Engineer

Systemmätvärden

tal som ett system rapporterar om sig självt

Vad det är

Systemmätvärden är mätningar av en tjänst i drift, var och en tagen vid ett visst ögonblick och sparad som ett värde med en tidsstämpel och några etiketter, så att samma storhet kan läsas av igen en minut, en dag eller en månad senare. En namngiven storhet är ett mätvärde. Sparad som en följd av värden över tid blir den en tidsserie, och OpenTelemetry räknar mätvärden som en av sina signaler, bredvid spårningar och loggar. Antal besvarade förfrågningar, använt minne och jobb som väntar i en kö är typiska mätvärden.

Det som skiljer mätvärden från de andra signalerna är aggregeringen. En logg sparar en rad per händelse, och en spårning följer en enda förfrågan genom systemet. Ett mätvärde viker ihop många mätningar till några få statistiska tal över ett tidsfönster: ett antal, en summa, ett aktuellt värde, en fördelning över intervall. Dokumentationen för OpenTelemetry kallar resultatet aggregerad statistisk information, och det är både styrkan och begränsningen: ett mätvärde säger hur många förfrågningar som misslyckades den senaste minuten, och en logg säger vilka.

OpenTelemetry och Prometheus är överens om ett litet antal former, och att välja rätt form är det mesta av designarbetet:

- En **räknare** (”counter”) ackumulerar och går bara uppåt, som mätarställningen i en bil: besvarade förfrågningar, skickade byte, upptäckta fel. Prometheus tillåter ett undantag, en återställning till noll när processen startar om.
- En **upp-och-ned-räknare** (”up-down counter”), som OpenTelemetry kallar den, ackumulerar men kan också gå nedåt, t.ex. längden på en kö när arbete kommer in och lämnar den.

- En **mätare** (”gauge”) är ett aktuellt värde som läses av i samma stund som någon läser det, som bränslemätaren i samma bil: använt minne, en temperatur, pågående förfrågningar.
- Ett **histogram** sorterar mätningar i intervall och sparar deras antal och summa, så att det kan svara på hur många förfrågningar som tog mindre än en sekund.

Prometheus har också en summary, som räknar ut kvantiler inne i programmet. Datamodellen i OpenTelemetry behåller den bara som en äldre typ, eftersom sådana sammanfattningar inte alltid går att slå ihop på ett meningsfullt sätt.

Textformatet som Prometheus hämtar visar delarna tillsammans: en hjälprad, en typrad och sedan en rad per kombination av etiketter med namnet, etiketterna, värdet och en valfri tidsstämpel i millisekunder sedan epoken.

```
# HELP http_requests_total The total number of HTTP requests.
# TYPE http_requests_total counter
http_requests_total{method="post",code="200"} 1027 1395066363000
http_requests_total{method="post",code="400"} 3 1395066363000
```

Använd begreppet i ett beslut

Använd ”Vad är systemmätvärden?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Övervakning och larm som säger till medan ett fel fortfarande är litet, och siffrorna som visar hur ofta det förblir så.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Byggde företagets första observerbarhetsdashboard — företagets första överblick över de egna systemen

Byggde företagets första observerbarhetsdashboard Hör av er →

Varför det spelar roll

Mätvärden är billiga nog att sparas för allting. Minnet de kräver växer med antalet olika kombinationer av etiketter, inte med trafiken, och dokumentationen för OpenTelemetry säger det med just de orden, som en kontrast till loggar. En räknare som har ökats en miljard gånger är fortfarande ett enda tal. Det är det som låter ett team räkna varje förfrågan i stället för ett urval, och spara räkningarna länge nog för att jämföra den här månaden med den förra.

De gör diskussioner till mätningar. Om en server är stor nog, om en release gjorde svaren långsammare, hur snart en disk blir full: alla har ett svar i en serie som spelades in innan någon frågade. Larm byggs på samma tal, eftersom en larmregel är ett villkor över ett mätvärde som någon har bedömt vara värt att vakna för.

Mätvärden lever också längre än koden som skickar ut dem, så formen är värd att få rätt. Namngivningsguiden för Prometheus ber om en enhet och en storhet per mätvärde, i basenheter som sekunder och byte i stället för millisekunder eller megabyte, med enheten som ett suffix i plural och `_total` på ett ackumulerande antal. Namn som `http_request_duration_seconds` och `process_cpu_seconds_total` säger då redan på ytan vad de innehåller.

Hur summorna färdas spelar också in. Prometheus rapporterar kumulativa summor, så luckor från en insamling som fallerar då och då jämnas ut, till priset av minne hos avsändaren i proportion till dess kombinationer av etiketter. StatsD rapporterar deltan, alltså förändringen sedan förra rapporten, och flyttar därmed den kostnaden ut ur processen.

Så ser det ut i praktiken

Företagets egen värddator för publika tjänster har en kärna och 512 MB minne, varav ungefär 464 MB går att använda, och där blev minnet det argument som avgjorde besluten. Rapporteringen är pull-baserad: sju kommandon som läser av värddatorn och renderar Markdown, inte ändrar något och bara körs på begäran. När frågan kom upp om samma värddator på 512 MB också kunde bära en onion-tjänst, kom svaret från ett dokumenterat dygns uppmätta stickprov snarare än en åsikt: det lägsta uppmätta tillgängliga minnet var ungefär 310 MB av 464, swapanvändningen låg på 2,6 procent och processorn var i vila 99,7 procent av tiden.

De rapporterna behövde senare säga vad deras tal var värda. Sex insamlare som bara läser fick ange det de aldrig läste, när det stod klart att en blind insamling och en frisk server gav samma sida: en spegel vars daemon hade bytt loggformat gav samma nolla som en spegel som ingen besökte. Tröskelvärdena anpassades efter de 116 stickprov som metrikrapporten redan hade mätt, med de observerade intervallen registrerade bredvid gränserna, och rapporten skiljer på tre utfall: en överträdelse, ingen överträdelse och ej bedömd eftersom ingenting mätte den. Varje insamlare rapporterar sin körtid och sin blindhet, t.ex. lästa filer mot oläsbara filer, även när svaret är noll. Gränserna gäller dagsgenomsnitt, som

kan dölja en topp som varar i en minut. Dödmansgreppet som körs var femtonde minut kontrollerar tillgängligt minne och ledigt diskutrymme vid varje körning, men en kortvarig överträdelse mellan kontrollerna kan ändå förbli oupptäckt.

De mest användbara mätvärdena kommer ofta från problemet snarare än från maskinen. Anteckningen om ett biljettsläpp med begränsat lager samlar väntetider på lås, reservationer som löpt ut och avståndet mellan grindens räkning och databasens på en och samma dashboard. Den kör invarianten `available + held + sold = total` schemalagt, visar den i ett diagram och larmar vid varje avvikelse. Vart och ett av de talen kommer ur kunskapen om hur just den försäljningen går fel.

Var det går fel

Systemmätvärden går sällan sönder med ett brak. Grafen fortsätter att ritas, och talet på den betyder i tysthet något annat än det som namnet säger.

Ett värde som sjunker publiceras som en räknare. Sidan om mätvärdestyper hos Prometheus säger att en räknare inte ska användas för något som kan minska, t.ex. antalet processer som körs: det är en mätare. Frågefunktionerna tolkar varje nedgång i en räknare som en omstart och korrigerar för den, så ett värde som sjunker med avsikt visar sig som ökningar som aldrig hände.

Enheterna glider isär mellan tjänsterna. En tjänst rapporterar en varaktighet i millisekunder, en annan i sekunder, och en panel som jämför dem slår fel med en faktor tusen. Namngivningsguiden säger att ett mätvärde måste avse en enda enhet och bör använda basenheter. Att räkna om en gång, vid källan, är billigare än att komma ihåg skillnaden i varje fråga.

En summa över etiketterna betyder ingenting. Tumregeln i namngivningsguiden är att summan eller medelvärdet över alla etiketter på ett mätvärde ska vara meningsfullt. Guidens eget exempel: kapaciteten hos flera köer hör ihop i ett mätvärde, medan en kös kapacitet och dess aktuella längd inte gör det.

Det som aldrig mättes läses som noll. En insamlare som har slutat läsa sin indata lämnar en nolla, eller en rad som saknas, och den ser exakt ut som ett lugnt system. Datamodellen i OpenTelemetry har en flagga för en punkt utan

registrerat värde, dess motsvarighet till markören för inaktuella värden i Prometheus, eftersom data som saknas och värdet noll är olika fakta. Rapportera dem olika, och ge ”ej bedömd” ett eget utfall, som insamlarna ovan gör.

Fönstret döljer toppen. En mätare är ett stickprov vid ett ögonblick, och datamodellen i OpenTelemetry rapporterar bara det sista värdet i varje intervall, så en topp mellan två stickprov lämnar inga spår. Ett dagsgenomsnitt jämnar på samma sätt ut en minut vid taket. Välj fönstret efter det fel som måste synas, och håll en snabb kontroll bredvid en långsam när båda spelar roll.

Överskriden kardinalitetsgräns ger för lågt antal i en filtrerad fråga. SDK:t för OpenTelemetry håller som standard reda på högst 2 000 olika kombinationer av attribut per mätvärdesström och viker ihop alla ytterligare till en enda datapunkt märkt `otel.metric.overflow=true`. Summan förblir korrekt, men datapunkten för överskjutande kombinationer ersätter hela kombinationen och inte bara dess brusiga del. I dokumentationens eget exempel bär en räknare för förfrågningar attributen `url.path` och `success`, och när den har svämmat över räknar en fråga efter `success=false` för lågt, och ett larm på felfrekvensen som bygger på den kan sluta utlösas. Håll attributen till små, kända mängder och håll utkik efter markören för överskriden kardinalitetsgräns.

Läs mer

- OpenTelemetry: Metrics — instrumentsorter, aggregering och gränsen för kardinalitet.
- OpenTelemetry: Metrics data model — mätare, summor, histogram, temporalitet och flaggan för saknat värde.
- Prometheus: Metric types — counter, gauge, histogram och summary.
- Prometheus: Metric and label naming — basenheter, suffix för enheter och `_total`.
- Prometheus: Histograms and summaries — var kvantiler räknas ut, och vilka som går att kombinera.
- Prometheus: Exposition formats — textformatet rad för rad.



Ordlista

Välj utifrån era faktiska förutsättningar

Ta med driftsättningsvägen, driftsriskerna och dem som har jour. Vi kan avgränsa en förändring med kontroller, en väg tillbaka och tydligt ansvar.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är systemmätvärden?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/metrics/