



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Intelligent Workload Distribution Private Edition Guide

1/17/2022

Table of Contents

Overview	
About IWD	6
Architecture	7
High availability and disaster recovery	9
Configure and deploy	
Before you begin	10
Configure IWD	14
Provision IWD	18
Deploy IWD	24
Upgrade, rollback, or uninstall IWD	28
Observability	
Observability in Intelligent Workload Distribution	30
IWD metrics and alerts	33

Contents

- [1 Overview](#)
- [2 Configure and deploy](#)
- [3 Observability](#)

Find links to all the topics in this guide.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Intelligent Workload Distribution (IWD) is a service available with the Genesys Multicloud CX private edition offering.

Overview

Learn more about IWD, its architecture, and how to support high availability and disaster recovery.

- About IWD
- Architecture
- High availability and disaster recovery

Configure and deploy

Find out how to configure and deploy IWD.

- Before you begin
- Configure IWD
- Provision IWD
- Deploy IWD
- Upgrade, rollback, or uninstall IWD

Observability

Learn how to monitor IWD with metrics and logging.

- IWD metrics and alerts

About IWD

Learn about IWD and how it works in Genesys Multicloud CX private edition.

Related documentation:

-
-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

IWD takes work items from existing enterprise software applications (such as ERP, BPM, DCM, Salesforce) and homegrown systems, analyzes the business context of the work item—for example, the associated business process, product requested, or value of the customer making the request—and creates a Universal Queue, sorted on business value, that ensures that the most critical or highest-value work items are distributed to the right resource at the right time, regardless of media type, system or location.

With IWD, enterprises can effectively manage all customer service resources and business processes across the enterprise, going beyond the walls of the formal contact center and into other areas of the business like branch offices and experts in the back office.

Architecture

Learn about IWD architecture.

Related documentation:

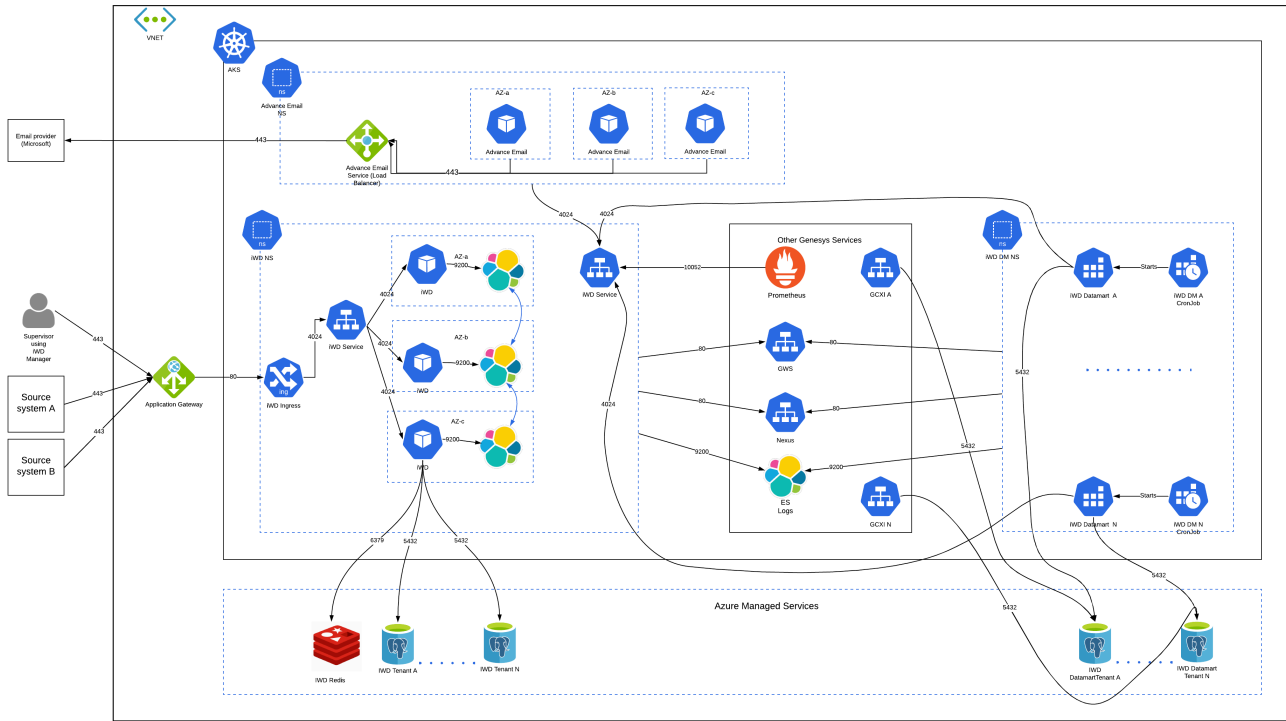
-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

The architecture diagram in this topic illustrates a sample deployment of IWD. The diagram shows Azure as the reference implementation.

For more information on the Genesys Multicloud CX private edition architecture, refer to the Architecture topic in the *Setting up Genesys Multicloud CX private edition* document.



High availability and disaster recovery

Find out how this service provides disaster recovery in the event the service goes down.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Service	High Availability	Disaster Recovery	Where can you host this service?
Intelligent Workload Distribution	N = 1 (singleton)	Not supported	Primary unit only

This information is under development: Flagged items aren't yet confirmed or have info coming soon; Checked items are valid.

See High Availability information for all services: High availability and disaster recovery

IWD fails if either postgres or Redis fails, or becomes unavailable.

Before you begin

Contents

- [1 Limitations and assumptions](#)
- [2 Download the Helm charts](#)
- [3 Third-party prerequisites](#)
- [4 Storage requirements](#)
- [5 Network requirements](#)
- [6 Browser requirements](#)
- [7 Genesys dependencies](#)
- [8 GDPR support](#)

Find out what to do before deploying IWD.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Limitations and assumptions

The current version of IWD:

- supports single-region model of deployment only
- requires dedicated PostgreSQL deployment per customer

Download the Helm charts

IWD in Genesys Multicloud CX private edition includes the following containers:

- iwd

The service also includes a Helm chart, which you must deploy to install the required containers for IWD:

- iwd

See Helm Charts and Containers for IWD and IWD Data Mart for the Helm chart version you must download for your release.

To download the Helm chart, navigate to the **iwd** folder in the JFrog repository. For information about how to download the Helm charts, see Downloading your Genesys Multicloud CX containers.

Third-party prerequisites

Third-party services

Name	Version	OpenShift	GKE	Purpose	Shared service?	Notes
Elasticsearch	7.x	Elasticsearch (ECK) Operator	OpenSearch Helm chart	Used for text searching and indexing. Deployed per service that needs Elasticsearch during runtime.	Yes	Dedicated - one per deployment of IWD
Redis	6.x	Redis Enterprise Operator	Redis Helm chart	Used for caching. Only distributions of Redis that support Redis cluster mode are supported, however, some services may not support cluster mode.	No	Dedicated - one per deployment of IWD
PostgreSQL	11.x			Relational database.	Optional	Dedicated instance for each tenant (recommended). In case of low load, one instance can host multiple DBs for multiple tenants (supported)

Storage requirements

All data is stored in the PostgreSQL, Elasticsearch, and Digital Channels which are external to IWD.

Sizing of Elasticsearch depends on the load. Allow on average 15 KB per work item, 50 KB per email. This can be adjusted depending on the size of items processed.

Network requirements

External Connections: IWD allows customer to configure webhooks. If configured, this establishes an HTTP or HTTPS connection to the configured host or port.

Browser requirements

Not applicable

Genesys dependencies

The following Genesys services are required:

- Genesys authentication service (GAuth)
- Universal Contact Service (UCS)
- Interaction Server
- Digital Channels (Nexus)

For the order in which the Genesys services must be deployed, refer to the Order of services deployment topic in the *Setting up Genesys Multicloud CX private edition* document.

GDPR support

Content coming soon

Configure IWD

Contents

- [1 Override Helm chart values](#)
- [2 Configure Kubernetes](#)
 - [2.1 ConfigMaps](#)
 - [2.2 Create the pull secret](#)
- [3 Configure security](#)
 - [3.1 Arbitrary UIDs in OpenShift](#)

Learn how to configure IWD.

Related documentation:

-
-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Override Helm chart values

The following table provides information on the IWD deployment settings that can be configured in the **values.yaml** file:

Parameter	Description	Default
<code>`deploymentType`</code>	Deployment type. Only two possible values are supported: Deployment, ReplicaSet	Deployment
<code>`replicaCount`</code>	Number of pods to be created	1
<code>`image.registry`</code>	Docker registry for iWD	pureengage-docker-staging.jfrog.io
<code>`image.repository`</code>	iWD Image name	nexus/iwd
<code>`image.pullPolicy`</code>	Image pull policy	IfNotPresent
<code>`image.pullSecrets`</code>	Specify docker-registry secret names as an array	[]
<code>`affinity`</code>	Map of node/pod affinities	{}
<code>`nodeSelector`</code>	Node labels for pod assignment	{}
<code>`tolerations`</code>	Tolerations for pod assignment	nil
<code>`priorityClassName`</code>	Priority class name	
<code>`podSecurityContext`</code>	Pod security context	{}
<code>`securityContext`</code>	Security context	{}
<code>`podDisruptionBudget.enabled`</code>	Enable or disable pod disruption budget	false
<code>`podDisruptionBudget.minAvailable`</code>	Set minimal number of pods	1

Parameter	Description	Default
	available during the disruption	
`podAnnotations`	Add annotations to pods	{}
`podLabels`	Add custom labels to pods	{}
`hpa.enabled`	Enable or disable Horizontal Pod Autoscaler (HPA)	false
`hpa.minReplicas`	Minimal replicas count for HPA	1
`hpa.maxReplicas`	Maximal replicas count for HPA	10
`hpa.targetCPUPercent`	Specify target CPU utilization for HPA	60
`resources.limits.cpu`	Maximum amount of CPU K8s allocates for container	2000m
`resources.limits.memory`	Maximum amount of Memory K8s allocates for container	2000Mi
`resources.requests.cpu`	Guaranteed CPU allocation for container	300m
`resources.requests.memory`	Guaranteed Memory allocation for container	500Mi
`serviceAccount.create`	Specifies whether a service account should be created	false
`serviceAccount.annotations`	Annotations to add to service account	{}
`serviceAccount.name`	Service account name	""
`existingSecret`	Specify Secret name to read application secrets from	nil
`gauth.auth.url`	URL to Authentication service	nil
`gauth.auth.redirectUrl`	Redirect URL to Authentication service	nil
`redis.nodes`	Comma separate list of Redis nodes to connect	nil
`redis.useCluster`	Redis deployment mode	false
`redis.enableTLS`	Either to use TLS on Redis connection	false
`redis.password`	Access key for Redis authentication	nil
`nexus.url`	URL to Nexus	nil
`nexus.apikey`	Nexus API key	nil
`service.type`	Service type	ClusterIP
`monitoring.enabled`	Enable or disable pod monitor	false
`monitoring.alarms`	Create PrometheusRule k8s object with alarm definitions	false
`monitoring.dashboards`	Create ConfigMap with Grafana Dashboards	false

Parameter	Description	Default
<code>`networkPolicies.enabled`</code>	Enable or disable network policies	false
<code>`dnsConfig.options`</code>	DNS Configuration options	{ name: ndots, value: "3" }

Configure Kubernetes

ConfigMaps

Not applicable as all required ConfigMaps are created via Helm Chart basing on the provided values.

Create the pull secret

Use the following code snippet as an example of how to create pull secret:

```
kubectll create secret docker-registry mycred --docker-server=pureengage.jfrog.io --docker-username= --docker-password=
```

You can add *mycred* to Helm override values by setting **image.pullSecrets** to *[mycred]*.

For OpenShift, you may add the *mycred* secret in current namespace as the default pull secret:

```
oc secrets link default mycred --for=pull
```

Configure security

Arbitrary UIDs in OpenShift

To use arbitrary UIDs in your OpenShift deployment, ensure the following settings:

```
podSecurityContext:  
  runAsNonRoot: true  
  runAsUser: null  
  runAsGroup: 0  
  fsGroup: null
```

Provision IWD

Contents

- [1 Provisioning via IWD API](#)
 - [1.1 Create tenant request](#)
 - [1.2 Parameters](#)
 - [1.3 Delete tenant request](#)
- [2 Manual provisioning](#)

- Administrator

Learn how to provision IWD.

Related documentation:

-
-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Important

Provisioning must be done after deploying IWD.

Workload Manager (UI) uses Roles. Agents must be assigned appropriate Roles.

All other provisioning is done through the tenant provisioning which can be performed manually or via IWD API.

Provisioning via IWD API

The following endpoints are available:

- POST /provisioning
- PUT /provisioning
- DELETE /provisioning/ccid

Currently, PUT and POST serve the same purpose.

Create tenant request

The following request assembles configurations for **GAPI**, **iWD**, and **iWDEmail**

services, requested API keys provided in the **iwd.apiKeys** object, and provisions them in the Digital Channels database using Digital Channels API.

POST /iwd/v3/provisioning

BODY:

```
{
  "tenant": {
    "name": "tenant-name",
    "ccid": "ccid",
    "apiKey": "API Key"
  },
  "iwd": {
    "url": "",
    "db": {
      "host": "",
      "port": 5434,
      "database": "bf",
      "user": "bf",
      "password": "bf",
      "ssl": 'false'
    },
    "apiKeys": {
      "IWD_APIKEY_TENANT": "123"
    }
  },
  "iwdEmail": {
    "url": "N/A"
  }
}
```

Parameters

The following parameters are supported:

Parameter name	Datatype	Required	Description
tenant	object	true	Tenant to provision
tenant.name	string	true	Tenant name (GWS domain)
tenant.id	string	true	Short tenant ID
tenant.ccid	string	true	GWS contact center ID
tenant.apikey	string	true	API key, issued by Digital Channels for a given tenant
iwd	object	true	IWD Service configuration
iwd.url	string	true	URL where IWD service is launched
iwd.db	object	true	IWD DB connection configuration
iwd.db.host	string	true	DB host
iwd.db.port	number	true	DB port
iwd.db.database	string	true	DB name
iwd.db.user	string	true	DB user
iwd.db.password	string	true	DB password
iwd.db.ssl	boolean	true	Use secure connection to DB or not
iwdEmail	object	true	IWD Email Service configuration
iwdEmail.url	string	true	URL where IWD Email service is launched, should be N/A
iwd.apiKeys	object	true	Tenant API keys for using various services
iwd.apiKeys.IWD_APIKEY_TENANT	string	true	Key for submitting API requests from customer
iwd.apiKeys.IWD_APIKEY_IWD	string	false	Required if you plan to deploy IWD Data Mart and use IWD reports

Response

The following success response is returned when the tenant is provisioned successfully:

```
{
  created: 'true'
}
```

As a result, the following records are created in the Digital Channels database:

- GAPI service with GWS API key, GWS client secret, GWS username, and token

- IWD service with default options, categories, filters, prioritization, and secret
- Email service with the default options and mailboxes
- API keys

An error is returned if any of the required parameters is missing.

Delete tenant request

The following request removes the specified tenant provisioning:

```
DELETE /iwd/v3/provisioning/ccid?[service=String || deleteTenant=Boolean]
```

Path parameters

Parameter name	Datatype	Required	Description
ccid	string	true	Contact Center id

Query parameters

A minimum of one parameter is mandatory.

Parameter name	Datatype	Required	Description
service	string	true (at least one)	Specific services names to be deleted from the database
deleteTenant	boolean		Flag to delete all tenant services from Digital Channels database

Delete tenant examples:

To delete the specified services from the Digital Channels database:

```
DELETE /iwd/v3/provisioning/ccid?service=iWD,iWDEmail
```

To delete all tenant services and tenant configuration from the Digital Channels database:

```
DELETE /iwd/v3/provisioning/ccid?deleteTenant=true
```

Manual provisioning

Example:

```
curl 'http://iwd.iwd:4024/iwd/v3/provisioning' \  
-H 'Content-Type: application/json; charset=utf-8' \  
-H 'x-api-key: ed99c91d-dd18-4c96-af8e-86f8e8105bc4' \  

```

Provision IWD

```
--data '{"tenant":{"id":"100",  
"name":"t100","ccid":"d846c51e-1fe8-4118-bf32-cf0b4ef29032","apiKey":"22552b96-8783-46a9-b0eb-075ddfa8893e"},"i  
rw.infra.svc.cluster.local","port":5432,"database":"iwd","user":"iwd","password":"iwd","ssl":false},"apiKeys":{"  
A"}}'
```

Deploy IWD

Contents

- [1 Prepare your environment](#)
- [2 Deploy](#)
- [3 Validate the deployment](#)

Learn how to deploy IWD.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Prepare your environment

Important

- Make sure to review [Before you begin](#) for the full list of prerequisites required to deploy IWD.
- The sample code snippets and install commands in this document use an example version of IWD. Ensure that you replace the example version with the version that is applicable for your deployment.

1. Create a new project using the following command:

```
kubectl create namespace iwd
```

For OpenShift:

```
oc get clusterversion  
oc new-project iwd
```

2. Create a pull secret for accessing the JFrog registry. See [Create the pull secret](#).
3. Download the IWD helm chart from the JFrog repository. See [Download the Helm charts](#).
4. Create gauth Client.
IWD requires *clientId* and *clientSecret* registered in Auth Service. These have to be provided during Helm Chart deployment. Create new client credentials if they are not already created . Refer to the [GWS documentation](#) for more information.

Deploy

Extract parameters from chart to see multiple (default) values used to fine tune the installation.

```
$ helm show values iwd-100.0.0741322.tgz > values.yaml
```

For information on parameters and values in the **values.yaml** file, see [Override Helm chart values](#).

Sample override file:

```
replicaCount: 1

image:
  registry: pureengage-docker-staging.jfrog.io
  repository: nexus/iwd
  pullSecrets: []

gauth:
  auth:
    url: http://gauth-auth.gauth
    redirectUrl: https://gauth.{{domain}}

redis:
  nodes: redis://infra-redis-redis-cluster.infra.svc.cluster.local:6379
  useCluster: true
  enableTLS: false
  password:

gws:
  url: http://gauth-auth.gauth
  clientId:
  clientSecret:
  apiKey:

ingress:
  enabled: true
  hosts:
    - host: iwd.{{domain}}
      paths:
        - path: '/iwd/'
          port: 4024
  tls:
    - hosts:
        - iwd.{{domain}}
      secretName: letsencrypt

nexus:
  url: http://nexus.nexus
  apikey:

elasticsearch:
  host: elastic-es-http.infra.svc.cluster.local
  port: 9200
```

Install IWD using the following command (replace with applicable values):

```
helm install iwd ./iwd-.tgz -f override_values.yaml --namespace=iwd
```

Add route (for OpenShift deployment only):

Deploy IWD

```
oc create route edge --service=iwd --hostname=iwd.apps. --port=api --path=/iwd
```

Validate the deployment

Watch the helm output at the end of installation. It provides the status and additional information about where to log in to the IWD UI.

See the following sample output:

```
Release "iwd" has been upgraded. Happy Helming!  
NAME: iwd  
LAST DEPLOYED: Tue Jul 13 10:18:07 2021  
NAMESPACE: iwd  
STATUS: deployed  
REVISION: 1  
TEST SUITE: None  
NOTES:  
Please be patient while iwd 100.0.0741322 is being deployed
```

Upgrade, rollback, or uninstall IWD

Contents

- [1 Upgrade IWD](#)
- [2 Rollback IWD](#)
- [3 Uninstall IWD](#)

Learn how to upgrade, rollback or uninstall IWD.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Upgrade IWD

Use the following sample command to upgrade:

```
$ helm upgrade -f ./values.yaml iwd iwd-900.67.1121.tgz -n iwd
```

Rollback IWD

Use **helm rollback** to rollback to the previous revision.

Uninstall IWD

Use the following command to uninstall IWD.

```
$ helm uninstall iwd -n iwd
```

Observability in Intelligent Workload Distribution

Contents

- **1 Monitoring**
 - 1.1 Enable monitoring
 - 1.2 Configure metrics
- **2 Alerting**
 - 2.1 Configure alerts
- **3 Logging**

Learn about the logs, metrics, and alerts you should monitor for Intelligent Workload Distribution.

Related documentation:

-
-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Monitoring

Private edition services expose metrics that can be scraped by Prometheus, to support monitoring operations and alerting.

- As described on Monitoring overview and approach, you can use a tool like Grafana to create dashboards that query the Prometheus metrics to visualize operational status.
- As described on Customizing Alertmanager configuration, you can configure Alertmanager to send notifications to notification providers such as PagerDuty, to notify you when an alert is triggered because a metric has exceeded a defined threshold.

The services expose a number of Genesys-defined and third-party metrics. The metrics that are defined in third-party software used by private edition services are available for you to use as long as the third-party provider still supports them. For descriptions of available Intelligent Workload Distribution metrics, see:

- IWD metrics

See also System metrics.

Enable monitoring

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
IWD	Both or either, depends on harvester	Default is 4024 (overridden by values)	/iwd/v3/metrics	15 sec recommended, depends on harvester

Configure metrics

Metrics are available when requested. No additional configuration is required.

Alerting

Private edition services define a number of alerts based on Prometheus metrics thresholds.

Important

While you can use general third-party functionality to create rules to trigger alerts based on metrics values you specify, private edition does not enable you to create custom alerts, and Genesys does not provide support for custom alerting.

For descriptions of available Intelligent Workload Distribution alerts, see:

- IWD alerts

Configure alerts

Private edition services define a number of alerts by default (for Intelligent Workload Distribution, see the pages linked to above). No further configuration is required.

The alerts are defined as **PrometheusRule** objects in a **prometheus-rule.yaml** file in the Helm charts. As described above, Intelligent Workload Distribution does not support customizing the alerts or defining additional **PrometheusRule** objects to create alerts based on the service-provided metrics.

Logging

Logging is done to *stdout*.

IWD metrics and alerts

Find the metrics IWD exposes and the alerts defined for IWD.

Related documentation:

-

Early Adopter Program

Genesys Multicloud CX private edition is being released to pre-approved customers as part of the Early Adopter Program. Please note that the documentation and the product are subject to change. For more details about the program, please contact your Genesys representative.

Contents

- [1 Metrics](#)
- [2 Alerts](#)

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
IWD	Both or either, depends on harvester	Default is 4024 (overridden by values)	/iwd/v3/metrics	15 sec recommended, depends on harvester

Metrics

Metric and description	Metric details	Indicator of
iwd_redis_connections_established Current number of established Redis connections	Unit: Type: gauge Label: Sample value: 0	
iwd_redis_connections_reconnecting Current number of reconnecting Redis connections	Unit: Type: gauge Label: Sample value: 0	
iwd_redis_connections_ready Current number of ready Redis connections	Unit: Type: gauge Label: Sample value: 1	
iwd_redis_duration_until_ready Duration until ready state reached	Unit: Type: histogram Label: 'le' Sample value: 0, 1, 39	
iwd_redis_errors_total Total number of Redis connection errors	Unit: Type: counter Label: Sample value: 0	
iwdTenantDB_db_connect_total The total number of all database connection requests	Unit: Type: counter Label: 'db' Sample value: 1252424, 1457770	
iwdTenantDB_db_disconnect_total The total number of all database disconnection requests	Unit: Type: counter Label: 'db' Sample value: 1252424, 1457770	
iwdTenantDB_db_request_total The total number of all Database	Unit: Type: counter	

Metric and description	Metric details	Indicator of
requests sent	Label: 'db' Sample value: 4850730, 5056452	
iwdTenantDB_db_success_total The total number of all all Database requests executed successfully	Unit: Type: counter Label: 'db', 'command' Sample value: 2307896, 2126805, 1221394, 1450355	
iwdTenantDB_db_errors_total The total number of all Database errors	Unit: Type: counter Label: 'db', 'code' Sample value: 131, 5, 4	
iwdTenantDB_db_request_duration_milliseconds Database transaction duration	Unit: Type: histogram Label: 'le', 'db', 'method' Sample value: 2290844, 2306385, 2307241, 2307894	
iwd_process_cpu_user_seconds_total Total user CPU time spent in seconds.	Unit: Type: counter Label: Sample value: 1634045655571	
iwd_process_cpu_system_seconds_total Total system CPU time spent in seconds.	Unit: Type: counter Label: Sample value: 1634045655571	
iwd_process_cpu_seconds_total Total user and system CPU time spent in seconds.	Unit: Type: counter Label: Sample value: 1634045655571	
iwd_process_start_time_seconds Start time of the process since unix epoch in seconds.	Unit: Type: gauge Label: Sample value: 1633992102	
iwd_process_resident_memory_bytes Resident memory size in bytes.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_process_virtual_memory_bytes Virtual memory size in bytes.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_process_heap_bytes	Unit:	

Metric and description	Metric details	Indicator of
Process heap size in bytes.	Type: gauge Label: Sample value: 1634045655572	
iwd_process_open_fds Number of open file descriptors.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_process_max_fds Maximum number of open file descriptors.	Unit: Type: gauge Label: Sample value: 197176	
iwd_nodejs_eventloop_lag_seconds Lag of event loop in seconds.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_nodejs_active_handles Number of active libuv handles grouped by handle type. Every handle type is C++ class name.	Unit: Type: gauge Label: 'type' Sample value: 17, 1, 69	
iwd_nodejs_active_handles_total Total number of active handles.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_nodejs_active_requests Number of active libuv requests grouped by request type. Every request type is C++ class name.	Unit: Type: gauge Label: 'type' Sample value: 2	
iwd_nodejs_active_requests_total Total number of active requests.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_nodejs_heap_size_total_bytes Process heap size from node.js in bytes.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_nodejs_heap_size_used_bytes Process heap size used from node.js in bytes.	Unit: Type: gauge Label: Sample value: 1634045655572	
iwd_nodejs_external_memory_bytes	Unit:	

Metric and description	Metric details	Indicator of
Nodejs external memory size in bytes.	Type: gauge Label: Sample value: 1634045655572	
iw_d_nodejs_heap_space_size_total_bytes Process heap space size total from node.js in bytes.	Unit: Type: gauge Label: 'space' Sample value: 262144, 16777216, 130428928, 6721536	
iw_d_nodejs_heap_space_size_used_bytes Process heap space size used from node.js in bytes.	Unit: Type: gauge Label: 'space' Sample value: 32808, 1479672, 92634792, 4852384	
iw_d_nodejs_heap_space_size_available_bytes Process heap space size available from node.js in bytes.	Unit: Type: gauge Label: 'space' Sample value: 0, 6899976, 37040456, 1542496	
iw_d_nodejs_version_info Node.js version info.	Unit: Type: gauge Label: 'version', 'major', 'minor', 'patch' Sample value: 1	
iw_d_request_total The total number of all API requests received	Unit: Type: counter Label: Sample value: 177186	
iw_d_success_total The total number of all API requests with success response	Unit: Type: counter Label: 'ccid' Sample value: 21400, 46769, 48539	
iw_d_errors_total The total number of all API requests with error response	Unit: Type: counter Label: 'ccid' Sample value: 438, 49, 4	
iw_d_client_error_total The total number of all API requests with client error response	Unit: Type: counter Label: 'ccid' Sample value: 204, 49, 2	
iw_d_server_error_total The total number of all API requests with server error response	Unit: Type: counter Label: 'ccid' Sample value: 234, 2	

Metric and description	Metric details	Indicator of
<p>iwd_api_request_total</p> <p>The total number of all API requests</p>	<p>Unit:</p> <p>Type: counter Label: 'method', 'path', 'code', 'ccid' Sample value: 3570, 3584, 25079, 19500</p>	
<p>iwd_api_request_long</p> <p>Number of API requests that took long time to execute</p>	<p>Unit:</p> <p>Type: counter Label: Sample value:</p>	
<p>iwd_api_request_closed</p> <p>Number of API requests that expired before response was sent</p>	<p>Unit:</p> <p>Type: counter Label: 'method', 'path' Sample value: 3, 14, 4, 9</p>	
<p>iwd_api_request_duration_milliseconds</p> <p>API requests duration</p>	<p>Unit:</p> <p>Type: histogram Label: 'le', 'method', 'path', 'code', 'ccid' Sample value: 6, 2708, 3502, 3570</p>	
<p>iwd_api_blacklist</p> <p>Total number of blacklisted requests</p>	<p>Unit:</p> <p>Type: counter Label: Sample value:</p>	
<p>iwd_cometd_connections_total</p> <p>The current number of client cometd connections to GWS</p>	<p>Unit:</p> <p>Type: gauge Label: 'type', 'ccid' Sample value: 27, 1, 41</p>	
<p>iwd_cometd_errors_total</p> <p>The total number of client cometd errors</p>	<p>Unit:</p> <p>Type: counter Label: 'type', 'ccid' Sample value: 1</p>	
<p>iwd_cometd_request_errors_total</p> <p>The total number of client cometd error response from GWS</p>	<p>Unit:</p> <p>Type: counter Label: 'type', 'name', 'ccid', 'domain' Sample value: 13026, 64, 1, 102</p>	
<p>iwd_cometd_request_current</p> <p>The current number of client cometd requests to GWS</p>	<p>Unit:</p> <p>Type: gauge Label: 'type', 'name', 'ccid', 'domain' Sample value: -6318, -11825, 0</p>	
<p>iwd_cometd_request_duration_milliseconds</p> <p>The cometd request duration (to GWS)</p>	<p>Unit:</p> <p>Type: histogram Label: 'le', 'type', 'name', 'ccid', 'domain' Sample value: 6298, 6320, 6345, 6395</p>	
<p>iwd_cometd_request_duration_milliseconds_summary</p>	<p>Unit:</p>	

Metric and description	Metric details	Indicator of
The cometd request duration (to GWS): summary	Type: summary Label: 'quantile', 'type', 'name', 'ccid', 'domain' Sample value: 0, 930700, 6577, 89959	
iwd_cometd_events_total The total number of client cometd events from GWS	Unit: Type: counter Label: 'type', 'name', 'ccid', 'domain' Sample value: 80, 443, 346, 17	

Alerts

No alerts are defined for IWD.