Engineer

Métricas

Cifras que un sistema comunica sobre sí mismo

Qué es

Las métricas son mediciones de un servicio en ejecución, tomadas en un instante y conservadas como un valor con marca de tiempo y unas pocas etiquetas. Así puede volver a consultarse la misma magnitud un minuto, un día o un mes después. Una magnitud con nombre es una métrica; almacenada como una secuencia de valores a lo largo del tiempo, forma una serie temporal. OpenTelemetry cuenta las métricas entre sus señales, junto con trazas y logs. Son ejemplos habituales las solicitudes atendidas, la memoria en uso y los trabajos que esperan en una cola.

La agregación distingue las métricas de las demás señales. Un log conserva una línea por evento y una traza sigue una solicitud por el sistema. Una métrica resume muchas mediciones en unas pocas estadísticas dentro de una ventana temporal: un recuento, una suma, un valor actual o una distribución por intervalos. La documentación de OpenTelemetry llama al resultado información estadística agregada. Ahí están su fuerza y su límite: una métrica dice cuántas solicitudes fallaron en el último minuto; un log indica cuáles.

OpenTelemetry y Prometheus coinciden en unos pocos tipos; elegir el adecuado es la mayor parte del diseño:

- Un **contador** acumula y solo crece, como el cuentakilómetros de un coche: solicitudes atendidas, bytes enviados, errores observados. Prometheus admite una excepción: restablecerlo a cero cuando se reinicia el proceso.
- Un **contador ascendente y descendente**, según la terminología de OpenTelemetry, acumula, pero también puede bajar; por ejemplo, la longitud de una cola cuando entran y salen trabajos.
- Un **indicador**, o gauge, es un valor actual leído en el momento de la medición, como el indicador de combustible del mismo coche: memoria en uso, temperatura, solicitudes en curso.

- Un **histograma** clasifica mediciones en intervalos y conserva su recuento y suma; así puede responder cuántas solicitudes tardaron menos de un segundo.

Prometheus también tiene un resumen, que calcula cuantiles dentro del programa. El modelo de datos de OpenTelemetry solo lo conserva como tipo heredado, porque los resúmenes no siempre pueden combinarse con sentido.

El formato de texto que recoge Prometheus reúne las piezas: una línea de ayuda, otra de tipo y, después, una línea por combinación de etiquetas con el nombre, las etiquetas, el valor y una marca de tiempo opcional en milisegundos desde el inicio de la época:

```
# 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
```

Utilice el concepto en una decisión

Aplique «¿Qué son las métricas?» a una decisión real. Podemos ayudarle a relacionar el concepto con el sistema que está construyendo o comprando. Monitorización y alertas que avisan cuando un fallo aún es pequeño, y cifras que muestran con qué frecuencia se mantiene así.

Explore nuestros servicios Póngase en contacto →

Compruebe los supuestos detrás del término

Utilice la explicación y los trabajos enlazados para formular preguntas sobre su proyecto. Podemos examinar los supuestos con usted antes de elegir una herramienta. Primer panel de observabilidad de la empresa — la primera visión de la empresa sobre sus propios sistemas

Primer panel de observabilidad de la empresa Póngase en contacto →

Por qué importa

Las métricas son lo bastante baratas como para conservar mediciones de todo el sistema. La memoria que necesitan crece con las combinaciones distintas de etiquetas, en lugar de con el tráfico; la documentación de OpenTelemetry lo dice expresamente, en contraste con los logs. Un contador incrementado mil millones de veces sigue siendo un número. Eso permite contar todas las solicitudes, en lugar de una muestra, y conservar los recuentos lo suficiente para comparar este mes con el anterior.

Convierten discusiones en mediciones. Si un servidor tiene capacidad suficiente, si una versión ralentizó las respuestas o cuándo se llenará un disco: cada pregunta encuentra respuesta en una serie registrada antes de formularla. Las alertas se construyen con las mismas cifras: una regla de alerta es una condición sobre una métrica por la que alguien decidió que merecía la pena despertarse.

También sobreviven al código que las emite; conviene acertar con su estructura. La guía de nombres de Prometheus pide una unidad y una magnitud por métrica, con unidades básicas como segundos y bytes, en lugar de milisegundos o megabytes, la unidad como sufijo plural y `_total` para un recuento acumulativo. Nombres como `http_request_duration_seconds` y `process_cpu_seconds_total` indican así a simple vista qué contienen.

También importa cómo se transmiten los totales. Prometheus informa de totales acumulados: los huecos causados por fallos intermitentes de recogida quedan compensados, a cambio de memoria en el emisor proporcional a sus combinaciones de etiquetas. StatsD informa de deltas, el cambio desde el último envío, trasladando ese coste fuera del proceso.

Cómo se aplica en la práctica

El servidor de la empresa para sus servicios de acceso público es una máquina de un solo núcleo con 512 MB de memoria, unos 464 MB utilizables; allí, la memoria se convirtió en el argumento que resolvía decisiones. Los informes se obtienen a petición: siete órdenes leen el servidor y generan Markdown, sin cambiar nada y solo cuando se solicitan. Cuando se planteó si ese servidor de 512 MB también podía alojar un servicio onion, la respuesta vino de muestras registradas durante un día, en lugar de una opinión: en esas muestras, la memoria disponible nunca bajó de unos 310 MB de 464, el intercambio se mantuvo en el 2,6 % y el procesador estuvo inactivo el 99,7 %.

Después, esos informes tuvieron que explicar qué valor tenían sus cifras. Se hizo que seis recolectores de solo lectura dejasen constancia de lo que nunca leían al comprobar que una recogida ciega y un servidor sano producían la misma página: una réplica cuyo demonio había cambiado el formato del log daba el mismo cero que otra que nadie visitaba. Los umbrales se ajustaron a las 116 muestras ya medidas por el informe de métricas, anotando los intervalos observados junto a los límites. El informe distingue tres resultados: incumplimiento, ausencia de incumplimiento y no evaluado porque nada lo midió. Cada recolector comunica su duración y sus puntos ciegos, por ejemplo archivos leídos frente a ilegibles,

incluso si la respuesta es cero. Los límites se aplican a los promedios diarios, que pueden ocultar un pico de un minuto. El mecanismo de vigilancia que se ejecuta cada quince minutos comprueba en cada ejecución la memoria disponible y el espacio libre en disco, pero aún puede pasar inadvertido un incumplimiento breve entre comprobaciones.

Las métricas más útiles suelen proceder del problema, no del servidor. La nota sobre vender existencias limitadas sin sobreventa reúne en un panel las esperas por bloqueos, las reservas caducadas y la diferencia entre el recuento de la puerta de admisión y el de la base de datos. Ejecuta de forma programada la invariante `available + held + sold = total`, la representa y avisa al personal de guardia ante cualquier desviación. Cada cifra nace de conocer cómo puede fallar esa venta.

Dónde surgen los problemas

Las métricas rara vez fallan de forma ruidosa. El gráfico sigue dibujándose y el número pasa a significar, silenciosamente, algo distinto de lo que dice su nombre.

Se publica como contador un valor que baja. La página de tipos de Prometheus indica que no debe usarse un contador para nada que pueda disminuir, como el número de procesos en ejecución: eso es un indicador. Las funciones de consulta interpretan cualquier caída de un contador como un reinicio y la corrigen; un valor que baja deliberadamente aparece entonces como aumentos que nunca ocurrieron.

Las unidades cambian entre servicios. Uno informa de la duración en milisegundos y otro en segundos; el panel que los compara se equivoca por un factor de mil. La guía de nombres exige que una métrica corresponda a una sola unidad y recomienda las básicas. Convertir una vez, en origen, cuesta menos que recordar la diferencia en cada consulta.

La suma entre etiquetas no significa nada. La regla orientativa de la guía de nombres es que la suma o el promedio sobre todas las etiquetas de una métrica debería tener sentido. Su propio ejemplo: las capacidades de varias colas pertenecen a una misma métrica; la capacidad de una cola y su longitud actual, no.

Lo no medido se lee como cero. Un recolector que deja de leer su entrada deja un cero o una fila ausente indistinguibles de un sistema con poca actividad. El modelo de OpenTelemetry incluye una marca para un punto sin valor registrado, equivalente al marcador de datos obsoletos de Prometheus, porque la ausencia de datos y un cero son hechos distintos. Repórtelos de forma diferente y dé a «no evaluado» un resultado propio, como hacen los recolectores anteriores.

La ventana oculta el pico. Un indicador es una muestra de un instante; el modelo de OpenTelemetry solo comunica el último valor de cada intervalo, por lo que una ráfaga entre dos muestras no deja huella. Del mismo modo, un promedio diario diluye un minuto al límite. Elija la ventana según el fallo que deba verse y mantenga una comprobación rápida junto a otra lenta cuando ambas importen.

El desbordamiento hace que una consulta filtrada cuente de menos. El SDK de OpenTelemetry registra, de forma predeterminada, como máximo 2.000 combinaciones distintas de atributos por flujo de métricas y agrupa las siguientes en un único punto marcado como `otel.metric.overflow=true`. El total sigue siendo correcto, pero el desbordamiento sustituye la combinación completa, no solo su parte ruidosa. En el ejemplo de la documentación, un contador de solicitudes lleva `url.path` y `success`; al desbordarse, una consulta con `success=false` cuenta de menos y una alerta de tasa de errores basada en ella puede dejar de dispararse. Limite los atributos a conjuntos pequeños y conocidos y vigile la marca de desbordamiento.

Lecturas adicionales

- OpenTelemetry: métricas — tipos de instrumentos, agregación y límite de cardinalidad.
- OpenTelemetry: modelo de datos de métricas — indicadores, sumas, histogramas, temporalidad y marca de ausencia de valor registrado.
- Prometheus: tipos de métricas — contador, indicador, histograma y resumen.
- Prometheus: nombres de métricas y etiquetas — unidades básicas, sufijos de unidad y `_total`.
- Prometheus: histogramas y resúmenes — dónde se calculan los cuantiles y cuáles pueden combinarse.
- Prometheus: formatos de exposición — el formato de texto línea por línea.

Glosario

Elija según sus restricciones reales

Describa la ruta de despliegue, los riesgos operativos y quiénes se encargan de las guardias. Podemos definir el alcance de un cambio con comprobaciones, una vía de recuperación y responsabilidades claras.

Vea cuánto cuesta Póngase en contacto →

Haga práctica la terminología

Pregunte cómo se aplica «¿Qué son las métricas?» a su organización. Basta un ejemplo concreto de su sistema actual para iniciar la conversación.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

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

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/glossary/metrics/