Site Reliability Engineering präzisiert die Messung der Servicequalität

Genauer hinschauen

Christoph Puppe

Site Reliability Engineering ist ein Ansatz von Google, über die Definition von Service Level Indicators (SLI), Service Level Objectives (SLO) und Fehlerbudgets die Servicequalität besser messbar und steuerbar zu machen. Das funktioniert mit Websites ebenso wie mit Datenbanken oder Speicherdiensten.

Eine möglichst hohe Verfügbarkeit ist bei Diensten, die für ein Unternehmen wichtig sind, unabdingbar. Ihre Messung ist allerdings komplexer als die Aussage, wie viele Minuten pro Jahr eine Anwendung auf Anfragen reagiert. Denn bei global verteilten Anwendungen kann es sein, dass die Nutzer in den USA fröhlich einkaufen, während die Nutzer in den Niederlanden den Warenkorb nicht öffnen können und in Indien jeder Mausklick eine sekundenlange Wartezeit nach sich zieht. Oder die Anwendung ist zwar verfügbar, liefert aber bei jeder zehnten Anfrage eine Fehlermeldung statt des erwarteten Ergebnisses. Andererseits macht es aus Nutzersicht keinen Unterschied, ob eine Website zu 99,999 Prozent oder nur zu 99,9 Prozent verfügbar ist – sie funktioniert halt „fast immer".

Ziel eines Monitorings sollte letztlich sein, zu erfahren, zu welchen Zeiten die Nutzer eine angenehme, schnelle und richtige Antwort auf ihre Anfragen erhalten – wann die Anwendung also benutzbar ist und wie schnell sie dabei ist. Zudem sollte das Monitoring eine Prognose erlauben, ob in Zukunft eine Verschlechterung der Servicequalität zu erwarten ist. Diese hehren Ziele sind mit einem reinen Blackbox-Ansatz nicht zu erreichen, der das Verhalten einer Anwendung von außen untersucht, da hier nur die Antworten der Anwendung verarbeitet werden, ohne in die Betriebsumgebung hineinzusehen. Whitebox-Monitoring wertet die Logfiles und Sensoren auf den Servern aus, um Probleme frühzeitig zu erkennen. Die Kombination beider Ansätze erlaubt einen umfassenden Einblick in die Anwendung.

So wie es wenig effizient ist, Fortschrittsbalken zu hypnotisieren, verspricht auch die Überwachung von Anwendungen nur dann Erfolg, wenn die Analyse automatisiert ist. Möglichst wenige Arbeitsschritte zur Behebung von Problemen dürfen Handarbeit erfordern und das Monitoring sollte nur dann einen Menschen benachrichtigen, wenn das Problem unbekannt und die Lösung noch nicht automatisiert ist. Ziel ist es, alle mehr als einmal auftretenden Fehlerzustände entweder durch Änderung der Anwendung oder durch Automatisierung der Reaktion zu beantworten.

Site Reliability Engineering

Site Reliability Engineering, kurz SRE, nennt Google einen neuen Ansatz für den Betrieb und die Überwachung seiner Dienste und Server. Im Zentrum steht da-

iX-TRACT

- Site Reliability Engineering ist ein neuer Ansatz von Google zur Überwachung von (Web-)Diensten.
- Service Level Indicators (SLI) und Service Level Objectives (SLO) legen fest, was genau gemessen wird und welche Qualität ein Dienst in diesen Werten einhalten muss.
- Definierte Fehlerbudgets schaffen Freiräume für neue Entwicklungen oder Kosteneinsparungen beim Betrieb.

Anteil der HTTP-Antworten pro Zeiteinheit in einer Grafik. Rot sind die Requests mit einer Fehlermeldung (20X/30X) als Antwort, grün die 40X- und ocker die 50X-Antworten. Die y-Achse ist logarithmisch, damit die sehr geringen Zahlen der Fehlermeldungen besser sichtbar sind. Aus diesen Metriken lassen sich Statistiken über beliebige Zeiteinheiten erstellen und als Grundlage für die Bewertung der Verfügbarkeit nutzen (Abb. 1).

bei eine neue Art, Verfügbarkeit zu definieren und zu messen.

Traditionell bezeichnet Verfügbarkeit den Prozentsatz der Zeit, in der eine Anwendung funktionsgemäß läuft. Die Messung erfolgt über einen externen Dienst, der eine Funktion der Anwendung aufruft und die Verfügbarkeit bestätigt, wenn die Antwort der Erwartung entspricht, oder den Ausfall feststellt, wenn die Antwort ausbleibt oder fehlerhaft ist. Eine solche Blackbox-Messung kann eine sehr tief greifende Prüfung beinhalten, die die Funktion aller Teile der Anwendung testet, oder lediglich auf die Schnelle prüfen, ob sich zu dem zuständigen Port auf dem Server eine Verbindung aufbauen lässt. Die Praxis liegt meist zwischen diesen beiden Extremen. Auch ist bei der Beauftragung externer Dienstleister nur selten vertraglich geregelt, was genau als verfügbar zu werten ist, auch wenn das vereinbarte Service Level Agreement Ziele vorgibt und deren Nichterreichung mit Strafen versieht.

Bei SRE geht das Monitoring einen anderen Weg. Grundannahme ist, dass die Anwendung zu nahezu 100 Prozent läuft und vollständige Ausfälle so selten sind, dass sie sowieso als Notfall betrachtet werden und nicht bloß als vorübergehende Störung. Unabhängig davon, ob die Anwendung in Dutzenden Rechenzentren global verteilt läuft oder nur auf einem Server: Gemessen wird bei SRE an erster Stelle die Zahl der Anfragen, die eine gültige Antwort erhalten. Bei einem Webserver sind dies die HTTP-Requests, die als Antwort 200 oder 304 erhalten. Alle Antworten mit 40X und 50X und alle Requests, die keine Antwort innerhalb einer festgelegten Zeitspanne erhalten, sind Fehler.

Dazu ist es notwendig, alle Requests zu erfassen und samt Antwortcode zu protokollieren. Bei einer Website kann das ein vorgelagerter Proxy als Messsystem erledigen. Die Statistik über dessen Logfiles belastet weder die Anwendung noch erzeugt sie einen Overhead, auch müssen die Programmierer der Anwendung keine speziellen Funktionen zur Überwachung der Anwendung implementieren und können sich mehr auf Kundennutzen und Qualität des Codes konzentrieren. Ein Proxy ist eine vergleichsweise einfache Komponente, deren Verfügbarkeit sehr leicht auf fast 100 Prozent zu bringen ist. Redundanzen und Lastverteilung machen weniger Aufwand als bei einer komplexen Anwendung.

Auf Basis der Logfile-Analyse des Proxys lässt sich Verfügbarkeit messen als Prozentsatz aller erfolgreichen Requests. Diese Zahl ist sehr viel näher an einer Messung der Nutzerzufriedenheit als ein simples Service-Monitoring mit einer Blackbox, das nur prüft, ob der Dienst antwortet: Sie gibt eine Auskunft darüber, ob die Nutzer den Service erhalten, den sie erwarten.

Ein genauerer Blick auf das Service Level

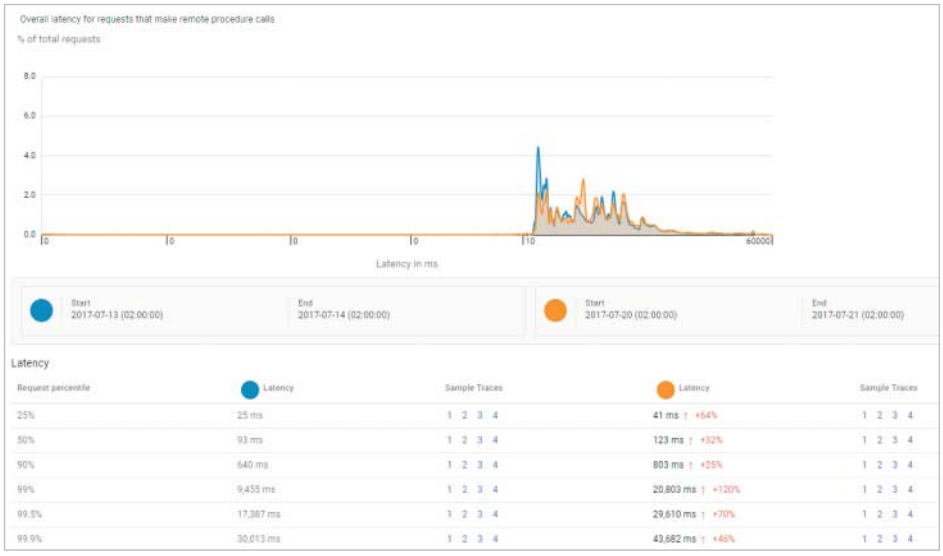
Jeder kennt den Begriff Service Level Agreement (SLA). SLAs fassen zumeist mehrere Aspekte zusammen: Sie definieren, was Verfügbarkeit bedeutet, wie sie gemessen wird und welche Abweichungen zu welchen Vertragsstrafen führen. SRE reduziert das SLA auf seine ursprüngliche Bedeutung als vertragliche Regelung zwischen Anbieter und Kunde über die Qualität des Service und führt zwei zusätzliche Begriffe ein: Service Level Indicator (SLI) und Service Level Objective (SLO).

SLI bezeichnet den zu betrachtenden Messwert. Das ist oft der Antwortcode der HTTP-Requests, kann aber auch die Latenz der Antwort, die Zahl der Requests pro Sekunde oder – eher für Datenbanken und Storage – der Durchsatz und die Dauerhaftigkeit der Speicherung sein. Auch wenn es möglich ist, Dutzende SLI zu messen, reichen meist einige wenige aus, um eine belastbare Aussage über die Qualität des Service und die Kundenzufriedenheit zu treffen. Daher ist die Auswahl weniger, möglichst aussagekräftiger SLI die Grundlage für eine aussagekräftige Messung.

Indikatoren, Ziele und Servicequalität

Das SLO ist das Ziel, das die Anwendung erreichen soll. Bei einer Webanwendung könnte das ein SLI für richtig beantwortete Anfragen mit einem SLO von 99,5 Prozent sein – nur 0,5 Prozent aller Anfragen dürfen einen Fehlercode zurückliefern.

Ein weiterer wichtiger Indikator vor allem bei zeitkritischen Diensten ist die



Die Verteilung der Antwortzeiten über eine Woche bei allen auf dem Proxy erfassten Anfragen. Eine Veränderung zwischen der ersten (blau) und der zweiten Messreihe (orange) hat die Latenzen insgesamt verschlechtert. Wurden zuvor 99 Prozent aller Anfragen in unter einer Sekunde beantwortet, sind es danach nur noch gut 90 Prozent der Anfragen (Abb. 2).

Latenz. Das SLI ist hierbei die Zeitdauer in Millisekunden, die die Antwort benötigt. Das SLO könnte für eine Webanwendung beispielsweise sein, dass sie 99 Prozent aller Anfragen in weniger als 100 ms beantwortet – nur 1 Prozent der Anfragen dürfen länger als 100 ms dauern. Damit ist allerdings noch kein Limit für sehr lang dauernde Anfragen gesetzt. Wenn das eine Prozent der Anfragen auf vielen Seiten der Website geschieht und sehr lang dauert, würden viele Anwender die Website dennoch als langsam erleben. Daher sollte ein zweites SLO festlegen, dass 99,9 Prozent aller Anfragen in unter einer Sekunde eine Antwort erhalten.

Um auch Ladezeiten im Browser zu messen, bietet sich ein Blackbox-Ansatz an, also eine externe Anwendung, die die Seite in einen Browser lädt und die Zeit misst, bis der Browser fertig ist. Auch die Korrektheit der zurückgelieferten Daten ist so besser überprüfbar. Das Messsystem vergleicht dabei die Antwort mit den als richtig definierten möglichen Antworten und meldet dies. Auch hierfür sind normalerweise keine zusätzlichen Funktionen in der Anwendung nötig, da die Blackbox vorhandene Funktionen abfragt.

Besser dank Fehlerbudgets

Auch wenn das Beispiel hier eine Webanwendung war, funktioniert diese Herangehensweise ebenso gut für andere Dienste wie Speicherlösungen, Datenbanken oder Verarbeitungspipelines. Passende SLIs und SLOs sind dann Latenz, richtig beantwortete Anfragen, Durchsatz der Pipelines und so weiter.

SRE geht davon aus, dass 100 Prozent Verfügbarkeit nicht erreichbar sind und – das ist das eigentlich Neue – dass dies auch gar nicht notwendig ist. Im Gegenteil: Das Streben nach hundertprozentiger Zuverlässigkeit wird als schädlich angesehen, denn ein Zwang zur Übererfüllung der Vorgaben für die Verfügbarkeit behindert Innovation. Was viele im Studium als „Mut zur Lücke“ kennengelernt haben, ist hier auf den Betrieb von Anwendungen angewendet.

Raum für Neues

Der Abstand zwischen dem SLO, also dem mit den Prozessinhabern und Kunden im SLA vereinbarten Ziel, und 100 Prozent ist das Fehlerbudget – eine Lücke, die nicht verschwinden soll, sondern die ihre Existenzberechtigung hat. In diesem Budget kann der Betrieb Kosten sparen und die Entwickler haben Freiheiten für neue Entwicklungen. So könnte beispielsweise die Geschäftsführung entscheiden, dass für eine neue Anwendung in einem sehr schnellen Markt ein SLO von nur 98 Prozent erfolgreicher Anfragen gut genug ist, weil es so möglich wird, schnell neue Features in die Produktion zu bringen. Während Villa Riba noch testet, verkauft Villa Bajo schon ein neues Feature an einen immer schnelleren Markt, der den Unterschied in der Verfügbarkeit kaum merkt.

Fehlerbudgets liegen in der Verantwortung von Entwicklern und Administratoren. Diese gemeinsame Verantwortung bringt zwei Gruppen zusammen, die sonst gerne im Clinch sind: Die Entwickler wollen neuen Code veröffentlichen, um ihre Ziele für die Weiterentwicklung der Anwendung zu erreichen. Die Administratoren möchten hingegen kein Risiko eingehen und daher am liebsten nichts ändern, um ihr Ziel eines unterbrechungsfreien Betriebs zu erreichen.

Da Fehlerbudgets pro Anwendung gelten, also unabhängig davon, wo der Fehler entstand, sind alle betroffen, wenn das Budget verbraucht ist. So sind beide Gruppen daran interessiert, dass die Anwendung innerhalb des Budgets bleibt: Das Fehlerbudget belohnt Programmierer, die wenig Fehler machen, mit der Freiheit, viele Features ausrollen zu dürfen; und es belohnt Administratoren, deren Betrieb läuft, mit wenig Arbeit.

Fehlerbudgets erlauben es so, Ziele zu setzen, die sich am Kundennutzen orientieren. Zudem reduzieren sie Stress, denn es gibt keinen Druck, die gesetzten Ziele zu übertreffen.

Fazit

Viele Organisationen scheitern an der Aufgabe, kritische Anwendungen präzise zu überwachen. Die notwendigen Ressourcen für zusätzliche Monitoring-Funktionen in Anwendungen sind oft nicht vorhanden oder die Entwicklung des Service-Monitoring geht zulasten der Produktentwicklung. Auch ist Verfügbarkeit in Zeiten preiswerter multipler Redundanzen keine Frage mehr von Minuten pro Jahr Stillstand, sondern es geht um die Messung der Kundenzufriedenheit in verteilten Anwendungen.

Der Ansatz des Site Reliability Engineering mit seiner präzisen Definition von SLI, SLO und SLA und der Nutzung von Fehlerbudgets erlaubt Anbietern wie Kunden eine klare Definition von Servicequalität, die Weiterentwicklung ermöglicht und die Qualität der Arbeit von Programmierern und Administratoren messbar macht. (odi)

Christoph Puppe

ist Security Consultant bei www.sva.de, seit 17 Jahren Vollzeit in der IT-Sicherheit aktiv, Audit-Teamleiter, IS-Revisor, Mitautor der Grundschutzkataloge und ehemaliger Penetrationstester.