendung der Programmiersprache und unerwartete Verhaltensweisen vermeiden.

Code-Reviews, Geräte- und Systemtests einschließlich Abdeckungsanalyse, funktionale Verifikationsschritte, Feature-Aufrüstungszyklen – durch CSE lassen sich diese Arbeiten verkürzen. Das Ziel lautet: Fehler vermeiden oder zumindest erkennen und korrigieren, bevor sie in die Testumgebung gelangen. Ein präventionsorientiertes Vorgehen kann den Entwicklungsaufwand um bis zu 50 Prozent senken. Für manche Kunden ist CSE auch wichtig zum Einhalten von Sicherheits- oder Qualitätsstandards. Alternativ eignet sich CSE, um das weite Feld der Portabilitätsprobleme von C/C++ abzudecken, die an unterschiedlichen Compiler-Dialekten und Hardwareplattformen liegen. Nicht zuletzt hilft CSE, um Best Practices in der Softwareentwicklung durchzusetzen.

all-electronics.de
ENTWICKLUNG. FERTIGUNG. AUTOMATISIERUNG

Entdecken Sie weitere interessante Artikel und News zum Thema auf all-electronics.de!

Hier klicken & informieren!

Bild: Fotolia, Yvonne Bogdanski

Vorsorge ist besser als heilen: Quellcode-Analysen und Berichte sind mit QAC++ schnell erstellt. Sie helfen, Fehler schon vor dem Test zu finden oder sie ganz zu vermeiden.

Mit den CSE-Werkzeugen von PRAQ kann der Anwender bestehende Codierungsstandards übernehmen oder eigene Standards entwickeln und anschließend deren Durchsetzung automatisieren. CSE kann auch vertraglich vorgeschrieben sein, zum Beispiel wenn ein OEM seine Softwareentwicklung nach außen vergibt und dem Lieferanten ein Qualitätsniveau vorschreibt. Hier wirkt CSE als Eingangskontrolle und überprüft den gelieferten Quellcode in Bezug auf einen vorgeschriebenen Standard.

Um herauszufinden, wo Entwickler die Programmiersprache auf gefährliche Weise nutzen, können PRQA-Werkzeuge den Quellcode getrennt nach Dateien oder über ein gesamtes Projekt analysieren. Die Tools spüren auch Code-Stellen auf, die nicht mit dem jeweiligen ISO-Standard konform ist.

Bibliotheken für die Standardfälle

Die beiden Werkzeuge QAC und QAC++ nutzen umfassende Bibliotheken mit wählbaren Warnmeldungen. Diese markieren Quellcode-Abschnitte, die nicht portabel, schwierig zu warten, übermäßig komplex oder in irgendeiner Weise so geschrieben sind, dass sie Probleme heraufbeschwören. Ein Meldungsbrowser zeigt Diagnosedaten zum Quellcode und den zugehörigen Header-Dateien für das gesamte Projekt. Diese Diagnosen werden in Kategorien eingeteilt und den Quelldateien zugeordnet. Darüber hinaus lassen sie sich als Quellcode-Listing mit Anmerkungen darstellen. Möglich ist auch eine HTML-Darstellung mit Links zu zusätzlichen Informationen und Tipps.

Die Probleme mit dem Plus

Angesichts der zunehmenden Komplexität von Embedded-Softwaresystemen entscheiden sich immer mehr Firmen für C++ anstelle von C. Dabei erreichen sie eine viel höhere Abstraktionsebene als mit C und Assembler-Code. Mit diesen Veränderungen ist die Industrie bei der Sicherung der Softwarequalität aber an einem Wendepunkt angelangt. In den vergangenen drei Jahren konnte sich CSE verstärkt verbreiten, Debugging gilt dagegen zunehmend als Zeit- und Arbeitsverschwendung. Die Überwachung von Codierungsstandards oder das Prebugging gewährleisten einen viel effizienteren Ressourceneinsatz.

Unternehmen müssen mit diesen Tools nicht mehr ihren Intellekt zur Lösung von problematischen Fehlern einsetzen. Viel genialer ist es, Fehler durch den Einsatz von Coding Standard Enforcement zu vermeiden. *(lei)* ■

Der Autor: Fergus Bolger ist Chief Technical Officer bei Programming Research und blickt auf über 30 Jahre Erfahrung in der Hard- und Softwareindustrie zurück.

Auf einen Blick

Fehlervermeider

Durch die Implementierung von Best Practices bei den Codierungsstandards verkürzen Organisationen nicht nur ihre Time to Market, sondern verringern auch den Aufwand für Nacharbeiten und erneute Tests. CSE-Werkzeuge (Coding Standard Enforcement) sorgen dafür, dass jeder Quellcode diese Richtlinien einhält. Enthält die freigegebene Software weniger Bugs, so verbessert sich das Ansehen der Entwickler. Zugleich ermöglicht es eine bessere Kunden-Zufriedenheit, sowie ein geringeres Haftungsrisiko bei Systemausfällen.

infoDIREKT www.elektronikjournal.de 504ejl7009
Link zu Programming Research (PRQA)

VORTEIL Automatische Prüfung von Quellcode auf Einhaltung der Code-Standards vermeidet Fehler, die andernfalls kostspieliges Debugging erfordern würden.