# Voice Microservices Private Edition Guide

Dial Plan Service metrics and alerts

6/4/2026

# Contents

Find the metrics Dial Plan Service exposes and the alerts defined for Dial Plan Service.

| Service | CRD or annotations? | Port | Endpoint/Selector | Metrics update interval |
|---------|---------------------|------|-------------------|-------------------------|
| Dial Plan Service | Supports both CRD and annotations | 8800 | http://:8800/metrics | 30 seconds |

See details about:

- Dial Plan Service metrics
- Dial Plan Service alerts

## Metrics

You can query Prometheus directly to see all the metrics that the Voice Dial Plan Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available Dial Plan Service metrics not documented on this page.

| Metric and description | Metric details | Indicator of |
|------------------------|----------------|--------------|
| **dialplan_health_level**<br><br>Aggregated health level of the dialplan node for dependent services such as Redis and the Envoy sidecar connection:<br><br>-1 – fail<br>0 – starting<br>1 – degraded<br>2 – pass | **Unit:** N/A<br><br>**Type:** gauge<br>**Label:**<br>**Sample value:** 2 | Health |
| **dialplan_redis_state**<br><br>Current Redis connection state:<br><br>0 – disconnected<br>1 – connecting<br>2 – connected | **Unit:** N/A<br><br>**Type:** gauge<br>**Label:** redis_cluster_name<br>**Sample value:** 2 | Health |
| **dialplan_total_request**<br><br>Number of dialplan requests received. | **Unit:** N/A<br><br>**Type:** counter<br>**Label:** tenant, pod, operation_type<br>**Sample value:** | Traffic |
| **dialplan_failure_response** | **Unit:** N/A | Traffic |

| Metric and description | Metric details | Indicator of |
|---|---|---|
| The number of Dial Plan failure responses. | **Type:** counter<br>**Label:** tenant, pod, operation_type, status, reason<br>**Sample value:** | |
| **dialplan_response_time**<br><br>Dialplan request processing duration histogram, in ms. | **Unit:** milliseconds<br><br>**Type:** histogram<br>**Label:**<br>**Sample value:** | Latency |
| **dialplan_redis_cache_latency_msec**<br><br>Redis fetch latency, measured in milliseconds. | **Unit:** milliseconds<br><br>**Type:** histogram<br>**Label:** tenant<br>**Sample value:** | Latency |

## Alerts

The following alerts are defined for Dial Plan Service.

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| DialPlan processing time > 0.5 seconds | Warning | Actions:<br><br>• If the alarm is generated for all dialplan pods, then Redis or network delay might be the most probable cause.<br><br>• If the alarm is generated in a single dialplan pod, then it might be due to Envoy or a network issue. | dialplan_response_time | When the latency for 95% of the dial plan messages is more than 0.5 seconds for a duration of 5 minutes, then this warning alarm is raised for the {{ $labels.container }}. |
| DialPlan processing time > 2 seconds | Critical | Actions:<br><br>• If the alarm is generated for all dialplan pods, then Redis or network delay might be the | dialplan_response_time | If the latency for 95% of the dial plan messages is more than 2 seconds for a duration of 5 minutes, then this warning alarm is raised for the {{ $labels.container |

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| | | most probable cause.<br><br>• If the alarm is generated in a single dialplan pod, then it might be due to Envoy or a network issue. | | }}. |
| Aggregated service health failing for 5 minutes | Critical | Actions:<br><br>• Check the dialplan dashboard for Aggregated Service Health errors and, in case of a Redis error, first check for any issues/crashes in the pod and then restart Redis.<br><br>• In the case of an Envoy error, the dialplan container will be restarted by the liveness probe. If the issue still exists after that, restart the pod. | dialplan_health_level | Dependent services or the Envoy sidecar is not available for 5 minutes in the pod {{ $labels.pod }}. |
| Redis disconnected for 5 minutes | Warning | Actions:<br><br>• If the alarm is triggered for multiple services, make sure there are no issues with Redis and then restart Redis.<br><br>• If the alarm is triggered only for the pod {{ $labels.pod }}, check to see if | redis_state | Redis is not available for the pod {{ $labels.pod }} for 5 minutes. |

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| | | there is an issue with the pod. | | |
| Redis disconnected for 10 minutes | Critical | Actions:<br><br>• If the alarm is triggered for multiple services, make sure there are no issues with Redis and then restart Redis.<br><br>• If the alarm is triggered only for the pod {{ $labels.pod }}, check to see if there is an issue with the pod. | redis_state | Redis is not available for the pod {{ $labels.pod }} for 10 minutes. |
| Pod Failed | Warning | Actions:<br><br>• One of the containers in the pod has entered a failed state. Check the Kibana logs for the reason. | kube_pod_status_phase | Pod {{ $labels.pod }} failed. |
| Pod Unknown state | Warning | Actions:<br><br>• If the alarm is triggered for multiple services, make sure there are no issues with the Kubernetes cluster.<br><br>• If the alarm is triggered only for the pod {{ $labels.pod }}, check whether the image is correct and if the container is | kube_pod_status_phase | Pod {{ $labels.pod }} is in Unknown state for 5 minutes. |

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| | | starting up. | | |
| Pod Pending state | Warning | Actions:<br><br>• If the alarm is triggered for multiple services, make sure the Kubernetes nodes where the pod is running are alive in the cluster.<br><br>• If the alarm is triggered only for the pod {{ $labels.pod }}, check the health of the pod. | kube_pod_status_phase | Pod {{ $labels.pod }} is in the Pending state for 5 minutes. |
| Pod Not ready for 10 minutes | Critical | Actions:<br><br>• If this alarm is triggered, check whether the CPU is available for the pods.<br><br>• Check whether the port of the pod is running and serving the request. | kube_pod_status_ready | Pod {{ $labels.pod }} is in the NotReady state for 10 minutes. |
| Pod memory greater than 65% | Warning | High memory usage for pod {{ $labels.pod }}.<br><br>Actions:<br><br>• Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. | container_memory_working_set_bytes<br>kube_pod_container_resource_limits | Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes. |

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| | | • Check Grafana for abnormal load.<br><br>• Collect the service logs; raise an investigation ticket | | |
| Pod memory greater than 80% | Critical | Critical memory usage for pod {{ $labels.pod }}.<br><br>Actions:<br><br>• Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached.<br><br>• Check Grafana for abnormal load.<br><br>• Restart the service.<br><br>• Collect the service logs; raise an investigation ticket. | container_memory_working_set_bytes, kube_pod_container_resource_limits | Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes. |
| Pod CPU greater than 65% | Warning | High CPU load for pod {{ $labels.pod }}.<br><br>Actions:<br><br>• Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached.<br><br>• Check Grafana | container_cpu_usage_seconds_total, kube_pod_container_resource_limits | Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes. |

| Alert | Severity | Description | Based on | Threshold |
|---|---|---|---|---|
| | | for abnormal load.<br><br>• Collect the service logs; raise an investigation ticket. | | |
| Pod CPU greater than 80% | Critical | Critical CPU load for pod {{ $labels.pod }}.<br><br>Actions:<br><br>• Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached.<br><br>• Check Grafana for abnormal load.<br><br>• Restart the service.<br><br>• Collect the service logs; raise an investigation ticket. | container_cpu_usage_seconds_total, kube_pod_container_resource_limits | Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes. |