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.

Manage your Contact Center in Agent Setup

Voice channel options

1/12/2026

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.

Interaction Server Private Edition Gui

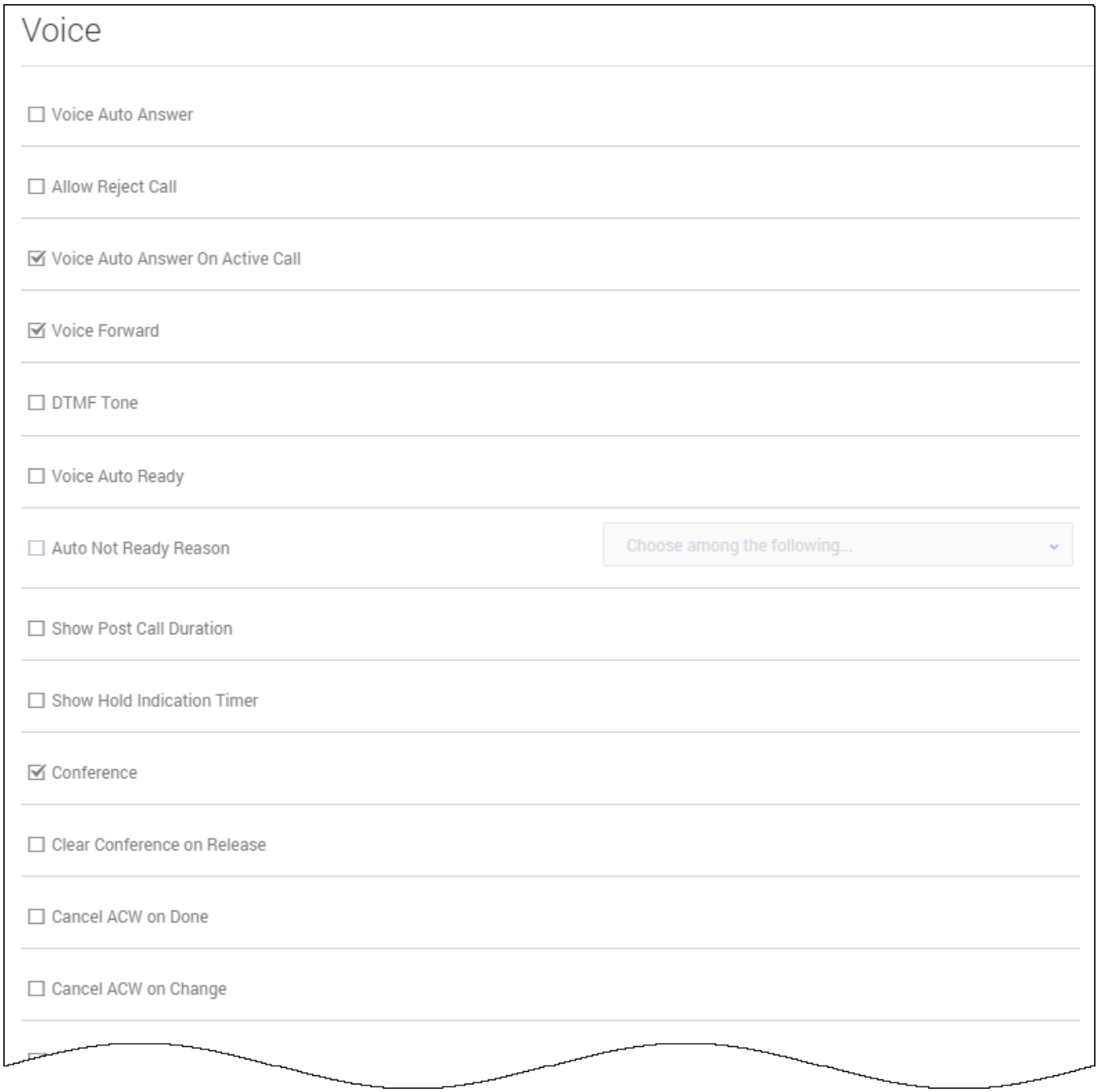
Interaction Server (IXN) metrics and a

-
- Administrator

Learn about the Voice channel options available in Agent Setup.

Related documentation:

-
-



In the **Voice** section of the **Contact Center Settings** page, configure the following options:

- **Voice Auto Answer** specifies that voice interactions are automatically answered when an agent receives the invitation.
- **Voice Auto Answer On Active Call** specifies that a voice interaction is not automatically answered if the agent has another active call.
- **Voice Forward** enables agents to forward voice calls.
- **DTMF Tone** specifies to play a tone when the agent selects a DTMF key.

- **Voice Auto Ready** places the agent in the Ready state as soon as they log in.
- **Auto Not Ready Reason** places the agent in Not Ready state as soon as they log in. When you select this option, define the default Not Ready reason code.
- **Show Post Call Duration** allows the post-call duration for a voice call to be shown in the interaction bar's tool tip area.
- **Show Hold Indication Timer** allows the hold duration to be shown to the agent in the interaction bar.
- **Release Voice Calls while on Hold** enables agents to manually end voice calls that are on hold.
- **Redirect / Release calls in Hunt Group** enables agents to redirect/release calls in the Hunt group.
- **Conference** enables one-step voice conferencing.
- **Two-Step Voice Conference** enables two-step voice conferencing.
- **Clear Conference on Release** clears a conference call of all parties when the last agent leaves the conference call.
- **Cancel ACW on Done** specifies that the voice channel is to be returned to the agent's former availability status when the agent marks the voice interaction as Done while in After Call Work status.
- **Cancel ACW on Change** specifies that After Call Work status should be cancelled when an agent switches from After Call Work to Ready or Not Ready during a call.
- **Lookup Contact for Voice** activates Agent Workspace for contact lookup when an interaction is presented to an agent.
- **Create Contact for Voice** will create a contact if the initial contact lookup fails to find an existing contact.
- **Enable History Record Creation for voice interaction without contact** enables the voice interaction record to be created in My History and Interaction search screens when there is no associated contact.
- **Caller ID Anonymous** specifies whether the anonymous Caller ID is enabled. This option applies to only outbound calls.
- **Caller ID for Consultations** enables agents to use a Caller ID for consultations.
- **Caller ID for Conferences** enables agents to use a Caller ID for single-step conferences.
- **Caller ID for Transfers** enables agents to use a Caller ID for single-step transfers.
- **Routing Point for Skill Based Transfers** specifies the call number used by the Routing Point feature.
- **Routing Based Actions** defines the list of routing-based actions that an agent can perform. Options include:
  - **Make Call**
  - **One Step Conference**
  - **Initiate Conference**
  - **One step Transfer**
  - **Initiate Transfer**
- **Routing Based Targets** defines the list of contact types to which an agent can route calls through the actions defined in the **Routing Based Actions** option. Options include Agent, Routing Point, Type Destination, Outbound Record, and Contact. Note that Agent Groups and Skills targets are not listed here, because they are configured through a routing strategy.

- Options to exclude a range of numbers from the Team Communicator. For each option, if you enable it, you must type a regular expression (RegEx) to match against the following items:
  - Entered phone number
  - Selected agent or routing point

    For example, to exclude extensions that are 7 digits and start with the numbers 7 or 8, type `^(7|8)\d{6}$`
  - **Make Call Exclude Number** specifies the format for a phone number to exclude from Team Communicator when making a call.
  - **Single Step Conference Exclude Number** specifies the format for a phone number to exclude from Team Communicator on a single step conference.
  - **Single Step Transfer Exclude Number** specifies the format for a phone number to exclude from Team Communicator on a single-step transfer.
  - **Consultations Exclude Number** specifies the format for a phone number to exclude from Team Communicator on consultation.

    These options are not applicable to global or personal favorites.
- **Voice Ringtone Type** specifies the voice ringtone type. Enable the option and then select the ringtone type from the drop-down list box.
- **Voice Ringtone Priority** specifies the priority level of the voice ringtone within the channels. Enable the option and then type the number for the priority level.
- **Voice Ringtone Duration** specifies how long the ringtone rings. Use the following values:
  - `-1` specifies to repeat the ringtone until the call is answered.
  - `0` specifies to play the ringtone only one time.
  - `1` or more specifies to play the ringtone for the specified duration (in milliseconds) and then repeat until the call is answered.
- **Voice Ringtone Volume** specifies the default volume for the voice ringtone.
- **Can One Step Transfer** enables instant conferencing of a voice call.
- **Can Two Step Transfer** enables two-step transfer of a voice call.
- **Can Extend After Call Work** enables the extension of After Call Work indefinitely, until the user's status is changed manually.
- **Can Use WebRTC** enables WebRTC in Agent Workspace. Note: The **Can Use Voice Channel** option must be enabled.
- **WebRTC Polling Timeout** specifies, in seconds, a period of time during which WebRTC library waits for the response for a polling request.
- **Expression to capture groups in GWS url** specifies that Workspace should extract some part of its URL to capture some groups containing shared information among services, like the tenant or the region.
- **WebRTC Server URN** specifies the URN that should be based on the following pattern: `WEBRTCGATEWAY_SERVER:WEBRTCGATEWAY_PORT?sip-proxy-address=SIPPROXY_SERVER:SIPPROXY_PORT.`
- **Can change speaker volume** allows an agent to change speaker volume. Note: The **Can Use Voice Channel** option must be enabled.

- **Can mute microphone** allows an agent to mute and unmute the microphone. Note: The **Can Use Voice Channel** option must be enabled.
- **Can mute speaker** allows an agent to mute and unmute the speaker. Note: The **Can Use Voice Channel** option must be enabled.
- **Can send DTMF** specifies if DTMF is available for WebRTC calls. Note: The **Can Use Voice Channel** option must be enabled.
- **WebRTC Ringtone Type** specifies the voice channel ringing sound -
- **WebRTC Ringtone Priority** is the priority of ringtone. The higher the integer, the higher the priority.
- **WebRTC Ringtone Duration:**
  - -1 plays means to play and repeat until an explicit message stops it with an established event.
  - 0 means to play the entire sound one time.
  - An integer greater than 0 means a time in milliseconds to play and repeat the sound.
- **WebRTC Ringtone Volume** specifies the default volume for WebRTC ringtone.
- **Voice SIP Preview Ringtone Type** allows the selection of a ringtone type from the default tone types.
- **Voice SIP Preview Ringtone Priority** specifies Voice SIP ringing priority between different channels.
- **Voice SIP Preview Ringtone Duration** specifies the Voice SIP preview duration.
- **Voice SIP Preview Ringtone Volume** specifies the default volume for the voice SIP preview ringtone.
- **Allow Alert for Opt-out Recording** alerts the agent that the current call is not being recorded by a supervisor.

Contents

- 1 Metrics
- 2 Alerts

Find the metrics Interaction Server (IXN) exposes and the alerts defined for Interaction Server (IXN).

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Interaction Server (IXN)	PodMonitor	13131, 13133, 13139	option **ixnServer.ports.health** - default port 13131 - Endpoint: “/health/prometheus/all” option **ixnNode.ports.default** - default port 13133 - Endpoint: “/metrics” option **ixnVQNode.ports.health** - default port 13139 - Endpoint: “/metrics” **Note**: The above options are references to ports that match endpoints. Use these options to perform the associated query.	Default

Metrics

This table includes IXN Server metrics and IXN Node metrics. IXN Node metrics begin with the prefix *ixnnode*.

Note: There are more metrics than the ones listed in the table. However, only the metrics listed in the table are supported.

Metric and description	Metric details	Indicator of
ixn_health_info_current_clients Indicates the number of clients that are connected to IXN at the moment.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:** 5	Workload
ixn_health_info_current_routers Indicates the number of 'connected to' IXN routers.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:** 1	Workload, Operability
ixn_health_info_client_count { client_type_name="Agent application" }	**Unit:** Amount **Type:** Gauge **Label:** client_type_name. See the metric description for more details.	Workload

Metric and description	Metric details	Indicator of

Indicates the number of clients with specified type, connected to IXN.
 	
Label descriptions:
 	
• client_type_name - type of connected clients. Possible values are:
  • Unknown
  • Proxy
  • Agent application
  • Media server
  • Reporting engine
  • Routing engine
  • Universal router
  • Third party client.
	Sample value: 101	

ixn_health_info_router_total_submitted { router_name="URServer" } Indicates the total number of interactions that have been submitted to the router. **Label descriptions**: router_name - the name of the router into which the interactions have been submitted.	**Unit:** Amount **Type:** Counter **Label:** router_name. See the metric description for more details. **Sample value:** 33	Workload, Operability
ixn_health_info_router_strategy_load_count { router_name="URServer", strategy_name="AAAstarterStrategy", strategy_tenant="1" } Indicates the number of strategies with specified name loaded into a specified router. **Label descriptions**: • router_name - the name of the router into which the interactions are loaded • strategy_name - name of strategy • strategy_tenant - tenant number	**Unit:** Amount **Type:** Gauge **Label:** router_name, strategy_name, strategy_tenant. See the metric description for more details. **Sample value:** 1	Workload

Metric and description	Metric details	Indicator of
ixn_health_info_router_current_capacity { router_name="URServer" } Indicates the current capacity of a specified router - the number of interactions, not including those already submitted, that can be submitted into the router. **Label descriptions**: router_name - name of router.	**Unit:** Amount **Type:** Gauge **Label:** router_name. See the metric description for more details. **Sample value:** 987	Workload, Operability
ixn_health_info_router_currently_submitted { router_name="URServer" } Indicates the number of interactions that are in a specified router. **Label descriptions**: router_name - name of router.	**Unit:** Amount **Type:** Gauge **Label:** router_name. See the metric description for more details. **Sample value:** 13	Workload, Operability
ixn_health_info_current_strategies Indicates number of strategies which are associated with active submitters.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:** 11	Workload
ixn_health_info_router_max_submitted { router_name="URServer" } Indicates the maximum capacity of specified router - the number of interactions, that can be submitted into the router. **Label descriptions**: • router_name - name of router. *ixn_health_info_router_max_submitted = ixn_health_info_router_currently_submitted + ixn_health_info_router_current_capacity*	**Unit:** Amount **Type:** Gauge **Label:** router_name. See the metric description for more details. **Sample value:** 1000	Workload, Operability
ixn_health_info_current_database_requests Indicates the current database requests queue length.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:** 0	Workload, Operability
ixn_health_info_total_database_requests Indicates the number of processed database requests from IXN application start till current moment.	**Unit:** Amount **Type:** Counter **Label:** None **Sample value:** 75	Workload, Operability
ixn_health_info_current_database_connections Indicates the current number of DB connections.	**Unit:** Amount **Type:** Gauge **Label:** None	Workload, Operability

Metric and description	Metric details	Indicator of
	Sample value: 5	
ixn_health_info_total_database_deadlocks Indicates the total number of database queries that end up with a deadlock for all the time since IXN started.	**Unit:** Amount **Type:** Counter **Label:** None **Sample value:** 0	Workload, Operability
ixn_health_info_router_strategy_last_submitted_at { router_name="URServer", strategy_name="AAAStarterToAgent", strategy_tenant="1" } Indicates the Unix timestamp when last interaction has been submitted to router for specified strategy. **Label descriptions**: • router_name - name of router; • strategy_name - name of strategy; • strategy_tenant - tenant number.	**Unit:** Unix timestamp **Type:** Gauge **Label:** router_name, strategy_name, strategy_tenant. See the metric description for more details. **Sample value:** 1618322383	Workload, Operability
ixn_health_info_queue_media_waiting_processing { queue_name="AAAstarterQueue", queue_tenant="1", media_name="chat" } Indicates the current number of interactions with specified media type that are waiting for processing in a specified queue. **Label descriptions**: • queue_name - name of queue; • queue_tenant - tenant number; • media_name - media type. Note: This value is provided in Pulse as well.	**Unit:** Amount **Type:** Gauge **Label:** queue_name, queue_tenant, media_name. See the metric description for more details. **Sample value:** 0	Workload
ixn_health_info_agent_logged_in_count { agent_tenant="1" } Indicates the current number of logged in agents. **Label descriptions**:	**Unit:** Amount **Type:** Gauge **Label:** agent_tenant. See the metric description for more details. **Sample value:** 565	Workload

Metric and description	Metric details	Indicator of
agent_tenant - tenant number.		
ixn_health_info_queue_media_in_router { queue_name="toAgent", queue_tenant="1", media_name="chat" } Indicates the number of the interactions with specified media type from a specified queue being routed. **Label descriptions**: • queue_name - name of queue • queue_tenant - tenant number • media_name - media type	**Unit:** Amount **Type:** Gauge **Label:** queue_name, queue_tenant, media_name. See the metric description for more details. **Sample value:** 10	Workload, Operability
ixn_health_info_queue_media_on_agent { queue_name="toAgent", queue_tenant="1", media_name="chat" } Indicates the number of the interactions with specified media type from specified queue being handled by agents. **Label descriptions**: • queue_name - name of queue • queue_tenant - tenant number • media_name - media type	**Unit:** Amount **Type:** Gauge **Label:** queue_name, queue_tenant, media_name. See the metric description for more details. **Sample value:** 5	Workload, Operability
ixn_health_info_queue_media_current_length { queue_name="toAgent", queue_tenant="1", media_name="chat" } Indicates the number of interactions with specified media type that are waiting processing in specified queue and were never delivered to agent. **Label descriptions**: • queue_name - name of queue • queue_tenant - tenant number • media_name - media type **Note**: This value is provided in Pulse as well.	**Unit:** Amount **Type:** Gauge **Label:** queue_name, queue_tenant, media_name. See the metric description for more details. **Sample value:** 2	Workload, Operability
ixn_health_info_queue_media_in_processing	**Unit:** Amount	Workload, Operability

Metric and description	Metric details	Indicator of
{ queue_name="toAgent", queue_tenant="1", media_name="chat" } Indicates the sum of the interactions with specified media type from specified queue being routed by routers and being handled by agents. **Label descriptions**: • queue_name - name of queue • queue_tenant - tenant number • media_name - media type *ixn_health_info_queue_media_in_processing = ixn_health_info_queue_media_in_router + ixn_health_info_queue_media_on_agent* **Note**: This value is provided in Pulse as well.	**Type:** Gauge **Label:** queue_name, queue_tenant, media_name. See the metric description for more details. **Sample value:** 15	
ixn_health_info_router_strategy_currently_submitted { router_name="URServer", strategy_name="AAAStarterToAgent", strategy_tenant="1" } Indicates the number of interactions which are submitted to specified router by specified strategy at the moment. **Label descriptions**: • router_name - name of router • strategy_name - name of strategy • strategy_tenant - tenant number	**Unit:** Amount **Type:** Gauge **Label:** router_name, strategy_name, strategy_tenant. See the metric description for more details. **Sample value:** 3	Workload, Operability
ixn_health_info_router_strategy_current_capacity { router_name="URServer", strategy_name="AAAStarterToAgent", strategy_tenant="1" } Indicates the number of interactions that can be submitted more to specified router by specified strategy. **Label descriptions**: • router_name - name of router • strategy_name - name of strategy • strategy_tenant - tenant number	**Unit:** Amount **Type:** Gauge **Label:** router_name, strategy_name, strategy_tenant. See the metric description for more details. **Sample value:** 197	Workload, Operability

Metric and description	Metric details	Indicator of
ixn_health_info_router_strategy_total_submitted { router_name="URServer", strategy_name="AAAStarterToAgent", strategy_tenant="1" } Indicates the number of interactions that were submitted to specified router by specified strategy since IXN app start till now. **Label descriptions**: • router_name - name of router • strategy_name - name of strategy • strategy_tenant - tenant number	**Unit:** Amount **Type:** Counter **Label:** router_name, strategy_name, strategy_tenant. See the metric description for more details. **Sample value:** 9	Workload, Operability
ixnnode_interactions_pulled_total Indicates the total number of the interactions pulled for the specific strategy. **Label descriptions**: strategy - The name of the strategy for which interactions are pulled.	**Unit:** Amount **Type:** Counter **Label:** strategy. See the metric description for more details. **Sample value:**	Workload, Operability
ixnnode_routing_sessions_current Indicates the current number of the routing sessions in routing.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:**	Workload
ixnnode_all_instructions_received_total Indicates the total number of instructions (of any type) received from ORS service.	**Unit:** Amount **Type:** Counter **Label:** None **Sample value:**	Workload, Operability
ixnnode_routing_instructions_received_total Indicates the total number of received routing instructions. **Label descriptions**: • strategy - The name of the strategy for which routing instructions is received. • type - The type of the instruction. It takes values "terminal" and "non-terminal". Terminal instructions are RequestDeliver, RequestPlaceInQueue,	**Unit:** Amount **Type:** Counter **Label:** strategy, type. See the metric description for more details. **Sample value:**	Workload, Operability

Metric and description	Metric details	Indicator of
RequestPlaceInWorkbin, RequestStopProcessing.		
ixnnode_redis_client_status Indicates the status of Redis client. **Label descriptions**: redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer". Value: 0 - Not Ready 1 - Ready	**Unit:** Status **Type:** Gauge **Label:** redis_client. See the metric description for more details. er". **Sample value:**	Operability
ixnnode_redis_client_errors_total Indicates the total number of errors occurred on Redis client. **Label descriptions**: redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer".	**Unit:** Amount **Type:** Counter **Label:** redis_client. See the metric description for more details. **Sample value:**	Error
ixnnode_redis_client_node_status Indicates the status of connection to individual nodes of Redis server (in singleton mode matches to ixnnode_redis_client_status). **Label descriptions**: • redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer". • node - The node of Redis server for which the metric is present as "host:port". Value: 0 - Ready 1 - Not Ready 2 - Wait (so far there have been no connection attempts)	**Unit:** Status **Type:** Gauge **Label:** redis_client, node. See the metric description for more details. **Sample value:**	Operability
ixnnode_redis_client_node_errors_total Indicates the total number of errors occurred on individual nodes of Redis client (in singleton mode matches to	**Unit:** Amount **Type:** Counter **Label:** redis_client, node. See the metric description for more details.	Error

Metric and description	Metric details	Indicator of
ixnnode_redis_client_errors_total). **Label descriptions**: • redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer". • node - The node of Redis server for which the metric is present as "host:port".	**Sample value:**	
ixnnode_redis_commands_completed_total Indicates the total number of successfully completed redis commands. **Label descriptions**: • redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer". • command - The Redis command for which the metric is present. It takes values "xadd", "xread", "xdel".	**Unit:** Amount **Type:** Counter **Label:** redis_client, command. See the metric description for more details. **Sample value:**	Workload, Operability
ixnnode_redis_commands_failed_total Indicates the total number of failed redis commands. **Label desriptions**: • redis_client - The Redis client instance for which the metric is present. It takes values "reader" and "writer". • command - The Redis command for which the metric is present. It takes values "xadd", "xread", "xdel".	**Unit:** Amount **Type:** Counter **Label:** redis_client, command. See the metric description for more details. **Sample value:**	Error
ixnnode_rq_client_status Indicates the status of connection to RQ Service nodes. **Label descriptions**: rq_node - RQ Service node for which the metric is present.	**Unit:** Status **Type:** Gauge **Label:** rq_node. See the metric description for more details. **Sample value:**	Operability

Metric and description	Metric details	Indicator of
ixnnode_rq_requests_failed_total Indicates the total number of failed requests to RQ Service. **Label descriptions**: type - The type of the failed requests. It takes values "isp_event" - interaction protocol evnts and "ixn_ping" - health check messages.	**Unit:** Amount **Type:** Counter **Label:** type. See the metric description for more details. **Sample value:**	Error
ixnnode_instructions_processing_queue_concurrency_limit Indicates the maximum number of routing instructions that can be processed in parallel.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:**	n/a
ixnnode_instructions_processing_queue_size Indicates the number of instructions received from ORS currently being processed. **Label descriptions**: type - The type of the instruction. It takes values "isp_request" - routing instruction and "ixn_ping" - reply to health check message.	**Unit:** Amount **Type:** Gauge **Label:** type. See the metric description for more details. **Sample value:**	Workload, Operability
ixnnode_pull_request_total Indicates the total number of RequestPull requests successfully completed by InteractionServer. **Label descriptions**: strategy - The strategy for which interactions are pulled.	**Unit:** Amount **Type:** Counter **Label:** strategy. See the metric description for more details. **Sample value:**	Workload, Operability
ixnnode_route_request_sent_total Indicates the total number of route requests successfully sent to ORS. **Label descriptions**: strategy - The strategy to which requests are sent.	**Unit:** Amount **Type:** Counter **Label:** strategy. See the metric description for more details. **Sample value:**	Workload, Operability
ixnnode_route_request_failed_total Indicates the total number of route requests failed to send to ORS. **Label descriptions**: strategy - The strategy to which requests are sent.	**Unit:** Amount **Type:** Counter **Label:** strategy. See the metric description for more details. **Sample value:**	Error
ixnnode_instructions_processed_by_ixn_current	**Unit:** Amount	Workload

Metric and description	Metric details	Indicator of
Indicates the number of routing instructions currently being processed by IXN Server.	**Type:** Gauge **Label:** None **Sample value:**	
ixnnode_interactions_placed_back_total Indicates the total number of times an interaction was placed back in queue. **Label descriptions**: • reason - The reason of placing back in queue. It takes values: "StrategyOldQueueRequest" - Strategy explicitly requested to place to valid queue with name (not "BACK") matching the name of queue interaction was pulled from. Set by ORS. "StrategyBackRequest" - Strategy requested placing interaction "BACK" explicitly. Set by ORS. "Implicit" - Strategy did nothing and ORS places interaction back cause there are no instructions for it. Set by ORS. "Error" - ORS places interaction back into queue due to some error regardless of the error source be it strategy itself or any other reason. Set by ORS. "SubmitError" - IXN Node failed to send interaction to ORS and places it back into queue. "Unknown" - The reason was not specified by ORS. • strategy - The strategy which routed interactions.	**Unit:** Amount **Type:** Counter **Label:** reason, strategy. See the metric description for more details. **Sample value:**	Workload, Error, ORS Error
ixnnode_running_strategies_current Indicates the number of the strategies for which interactions currently are being pulled.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:**	Operability
ixnnode_configured_strategies_current Indicates the number of the strategies read from configuration for which interactions should be pulled.	**Unit:** Amount **Type:** Gauge **Label:** None **Sample value:**	Operability

Metric and description	Metric details	Indicator of
ixnnode_configuration_fetch_errors_total Indicates the total number of error occurred while fetching configuration from Configuration Service.	**Unit:** Count **Type:** Counter **Label:** None **Sample value:**	Error
ixnnode_last_fetched_configuration_timestamp Indicates the last time the configuration was successfully fetched from Configuration Service as the number of seconds since January 1 1970 UTC.	**Unit:** Timestamp **Type:** Gauge **Label:** None **Sample value:**	Operability

Alerts

No alerts are defined for Interaction Server.