



□ Stefan Harms

(stefan.harms@parasoft.com)

ist seit Sommer dieses Jahres für den Vertrieb von Parasofts Testautomatisierungsprodukten innerhalb Deutschlands, der Schweiz und Österreich zuständig. Er war vorher sechs Jahre für Parasoft an der Westküste der USA tätig, wo er schwerpunktmäßig mit Herstellern medizinischer Geräte, Firmen der Luft- und Raumfahrttechnik und Herstellern elektronischer Produkte im Bereich Software-Testautomatisierung, Fehlerbehebung und Arbeitsablauffizienz zusammenarbeitete.



□ Adam Trujillo

ist Fachredakteur bei Parasoft in Monrovia, Kalifornien. Er unterhält einen Blog auf <http://blog.parasoft.com/author/adam-trujillo>.

DevOps und die Notwendigkeit einer Entwicklungstestplattform

Die Rolle von statischer Analyse

DevOps verlangt von Softwareentwicklern immer häufiger, dass sie über das einfache Finden von Fehlern mit statischer Analyse hinausgehen und stattdessen dazu übergehen, bestehende Analysetechniken im Entwicklungsprozess zu überdenken. DevOps ist eine Ansammlung von Methoden und Praktiken, welche die notwendige Zusammenarbeit und Kommunikation über Abteilungsgrenzen hinweg fördert und somit den Entwicklungsprozess optimiert und beschleunigt. Ihre Wurzeln können auf die Entstehung iterativer Entwicklungsmethoden zurückgeführt werden, die eine neue Herangehensweise verlangen und Entwicklung, IT und Qualitätskontrolle kombinieren. DevOps hat somit das Potenzial, jene Engpässe zu beseitigen, die Entwicklungsorganisationen daran hindern, ihren Herausforderungen gerecht zu werden.

Um es vorwegzunehmen, es gibt keine kommerziell verfügbaren „DevOps-Tools“, die eine Organisation plötzlich in die Lage versetzen, „DevOps“ zu praktizieren. Es gibt jedoch vorhandene Mechanismen, DevOps-Strategien umzusetzen. In diesem Artikel zeigen wir, wie automatische statische Analysen im Entwicklungsprozess Organisationen dabei helfen, DevOps zu verwirklichen.

Automatisierte Feedbackschleife

DevOps fördert die Partnerschaft zwischen jenen Abteilungen, die Einfluss auf die Umsetzung von Anforderung haben. Durch Wissensaustausch und das Teilen von Aufgaben über Abteilungsgrenzen hinweg können Organisationen effizient den Entwicklungsprozess beschleunigen und gleichzeitig eine Verbesserung der Code-Qualität erreichen. Die Voraussetzung hierfür ist allerdings die Einführung

einer automatisierten Feedbackschleife, welche die konsistente Anwendung von Qualitätsvorgaben ermöglicht, angefangen bei der Anforderung über das Erstellen des Codes bis hin zur Produktion.

Die meisten Organisationen haben in der Regel bereits eine Feedbackschleife für die Implementierung von Anforderungen. Auf der einfachsten Ebene beginnt die Feedbackschleife bei den Vorgaben und arbeitet sich bis zur Implementierung, Messung, Post-Mortem-Analyse, Vorgabenänderungen und Re-Implementierung fort. Allerdings ist diese Schleife in den meisten traditionell geführten Organisationen ein umständlicher Prozess, der von strikt getrennten Abteilungen durchgeführt wird, die in der Regel wenig Kenntnis von den Vorgängen außerhalb ihres Bereiches haben.

Die Automatisierung ist daher von kritischer Bedeutung. Der Kostenaufwand ist

erheblich zu hoch, wenn Entwickler und Tester ihre eigentlichen Kernaufgaben unterbrechen, um Daten für den Feedbackprozess zu generieren. Ebenso birgt es erhebliche Risiken, diesen Prozess manuell durchzuführen, da diese Ergebnisse in der Regel nicht konsistent und mit erheblichen Fehlern behaftet sind. Das wiederum führt dazu, dass Organisationen oft nicht in der Lage sind, den Entwicklungsprozess und letztendlich die Qualität der Software zu verbessern.

Die Feedbackschleife kann durch die Einführung eines zentralen Integrationspunktes hergestellt werden, der all jene Komponenten einbezieht, die mit der Entwicklungs- und Testinfrastruktur im Zusammenhang stehen. Eine zentrale Plattform versetzt BTS (Bug Tracking System), RMS (Resource Management System), Versionskontrolle und andere Komponenten in die Lage, Prüfgegenstände zu sam-

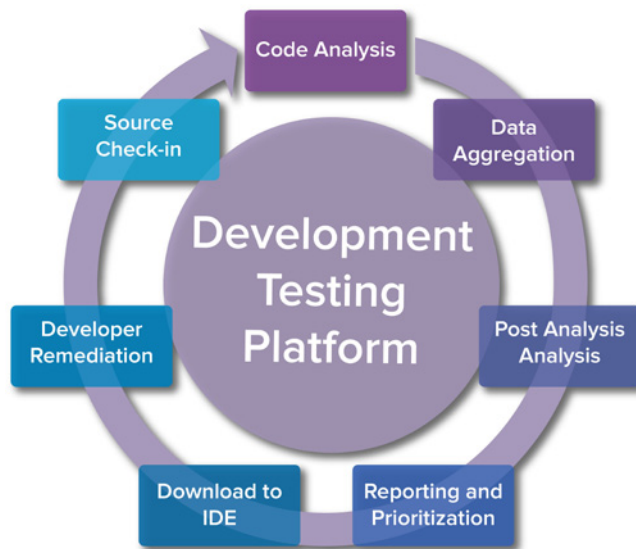


Abb. 1: Feedbackschleife

meln. Das DevOps-Team erhält so Zugriff auf verlässliche und konsistente Daten und wird dabei unterstützt, Fehler zu vermeiden. Es kann darüber hinaus den gesamten Entwicklungsprozess beschleunigen.

Entwicklungstests

In dem Maße, in dem die Implementierung von Anforderung voranschreitet, hilft die Feedbackschleife (siehe [Abbildung 1](#)) Entwicklern dabei, Fehler zu vermeiden, und fördert gleichzeitig das Verständnis für Veränderung durch den Reifeprozess dieser Anforderungen. Im Zusammenhang mit DevOps erlaubt die Feedbackschleife den Fluss von Analyse- und umgekehrt. Das operative Geschäft kann zu dieser Zusammenarbeit beitragen, indem es beim Ausformulieren einiger Aspekte von Entwicklungsvorgaben mithilft. Gleichzeitig trägt die Entwicklung dazu bei, durch die Umsetzung dieser Vorgaben Fehler zu vermeiden, die wiederum das operative Geschäft behindern können.

Diese Präventivphase des Entwicklungsprozesses ist jene Schwerarbeit, die den meisten Organisationen im Zusammenhang mit statischer Analyse vertraut ist:

- Bestimmen Sie, welche statische Analyseregeln gemäß den Entwicklungsvorgaben Ihrer Organisation angewandt werden (unter Berücksichtigung der Rückmeldung vom operativen Geschäft).
- Wenden Sie die Analyse an und adressieren Sie alle Fehler gemäß Ihren Vorgaben.

- Wiederholen Sie die Analyse solange, bis Sie ein für Sie akzeptables Fehlerniveau erreicht haben.

Vom DevOps-Standpunkt ist es während dieser Phase besonders wichtig, die Zeiteffizienz dieser Aktivitäten zu messen und zu analysieren. Organisationen müssen verstehen lernen, welche Funktionen des Qualitätsprozesses dabei helfen, ihre Geschäftsziele zu verwirklichen, und welche davon ablenken.

Das führt uns zur zweiten Phase von Entwicklungstests für DevOps: die Anwendung von Post-Mortem-Analysen in Bezug auf die Entwicklungsvorgaben. Die Ergebnisse, die während der Präventivphase ermittelt wurden, müssen nun nach Rang und Bedeutung geordnet und in den Entwicklungsprozess in Form von iterativen Vorgaben rückgeführt werden. Das hilft nachgeordneten Teams wie Qualitätskontrolle und IT nicht nur, ihre jeweiligen Aufgaben zu erfüllen, sondern auch dabei, Teil eines kontinuierlichen Prozesses zu werden, der in sicheren und qualitativ verlässlichen Code resultiert.

Prozessintelligenz für DevOps

Eine Entwicklungstestplattform erlaubt, all die Prüfgegenstände zu sammeln, die mit dem Entwicklungsprozess zusammenhängen, und sie miteinander in Beziehung zu setzen. Eine Infrastruktur, welche die Feedbackschleife automatisiert und Post-Mortem-Analyse durchführt, ist eine Minimalbedingung für DevOps. Eine hochintegrierte Testplattform weitet dieses Konzept noch aus, indem es Abteilungen in die Lage versetzt, zusätzliche Datensätze zu überlagern, die nicht nur bei der

Qualität der Software helfen, sondern auch auf Risiken hinweisen und Bereiche hervorheben, bei denen es der Verbesserung bedarf.

Diese Ebene der Prozessintelligenz treibt die Post-Mortem-Analyse erheblich durch die Aufdeckung von potenziell risikobehafteten Datenmustern an, die durch statische Analyse und andere Softwarequalitätsprozesse aufgezeigt wurden. Die Befunde werden schwerpunktmäßig geordnet und können in entsprechende Handlungen umgesetzt werden. Für DevOps stellt Prozessintelligenz das Hirn einer Entwicklungsplattform dar, welche die Feedbackschleife bildet.

Statische Analyse für DevOps

Daten sind das Futter, das eine automatisierte Feedbackschleife benötigt, um die Vorteile von DevOps zu verwirklichen. Zusätzlich zu Unit- und Regressions-Tests erzeugt die statische Analyse die notwendigen Daten zur Versorgung der Schleife mit Informationen, die aus der Post-Mortem-Analyse gewonnen wurden. Von diesem Blickwinkel erfüllt statische Analyse folgende Aufgaben:

- Qualitätssicherstellung während der Präventivphase,
- die Bereitstellung von Big Data, was für die Verbesserungen des Entwicklungsprozesses notwendig ist.

Entwicklungsorganisationen können statische Analyse in vielerlei Hinsicht einsetzen; in Bezug auf DevOps diskutieren wir zwei Formen, die sich in ihrer jeweiligen Funktion unterscheiden: das Finden von Fehlern vor einem Release und das Festigen des Codes an sich.

Ersteres ist eine Minimalanwendung von statischer Analyse, die von jeder Organisation durchgeführt werden sollte. Vom Gesichtspunkt der Verbesserung des Gesamtprozesses ist es jedoch von noch größerem Vorteil, jene Fehler zu untersuchen, die während des Release oder während der Qualitätskontrolle gefunden werden, um ein Wiederauftreten zu verhindern.

Hierbei geht es nicht darum, von den Vorteilen abzulenken, die statische Analyse zum Finden von Fehlern mit sich bringt, dennoch ist es unwahrscheinlich, dass Tester alle Auswirkungen eines Fehlers fänden, würde statische Analyse lediglich in dieser Form eingesetzt. Die Präventivanwendung statischer Analyse hingegen beinhaltet den Einsatz von Regeln, die auf

systemimmanente Fehler abzielen und nachgeordnete Defekte verhindern. Beispielsweise kann dynamische Testanalyse einen Datenfluss mit dem Ziel simulieren, unreine Daten in den Code zu injizieren. Das Anwenden spezifischer statischer Analyseregeln würde es Organisationen ermöglichen, dieses im Ansatz zu verhindern.

Vorgaben

Die Form der statischen Analyse, die eine Organisation unter DevOps-Gesichtspunkten einsetzt, hängt von ihren Entwicklungsvorgaben ab. Nicht jede Organisation schreibt sicherheitsrelevante oder betriebsnotwendige Software und nicht jeder Einsatz statischer Analyse muss daher auf das Stärken oder Festigen des Codes abzielen. Jedoch könnte DevOps ihre Vorgaben als Antwort auf die Post-Mortem-Analyse als Ergebnis des Durchlaufs der Feedbackschleife justieren.

Wenn eine Vorgabe beispielsweise bestimmt, dass eine bestimmte Klasse von Verletzungen während des Qualitätskontrollprozesses akzeptabel ist, diese Verletzungen aber im Zusammenhang mit schwereren Anwendungsversagen im Endnutzerbereich einhergehen, sollte eine Organisation zusätzliche Regeln aktivieren, mit dem Ziel, jene Fehler zu verhindern.

Wenn dieser Prozess innerhalb einer Entwicklungstestplattform stattfindet, die eine Prozess-Intelligenz beinhaltet, können noch zusätzliche Metriken angewandt werden, um Vorgaben weiter zu verfeinern, zum Beispiel Nennung des Autors des Codes oder der Codezeile, Verweigerung des Zugriffs auf betriebsnotwendigen Code durch unerfahrene Entwickler usw. In diesem Sinne formiert statische Analyse die Vorgaben und umgekehrt.

Zusammenfassung

Um es noch einmal zu verdeutlichen: DevOps kann nicht käuflich erworben und auf den Weg geschickt werden. Es ist eine Form des Schaffens von Software, die en-

gere Zusammenarbeit und Kommunikation über Abteilungsgrenzen hinweg fördert. Es ist das Ziel, von der alten Vorgehensweise abzukehren, die darin bestand, einfach die Finger zu kreuzen und zu hoffen, dass die Anwendung wie erwartet funktioniert, wenn sie an die nächste Abteilung weitergereicht wird. Diese Praxis muss sich unbedingt ändern.

Es sei jedoch darauf hingewiesen, dass es Strategien und Produkte gibt, die dabei helfen, DevOps anzunehmen. Statische Analyse - vor allem als Bestandteil einer Entwicklungstestplattform - kann dazu beitragen, das DevOps-Versprechen eines reibungslosen und schnelleren Release-Zyklus einzulösen. ■

Seit 28 Jahren erforscht und entwickelt Parasoft (www.parasoft.de) Lösungen, die Organisationen dabei helfen, fehlerfreie Software effizient abzuliefern. Durch die Integration von Development Testing (<https://www.parasoft.com/solution/development-testing/>), API/Cloud/SOA sowie Composite App Testing (<https://www.parasoft.com/solution/api-testing/>) und Service-Virtualisierung (<https://www.parasoft.com/solution/service-virtualization/>) helfen wir, Zeit, Aufwand und Kosten zu senken, um sichere, verlässliche und konforme Software zu liefern. Parasofts betriebsweite und eingebettete Entwicklungslösungen sind die umfassendsten in der Industrie, einschließlich statischer Analyse (<https://www.parasoft.com/capability/static-analysis/>), Unit Testing (<https://www.parasoft.com/capability/unit-testing/>), funktionalen Tests, Lasttests, Umgebungsmanagement (<https://www.parasoft.com/capability/test-environment-management/>) und mehr. Ungefähr 60 Prozent der Firmen im Fortune 500 verlassen sich auf Parasoft, um qualitativ hochwertige Software effizient und konstant herzustellen.