Metrics and Alerts

3/23/2026

Contents

Learn which metrics you should monitor for and when to sound the alarm.

Metrics

Designer supports the Prometheus monitoring system. Desginer and DAS generate application related metrics at the `/metric` API in the standard Prometheus client format.

Designer metrics

[+] Some of the metrics exposed by Designer are as follows:

```
# HELP process_cpu_user_seconds_total Total user CPU time spent in seconds.
# TYPE process_cpu_user_seconds_total counter
process_cpu_user_seconds_total 1968.5202939999933 1622782805365

# HELP process_cpu_system_seconds_total Total system CPU time spent in seconds.
# TYPE process_cpu_system_seconds_total counter
process_cpu_system_seconds_total 264.1404249999979 1622782805365

# HELP process_cpu_seconds_total Total user and system CPU time spent in seconds.
# TYPE process_cpu_seconds_total counter
process_cpu_seconds_total 2232.660718999986 1622782805365

# HELP process_start_time_seconds Start time of the process since unix epoch in seconds.
# TYPE process_start_time_seconds gauge
process_start_time_seconds 1622618390

# HELP process_resident_memory_bytes Resident memory size in bytes.
# TYPE process_resident_memory_bytes gauge
process_resident_memory_bytes 266035200 1622782805365

# HELP process_virtual_memory_bytes Virtual memory size in bytes.
# TYPE process_virtual_memory_bytes gauge
process_virtual_memory_bytes 1190600704 1622782805365

# HELP process_heap_bytes Process heap size in bytes.
# TYPE process_heap_bytes gauge
process_heap_bytes 381554688 1622782805365

# HELP process_open_fds Number of open file descriptors.
# TYPE process_open_fds gauge
process_open_fds 35 1622782805365

# HELP process_max_fds Maximum number of open file descriptors.
# TYPE process_max_fds gauge
process_max_fds 3079104

# HELP nodejs_eventloop_lag_seconds Lag of event loop in seconds.
# TYPE nodejs_eventloop_lag_seconds gauge
nodejs_eventloop_lag_seconds 0.000214849 1622782805365

# HELP nodejs_active_handles_total Number of active handles.
```

```
# TYPE nodejs_active_handles_total gauge
nodejs_active_handles_total 20 1622782805365

# HELP nodejs_active_requests_total Number of active requests.
# TYPE nodejs_active_requests_total gauge
nodejs_active_requests_total 2 1622782805365

# HELP nodejs_heap_size_total_bytes Process heap size from node.js in bytes.
# TYPE nodejs_heap_size_total_bytes gauge
nodejs_heap_size_total_bytes 65863680 1622782805365

# HELP nodejs_heap_size_used_bytes Process heap size used from node.js in bytes.
# TYPE nodejs_heap_size_used_bytes gauge
nodejs_heap_size_used_bytes 54374768 1622782805365

# HELP nodejs_external_memory_bytes Nodejs external memory size in bytes.
# TYPE nodejs_external_memory_bytes gauge
nodejs_external_memory_bytes 103960700 1622782805365

# HELP nodejs_heap_space_size_total_bytes Process heap space size total from node.js in bytes.
# TYPE nodejs_heap_space_size_total_bytes gauge
nodejs_heap_space_size_total_bytes{space="read_only"} 524288 1622782805365
nodejs_heap_space_size_total_bytes{space="new"} 4194304 1622782805365
nodejs_heap_space_size_total_bytes{space="old"} 45846528 1622782805365
nodejs_heap_space_size_total_bytes{space="code"} 3928064 1622782805365
nodejs_heap_space_size_total_bytes{space="map"} 3158016 1622782805365
nodejs_heap_space_size_total_bytes{space="large_object"} 8212480 1622782805365

# HELP nodejs_heap_space_size_used_bytes Process heap space size used from node.js in bytes.
# TYPE nodejs_heap_space_size_used_bytes gauge
nodejs_heap_space_size_used_bytes{space="read_only"} 35200 1622782805365
nodejs_heap_space_size_used_bytes{space="new"} 1500384 1622782805365
nodejs_heap_space_size_used_bytes{space="old"} 41236472 1622782805365
nodejs_heap_space_size_used_bytes{space="code"} 2817280 1622782805365
nodejs_heap_space_size_used_bytes{space="map"} 1949376 1622782805365
nodejs_heap_space_size_used_bytes{space="large_object"} 6837664 1622782805365

# HELP nodejs_heap_space_size_available_bytes Process heap space size available from node.js
in bytes.
# TYPE nodejs_heap_space_size_available_bytes gauge
nodejs_heap_space_size_available_bytes{space="read_only"} 480384 1622782805365
nodejs_heap_space_size_available_bytes{space="new"} 561952 1622782805365
nodejs_heap_space_size_available_bytes{space="old"} 3666544 1622782805365
nodejs_heap_space_size_available_bytes{space="code"} 685280 1622782805365
nodejs_heap_space_size_available_bytes{space="map"} 1147096 1622782805365
nodejs_heap_space_size_available_bytes{space="large_object"} 17438006784 1622782805365

# HELP nodejs_version_info Node.js version info.
# TYPE nodejs_version_info gauge
nodejs_version_info{version="v10.24.1",major="10",minor="24",patch="1"} 1

# HELP des_csp_violations_total Number of CSP violations
# TYPE des_csp_violations_total counter
des_csp_violations_total 0
designer_ofs_iused_count{mount="/dev" } 373
designer_ofs_iused_count{mount="/dev/shm" } 1
designer_ofs_iused_count{mount="/run" } 522
designer_ofs_iused_count{mount="/sys/fs/cgroup" } 16
designer_ofs_iused_count{mount="/" } 36208
designer_ofs_iused_count{mount="/genesys" } 11
designer_ofs_iused_count{mount="/docker" } 110833
designer_ofs_iused_count{mount="/mnt" } 114981
designer_ofs_iused_count{mount="/mnt1" } 11
```

```
designer_ofs_iused_count{mount="/ofs" } 107805
designer_ofs_iused_count{mount="/ofs-settings" } 161
designer_ofs_iused_count{mount="/docker/overlay2/
5624a0b6fa40a83ef0f80e62495f2800a4222ffe7cb16034d24dcc08dcd23fbe/merged" } 110833
designer_ofs_iused_count{mount="/docker/overlay2/
f9f53cdeba0d5fae7da6ccee75a3ca76bb6899a15a891ce58ba2e69e1836d1ec/merged" } 110833
designer_ofs_iused_count{mount="/docker/overlay2/
31bcb428ac83e9c6006b447e7ddbfbf52df9ef8136f30e80a739bec516898236/merged" } 110833
designer_ofs_iused_count{mount="/docker/containers/
40049ec312ec61f39f7505707f6b311f06a536fddbfac7137540f3b997814a25/mounts/shm" } 1
designer_ofs_iused_count{mount="/docker/overlay2/
82e1de7a7904d66692fe3f84d64ddfcae58cb1b1c3db446da9196bc603f0936e/merged" } 110833
designer_ofs_iused_count{mount="/docker/containers/
e04cf03cf4dc36601bc47bc0ae7c7893a37e94c176c5b48d555c926587b1aeea/mounts/shm" } 1
designer_ofs_iused_count{mount="/run/user/0" } 1
designer_ofs_iused_total{mount="/dev" } 4054633
designer_ofs_iused_total{mount="/dev/shm" } 4058065
designer_ofs_iused_total{mount="/run" } 4058065
designer_ofs_iused_total{mount="/sys/fs/cgroup" } 4058065
designer_ofs_iused_total{mount="/" } 1966080
designer_ofs_iused_total{mount="/genesys" } 655360
designer_ofs_iused_total{mount="/docker" } 1310720
designer_ofs_iused_total{mount="/mnt" } 18317312
designer_ofs_iused_total{mount="/mnt1" } 1310720
designer_ofs_iused_total{mount="/ofs" } 1073741824
designer_ofs_iused_total{mount="/ofs-settings" } 1073741824
designer_ofs_iused_total{mount="/docker/overlay2/
5624a0b6fa40a83ef0f80e62495f2800a4222ffe7cb16034d24dcc08dcd23fbe/merged" } 1310720
designer_ofs_iused_total{mount="/docker/overlay2/
f9f53cdeba0d5fae7da6ccee75a3ca76bb6899a15a891ce58ba2e69e1836d1ec/merged" } 1310720
designer_ofs_iused_total{mount="/docker/overlay2/
31bcb428ac83e9c6006b447e7ddbfbf52df9ef8136f30e80a739bec516898236/merged" } 1310720
designer_ofs_iused_total{mount="/docker/containers/
40049ec312ec61f39f7505707f6b311f06a536fddbfac7137540f3b997814a25/mounts/shm" } 4058065
designer_ofs_iused_total{mount="/docker/overlay2/
82e1de7a7904d66692fe3f84d64ddfcae58cb1b1c3db446da9196bc603f0936e/merged" } 1310720
designer_ofs_iused_total{mount="/docker/containers/
e04cf03cf4dc36601bc47bc0ae7c7893a37e94c176c5b48d555c926587b1aeea/mounts/shm" } 4058065
designer_ofs_iused_total{mount="/run/user/0" } 4058065

designer_ofs_mount_availability{mount="/ofs" } 1
```

DAS metrics

[+] Some of the metrics exposed by DAS are as follows:

```
# HELP process_cpu_user_seconds_total Total user CPU time spent in seconds.
# TYPE process_cpu_user_seconds_total counter
process_cpu_user_seconds_total 5037.269190999957 1622782852031

# HELP process_cpu_system_seconds_total Total system CPU time spent in seconds.
# TYPE process_cpu_system_seconds_total counter
process_cpu_system_seconds_total 543.1887040000464 1622782852031

# HELP process_cpu_seconds_total Total user and system CPU time spent in seconds.
# TYPE process_cpu_seconds_total counter
process_cpu_seconds_total 5580.457894999958 1622782852031

# HELP process_start_time_seconds Start time of the process since unix epoch in seconds.
# TYPE process_start_time_seconds gauge
process_start_time_seconds 1622227763
```

```
# HELP process_resident_memory_bytes Resident memory size in bytes.
# TYPE process_resident_memory_bytes gauge
process_resident_memory_bytes 112586752 1622782852031

# HELP process_virtual_memory_bytes Virtual memory size in bytes.
# TYPE process_virtual_memory_bytes gauge
process_virtual_memory_bytes 963629056 1622782852031

# HELP process_heap_bytes Process heap size in bytes.
# TYPE process_heap_bytes gauge
process_heap_bytes 162521088 1622782852031

# HELP process_open_fds Number of open file descriptors.
# TYPE process_open_fds gauge
process_open_fds 23 1622782852031

# HELP process_max_fds Maximum number of open file descriptors.
# TYPE process_max_fds gauge
process_max_fds 3079104

# HELP nodejs_eventloop_lag_seconds Lag of event loop in seconds.
# TYPE nodejs_eventloop_lag_seconds gauge
nodejs_eventloop_lag_seconds 0.000255635 1622782852031

# HELP nodejs_active_handles_total Number of active handles.
# TYPE nodejs_active_handles_total gauge
nodejs_active_handles_total 9 1622782852031

# HELP nodejs_active_requests_total Number of active requests.
# TYPE nodejs_active_requests_total gauge
nodejs_active_requests_total 2 1622782852031

# HELP nodejs_heap_size_total_bytes Process heap size from node.js in bytes.
# TYPE nodejs_heap_size_total_bytes gauge
nodejs_heap_size_total_bytes 71675904 1622782852031

# HELP nodejs_heap_size_used_bytes Process heap size used from node.js in bytes.
# TYPE nodejs_heap_size_used_bytes gauge
nodejs_heap_size_used_bytes 31402264 1622782852031

# HELP nodejs_external_memory_bytes Nodejs external memory size in bytes.
# TYPE nodejs_external_memory_bytes gauge
nodejs_external_memory_bytes 105419 1622782852031

# HELP nodejs_heap_space_size_total_bytes Process heap space size total from node.js in bytes.
# TYPE nodejs_heap_space_size_total_bytes gauge
nodejs_heap_space_size_total_bytes{space="read_only"} 524288 1622782852031
nodejs_heap_space_size_total_bytes{space="new"} 33554432 1622782852031
nodejs_heap_space_size_total_bytes{space="old"} 28020736 1622782852031
nodejs_heap_space_size_total_bytes{space="code"} 2621440 1622782852031
nodejs_heap_space_size_total_bytes{space="map"} 2633728 1622782852031
nodejs_heap_space_size_total_bytes{space="large_object"} 4321280 1622782852031

# HELP nodejs_heap_space_size_used_bytes Process heap space size used from node.js in bytes.
# TYPE nodejs_heap_space_size_used_bytes gauge
nodejs_heap_space_size_used_bytes{space="read_only"} 35200 1622782852031
nodejs_heap_space_size_used_bytes{space="new"} 5107888 1622782852031
nodejs_heap_space_size_used_bytes{space="old"} 20406240 1622782852031
nodejs_heap_space_size_used_bytes{space="code"} 1801312 1622782852031
nodejs_heap_space_size_used_bytes{space="map"} 1100088 1622782852031
nodejs_heap_space_size_used_bytes{space="large_object"} 2953144 1622782852031
```

```
# HELP nodejs_heap_space_size_available_bytes Process heap space size available from node.js
in bytes.
# TYPE nodejs_heap_space_size_available_bytes gauge
nodejs_heap_space_size_available_bytes{space="read_only"} 480384 1622782852031
nodejs_heap_space_size_available_bytes{space="new"} 11390800 1622782852031
nodejs_heap_space_size_available_bytes{space="old"} 7069200 1622782852031
nodejs_heap_space_size_available_bytes{space="code"} 693344 1622782852031
nodejs_heap_space_size_available_bytes{space="map"} 1481240 1622782852031
nodejs_heap_space_size_available_bytes{space="large_object"} 1455042048 1622782852031

# HELP nodejs_version_info Node.js version info.
# TYPE nodejs_version_info gauge
nodejs_version_info{version="v10.24.1",major="10",minor="24",patch="1"} 1

# HELP fpm_accepted_conn accepted conn
# TYPE fpm_accepted_conn gauge
fpm_accepted_conn 37340 1622782856539

# HELP fpm_listen_queue listen queue
# TYPE fpm_listen_queue gauge
fpm_listen_queue 0 1622782856539

# HELP fpm_max_listen_queue max listen queue
# TYPE fpm_max_listen_queue gauge
fpm_max_listen_queue 1 1622782856539

# HELP fpm_listen_queue_len listen queue len
# TYPE fpm_listen_queue_len gauge
fpm_listen_queue_len 128 1622782856539

# HELP fpm_idle_processes idle processes
# TYPE fpm_idle_processes gauge
fpm_idle_processes 60 1622782856539

# HELP fpm_active_processes active processes
# TYPE fpm_active_processes gauge
fpm_active_processes 1 1622782856539

# HELP fpm_total_processes total processes
# TYPE fpm_total_processes gauge
fpm_total_processes 61 1622782856539

# HELP fpm_max_active_processes max active processes
# TYPE fpm_max_active_processes gauge
fpm_max_active_processes 2 1622782856539

# HELP fpm_max_children_reached max children reached
# TYPE fpm_max_children_reached gauge
fpm_max_children_reached 0 1622782856539

# HELP fpm_slow_requests slow requests
# TYPE fpm_slow_requests gauge
fpm_slow_requests 0 1622782856539



# HELP sdr_requests_received # requests received since DAS running
# TYPE sdr_requests_received counter
sdr_requests_received{ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"} 1998352 1622782853216
sdr_requests_received{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 2514 1622775582331
sdr_requests_received{ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"} 92 1622752878183
sdr_requests_received{ccid="36969587-76bf-410a-8764-532390bbbf90"} 55 1622768847657
```

```
sdr_requests_received{ccid="dapi_tenant"} 3 1622774405837
sdr_requests_received{ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"} 1 1622743333299

# HELP sdr_requests_rejected # requests rejected since DAS running
# TYPE sdr_requests_rejected counter


# HELP data_tables_requests_failures # failed datatables requests since DAS running
# TYPE data_tables_requests_failures counter
data_tables_requests_failures{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 80 1622715123986

# HELP data_tables_request_duration # data table requests duration since DAS running
# TYPE data_tables_request_duration histogram
data_tables_request_duration_bucket{le="50",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 189
data_tables_request_duration_bucket{le="100",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 201
data_tables_request_duration_bucket{le="500",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 214
data_tables_request_duration_bucket{le="+Inf",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 216
data_tables_request_duration_sum{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 9394
data_tables_request_duration_count{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 216

# HELP business_hours_requests_failures # failed business hours requests since DAS running
# TYPE business_hours_requests_failures counter

# HELP business_hours_request_duration # business hours requests duration since DAS running
# TYPE business_hours_request_duration histogram
business_hours_request_duration_bucket{le="20",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 26
business_hours_request_duration_bucket{le="50",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 37
business_hours_request_duration_bucket{le="+Inf",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"}
40
business_hours_request_duration_sum{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 904
business_hours_request_duration_count{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 40

# HELP special_days_requests_failures # failed special days requests since DAS running
# TYPE special_days_requests_failures counter

# HELP special_days_request_duration # special days requests duration since DAS running
# TYPE special_days_request_duration histogram
special_days_request_duration_bucket{le="20",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 34
special_days_request_duration_bucket{le="50",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 39
special_days_request_duration_bucket{le="+Inf",ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 40
special_days_request_duration_sum{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 810
special_days_request_duration_count{ccid="cdc2e2fd-406c-4bac-a857-c2744bcc902a"} 40

# HELP external_requests_failures # failed external requests since DAS running
# TYPE external_requests_failures counter

# HELP external_requests_timedout # timed out external requests since DAS running
# TYPE external_requests_timedout counter

# HELP external_requests_duration # external requests duration since DAS running
# TYPE external_requests_duration histogram

# HELP das_http_request_duration_seconds HTTP request latency
# TYPE das_http_request_duration_seconds histogram
das_http_request_duration_seconds_bucket{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="0.5"} 40
das_http_request_duration_seconds_bucket{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="1.5"} 40
das_http_request_duration_seconds_bucket{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="3.0"} 40
das_http_request_duration_seconds_bucket{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="+Inf"} 40
das_http_request_duration_seconds_bucket{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
```

```
a857-c2744bcc902a",le="0.5"} 216
das_http_request_duration_seconds_bucket{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="1.5"} 216
das_http_request_duration_seconds_bucket{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="3.0"} 216
das_http_request_duration_seconds_bucket{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="+Inf"} 216
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="0.
140
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="1.
140
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="3.
140
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="+I
140
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90",le="0.
139
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90",le="1.
139
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90",le="3.
139
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90",le="+I
139
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="0.
2
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="1.
2
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="3.
2
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="+I
2
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="0.
41957
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="1.
41957
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="3.
41957
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="+I
41957
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="0.5"} 4968
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="1.5"} 4968
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="3.0"} 4968
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="+Inf"} 4968
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="none",le="0.5"} 16
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="none",le="1.5"} 16
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="none",le="3.0"} 16
das_http_request_duration_seconds_bucket{req_type="metadata",ccid="none",le="+Inf"} 16
das_http_request_duration_seconds_bucket{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b",le="0.5"}
2
das_http_request_duration_seconds_bucket{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b",le="1.5"}
2
das_http_request_duration_seconds_bucket{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b",le="3.0"}
2
das_http_request_duration_seconds_bucket{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b",le="+Inf"
2
das_http_request_duration_seconds_bucket{req_type="other",ccid="none",le="0.5"} 1
das_http_request_duration_seconds_bucket{req_type="other",ccid="none",le="1.5"} 1
das_http_request_duration_seconds_bucket{req_type="other",ccid="none",le="3.0"} 1
das_http_request_duration_seconds_bucket{req_type="other",ccid="none",le="+Inf"} 1
```

```
das_http_request_duration_seconds_bucket{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="0.5"}
92
das_http_request_duration_seconds_bucket{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="1.5"}
92
das_http_request_duration_seconds_bucket{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="3.0"}
92
das_http_request_duration_seconds_bucket{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="+Inf"}
92
das_http_request_duration_seconds_bucket{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90",le="0.5"}
65
das_http_request_duration_seconds_bucket{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90",le="1.5"}
83
das_http_request_duration_seconds_bucket{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90",le="3.0"}
86
das_http_request_duration_seconds_bucket{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90",le="+Inf"}
86
das_http_request_duration_seconds_bucket{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="0.5"}
1
das_http_request_duration_seconds_bucket{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="1.5"}
1
das_http_request_duration_seconds_bucket{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="3.0"}
1
das_http_request_duration_seconds_bucket{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="+Inf"}
1
das_http_request_duration_seconds_bucket{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="0.5"}
1998352
das_http_request_duration_seconds_bucket{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="1.5"}
1998352
das_http_request_duration_seconds_bucket{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="3.0"}
1998352
das_http_request_duration_seconds_bucket{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="+Inf"}
1998352
das_http_request_duration_seconds_bucket{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="0.5"} 2594
das_http_request_duration_seconds_bucket{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="1.5"} 2594
das_http_request_duration_seconds_bucket{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="3.0"} 2594
das_http_request_duration_seconds_bucket{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="+Inf"} 2594
das_http_request_duration_seconds_bucket{req_type="php",ccid="none",le="0.5"} 36979
das_http_request_duration_seconds_bucket{req_type="php",ccid="none",le="1.5"} 36979
das_http_request_duration_seconds_bucket{req_type="php",ccid="none",le="3.0"} 36979
das_http_request_duration_seconds_bucket{req_type="php",ccid="none",le="+Inf"} 36979
das_http_request_duration_seconds_bucket{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="0.5"
231
das_http_request_duration_seconds_bucket{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="1.5"
231
das_http_request_duration_seconds_bucket{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="3.0"
231
das_http_request_duration_seconds_bucket{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6",le="+Inf
231
das_http_request_duration_seconds_bucket{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90",le="0.5"
497
das_http_request_duration_seconds_bucket{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90",le="1.5"
497
das_http_request_duration_seconds_bucket{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90",le="3.0"
497
das_http_request_duration_seconds_bucket{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90",le="+Inf
497
das_http_request_duration_seconds_bucket{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="0.5"
9
das_http_request_duration_seconds_bucket{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="1.5"
```

```
9
das_http_request_duration_seconds_bucket{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="3.0"
9
das_http_request_duration_seconds_bucket{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76",le="+Inf
9
das_http_request_duration_seconds_bucket{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="0.5"
7497
das_http_request_duration_seconds_bucket{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="1.5"
7497
das_http_request_duration_seconds_bucket{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="3.0"
7497
das_http_request_duration_seconds_bucket{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30",le="+Inf
7497
das_http_request_duration_seconds_bucket{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="0.5"} 9888
das_http_request_duration_seconds_bucket{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="1.5"} 9888
das_http_request_duration_seconds_bucket{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="3.0"} 9888
das_http_request_duration_seconds_bucket{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a",le="+Inf"} 9888
das_http_request_duration_seconds_count{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 40
das_http_request_duration_seconds_count{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 216
das_http_request_duration_seconds_count{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
140
das_http_request_duration_seconds_count{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90"}
139
das_http_request_duration_seconds_count{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
2
das_http_request_duration_seconds_count{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
41956
das_http_request_duration_seconds_count{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 4968
das_http_request_duration_seconds_count{req_type="metadata",ccid="none"} 16
das_http_request_duration_seconds_count{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b"}
2
das_http_request_duration_seconds_count{req_type="other",ccid="none"} 1
das_http_request_duration_seconds_count{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
92
das_http_request_duration_seconds_count{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90"}
86
das_http_request_duration_seconds_count{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
1
das_http_request_duration_seconds_count{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
1998352
das_http_request_duration_seconds_count{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 2594
das_http_request_duration_seconds_count{req_type="php",ccid="none"} 36979
das_http_request_duration_seconds_count{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
231
das_http_request_duration_seconds_count{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90"}
497
das_http_request_duration_seconds_count{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
9
das_http_request_duration_seconds_count{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
7497
das_http_request_duration_seconds_count{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 9888
das_http_request_duration_seconds_sum{req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 0.152
das_http_request_duration_seconds_sum{req_type="datatable",ccid="cdc2e2fd-406c-4bac-
```

```
a857-c2744bcc902a"} 2.799
das_http_request_duration_seconds_sum{req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
0
das_http_request_duration_seconds_sum{req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90"}
0
das_http_request_duration_seconds_sum{req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
0
das_http_request_duration_seconds_sum{req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
0
das_http_request_duration_seconds_sum{req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 0
das_http_request_duration_seconds_sum{req_type="metadata",ccid="none"} 0
das_http_request_duration_seconds_sum{req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b"}
0
das_http_request_duration_seconds_sum{req_type="other",ccid="none"} 0.001
das_http_request_duration_seconds_sum{req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
0.321
das_http_request_duration_seconds_sum{req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90"}
20.781
das_http_request_duration_seconds_sum{req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
0.007
das_http_request_duration_seconds_sum{req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
5526.6980000495
das_http_request_duration_seconds_sum{req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 8.508
das_http_request_duration_seconds_sum{req_type="php",ccid="none"} 20.103000000002
das_http_request_duration_seconds_sum{req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
0.027
das_http_request_duration_seconds_sum{req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90"}
0.093
das_http_request_duration_seconds_sum{req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
0
das_http_request_duration_seconds_sum{req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
8.363
das_http_request_duration_seconds_sum{req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 0.01
# HELP das_http_requests_total Number of HTTP requests
# TYPE das_http_requests_total counter
das_http_requests_total{status="4xx",req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 80
das_http_requests_total{status="4xx",req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90"}
5
das_http_requests_total{status="ok",req_type="businesshours",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 40
das_http_requests_total{status="ok",req_type="datatable",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 136
das_http_requests_total{status="ok",req_type="metadata",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
140
das_http_requests_total{status="ok",req_type="metadata",ccid="36969587-76bf-410a-8764-532390bbbf90"}
139
das_http_requests_total{status="ok",req_type="metadata",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
2
das_http_requests_total{status="ok",req_type="metadata",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
41957
das_http_requests_total{status="ok",req_type="metadata",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 4968
das_http_requests_total{status="ok",req_type="metadata",ccid="none"} 16
das_http_requests_total{status="ok",req_type="other",ccid="d1e853d0-dd36-45c6-a07c-97fedc41102b"}
2
das_http_requests_total{status="ok",req_type="other",ccid="none"} 1
das_http_requests_total{status="ok",req_type="php",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
92
das_http_requests_total{status="ok",req_type="php",ccid="36969587-76bf-410a-8764-532390bbbf90"}
```

```
81
das_http_requests_total{status="ok",req_type="php",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
1
das_http_requests_total{status="ok",req_type="php",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
1998352
das_http_requests_total{status="ok",req_type="php",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 2594
das_http_requests_total{status="ok",req_type="php",ccid="none"} 36979
das_http_requests_total{status="ok",req_type="static",ccid="038ae9c6-8e1e-4377-8040-2f07425612c6"}
231
das_http_requests_total{status="ok",req_type="static",ccid="36969587-76bf-410a-8764-532390bbbf90"}
497
das_http_requests_total{status="ok",req_type="static",ccid="3bca5841-4980-4210-b0e8-bf1988f44f76"}
9
das_http_requests_total{status="ok",req_type="static",ccid="77ed875a-4341-42b6-9abe-4f03d53dbf30"}
7497
das_http_requests_total{status="ok",req_type="static",ccid="cdc2e2fd-406c-4bac-
a857-c2744bcc902a"} 9888
# HELP nginx_metric_errors_total Number of nginx-lua-prometheus errors
# TYPE nginx_metric_errors_total counter
nginx_metric_errors_total 0
```

> **Important**
>
> In addition to the above metrics, we can obtain infrastructure related metrics by installing standard Prometheus clients in the Kubernetes cluster.

Alerts

This section provides a list of available alerts and information on enabling, disabling, and updating alerts.

Available Alerts

Microservice	Alarm	Description	Alert Name	Default Threshold	Default Interval (seconds)	Default Alert Severity
DES and DAS	Pod CPU Usage	Triggered when a pod's CPU utilization is beyond the threshold.	CPUUtilization	75%	180	CRITICAL
DES and DAS	Pod Memory Usage	Triggered when a pod's memory utilization is beyond the threshold.	MemoryUtilization	75%	180	CRITICAL

DES and DAS	Pod Restarts Count	Triggered when a pod's restart count is beyond the threshold.	containerRestartsAlert	5	180	CRITICAL
DES and DAS	Pod Ready Count	Triggered when a pod's ready count is less than the threshold (1).	containerReadyAlert	1	60	CRITICAL
DES and DAS	Deployment availability	Triggered when Designer/ DAS pod metrics are unavailable.	AbsentAlert	1	60	CRITICAL
DES and DAS	Azure Fileshare PVC Usage	Triggered when file share usage is greater than the threshold.	WorkspaceUtilization	80%	180	HIGH
DES and DAS	Health Status	Triggered when Designer/ DAS health status is 0.	Health	0	60	CRITICAL
DES and DAS	Workspace Health Status	Triggered when Designer/ DAS is not able to communicate with the workspace.	WorkspaceHealth	0	60	CRITICAL
DES	ElasticSearch Health Status	Triggered when Designer/ DAS is not able to reach the Elasticsearch server.	ESHealth	0	60	CRITICAL
DES	GWS Health Status	Triggered when Designer/ DAS is not able to reach the GWS server.	GWSHealth	0	60	CRITICAL

DAS	PHP Health Status	Triggered when Designer/ DAS experiences a PHP Health check failure.	PHPHealth	0	60	CRITICAL
DAS	Proxy Health Status	Triggered when Designer/ DAS experiences a Proxy Health check failure.	ProxyHealth	0	60	CRITICAL
DAS	Application 5XX Error Alarm	Triggered when DAS exceeds the allowed 5xx error count threshold specified here.	HTTP5XXCount	10	180	HIGH
DAS	Application 4XX Error Alarm	Triggered when DAS exceeds the 4xx error count threshold specified here.	HTTP4XXCount	100	180	HIGH
DAS	DAS PHP Latency Alert	Triggered when the average time taken by a PHP request is greater than the threshold (in seconds) specified here.	PhpLatency	10 seconds	180	HIGH
DAS	DAS HTTP Latency Alert	Triggered when the average time taken by a HTTP request is greater than the threshold (in seconds) specified	HTTPLatency	10 seconds	180	HIGH

		here.				

Enable alerts in Designer

To enable alerts in Designer, use either of the following methods:

Method 1: Enable Prometheus alerts in the values.yaml file.

```
designer:
    prometheus:
        alerts:
            enabled: true # this will be false by default.
```

Method 2: Find out the active deployment color and execute the below command in the corresponding deployment:

```
helm upgrade --install designer-blue -f designer-values.yaml designer-9.0.xx.tgz --set designer.deployment.strategy=blue-green --set designer.prometheus.alerts.enabled=true
```

Disable alerts in Designer

To disable or delete alerts, use either of the following methods:

Method 1: Disable Prometheus alerts in the values.yaml file.

```
designer:
    prometheus:
        alerts:
            enabled: false # this will be false default.
```

Method 2: Pass the below parameter along with the Helm upgrade command.

```
helm upgrade --install designer-blue -f designer-values.yaml designer-9.0.xx.tgz --set designer.deployment.strategy=blue-green --set designer.prometheus.alerts.enabled=false
```

Enable alerts in DAS

To enable alerts, use either of the following methods:

Method 1: Enable Prometheus alerts in the values.yaml file.

```
das:
    prometheus:
        alerts:
            enabled: true # this will be false default.
```

Method 2: Pass the below parameter along with the Helm upgrade command.

```
helm upgrade --install designer-das-blue -f designer-values.yaml designer-das-9.0.xx.tgz --set das.deployment.strategy=blue-green --set das.prometheus.alerts.enabled=true
```

Disable alerts in DAS

To disable or delete alerts, use either of the following methods:

Method 1: Disable Prometheus alerts in the values.yaml file.

```
das:
    prometheus:
        alerts:
            enabled: false # this will be false default.
```

Method 2: Pass the below parameter along with the Helm upgrade command.

```
helm upgrade --install designer-das-blue -f designer-values.yaml designer-das-9.0.xx.tgz --set das.deployment.strategy=blue-green --set das.prometheus.alerts.enabled=false
```

Update alert parameters

The following alert parameters can be updated:

- Alert Threshold (ALERT_PARAMETER_NAME: threshold)
- Alert Interval (ALERT_PARAMETER_NAME: interval)
- Alert Severity (ALERT_PARAMETER_NAME: AlertPriority)

Perform the following steps to update the above alerts:

1. Refer to the list of alerts and identify the name of the alert you want to update or modify.
2. Update the alert by adding a parameter in the below format in the values.yaml file:

```
designer:
     prometheus:
         alerts:
             :
                 :
                 :
```

For example, consider the CPU utilization alert. The alert name is CPUUtilization with a default threshold of 75, severity set to CRITICAL and interval set to 180s. To modify its threshold to 80, severity to HIGH, and interval to 120 seconds, you will have to make the following changes in the values.yaml file:

```
designer:
    prometheus:
        alerts:
            CPUUtilization:
                threshold: 80
                interval: 120
                AlertPriority: HIGH
```

Add new Prometheus alerts

If you want to add new alerts for the metrics available in the Prometheus server,

you can use a custom alert block.

> **Important**
> Currently, custom alert blocks support only simple PromQL expressions.

A simple PromQL expression contains the below elements:

```
METRIC_NAME               {LABEL_NAME1=LABEL_VALUE1, LABEL_NAME2=LABEL_VALUE2} OPERATOR THRESHOLD
cpu_utilization_percentage{service="designer",        contianer="designer-1"}     <      70
```

- To create custom alerts, define the alerts using the below format in the values.yaml file:

```
designer(das):
     prometheus:
         alerts:
             customalerts:
                 -
                   enabled: true ## we must set it to true to create custom alerts
                   name:
                   expr:
                     metric:
                     labels:
                           :
                           :
                   operator: OPERATOR # '' or '=' or '!='
                   interval:
                   threshold:
                   AlertPriority:
```

- Custom alerts is a list and you can add any number of alerts to it. Custom alerts appear as follows in the **alertfile.yaml** file:

```
groups:
     - name:
       rules:
         - alert: DESIGNER
           expr:  { =, =} OPERATOR
           for:
           labels:
               severity:
           annotations:
           summary: DESIGNER has crossed the threshold of
           information: DESIGNER has crossed the threshold of  for
```

- To disable custom alerts, set `designer.prometheus.alerts.customalerts..enabled` to `false` or remove the custom alerts from the list.

Expected output

The above instructions will create a Kubernetes custom resource object, `PrometheusRule`, and add its name to the Helm chart. After executing the above steps, you can check if the `PrometheusRule` resource object is created for Designer and DAS by using the following command `kubectl` command:

```
kubectl get prometheusrule # designer-prometheus-alerts or designer-das-prometheus-alerts
```

Grafana dashboard

To create a Grafana dashboard, you must run the Designer/DAS Helm chart with the deployment strategy set to `grafana`. The default Grafana dashboard will have a graph for almost all of the alerts listed above and in addition, a few graphs for metrics exposed by the Designer and DAS applications.

Enable Grafana dashboard

To enable the Grafana dashboard execute the following commands:

Designer

```
helm upgrade --install designer-dashboard -f designer-values.yaml designer-9.0.xx.tgz --set designer.deployment.strategy=grafana --set designer.grafana.enabled=true
```

DAS

```
helm upgrade --install designer-das-dashboard -f designer-das-values.yaml designer-das-9.0.xx.tgz --set das.deployment.strategy=grafana --set das.grafana.enabled=true
```

Disable Grafana dashboard

To disable the Grafana dashboard execute the following commands:

Designer

```
helm upgrade --install designer-dashboard -f designer-values.yaml designer-9.0.xx.tgz --set designer.deployment.strategy=grafana --set designer.grafana.enabled=false
```

DAS

```
helm upgrade --install designer-das-dashboard -f designer-das-values.yaml designer-das-9.0.xx.tgz --set das.deployment.strategy=grafana --set das.grafana.enabled=false
```

Expected output

The above steps will create a ConfigMap resource containing the Grafana **dashboard.json** file. The monitoring service needs to be configured to read the ConfigMap and create the Grafana dashboard.