

SRE Observability Framework

- [Summary](#)
- [Standard Metrics](#)
 - [Infra Layer Metrics](#)
 - [Network Layer Metrics](#)
 - [Application Layer Metrics](#)
 - [Database Layer Metrics](#)
 - [The Four Golden Signals](#)
 - [Latency \(How Long?\)](#)
 - [Traffic \(How Much?\)](#)
 - [Errors \(How Many Valid?\)](#)
 - [Saturation/Throughput \(How Full?\)](#)
- [Services to cover](#)
 - [Azure](#)
 - [Infra Layer](#)
 - [Network Layer](#)
 - [AWS](#)
- [Make It Observable](#)
- [DataDog](#)
 - [DataDog Integration](#)
 - [Azure](#)
 - [AWS](#)
 - [DataDog Dashboards](#)
 - [Create Dashboard](#)
 - [DataDog Alerts](#)
 - [DataDog Integration with WebEx Teams](#)
 - [WebHook creation in WebEx](#)
 - [WebHook creation in DataDog](#)
 - [Enabling WebHook notifications](#)

Summary

This document provides guidelines and standards for observability and adding monitoring metrics to cloud factory core services in all layers (Infra, Network, Application and Database).

Standard Metrics

Infra Layer Metrics

- CPU
- Memory
- I/O
- Disk Space
- VM status (Running, Stopped, Starting)

Network Layer Metrics

- Bandwidth usage. Bandwidth is the maximum data transmission rate possible on a network
- Throughput
- Latency
- Packet loss
- Retransmission
- Availability
- Connectivity

Application Layer Metrics

- User Satisfaction / Apdex Scores
- Average Response Time
- Error Rates
- Count of Application Instances
- Request Rate
- Application & Server CPU

- Application Availability
- Garbage Collection

Database Layer Metrics

- Database Throughput. Database throughput is one of the most important database performance metrics
- Database Response or Latency. Database Response is one of the most common database performance metrics
- Database Connections
- Number of errors
- Most Frequent Queries

The chosen monitoring tool is DataDog and this documents explains how we make our systems observable in order to be monitored using DataDog.

We will cover the 4 golden signals of SRE monitoring (Latency, Traffic, Error and Saturation)

The Four Golden Signals

The four golden signals shouldn't be considered the be-all and end-all of your monitoring. These signals represent the beginning of your monitoring and should always be included.

Latency (How Long?)

Latency measures the time it takes to successfully process a request. Failed authentication requests that respond within milliseconds don't count toward your overall latency number. Neither do validation errors that require the client to send a different request. This metric typically gets measured using milliseconds.

Traffic (How Much?)

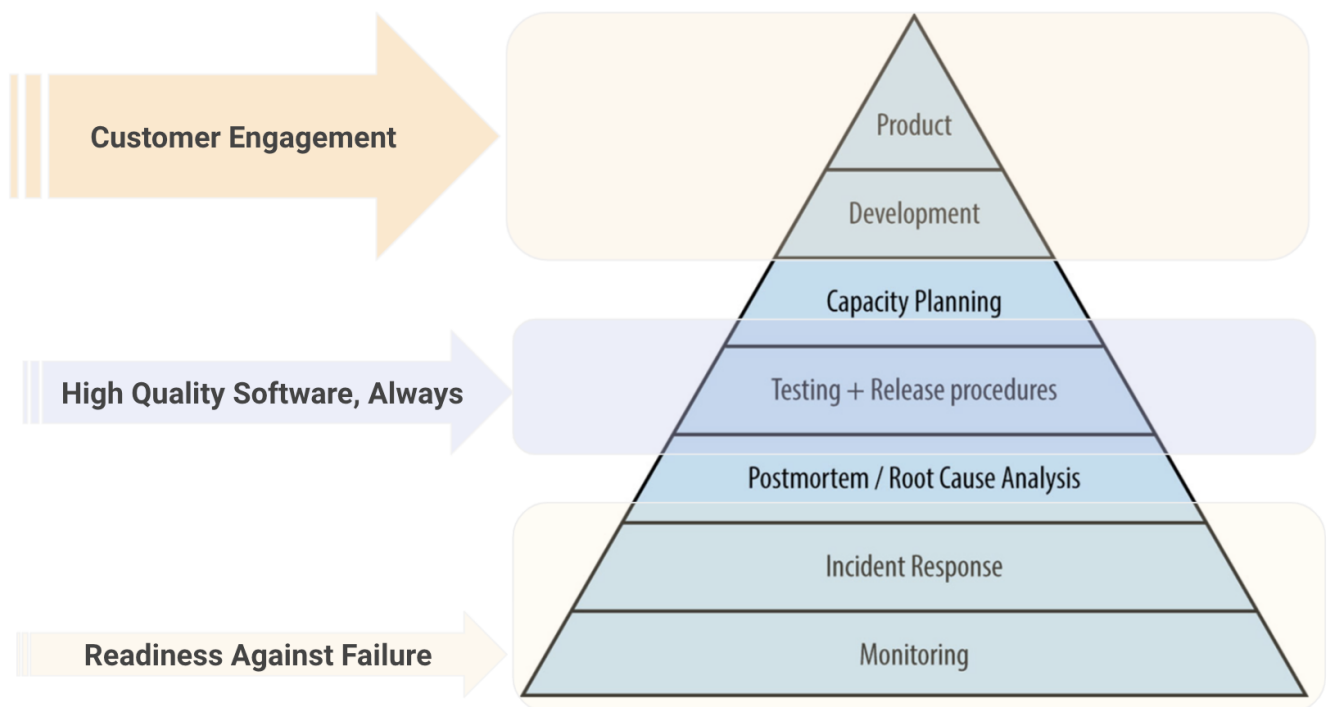
Traffic indicates how much demand exists on your system. For example, in web services, this metric measures things like requests per second (RPS). Or, for back-end remote procedure calls or database calls, this may be transactions per second (TPS). The way it's measured varies based on the type of system being tracked.

Errors (How Many Valid?)

A software system wouldn't be truly complete without error tracking. This golden signal tracks the rate of requests or transactions that fail. This includes blatant errors like HTTP 500 as well as responses that don't provide the correct data. It's also good if client errors and server errors report separately so that you're aware of client-based problems.

Saturation/Throughput (How Full?)

Saturation is one of the more complicated golden signals. It tracks how much of your capacity is being used. For example, saturation can include specific metrics on CPU, memory, and disk space utilisation. It works as an early warning indicator of system failures or slowdowns.



Services to cover

Azure

Virtual Machine

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-vm-monitor>

[Azure Virtual Machine monitoring](#)

Infra Layer

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.vm.cpu_credits_consumed (count)	Total number of credits consumed by the Virtual Machine	SATURATION	✓
azure.vm.cpu_credits_remaining (count)	Total number of credits available to burst	SATURATION	✓
azure.vm.data_disk_qd_deprecated (count)	Data Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.vm.data_disk_queue_depth_preview (count)	Data Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.vm.data_disk_read_bytes_sec_deprecated (count)	Bytes/Sec read from a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.vm.data_disk_read_bytes_sec_preview (count)	Bytes/Sec read from a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.vm.data_disk_read_operations_sec_deprecated (count)	Read IOPS from a single disk during monitoring period <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.data_disk_read_operations_sec_preview (count)	Read IOPS from a single disk during monitoring period <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.data_disk_write_bytes_sec_deprecated (count)	Bytes/Sec written to a single disk during monitoring period <i>Shown as byte</i>	SATURATION	✗
azure.vm.data_disk_write_bytes_sec_preview (count)	Bytes/Sec written to a single disk during monitoring period <i>Shown as byte</i>	SATURATION	✗

azure.vm.data_disk_write_operations_sec_deprecated (count)	Write IOPS from a single disk during monitoring period <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.data_disk_write_operations_sec_preview (count)	Write IOPS from a single disk during monitoring period <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.disk_read_bytes (count)	(ARM VM only) Amount of bytes read <i>Shown as byte</i>	SATURATION	✗
azure.vm.disk_read_bytes_sec (gauge)	(Classic VM only) Amount of bytes read per second <i>Shown as byte</i>	SATURATION	✓
azure.vm.disk_read_operations_sec (gauge)	(ARM VM only) Amount of read operations per second <i>Shown as operation</i>	TRAFFIC	✓
azure.vm.disk_write_bytes (count)	(ARM VM only) Amount of bytes written <i>Shown as byte</i>	SATURATION	✗
azure.vm.disk_write_bytes_sec (gauge)	(Classic VM only) Amount of bytes written <i>Shown as byte</i>	SATURATION	✓
azure.vm.disk_write_operations_sec (gauge)	(ARM VM only) Amount of write operations per second <i>Shown as operation</i>	TRAFFIC	✓
azure.vm.inbound_flows_maximum_creation_rate (count)	The maximum creation rate of inbound flows (traffic going into the VM) <i>Shown as item</i>	SATURATION	✗
azure.vm.inbound_flows (count)	Inbound Flows are number of current flows in the inbound direction (traffic going into the VM) <i>Shown as item</i>	SATURATION	✗
azure.vm.memory_available_bytes (gauge)	(Windows) Pool size of available pages in RAM that the system uses to satisfy requests for new pages <i>Shown as byte</i>	SATURATION	✗
azure.vm.memory_available_swap (gauge)	(Linux) Available swap <i>Shown as byte</i>	SATURATION	✓
azure.vm.memory_cache_faults_per_sec (gauge)	(Windows) Rate at which pages sought in the cache were not found there and had to be obtained elsewhere in memory or on the disk <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_committed_bytes (gauge)	(Windows) Amount of committed virtual memory <i>Shown as byte</i>	SATURATION	✓
azure.vm.memory_page_faults_per_sec (gauge)	(Windows) Overall rate at which the processor handles both hard and soft page faults. <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_page_reads_per_sec (gauge)	(Windows) Rate at which the disk is read to resolve hard page fault <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_pages_per_sec (gauge)	(Windows+Linux) Rate at which pages are read from or written to disk to resolve hard page faults <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_pages_read_per_sec (gauge)	(Linux) Rate of pages read to resolve hard page fault <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_pages_written_per_sec (gauge)	(Linux) Rate of pages writes <i>Shown as page</i>	SATURATION	✗
azure.vm.memory_pct_committed_bytes_in_use (gauge)	(Windows) Ratio of Memory \ Committed Bytes to the Memory \ Commit Limit. <i>Shown as percent</i>	SATURATION	✗
azure.vm.memory_percent_available_memory (gauge)	(Linux) Percentage of available memory available <i>Shown as percent</i>	SATURATION	✓
azure.vm.memory_percent_available_swap (gauge)	(Linux) Percentage of available swap available <i>Shown as percent</i>	SATURATION	✓
azure.vm.memory_percent_used_by_cache (gauge)	(Linux) Percentage of used memory per cache <i>Shown as percent</i>	SATURATION	✓
azure.vm.memory_percent_used_memory (gauge)	(Linux) Percentage of available memory used <i>Shown as percent</i>	SATURATION	✓
azure.vm.memory_percent_used_swap (gauge)	(Linux) Percentage of available swap used <i>Shown as percent</i>	SATURATION	✓

azure.vm.memory_pool_nonpaged_bytes (gauge)	(Windows) Size of the nonpaged pool <i>Shown as byte</i>	SATURATION	✗
azure.vm.memory_pool_paged_bytes (gauge)	(Windows) Size of the paged pool <i>Shown as byte</i>	SATURATION	✗
azure.vm.memory_transition_faults_per_sec (gauge)	(Windows) Rate at which page faults are resolved by recovering pages without incurring additional disk activity <i>Shown as fault</i>	SATURATION	✗
azure.vm.memory_used_memory (gauge)	(Linux) Used memory <i>Shown as byte</i>	SATURATION	✓
azure.vm.memory_used_swap (gauge)	(Linux) Used swap <i>Shown as byte</i>	SATURATION	✓
azure.vm.os_disk_qd_deprecated (count)	OS Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.vm.os_disk_queue_depth_preview (gauge)	OS Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.vm.os_disk_read_bytes_sec_deprecated (count)	Bytes/Sec read from a single disk during monitoring period for OS disk <i>Shown as byte</i>	SATURATION	✗
azure.vm.os_disk_read_bytes_sec_preview (count)	Bytes/Sec read from a single disk during monitoring period for OS disk <i>Shown as byte</i>	SATURATION	✗
azure.vm.os_disk_read_operations_sec_deprecated (count)	Read IOPS from a single disk during monitoring period for OS disk <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.os_disk_read_operations_sec_preview (count)	Read IOPS from a single disk during monitoring period for OS disk <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.os_disk_write_bytes_sec_deprecated (count)	Bytes/Sec written to a single disk during monitoring period for OS disk <i>Shown as byte</i>	SATURATION	✗
azure.vm.os_disk_write_bytes_sec_preview (count)	Bytes/Sec written to a single disk during monitoring period for OS disk <i>Shown as byte</i>	SATURATION	✗
azure.vm.os_disk_write_operations_sec_deprecated (count)	Write IOPS from a single disk during monitoring period for OS disk <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.os_disk_write_operations_sec_preview (count)	Write IOPS from a single disk during monitoring period for OS disk <i>Shown as operation</i>	TRAFFIC	✗
azure.vm.outbound_flows_maximum_creation_rate (count)	The maximum creation rate of outbound flows (traffic going out of the VM) <i>Shown as item</i>	SATURATION	✗
azure.vm.outbound_flows (count)	Outbound Flows are number of current flows in the outbound direction (traffic going out of the VM) <i>Shown as item</i>	SATURATION	✗
azure.vm.percentage_cpu (gauge)	Percentage of CPU resources used <i>Shown as percent</i>	SATURATION	✓
azure.vm.physical_disk_average_disk_queue_length (gauge)	Number of requests that are queued and waiting for a disk during the sample interval	TRAFFIC	✗
azure.vm.physical_disk_average_read_time (gauge)	Percentage of time that the selected disk drive is busy servicing read requests. <i>Shown as percent</i>	SATURATION	✓
azure.vm.physical_disk_average_transfer_time (gauge)	The average disk transfer time <i>Shown as percent</i>	LATENCY	✓
azure.vm.physical_disk_average_write_time (gauge)	Percentage of time that the selected disk drive is busy servicing write requests. <i>Shown as percent</i>	LATENCY	✓
azure.vm.physical_disk_bytes_per_second (gauge)	Rate of bytes <i>Shown as byte</i>	SATURATION	✗

azure.vm.physical_disk_read_bytes_per_second (gauge)	Rate at which bytes are read <i>Shown as byte</i>	SATURATION	✗
azure.vm.physical_disk_reads_per_second (gauge)	Rate of reads <i>Shown as read</i>	SATURATION	✓
azure.vm.physical_disk_total_disk_read_bytes_per_sec (gauge)	Rate at which bytes are read on all disks <i>Shown as byte</i>	SATURATION	✗
azure.vm.physical_disk_total_disk_write_bytes_per_sec (gauge)	Rate at which bytes are written on all disks <i>Shown as byte</i>	SATURATION	✗
azure.vm.physical_disk_transfers_per_second (gauge)	Rate of transfers	SATURATION	✗
azure.vm.physical_disk_write_bytes_per_second (gauge)	Rate at which bytes are written <i>Shown as byte</i>	SATURATION	✗
azure.vm.physical_disk_writes_per_second (gauge)	Rate of writes <i>Shown as write</i>	SATURATION	✗
azure.vm.premium_data_disk_cache_read_hit_preview (gauge)	Premium Data Disk Cache Read Hit <i>Shown as percent</i>	SATURATION	✗
azure.vm.premium_data_disk_cache_read_miss_preview (gauge)	Premium Data Disk Cache Read Miss <i>Shown as percent</i>	SATURATION	✗
azure.vm.premium_os_disk_cache_read_hit_preview (gauge)	Premium OS Disk Cache Read Hit <i>Shown as percent</i>	SATURATION	✗
azure.vm.premium_os_disk_cache_read_miss_preview (gauge)	Premium OS Disk Cache Read Miss <i>Shown as percent</i>	SATURATION	✗
azure.vm.process_total_handle_count (gauge)	Total number of handles set for a process by the system	SATURATION	✗
azure.vm.process_total_page_faults_per_sec (gauge)	Number of page faults per second <i>Shown as fault</i>	TRAFFIC	✗
azure.vm.process_total_pct_processor_time (gauge)	Percentage of elapsed time that all of process threads used the processor to execution instructions <i>Shown as percent</i>	SATURATION	✗
azure.vm.process_total_private_bytes (gauge)	Total amount of memory that a process has allocated, not including memory shared with other processes <i>Shown as byte</i>	SATURATION	✗
azure.vm.process_total_working_set (gauge)	Collection of those pages in a virtual address space that have been recently referenced for a given process. It includes both shared and private data	SATURATION	✗
azure.vm.process_total_working_set_private (gauge)	Collection of those pages in a virtual address space that have been recently referenced for a given process. It includes only private data	SATURATION	✗
azure.vm.processor_percent_dpc_time (gauge)	Percentage of the time in which the system was executing in Deferred Procedure Call <i>Shown as percent</i>	SATURATION	✗
azure.vm.processor_percent_interrupt_time (gauge)	Percentage of the time the processor spends receiving and servicing hardware interrupts <i>Shown as percent</i>	SATURATION	✗
azure.vm.processor_percent_io_wait_time (gauge)	Percentage of the time the processor spends on IO wait <i>Shown as percent</i>	SATURATION	✓
azure.vm.processor_percent_nice_time (gauge)	Percentage of time occupied by user level processes with a positive nice value <i>Shown as percent</i>	SATURATION	✗
azure.vm.processor_percent_privileged_time (gauge)	Percentage of the time in which the system was executing in Privileged mode <i>Shown as percent</i>	SATURATION	✗
azure.vm.processor_percent_processor_time (gauge)	Percentage of elapsed time that the processor spends to execute a non-Idle thread <i>Shown as percent</i>	SATURATION	✗
azure.vm.processor_percent_user_time (gauge)	Percentage of the time that the system is executing in User mode. <i>Shown as percent</i>	SATURATION	✗

azure.vm.processor_total_pct_interrupt_time (gauge)	Percentage of time the processor handles interrupts from applications or hardware devices.	SATURATION	✗
azure.vm.processor_total_pct_privileged_time (gauge)	Percentage of time the processor was running in Privileged mode.	SATURATION	✗
azure.vm.processor_total_pct_processor_time (gauge)	Percentage of time the processor was busy during the sampling interval	SATURATION	✗
azure.vm.processor_total_pct_user_time (gauge)	Percentage of time the processor was running in user mode.	SATURATION	✗
azure.vm.status (gauge)	Status of Azure VM	ERROR	✓
azure.vm.system_processes (gauge)	Number of processes <i>Shown as process</i>	SATURATION	✓
azure.vm.system_threads (gauge)	Number of threads that are in the processor queue. <i>Shown as thread</i>	SATURATION	✓

Network Layer

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.vm.network_in (gauge)	Number of bytes received on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✓
azure.vm.network_in_total (gauge)	The number of bytes received on all network interfaces by the Virtual Machine(s) (Incoming Traffic) <i>Shown as byte</i>	SATURATION	✓
azure.vm.network_interface_bytes_received (gauge)	Rate at which bytes are received over each network adapte <i>Shown as byte</i>	SATURATION	✗
azure.vm.network_interface_bytes_total (gauge)	Rate at which bytes are sent and received on the network interface <i>Shown as byte</i>	SATURATION	✗
azure.vm.network_interface_bytes_transmitted (gauge)	Rate at which bytes are sent over each network adapte <i>Shown as byte</i>	SATURATION	✗
azure.vm.network_interface_packets_received (gauge)	Rate at which packets are received on the network interface <i>Shown as packet</i>	SATURATION	✓
azure.vm.network_interface_packets_transmitted (gauge)	Rate at which packets are transmitted on the network interface. <i>Shown as packet</i>	SATURATION	✓
azure.vm.network_interface_total_collisions (gauge)	Rate of collisions	TRAFFIC	✗
azure.vm.network_interface_total_rx_errors (gauge)	Rate of errors when receiving packets <i>Shown as error</i>	ERROR	✓
azure.vm.network_interface_total_tx_errors (gauge)	Rate of errors when sending packets <i>Shown as error</i>	ERROR	✓
azure.vm.network_out (gauge)	Number of bytes sent out on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✗
azure.vm.network_out_total (gauge)	The number of bytes out on all network interfaces by the Virtual Machine(s) (Outgoing Traffic) <i>Shown as byte</i>	SATURATION	✗
azure.vm.tcpv4.connections.established (gauge)	Number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT <i>Shown as connection</i>	TRAFFIC	✓
azure.vm.tcpv4.connections.failures (gauge)	Number of times that TCP connections have made a direct transition to the CLOSED state from the SYN-SENT or SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state. <i>Shown as connection</i>	TRAFFIC	✓

azure.vm.tcpv4.connections.reset (gauge)	Number of times that TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED or CLOSE-WAIT state. <i>Shown as connection</i>	TRAFFIC	✗
azure.vm.tcpv4.segments.received_per_sec (gauge)	Rate at which TCP segments are received by using the TCP protocol <i>Shown as segment</i>	SATURATION	✗
azure.vm.tcpv4.segments.retransmitted_per_sec (gauge)	Rate at which segments are transmitted that contain one or more bytes that TCP recognizes as having been transmitted before. <i>Shown as segment</i>	SATURATION	✗
azure.vm.tcpv4.segments.sent_per_sec (gauge)	Rate at which TCP segments are sent by using the TCP protocol. <i>Shown as segment</i>	SATURATION	✗
azure.vm.tcpv4.connections.established (gauge)	Number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT <i>Shown as connection</i>	TRAFFIC	✓

▼ Azure VM Scale Set

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-vmsscaleset-monitors>

Azure Virtual Machine Scaleset monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.compute_virtualmachinescalesets.cpu_credits_consumed (count)	Total number of credits consumed by the Virtual Machine	SATURATION	✓
azure.compute_virtualmachinescalesets.cpu_credits_remaining (count)	Total number of credits available to burst	SATURATION	✓
azure.compute_virtualmachinescalesets.data_disk_qd_deprecated (count)	Data Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.compute_virtualmachinescalesets.data_disk_queue_depth_preview (count)	Data Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.compute_virtualmachinescalesets.data_disk_read_bytes_sec_deprecated (count)	Bytes/Sec read from a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_read_bytes_sec_preview (count)	Bytes/Sec read from a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_read_operations_sec_deprecated (count)	Read IOPS from a single disk during monitoring period <i>Shown as read</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_read_operations_sec_preview (count)	Read IOPS from a single disk during monitoring period <i>Shown as read</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_write_bytes_sec_deprecated (count)	Bytes/Sec written to a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_write_bytes_sec_preview (count)	Bytes/Sec written to a single disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_write_operations_sec_deprecated (count)	Write IOPS from a single disk during monitoring period <i>Shown as write</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.data_disk_write_operations_sec_preview (count)	Write IOPS from a single disk during monitoring period <i>Shown as write</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.disk_read_bytes (gauge)	Bytes read from disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✓

azure.compute_virtualmachinescalesets.disk_read_operations_sec (gauge)	Disk Read IOPS <i>Shown as read</i>	TRAFFIC	✓
azure.compute_virtualmachinescalesets.disk_write_bytes (gauge)	Bytes written to disk during monitoring period <i>Shown as byte</i>	TRAFFIC	✓
azure.compute_virtualmachinescalesets.disk_write_operations_sec (gauge)	Disk Write IOPS <i>Shown as write</i>	TRAFFIC	✓
azure.compute_virtualmachinescalesets.inbound_flows (count)	Inbound Flows are number of current flows in the inbound direction (traffic going into the VM)	TRAFFIC	✗
azure.compute_virtualmachinescalesets.inbound_flows_maximum_creation_rate (count)	The maximum creation rate of inbound flows (traffic going into the VM) <i>Shown as item</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.network_in (gauge)	The number of billable bytes received on all network interfaces by the Virtual Machine(s) (Incoming Traffic) <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.network_in_total (gauge)	The number of bytes received on all network interfaces by the Virtual Machine(s) (Incoming Traffic) <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.network_out (gauge)	The number of billable bytes out on all network interfaces by the Virtual Machine(s) (Outgoing Traffic) <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.network_out_total (gauge)	The number of bytes out on all network interfaces by the Virtual Machine(s) (Outgoing Traffic) <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_qd_deprecated (count)	OS Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.compute_virtualmachinescalesets.os_disk_queue_depth_preview (count)	OS Disk Queue Depth(or Queue Length)	SATURATION	✗
azure.compute_virtualmachinescalesets.os_disk_read_bytes_sec_deprecated (count)	Bytes/Sec read from a single disk during monitoring period for OS disk <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_read_bytes_sec_preview (count)	Bytes/Sec read from a single disk during monitoring period for OS disk <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_read_operations_sec_deprecated (count)	Read IOPS from a single disk during monitoring period for OS disk <i>Shown as read</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_read_operations_sec_preview (count)	Read IOPS from a single disk during monitoring period for OS disk <i>Shown as read</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_write_bytes_sec_deprecated (count)	Bytes/Sec written to a single disk during monitoring period for OS disk <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_write_bytes_sec_preview (count)	Bytes/Sec written to a single disk during monitoring period for OS disk <i>Shown as byte</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_write_operations_sec_deprecated (count)	Write IOPS from a single disk during monitoring period for OS disk <i>Shown as write</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.os_disk_write_operations_sec_preview (count)	Write IOPS from a single disk during monitoring period for OS disk <i>Shown as write</i>	TRAFFIC	✗
azure.compute_virtualmachinescalesets.outbound_flows (count)	Outbound Flows are number of current flows in the outbound direction (traffic going out of the VM)	TRAFFIC	✗
azure.compute_virtualmachinescalesets.outbound_flows_maximum_creation_rate (count)	The maximum creation rate of outbound flows (traffic going out of the VM) <i>Shown as item</i>	TRAFFIC	✗

azure.compute_virtualmachinescalesets.percentage_cpu (gauge)	The percentage of allocated compute units that are currently in use by the Virtual Machine(s) <i>Shown as percent</i>	SATURATION	✓
azure.compute_virtualmachinescalesets.premium_data_disk_cache_read_hit_preview (gauge)	Premium Data Disk Cache Read Hit <i>Shown as percent</i>	SATURATION	✗
azure.compute_virtualmachinescalesets.premium_data_disk_cache_read_miss_preview (gauge)	Premium Data Disk Cache Read Miss <i>Shown as percent</i>	SATURATION	✗
azure.compute_virtualmachinescalesets.premium_os_disk_cache_read_hit_preview (gauge)	Premium OS Disk Cache Read Hit <i>Shown as percent</i>	SATURATION	✗
azure.compute_virtualmachinescalesets.premium_os_disk_cache_read_miss_preview (gauge)	Premium OS Disk Cache Read Miss <i>Shown as percent</i>	SATURATION	✗
azure.compute_virtualmachinescalesets.status (gauge)	Status of the Azure VM Scale Set integration	ERROR	✓
azure.compute_virtualmachinescalesets.vm_count (count)	Virtual Machine count	SATURATION	✓

CloudFlare

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-cloudflare-monitors>

Cloudflare monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
cloudflare.requests.all (count)	Total request count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.cached (count)	Cached requests count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.requests.uncached (count)	Uncached requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.ssl.encrypted (count)	SSL encrypted requests count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.requests.ssl.unencrypted (count)	Unencrypted requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.country (count)	Request count, tagged by IATA country code <i>Shown as request</i>	TRAFFIC	✗
cloudflare.requests.status (count)	Request count, tagged by HTTP response code <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.content_type (count)	Request count, tagged by Content-Type <i>Shown as request</i>	TRAFFIC	✗
cloudflare.requests.ip_class (count)	Request count, tagged by IP class <i>Shown as request</i>	TRAFFIC	✗
cloudflare.bandwidth.all (count)	Total bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.cached (count)	Cached bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.uncached (count)	Uncached bandwidth <i>Shown as byte</i>	SATURATION	✓
cloudflare.bandwidth.ssl.encrypted (count)	SSL encrypted bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.ssl.unencrypted (count)	Unencrypted bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.country (count)	Bandwidth tagged by IATA country code <i>Shown as byte</i>	SATURATION	✗

cloudflare.bandwidth.content_type (count)	Bandwidth tagged by Content-Type <i>Shown as byte</i>	SATURATION	✗
cloudflare.threats.all (count)	Total threats <i>Shown as operation</i>	ERROR	✓
cloudflare.threats.type (count)	Threats tagged by type <i>Shown as operation</i>	ERROR	✗
cloudflare.threats.country (count)	Threats tagged by IATA country code <i>Shown as operation</i>	ERROR	✗
cloudflare.pageviews.all (count)	Total page views <i>Shown as page</i>	TRAFFIC	✓
cloudflare.pageviews.search_engine (count)	Page views tagged by search engine <i>Shown as page</i>	TRAFFIC	✗
cloudflare.uniques.all (count)	Unique visitors count <i>Shown as connection</i>	TRAFFIC	✗
cloudflare.dns.query.all (count)	DNS query count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.dns.query.uncached (count)	Uncached DNS query count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.dns.query.stale (count)	Stale DNS query count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.dns.response_time.avg (gauge)	DNS query average response time <i>Shown as millisecond</i>	LATENCY	✓
cloudflare.dns.response_time.median (gauge)	DNS query median response time <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.dns.response_time.90p (gauge)	DNS query response time to the 90th percentile <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.dns.response_time.99p (gauge)	DNS query response time to the 99th percentile <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.workers.requests.all (count)	The request count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	TRAFFIC	✗
cloudflare.workers.requests.errors (count)	The error count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	ERROR	✗
cloudflare.workers.requests.subrequests (count)	The subrequest count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	TRAFFIC	✗
cloudflare.workers.response_time.75p (gauge)	The worker response time to the 75th percentile (metrics may not show without enabled API Key permissions) <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.workers.response_time.99p (gauge)	The worker response time to the 99th percentile (metrics may not show without enabled API Key permissions) <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.load_balancer.pool.round_trip_time.average (gauge)	The average round trip time to reach the load balancer pool <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.load_balancer.pool.health.status (count)	The load balancer pool health status <i>Shown as request</i>	ERROR	✓

▼ Azure Firewall

Confluence page URL: [Azure Firewall Monitoring](#)

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-firewall-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.network_azurefirewalls.application_rule_hit (count)	The number of times application rules were hit <i>Shown as hit</i>	TRAFFIC	✓
azure.network_azurefirewalls.count (count)	The number of Azure firewalls	SATURATION	✓

azure.network_azurefirewalls.data_processed (gauge)	The total amount of data processed by a firewall <i>Shown as byte</i>	SATURATI ON	✗
azure.network_azurefirewalls.firewall_health (gauge)	Indicates the overall health of a firewall <i>Shown as percent</i>	SATURATI ON	✓
azure.network_azurefirewalls.network_rule_hit (count)	The number of times network rules were hit <i>Shown as hit</i>	TRAFFIC	✓
azure.network_azurefirewalls.snat_port_utilization (gauge)	The percentage of outbound SNAT ports currently in use <i>Shown as percent</i>	SATURATI ON	✗
azure.network_azurefirewalls.throughput (gauge)	The throughput processed by a firewall <i>Shown as bit</i>	SATURATI ON	✗

▼ Azure Load Balancer

Confluence page URL:

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-load-balancer-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.network_loadbalancers.allocated_snat_ports (count)	<i>Total number of SNAT ports allocated within time period</i>	TRAFFIC	✗
azure.network_loadbalancers.backend_pool_host_count (gauge)	Total number of hosts in the backed end pool for the Load Balancer <i>Shown as host</i>	SATURATION	✗
azure.network_loadbalancers.byte_count (count)	<i>Total number of Bytes transmitted within time period</i> <i>Shown as byte</i>	SATURATION	✗
azure.network_loadbalancers.dip_availability (gauge)	<i>Average Load Balancer health probe status per time duration</i> <i>Shown as percent</i>	SATURATION	✓
azure.network_loadbalancers.packet_count (count)	Total number of Packets transmitted within time period <i>Shown as packet</i>	TRAFFIC	✗
azure.network_loadbalancers.snat_connection_count (count)	Total number of new SNAT connections created within time period <i>Shown as connection</i>	SATURATION	✗
azure.network_loadbalancers.status (gauge)	<i>Status of Azure Load Balancer</i>	SATURATION	✓
azure.network_loadbalancers.syn_count (count)	Total number of SYN Packets transmitted within time period <i>Shown as packet</i>	SATURATION	✗
azure.network_loadbalancers.used_snat_ports (count)	Total number of SNAT ports used within time period	SATURATION	✓
azure.network_loadbalancers.vip_availability (gauge)	<i>Average Load Balancer data path availability per time duration</i> <i>Shown as percent</i>	ERROR /STATUS	✓

▼ Azure App Service

Confluence Page URL: [Azure App Service monitoring](#)

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-appservice-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.app_services.average_memory_working_set (gauge)	The average memory used by the app <i>Shown as byte</i>	SATURATION	✓
azure.app_services.average_response_time (gauge)	The average response time of the app (Deprecated) <i>Shown as second</i>	SATURATION	✓
azure.app_services.bytes_received (gauge)	The average number of bytes received <i>Shown as byte</i>	TRAFFIC	✗
azure.app_services.bytes_sent (gauge)	The average number of bytes sent <i>Shown as byte</i>	TRAFFIC	✗
azure.app_services.connections (count)	The number of bound sockets existing in the sandbox <i>Shown as connection</i>	SATURATION	✗

azure.app_services.cpu_time (gauge)	The average cpu time of the app <i>Shown as second</i>	SATURATION	✓
azure.app_services.current_assemblies (count)	The current number of assemblies loaded across all app domains	SATURATION	✗
azure.app_services.file_system_usage (gauge)	The percentage of filesystem quota consumed by the app <i>Shown as byte</i>	SATURATION	✗
azure.app_services.function_execution_count (count)	The number of function execution count.	SATURATION	✗
azure.app_services.function_execution_units (count)	The number of function execution units.	SATURATION	✗
azure.app_services.gen_0_garbage_collections (count)	The count of Gen 0 Garbage Collections <i>Shown as garbage collection</i>	SATURATION	✗
azure.app_services.gen_1_garbage_collections (count)	The count of Gen 1 Garbage Collections <i>Shown as garbage collection</i>	SATURATION	✗
azure.app_services.gen_2_garbage_collections (count)	The count of Gen 2 Garbage Collections <i>Shown as garbage collection</i>	SATURATION	✗
azure.app_services.handle_count (count)	The number of handles currently open by the app process	TRAFFIC	✗
azure.app_services.health_check_status (gauge)	The level of health check status <i>Shown as percent</i>	ERROR /STATUS	✗
azure.app_services.http101 (count)	The total number of 101 requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http2xx (count)	The total number of 2xx requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http3xx (count)	The total number of 3xx requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http401 (count)	The total number of 401 requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http403 (count)	The total number of 403 requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http404 (count)	The total number of 404 requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http406 (count)	The total number of 406 requests served by the app <i>Shown as request</i>	TRAFFIC	✗
azure.app_services.http4xx (count)	The total number of 4xx requests served by the app <i>Shown as request</i>	TRAFFIC	✓
azure.app_services.http5xx (count)	The total number of 5xx requests served by the app <i>Shown as request</i>	TRAFFIC	✓
azure.app_services.io_other_bytes_per_second (rate)	The rate of IO other bytes per Second <i>Shown as byte</i>	TRAFFIC	✗
azure.app_services.io_other_operations_per_second (rate)	The rate of IO other Operations per Second <i>Shown as operation</i>	TRAFFIC	✗
azure.app_services.io_read_bytes_per_second (rate)	The rate of IO read Bytes per Second <i>Shown as byte</i>	TRAFFIC	✗
azure.app_services.io_read_operations_per_second (rate)	The rate of IO read operations per Second <i>Shown as operation</i>	TRAFFIC	✗
azure.app_services.io_write_bytes_per_second (rate)	The rate of IO write bytes per Second <i>Shown as byte</i>	TRAFFIC	✗
azure.app_services.io_write_operations_per_second (rate)	The rate of IO write operations per Second <i>Shown as operation</i>	TRAFFIC	✗
azure.app_services.memory_working_set (gauge)	The average memory used by the app <i>Shown as byte</i>	SATURATION	✗
azure.app_services.private_bytes (gauge)	The current size of memory allocated that can't be shared with other process <i>Shown as byte</i>	SATURATION	✗

azure.app_services.requests (count)	The total number of requests served by the app <i>Shown as request</i>	SATURATION	✓
azure.app_services.requests_in_application_queue (count)	The number of requests in the application request queue <i>Shown as request</i>	SATURATION	✗
azure.app_services.response_time (gauge)	The time taken for the app to serve requests <i>Shown as second</i>	SATURATION	✗
azure.app_services.status (gauge)	The status of Azure App Services	ERROR /STATUS	✓
azure.app_services.count (gauge)	The count of Azure App Services resources	SATURATION	✗
azure.app_services.thread_count (count)	The number of threads currently active in the app process <i>Shown as thread</i>	SATURATION	✗
azure.app_services.total_app_domains (count)	The current number of App Domains loaded in this application	SATURATION	✗
azure.app_services.total_app_domains_unloaded (count)	The total number of App Domains unloaded since the start of the application	SATURATION	✗
azure.app_services.webjob_count (gauge)	The current number of webjobs configured per-webapp	SATURATION	✗
azure.web_serverfarms.maximum_number_of_workers (gauge)	The maximum number of workers configured for an App Service Plan	SATURATION	✗
azure.web_serverfarms.number_of_sites (gauge)	The number of sites configured for an App Service Plan	SATURATION	✗
azure.web_serverfarms.target_worker_count (gauge)	The target number of workers configured for an App Service Plan	SATURATION	✗
azure.web_serverfarms.current_instance_count (gauge)	The current number of instances running inside an App Service Plan	SATURATION	✗

▼ Azure Application Gateway
Application Gateway monitoring

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-application-gateway-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.network_applicationgateways.current_connections (gauge)	Count of current connections established with Application Gateway <i>Shown as connection</i>	TRAFFIC	✓
azure.network_applicationgateways.failed_requests (count)	Count of failed requests that Application Gateway has served <i>Shown as request</i>	TRAFFIC	✓
azure.network_applicationgateways.healthy_host_count (gauge)	Number of healthy backend hosts <i>Shown as host</i>	SATURAT ION	✓
azure.network_applicationgateways.response_status (count)	Http response status returned by Application Gateway	ERROR /STATUS	✓
azure.network_applicationgateways.backend_response_status (count)	Backend Response Status	ERROR /STATUS	✗
azure.network_applicationgateways.status (gauge)	Status of Azure Application Gateway	ERROR /STATUS	✓
azure.network_applicationgateways.throughput (rate)	Number of bytes per second the Application Gateway has served <i>Shown as byte</i>	LATENCY	✗
azure.network_applicationgateways.total_requests (count)	Count of successful requests that Application Gateway has served <i>Shown as request</i>	TRAFFIC	✗
azure.network_applicationgateways.unhealthy_host_count (gauge)	Number of unhealthy backend hosts <i>Shown as host</i>	SATURAT ION	✓

azure.network_applicationgateways.bytes_sent (count)	Number of bytes sent through the Application Gateway <i>Shown as byte</i>	LATENCY	✗
azure.network_applicationgateways.bytes_received (count)	Number of bytes received through the Application Gateway <i>Shown as byte</i>	LATENCY	✗
azure.network_applicationgateways.compute_units (count)	Compute units consumed	SATURAT ION	✗
azure.network_applicationgateways.capacity_units (count)	Capacity units consumed	SATURAT ION	✗
azure.network_applicationgateways.backend_connect_time (gauge)	Time spent establishing a connection with a backend server <i>Shown as millisecond</i>	LATENCY	✓
azure.network_applicationgateways.avg_request_count_per_healthy_host (gauge)	Average request count per minute per healthy backend host in a pool <i>Shown as request</i>	TRAFFIC	✗
azure.network_applicationgateways.backend_first_byte_response_time (gauge)	Time interval between start of establishing a connection to backend server and receiving the first byte of the response header <i>Shown as millisecond</i>	LATENCY	✓
azure.network_applicationgateways.backend_last_byte_response_time (gauge)	Time interval between start of establishing a connection to backend server and receiving the last byte of the response body <i>Shown as millisecond</i>	LATENCY	✗
azure.network_applicationgateways.fixed_billable_capacity_units (count)	Minimum capacity units that will be charged	SATURAT ION	✗
azure.network_applicationgateways.client_rtt (gauge)	Average round trip time between clients and Application Gateway <i>Shown as millisecond</i>	LATENCY	✗
azure.network_applicationgateways.tls_protocol (count)	The number of TLS and non-TLS requests initiated by the client that established connection with the Application Gateway.	TRAFFIC	✗
azure.network_applicationgateways.new_connections_per_second (rate)	New connections per second established with Application Gateway <i>Shown as connection</i>	TRAFFIC	✗
azure.network_applicationgateways.cpu_utilization (gauge)	Current CPU utilization of the Application Gateway <i>Shown as percent</i>	SATURAT ION	✗
azure.network_applicationgateways.estimated_billed_capacity_units (count)	Estimated capacity units that will be charged	SATURAT ION	✗
azure.network_applicationgateways.matched_count (count)	Web Application Firewall Total Rule Distribution for the incoming traffic	TRAFFIC	✗
azure.network_applicationgateways.application_gateway_total_time (gauge)	Average time that it takes for a request to be processed and its response to be sent. <i>Shown as millisecond</i>	LATENCY	✗
azure.network_applicationgateways.blocked_req_count (count)	Web Application Firewall blocked requests count	TRAFFIC	✗
azure.network_applicationgateways.blocked_count (count)	Web Application Firewall blocked requests rule distribution	TRAFFIC	✗

▼ Azure SQL

Confluence page URL: [PaaS SQL monitoring](#)

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-paas-sql-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.sql_servers_databases.active_queries (count)	Active queries across all workload groups. Applies only to data warehouses. <i>Shown as query</i>	TRAFFIC	✗
azure.sql_servers_databases.app_cpu_billed (count)	App CPU billed. Applies to serverless databases.	SATURATI ON	✗

azure.sql_servers_databases.app_cpu_percentage (gauge)	App CPU percentage. Applies to serverless databases. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.app_memory_percentage (gauge)	App memory percentage. Applies to serverless databases. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.base_blob_storage_size (gauge)	Base blob storage size. Applies to Hyperscale databases. <i>Shown as byte</i>	SATURATI ON	✗
azure.sql_servers_databases.blocked_by_firewall (count)	Blocked by Firewall <i>Shown as connection</i>	TRAFFIC	✗
azure.sql_servers_databases.cache_hit_percentage (gauge)	Cache hit percentage. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.cache_used_percentage (gauge)	Cache used percentage. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.connection_failed (count)	Failed Connections <i>Shown as connection</i>	TRAFFIC	✓
azure.sql_servers_databases.connection_successful (count)	Successful Connections <i>Shown as connection</i>	TRAFFIC	✓
azure.sql_servers_databases.cpu_limit (count)	CPU limit. Applies to vCore-based databases.	SATURATI ON	✗
azure.sql_servers_databases.cpu_percent (gauge)	CPU percentage <i>Shown as percent</i>	SATURATI ON	✓
azure.sql_servers_databases.cpu_used (count)	CPU used. Applies to vCore-based databases.	SATURATI ON	✗
azure.sql_servers_databases.data_space_allocated (gauge)	Allocated data storage. Not applicable to data warehouses. <i>Shown as byte</i>	SATURATI ON	✗
azure.sql_servers_databases.data_storage_size (count)	Data Storage Size is comprised of the size of your data and the transaction log. The metric is counted towards the 'Storage' portion of your bill. Applies only to data warehouses.	SATURATI ON	✗
azure.sql_servers_databases.deadlock (count)	Deadlocks. Not applicable to data warehouses.	SATURATI ON	✗
azure.sql_servers_databases.differential_backup_storage_size (gauge)	Cumulative differential backup storage size. Applies to vCore-based databases. Not applicable to Hyperscale databases. <i>Shown as byte</i>	SATURATI ON	✗
azure.sql_servers_databases.disaster_recovery_storage_size (count)	Disaster Recovery Storage Size is reflected as 'Disaster Recovery Storage' in your bill. Applies only to data warehouses.	SATURATI ON	✗
azure.sql_servers_databases.dtu_consumption_percent (gauge)	DTU Percentage. Applies to DTU-based databases. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.dtu_limit (count)	DTU Limit. Applies to DTU-based databases. <i>Shown as unit</i>	SATURATI ON	✗
azure.sql_servers_databases.dtu_used (count)	DTU used. Applies to DTU-based databases. <i>Shown as unit</i>	SATURATI ON	✗
azure.sql_servers_databases.dwu_consumption_percent (gauge)	DWU percentage. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.dwu_limit (count)	DWU limit. Applies only to data warehouses. <i>Shown as unit</i>	SATURATI ON	✗
azure.sql_servers_databases.dwu_used (count)	DWU used. Applies only to data warehouses. <i>Shown as unit</i>	SATURATI ON	✗
azure.sql_servers_databases.effective_cap_resource_percent (gauge)	A hard limit on the percentage of resources allowed for the workload group, taking into account Effective Min Resource Percentage allocated for other workload groups. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.effective_min_resource_percent (gauge)	Minimum percentage of resources reserved and isolated for the workload group, taking into account the service level minimum. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.full_backup_storage_size (gauge)	Cumulative full backup storage size. Applies to vCore-based databases. Not applicable to Hyperscale databases. <i>Shown as byte</i>	SATURATI ON	✗

azure.sql_servers_databases.local_tempdb_percentage (gauge)	Local tempdb percentage. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.log_backup_storage_size (gauge)	Cumulative log backup storage size. Applies to vCore-based and Hyperscale databases. <i>Shown as byte</i>	SATURATI ON	✗
azure.sql_servers_databases.log_write_percent (gauge)	Log IO percentage. Not applicable to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.memory_percentage (gauge)	Memory percentage. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.physical_data_read_percent (gauge)	Data IO percentage <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.queued_queries (count)	Queued queries across all workload groups. Applies only to data warehouses. <i>Shown as query</i>	TRAFFIC	✗
azure.sql_servers_databases.replication_links.count (gauge)	The amount of replication links per database	SATURATI ON	✗
azure.sql_servers_databases.sessions_percent (gauge)	Sessions percentage. Not applicable to data warehouses. <i>Shown as percent</i>	TRAFFIC	✗
azure.sql_servers_databases.snapshot_backup_storage_size (gauge)	Cumulative snapshot backup storage size. Applies to Hyperscale databases. <i>Shown as byte</i>	SATURATI ON	✗
azure.sql_servers_databases.snapshot_storage_size (count)	Snapshot Storage Size is the size of the incremental changes captured by snapshots to create user-defined and automatic restore points. The metric is counted towards the 'Storage' portion of your bill. Applies only to data warehouses.	SATURATI ON	✗
azure.sql_servers_databases.sql_server_process_core_percent (gauge)	CPU usage percentage for the SQL Server process, as measured by the operating system. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.sql_server_process_memory_percent (gauge)	Memory usage percentage for the SQL Server process, as measured by the operating system. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.status (gauge)	Status of Azure SQL Database	ERROR /STATUS	✓
azure.sql_servers_databases.storage (gauge)	Data space used. Not applicable to data warehouses. <i>Shown as byte</i>	SATURATI ON	✓
azure.sql_servers_databases.storage_percent (gauge)	Data space used percent. Not applicable to data warehouses or hyperscale databases. <i>Shown as percent</i>	SATURATI ON	✓
azure.sql_servers_databases.tempdb_data_file_size_kilobytes (count)	Tempdb Data File Size Kilobytes.	SATURATI ON	✗
azure.sql_servers_databases.tempdb_log_file_size_kilobytes (count)	Tempdb Log File Size Kilobytes.	SATURATI ON	✗
azure.sql_servers_databases.tempdb_percent_log_used (gauge)	Tempdb Percent Log Used. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.workers_percent (gauge)	Workers percentage. Not applicable to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.workload_group_active_queries (count)	Queued queries within the workload group. Applies only to data warehouses. <i>Shown as query</i>	TRAFFIC	✗
azure.sql_servers_databases.workload_group_allocation_by_cap_resource_percent (gauge)	Allocated percentage of resources relative to the specified cap resources per workload group. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.workload_group_allocation_by_system_percent (gauge)	Allocated percentage of resources relative to the entire system per workload group. Applies only to data warehouses. <i>Shown as percent</i>	SATURATI ON	✗
azure.sql_servers_databases.workload_group_query_timeouts (count)	Queries that have timed out for the workload group. Applies only to data warehouses. <i>Shown as query</i>	SATURATI ON	✗

azure.sql_servers_databases.workload_group_queued_queries (count)	Active queries within the workload group. Applies only to data warehouses. <i>Shown as query</i>	TRAFFIC	✗
azure.sql_servers_databases.xtp_storage_percent (gauge)	In-Memory OLTP storage percent. Not applicable to data warehouses. <i>Shown as percent</i>	SATURATION	✗

▼ Azure Express Route

ExpressRoute monitoring

Terraform Module Repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-expressroute-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.network_expressrouteircuits.bits_in_per_second (rate)	Bits ingressing Azure per second <i>Shown as bit</i>	TRAFFIC	✓
azure.network_expressrouteircuits.bits_out_per_second (rate)	Bits egressing Azure per second <i>Shown as bit</i>	TRAFFIC	✓
azure.network_expressrouteircuits.status (gauge)	Status of Azure ExpressRoute integration	ERROR	✓
azure.network_expressrouteircuits.arp_availability (gauge)	ARP Availability %	ERROR	✓
azure.network_expressrouteircuits.bgp_availability (gauge)	BGP Availability %	ERROR	✓
azure.network_expressrouteircuits.qos_drop_bits_in_per_second (rate)	QoS Drop Bits ingressing Azure per second <i>Shown as bit</i>	TRAFFIC	✓
azure.network_expressrouteircuits.qos_drop_bits_out_per_second (rate)	QoS Drop Bits egressing Azure per second <i>Shown as bit</i>	TRAFFIC	✓

▼ Azure Blob Storage Account

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-blob-storage-monitors>

Azure Blob Storage Accounts monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.storage_storageaccounts_blobservices.availability (gauge)	The percentage of availability for the storage service or the specified API operation. Availability is calculated by taking the TotalBillableRequests value and dividing it by the number of applicable requests including those that produced unexpected errors. All unexpected errors result in reduced availability for the storage service or the specified API operation. <i>Shown as percent</i>	SATURATION	✓
azure.storage_storageaccounts_blobservices.blob_capacity (gauge)	The amount of storage used by the storage account's Blob service in bytes. <i>Shown as byte</i>	SATURATION	✓
azure.storage_storageaccounts_blobservices.blob_count (count)	The number of Blob in the storage account's Blob service.	SATURATION	✓
azure.storage_storageaccounts_blobservices.container_count (count)	The number of containers in the storage account's Blob service.	SATURATION	✗
azure.storage_storageaccounts_blobservices.egress (gauge)	The amount of egress data in bytes. This number includes egress from an external client into Azure Storage as well as egress within Azure. As a result this number does not reflect billable egress. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_blobservices.ingress (gauge)	The amount of ingress data in bytes. This number includes ingress from an external client into Azure Storage as well as ingress within Azure. <i>Shown as byte</i>	TRAFFIC	✓

azure.storage_storageaccounts_blobservices.success_e2_elatency (gauge)	The average end-to-end latency of successful requests made to a storage service or the specified API operation in milliseconds. This value includes the required processing time within Azure Storage to read the request send the response and receive acknowledgment of the response. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_blobservices.success_server_latency (gauge)	The average latency used by Azure Storage to process a successful request in milliseconds. This value does not include the network latency specified in AverageE2ELatency. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_blobservices.transactions (count)	The number of requests made to a storage service or the specified API operation. This number includes successful and failed requests as well as requests which produced errors. Use ResponseType dimension for the number of different type of response.	TRAFFIC	✓

▼ Azure Virtual Networks

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-virtualnetworks-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.network_virtualnetworks.connected_peerings (gauge)	Peers connected on the virtual network	SATURATION	✓
azure.network_virtualnetworks.subnets.assigned_addresses (gauge)	Addresses assigned on the subnet	SATURATION	✓
azure.network_virtualnetworks.subnets.available_addresses (gauge)	Addresses available on the subnet	SATURATION	✓
azure.network_virtualnetworks.total_addresses (gauge)	Total addresses on the virtual network	SATURATION	✓
azure.network_virtualnetworks.total_peerings (gauge)	Total peerings on the virtual network	SATURATION	✓
azure.network_virtualnetworks.total_subnets (gauge)	Total number of subnets on the virtual network	SATURATION	✓

▼ Azure Table Storage Account

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-table-storage-monitors>

Azure Table Storage Accounts monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.storage_storageaccounts_tableservices.availability (gauge)	The percentage of availability for the storage service or the specified API operation. Availability is calculated by taking the TotalBillableRequests value and dividing it by the number of applicable requests including those that produced unexpected errors. All unexpected errors result in reduced availability for the storage service or the specified API operation. <i>Shown as percent</i>	SATURATION	✓
azure.storage_storageaccounts_tableservices.table_capacity (gauge)	The amount of storage used by the storage account's Table service in bytes. <i>Shown as byte</i>	SATURATION	✓
azure.storage_storageaccounts_tableservices.table_count (count)	The number of Table in the storage account's Table service.	SATURATION	✓
azure.storage_storageaccounts_tableservices.container_count (count)	The number of containers in the storage account's Table service.	SATURATION	✗
azure.storage_storageaccounts_tableservices.egress (gauge)	The amount of egress data in bytes. This number includes egress from an external client into Azure Storage as well as egress within Azure. As a result this number does not reflect billable egress. <i>Shown as byte</i>	TRAFFIC	✓

azure.storage_storageaccounts_tableservices.ingress (gauge)	The amount of ingress data in bytes. This number includes ingress from an external client into Azure Storage as well as ingress within Azure. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_tableservices.success_e2_elatency (gauge)	The average end-to-end latency of successful requests made to a storage service or the specified API operation in milliseconds. This value includes the required processing time within Azure Storage to read the request send the response and receive acknowledgment of the response. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_tableservices.success_server_latency (gauge)	The average latency used by Azure Storage to process a successful request in milliseconds. This value does not include the network latency specified in AverageE2ELatency. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_tableservices.transactions (count)	The number of requests made to a storage service or the specified API operation. This number includes successful and failed requests as well as requests which produced errors. Use ResponseType dimension for the number of different type of response.	TRAFFIC	✓

▼ Azure File Storage Account

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-file-storage-monitors>

Azure File Storage Accounts monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.storage_storageaccounts_fileservices.availability (gauge)	The percentage of availability for the storage service or the specified API operation. Availability is calculated by taking the TotalBillableRequests value and dividing it by the number of applicable requests including those that produced unexpected errors. All unexpected errors result in reduced availability for the storage service or the specified API operation. <i>Shown as percent</i>	SATURATION	✓
azure.storage_storageaccounts_fileservices.file_capacity (gauge)	The amount of storage used by the storage account's file service in bytes. <i>Shown as byte</i>	SATURATION	✓
azure.storage_storageaccounts_fileservices.file_count (count)	The number of File in the storage account's File service.	SATURATION	✓
azure.storage_storageaccounts_fileservices.container_count (count)	The number of containers in the storage account's File service.	SATURATION	✗
azure.storage_storageaccounts_fileservices.egress (gauge)	The amount of egress data in bytes. This number includes egress from an external client into Azure Storage as well as egress within Azure. As a result this number does not reflect billable egress. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_fileservices.ingress (gauge)	The amount of ingress data in bytes. This number includes ingress from an external client into Azure Storage as well as ingress within Azure. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_fileservices.success_e2_elatency (gauge)	The average end-to-end latency of successful requests made to a storage service or the specified API operation in milliseconds. This value includes the required processing time within Azure Storage to read the request send the response and receive acknowledgment of the response. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_fileservices.success_server_latency (gauge)	The average latency used by Azure Storage to process a successful request in milliseconds. This value does not include the network latency specified in AverageE2ELatency. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_fileservices.transactions (count)	The number of requests made to a storage service or the specified API operation. This number includes successful and failed requests as well as requests which produced errors. Use ResponseType dimension for the number of different type of response.	TRAFFIC	✓

▼ Azure Queue Storage Account

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-table-storage-monitors>

Azure Queue Storage Accounts monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
azure.storage_storageaccounts_queueservices.availability (gauge)	The percentage of availability for the storage service or the specified API operation. Availability is calculated by taking the TotalBillableRequests value and dividing it by the number of applicable requests including those that produced unexpected errors. All unexpected errors result in reduced availability for the storage service or the specified API operation. <i>Shown as percent</i>	SATURATION	✓
azure.storage_storageaccounts_queueservices.queue_capacity (gauge)	The amount of storage used by the storage account's Queue service in bytes. <i>Shown as byte</i>	SATURATION	✓
azure.storage_storageaccounts_queueservices.table_count (count)	The number of Table in the storage account's Queue service.	SATURATION	✓
azure.storage_storageaccounts_queueservices.container_count (count)	The number of containers in the storage account's Queue service.	SATURATION	✗
azure.storage_storageaccounts_queueservices.egress (gauge)	The amount of egress data in bytes. This number includes egress from an external client into Azure Storage as well as egress within Azure. As a result this number does not reflect billable egress. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_queueservices.ingress (gauge)	The amount of ingress data in bytes. This number includes ingress from an external client into Azure Storage as well as ingress within Azure. <i>Shown as byte</i>	TRAFFIC	✓
azure.storage_storageaccounts_queueservices.success_e2_elatency (gauge)	The average end-to-end latency of successful requests made to a storage service or the specified API operation in milliseconds. This value includes the required processing time within Azure Storage to read the request send the response and receive acknowledgment of the response. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_queueservices.success_server_latency (gauge)	The average latency used by Azure Storage to process a successful request in milliseconds. This value does not include the network latency specified in AverageE2ELatency. <i>Shown as millisecond</i>	LATENCY	✓
azure.storage_storageaccounts_queueservices.transactions (count)	The number of requests made to a storage service or the specified API operation. This number includes successful and failed requests as well as requests which produced errors. Use ResponseType dimension for the number of different type of response.	TRAFFIC	✓

▼ Azure Gitlab Runners Machine

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-gitlab-runners-monitor>

Azure Gitlab Runners monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
sys.mem.used (count)	System Memory Used (<i>Shown as bytes</i>)	SATURATION	✓
sys.mem.free (count)	System Memory Free (<i>Shown as bytes</i>)	SATURATION	✓
sys.disk.used (count)	System Disk Used (<i>Shown as bytes</i>)	SATURATION	✓



sys.disk.free (count)	System Disk Free (<i>Shown as bytes</i>)	SATURATION	✓
azure.vm.disk_read_operations_sec (gauge)	(ARM VM only) Amount of read operations per second <i>Shown as operation</i>	TRAFFIC	✓
azure.vm.disk_write_bytes (count)	(ARM VM only) Amount of bytes written <i>Shown as byte</i>	SATURATION	✓
azure.vm.disk_write_bytes_sec (gauge)	(Classic VM only) Amount of bytes written <i>Shown as byte</i>	SATURATION	✓
azure.vm.disk_write_operations_sec (gauge)	(ARM VM only) Amount of write operations per second <i>Shown as operation</i>	TRAFFIC	✓
azure.vm.percentage_cpu (gauge)	Percentage of CPU resources used <i>Shown as percent</i>	SATURATION	✓
azure.vm.status (gauge)	Status of Azure VM	ERROR	✓
azure.vm.network_in (gauge)	Number of bytes received on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✓
azure.vm.network_in_total (gauge)	The number of bytes received on all network interfaces by the Virtual Machine(s) (Incoming Traffic) <i>Shown as byte</i>	SATURATION	✓
azure.vm.network_out (gauge)	Number of bytes sent on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✓
azure.vm.network_out_total (gauge)	The number of bytes sent on all network interfaces by the Virtual Machine(s) (Incoming Traffic) <i>Shown as byte</i>	SATURATION	✓

▼ Azure Cosmos DB

Terraform Module Repo: <https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-azr-cosmosdb-monitors>

METRIC NAME	DESCRIPTION	SIGNAL	COVER	SLO
azure.cosmosdb.add_region (count)	Region Added	SATURATION	✗	
azure.cosmosdb.autoscale_max_throughput (count)	Autoscale Max Throughput	SATURATION	✗	
azure.cosmosdb.available_storage (gauge)	Total available storage reported at 5 minutes granularity per region <i>Shown as byte</i>	SATURATION	✓	
azure.cosmosdb.cassandra_connection_closures (count)	The number of Cassandra connections that were closed <i>Shown as connection</i>	TRAFFIC	✗	
azure.cosmosdb.cassandra_connector_avg_replication_latency (count)	Cassandra Connector Average Replication Latency <i>Shown as millisecond</i>	LATENCY	✗	
azure.cosmosdb.cassandra_connector_replication_health_status (count)	Cassandra Connector Replication Health Status	ERROR	✗	
azure.cosmosdb.cassandra_keyspace_create (count)	Cassandra Keyspace Created	TRAFFIC	✗	

azure.cosmosdb.cassandra_keyspace_delete (count)	Cassandra Keyspace Deleted	TRAFFIC	✗	
azure.cosmosdb.cassandra_keyspace_throughput_update (count)	Cassandra Keyspace Throughput Updated	TRAFFIC	✗	
azure.cosmosdb.cassandra_keyspace_update (count)	Cassandra Keyspace Updated	TRAFFIC	✗	
azure.cosmosdb.cassandra_request_charges (count)	RUs consumed for Cassandra requests made	TRAFFIC	✗	
azure.cosmosdb.cassandra_requests (count)	The number of Cassandra requests made <i>Shown as request</i>	TRAFFIC	✗	
azure.cosmosdb.cassandra_table_create (count)	Cassandra table created <i>Shown as table</i>	TRAFFIC	✗	
azure.cosmosdb.cassandra_table_delete (count)	Cassandra Table Deleted <i>Shown as table</i>	TRAFFIC	✗	
azure.cosmosdb.cassandra_table_throughput_update (count)	Cassandra Table Throughput Updated	TRAFFIC	✗	
azure.cosmosdb.cassandra_table_update (count)	Cassandra Table Updated <i>Shown as table</i>	TRAFFIC	✗	
azure.cosmosdb.count (count)	Count of Cosmos DB	TRAFFIC	✓	
azure.cosmosdb.create_account (count)	Account Created	TRAFFIC	✗	
azure.cosmosdb.data_usage (gauge)	Total data usage reported at 5 minutes granularity per region <i>Shown as byte</i>	SATURATION	✓	
azure.cosmosdb.dedicated_gateway_average_cpu_usage (gauge)	The average CPU usage across dedicated gateway instances <i>Shown as percent</i>	SATURATION	✓	
azure.cosmosdb.dedicated_gateway_average_memory_usage (gauge)	The average memory usage across dedicated gateway instances, which is used for both routing requests and caching data <i>Shown as byte</i>	SATURATION	✓	
azure.cosmosdb.dedicated_gateway_maximum_cpu_usage (gauge)	The average Maximum CPU usage across dedicated gateway instances <i>Shown as percent</i>	SATURATION	✗	
azure.cosmosdb.dedicated_gateway_requests (count)	Dedicated Gateway Requests <i>Shown as request</i>	TRAFFIC	✓	
azure.cosmosdb.delete_account (count)	Account Deleted	TRAFFIC	✗	
azure.cosmosdb.document_count (count)	Total document count reported at 5 minutes granularity per region. <i>Shown as document</i>	TRAFFIC	✗	
azure.cosmosdb.document_quota (gauge)	Total storage quota reported at 5 minutes granularity per region. <i>Shown as byte</i>	SATURATION	✗	

azure.cosmosdb.gremlin_database_create (count)	Gremlin Database Created	TRAFFIC		
azure.cosmosdb.gremlin_database_delete (count)	Gremlin Database Deleted	TRAFFIC		
azure.cosmosdb.gremlin_database_throughput_update (count)	Gremlin Database Throughput Updated	TRAFFIC		
azure.cosmosdb.gremlin_database_update (count)	Gremlin Database Updated	TRAFFIC		
azure.cosmosdb.gremlin_graph_create (count)	Gremlin Graph Created	TRAFFIC		
azure.cosmosdb.gremlin_graph_delete (count)	Gremlin Graph Deleted	TRAFFIC		
azure.cosmosdb.gremlin_graph_throughput_update (count)	Gremlin Graph Throughput Updated	TRAFFIC		
azure.cosmosdb.gremlin_graph_update (count)	Gremlin Graph Updated	TRAFFIC		
azure.cosmosdb.index_usage (gauge)	Total Index usage reported at 5 minutes granularity per region <i>Shown as byte</i>	SATURATION		
azure.cosmosdb.integrated_cache_evicted_entries_size (count)	Size of the entries evicted from the integrated cache <i>Shown as byte</i>	SATURATION		
azure.cosmosdb.integrated_cache_item_expiration_count (count)	The number of items evicted from the integrated cache due to TTL expiration	SATURATION		
azure.cosmosdb.integrated_cache_item_hit_rate (rate)	The number of point reads that used the integrated cache divided by the number of point reads routed through the dedicated gateway with eventual consistency <i>Shown as percent</i>	TRAFFIC		
azure.cosmosdb.integrated_cache_query_expiration_count (count)	The number of queries evicted from the integrated cache due to TTL expiration	TRAFFIC		
azure.cosmosdb.integrated_cache_query_hit_rate (rate)	The number of queries that used the integrated cache divided by the number of queries routed through the dedicated gateway with eventual consistency <i>Shown as percent</i>	TRAFFIC		
azure.cosmosdb.metadata_requests (count)	Count of metadata requests. Cosmos DB maintains system metadata collection for each account that allows you to enumerate collections databases etc and their configurations. <i>Shown as request</i>	TRAFFIC		
azure.cosmosdb.mongo_collection_create (count)	Mongo Collection Created	TRAFFIC		
azure.cosmosdb.mongo_collection_delete (count)	Mongo Collection Deleted	TRAFFIC		

azure.cosmosdb. mongo_collection_throughput_update (count)	Mongo Collection Throughput Updated	TRAFFIC		
azure.cosmosdb. mongo_collection_update (count)	Mongo Collection Updated	TRAFFIC		
azure.cosmosdb. mongo_database_delete (count)	Mongo Database Deleted	TRAFFIC		
azure.cosmosdb. mongo_database_throughput_update (count)	Mongo Database Throughput Updated	TRAFFIC		
azure.cosmosdb. mongodb_database_create (count)	Mongo Database Created	TRAFFIC		
azure.cosmosdb. mongodb_database_update (count)	Mongo Database Updated	TRAFFIC		
azure.cosmosdb. mongo_request_charge (count)	Mongo Request Units Consumed	TRAFFIC		
azure.cosmosdb. mongo_requests (count)	The number of Mongo requests made <i>Shown as request</i>	TRAFFIC		
azure.cosmosdb. normalized_ru_consumption (count)	Max RU consumption percentage per minute <i>Shown as percent</i>	SATURATION		
azure.cosmosdb. provisioned_throughput (count)	Provisioned Throughput	TRAFFIC		
azure.cosmosdb. region_failover (count)	Region Failed Over	TRAFFIC		
azure.cosmosdb. remove_region (count)	Region Removed	TRAFFIC		
azure.cosmosdb. replication_latency (count)	P99 Replication Latency across source and target regions for geo-enabled accounts <i>Shown as millisecond</i>	LATENCY		
azure.cosmosdb. server_side_latency (count)	Server Side Latency <i>Shown as millisecond</i>	LATENCY		 99.9%
azure.cosmosdb. service_availability (count)	Account requests availability at one hour, day, or month granularity <i>Shown as percent</i>	ERROR		 99.9%
azure.cosmosdb. sql_container_create (count)	Sql Container Created	TRAFFIC		
azure.cosmosdb. sql_container_delete (count)	Sql Container Deleted	TRAFFIC		
azure.cosmosdb. sql_container_throughput_update (count)	Sql Container Throughput Updated	TRAFFIC		
azure.cosmosdb. sql_container_update (count)	Sql Container Updated	TRAFFIC		
azure.cosmosdb. sql_database_create (count)	Sql Database Created	TRAFFIC		

azure.cosmosdb. sql_database_delete (count)	Sql Database Deleted	TRAFFIC	✗	
azure.cosmosdb. sql_database_throughput_update (count)	Sql Database Throughput Updated	TRAFFIC	✗	
azure.cosmosdb. sql_database_update (count)	Sql Database Updated	TRAFFIC	✗	
azure.cosmosdb. table_table_create (count)	AzureTable Table Created	TRAFFIC	✗	
azure.cosmosdb. table_table_delete (count)	AzureTable Table Deleted	TRAFFIC	✗	
azure.cosmosdb. table_table_throughput_update (count)	AzureTable Table Throughput Updated	TRAFFIC	✗	
azure.cosmosdb. table_table_update (count)	AzureTable Table Updated	TRAFFIC	✗	
azure.cosmosdb. total_requests (count)	The number of requests made <i>Shown as request</i>	TRAFFIC	✓	✓ 99.9%
azure.cosmosdb. total_request_units (count)	Request Units consumed	TRAFFIC	✗	
azure.cosmosdb. update_account_keys (count)	Account Keys Updated	TRAFFIC	✗	
azure.cosmosdb. update_account_network_settings (count)	Account Network Settings Updated	TRAFFIC	✗	
azure.cosmosdb. update_account_replication_settings (count)	Account Replication Settings Updated	TRAFFIC	✗	
azure.cosmosdb. update_diagnostics_settings (count)	Account Diagnostic Settings Updated	TRAFFIC	✗	

AWS

▼ EC2 Instance EC2 Monitoring

Terraform Module Repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-ec2-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.ec2.cpuscredit_balance (gauge)	Number of CPU credits that an instance has accumulated. <i>Shown as unit</i>	SATURATION	✗
aws.ec2.cpuscredit_usage (gauge)	Number of CPU credits consumed. <i>Shown as unit</i>	SATURATION	✗
aws.ec2.cpusurplus_credit_balance (gauge)	The number of surplus credits that have been spent by an unlimited instance when its CPUCreditBalance value is zero. <i>Shown as unit</i>	SATURATION	✗

aws.ec2.cpusurplus_credits_charged (gauge)	The number of spent surplus credits that are not paid down by earned CPU credits, and which thus incur an additional charge. <i>Shown as unit</i>	SATURAT ION	✗
aws.ec2.cpuutilization (gauge)	Average percentage of allocated EC2 compute units that are currently in use on the instance. <i>Shown as percent</i>	SATURAT ION	✓
aws.ec2.cpuutilization.maximum (gauge)	Maximum percentage of allocated EC2 compute units that are currently in use on the instance. <i>Shown as percent</i>	SATURAT ION	✗
aws.ec2.disk_read_bytes (gauge)	Bytes read from all ephemeral disks available to the instance. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.disk_read_ops (gauge)	Completed read operations from all ephemeral disks available to the instance. <i>Shown as operation</i>	SATURAT ION	✗
aws.ec2.disk_write_bytes (gauge)	Bytes written to all ephemeral disks available to the instance. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.disk_write_ops (gauge)	Completed write operations to all ephemeral disks available to the instance. <i>Shown as operation</i>	SATURAT ION	✗
aws.ec2.ebsbyte_balance (gauge)	Available only for the smaller instance sizes. Provides information about the percentage of throughput credits remaining in the burst bucket. This metric is available for basic monitoring only. <i>Shown as percent</i>	SATURAT ION	✗
aws.ec2.ebsiobalance (gauge)	Available only for the smaller instance sizes. Provides information about the percentage of I/O credits remaining in the burst bucket. This metric is available for basic monitoring only. <i>Shown as percent</i>	SATURAT ION	✗
aws.ec2.ebsread_bytes (gauge)	Bytes read from all EBS volumes attached to the instance in a specified period of time. <i>Shown as byte</i>	SATURAT ION	✓
aws.ec2.ebsread_ops (count)	Completed read operations from all Amazon EBS volumes attached to the instance in a specified period of time. <i>Shown as operation</i>	SATURAT ION	✓
aws.ec2.ebswrite_bytes (gauge)	Bytes written to all EBS volumes attached to the instance in a specified period of time. <i>Shown as byte</i>	SATURAT ION	✓
aws.ec2.ebswrite_ops (count)	Completed write operations to all EBS volumes attached to the instance in a specified period of time. <i>Shown as operation</i>	SATURAT ION	✓
aws.ec2.host_ok (gauge)	1 if the instance's system status is ok.	ERROR	✓
aws.ec2.instance_age (gauge)	Time since instance launch <i>Shown as second</i>	SATURAT ION	✗
aws.ec2.network_in (gauge)	Average number of bytes received on all network interfaces by the instance. <i>Shown as byte</i>	SATURAT ION	✓
aws.ec2.network_in.maximum (gauge)	Maximum number of bytes received on all network interfaces by the instance. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.network_out (gauge)	Average number of bytes sent out on all network interfaces by the instance. <i>Shown as byte</i>	SATURAT ION	✓
aws.ec2.network_out.maximum (gauge)	Maximum number of bytes sent out on all network interfaces by the instance. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.network_packets_in (gauge)	Number of packets received on all network interfaces by the instance <i>Shown as packet</i>	TRAFFIC	✗

aws.ec2.network_packets_out (gauge)	Number of packets sent out on all network interfaces by the instance <i>Shown as packet</i>	TRAFFIC	✗
aws.ec2.status_check_failed_instance (gauge)	0 if the instance has passed the EC2 instance status check.	ERROR	✓
aws.ec2.status_check_failed_system (gauge)	0 if the instance has passed the EC2 system status check.	ERROR	✗
aws.ec2.status_check_failed (gauge)	1 if one of the status checks failed.	ERROR	✗
aws.ec2.ebsread_ops (count)	Average completed read operations from all Amazon EBS volumes attached to the instance for Nitro-based instances. <i>Shown as operation</i>	TRAFFIC	✗
aws.ec2.ebsread_ops.sum (count)	Total completed read operations from all Amazon EBS volumes attached to the instance for Nitro-based instances. <i>Shown as operation</i>	TRAFFIC	✗
aws.ec2.ebswrite_ops (gauge)	Average completed write operations to all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as operation</i>	TRAFFIC	✗
aws.ec2.ebswrite_ops.sum (gauge)	Total completed write operations to all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as operation</i>	TRAFFIC	✗
aws.ec2.ebsread_bytes (gauge)	Average bytes read from all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.ebsread_bytes.sum (gauge)	Total bytes read from all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.ebswrite_bytes (gauge)	Average bytes written to all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.ebswrite_bytes.sum (gauge)	Total bytes written to all EBS volumes attached to the instance for Nitro-based instances. <i>Shown as byte</i>	SATURAT ION	✗
aws.ec2.ebsiobalance (gauge)	Percentage of I/O credits remaining in the burst bucket for Nitro-based instances." <i>Shown as percent</i>	SATURAT ION	✗
aws.ec2.ebsbyte_balance (gauge)	Percentage of throughput credits remaining in the burst bucket for Nitro-based instances. <i>Shown as percent</i>	SATURAT ION	✗

Autoscaling

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.autoscaling.group_desired_capacity (gauge)	The number of instances that the Auto Scaling group attempts to maintain.	SATURAT ION	✓
aws.autoscaling.group_in_service_instances (gauge)	<i>The number of instances that are running as part of the Auto Scaling group. This metric does not include instances that are pending or terminating.</i>	SATURAT ION	✓
aws.autoscaling.group_max_size (gauge)	<i>The maximum size of the Auto Scaling group.</i>	SATURAT ION	✓
aws.autoscaling.group_min_size (gauge)	<i>The minimum size of the Auto Scaling group.</i>	SATURAT ION	✓
aws.autoscaling.group_pending_instances (gauge)	<i>The number of instances that are pending. A pending instance is not yet in service. This metric does not include instances that are in service or terminating.</i>	SATURAT ION	✓
aws.autoscaling.group_terminating_instances (gauge)	<i>The number of instances that are in the process of terminating. This metric does not include instances that are in service or pending.</i>	SATURAT ION	✓
aws.autoscaling.group_total_instances (gauge)	The total number of instances in the Auto Scaling group. This metric identifies the number of instances that are in service and/or pending and/or terminating.	SATURAT ION	✓


▼ AWS ELB (Elastic Load Balancers)

ELB Monitoring

Terraform Module Repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-elb-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.applicationelb.active_connection_count (count)	The total number of concurrent TCP connections active from clients to the load balancer and from the load balancer to targets. <i>Shown as connection</i>	TRAFFIC	✗
aws.applicationelb.client_tlsnegotiation_error_count (count)	Number of TLS negotiation errors <i>Shown as error</i>	ERROR	✓
aws.applicationelb.consumed_lcus (gauge)	The number of load balancer capacity units (LCU) used by your load balancer. <i>Shown as unit</i>	SATURATION	✗
aws.applicationelb.elb_auth_error (count)	The number of user authentications that could not be completed because an authenticate action was misconfigured, the load balancer couldn't establish a connection with the IdP, or the load balancer couldn't complete the authentication flow due to an internal error. <i>Shown as error</i>	ERROR	✗
aws.applicationelb.elb_auth_failure (count)	The number of user authentications that could not be completed because the IdP denied access to the user or an authorization code was used more than once. <i>Shown as error</i>	SATURATION	✗
aws.applicationelb.elb_auth_latency (gauge)	The time elapsed, in milliseconds, to query the IdP for the ID token and user info. If one or more of these operations fail, this is the time to failure. <i>Shown as millisecond</i>	LATENCY	✗
aws.applicationelb.elb_auth_refresh_token_success (count)	The number of times the load balancer successfully refreshed user claims using a refresh token provided by the IdP. <i>Shown as success</i>	TRAFFIC	✗
aws.applicationelb.elb_auth_success (count)	The number of authenticate actions that were successful. <i>Shown as success</i>	TRAFFIC	✗
aws.applicationelb.elb_authuser_claims_size_exceeded (count)	The number of times that a configured IdP returned user claims that exceeded 11K bytes in size.	TRAFFIC	✗
aws.applicationelb.healthy_host_count (gauge)	Average number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✓
aws.applicationelb.healthy_host_count_deduped (count)	The number of healthy instances per Availability Zone, regardless of if the Cross-Zone Load Balancing option is enabled or not. <i>Shown as host</i>	SATURATION	✓
aws.applicationelb.healthy_host_count.maximum (gauge)	Maximum number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.applicationelb.healthy_host_count.minimum (gauge)	Minimum number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.applicationelb.httpcode_elb_3xx (count)	The number of HTTP 3XX redirection codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✓
aws.applicationelb.httpcode_elb_4xx (count)	Number of HTTP 4XX client error codes generated by the load balancer. <i>Shown as response</i>	ERROR	✓
aws.applicationelb.httpcode_elb_5_0_0 (count)	The number of HTTP 500 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✗
aws.applicationelb.httpcode_elb_5_0_2 (count)	The number of HTTP 502 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✓

aws.applicationelb.httpcode_elb_5_0_3 (count)	The number of HTTP 503 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_elb_5_0_4 (count)	The number of HTTP 504 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_elb_5xx (count)	Number of HTTP 5XX client error codes generated by the load balancer. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_target_2xx (count)	Number of HTTP 2XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_target_3xx (count)	Number of HTTP 3XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_target_4xx (count)	Number of HTTP 4XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_target_5xx (count)	Number of HTTP 5XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	
aws.applicationelb.httpcode_redirect (count)	The number of redirect actions that were successful. <i>Shown as response</i>	TRAFFIC	
aws.applicationelb.httpfixed_response (count)	The number of fixed-response actions that were successful. <i>Shown as response</i>	TRAFFIC	
aws.applicationelb.httppredirect (count)	The number of redirect actions that were successful.	TRAFFIC	
aws.applicationelb.httppredirect_url_limit_exceeded (count)	The number of redirect actions that couldn't be completed because the URL in the response location header is larger than 8K.	SATURATION	
aws.applicationelb.ipv_6processed_bytes (count)	The total number of bytes processed by the load balancer over IPv6. <i>Shown as byte</i>	TRAFFIC	
aws.applicationelb.ipv_6request_count (count)	The number of IPv6 requests received by the load balancer. <i>Shown as request</i>	TRAFFIC	
aws.applicationelb.lambda_internal_error (count)	The number of requests to a Lambda function that failed because of an issue internal to the load balancer or AWS Lambda. <i>Shown as request</i>	ERROR	
aws.applicationelb.lambda_target_processed_bytes (gauge)	The total number of bytes processed by the load balancer for requests to and responses from a Lambda function. <i>Shown as byte</i>	TRAFFIC	
aws.applicationelb.lambda_user_error (count)	The number of requests to a Lambda function that failed because of an issue with the Lambda function. <i>Shown as request</i>	ERROR	
aws.applicationelb.new_connection_count (count)	The total number of new TCP connections established from clients to the load balancer and from the load balancer to targets. <i>Shown as connection</i>	TRAFFIC	
aws.applicationelb.non_sticky_request_count (count)	The number of requests where the load balancer chose a new target because it couldn't use an existing sticky session. <i>Shown as request</i>	SATURATION	
aws.applicationelb.processed_bytes (count)	The total number of bytes processed by the load balancer over IPv4 and IPv6. <i>Shown as byte</i>	TRAFFIC	
aws.applicationelb.rejected_connection_count (count)	The number of connections that were rejected because the load balancer had reached its maximum number of connections. <i>Shown as connection</i>	SATURATION	
aws.applicationelb.request_count (count)	Total number of completed requests that were received and routed to the registered instances. <i>Shown as request</i>	TRAFFIC	

aws.applicationelb.request_count_per_target (count)	The average number of requests received by each target in a target group. <i>Shown as request</i>	TRAFFIC	✓
aws.applicationelb.rule_evaluations (count)	The number of rules processed by the load balancer given a request rate averaged over an hour.	SATURATION	✗
aws.applicationelb.desync_mitigation_mode_non_compliant_request (count)	The number of requests that do not comply with RFC 7230. <i>Shown as request</i>	ERROR	✗
aws.applicationelb.target_connection_error_count (count)	Number of connections that were not successfully established between the load balancer and the registered instances. <i>Shown as error</i>	ERROR	✗
aws.applicationelb.target_response_time.average (gauge)	Average time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	LATENCY	✓
aws.applicationelb.target_response_time.maximum (gauge)	Maximum time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	LATENCY	✗
aws.applicationelb.target_response_time.p50 (gauge)	50th percentile of the time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	LATENCY	✗
aws.applicationelb.target_response_time.p90 (gauge)	90th percentile of the time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	LATENCY	✗
aws.applicationelb.target_response_time.p95 (gauge)	95th percentile of the time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	LATENCY	✓
aws.applicationelb.target_response_time.p99 (gauge)	99th percentile of the time elapsed after the request leaves the load balancer until a response is received. <i>Shown as second</i>	ERROR	✗
aws.applicationelb.target_tlsnegotiation_error_count (count)	The number of TLS connections initiated by the load balancer that did not establish a session with the target. <i>Shown as connection</i>	SATURATION	✗
aws.applicationelb.un_healthy_host_count (gauge)	Average number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✓
aws.applicationelb.un_healthy_host_count_deduped (count)	The number of unhealthy instances per Availability Zone, regardless of if the Cross-Zone Load Balancing option is enabled or not. <i>Shown as host</i>	SATURATION	✓
aws.applicationelb.un_healthy_host_count.maximum (gauge)	Maximum number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.applicationelb.un_healthy_host_count.minimum (gauge)	Minimum number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.elb.active_connection_count (count)	The total number of concurrent TCP connections active from clients to the load balancer and from the load balancer to targets. <i>Shown as connection</i>	TRAFFIC	✗
aws.elb.backend_connection_errors (rate)	Number of connections that were not successfully established between the load balancer and the registered instances. <i>Shown as error</i>	TRAFFIC	✗
aws.elb.client_tlsnegotiation_error_count (count)	Number of TLS negotiation errors <i>Shown as error</i>	ERROR	✗
aws.elb.consumed_lbcapacity_units (gauge)	Number of ELB capacity units consumed <i>Shown as unit</i>	SATURATION	✗
aws.elb.consumed_lcus (gauge)	The number of load balancer capacity units (LCU) used by your load balancer. <i>Shown as unit</i>	SATURATION	✗
aws.elb.estimated_albactive_connection_count (count)	The estimated total number of concurrent TCP connections active from clients to the load balancer and from the load balancer to targets. <i>Shown as connection</i>	TRAFFIC	✗

aws.elb.estimated_albconsumed_lcus (gauge)	The estimated total number of load balancer capacity units (LCU) used by an gApplication Load Balancer. <i>Shown as unit</i>	SATURATION	✖
aws.elb.estimated_albnew_connection_count (count)	The estimated total number of new TCP connections established from clients to the load balancer and from the load balancer to targets <i>Shown as connection</i>	SATURATION	✖
aws.elb.estimated_processed_bytes (count)	The estimated total number of bytes processed by an Application Load Balancer. <i>Shown as byte</i>	TRAFFIC	✖
aws.elb.healthy_host_count (gauge)	Average number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✖
aws.elb.healthy_host_count_deduped (count)	The number of healthy instances per Availability Zone, regardless of if the Cross-Zone Load Balancing option is enabled or not. <i>Shown as host</i>	SATURATION	✖
aws.elb.healthy_host_count.maximum (gauge)	Maximum number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✖
aws.elb.healthy_host_count.minimum (gauge)	Minimum number of healthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✖
aws.elb.httpcode_backend_2xx (rate)	Number of HTTP 2XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_backend_3xx (rate)	Number of HTTP 3XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_backend_4xx (rate)	Number of HTTP 4XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_backend_5xx (rate)	Number of HTTP 5XX response codes generated by registered instances. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_4xx (rate)	Number of HTTP 4XX client error codes generated by the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_5_0_0 (count)	The number of HTTP 500 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_5_0_2 (count)	The number of HTTP 502 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_5_0_3 (count)	The number of HTTP 503 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_5_0_4 (count)	The number of HTTP 504 error codes that originate from the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_elb_5xx (rate)	Number of HTTP 5XX client error codes generated by the load balancer. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_target_2xx (count)	Number of HTTP 2XX response codes generated by the targets. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_target_3xx (count)	Number of HTTP 3XX response codes generated by the targets. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_target_4xx (count)	Number of HTTP 4XX response codes generated by the targets. <i>Shown as response</i>	ERROR	✖

aws.elb.httpcode_target_5xx (count)	Number of HTTP 5XX response codes generated by the targets. <i>Shown as response</i>	ERROR	✖
aws.elb.httpcode_redirect (count)	The number of redirect actions that were successful. <i>Shown as response</i>	SATURATION	✖
aws.elb.ipv_6processed_bytes (count)	The total number of bytes processed by the load balancer over IPv6. <i>Shown as byte</i>	TRAFFIC	✖
aws.elb.ipv_6request_count (count)	The number of IPv6 requests received by the load balancer. <i>Shown as request</i>	TRAFFIC	✖
aws.elb.latency (gauge)	Average time elapsed after the request leaves the load balancer until a response is received. (ELB v1) <i>Shown as second</i>	LATENCY	✖
aws.elb.latency.maximum (gauge)	Maximum time elapsed after the request leaves the load balancer until a response is received. (ELB v1) <i>Shown as second</i>	LATENCY	✖
aws.elb.latency.minimum (gauge)	Minimum time elapsed after the request leaves the load balancer until a response is received. (ELB v1) <i>Shown as second</i>	LATENCY	✖
aws.elb.latency.p95 (gauge)	95th percentile of the time elapsed after the request leaves the load balancer until a response is received. (ELB v1) <i>Shown as second</i>	LATENCY	✖
aws.elb.latency.p99 (gauge)	99th percentile of the time elapsed after the request leaves the load balancer until a response is received. (ELB v1) <i>Shown as second</i>	LATENCY	✖
aws.elb.new_connection_count (count)	The total number of new TCP connections established from clients to the load balancer and from the load balancer to targets. <i>Shown as connection</i>	TRAFFIC	✖
aws.elb.processed_bytes (count)	The total number of bytes processed by the load balancer over IPv4 and IPv6. <i>Shown as byte</i>	TRAFFIC	✖
aws.elb.request_count (rate)	Total number of completed requests that were received and routed to the registered instances. <i>Shown as request</i>	TRAFFIC	✖
aws.elb.request_count_per_target (count)	The average number of requests received by each target in a target group. <i>Shown as request</i>	TRAFFIC	✖
aws.elb.rule_evaluations (count)	The number of rules processed by the load balancer given a request rate averaged over an hour.	TRAFFIC	✖
aws.elb.spillover_count (rate)	Total number of requests that were rejected because the queue was full. <i>Shown as request</i>	SATURATION	✖
aws.elb.spillover_count.maximum (rate)	Maximum number of requests that were rejected because the queue was full per load balancer node. <i>Shown as request</i>	SATURATION	✖
aws.elb.surge_queue_length (gauge)	Maximum number of requests that are pending submission to a registered instance. <i>Shown as request</i>	SATURATION	✖
aws.elb.target_connection_error_count (count)	Number of connections that were not successfully established between the load balancer and the registered instances. <i>Shown as error</i>	ERROR	✖
aws.elb.target_response_time.average (gauge)	Average time elapsed after the request leaves the load balancer until a response is received. (ELB v2) <i>Shown as second</i>	LATENCY	✖
aws.elb.target_response_time.maximum (gauge)	Maximum time elapsed after the request leaves the load balancer until a response is received. (ELB v2) <i>Shown as second</i>	LATENCY	✖
aws.elb.target_response_time.p95 (gauge)	95th percentile of the time elapsed after the request leaves the load balancer until a response is received. (ELB v2) <i>Shown as second</i>	LATENCY	✖

aws.elb.target_response_time.p99 (gauge)	99th percentile of the time elapsed after the request leaves the load balancer until a response is received. (ELB v2) <i>Shown as second</i>	LATENCY	✗
aws.elb.un_healthy_host_count (gauge)	Average number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.elb.un_healthy_host_count_deduped (count)	The number of unhealthy instances per Availability Zone, regardless of if the Cross-Zone Load Balancing option is enabled or not. <i>Shown as host</i>	SATURATION	✗
aws.elb.un_healthy_host_count.maximum (gauge)	Maximum number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.elb.un_healthy_host_count.minimum (gauge)	Minimum number of unhealthy instances in each Availability Zone. <i>Shown as host</i>	SATURATION	✗
aws.networkelb.active_flow_count (gauge)	The average number of active established connections from clients to targets <i>Shown as connection</i>	TRAFFIC	✓
aws.networkelb.active_flow_count_tls (count)	The total number of concurrent TLS flows (or connections) from clients to targets. <i>Shown as connection</i>	TRAFFIC	✗
aws.networkelb.active_flow_count.maximum (gauge)	The maximum number of active established connections from clients to targets <i>Shown as connection</i>	TRAFFIC	✗
aws.networkelb.active_flow_count.minimum (gauge)	The minimum number of active established connections from clients to targets <i>Shown as connection</i>	TRAFFIC	✗
aws.networkelb.client_tlsnegotiation_error_count (count)	The total number of TLS handshakes that failed during negotiation between a client and a TLS listener. <i>Shown as error</i>	ERROR	✓
aws.networkelb.consumed_lcus (count)	The number of LCUs used by the load balancer <i>Shown as unit</i>	SATURATION	✗
aws.networkelb.healthy_host_count (gauge)	Average number of healthy targets <i>Shown as host</i>	SATURATION	✓
aws.networkelb.healthy_host_count.maximum (gauge)	Maximum number of healthy targets <i>Shown as host</i>	SATURATION	✗
aws.networkelb.healthy_host_count.minimum (gauge)	Minimum number of healthy targets <i>Shown as host</i>	SATURATION	✗
aws.networkelb.new_flow_count (count)	The number of new TCP connections from clients to targets <i>Shown as connection</i>	TRAFFIC	✗
aws.networkelb.new_flow_count_tls (count)	The total number of new TLS flows (or connections) established from clients to targets in the time period. <i>Shown as connection</i>	TRAFFIC	✗
aws.networkelb.processed_bytes (count)	The number of LCUs used by the load balancer <i>Shown as byte</i>	TRAFFIC	✓
aws.networkelb.processed_bytes_tls (gauge)	The total number of bytes processed by TLS listeners. <i>Shown as byte</i>	TRAFFIC	✗
aws.networkelb.target_tlsnegotiation_error_count (count)	The total number of TLS handshakes that failed during negotiation between a TLS listener and a target. <i>Shown as error</i>	ERROR	✓
aws.networkelb.tcpclient_reset_count (count)	The number of reset (RST) packets created by a client and sent to a target <i>Shown as packet</i>	SATURATION	✓
aws.networkelb.tcpebreset_count (count)	The number of reset (RST) packets created by a load balancer <i>Shown as packet</i>	SATURATION	✓
aws.networkelb.tcptarget_reset_count (count)	The number of reset (RST) packets created by a target and sent to a client <i>Shown as packet</i>	SATURATION	✓
aws.networkelb.un_healthy_host_count (gauge)	Average number of unhealthy targets <i>Shown as host</i>	SATURATION	✓

aws.networkelb.un_healthy_host_count.maximum (gauge)	Maximum number of unhealthy targets <i>Shown as host</i>	SATURATION	✗
aws.networkelb.un_healthy_host_count.minimum (gauge)	Minimum number of unhealthy targets <i>Shown as host</i>	SATURATION	✗

▼ AWS S3

S3 Monitoring

Terraform Module repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-s3-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.s3.bucket_size_bytes (gauge)	The amount of data in bytes stored in a bucket in the Standard storage class, Standard - Infrequent Access (Standard_IA) storage class, or the Reduced Redundancy Storage (RRS) class. <i>Shown as byte</i>	SATURATION	✓
aws.s3.number_of_objects (gauge)	The total number of objects stored in a bucket for all storage classes except for the GLACIER storage class.	SATURATION	✓
aws.s3.all_requests (count)	The total number of HTTP requests made to a bucket, regardless of type.	TRAFFIC	✓
aws.s3.get_requests (count)	The number of HTTP GET requests made for objects in a bucket. This doesn't include list operations.		
aws.s3.put_requests (count)	The number of HTTP PUT requests made for objects in a bucket.		
aws.s3.delete_requests (count)	The number of HTTP DELETE requests made for objects in a bucket. This also includes Delete Multiple Objects requests.	TRAFFIC	✓
aws.s3.head_requests (count)	The number of HTTP HEAD requests made to a bucket.		
aws.s3.post_requests (count)	The number of HTTP POST requests made to a bucket.		
aws.s3.list_requests (count)	The number of HTTP requests that list the contents of a bucket.		
aws.s3.bytes_downloaded (count)	The total number bytes downloaded from the bucket. <i>Shown as byte</i>	TRAFFIC	✓
aws.s3.bytes_uploaded (count)	The total number bytes uploaded to the bucket. <i>Shown as byte</i>	TRAFFIC	✓
aws.s3.4xx_errors (count)	The total number of HTTP 4xx server error status code requests made to a bucket	ERROR	✓
aws.s3.5xx_errors (count)	The total number of HTTP 5xx server error status code requests made to a bucket	ERROR	✓
aws.s3.first_byte_latency (gauge)	The average per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>	LATENCY	✓
aws.s3.first_byte_latency.minimum (gauge)	The minimum per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.first_byte_latency.maximum (gauge)	The maximum per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.first_byte_latency.p50 (gauge)	The 50 percentile per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		

aws.s3.first_byte_latency.p90 (gauge)	The 90 percentile per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.first_byte_latency.p95 (gauge)	The 95 percentile per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.first_byte_latency.p99 (gauge)	The 99 percentile per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.first_byte_latency.p99.99 (gauge)	The 99.99 percentile per-request time from the complete request being received by a bucket to when the response starts to be returned. <i>Shown as millisecond</i>		
aws.s3.total_request_latency (gauge)	The average elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>	LATENCY	✓
aws.s3.total_request_latency.minimum (gauge)	The minimum elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.maximum (gauge)	The maximum elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.p50 (gauge)	The 50 percentile elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.p90 (gauge)	The 90 percentile elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.p95 (gauge)	The 95 percentile elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.p99 (gauge)	The 99 percentile elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.total_request_latency.p99.99 (gauge)	The 99.99 percentile elapsed per-request time from the first byte received to the last byte sent to a bucket <i>Shown as millisecond</i>		
aws.s3.replication_latency.maximum (gauge)	The maximum number of seconds by which the replication destination Region is behind the source Region <i>Shown as second</i>		
aws.s3.bytes_pending_replication.maximum (gauge)	The total number of bytes of objects pending replication <i>Shown as byte</i>		
aws.s3.operations_pending_replication.maximum (gauge)	The number of operations pending replication		

▼ AWS API Gateway

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.apigateway.4xxerror (count)	The number of client-side errors <i>Shown as operation</i>	ERROR	✓
aws.apigateway.5xxerror (count)	The number of server-side errors <i>Shown as operation</i>	ERROR	✓
aws.apigateway.cache_hit_count (count)	The number of requests served from the API cache <i>Shown as operation</i>	SATURATION	✗
aws.apigateway.cache_miss_count (count)	The number of requests served from the back end when API caching is enabled <i>Shown as operation</i>	SATURATION	✗

aws.apigateway.client_error.sum (count)	The total number of requests that have a 4XX response returned by API Gateway before the integration is invoked. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.client_error (count)	The average number of requests that have a 4XX response returned by API Gateway before the integration is invoked. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.connect_count.sum (count)	The total number of messages sent to the \$connect route integration. <i>Shown as operation</i>	SATURATION	✗
aws.apigateway.connect_count (count)	The average number of messages sent to the \$connect route integration. <i>Shown as operation</i>	SATURATION	✗
aws.apigateway.count (count)	The number call to API methods <i>Shown as operation</i>	SATURATION	✓
aws.apigateway.execution_error.sum (count)	Total errors that occurred when calling the integration. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.execution_error (count)	Average errors that occurred when calling the integration. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.integration_error.sum (count)	The total number of requests that return a 4XX/5XX response from the integration. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.integration_error (count)	The average number of requests that return a 4XX/5XX response from the integration. <i>Shown as operation</i>	ERROR	✗
aws.apigateway.integration_latency.maximum (gauge)	The maximum time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.integration_latency.minimum (gauge)	The minimum time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.integration_latency.p90 (gauge)	The 90th percentile time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✗
aws.apigateway.integration_latency.p95 (gauge)	The 95th percentile time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✗
aws.apigateway.integration_latency.p99 (gauge)	The 99th percentile time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✗
aws.apigateway.integration_latency (gauge)	The time between when API Gateway relays a request to the back end and when it receives a response from the back end. <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.latency.maximum (gauge)	The maximum time between when requests are received and when responses returned <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.latency.minimum (gauge)	The minimum time between when requests are received and when responses returned <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.latency.p90 (gauge)	The 90th percentile time between when requests are received and when responses returned <i>Shown as millisecond</i>	SATURATION	✗
aws.apigateway.latency.p95 (gauge)	The 95th percentile time between when requests are received and when responses returned <i>Shown as millisecond</i>	SATURATION	✗
aws.apigateway.latency.p99 (gauge)	The 99th percentile time between when requests are received and when responses returned <i>Shown as millisecond</i>	SATURATION	✗

aws.apigateway.latency (gauge)	The time between when API Gateway receives a request from a client and when it returns a response to the client. The latency includes the integration_latency and other API Gateway overhead. <i>Shown as millisecond</i>	SATURATION	✓
aws.apigateway.message_count.sum (count)	The total number of messages sent to the WebSocket API, either from or to the client. <i>Shown as operation</i>	SATURATION	✗
aws.apigateway.message_count (count)	The average number of messages sent to the WebSocket API, either from or to the client. <i>Shown as operation</i>	SATURATION	✗

▼ AWS Web Application Firewall (WAF)

WAF Monitoring

Terraform Module repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-waf-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.wafv2.allowed_requests (gauge)	The number of allowed web requests. <i>Shown as request</i>	SATURATION	✓
aws.wafv2.blocked_requests (gauge)	The number of blocked web requests. <i>Shown as request</i>	SATURATION	✓
aws.wafv2.counted_requests (gauge)	The number of counted web requests. <i>Shown as request</i>	SATURATION	✓
aws.wafv2.passed_requests (gauge)	The number of passed web requests. <i>Shown as request</i>	SATURATION	✗
aws.waf.allowed_requests (gauge)	The number of allowed web requests. <i>Shown as request</i>	SATURATION	✗
aws.waf.blocked_requests (gauge)	The number of blocked web requests. <i>Shown as request</i>	SATURATION	✗
aws.waf.counted_requests (gauge)	The number of counted web requests. <i>Shown as request</i>	SATURATION	✗
aws.waf.passed_requests (gauge)	The number of passed web requests. <i>Shown as request</i>	SATURATION	✗
waf.allowed_requests (gauge)	The number of allowed web requests. <i>Shown as request</i>	SATURATION	✗
waf.blocked_requests (gauge)	The number of blocked web requests. <i>Shown as request</i>	SATURATION	✗
waf.counted_requests (gauge)	The number of counted web requests. <i>Shown as request</i>	SATURATION	✗
waf.passed_requests (gauge)	The number of passed web requests. <i>Shown as request</i>	SATURATION	✗

▼ AWS RDS (Relational Database Service)

RDS Monitoring

Terraform Module repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-rds-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.rds.bin_log_disk_usage (gauge)	Amount of disk space occupied by binary logs on the master. (Standard) <i>Shown as byte</i>		
aws.rds.burst_balance (gauge)	The percent of General Purpose SSD (gp2) burst-bucket I/O credits available. (Standard) <i>Shown as percent</i>		
aws.rds.cpucredit_balance (gauge)	[T2 instances] Number of CPU credits that an instance has accumulated. (Standard)		

aws.rds.cpucredit_usage (gauge)	[T2 instances] Number of CPU credits consumed. (Standard)		
aws.rds.cpusurplus_credit_balance (gauge)	The number of surplus credits that have been spent by an unlimited instance when its CPUCreditBalance value is zero.		
aws.rds.cpusurplus_credits_charged (gauge)	The number of spent surplus credits that are not paid down by earned CPU credits, and which thus incur an additional charge.		
aws.rds.cpuutilization (gauge)	Percentage of CPU utilization. Recommended metric for standard monitoring. (Standard) <i>Shown as percent</i>	SATURATION	✓
aws.rds.database_connections (gauge)	Number of database connections in use. (Standard) <i>Shown as connection</i>	TRAFFIC	✓
aws.rds.dbload (gauge)	The number of active sessions for the DB engine (Performance Insights must be enabled). <i>Shown as session</i>		
aws.rds.dbload_cpu (gauge)	The number of active sessions where the wait event type is CPU (Performance Insights must be enabled). <i>Shown as session</i>	TRAFFIC	✓
aws.rds.dbload_non_cpu (gauge)	The number of active sessions where the wait event type is not CPU (Performance Insights must be enabled). <i>Shown as session</i>	TRAFFIC	✓
aws.rds.disk_queue_depth (gauge)	Number of outstanding IOs (read/write requests) waiting to access the disk. (Standard) <i>Shown as request</i>	SATURATION	✓
aws.rds.failed_sqlserver_agent_jobs_count (count)	The number of failed SQL Server Agent jobs during the last minute. <i>Shown as minute</i>		
aws.rds.free_storage_space (gauge)	Amount of available storage space. (Standard) <i>Shown as byte</i>		
aws.rds.freeable_memory (gauge)	Amount of available random access memory. (Standard) <i>Shown as byte</i>	SATURATION	✓
aws.rds.maximum_used_transaction_ids (count)	The maximum transaction ID that has been used. Applies to PostgreSQL.		
aws.rds.network_receive_throughput (rate)	Incoming (Receive) network traffic on the DB instance. (Standard) <i>Shown as byte</i>	TRAFFIC	✓
aws.rds.network_transmit_throughput (rate)	Outgoing (Transmit) network traffic on the DB instance. (Standard) <i>Shown as byte</i>	TRAFFIC	✓
aws.rds.oldest_replication_slot_lag (gauge)	The lagging size of the replica lagging the most in terms of WAL data received. Applies to PostgreSQL. <i>Shown as byte</i>		
aws.rds.read_iops (rate)	Average number of disk read I/O operations. (Standard) <i>Shown as operation</i>		
aws.rds.read_latency (gauge)	Average amount of time taken per disk read I/O operation. (Standard) <i>Shown as second</i>		
aws.rds.read_throughput (rate)	Average number of bytes read from disk. (Standard) <i>Shown as byte</i>		
aws.rds.replica_lag (gauge)	Amount of time a Read Replica DB Instance lags behind the source DB Instance.(Standard) <i>Shown as second</i>		
aws.rds.swap_usage (gauge)	Amount of swap space used on the DB Instance. (Standard) <i>Shown as byte</i>		
aws.rds.total_storage_space (gauge)	Total amount of storage available on an instance. (Standard) <i>Shown as byte</i>		
aws.rds.transaction_logs_generation (gauge)	The size of transaction logs generated per second. Applies to PostgreSQL. <i>Shown as byte</i>		

aws.rds.write_iops (rate)	Average number of disk write I/O operations per second. (Standard) <i>Shown as operation</i>		
aws.rds.write_latency (gauge)	Average amount of time taken per disk write I/O operation. (Standard) <i>Shown as second</i>		
aws.rds.write_throughput (rate)	Average number of bytes written to (Standard) <i>Shown as byte</i>		
aws.rds.active_transactions (gauge)	The average rate of current transactions executing on a DB instance. (Standard, Aurora-MySQL only) <i>Shown as transaction</i>		
aws.rds.aurora_binlog_replica_lag (gauge)	The amount of time a replica DB cluster running on Aurora with MySQL compatibility lags behind the source DB cluster. (Standard, Aurora-MySQL only) <i>Shown as second</i>		
aws.rds.aurora_replica_lag_maximum (gauge)	The maximum amount of lag between the primary instance and each Aurora instance in the DB cluster. (Standard, Aurora only) <i>Shown as millisecond</i>		
aws.rds.aurora_replica_lag_minimum (gauge)	The minimum amount of lag between the primary instance and each Aurora instance in the DB cluster. (Standard, Aurora only) <i>Shown as millisecond</i>		
aws.rds.aurora_replica_lag (gauge)	The average lag when replicating updates from the primary instance. (Standard, Aurora only) <i>Shown as millisecond</i>		
aws.rds.backup_retention_period_storage_used (gauge)	The amount of backup storage used for storing continuous backups at the current time (Aurora). <i>Shown as gibibyte</i>		
aws.rds.blocked_transactions (count)	The average rate of transactions in the database that are blocked. (Standard, Aurora-MySQL only) <i>Shown as transaction</i>		
aws.rds.buffer_cache_hit_ratio (gauge)	The percentage of requests that are served by the Buffer cache. (Standard, Aurora only) <i>Shown as percent</i>		
aws.rds.commit_latency (gauge)	The amount of latency for committed transactions. (Standard, Aurora only) <i>Shown as millisecond</i>		
aws.rds.commit_throughput (rate)	The average rate of committed transactions. (Standard, Aurora only) <i>Shown as transaction</i>		
aws.rds.cpuutilization.guest (gauge)	The percentage of CPU in use by guest programs. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.idle (gauge)	The percentage of CPU that is idle. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.interrupt (gauge)	The percentage of CPU in use by software interrupts. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.kernel (gauge)	The percentage of CPU in use by the kernel. (Enhanced, SQL Server Only) <i>Shown as percent</i>		
aws.rds.cpuutilization.nice (gauge)	The percentage of CPU in use by programs running at lowest priority. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.steal (gauge)	The percentage of CPU in use by other virtual machines. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.system (gauge)	The percentage of CPU in use by the kernel. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.total (gauge)	The total percentage of the CPU in use. This value excludes the nice value. Recommended metric for enhanced monitoring. (Enhanced) <i>Shown as percent</i>		

aws.rds.cpuutilization.user (gauge)	The percentage of CPU in use by user programs. (Enhanced) <i>Shown as percent</i>		
aws.rds.cpuutilization.wait (gauge)	The percentage of CPU unused while waiting for I/O access. (Enhanced) <i>Shown as percent</i>		
aws.rds.ddllatency (gauge)	The amount of latency for DDL requests (create/alter/drop). (Standard, Aurora-MySQL only) <i>Shown as millisecond</i>		
aws.rds.ddlthroughput (rate)	The average rate of DDL requests per second. (Standard, Aurora-MySQL only) <i>Shown as request</i>		
aws.rds.deadlocks (count)	The average number of deadlocks in the database per second. (Standard, Aurora only) <i>Shown as lock</i>		
aws.rds.delete_latency (gauge)	The average latency for delete queries. (Standard, Aurora only) <i>Shown as millisecond</i>		
aws.rds.delete_throughput (rate)	The average rate of delete queries. (Standard, Aurora only) <i>Shown as query</i>		
aws.rds.diskio.avgQueueLen (gauge)	The number of requests waiting in the I/O device's queue. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as request</i>		
aws.rds.diskio.avgReqSz (gauge)	The average request size. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.diskio.await (gauge)	The number of milliseconds required to respond to requests including queue time and service time. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as millisecond</i>		
aws.rds.dmlatency (gauge)	The average latency for inserts and updates and deletes. (Standard, Aurora-MySQL only) <i>Shown as millisecond</i>	LATENCY	✓
aws.rds.dmlthroughput (rate)	The average rate of inserts and updates and deletes. (Standard, Aurora-MySQL only) <i>Shown as operation</i>		
aws.rds.engine_uptime (gauge)	The amount of time that the DB instance has been active. (Standard, Aurora only) <i>Shown as second</i>		
aws.rds.free_local_storage (gauge)	The amount of local storage that is free on an instance. (Standard, Aurora only) <i>Shown as byte</i>		
aws.rds.insert_latency (gauge)	The amount of latency for insert queries. (Standard, Aurora-MySQL only) <i>Shown as millisecond</i>		
aws.rds.insert_throughput (rate)	The average rate of insert queries. (Standard, Aurora-MySQL only) <i>Shown as query</i>		
aws.rds.login_failures (count)	The average number of failed login attempts per second (Standard, Aurora-MySQL only) <i>Shown as operation</i>	ERROR	✓
aws.rds.network_throughput (rate)	The rate of network throughput sent and received from clients by each instance in the DB cluster. (Standard, Aurora only) <i>Shown as byte</i>		
aws.rds.queries (rate)	The average rate of queries. (Standard, Aurora-MySQL only) <i>Shown as query</i>		
aws.rds.rdsto_aurora_postgre_sqlreplica_lag (gauge)	The amount of lag in seconds when replicating updates from the primary RDS PostgreSQL instance to other nodes in the cluster. (Standard, Aurora-Postgres Only) <i>Shown as second</i>		
aws.rds.replication_slot_disk_usage (gauge)	The disk space used by replication slot files. (Standard, Postgres Only) <i>Shown as byte</i>		

aws.rds.result_set_cache_hit_ratio (gauge)	The percentage of requests that are served by the Resultset cache. (Standard, Aurora-MySQL only) <i>Shown as percent</i>		
aws.rds.select_latency (gauge)	The average latency for select queries. (Standard, Aurora-MySQL only) <i>Shown as millisecond</i>	LATENCY	✓
aws.rds.select_throughput (rate)	The average rate of select queries. (Standard, Aurora-MySQL only) <i>Shown as query</i>		
aws.rds.snapshot_storage_used (gauge)	The amount of backup storage used for storing manual snapshots beyond the backup retention period (Aurora). <i>Shown as gibibyte</i>		
aws.rds.total_backup_storage_billed (gauge)	The sum of BackupRetentionPeriodStorageUsed and SnapshotStorageUsed minus an amount of free backup storage which equals the size of the cluster volume for one day (Aurora). <i>Shown as gibibyte</i>		
aws.rds.transaction_logs_disk_usage (gauge)	Amount of disk space occupied by transaction logs. (Standard, Postgres Only) <i>Shown as byte</i>		
aws.rds.update_latency (gauge)	The average latency for update queries. (Standard, Aurora-MySQL only) <i>Shown as millisecond</i>		
aws.rds.update_throughput (rate)	The average rate of update queries. (Standard, Aurora-MySQL only) <i>Shown as query</i>		
aws.rds.volume_bytes_used (gauge)	The amount of storage in bytes used by your Aurora database. (Standard, Aurora only) <i>Shown as byte</i>		
aws.rds.volume_read_iops (count)	The number of billed read I/O operations from a cluster volume, reported at 5-minute intervals (Standard, Aurora only) <i>Shown as operation</i>		
aws.rds.volume_write_iops (count)	The average number of write disk I/O operations to the cluster volume reported at 5-minute intervals (Standard, Aurora only) <i>Shown as operation</i>		
aws.rds.diskio.readIOsPS (rate)	The rate of read operations. (Enhanced) <i>Shown as operation</i>		
aws.rds.diskio.readKb (gauge)	The total amount of data read. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.diskio.readKbPS (rate)	The rate that data is read. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.diskio.rrqmPS (rate)	The rate of merged read requests queue. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as request</i>		
aws.rds.diskio.tps (rate)	The rate of I/O transactions. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as transaction</i>		
aws.rds.diskio.util (gauge)	The percentage of CPU time during which requests were issued. The percentage of CPU time during which requests were issued. (Enhanced) <i>Shown as percent</i>		
aws.rds.diskio.writeIOsPS (rate)	The rate of write operations. (Enhanced) <i>Shown as operation</i>		
aws.rds.diskio.writeKb (gauge)	The total amount of data written. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.diskio.writeKbPS (rate)	The rate that data is written. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as kibibyte</i>		

aws.rds.diskio.wrqmPS (rate)	The rate of merged write requests queue. This metric is not available for Amazon Aurora. (Enhanced) <i>Shown as request</i>		
aws.rds.filesystem.maxFiles (gauge)	The maximum number of files that can be created for the file system. (Enhanced) <i>Shown as file</i>		
aws.rds.filesystem.total (gauge)	The total amount of disk space available for the file system. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.filesystem.used (gauge)	The amount of disk space used by files in the file system. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.filesystem.usedFilePercent (gauge)	The percentage of available files in use. (Enhanced) <i>Shown as percent</i>		
aws.rds.filesystem.usedFiles (gauge)	The number of files in the file system. (Enhanced) <i>Shown as file</i>		
aws.rds.filesystem.usedPercent (gauge)	The percentage of the file-system disk space in use. (Enhanced) <i>Shown as percent</i>		
aws.rds.load.1 (gauge)	The number of processes requesting CPU time over the last minute. (Enhanced) <i>Shown as process</i>		
aws.rds.load.15 (gauge)	The number of processes requesting CPU time over the last 15 minutes. (Enhanced) <i>Shown as process</i>		
aws.rds.load.5 (gauge)	The number of processes requesting CPU time over the last 5 minutes. (Enhanced) <i>Shown as process</i>		
aws.rds.memory.active (gauge)	The amount of assigned memory. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.buffers (gauge)	The amount of memory used for buffering I/O requests prior to writing to the storage device. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.cached (gauge)	The amount of memory used for caching file system-based I/O. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.commitLimitKb (gauge)	The maximum possible value for the commitTotKb metric. This value is the sum of the current pagefile size plus the physical memory available for pageable contents—excluding RAM that is assigned to non-pageable areas. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.commitPeakKb (gauge)	The largest value of the commitTotKb metric since the operating system was last started. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.commitTotKb (gauge)	The amount of pagefile-backed virtual address space in use, that is, the current commit charge. This value is composed of main memory (RAM) and disk (pagefiles). (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.dirty (gauge)	The amount of memory pages in RAM that have been modified but not written to their related data block in storage. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.free (gauge)	The amount of unassigned memory. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.hugePagesFree (gauge)	The number of free huge pages. (Enhanced) <i>Shown as page</i>		
aws.rds.memory.hugePagesRsvd (gauge)	The number of committed huge pages. (Enhanced) <i>Shown as page</i>		
aws.rds.memory.hugePagesSize (gauge)	The size for each huge pages unit. (Enhanced) <i>Shown as kibibyte</i>		

aws.rds.memory.hugePagesSurp (gauge)	The number of available surplus huge pages over the total. (Enhanced) <i>Shown as page</i>		
aws.rds.memory.hugePagesTotal (gauge)	The total number of huge pages for the system. (Enhanced) <i>Shown as page</i>		
aws.rds.memory.inactive (gauge)	The amount of inactive memory (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.kernNonpagedKb (gauge)	The amount of memory in the non-paged kernel pool. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.kernPagedKb (gauge)	The amount of memory in the paged kernel pool. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.kernTotKb (gauge)	The sum of the memory in the paged and non-paged kernel pools. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.mapped (gauge)	The total amount of file-system contents that is memory mapped inside a process address space. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.pageSize (gauge)	The size of a page. (Enhanced, SQL Server Only) <i>Shown as byte</i>		
aws.rds.memory.pageTables (gauge)	The amount of memory used by page tables. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.physAvailKb (gauge)	The amount of available physical memory. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.physTotKb (gauge)	The amount of physical memory. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.slab (gauge)	The amount of reusable kernel data structures. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.sqlServerTotKb (gauge)	The amount of memory committed to Microsoft SQL Server. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.sysCacheKb (gauge)	The amount of system cache memory. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.memory.total (gauge)	The total amount of memory. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.memory.writeback (gauge)	The amount of dirty pages in RAM that are still being written to the backing storage. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.network.rdBytesPS (gauge)	The number of bytes received per second. (Enhanced, SQL Server Only) <i>Shown as byte</i>		
aws.rds.network.rx (gauge)	The number of packets received. (Enhanced) <i>Shown as packet</i>		
aws.rds.network.tx (gauge)	The number of packets uploaded. (Enhanced) <i>Shown as packet</i>		
aws.rds.network.wrBytesPS (gauge)	The number of bytes sent per second. (Enhanced, SQL Server Only) <i>Shown as byte</i>		
aws.rds.process.cpuUsedPc (gauge)	The percentage of CPU used by the process. (Enhanced) <i>Shown as percent</i>		
aws.rds.process.memoryUsedPc (gauge)	The amount of memory used by the process. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.process.memUsedPc (gauge)	The percentage of total memory used by the process. (Enhanced, SQL Server Only) <i>Shown as percent</i>		

aws.rds.process.parentID (gauge)	The process identifier for the parent proces of the process. (Enhanced)		
aws.rds.process.pid (gauge)	The identifier of the process. This value is not present for processes that are owned by Amazon RDS. (Enhanced, SQL Server Only)		
aws.rds.process.ppid (gauge)	The process identifier for the parent of this process. This value is only present for child processes. (Enhanced, SQL Server Only)		
aws.rds.process.rss (gauge)	The amount of RAM allocated to the process. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.process.tgid (gauge)	The thread group identifier which is a number representing the process ID to which a thread belongs. This identifier is used to group threads from the same process. (Enhanced)		
aws.rds.process.tid (gauge)	The thread identifier. This value is only present for threads. The owning process can be identified by using the pid value. (Enhanced, SQL Server Only)		
aws.rds.process.virtKb (gauge)	The amount of virtual address space the process is using. Use of virtual address space does not necessarily imply corresponding use of either disk or main memory pages. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.process.vss (gauge)	The amount of virtual memory allocated to the process. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.process.workingSetKb (gauge)	The amount of memory in the private working set plus the amount of memory that is in use by the process and can be shared with other processes. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.process.workingSetPrivKb (gauge)	The amount of memory that is in use by a process, but can't be shared with other processes. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.process.workingSetShareableKb (gauge)	The amount of memory that is in use by a process and can be shared with other processes. (Enhanced, SQL Server Only) <i>Shown as kibibyte</i>		
aws.rds.swap.cached (gauge)	The amount of swap memory used as cache memory. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.swap.free (gauge)	The total amount of swap memory free. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.swap.in (gauge)	The amount of memory swapped in from disk. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.swap.out (gauge)	The amount of memory swapped out from disk. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.swap.total (gauge)	The total amount of swap memory available. (Enhanced) <i>Shown as kibibyte</i>		
aws.rds.tasks.blocked (gauge)	The number of tasks that are blocked. (Enhanced) <i>Shown as task</i>		
aws.rds.tasks.running (gauge)	The number of tasks that are running. (Enhanced) <i>Shown as task</i>		
aws.rds.tasks.sleeping (gauge)	The number of tasks that are sleeping. (Enhanced) <i>Shown as task</i>		
aws.rds.tasks.stopped (gauge)	The number of tasks that are stopped. (Enhanced) <i>Shown as task</i>		
aws.rds.tasks.total (gauge)	The total number of tasks. (Enhanced) <i>Shown as task</i>		
aws.rds.tasks.zombie (gauge)	The number of child tasks that are inactive with an active parent task. (Enhanced) <i>Shown as task</i>		
aws.rds.uptime (gauge)	RDS instance uptime. (Enhanced) <i>Shown as second</i>		

aws.rds.virtual_cpus (gauge)	The number of virtual CPUs for the DB instance. (Enhanced) <i>Shown as cpu</i>		
--	---	--	--

▼ AWS Route 53

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.route53.health_check_percentage_healthy (gauge)	The percentage of Amazon Route 53 health checkers that consider the selected endpoint to be healthy. <i>Shown as percent</i>	SATURATION	✓
aws.route53.health_check_status (gauge)	The status of the health check endpoint that CloudWatch is checking. 1 indicates healthy, and 0 indicates unhealthy.	ERROR	✓
aws.route53.connection_time (gauge)	The average time that it took Amazon Route 53 health checkers to establish a TCP connection with the endpoint. <i>Shown as millisecond</i>	SATURATION	✗
aws.route53.time_to_first_byte (gauge)	The average time that it took Amazon Route 53 health checkers to receive the first byte of the response to an HTTP or HTTPS request. <i>Shown as millisecond</i>	SATURATION	✗
aws.route53.sslhandshake_time (gauge)	The average time that it took Amazon Route 53 health checkers to complete the SSL handshake.	SATURATION	✗
aws.route53.child_health_check_healthy_count (gauge)	For a calculated health check, the number of health checks that are healthy among the health checks that Amazon Route 53 is monitoring.	ERROR	✗
aws.route53.dnsqueries (count)	The number of DNS queries that Route53 responds to for all records in a hosted zone. <i>Shown as query</i>	ERROR	✓
aws.route53resolver.inbound_query_volume (count)	The number of DNS queries forwarded from your network to your VPCs through the endpoint specified by EndpointId. <i>Shown as query</i>	TRAFFIC	✓
aws.route53resolver.outbound_query_volume (count)	The number of DNS queries forwarded from your VPCs to your network through the endpoint specified by EndpointId. <i>Shown as query</i>	TRAFFIC	✓
aws.route53resolver.outbound_query_aggregate_volume (count)	The number of DNS queries forwarded from your VPCs to your network through the endpoint that is specified by EndpointId. <i>Shown as query</i>	TRAFFIC	✗

▼ CloudFlare

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
cloudflare.requests.all (count)	Total request count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.cached (count)	Cached requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.uncached (count)	Uncached requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.ssl.encrypted (count)	SSL encrypted requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.ssl.unencrypted (count)	Unencrypted requests count <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.country (count)	Request count, tagged by IATA country code <i>Shown as request</i>	TRAFFIC	✗
cloudflare.requests.status (count)	Request count, tagged by HTTP response code <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.content_type (count)	Request count, tagged by Content-Type <i>Shown as request</i>	TRAFFIC	✓
cloudflare.requests.ip_class (count)	Request count, tagged by IP class <i>Shown as request</i>	TRAFFIC	✗

cloudflare.bandwidth.all (count)	Total bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.cached (count)	Cached bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.uncached (count)	Uncached bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.ssl.encrypted (count)	SSL encrypted bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.ssl.unencrypted (count)	Unencrypted bandwidth <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.country (count)	Bandwidth tagged by IATA country code <i>Shown as byte</i>	SATURATION	✗
cloudflare.bandwidth.content_type (count)	Bandwidth tagged by Content-Type <i>Shown as byte</i>	SATURATION	✗
cloudflare.threats.all (count)	Total threats <i>Shown as operation</i>	ERROR	✓
cloudflare.threats.type (count)	Threats tagged by type <i>Shown as operation</i>	ERROR	✓
cloudflare.threats.country (count)	Threats tagged by IATA country code <i>Shown as operation</i>	ERROR	✗
cloudflare.pageviews.all (count)	Total page views <i>Shown as page</i>	TRAFFIC	✓
cloudflare.pageviews.search_engine (count)	Page views tagged by search engine <i>Shown as page</i>	TRAFFIC	✗
cloudflare.uniques.all (count)	Unique visitors count <i>Shown as connection</i>	TRAFFIC	✗
cloudflare.dns.query.all (count)	DNS query count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.dns.query.uncached (count)	Uncached DNS query count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.dns.query.stale (count)	Stale DNS query count <i>Shown as request</i>	TRAFFIC	✗
cloudflare.dns.response_time.avg (gauge)	DNS query average response time <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.dns.response_time.median (gauge)	DNS query median response time <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.dns.response_time.90p (gauge)	DNS query response time to the 90th percentile <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.dns.response_time.99p (gauge)	DNS query response time to the 99th percentile <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.workers.requests.all (count)	The request count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	TRAFFIC	✗
cloudflare.workers.requests.errors (count)	The error count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	ERROR	✗
cloudflare.workers.requests.subrequests (count)	The subrequest count to the worker script (metrics may not show without enabled API Key permissions) <i>Shown as request</i>	TRAFFIC	✗
cloudflare.workers.response_time.75p (gauge)	The worker response time to the 75th percentile (metrics may not show without enabled API Key permissions) <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.workers.response_time.99p (gauge)	The worker response time to the 99th percentile (metrics may not show without enabled API Key permissions) <i>Shown as millisecond</i>	LATENCY	✗
cloudflare.load_balancer.pool.round_trip_time.average (gauge)	The average round trip time to reach the load balancer pool <i>Shown as millisecond</i>	LATENCY	✗

cloudflare.load_balancer.pool.health.status (count)	The load balancer pool health status <i>Shown as request</i>	ERROR	✓
---	---	-------	---

[VPC \(Virtual Private Cloud\)](#)

VPC Monitoring

Terraform Module repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-vpc-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.transitgateway.bytes_in (count)	The number of bytes received by the transit gateway. <i>Shown as byte</i>	TRAFFIC	✓
aws.transitgateway.bytes_out (count)	The number of bytes sent from the transit gateway. <i>Shown as byte</i>	TRAFFIC	✓
aws.transitgateway.packet_drop_count_blackhole (count)	The number of packets dropped because they matched a blackhole route. <i>Shown as packet</i>	SATURATION	✓
aws.transitgateway.packet_drop_count_no_route (count)	The number of packets dropped because they did not match a route. <i>Shown as packet</i>	SATURATION	✓
aws.transitgateway.packets_in (count)	The number of packets received by the transit gateway. <i>Shown as packet</i>	TRAFFIC	✗
aws.transitgateway.packets_out (count)	The number of packets sent by the transit gateway. <i>Shown as packet</i>	TRAFFIC	✗
aws.vpc.flowlogs.action (count)	ACCEPT or REJECT if the traffic was permitted or not by the security groups or network ACLs	TRAFFIC	✓
aws.vpc.flowlogs.bytes.per_request.max (gauge)	The maximum number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✗
aws.vpc.flowlogs.bytes.per_request.median (gauge)	The median number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✓
aws.vpc.flowlogs.bytes.per_request.min (gauge)	The minimum number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✗
aws.vpc.flowlogs.bytes.per_request.p90 (gauge)	The 90th percentile number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✗
aws.vpc.flowlogs.bytes.per_request.p95 (gauge)	The 95th percentile number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✓
aws.vpc.flowlogs.bytes.per_request.p99 (gauge)	The 99th percentile number of bytes transferred per request during the capture window <i>Shown as byte</i>	TRAFFIC	✗
aws.vpc.flowlogs.bytes.total (count)	The total number of bytes transferred during the capture window <i>Shown as byte</i>	TRAFFIC	✓
aws.vpc.flowlogs.duration.per_request.max (gauge)	The maximum duration per request during the capture window <i>Shown as second</i>	LATENCY	✗
aws.vpc.flowlogs.duration.per_request.median (gauge)	The median duration per request during the capture window <i>Shown as second</i>	LATENCY	✓
aws.vpc.flowlogs.duration.per_request.min (gauge)	The minimum duration per request during the capture window <i>Shown as second</i>	LATENCY	✗
aws.vpc.flowlogs.duration.per_request.p90 (gauge)	The 90th percentile duration per request during the capture window <i>Shown as second</i>	LATENCY	✗
aws.vpc.flowlogs.duration.per_request.p95 (gauge)	The 95th percentile duration per request during the capture window <i>Shown as second</i>	LATENCY	✗

aws.vpc.flowlogs.duration.per_request.p99 (gauge)	The 99th percentile duration per request during the capture window <i>Shown as second</i>	LATENCY	✗
aws.vpc.flowlogs.log_status (count)	The logging status of the flow log: OK NODATA or SKIPDATA	TRAFFIC	✗
aws.vpc.flowlogs.packets.per_request.max (gauge)	The maximum number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✗
aws.vpc.flowlogs.packets.per_request.median (gauge)	The median number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✗
aws.vpc.flowlogs.packets.per_request.min (gauge)	The minimum number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✓
aws.vpc.flowlogs.packets.per_request.p90 (gauge)	The 90th percentile number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✗
aws.vpc.flowlogs.packets.per_request.p95 (gauge)	The 95th percentile number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✓
aws.vpc.flowlogs.packets.per_request.p99 (gauge)	The 99th percentile number of packets transferred per request during the capture window <i>Shown as packet</i>	TRAFFIC	✗
aws.vpc.flowlogs.packets.total (count)	The total number of packets transferred during the capture window <i>Shown as packet</i>	TRAFFIC	✓
aws.vpc.subnet.total_ip_address_count (gauge)	The total number of IP addresses contained within the subnet	TRAFFIC	✗
aws.vpc.subnet.available_ip_address_count (gauge)	The number of available IP addresses in the subnet	TRAFFIC	✗

▼ Lambda

Lambda Monitoring

Terraform Module repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-lambda-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.lambda.duration (gauge)	Measures the average elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>	LATENCY	✓
aws.lambda.duration.maximum (gauge)	Measures the maximum elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.duration.minimum (gauge)	Measures the minimum elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.duration.sum (gauge)	Measures the total execution time of the lambda function executing. <i>Shown as millisecond</i>		
aws.lambda.duration.p50 (gauge)	Measures the p50 elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.duration.p80 (gauge)	Measures the p80 elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		

aws.lambda.duration.p95 (gauge)	Measures the p95 elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.duration.p99 (gauge)	Measures the p99 elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.duration.p99.9 (gauge)	Measures the p99.9 elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as millisecond</i>		
aws.lambda.timeout (gauge)	Measures the amount of allowed execution time for the function before the Lambda runtime stops it. <i>Shown as second</i>	LATENCY	✓
aws.lambda.memorysize (gauge)	Measures the amount of allocated memory available to the function during execution. <i>Shown as mebibyte</i>	SATURATION	✓
aws.lambda.errors (count)	Measures the number of invocations that failed due to errors in the function (response code 4XX). <i>Shown as error</i>	ERROR	✓
aws.lambda.invocations (count)	Measures the number of times a function is invoked in response to an event or invocation API call. <i>Shown as invocation</i>	TRAFFIC	✓
aws.lambda.throttles (count)	Measures the number of Lambda function invocation attempts that were throttled due to invocation rates exceeding the customer's concurrent limits (error code 429). Failed invocations may trigger a retry attempt that succeeds. <i>Shown as throttle</i>	ERROR	✓
aws.lambda.iterator_age (gauge)	Measures the age of the last record for each batch of records processed <i>Shown as millisecond</i>		
aws.lambda.iterator_age.minimum (gauge)	Measures the minimum age of the last record for each batch of records processed <i>Shown as millisecond</i>		
aws.lambda.iterator_age.maximum (gauge)	Measures the maximum age of the last record for each batch of records processed <i>Shown as millisecond</i>		
aws.lambda.iterator_age.sum (gauge)	Measures the sum of the ages of the last record for each batch of records processed <i>Shown as millisecond</i>		
aws.lambda.dead_letter_errors (count)	Measures the sum of times Lambda is unable to write the failed event payload to your configured Dead Letter Queues. <i>Shown as error</i>		
aws.lambda.concurrent_executions (gauge)	Measures the average of concurrent executions for a given function at a given point in time. <i>Shown as execution</i>	TRAFFIC	✓
aws.lambda.concurrent_executions.minimum (gauge)	Measures the minimum of concurrent executions for a given function at a given point in time. <i>Shown as execution</i>		
aws.lambda.concurrent_executions.maximum (gauge)	Measures the maximum of concurrent executions for a given function at a given point in time. <i>Shown as execution</i>		
aws.lambda.concurrent_executions.sum (gauge)	Measures the sum of concurrent executions for a given function at a given point in time. <i>Shown as execution</i>		
aws.lambda.concurrent_executions_global (gauge)	Measures the average of concurrent executions for all functions in an account at a given point in time. <i>Shown as execution</i>		
aws.lambda.concurrent_executions_global.minimum (gauge)	Measures the minimum of concurrent executions for all functions in an account at a given point in time. <i>Shown as execution</i>		

aws.lambda.concurrent_executions_global.maximum (gauge)	Measures the maximum of concurrent executions for all functions in an account at a given point in time. <i>Shown as execution</i>		
aws.lambda.concurrent_executions_global.sum (gauge)	Measures the sum of concurrent executions for all functions in an account at a given point in time. <i>Shown as execution</i>		
aws.lambda.unreserved_concurrent_executions (gauge)	Measures the sum of the concurrency of the functions that don't have a custom concurrency limit specified. <i>Shown as execution</i>		
aws.lambda.provisioned_concurrent_executions (gauge)	Measures the average number of events that are being processed on provisioned concurrency <i>Shown as execution</i>	TRAFFIC	✓
aws.lambda.provisioned_concurrent_executions.minimum (gauge)	Measures the minimum number of events that are being processed on provisioned concurrency <i>Shown as execution</i>		
aws.lambda.provisioned_concurrent_executions.maximum (gauge)	Measures the maximum number of events that are being processed on provisioned concurrency <i>Shown as execution</i>		
aws.lambda.provisioned_concurrency_invocations (count)	Measures the number of invocations that are run on provisioned concurrency <i>Shown as invocation</i>	TRAFFIC	✓
aws.lambda.provisioned_concurrency_spillover_invocations (count)	Measures the number of invocations that are run on non-provisioned concurrency when all provisioned concurrency is in use <i>Shown as invocation</i>		
aws.lambda.provisioned_concurrency_utilization (gauge)	Measures the average fraction of provisioned concurrency in use for a given function at a given point in time <i>Shown as percent</i>	SATURATION	✓
aws.lambda.provisioned_concurrency_utilization.minimum (gauge)	Measures the minimum fraction of provisioned concurrency in use for a given function at a given point in time <i>Shown as percent</i>		
aws.lambda.provisioned_concurrency_utilization.maximum (gauge)	Measures the maximum fraction of provisioned concurrency in use for a given function at a given point in time <i>Shown as percent</i>		
aws.lambda.signature_validation_errors (count)	Measures the number of times a function is successfully deployed but fails a signature check <i>Shown as error</i>		
aws.lambda.enhanced.invocations (count)	Measures the number of times a function is invoked in response to an event or invocation API call. <i>Shown as invocation</i>		
aws.lambda.enhanced.errors (count)	Measures the number of invocations that failed due to errors in the function (response code 4XX). <i>Shown as error</i>		
aws.lambda.enhanced.max_memory_used (gauge)	Measures the maximum amount of memory (mb) used by the function. <i>Shown as mebibyte</i>		
aws.lambda.enhanced.duration (gauge)	Measures the elapsed seconds from when the function code starts executing as a result of an invocation to when it stops executing. <i>Shown as second</i>		
aws.lambda.enhanced.billed_duration (gauge)	Measures the billed amount of time the function ran for (100ms increments). <i>Shown as second</i>		
aws.lambda.enhanced.init_duration (gauge)	Measures the initialization time (second) of a function during a cold start. <i>Shown as second</i>		
aws.lambda.enhanced.estimated_cost (gauge)	Measures the total estimated cost of the function invocation (US dollars). <i>Shown as dollar</i>		
aws.lambda.enhanced.timeouts (count)	Measures the number of times a function times out. <i>Shown as timeout</i>		
aws.lambda.enhanced.out_of_memory (count)	Measures the number of times a function runs out of memory. <i>Shown as error</i>		

▼ Direct Connect
DirectConnect Monitoring

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.dx.connection_state (gauge)	The state of the connection. 0 indicates DOWN and 1 indicates UP.	ERROR	✓
aws.dx.connection_bps_egress (rate)	The bit rate for outbound data from the AWS side of the connection. <i>Shown as bit</i>	TRAFFIC	✓
aws.dx.connection_bps_ingress (rate)	The bit rate for inbound data to the AWS side of the connection. <i>Shown as bit</i>	TRAFFIC	✓
aws.dx.connection_pps_egress (rate)	The packet rate for outbound data from the AWS side of the connection. <i>Shown as packet</i>	TRAFFIC	✓
aws.dx.connection_pps_ingress (rate)	The packet rate for inbound data to the AWS side of the connection. <i>Shown as packet</i>	TRAFFIC	✓
aws.dx.connection_crcerror_count (count)	The number of times cyclic redundancy check (CRC) errors are observed for the data received at the connection. <i>Shown as error</i>	ERROR	✓
aws.dx.connection_light_level_tx (gauge)	Indicates the health of the fiber connection for egress (outbound) traffic from the AWS side of the connection.	TRAFFIC	✗
aws.dx.connection_light_level_rx (gauge)	Indicates the health of the fiber connection for ingress (inbound) traffic to the AWS side of the connection.	TRAFFIC	✗

▼ Route Table Modification

- [CreateRoute](#)
- [CreateRouteTable](#)
- [ReplaceRoute](#)
- [ReplaceRouteTableAssociation](#)
- [DeleteRouteTable](#)
- [DeleteRoute](#)
- [DisassociateRouteTable](#)

▼ AWS Gitlab Runner VM
AWS Gitlab Runners monitoring

Terraform Module Repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-gitlab-runners-monitors>

METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.ec2.cpuutilization (gauge)	Average percentage of allocated EC2 compute units that are currently in use on the instance. <i>Shown as percent</i>	SATURATION	✓
aws.ec2.disk_read_bytes (gauge)	Bytes read from all ephemeral disks available to the instance. <i>Shown as byte</i>	SATURATION	✓
aws.ec2.disk_read_ops (gauge)	Completed read operations from all ephemeral disks available to the instance. <i>Shown as operation</i>	SATURATION	✓
aws.ec2.disk_write_bytes (gauge)	Bytes written to all ephemeral disks available to the instance. <i>Shown as byte</i>	SATURATION	✓
aws.ec2.disk_write_ops (gauge)	Completed write operations to all ephemeral disks available to the instance. <i>Shown as operation</i>	SATURATION	✓
aws.ec2.host_ok (gauge)	1 if the instance's system status is ok.	ERROR	✓
aws.ec2.network_in (gauge)	Average number of bytes received on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✓

aws.ec2.network_out (gauge)	Average number of bytes sent out on all network interfaces by the instance. <i>Shown as byte</i>	SATURATION	✓
aws.ec2.status_check_failed_instance (gauge)	0 if the instance has passed the EC2 instance status check.	ERROR	✓
sys.mem.used (count)	System Memory Used (<i>Shown as bytes</i>)	SATURATION	✓
sys.mem.free (count)	System Memory Free (<i>Shown as bytes</i>)	SATURATION	✓
sys.disk.used (count)	System Disk Used (<i>Shown as bytes</i>)	SATURATION	✓
sys.disk.free (count)	System Disk Free (<i>Shown as bytes</i>)	SATURATION	✓

▼ Kinesis

Kinesis Monitoring

Terraform Module Repo:

<https://gitlab.com/bhp-cloudfactory/tooling-blueprints/sob/common-components/terraform-datadog-aws-kinesis-monitors>



METRIC NAME	DESCRIPTION	SIGNAL TYPE	COVER
aws.kinesis.get_records_bytes.maximum (gauge)	Maximum number of bytes per GetRecords operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.get_records_bytes.minimum (gauge)	Minimum number of bytes per GetRecords operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.get_records_bytes.sum (count)	Total number of bytes returned over all GetRecords operations <i>Shown as byte</i>	SATURATION	✓
aws.kinesis.get_records_bytes (gauge)	Average number of bytes per GetRecords operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.get_records_iterator_age_milliseconds.maximum (gauge)	Maximum difference between the current time and when the last record of a GetRecords call was written to the stream. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.get_records_iterator_age_milliseconds (gauge)	Difference between the current time and when the last record of a GetRecords call was written to the stream. <i>Shown as millisecond</i>	LATENCY	✓
aws.kinesis.get_records_iterator_age.maximum (gauge)	Maximum difference between the current time and when the last record of a GetRecords call was written to the stream. <i>Shown as second</i>	LATENCY	✗
aws.kinesis.get_records_iterator_age (gauge)	Average difference between the current time and when the last record of a GetRecords call was written to the stream. <i>Shown as second</i>	LATENCY	✗
aws.kinesis.get_records_latency.maximum (gauge)	Maximum time taken per GetRecords operation. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.get_records_latency (gauge)	Average time taken per GetRecords operation. <i>Shown as millisecond</i>	LATENCY	✓
aws.kinesis.get_records_records.maximum (gauge)	Maximum number of records per GetRecords operation <i>Shown as record</i>	SATURATION	✗
aws.kinesis.get_records_records.minimum (gauge)	Minimum number of records per GetRecords operation <i>Shown as record</i>	SATURATION	✗

aws.kinesis.get_records_records.sum (count)	Total number of records returned over all GetRecords operations <i>Shown as record</i>	SATURATION	✗
aws.kinesis.get_records_records (gauge)	Average number of records per GetRecords operation <i>Shown as record</i>	SATURATION	✗
aws.kinesis.get_records_success.average (count)	Average number of successful GetRecords operations per stream. <i>Shown as event</i>	SATURATION	✗
aws.kinesis.get_records_success (count)	Sum of successful GetRecords operations per stream. <i>Shown as event</i>	SATURATION	✗
aws.kinesis.incoming_bytes.sum (count)	Total number of bytes successfully put to the Amazon Kinesis stream. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.incoming_bytes (gauge)	Average number of bytes successfully put to the Amazon Kinesis stream per operation. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.incoming_records.average (gauge)	Average number of records successfully put to the Amazon Kinesis stream per operation. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.incoming_records (count)	Total number of records successfully put to the Amazon Kinesis stream. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.iterator_age_milliseconds (gauge)	The age of the last record in all GetRecords calls made against a shard, measured over the specified time period. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.outgoing_bytes.sum (count)	Total number of bytes retrieved from the Amazon Kinesis stream <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.outgoing_bytes (gauge)	Average number of bytes retrieved in a GetRecords operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.outgoing_records.average (gauge)	The average number of records retrieved from the shard, measured over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.outgoing_records (count)	The number of records retrieved from the shard, measured over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.outgoing_records (count)	The number of records retrieved from the shard, measured over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_record_bytes.maximum (gauge)	Maximum number of bytes per PutRecord operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_record_bytes.minimum (gauge)	Minimum number of bytes per PutRecord operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_record_bytes.sum (count)	Total number of bytes for all PutRecord operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_record_bytes (gauge)	Average number of bytes per PutRecord operation <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_record_latency.maximum (gauge)	Maximum time taken per PutRecord operation. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.put_record_latency (gauge)	Average time taken per PutRecord operation. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.put_record_success.average (count)	Average number of successful PutRecord operations per Amazon Kinesis stream. <i>Shown as event</i>	SATURATION	✗
aws.kinesis.put_record_success (count)	Sum of successful PutRecord operations per Amazon Kinesis stream. <i>Shown as event</i>	SATURATION	✗

aws.kinesis.put_records_bytes.maximum (gauge)	Maximum number of bytes per PutRecords operation. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_records_bytes.minimum (gauge)	Minimum number of bytes per PutRecords operation. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_records_bytes.sum (count)	Total number of bytes for all PutRecords operation. <i>Shown as byte</i>	SATURATION	✓
aws.kinesis.put_records_bytes (gauge)	Average number of bytes per PutRecords operation. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.put_records_failed_records.average (gauge)	The number of records rejected due to internal failures in a PutRecords operation per Kinesis data stream, averaged over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_failed_records.maximum (gauge)	The number of records rejected due to internal failures in a PutRecords operation per Kinesis data stream, maximum over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_failed_records.minimum (gauge)	The number of records rejected due to internal failures in a PutRecords operation per Kinesis data stream, minimum over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_failed_records (gauge)	The number of records rejected due to internal failures in a PutRecords operation per Kinesis data stream, summed over the specified time period. <i>Shown as record</i>	ERROR	✓
aws.kinesis.put_records_latency.maximum (gauge)	Maximum time taken for all PutRecords operation. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis.put_records_latency (gauge)	Average time taken per PutRecords operation. <i>Shown as millisecond</i>	LATENCY	✓
aws.kinesis.put_records_records.maximum (gauge)	Maximum number of records for PutRecords operations <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_records.minimum (gauge)	Minimum number of records for PutRecords operations <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_records.sum (count)	Total number of records for PutRecords operations <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_records (gauge)	Average number of records per PutRecords operation <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_success.average (count)	Average number of successful PutRecords operations per Amazon Kinesis stream. <i>Shown as event</i>	SATURATION	✗
aws.kinesis.put_records_success (count)	Sum of successful PutRecords operations per Amazon Kinesis stream. <i>Shown as event</i>	SATURATION	✗
aws.kinesis.put_records_successful_records.average (gauge)	The number of successful records in a PutRecords operation per Kinesis data stream, averaged over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_successful_records.maximum (gauge)	The number of successful records in a PutRecords operation per Kinesis data stream, maximum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_successful_records.minimum (gauge)	The number of successful records in a PutRecords operation per Kinesis data stream, minimum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_successful_records (gauge)	The number of successful records in a PutRecords operation per Kinesis data stream, summed over the specified time period. <i>Shown as record</i>	SATURATION	✗

aws.kinesis.put_records_throttled_records.average (gauge)	The number of records rejected due to throttling in a PutRecords operation per Kinesis data stream, averaged over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_throttled_records.maximum (gauge)	The number of records rejected due to throttling in a PutRecords operation per Kinesis data stream, maximum over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_throttled_records.minimum (gauge)	The number of records rejected due to throttling in a PutRecords operation per Kinesis data stream, minimum over the specified time period. <i>Shown as record</i>	ERROR	✗
aws.kinesis.put_records_throttled_records (gauge)	The number of records rejected due to throttling in a PutRecords operation per Kinesis data stream, summed over the specified time period. <i>Shown as record</i>	ERROR	✓
aws.kinesis.put_records_total_records.average (gauge)	The total number of records sent in a PutRecords operation per Kinesis data stream, averaged over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_total_records.maximum (gauge)	The total number of records sent in a PutRecords operation per Kinesis data stream, maximum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_total_records.minimum (gauge)	The total number of records sent in a PutRecords operation per Kinesis data stream, minimum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.put_records_total_records (gauge)	The total number of records sent in a PutRecords operation per Kinesis data stream, summed over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis.read_provisioned_throughput_exceeded.average (count)	Average of GetRecords calls throttled for the stream <i>Shown as record</i>	ERROR	✗
aws.kinesis.read_provisioned_throughput_exceeded.maximum (count)	Maximum number of GetRecords calls throttled for the stream <i>Shown as record</i>	ERROR	✗
aws.kinesis.read_provisioned_throughput_exceeded.minimum (count)	Minimum number of GetRecords calls throttled for the stream <i>Shown as record</i>	ERROR	✗
aws.kinesis.read_provisioned_throughput_exceeded (count)	Number of GetRecords calls throttled for the stream <i>Shown as record</i>	ERROR	✓
aws.kinesis.subscribe_to_shard_event_bytes.maximum (gauge)	The number of bytes received from the shard, maximum over the specified time period. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.subscribe_to_shard_event_bytes.minimum (gauge)	The number of bytes received from the shard, minimum over the specified time period. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.subscribe_to_shard_event_bytes.sum (gauge)	The number of bytes received from the shard, summed over the specified time period. <i>Shown as byte</i>	SATURATION	✓
aws.kinesis.subscribe_to_shard_event_bytes (gauge)	The number of bytes received from the shard, averaged over the specified time period. <i>Shown as byte</i>	SATURATION	✗
aws.kinesis.subscribe_to_shard_event_millis_behind_latest.maximum (gauge)	The difference between the current time and when the last record of the SubscribeToShard event was written to the stream. <i>Shown as millisecond</i>	LATENCY	✗

aws.kinesis. subscribe_to_shard_event_millis_behind _latest (gauge)	The difference between the current time and when the last record of the SubscribeToShard event was written to the stream. <i>Shown as millisecond</i>	LATENCY	✗
aws.kinesis. subscribe_to_shard_event_records. maximum (count)	The number of records received from the shard, maximum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_event_records. minimum (count)	The number of records received from the shard, minimum over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_event_records.sum (count)	The number of records received from the shard, summed over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_event_records (count)	The number of records received from the shard, averaged over the specified time period. <i>Shown as record</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_event_success. average (count)	This metric is emitted every time an event is published successfully. It is only emitted when there's an active subscription. <i>Shown as event</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_event_success (count)	This metric is emitted every time an event is published successfully. It is only emitted when there's an active subscription. <i>Shown as event</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_rate_exceeded. average (count)	This metric is emitted when a new subscription attempt fails because there already is an active subscription by the same consumer or if you exceed the number of calls per second allowed for this operation. <i>Shown as record</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_rate_exceeded. maximum (count)	This metric is emitted when a new subscription attempt fails because there already is an active subscription by the same consumer or if you exceed the number of calls per second allowed for this operation. <i>Shown as record</i>	ERROR	✗
aws.kinesis. subscribe_to_shard_rate_exceeded. minimum (count)	This metric is emitted when a new subscription attempt fails because there already is an active subscription by the same consumer or if you exceed the number of calls per second allowed for this operation. <i>Shown as record</i>	ERROR	✗
aws.kinesis. subscribe_to_shard_rate_exceeded (count)	This metric is emitted when a new subscription attempt fails because there already is an active subscription by the same consumer or if you exceed the number of calls per second allowed for this operation. <i>Shown as record</i>	ERROR	✓
aws.kinesis. subscribe_to_shard_success.average (gauge)	This metric records whether the SubscribeToShard subscription was successfully established. The subscription only lives for at most 5 minutes. Therefore, this metric gets emitted at least once every 5 minutes. <i>Shown as event</i>	SATURATION	✗
aws.kinesis. subscribe_to_shard_success (gauge)	This metric records whether the SubscribeToShard subscription was successfully established. The subscription only lives for at most 5 minutes. Therefore, this metric gets emitted at least once every 5 minutes. <i>Shown as event</i>	SATURATION	✗
aws.kinesis. write_provisioned_throughput_exceeded. average (count)	Average of records rejected due to throttling for the stream <i>Shown as record</i>	ERROR	✗
aws.kinesis. write_provisioned_throughput_exceeded. maximum (count)	Maximum number of records rejected due to throttling for the stream <i>Shown as record</i>	ERROR	✗

aws.kinesis. write_provisioned_throughput_exceeded .minimum (count)	Minimum number of records rejected due to throttling for the stream <i>Shown as record</i>	ERROR	
aws.kinesis. write_provisioned_throughput_exceeded (count)	Number of records rejected due to throttling for the stream <i>Shown as record</i>	ERROR	

Make It Observable

To make the services observable we need to integrate DataDog with each service. DataDog has many integration pieces with Azure nad AWS services which makes those services observable in order to be monitored.

Below we introduce DataDog as our monitoring tool and how it can be integrated with all the services we mentioned above for Azure and AWS.

DataDog

Datadog is an essential cloud monitoring and operational analytics tool which enables the monitoring of servers, virtual machines, containers, databases, third-party tools, and application services. IT and DevOps teams can easily leverage Datadog to monitor infrastructure and cloud services

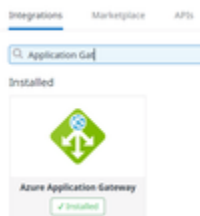
DataDog Integration

Azure

Azure Virtual Machine
 Azure Application Gateway
 Azure Express Route
 Azure Storage Account
 Azure Virtual Networks
 Azure Firewall
 Azure VM Scaleset

Installation:

- Login to DataDog console (<https://app.datadoghq.com/>)
- Go to "Integrations" tab and Click on Integrations
- Search for above Azure service one at a time (Ignore Azure Firewall) to list the available extensions
- Place the cursor on "+Available" button, it will display the Installation Popup and then Click on "+Install"
- If the extension shows as "Installed", you can ignore the above step and proceed for the configuration



Configuration:

No additional configuration is required for the above 5 Azure services if you have already integrated Azure with DataDog. if not please follow steps mentioned in this [link](#) to integrate Azure and DataDog manually from Azure portal

User Permission:

The App or user which is configured as part of "Azure - DataDog Integration" should have at least "Monitoring Reader" Role on the specific subscription.

Add role assignment



Role ⓘ

Monitoring Reader ⓘ



Assign access to ⓘ

User, group, or service principal



AWS

DataDog Dashboards

A dashboard is Datadog's tool for visually tracking, analysing, and displaying key performance metrics, which enable you to monitor the health of your infrastructure.

[Dashboards](#) are on a next-gen grid based layout which can include a variety of objects such as images, graphs, and logs. They are commonly used as status boards or storytelling view, which update in real-time and can represent fixed points in the past. They also work well for debugging.

Create Dashboard

Method 1 (Manually in DataDog Console):

- Login to DataDog console
- Click on Dashboards from the menu and choose New Dashboard
- Click on New Dashboard Button
- Click on Add Widgets
- Choose widget type based on the metric type
- Based on the chosen widget different form fields need to be filled
- Once those fields are populated click on Preview to see if it works and then choose Done

Method 2 (Terraform Code):

To create a dashboard as code, below is an example that includes most of the dashboard widget types:

✓ [Terraform Dashboard Example](#)

```
# Example Ordered Layout
resource "datadog_dashboard" "ordered_dashboard" {
  title          = "Ordered Layout Dashboard"
  description    = "Created using the Datadog provider in Terraform"
  layout_type    = "ordered"
  is_read_only   = true

  widget {
    alert_graph_definition {
      alert_id = "895605"
      viz_type = "timeseries"
      title    = "Widget Title"
      live_span = "1h"
    }
  }
}
```

```

widget {
  alert_value_definition {
    alert_id    = "895605"
    precision   = 3
    unit        = "b"
    text_align  = "center"
    title       = "Widget Title"
  }
}

widget {
  alert_value_definition {
    alert_id    = "895605"
    precision   = 3
    unit        = "b"
    text_align  = "center"
    title       = "Widget Title"
  }
}

widget {
  change_definition {
    request {
      q          = "avg:system.load.1{env:staging} by {account}"
      change_type = "absolute"
      compare_to  = "week_before"
      increase_good = true
      order_by    = "name"
      order_dir   = "desc"
      show_present = true
    }
    title       = "Widget Title"
    live_span   = "1h"
  }
}

widget {
  distribution_definition {
    request {
      q = "avg:system.load.1{env:staging} by {account}"
      style {
        palette = "warm"
      }
    }
    title       = "Widget Title"
    live_span   = "1h"
  }
}

widget {

```

```

    check_status_definition {
      check      = "aws.ecs.agent_connected"
      grouping   = "cluster"
      group_by   = ["account", "cluster"]
      tags       = ["account:demo", "cluster:awseb-ruthebdog-env-8-
dn3m6u3gvk"]
      title      = "Widget Title"
      live_span  = "1h"
    }
  }

  widget {
    heatmap_definition {
      request {
        q = "avg:system.load.1{env:staging} by {account}"
        style {
          palette = "warm"
        }
      }
      yaxis {
        min      = 1
        max      = 2
        include_zero = true
        scale     = "sqrt"
      }
      title      = "Widget Title"
      live_span  = "1h"
    }
  }

  widget {
    hostmap_definition {
      request {
        fill {
          q = "avg:system.load.1{*} by {host}"
        }
        size {
          q = "avg:memcache.uptime{*} by {host}"
        }
      }
      node_type      = "container"
      group           = ["host", "region"]
      no_group_hosts  = true
      no_metric_hosts = true
      scope           = ["region:us-east-1", "aws_account:
727006795293"]
      style {
        palette      = "yellow_to_green"
        palette_flip  = true
        fill_min      = "10"

```



```

        fill_max      = "20"
    }
    title = "Widget Title"
}
}

widget {
    note_definition {
        content          = "note text"
        background_color = "pink"
        font_size        = "14"
        text_align       = "center"
        show_tick        = true
        tick_edge        = "left"
        tick_pos         = "50%"
    }
}

widget {
    query_value_definition {
        request {
            q          = "avg:system.load.1{env:staging} by {account}"
            aggregator = "sum"
            conditional_formats {
                comparator = "<"
                value      = "2"
                palette    = "white_on_green"
            }
            conditional_formats {
                comparator = ">"
                value      = "2.2"
                palette    = "white_on_red"
            }
        }
        autoscale      = true
        custom_unit     = "xx"
        precision       = "4"
        text_align      = "right"
        title           = "Widget Title"
        live_span       = "1h"
    }
}

widget {
    query_table_definition {
        request {
            q          = "avg:system.load.1{env:staging} by {account}"
            aggregator = "sum"
            limit       = "10"
            conditional_formats {

```

```

        comparator = "<"
        value       = "2"
        palette     = "white_on_green"
    }
    conditional_formats {
        comparator = ">"
        value       = "2.2"
        palette     = "white_on_red"
    }
}
title       = "Widget Title"
live_span = "1h"
}

}

widget {
    scatterplot_definition {
        request {
            x {
                q          = "avg:system.cpu.user{*} by {service, account}"
                aggregator = "max"
            }
            y {
                q          = "avg:system.mem.used{*} by {service, account}"
                aggregator = "min"
            }
        }
        color_by_groups = ["account", "apm-role-group"]
        xaxis {
            include_zero = true
            label         = "x"
            min           = "1"
            max           = "2000"
            scale         = "pow"
        }
        yaxis {
            include_zero = false
            label         = "y"
            min           = "5"
            max           = "2222"
            scale         = "log"
        }
        title       = "Widget Title"
        live_span = "1h"
    }
}

}

widget {
    servicemap_definition {
        service       = "master-db"
    }
}

```

```

        filters      = ["env:prod", "datacenter:us1.prod.dog"]
        title        = "env: prod, datacenter:us1.prod.dog, service:
master-db"
        title_size   = "16"
        title_align  = "left"
    }
}

widget {
    timeseries_definition {
        request {
            q          = "avg:system.cpu.user{app:general} by {env}"
            display_type = "line"
            style {
                palette      = "warm"
                line_type    = "dashed"
                line_width   = "thin"
            }
            metadata {
                expression = "avg:system.cpu.user{app:general} by {env}"
                alias_name = "Alpha"
            }
        }
    }
    request {
        log_query {
            index = "mcnulty"
            compute_query {
                aggregation = "avg"
                facet       = "@duration"
                interval    = 5000
            }
            search_query = "status:info"
            group_by {
                facet = "host"
                limit = 10
                sort_query {
                    aggregation = "avg"
                    order       = "desc"
                    facet       = "@duration"
                }
            }
        }
        display_type = "area"
    }
    request {
        apm_query {
            index = "apm-search"
            compute_query {
                aggregation = "avg"
                facet       = "@duration"
            }
        }
    }
}

```

```

        interval      = 5000
    }
    search_query = "type:web"
    group_by {
        facet = "resource_name"
        limit = 50
        sort_query {
            aggregation = "avg"
            order       = "desc"
            facet       = "@string_query.interval"
        }
    }
}
display_type = "bars"
}
request {
    process_query {
        metric      = "process.stat.cpu.total_pct"
        search_by   = "error"
        filter_by   = ["active"]
        limit       = 50
    }
    display_type = "area"
}
marker {
    display_type = "error dashed"
    label       = " z=6 "
    value       = "y = 4"
}
marker {
    display_type = "ok solid"
    value       = "10 < y < 999"
    label       = " x=8 "
}
title      = "Widget Title"
show_legend = true
legend_size = "2"
live_span  = "1h"
event {
    q = "sources:test tags:1"
}
event {
    q = "sources:test tags:2"
}
yaxis {
    scale          = "log"
    include_zero  = false
    max            = 100
}
}

```

```

}

widget {
  toplist_definition {
    request {
      q = "avg:system.cpu.user{app:general} by {env}"
      conditional_formats {
        comparator = "<"
        value      = "2"
        palette     = "white_on_green"
      }
      conditional_formats {
        comparator = ">"
        value      = "2.2"
        palette     = "white_on_red"
      }
    }
    title = "Widget Title"
  }
}

widget {
  group_definition {
    layout_type = "ordered"
    title       = "Group Widget"

    widget {
      note_definition {
        content          = "cluster note widget"
        background_color = "pink"
        font_size        = "14"
        text_align       = "center"
        show_tick        = true
        tick_edge        = "left"
        tick_pos         = "50%"
      }
    }

    widget {
      alert_graph_definition {
        alert_id = "123"
        viz_type = "toplist"
        title    = "Alert Graph"
        live_span = "1h"
      }
    }
  }
}

widget {

```

```

    service_level_objective_definition {
      title           = "Widget Title"
      view_type       = "detail"
      slo_id          = "56789"
      show_error_budget = true
      view_mode       = "overall"
      time_windows    = ["7d", "previous_week"]
    }
  }

  template_variable {
    name      = "var_1"
    prefix    = "host"
    default   = "aws"
  }
  template_variable {
    name      = "var_2"
    prefix    = "service_name"
    default   = "autoscaling"
  }

  template_variable_preset {
    name = "preset_1"
    template_variable {
      name = "var_1"
      value = "host.dc"
    }
    template_variable {
      name = "var_2"
      value = "my_service"
    }
  }
}

# Example Free Layout
resource "datadog_dashboard" "free_dashboard" {
  title           = "Free Layout Dashboard"
  description      = "Created using the Datadog provider in Terraform"
  layout_type     = "free"
  is_read_only    = false

  widget {
    event_stream_definition {
      query          = "*"
      event_size     = "1"
      title          = "Widget Title"
      title_size     = 16
      title_align    = "left"
      live_span      = "1h"
    }
  }
}

```

```

    widget_layout {
        height = 43
        width  = 32
        x      = 0
        y      = 0
    }
}

widget {
    event_timeline_definition {
        query          = "*"
        title           = "Widget Title"
        title_size     = 16
        title_align     = "left"
        live_span      = "1h"
    }
    widget_layout {
        height = 9
        width  = 66
        x      = 33
        y      = 60
    }
}

widget {
    free_text_definition {
        text          = "free text content"
        color         = "#d00"
        font_size     = "36"
        text_align    = "left"
    }
    widget_layout {
        height = 20
        width  = 34
        x      = 33
        y      = 0
    }
}

widget {
    iframe_definition {
        url = "http://google.com"
    }
    widget_layout {
        height = 46
        width  = 39
        x      = 101
        y      = 0
    }
}

```



```

widget {
  image_definition {
    url      = "https://images.pexels.com/photos/67636/rose-blue-
flower-rose-blooms-67636.jpeg?auto=compress&cs=tinysrgb&h=350"
    sizing   = "fit"
    margin   = "small"
  }
  widget_layout {
    height = 20
    width  = 30
    x      = 69
    y      = 0
  }
}

```

```

widget {
  log_stream_definition {
    indexes      = ["main"]
    query        = "error"
    columns      = ["core_host", "core_service",
"tag_source"]
    show_date_column = true
    show_message_column = true
    message_display  = "expanded-md"
    sort {
      column = "time"
      order  = "desc"
    }
  }
  widget_layout {
    height = 36
    width  = 32
    x      = 0
    y      = 45
  }
}

```

```

widget {
  manage_status_definition {
    color_preference      = "text"
    display_format        = "countsAndList"
    hide_zero_counts      = true
    query                 = "type:metric"
    show_last_triggered    = false
    sort                  = "status,asc"
    summary_type          = "monitors"
    title                 = "Widget Title"
    title_size            = 16
    title_align           = "left"
  }
}

```

```

    }
    widget_layout {
        height = 40
        width  = 30
        x      = 101
        y      = 48
    }
}

widget {
    trace_service_definition {
        display_format      = "three_column"
        env                 = "datad0g.com"
        service             = "alerting-cassandra"
        show_breakdown      = true
        show_distribution    = true
        show_errors         = true
        show_hits           = true
        show_latency        = false
        show_resource_list  = false
        size_format         = "large"
        span_name           = "cassandra.query"
        title               = "alerting-cassandra #env:datad0g.com"
        title_align         = "center"
        title_size          = "13"
        live_span           = "1h"
    }
    widget_layout {
        height = 38
        width  = 66
        x      = 33
        y      = 21
    }
}

widget {
    timeseries_definition {
        request {
            formula {
                formula_expression = "my_query_1 + my_query_2"
                alias              = "my ff query"
            }
            formula {
                formula_expression = "my_query_1 * my_query_2"
                limit {
                    count = 5
                    order  = "desc"
                }
                alias = "my second ff query"
            }
        }
    }
}

```

```

    query {
      metric_query {
        data_source = "metrics"
        query       = "avg:system.cpu.user{app:general} by {env}"
        name        = "my_query_1"
        aggregator   = "sum"
      }
    }
    query {
      metric_query {
        query       = "avg:system.cpu.user{app:general} by {env}"
        name        = "my_query_2"
        aggregator   = "sum"
      }
    }
  }
}

widget_layout {
  height = 16
  width  = 25
  x      = 58
  y      = 83
}

}

widget {
  timeseries_definition {
    request {
      query {
        event_query {
          name      = "my-query"
          data_source = "logs"
          indexes   = ["days-3"]
          compute {
            aggregation = "count"
          }
          group_by {
            facet = "host"
            sort {
              metric      = "@lambda.max_memory_used"
              aggregation = "avg"
            }
            limit = 10
          }
        }
      }
    }
  }
}

}

}

}

widget_layout {
  height = 16
  width  = 28

```

```

        x      = 29
        y      = 83
    }
}
widget {
    timeseries_definition {
        request {
            query {
                process_query {
                    data_source      = "process"
                    text_filter       = "abc"
                    metric            = "process.stat.cpu.total_pct"
                    limit            = 10
                    tag_filters       = ["some_filter"]
                    name              = "my_process_query"
                    sort              = "asc"
                    is_normalized_cpu = true
                    aggregator        = "sum"
                }
            }
        }
    }
    widget_layout {
        height = 16
        width  = 28
        x      = 0
        y      = 83
    }
}

template_variable {
    name      = "var_1"
    prefix    = "host"
    default   = "aws"
}

template_variable {
    name      = "var_2"
    prefix    = "service_name"
    default   = "autoscaling"
}

template_variable_preset {
    name = "preset_1"
    template_variable {
        name = "var_1"
        value = "host.dc"
    }
    template_variable {
        name = "var_2"
        value = "my_service"
    }
}

```



DataDog Alerts

Monitoring all of your infrastructure in one place wouldn't be complete without the ability to know when critical changes are occurring. Datadog gives you the ability to create monitors that actively check metrics, integration availability, network endpoints, and more.

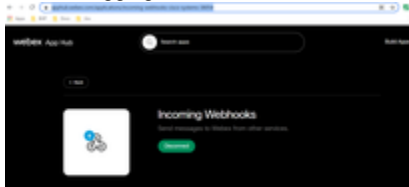
Currently we use Webex to capture alert notifications, our next step is to integrate DataDog with SNOW Incident Management module to trigger a SNOW incident and notify on-call groups on defined schedules

DataDog Integration with WebEx Teams

DataDog Integration with Webex teams will help to trigger alert notifications over chat. To execute the below steps, you need admin access on DataDog. this requires webhook creation from both WebEx and DataDog sides. below are the steps.

WebHook creation in WebEx

- To trigger alert notifications from DataDog to WebEx teams a webhook is required. if you have an existing webhook, please proceed for the DataDog side configuration. if not, below are the steps to create Webhook in webex.
- Create a space in WebEx teams with the list of people to whom you want to send the chat notifications.
- Open this [link](#) and Login(Login button on the top right side) in your BHP credentials.
- After logging in, Click on connect button under "Incoming Webhooks" and wait for few seconds till the button name turns as "Disconnect"



- Scroll down to the bottom, update the field "Webhook Name" with any value and select the space which you have created in the first step in "Select a space" dropdown.

Webhook name:

Select a space:

- Click "Add" and copy the Webhook URL which got created with the name mentioned in above step.

Incoming Webhooks

Test-WebHook Room: Webex Webhook Test

Webhook URL

<https://webexapi.com/v1/webhooks/incoming/Y2izY>

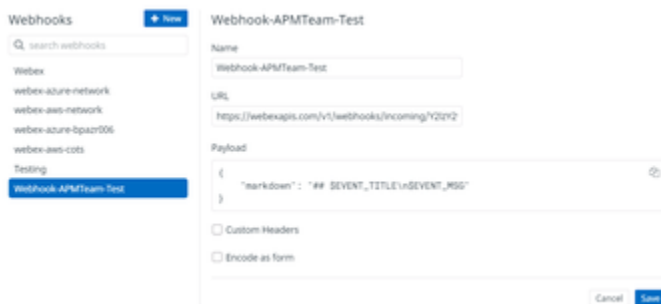
WebHook creation in DataDog

- Login to DataDog Console "<http://app.datadoghq.com/>"
- Goto Integrations tab and click on "Integrations"
- Search for WebHook extension and click on "configure" (Install the WebHook extension if it is not installed already)



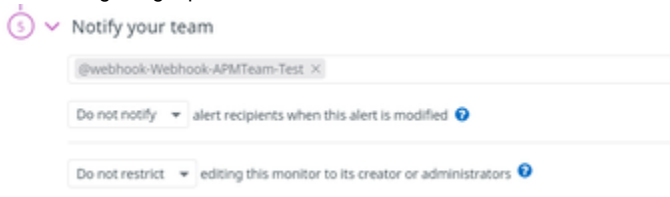
- In WebHook popup window, Go to "Configuration" tab and scroll down to the bottom and Click on "New" button next to "WebHook"
- Enter a name for Webhook and for URL field, enter the webhook url which you have copied in the last step of ["WebHook creation in WebEx"](#)
- mention the payload as below

```
{
  "markdown": "## $EVENT_TITLE\n$EVENT_MSG"
}
```
- Click "Save" button on the bottom.

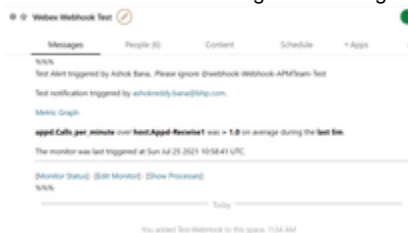


Enabling WebHook notifications

- Create a new monitor or open your existing monitor in DataDog
- Scroll down to the 5th step "Notify your team"
- Search for the webhook name which you have mentioned in 5th step of ["WebHook Creation in DataDog"](#) (by default all webhooks created in DataDog will get prefix "webhook-").



- Test the webhook configuration using "Test notifications" button.



- Click on "save" to complete the configuration.