Engineer

Fehlerbudget

der Fehleranteil, den ein Zuverlässigkeitsziel noch zulässt

Was es ist

Ein Fehlerbudget ist die aus dem Service Level Objective abgeleitete Menge an Fehlern, die ein Dienst in einem festgelegten Zeitraum haben darf. Verlangt das Ziel in diesem Quartal 99,99 % erfolgreiche Anfragen, sind die übrigen 0,01 % das Budget: Anfragen, die versehentlich oder durch bewusstes Risiko scheitern dürfen, ohne ein Versprechen zu brechen. Es wird in derselben Einheit wie das Ziel gezählt, meist als Zahl fehlgeschlagener Anfragen, und verbraucht sich, sobald Nutzer einen Fehler erleben.

Die Idee stammt aus Googles Buch Site Reliability Engineering. Im Kapitel über den Umgang mit Risiko legt das Produktmanagement das Ziel fest. Das Monitoring misst als neutraler Dritter, was tatsächlich geschah. Die Differenz ist die verbleibende Unzuverlässigkeit für das Quartal. Produktentwicklung und SRE vereinbaren dieses Budget vorab gemeinsam. Das Buchbeispiel liegt am anspruchsvollen Ende: Ein Dienst, der 99,999 % seiner Abfragen erfolgreich beantworten muss, hat ein Budget von 0,001 %. Ein einziges Problem mit 0,0002 % fehlgeschlagenen Quartalsabfragen hat davon ein Fünftel verbraucht.

Das Budget fragt nicht nach der Schuld. Laut Buch beanspruchen ein Netzwerkausfall oder ein ausgefallenes Rechenzentrum das Budget genauso wie ein fehlerhaftes Release. Im Rest des Quartals können dadurch weniger Bereitstellungen möglich sein. Nutzer spürten den Fehler, also zählt er. Wer darauf reagieren muss, ist eine andere Frage. Die Antwort gehört in eine vorab vereinbarte schriftliche Richtlinie.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist ein Fehlerbudget?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Monitoring und Alarme melden sich, solange ein Fehler noch klein ist; Kennzahlen zeigen, wie oft das gelingt.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. Das erste Observability-Dashboard aufgebaut — der erste Einblick des Unternehmens in die eigenen Systeme

Das erste Observability-Dashboard aufgebaut Kontakt aufnehmen →

Warum es wichtig ist

Die Menschen, die Funktionen ausliefern, und die, die einen Dienst betreiben, wollen Unterschiedliches. Ohne gemeinsame Zahl gewinnt, wer besser verhandelt. Laut SRE-Buch sollten solche Entscheidungen weder auf Politik noch Angst oder Hoffnung beruhen. Es erinnert an das inoffizielle Motto von Google SRE: „Hoffnung ist keine Strategie." Das Budget ersetzt den Streit durch eine vorher von beiden Seiten akzeptierte Zahl. Solange Budget bleibt, gehen Releases hinaus. Ist es verbraucht, pausieren sie, und die Arbeit fließt in Tests und Widerstandsfähigkeit. Das Buch beschreibt auch sanftere Varianten, etwa seltenere Releases oder Rollbacks bei fast leerem Budget. Eine Fußnote nennt die reine An-Aus-Variante Bang-Bang-Regelung.

Interessanter ist die Verhaltensänderung. Bei großem Budget können Entwickler Risiken eingehen: häufiger bereitstellen, ein Experiment versuchen. Bei fast leerem Budget verlangen sie laut Buch selbst mehr Tests und seltenere Bereitstellungen, weil niemand die eigene Freigabe blockieren will. Das Team reguliert sich selbst. Das Buch knüpft eine Bedingung daran: SRE muss tatsächlich befugt sein, Freigaben nach Verletzung des Ziels zu stoppen.

Das Budget steuert vor allem Releases, weil diese die meisten Störungen verursachen. Die Beispielrichtlinie des SRE Workbook schreibt Änderungen ungefähr 70 % der Ausfälle bei Google zu. Das Budget hilft deshalb, den Arbeitseinsatz zu bestimmen. Das Workbook-Kapitel über SLO-Einführung rechnet einen Dienst mit 97 % Ziel und einem Vierwochenbudget von 109.897 fehlgeschlagenen Anfragen durch. Ein schlechtes API-Release, bei dem vier Stunden lang jede Anfrage scheiterte, kostete 14.066 davon, etwa 13 %. Ein ausgefallener Datenbankserver mit 20 Stunden Wiederherstellung aus der Sicherung kostete geschätzt 72.000 oder 65 %. Der Serverausfall sieht viel schlimmer aus. Das Beispielunternehmen erlebt aber einen solchen in fünf Jahren und zwei oder drei schlechte Releases pro Jahr. Im Mittel kosten die Releases daher ungefähr doppelt so viel Budget. Die Verbesserung des Releaseprozesses zahlt sich zuerst aus.

Auch ein nie verbrauchtes Budget sagt etwas aus. Laut Buch zeigt es die Kosten eines zu hohen Ziels in Form von Unflexibilität und langsamer Innovation. Ein Team mit Schwierigkeiten beim Ausliefern kann sein Ziel lockern und damit das Budget vergrößern.

Wie es in der Praxis aussieht

Beim Verbrauch zählen sowohl die Geschwindigkeit als auch die Gesamtmenge. Das Workbook-Kapitel über Alarmierung nennt die Geschwindigkeit Burn-Rate: wie schnell der Dienst sein Budget im Verhältnis zum Ziel verbraucht. Bei einer Burn-Rate von 1 ist das Budget genau am Ende des Zeitfensters aufgebraucht. Für 99,9 % Ziel über 30 Tage ergibt seine Tabelle:

```
Burn-Rate   Fehlerquote   Budget aufgebraucht nach
1           0,1 %         30 Tagen
2           0,2 %         15 Tagen
10          1 %           3 Tagen
1.000       100 %         43 Minuten
```

Nach Burn-Rate zu alarmieren ist besser, als bei jedem Überschreiten von 0,1 % Fehlern jemanden zu wecken. Das würde jede kurze Spitze melden. Das Workbook empfiehlt mehrere Burn-Raten zugleich, jeweils über ein langes und ein kurzes Fenster. Sein Ausgangspunkt für 99,9 % Ziel weckt jemanden, wenn 2 % des 30-Tage-Budgets in einer Stunde verbraucht werden (Burn-Rate 14,4) oder 5 % in sechs Stunden (Burn-Rate 6). Es eröffnet ein Ticket bei 10 % in drei Tagen (Burn-Rate 1). Das kurze Fenster mit einem Zwölftel der Länge lässt den Alarm bald nach dem Problem enden. Im Workbook-Beispiel liegt der Kurzfenstermittelwert fünf Minuten nach Fehlerende wieder unter der Schwelle. Der Stundenmittelwert bleibt eine volle Stunde darüber.

Als Prometheus-Regeldatei lauten die beiden Bedingungen für die Alarmierung der Rufbereitschaft wie folgt. Die Verhältnisse sind Recording Rules mit Namen nach dem Workbook. Jede enthält den Anteil fehlgeschlagener Anfragen über das im Namen genannte Fenster:

```
groups:
  - name: orders-error-budget
    rules:
      - alert: OrdersErrorBudgetBurn
        expr: |
```

```
        (
          job:slo_errors_per_request:ratio_rate1h{job="orders"}
            > (14.4 * 0.001)
          and
          job:slo_errors_per_request:ratio_rate5m{job="orders"}
            > (14.4 * 0.001)
        )
        or
        (
          job:slo_errors_per_request:ratio_rate6h{job="orders"}
            > (6 * 0.001)
          and
          job:slo_errors_per_request:ratio_rate30m{job="orders"}
            > (6 * 0.001)
        )
      labels:
        severity: page
```

Verfolgen Sie eine fehlerhafte Bereitstellung. Um 14:00 lässt ein Release 2 % der Bestellanfragen scheitern, eine Burn-Rate von 20. Das Fünfminutenverhältnis überschreitet die 1,44-%-Grenze innerhalb von vier Minuten. Das Stundenverhältnis erreicht sie erst nach ungefähr 43 Minuten Fehlerdauer. Die Rufbereitschaft wird daher gegen 14:43 alarmiert. Genau 2 % des Monatsbudgets sind dann weg. Das ist die Abwägung der Regel: Eine kurze Störung weckt niemanden, ein anhaltender Verbrauch bleibt nicht verborgen. Ein Rollback um 14:50 lässt das Fünfminutenverhältnis binnen zwei Minuten unter die Grenze fallen und beendet den Alarm. Das Stundenverhältnis allein hätte ihn bis etwa 15:07 aktiv gehalten.

Eine angemessene Alarmierung der Rufbereitschaft war auch das Anliegen hinter dem eigenen Fehlermonitoring mit Tracing des Unternehmens aus der seit 2025 laufenden Tätigkeit. Es erkennt Spitzen und Regressionen und hält zwischen Alarmen eine Wartezeit ein, damit eine fehlerhafte Bereitstellung niemanden vierzigmal weckt. Seine Alarmregeln waren auf diese Plattform statt auf eine allgemeine zugeschnitten. Eine seiner 11 Ansichten für den Betrieb ist den Service Level Objectives gewidmet.

Wo es schiefgeht

Fehlerzählung wird mit Schuldzuweisung verwechselt. Jeder von Nutzern erlebte Fehler verbraucht Budget. Nicht alles sollte aber die eigenen Releases stoppen. Die Workbook-Beispielrichtlinie erlaubt die weitere Arbeit an Funktionen, wenn ein unternehmensweites Netzwerkproblem den Ausfall verursachte oder der Dienst eines anderen Teams, das seine Releases schon

eingefroren hat. Verbrauchte der eigene Bug oder Ablauf das Budget, geht Zuverlässigkeit vor. Für Ausfälle durch die Abhängigkeit vom Dienst eines anderen Teams nennt das Kapitel Implementing SLOs zwei Denkschulen: weitermachen oder trotzdem einfrieren. Laut Kapitel macht das Einfrieren Nutzer zufriedener. Entscheiden Sie vorab, schreiben Sie die Wahl in die Richtlinie und benennen Sie, wer Streitigkeiten über die Berechnung entscheidet. Die Beispielrichtlinie lässt den CTO entscheiden.

Fehler landen in der falschen Kategorie. Ein fehlgeschlagener Lasttest oder abgewiesene Penetrationstestanfragen verbrauchen Budget ohne betroffene Nutzer. Die Beispielrichtlinie nennt falsch kategorisierte Fehler als Ursache unnötigen Budgetverlusts. Das Gegenteil ist schlimmer: Als harmlos eingeordnete echte Ausfälle halten den Dienst auf dem Papier im Ziel, während Nutzer leiden. Dieselbe Richtlinie schreibt bei Entdeckung Arbeit an der Zuverlässigkeit vor. Halten Sie nicht vom Ziel erfassten Verkehr aus dem Indikator heraus und prüfen Sie die Einordnung nach jedem Vorfall mit unplausiblem Budgetverbrauch.

Ein großer Vorfall wird bezahlt und vergessen. Ein Einfrieren kauft Zeit, kein Verständnis. Die Beispielrichtlinie verbindet beides: Ein einzelner Vorfall mit mehr als 20 % des Vierwochenbudgets erfordert eine Nachbesprechung mit mindestens einer Maßnahme höchster Priorität. Eine Ausfallklasse, die in einem Quartal mehr als 20 % des Budgets verbraucht, muss in der Planung des nächsten Quartals stehen.

Der Alarm kann nie auslösen. Schwellenwerte aus der 99,9-%-Tabelle passen nicht für andere Ziele. Laut Workbook verbraucht ein Totalausfall bei 90 % Verfügbarkeit in einer Stunde nur 1,4 % des Budgets. Eine Regel mit 2 % pro Stunde bleibt dabei stumm. Am anderen Ende ist bei einem monatlichen Ziel von 99,999 % das Budget durch einen Totalausfall in 26 Sekunden aufgebraucht, schneller als die Messwerterfassung vieler Monitoringsysteme. Stimmen Sie Fenster und Schwellen auf das Ziel ab. Akzeptieren Sie bei sehr hohen Zielen die Workbook-Folgerung: Nur der Entwurf schützt sie. Rollen Sie eine Änderung zuerst an 1 % der Nutzer aus. Derselbe Fehler lässt dann 43 Minuten bis zum Budgetende.

Bei wenig Verkehr alarmiert ein Fehler die Rufbereitschaft. Bei zehn Anfragen pro Stunde entspricht eine fehlgeschlagene Anfrage 10 % Fehlerquote in dieser Stunde. Bei einem Ziel von 99,9 % alarmiert diese eine Anfrage laut Workbook sofort die Rufbereitschaft. Sie verbraucht 13,9 % des 30-Tage-Budgets; der ganze

Monat erlaubt nur sieben Fehler. Das Workbook empfiehlt künstlichen Verkehr, mehrere kleine Dienste für das Monitoring zu einem zusammenzufassen oder das Produkt so zu ändern, dass eine fehlgeschlagene Anfrage weniger ausmacht.

Das Ziel wird erhöht, bevor der Dienst es erfüllen kann. Ein Team, das ein strengeres Ziel braucht und es sofort übernimmt, lebt im Dauernotfall mit ständig verbrauchtem Budget. Das Workbook empfiehlt stattdessen ein angestrebtes Ziel, das neben dem aktuellen gemessen und verfolgt wird. Die Richtlinie hält aber fest, dass es bis zur Bereitschaft des Dienstes keine Maßnahmen verlangt.

Weiterführende Lektüre

- Risiko annehmen — Kapitel 3 aus Googles Buch Site Reliability Engineering: Fehlerbudgets als datenbasierte Grundlage für Releaseentscheidungen.
- Alarmierung anhand von SLOs — das SRE Workbook über Burn-Raten, Mehrfensteralarme und Dienste mit wenig Verkehr.
- Beispiel einer Fehlerbudgetrichtlinie — eine vollständige Richtlinie: wann Releases stoppen, welche Ausfälle ausgenommen sind und wann eine Nachbesprechung nötig ist.
- SLOs einführen — ein Budget aus einem Ziel berechnen und damit zwischen Zuverlässigkeitsinvestitionen wählen.
- Alarmregeln — die Prometheus-Dokumentation zum oben verwendeten Regeldateiformat.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Bereitstellungsweg, die Betriebsrisiken und die Bereitschaftsverantwortlichen. Wir können eine Änderung mit Prüfungen, einem Wiederherstellungsweg und klaren Zuständigkeiten abgrenzen.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist ein Fehlerbudget?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/error-budget/