



Developer Guide

Amazon Connect



Amazon Connect: Developer Guide

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What is the Amazon Connect Customer Developer Guide?	1
Who this guide is for	1
What's in this guide	1
How to use this guide	2
Programming with the Connect Customer APIs	3
Best practices	3
Types of errors	3
Throttling in Connect Customer APIs	4
How to configure your client Read API(s)	4
How to make 2 TPS work for List APIs	6
How to make 2 TPS work for Create/Update APIs	7
Hitting a resource quota? Delete resources	7
How to request an increase to an API throttling quota	7
Supported SDKs	8
Working with Attachments	9
Uploading Attachments with the Participant Service	9
Uploading Attached Files with the Connect Service	11
Best practices for using PutDialRequestBatch	11
Recommendations for tracking contact dispositions	12
Eventual consistency	14
Resource integrations	14
CloudFormation	14
CloudTrail	15
EventBridge	15
Connect Customer actions by resource	16
Analytics data lake actions	17
Agent status actions	17
Chat actions	18
Contacts actions	18
Data table actions	19
Email actions	20
Evaluation actions	21
Files actions	22
Flows / Flow module actions	22

Hierarchy group actions	23
Hours of operation actions	24
Instance actions	25
Integration association actions	26
Metrics actions	26
Notifications actions	27
Phone number actions	27
Predefined attributes actions	28
Prompt actions	28
Queue actions	29
Quick connect actions	30
Routing profile actions	30
Rules actions	31
Security profile actions	32
Tag actions	32
Task actions	32
Traffic distribution group actions	33
Use cases actions	34
User management actions	34
View actions	35
Vocabulary actions	36
Voice actions	36
Workspace actions	36
Flow language	38
Concepts	38
Contact	38
Participant	38
Action types	38
Example flow	39
Actions	41
Identifier	41
Type	41
Parameters	41
Transitions	42
List of Operators	43
Parameter restrictions	45

Contact actions	46
Flow control actions	75
Interactions	88
Participant actions	105
Rules Function language	117
Concepts	117
Example rule function	118
Conditions	119
OnMetricDataUpdate	119
OnContactEvaluationSubmit	152
OnPostCallAnalysisAvailable	160
OnRealTimeCallAnalysisAvailable	175
OnPostChatAnalysisAvailable	178
OnEmailAnalysisAvailable	191
OnZendeskTicketCreate	194
OnZendeskTicketStatusUpdate	198
OnSalesforceCaseCreate	201
OnCaseCreate	205
OnCaseUpdate	210
OnSlaBreach	216
PatternMatch Operands	216
Testing language	218
Concepts	218
Observations	218
Events	218
Actions	218
Actors	218
Example test	219
Observations	222
Parameters	222
Observation object	222
Events	223
Test initiated	223
Test completed	224
Message received	224
Flow action started	226

Actions	229
Assertion	229
Send Instruction	230
Override system behavior	233
Test Control Actions	245
Amazon CX Designer SDK	248
Design Philosophy	248
Getting Started	248
Prerequisites	248
Authentication	249
Step 1: Create a Programmatic User	249
Step 2: Generate an API Key	250
Step 3: Install the SDK	250
Step 4: Make your First Call	250
Permissions	251
Next Steps	251
API Reference	251
Analytics Tags	258
API Tokens	264
Applications	267
Application Builds	284
Application Deployments	291
Context Variables	301
Conversations	307
Data Requests	317
Downloads	329
Flows	330
Guardrails	345
Knowledge Bases	364
Knowledge Base Articles	378
KnowledgeBase Documents	387
Live Sync Scripts	393
Logs	415
Modalities	423
Programmatic Users	434
Roles	442

Secrets	454
Slot Types	461
Team	471
Trails	472
Users	482
Versions	492
Workspaces	496
Common Types	502
Concepts	505
Workspace	506
Application	506
Flow	506
Build	506
Deployment	506
Secret	506
Context Variable	506
Slot Type	506
Data Request	507
Knowledge Base	507
Guardrail	507
Evaluation	507
Simulation	507
Scenario	507
Modality	507
Analytics Tag	507
Trail	508
Programmatic User	508
Role	508
Common Errors	508
Common Errors	508
Error Response Shape	509
A2A protocol contract	510
Prerequisites	511
Transport and discovery	511
Event conventions	511
Channels	511

Session initiation (handoff and immediate handoff)	512
Channel readiness (CHANNEL_STATE)	512
Finishing: hand-back and terminal outcomes	513
Errors	513
Interruptions (barge-in)	514
DTMF input	514
Silence input	515
Transport fragmentation (WebSocket only)	515
Tracing	515
Note on A2A spec compliance	516
Sample events	516
A.1 INIT_SESSION (Connect to your agent)	517
A.2 INIT_SESSION_RESPONSE (your agent to Connect)	518
A.2b CHANNEL_STATE READY (Connect to your agent, orchestrated/tool handoff)	519
A.3 Text turn input (Connect to your agent)	520
A.3b Audio input chunk (Connect to your agent, voice, audio enabled 3P agent)	520
A.4 Text response chunk (your agent to Connect)	521
A.4b Audio response chunk (your agent to Connect, voice, audio enabled 3P agent)	522
A.5 Terminal status update with FINISH (your agent to Connect)	523
A.6 Task failure with structured error (your agent to Connect)	524
A.6b Informational ERROR message (Connect to your agent)	524
A.7 INTERRUPTION (either direction)	525
A.8 DTMF_INPUT_COMPLETE (Connect to your agent)	527
A.9 FRAGMENT (WebSocket only, either direction)	527
A.10 TRACING_SPAN (your agent to Connect)	529
A.10b TRACING_SPAN with per-segment latency (multiple spans in one trace)	531
Related topics	535
Touchpoint	536
What Touchpoint provides	536
Prerequisites	537
Setting up Amazon Connect Customer	538
Adding Live Sync (optional)	538
Examples	538
Chat	539
Voice mini	539
Live Sync	539

External (no UI)	541
Interactive playground	541
API reference	542
License	542
Document history	543

What is the Amazon Connect Customer Developer Guide?

Note

Amazon Connect now refers to a portfolio of agentic solutions for business functions. The legacy product is now called Amazon Connect Customer, or simply Customer. The legacy name is used interchangeably in this documentation.

The Amazon Connect Developer Guide is for software developers who build applications that integrate with Amazon Connect using its APIs, flow language, and event-driven architecture.

Who this guide is for

This guide is for developers who:

- Build custom contact center solutions using the Amazon Connect APIs
- Create and manage contact flows programmatically using the Connect flow language
- Automate rules and business logic using the rules function language
- Test contact flows using the testing language
- Integrate Amazon Connect with other AWS services and third-party applications

If you are an administrator setting up and managing Amazon Connect through the console, see the [Amazon Connect Administrator Guide](#).

What's in this guide

Programming with the Amazon Connect Customer APIs

Best practices for calling the APIs, including throttling guidance, SDK support, and eventual consistency behavior.

Actions by resource

A navigation reference that organizes API operations by the resource they act on.

Connect flow language

The JSON-based language for defining contact flows programmatically, including action types, parameters, transitions, and conditions.

Rules function language

The syntax for defining rule conditions and event triggers that automate actions in your contact center.

Testing language

The schema for writing automated tests against your contact flows, including assertions, overrides, and simulated events.

How to use this guide

To understand best practices for integrating with the APIs, start with [Programming with the Connect Customer APIs](#). To build or modify flows programmatically, refer to [Connect Customer Flow language](#) for the JSON schema and action reference. To validate your flows before deploying them to production, use [Connect Customer testing language](#).

For the complete API reference with request/response syntax and data types, see the [Amazon Connect API Reference](#).

Programming with the Connect Customer APIs

The following sections provide additional information about using the Connect Customer service APIs.

Contents

- [Best practices for using Connect Customer APIs](#)
- [Working with Attachments](#)
- [Best practices for using PutDialRequestBatch for outbound campaign calling](#)
- [Connect Customer eventual consistency](#)
- [Resource integrations](#)

Best practices for using Connect Customer APIs

This topic provides guidance for using Connect Customer Describe and List APIs so you don't get unexpected 4xx errors for the response. It also explains how to configure your client Read APIs.

Contents

- [Types of errors](#)
- [Throttling in Connect Customer APIs](#)
- [How to configure your client Read API\(s\)](#)
- [How to make 2 TPS work for List APIs when you have a large number of resources](#)
- [How to make 2 TPS work for Create/Update APIs when you have a large number of resources](#)
- [Hitting a resource quota? Delete unused / stale resources](#)
- [How to request an increase to an API throttling quota](#)
- [Supported SDKs for all Connect Customer APIs](#)

Types of errors

The Connect Customer APIs provide an HTTP interface. HTTP defines ranges of HTTP Status Codes for different types of error responses.

- Client errors are indicated by HTTP Status Code class of 4xx

- Service errors are indicated by HTTP Status Code class of 5xx

In this reference guide, the documentation for each API has an **Errors** section that includes a brief discussion about HTTP status codes. We recommend looking there as part of your investigation when you get an error.

For information about the common errors returned by Connect Customer public APIs, see [Common Errors](#).

Throttling in Connect Customer APIs

Throttling errors in Connect Customer public API(s) are defined by HTTP status code 429. This HTTP status code can be retried by the client based on their requirement.

Important

The throttling limits are defined for each API separately at the AWS account level, not for the individual Connect Customer instance.

To use any API for Connect Customer resources (such as users, queues, and routing profiles), you need the [ID/ARN for the Connect Customer instance](#).

By default, Connect Customer limits the steady-state requests per second (RPS) across all APIs within an AWS account, per Region. It also limits the burst (that is, the maximum bucket size) across all APIs within an AWS account, per Region.

In Connect Customer the burst limit represents the target maximum number of concurrent request submissions that APIs will fulfill before returning *429 Too Many Requests* error responses.

For more information about throttling quotas, see [Connect Customer throttling quotas](#).

How to configure your client Read API(s)

Your client configuration will vary based on number of resources that your API tries to describe/list per second.

In the following Java example, the number of retries is set to 3. This means after your Connect Customer client implementation experiences throttling, it retries for maximum of 3 times. Instead

of retrying immediately and aggressively, the following snippet waits a specified amount of time (between 0 to max of 5 seconds as defined by `maxBackoffTime` parameter) between tries and uses [EqualJitterBackoffStrategy](#).

```
final class ClientBuilder {

    private static final int NUMBER_OF_RETRIES = 3;

    private static final RetryPolicy RETRY_POLICY =
        RetryPolicy.builder()
            .numRetries(NUMBER_OF_RETRIES)
            .retryCondition(RetryCondition.defaultRetryCondition())
            .backoffStrategy(EqualJitterBackoffStrategy.builder()
                .baseDelay(Duration.ofSeconds(1))
                .maxBackoffTime(Duration.ofSeconds(5))
                .build())
            .build();

    public static ConnectClient getClient() {
        return ConnectClient.builder()
            .httpClient(LambdaWrapper.HTTP_CLIENT)

        .overrideConfiguration(ClientOverrideConfiguration.builder().retryPolicy(RETRY_POLICY).build())
            .build();
    }
}
```

When failures are caused by overload or contention, backing off often doesn't help as much as it seems like it should. This is because there's a correlation between failures and backing off/contention:

- If all the failed calls back off to the same time, they cause contention or overload again when they are retried.

To address this, we recommend adding jitter. Jitter adds some amount of randomness to the backoff which spreads the retries around in time. For more information about how much jitter to add and the best ways to add it, see this AWS blog post: [Exponential Backoff and Jitter](#).

For information about types of backoff strategies, see [Interface BackoffStrategy](#).

How to make 2 TPS work for List APIs when you have a large number of resources

There are two options: use List APIs with `maxResults = 1,000`, or use Search APIs as an alternative to List/Describe round trips. Both options are discussed here.

The List API of a particular Connect Customer resource supports a `maxResults` parameter as part of request body. List API(s) support a maximum of 1,000 results in single API call unless specified otherwise in the documentation.

The following example shows the `maxResults` of the [ListUsers](#) API.

```
String nextToken = null;
do {
    ListUsersRequest listUsersRequest = ListUsersRequest.builder()
        .instanceId(your Connect Customer instanceId)
        .maxResults(1000)
        .nextToken(nextToken)
        .build();
    ListUsersResponse response = client.listUsers(listUsersRequest);
    nextToken = response.nextToken();
    System.out.println(response.sdkHttpResponse().statusCode());
} while (nextToken != null);
```

If `nextToken` is returned, then more results are available. The value of `nextToken` is a unique pagination token for each page. Make the call again using the returned token to retrieve the next page. Keep all other arguments unchanged. Each pagination token expires after 24 hours. Using an expired pagination token will return an HTTP 400 `InvalidToken` error.

When to use Search APIs instead of List APIs

We recommend you assess the speed of pulling details for 100 records at a time (the Search API limit) instead of pulling 1,000 IDs and doing Describe round trips. It's better to try using Search APIs instead of combination of List and Describe API for a specific resource.

Let's say you have a situation where you're listing specific resources in your Connect Customer instance and then call a Describe API on an individual resource. Instead, we recommend leveraging the Search API for that corresponding resource. Search APIs support several filters that can help to reduce response set as per requirement.

How to make 2 TPS work for Create/Update APIs when you have a large number of resources

There is a performance impact behind creating/updating resources at a default 2 TPS. For example, 100 resources can be created/updated with 2 TPS within 50 seconds. A 1,000 resources with this TPS would need nearly 8 minutes. Based on your use case, if the operation is impacting performance, contact Support and provide a business justification for your request to increase your throttling quota. See [How to request an increase to an API throttling quota](#).

It is your responsibility to always implement the following best practices:

- Check your logic and implement best practices to make your requests as efficient as possible. Check out [AWS Well-Architected Tool](#) (AWS WA Tool) for processes that help measure your architecture using AWS best practices.
- Test your requests and any custom processes *before adding them to production operations*.

Hitting a resource quota? Delete unused / stale resources

If you keep hitting the quota limit for a specific resource, we recommend deleting any unused or stale resources. You can find the Delete API for a resource on the resource-specific [Action pages](#). These pages list all the APIs for a given resource.

How to request an increase to an API throttling quota

Important

- We analyze all requests for quota increases and provide guidance for all queries.
- We rarely approve requests if they apply to situations other than those listed below.
- For smaller increase requests, we can approve in hours. Larger increase requests take time to review, process, approve, and deploy. Depending on your specific implementation, your resource, and the size of quota that you want, a request can take up to 3 weeks. An extra-large worldwide increase can potentially take months. If you're increasing your quotas as part of a larger project, keep this information in mind and plan accordingly.

For instructions about how to use the [Service Quotas console](#), see [Using the AWS Management Console to request an increase](#).

In the Services Quotas console, open an Support case and provide the following information:

1. Have you implemented the best practices explained in the [Retry behavior](#) topic of the *AWS SDKs and Tools Reference Guide*?
2. What is the performance impact without the requested limit increase? Provide some calculations.
3. What is the expected number of resources customer is trying to create/update/describe every second with the APIs?
4. What is the new quota for the API that you want?

Include in your case if the following situation(s) apply:

- It is a migration request and you need high TPS for a specific time range to configure your instance(s).
- There are performance or business impacting usecases, such as handling huge call volume for peak season.
- You have thousands of resources with multiple concurrent agents working at the same time which might increase the overall traffic from your contact center.

Supported SDKs for all Connect Customer APIs

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Working with Attachments

This topic explains how to work with attachments for Chat and Cases in Connect Customer. For Chat Attachments, refer to the first section on the Connect Customer Participant Service API. For Case Attachments, refer to the second section on the Connect Customer service's Attached Files APIs.

Uploading Attachments with the Participant Service

There are three basic steps for uploading a file using the Connect Customer Participant Service API.

1. HTTP POST file metadata to `StartAttachmentUpload` API, which will provide a signed Amazon S3 URL and attachment ID for uploading the file directly to Amazon S3.
2. HTTP PUT file data to the signed Amazon S3 URL.
3. HTTP POST attachment ID to `CompleteAttachmentUpload` to finalize the upload to Amazon S3.

Below is a basic JavaScript implementation for reference purposes.

```
//Define the html element for file using input tag

<input type="file" id="fileUpload">
<input type="button" id="btnUploadFile" onclick="uploadFile()" value="Upload file">

async function uploadFile() {
    //Initiate the file upload by calling StartAttachmentUpload, providing the name,
    size, and content type of the file you want to upload.
    //The response will include a pre-signed URL and headers to use when building the
    file upload request, as well as an AttachmentId.
    const files = document.getElementById('fileUpload').files;
    const file = files[0];

    const startUploadRequest = {
        AttachmentName: file.name,
        AttachmentSizeInBytes: file.size,
        ContentType: file.type
    };

    const { AttachmentId, UploadMetadata } = await
    startAttachmentUpload(startUploadRequest);
```

```
//Send the file data to the pre-signed S3 URL.
//The file is stored in a temporary location in the S3 bucket.

await uploadFileToS3(file, UploadMetadata.Url, UploadMetadata.HeadersToInclude);

//Finalize the file upload by calling CompleteAttachmentUpload, providing the
AttachmentId.
//This moves the file to the final Attachments S3 path configured for the connect
instance.
const completeUploadRequest = {
  AttachmentIds: [ AttachmentId ]
};
await completeAttachmentUpload(completeUploadRequest);
}

async function startAttachmentUpload(requestData){
  const response = await fetch(StartAttachmentUploadEndpoint, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
      'X-Amz-Bearer': ConnectionToken,
    },
    body: JSON.stringify(requestData)
  });

  return await response.json();
}

async function uploadFileToS3(file, signedUrl, headersToInclude) {
  return fetch(signedUrl, {
    method: 'PUT',
    headers: headersToInclude,
    body: file
  });
}

async function completeAttachmentUpload(requestData){
  return fetch(CompleteAttachmentUploadEndpoint, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
      'X-Amz-Bearer': ConnectionToken,
    },
  },
```

```
        body: JSON.stringify(requestData)
    });
}
```

For an example of a real world implementation, view the [Connect Customer ChatJS JavaScript client](#) on GitHub.

Uploading Attached Files with the Connect Service

The same basic steps follow from the Participant service instructions, for usage with the Connect Customer service's attached file APIs.

1. Call the `StartAttachedFileUpload` API with the required attached file information, which will provide a signed Amazon S3 URL, headers, and file ID for uploading the file directly to Amazon S3.
 - If you are attaching a file to a Case, the calling identity must also have the `cases:CreateRelatedItem` permission on the target Case resource.
2. Using your http client of choice, PUT file data to the signed Amazon S3 URL, ensuring that you add the headers as required from the response of Step 1.
3. Call the `CompleteAttachedFileUpload` to finalize the upload to Amazon S3.

Best practices for using `PutDialRequestBatch` for outbound campaign calling

This topic provides guidance and best practices for using the [PutDialRequestBatch](#) API for outbound campaign calling. `PutDialRequestBatch` takes in a list of [DialRequest](#) to be dialed as part of an outbound campaign.

- Before using [PutDialRequestBatch](#), always use the [StartCampaign](#) API to start the outbound campaign.
- In [DialRequest](#), set `expirationTime`—the timestamp when a dial request expires—to a future time, preferably a few minutes in the future to allow the outbound campaigns dialer algorithm to attempt to dial.

Note the following failure codes:

- **InvalidInput:** There's a request input mismatch with the parameters defined in [DialRequest](#). Fix the request input and try again.
- **RequestThrottled:** There aren't enough agents or telecom capacity available. The service maintains an upper limit on the number of records it can accept. When this limit is reached, the service returns a **RequestThrottled** message. This is a signal for you to stop sending more records and instead wait before attempting to retry, allowing the service to process and clear some records so it has room to accept more.
- **UnknownError:** There is an internal failure in outbound campaigns. Please try again.

Recommendations for tracking contact dispositions

A contact disposition is a type of state for a contact. It is either assigned by an agent from a pre-defined list that you provide (for example, **Sold**, **Wrong person**, **Right person**, **didn't sell**) or assigned by Connect Customer for a contact (for example, **EXPIRED**, **CONTACT_FLOW_DISCONNECT**, or **TELECOM_PROBLEM**). For outbound campaigns, the contact disposition is assigned by Connect Customer.

You can use Connect Customer contact records or contact events to track the contact dispositions for outbound campaigns. The key attribute to track contact disposition is **DisconnectReason**, which indicates how the voice contact was terminated.

Disconnect reasons can be grouped into the following 3 categories:

- **Success:** A contact is successfully dialed out. Values: **CUSTOMER_DISCONNECT** | **AGENT_DISCONNECT** | **THIRD_PARTY_DISCONNECT** | **BARGED** | **CONTACT_FLOW_DISCONNECT** | **OTHER**
- **Expired:** This is expected behavior of the dialing algorithm. When the dialing algorithm determines insufficient dialing capacity (for example, because all agents are occupied or all telecom capacity is being used) during the duration set by the **expirationTime** parameter, Connect Customer does not attempt to call such endpoints. In other words, those endpoints are expired. If you need higher throughput, you can submit a request for higher dialing capacity. Values: **EXPIRED**
- **Failed:** Outbound campaigns failed a dial attempt because there are specific systematic errors. Values: **OUTBOUND_DESTINATION_ENDPOINT_ERROR** | **OUTBOUND_RESOURCE_ERROR** | **OUTBOUND_ATTEMPT_FAILED** | **TELECOM_PROBLEM**

For a description of each of these values, see `DisconnectReason` in the [Contact records data model](#) topic in the *Connect Customer Administrator Guide*.

If the disconnect reason is expired or failed type, take one of the following next steps:

Disconnect reason value	Recommendation
EXPIRED	We recommend that you use PutDialRequestBatch to create a new contact for this destination with a different client token.
OUTBOUND_DESTINATION_ENDPOINT_ERROR	Check whether the current configuration of your Connect Customer instance allows this destination to be dialed. For more information, see Countries you can call by default .
OUTBOUND_RESOURCE_ERROR	- Check whether your Connect Customer instance allows outbound calls. For more information, see Enable outbound calls. - Check that the following resources exist: queue, flow, Connect Customer source phone number, and campaign. - After fixing the configuration of the instance, use PutDialRequestBatch to create a new contact for this destination with a different client token.
OUTBOUND_ATTEMPT_FAILED	There is an unidentified error. We recommend that you use PutDialRequestBatch to create a new contact for this destination with a different client token.
TELECOM_PROBLEM	Disconnected due to an issue with connecting the call from the carrier, network congestion, network error, etc. We recommend that you use PutDialRequestBatch to create a new contact for this destination with a different client token.

Connect Customer eventual consistency

The Connect Customer API follows an [eventual consistency](#) model due to the distributed nature of the system. As a result, changes to Connect Customer resources might not be immediately visible to the subsequent commands you run.

When you perform Connect Customer API calls, there might be a brief delay before the change is available throughout Connect Customer. It typically takes less than a few seconds for the change to propagate throughout the system, but in some cases it can take several minutes. You might get unexpected errors, such as a `ResourceNotFoundException` or an `InvalidRequestException`, during this time. For example, Connect Customer might return a `ResourceNotFoundException` if you call [DescribeUser](#) immediately after calling [CreateUser](#).

We recommend that you configure a retry strategy on your Connect Customer clients to automatically retry operations after a brief waiting period. For more information, see [Retry behavior](#) in the *AWS SDKs and Tools Reference Guide*.

Resource integrations

CloudFormation

Connect Customer is integrated with [CloudFormation](#), a service that allows you to treat infrastructure as code. Use CloudFormation to model, provision, and manage AWS and third-party resources.

The following Connect Customer resource APIs support CloudFormation templates:

- [AWS::Connect::ApprovedOrigin](#)
- [AWS::Connect::ContactFlow](#)
- [AWS::Connect::ContactFlowModule](#)
- [AWS::Connect::EvaluationForm](#)
- [AWS::Connect::HoursOfOperation](#)
- [AWS::Connect::Instance](#)
- [AWS::Connect::InstanceStorageConfig](#)
- [AWS::Connect::IntegrationAssociation](#)
- [AWS::Connect::PhoneNumber](#)

- [AWS::Connect::Prompt](#)
- [AWS::Connect::Queue](#)
- [AWS::Connect::QuickConnect](#)
- [AWS::Connect::RoutingProfile](#)
- [AWS::Connect::Rule](#)
- [AWS::Connect::SecurityKey](#)
- [AWS::Connect::TaskTemplate](#)
- [AWS::Connect::TrafficDistributionGroup](#)
- [AWS::Connect::User](#)
- [AWS::Connect::UserHierarchyGroup](#)
- [AWS::Connect::View](#)
- [AWS::Connect::ViewVersion](#)

CloudTrail

Connect Customer is integrated with AWS CloudTrail, a service that provides a record of the Connect Customer API calls that a user, role, or AWS service makes. CloudTrail captures Connect Customer API calls as events. All public Connect Customer APIs support CloudTrail.

For more information, see [Logging Connect Customer API calls with AWS CloudTrail](#).

EventBridge

Connect Customer is integrated with Amazon EventBridge, a service that provides a record of the Connect Customer API calls that a user, role, or AWS service makes. All public Connect Customer APIs support EventBridge, with events published to CloudTrail consumable in EventBridge.

Some Connect Customer resources are integrated directly into EventBridge. For more information, see [EventBridge events emitted by Connect Customer](#).

Connect Customer actions by resource

The following section lists the Connect Customer API actions by resource.

- [Agent status actions](#)
- [Analytics data lake actions](#)
- [Chat actions](#)
- [Contacts actions](#)
- [Data table actions](#)
- [Email actions](#)
- [Evaluation actions](#)
- [Files actions](#)
- [Flows / Flow module actions](#)
- [Hierarchy group actions](#)
- [Hours of operation actions](#)
- [Instance actions](#)
- [Integration association actions](#)
- [Metrics actions](#)
- [Notifications actions](#)
- [Phone number actions](#)
- [Predefined attributes actions](#)
- [Prompt actions](#)
- [Queue actions](#)
- [Quick connect actions](#)
- [Routing profile actions](#)
- [Rules actions](#)
- [Security profile actions](#)
- [Tag actions](#)
- [Task actions](#)
- [Traffic distribution group actions](#)

- [Use cases actions](#)
- [User management actions](#)
- [View actions](#)
- [Vocabulary actions](#)
- [Voice actions](#)
- [Workspace actions](#)

Analytics data lake actions

These APIs are in preview release for Connect Customer and are subject to change.

The following API actions are available for analytics datasets:

- [AssociateAnalyticsDataSet](#)
- [BatchAssociateAnalyticsDataSet](#)
- [BatchDisassociateAnalyticsDataSet](#)
- [DisassociateAnalyticsDataSet](#)
- [ListAnalyticsDataAssociations](#)

Agent status actions

The following API actions are available for agent status:

- [CreateAgentStatus](#)
- [DescribeAgentStatus](#)
- [ListAgentStatuses](#)
- [PutUserStatus](#)
- [SearchAgentStatuses](#)
- [UpdateAgentStatus](#)

Note

Agent status ARNs cannot be deleted but can be renamed or disabled if needed.

For more information about agent status, see [About agent status](#) in the *Connect Customer Administrator Guide*.

Chat actions

The following API actions are available for chats:

- [CreatePersistentContactAssociation](#)
- [SendChatIntegrationEvent](#)
- [StartChatContact](#)

For more information about chat, see [Concepts: Web and mobile chat capabilities in Connect Customer](#) in the *Connect Customer Administrator Guide*.

Contacts actions

The following API actions are available for contacts:

- [AssociateContactWithUser](#)
- [BatchPutContact](#)
- [CreateContact](#)
- [CreateParticipant](#)
- [CreatePushNotificationRegistration](#)
- [DeletePushNotificationRegistration](#)
- [DescribeContact](#)
- [DismissUserContact](#)
- [GetContactAttributes](#)
- [ListAssociatedContacts](#)
- [ListContactReferences](#)
- [ListRealtimeContactAnalysisSegments](#)
- [ListRealtimeContactAnalysisSegmentsV2](#)
- [MonitorContact](#)
- [SearchContacts](#)

- [StartContactStreaming](#)
- [StartOutboundChatContact](#)
- [StartOutboundVoiceContact](#)
- [StartScreenSharing](#)
- [StartTaskContact](#)
- [StartWebRTCContact](#)
- [StopContactStreaming](#)
- [StopContact](#)
- [TagContact](#)
- [TransferContact](#)
- [UntagContact](#)
- [UpdateContact](#)
- [UpdateContactAttributes](#)
- [UpdateContactRoutingData](#)
- [UpdateContactSchedule](#)

For more information about contacts, see the following topics in the *Connect Customer Administrator Guide*:

- [Connect Customer contact events](#)
- [Contact records data model](#)
- [Capture customer audio: live media streaming](#)
- [Customize chat flow experiences by integrating custom participants](#)

Data table actions

The following API actions are available for data tables:

- [BatchCreateDataTableValue](#)
- [BatchDeleteDataTableValue](#)
- [BatchDescribeDataTableValue](#)
- [BatchUpdateDataTableValue](#)

- [CreateDataTable](#)
- [CreateDataTableAttribute](#)
- [DeleteDataTable](#)
- [DeleteDataTableAttribute](#)
- [DescribeDataTable](#)
- [DescribeDataTableAttribute](#)
- [EvaluateDataTableValues](#)
- [ListDataTableAttributes](#)
- [ListDataTablePrimaryValues](#)
- [ListDataTables](#)
- [ListDataTableValues](#)
- [SearchDataTables](#)
- [UpdateDataTableAttribute](#)
- [UpdateDataTableMetadata](#)
- [UpdateDataTablePrimaryValues](#)

For more information about data tables, see [Data tables](#) in the *Connect Customer Administrator Guide*.

Email actions

The following API actions are available for email:

- [AssociateEmailAddressAlias](#)
- [CreateEmailAddress](#)
- [DeleteEmailAddress](#)
- [DescribeEmailAddress](#)
- [DisassociateEmailAddressAlias](#)
- [SearchEmailAddresses](#)
- [SendOutboundEmail](#)
- [StartEmailContact](#)

- [StartOutboundEmailContact](#)
- [UpdateEmailAddressMetadata](#)
- [UpdateQueueOutboundEmailConfig](#)

For more information about email, see [Set up email in Connect Customer](#) in the *Connect Customer Administrator Guide*.

Evaluation actions

The following API actions are available for evaluations:

- [ActivateEvaluationForm](#)
- [CreateEvaluationForm](#)
- [DeactivateEvaluationForm](#)
- [DeleteContactEvaluation](#)
- [DescribeContactEvaluation](#)
- [DeleteEvaluationForm](#)
- [DescribeEvaluationForm](#)
- [ListContactEvaluations](#)
- [ListEvaluationForms](#)
- [ListEvaluationFormVersions](#)
- [StartContactEvaluation](#)
- [SubmitContactEvaluation](#)
- [UpdateContactEvaluation](#)
- [UpdateEvaluationForm](#)

To create a CloudFormation template for evaluation forms, see the following topic:

- [AWS::Connect::EvaluationForm](#) resource in the *CloudFormation User Guide*

For more information about evaluations, see [Evaluate agent performance](#) in the *Connect Customer Administrator Guide*.

Files actions

The following API actions are available for Files:

- [StartAttachedFileUpload](#)
- [CompleteAttachedFileUpload](#)
- [GetAttachedFile](#)
- [BatchGetAttachedFileMetadata](#)
- [DeleteAttachedFile](#)

For more information about attached files, see [Enable attachments to share files using chat and upload files to cases](#) in the *Connect Customer Administrator Guide*.

Flows / Flow module actions

The following API actions are available for flows and flow modules:

- [AssociateFlow](#)
- [BatchGetFlowAssociation](#)
- [CreateContactFlow](#)
- [CreateContactFlowModule](#)
- [CreateContactFlowVersion](#)
- [DeleteContactFlow](#)
- [DeleteContactFlowModule](#)
- [DescribeContactFlow](#)
- [DescribeContactFlowModule](#)
- [DisassociateFlow](#)
- [GetFlowAssociation](#)
- [ListContactFlows](#)
- [ListContactFlowModules](#)
- [ListContactFlowVersions](#)
- [ListFlowAssociations](#)
- [SearchContactFlows](#)

- [SearchContactFlowModules](#)
- [UpdateContactFlowContent](#)
- [UpdateContactFlowMetadata](#)
- [UpdateContactFlowModuleContent](#)
- [UpdateContactFlowModuleMetadata](#)
- [UpdateContactFlowName](#)

To create a CloudFormation template for flows, see the following topics:

- [AWS::Connect::ContactFlow](#)
- [AWS::Connect::ContactFlowModule](#)

Use the [Flow language](#) to programmatically write flows rather than drag blocks onto the flow designer.

For more information about flows, see [Create Connect Customer flows](#) in the *Connect Customer Administrator Guide*.

Hierarchy group actions

The following API actions are available for hierarchy groups:

- [CreateUserHierarchyGroup](#)
- [DeleteUserHierarchyGroup](#)
- [DescribeUserHierarchyGroup](#)
- [DescribeUserHierarchyStructure](#)
- [ListUserHierarchyGroups](#)
- [SearchUserHierarchyGroups](#)
- [UpdateUserHierarchy](#)
- [UpdateUserHierarchyGroupName](#)
- [UpdateUserHierarchyStructure](#)

To create a CloudFormation template for hierarchy groups, see the following topic:

- [AWS::Connect::UserHierarchyGroup](#)

For more information about managing users, see [Manage users in Connect Customer](#) in the *Connect Customer Administrator Guide*.

For more information about agent hierarchies, see [Set up agent hierarchies](#) in the *Connect Customer Administrator Guide*.

Hours of operation actions

The following API actions are available for managing hours of operation:

- [CreateHoursOfOperation](#)
- [DescribeHoursOfOperation](#)
- [DeleteHoursOfOperation](#)
- [ListHoursOfOperations](#)
- [SearchHoursOfOperations](#)
- [UpdateHoursOfOperation](#)
- [ListChildHoursOfOperations](#)
- [AssociateHoursOfOperations](#)
- [DisassociateHoursOfOperations](#)

The following API actions are available for managing hours of operation overrides:

- [CreateHoursOfOperationOverride](#)
- [DeleteHoursOfOperationOverride](#)
- [DescribeHoursOfOperationOverride](#)
- [GetEffectiveHoursOfOperations](#)
- [ListHoursOfOperationOverrides](#)
- [SearchHoursOfOperationOverrides](#)
- [UpdateHoursOfOperationOverride](#)

To create a CloudFormation template for hours of operation, see the following topic:

- [AWS::Connect::HoursOfOperation](#)

For more information about hours of operation and overrides, see [Set the hours of operation and timezone for a queue](#) in the *Connect Customer Administrator Guide*.

Instance actions

The following API actions are available for instances:

- [AssociateApprovedOrigin](#)
- [AssociateBot](#)
- [AssociateInstanceStorageConfig](#)
- [AssociateLambdaFunction](#)
- [AssociateLexBot](#)
- [AssociateSecurityKey](#)
- [CreateInstance](#)
- [DeleteInstance](#)
- [DescribeInstance](#)
- [DescribeInstanceAttribute](#)
- [DescribeInstanceStorageConfig](#)
- [DisassociateApprovedOrigin](#)
- [DisassociateBot](#)
- [DisassociateInstanceStorageConfig](#)
- [DisassociateLambdaFunction](#)
- [DisassociateLexBot](#)
- [DisassociateSecurityKey](#)
- [GetFederationToken](#)
- [ListApprovedOrigins](#)
- [ListBots](#)
- [ListInstances](#)
- [ListInstanceAttributes](#)
- [ListInstanceStorageConfigs](#)

- [ListLambdaFunctions](#)
- [ListLexBots](#)
- [ListSecurityKeys](#)
- [ReplicateInstance](#)
- [UpdateInstanceAttribute](#)
- [UpdateInstanceStorageConfig](#)

To create a CloudFormation template for instances, see the following topics:

- [AWS::Connect::Instance](#)
- [AWS::Connect::InstanceStorageConfig](#)

For more information about instances, see [Create an Connect Customer instance](#) in the *Connect Customer Administrator Guide*.

Integration association actions

The following API actions are available for integration associations:

- [CreateIntegrationAssociation](#)
- [DeleteIntegrationAssociation](#)
- [ListIntegrationAssociations](#)

To create a CloudFormation template for instance associations, see the following topic:

- [AWS::Connect::IntegrationAssociation](#)

For more information about integration associations, see [Set up pre-built integrations](#) in the *Connect Customer Administrator Guide*.

Metrics actions

The following API actions are available for metrics:

- [GetContactMetrics](#)

- [GetCurrentMetricData](#)
- [GetCurrentUserData](#)
- [GetMetricData](#)
- [GetMetricDataV2](#)

For more information about metrics, see [Metrics, dashboards, and reports in Connect Customer](#) in the *Connect Customer Administrator Guide*.

Notifications actions

The following API actions are available for notifications:

- [CreateNotification](#)
- [DeleteNotification](#)
- [DescribeNotification](#)
- [ListNotifications](#)
- [ListUserNotifications](#)
- [SearchNotifications](#)
- [UpdateNotificationContent](#)
- [UpdateUserNotificationStatus](#)

For more information about notifications, see [Notifications Connect Customer](#) in the *Connect Customer Administrator Guide*.

Phone number actions

The following API actions are available for phone numbers:

- [AssociatePhoneNumberContactFlow](#)
- [ClaimPhoneNumber](#)
- [DescribePhoneNumber](#)
- [DisassociatePhoneNumberContactFlow](#)
- [ImportPhoneNumber](#)

- [ListPhoneNumbers](#)
- [ListPhoneNumbersV2](#)
- [ReleasePhoneNumber](#)
- [SearchAvailablePhoneNumbers](#)
- [UpdatePhoneNumber](#)
- [UpdatePhoneNumberMetadata](#)

To create a CloudFormation template for phone numbers, see the following topic:

- [AWS::Connect::PhoneNumber](#)

For more information about phone numbers, see [Set up phone numbers for your contact center](#) in the *Connect Customer Administrator Guide*.

Predefined attributes actions

The following API actions are available for predefined attributes:

- [CreatePredefinedAttribute](#)
- [DescribePredefinedAttribute](#)
- [ListPredefinedAttributes](#)
- [SearchPredefinedAttributes](#)
- [UpdatePredefinedAttribute](#)

To create a CloudFormation template for predefined attributes, see the following topic:

- [AWS::Connect::PredefinedAttribute](#)

For more information about predefined attributes, see [Create predefined attributes for routing contacts to agents](#) in the *Connect Customer Administrator Guide*.

Prompt actions

The following API actions are available for prompts:

- [CreatePrompt](#)
- [DeletePrompt](#)
- [DescribePrompt](#)
- [GetPromptFile](#)
- [ListPrompts](#)
- [SearchPrompts](#)
- [UpdatePrompt](#)

To create a CloudFormation template for prompts, see the following topic:

- [AWS::Connect::Prompt](#)

For more information about prompts, see [Create prompts](#) in the *Connect Customer Administrator Guide*.

Queue actions

The following API actions are available for queues:

- [CreateQueue](#)
- [DeleteQueue](#)
- [DescribeQueue](#)
- [ListQueues](#)
- [SearchQueues](#)
- [UpdateQueueHoursOfOperation](#)
- [UpdateQueueMaxContacts](#)
- [UpdateQueueName](#)
- [UpdateQueueOutboundCallerConfig](#)
- [UpdateQueueStatus](#)

To create a CloudFormation template for queues, see the following topic:

- [AWS::Connect::Queue](#)

For more information about queues, see [Set up routing](#) in the *Connect Customer Administrator Guide*.

Quick connect actions

The following API actions are available for quick connects:

- [AssociateQueueQuickConnects](#)
- [CreateQuickConnect](#)
- [DeleteQuickConnect](#)
- [DescribeQuickConnect](#)
- [DisassociateQueueQuickConnects](#)
- [ListQueueQuickConnects](#)
- [ListQuickConnects](#)
- [SearchQuickConnects](#)
- [UpdateQuickConnectConfig](#)
- [UpdateQuickConnectName](#)

To create a CloudFormation template for quick connects, see the following topic:

- [AWS::Connect::QuickConnect](#)

For more information about quick connects, see [Create quick connects](#) in the *Connect Customer Administrator Guide*.

Routing profile actions

The following API actions are available for routing profiles:

- [AssociateRoutingProfileQueues](#)
- [CreateRoutingProfile](#)
- [DeleteRoutingProfile](#)
- [DescribeRoutingProfile](#)
- [DisassociateRoutingProfileQueues](#)

- [ListRoutingProfileManualAssignmentQueues](#)
- [ListRoutingProfiles](#)
- [ListRoutingProfileQueues](#)
- [SearchRoutingProfiles](#)
- [UpdateRoutingProfileConcurrency](#)
- [UpdateRoutingProfileDefaultOutboundQueue](#)
- [UpdateRoutingProfileName](#)
- [UpdateRoutingProfileQueues](#)

To create a CloudFormation template for routing profiles, see the following topic:

- [AWS::Connect::RoutingProfile](#)

For more information about routing profiles, see [Create a routing profile](#) in the *Connect Customer Administrator Guide*.

Rules actions

The following API actions are available for rules:

- [CreateRule](#)
- [DeleteRule](#)
- [DescribeRule](#)
- [ListRules](#)
- [UpdateRule](#)

To create a CloudFormation template for rules, see the following topic:

- [AWS::Connect::Rule](#)

Use the [Rules Function language](#) to code conditions for the rules.

For more information about Connect Customer rules, see [Create rules](#) and [Create rules with Contact Lens](#) in the *Connect Customer Administrator Guide*.

Security profile actions

The following API actions are available for security profiles:

- [CreateSecurityProfile](#)
- [DescribeSecurityProfile](#)
- [DeleteSecurityProfile](#)
- [ListSecurityProfiles](#)
- [ListSecurityProfileApplications](#)
- [ListSecurityProfilePermissions](#)
- [SearchSecurityProfiles](#)
- [UpdateSecurityProfile](#)

To create a CloudFormation template for tasks, see the following topic:

- [AWS::Connect::SecurityProfile](#)

For more information about security profiles, see [Security profiles](#) in the *Connect Customer Administrator Guide*.

Tag actions

The following API actions are available for tags:

- [SearchResourceTags](#)
- [TagResource](#)
- [ListTagsForResource](#)
- [UntagResource](#)

For more information about tags, see [Tagging resources in Connect Customer](#) in the *Connect Customer Administrator Guide*.

Task actions

The following API actions are available for tasks:

- [CreateTaskTemplate](#)
- [DeleteTaskTemplate](#)
- [GetTaskTemplate](#)
- [ListTaskTemplates](#)
- [StartTaskContact](#)
- [UpdateTaskTemplate](#)

To create a CloudFormation template for tasks, see the following topic:

- [AWS::Connect::TaskTemplate](#)

For more information about tasks, see the following topics in the *Connect Customer Administrator Guide*:

- [Tasks](#)
- [Set up tasks](#)
- [Set up applications for task creation](#)

Traffic distribution group actions

The following API actions are available for traffic distribution groups:

- [AssociateTrafficDistributionGroupUser](#)
- [CreateTrafficDistributionGroup](#)
- [DeleteTrafficDistributionGroup](#)
- [DescribeTrafficDistributionGroup](#)
- [DisassociateTrafficDistributionGroupUser](#)
- [GetTrafficDistribution](#)
- [ListTrafficDistributionGroups](#)
- [ListTrafficDistributionGroupUsers](#)
- [UpdateTrafficDistribution](#)

To create a CloudFormation template for traffic distribution groups, see the following topic:

- [AWS::Connect::TrafficDistributionGroup](#)

For more information about traffic distribution groups, see [Set up Connect Customer Global Resiliency](#) in the *Connect Customer Administrator Guide*.

Use cases actions

The following API actions are available for use cases:

- [CreateUseCase](#)
- [DeleteUseCase](#)
- [ListUseCases](#)

User management actions

The following API actions are available for user management:

- [AssociateUserProficiencies](#)
- [CreateUser](#)
- [DeleteUser](#)
- [DescribeUser](#)
- [DisassociateUserProficiencies](#)
- [ListUsers](#)
- [ListUserProficiencies](#)
- [PutUserStatus](#)
- [SearchUsers](#)
- [UpdateUserConfig](#)
- [UpdateUserIdentityInfo](#)
- [UpdateUserPhoneConfig](#)
- [UpdateUserRoutingProfile](#)
- [UpdateUserSecurityProfiles](#)

To create a CloudFormation template for users, see the following topic:

- [AWS::Connect::User](#)

For more information about managing users, see [Manage users in Connect Customer](#) in the *Connect Customer Administrator Guide*.

View actions

The following API actions are available for views:

- [CreateView](#)
- [CreateViewVersion](#)
- [DeleteView](#)
- [DeleteViewVersion](#)
- [DescribeView](#)
- [ListViews](#)
- [ListViewVersions](#)
- [UpdateViewContent](#)
- [UpdateViewMetadata](#)
- [View](#)
- [ViewContent](#)
- [ViewInputContent](#)
- [ViewSummary](#)
- [ViewVersionSummary](#)

To create a CloudFormation template for views, see the following topics:

- [AWS::Connect::View](#)
- [AWS::Connect::ViewVersion](#)

For more information about views, see the following topic in the *Connect Customer Administrator Guide*:

- [View resource](#)

Vocabulary actions

The following API actions are available for vocabularies:

- [AssociateDefaultVocabulary](#)
- [CreateVocabulary](#)
- [DeleteVocabulary](#)
- [DescribeVocabulary](#)
- [ListDefaultVocabularies](#)
- [SearchVocabularies](#)

For more information about custom vocabularies, see [Add custom vocabularies](#) in the *Connect Customer Administrator Guide*.

Voice actions

The following API actions are available for voice:

- [ResumeContactRecording](#)
- [StartContactRecording](#)
- [StartOutboundVoiceContact](#)
- [StopContactRecording](#)
- [SuspendContactRecording](#)

For more information, see the following topics in the *Connect Customer Administrator Guide*:

- [Monitor live conversations](#)
- [Review recorded conversations](#)

Workspace actions

The following API actions are available for workspaces:

- [AssociateWorkspace](#)

- [CreateWorkspace](#)
- [CreateWorkspacePage](#)
- [DeleteWorkspace](#)
- [DeleteWorkspaceMedia](#)
- [DeleteWorkspacePage](#)
- [DescribeWorkspace](#)
- [DisassociateWorkspace](#)
- [ImportWorkspaceMedia](#)
- [ListWorkspaceMedia](#)
- [ListWorkspacePages](#)
- [ListWorkspaces](#)
- [SearchWorkspaceAssociations](#)
- [SearchWorkspaces](#)
- [UpdateWorkspaceMetadata](#)
- [UpdateWorkspacePage](#)
- [UpdateWorkspaceTheme](#)
- [UpdateWorkspaceVisibility](#)

For more information about workspaces, see [Workspaces](#) in the *Connect Customer Administrator Guide*.

Connect Customer Flow language

This section describes the Connect Customer Flow language and how to use it. The Flow language is a JSON-based representation of a series of flow actions, and the criteria for moving between them.

We've provided you with the Flow language so you can:

- Efficiently update flows that you're migrating from one instance to another.
- Write flows rather than drag blocks onto the flow designer.

Contents

- [Connect Customer Flow language concepts](#)
- [Example flow in Connect Customer Flow language](#)
- [Actions in the Connect Customer Flow Language](#)

Connect Customer Flow language concepts

The following terms are used in the Flow language.

Contact

Flows can be run in context of a contact. In this case, they are referred to as *flows*.

Participant

Flows can additionally be run in a participant context. This allows participant actions—such as playing prompts or getting customer input—to be run. Certain types of flows, such as "No participants remaining" disconnect flows and Workitem flows, don't have a participant associated.

Action types

Flow actions have the following implicit types associated with them. A type determines when an action is attempted.

- [Contact actions in the Connect Customer Flow language](#). These actions are attempted only when the flow is run in context of a contact. They generally result in contact data being manipulated in some way.
- [Flow control actions in the Connect Customer Flow language](#). These actions are used only to determine the path through a flow. They have no side effects. Certain data may not be available. For example, contact data isn't available if the action is determining its path based on contact data. These actions generally work in every circumstance.
- [Interactions in the Connect Customer Flow language](#). These actions have side effects, but don't require a contact or a participant. Interactions include actions such as invoking an AWS Lambda function. They generally work in every circumstance.
- [Participant actions in the Connect Customer Flow language](#). These actions are attempted only when the flow is run in context of a participant. They generally result in an action that the participant experiences, such as playing a prompt or disconnecting.

Example flow in Connect Customer Flow language

The following example shows a JSON string that represents the content of the flow. It shows a simple flow that plays a prompt using static text and disconnects.

To learn how to get block identifiers, we recommend creating a new flow in Connect Customer console, and then calling the [DescribeContactFlow](#) API for it.

```
{
  "Version": "2019-10-30", //A string representing the version of the Flow.
  //Currently the only supported version is 2019-10-30.

  "StartAction": "12345678-1234-1234-1234-123456789012", //A string representing the
  //first Action to run when the flow starts running.
  //In this case, it's the
  //identifier of the Play prompt block.
  //The value of this field must
  //match the Identifier of an Action in the Actions list.
  "Metadata": { //An object that may be filled in with data as desired.
    "EntryPointPosition": {
      "x": 88,
      "y": 100
    },
    "ActionMetadata": {
```

```

    "12345678-1234-1234-1234-123456789012": {//The identifier of the Play
prompt block.
        "Position": {
            "x": 270,
            "y": 98
        }
    },
    "abcdef-abcd-abcd-abcd-abcdefghijkl": {//The identifier of the Disconnect/
hang up block.
        "Position": {
            "x": 545,
            "y": 92
        }
    }
},
"Actions": [ //A list of individual Action objects. These Actions are the
definition of the Flow's behavior and are detailed below.
    //A single Flow may have no more than 250 Actions defined.
    {
        "Identifier": "12345678-1234-1234-1234-123456789012", //The identifier of
the Play prompt block.
        "Type": "MessageParticipant", //This is the flow action.
        "Transitions": {
            "NextAction": "abcdef-abcd-abcd-abcd-abcdefghijkl", //The identifier of
the Disconnect/hang up block.
            "Errors": [],
            "Conditions": []
        },
        "Parameters": {
            "Text": "Thanks for calling the sample flow!"
        }
    },
    {
        "Identifier": "abcdef-abcd-abcd-abcd-abcdefghijkl", //The identifier of
the Disconnect/hang up block.
        "Type": "DisconnectParticipant", //This is the flow action.
        "Transitions": {},
        "Parameters": {}
    }
]
}

```

Actions in the Connect Customer Flow Language

An Action is a single step of a flow's run. This topic describes the fields that must be defined.

Identifier

A string that must be unique among all Actions within the same Flow. This Identifier can be up to 50 characters long, and can include any characters (including unicode and spaces). They can be opaque or user-friendly.

Important

- The following characters are not allowed in the Identifier field: (% : (\ /) = \$, ; [] { })
- The following strings are not allowed in the Identifier field: __proto__, constructor, __defineGetter__, __defineSetter__, toString, hasOwnProperty, isPrototypeOf, propertyIsEnumerable, toLocaleString, and valueOf.

Type

A string that identifies the type of action being performed for a particular step of the Flow. This type must be one of a list of allowable Types, which are covered later.

Parameters

An object that defines the customizable behavior of a particular Action block. Each Action has its own format of this Parameters object, which is detailed in the individual Actions definition.

The Parameters object defines customizable behavior for the Action. For example, it defines which Attributes to set or which AWS Lambda function to run. The format differs for each Action type. To find the specific format of a specific Action's Parameter object, see the individual Action's definition below.

Transitions

An object that defines the behavior for choosing the next Action after the current Action completes. Certain Actions terminate, meaning that they finish running the flow when they're run. This is because Transitions must be defined as an empty object.

The Transitions object defines how to proceed to the next Action during flow runtime. This object must have the following fields specified:

NextAction

NextAction is a string that contains the Identifier of the Action that should be run after this Action, if no error or condition is preferentially chosen.

Errors

Errors is a list of error objects. Each error object contains a type or category of error (ErrorType), and the Identifier of the Action that should be run subsequently when that error occurs (NextAction).

Each individual Action supports specific Errors. These are detailed in the Action's definition.

Conditions

Conditions are an ordered list that defines a series of checks to evaluate against the Action's result. This result changes per Action and can also change based on Parameters - examples of these are "the number of contacts in queue" for the CheckMetricData Action if the MetricType parameter refers to the NumberOfContactsInQueue, and "the value of the attribute" for the Compare Action. Conditions are evaluated in order, and the first Condition that evaluates to true will result in it being chosen as the Transition to occur, making that Condition's Target the next Action run. The Conditions object is explained in more detail below.

A Condition is a definition of how to evaluate an Action's result, and may evaluate to true or false. The Conditions object on the flow contains an ordered list of objects. Each object contains a NextAction (the Identifier of the Action to be invoked if the Condition evaluates to be true) and the Condition to evaluate:

- **NextAction:** A string that contains the Identifier of the Action that should be run after this Action if this Condition is the first condition to evaluate to true.
- **Condition:** An object that defines the evaluation logic.

The Condition object

The Condition object must contain the following fields:

- **Operator:** A string that indicates which comparison operator that is applied to the Operands. The list of allowed Operators and a description of their logic is defined in the following table.
- **Operands:** A list of operands to which the Operator is applied. Depending on the Operator, these Operands may be strings or they may be Condition objects. The specific Operator defines which type of Operand is expected, along with the number of Operands expected (some Operators will require only one Operand, some will support a list of up to ten Operands). Conditions may be nested no more than five Conditions deep, and a single Condition may not contain more than 50 sub-Conditions, regardless of how deeply nested they are.

List of Operators

Operator	Description	Operand type	Operand count
Equals	Returns true if the string specified exactly equals the result.	String	One
TextStartsWith	Returns true if the result, interpreted as text, begins with the specified string.	String	One
TextEndsWith	Returns true if the result, interpreted as text, ends with the specified string.	String	One
TextContains	Returns true if the result, interpreted as text, contains the specified string at least once.	String	One

Operator	Description	Operand type	Operand count
NumberGreaterThan	Returns true if the result, interpreted as a numeric value, is larger than the specified string. If either the result or the specified string are not numeric, returns false .	String	One
NumberGreaterOrEqualTo	Returns true if the result, interpreted as a numeric value, is larger than or equal to the specified string. If either the result or the specified string are not numeric, returns false .	String	One
NumberLessThan	Returns true if the result, interpreted as a numeric value, is smaller than the specified string. If either the result or the specified string are not numeric, returns false .	String	One

Operator	Description	Operand type	Operand count
NumberLessOrEqualTo	Returns true if the result, interpreted as a numeric value, is smaller than or equal to the specified string. If either the result or the specified string are not numeric, returns false .	String	One

Example Condition

Following is an example of a condition that returns true if the result starts with "ABC":

```
{
  "Operator": "TextStartsWith",
  "Operands": [
    "ABC"
  ]
}
```

Parameter restrictions for actions in the Connect Customer Flow language

There are several restrictions on parameters. Here's what they mean:

- Must be defined statically. This means that JSONPath cannot be used at all in this value.
- Must be defined statically or as a single valid JSONPath identifier.

If JSONPath is used, it must be the entirety of the value; you can't specify an input of "My name is \$.Name". Further, the JSONPath must be valid - \$.Attributes.stuff is okay, \$.BadValue is not okay because there's no "BadValue" path on the object used by flows.

- May be defined statically or dynamically. Anything goes. A value of "My name is \$.Name" is fine here, as well as a fully static value.

Contact actions in the Connect Customer Flow language

Contact actions are attempted only for flows that run in context of a contact. They generally result in contact data being manipulated in some way.

Contents

- [CompleteOutboundCall](#)
- [CreateCase](#)
- [CreateTask](#)
- [CreateWisdomSession](#)
- [DequeueContactAndTransferToQueue](#)
- [EndFlowModuleExecution](#)
- [GetCase](#)
- [InvokeFlowModule](#)
- [StartOutboundChatContact](#)
- [TagContact](#)
- [TransferContactToAgent](#)
- [TransferContactToQueue](#)
- [UnTagContact](#)
- [UpdateCase](#)
- [UpdateContactAttributes](#)
- [UpdateContactCallbackNumber](#)
- [UpdateContactData](#)
- [UpdateContactEventHooks](#)
- [UpdateContactMediaProcessing](#)
- [UpdateContactMediaStreamingBehavior](#)
- [UpdateContactRecordingAndAnalyticsBehavior](#)
- [UpdateContactRecordingBehavior](#)
- [ResumeContact](#)
- [UpdateContactRoutingBehavior](#)
- [UpdateContactTargetQueue](#)

- [UpdateContactTextToSpeechVoice](#)
- [UpdatePreviousContactParticipantState](#)

CompleteOutboundCall

When a flow is run before an outbound call is made as part of an outbound contact, this action calls the outbound destination. If this action is not used, the first participant action implicitly completes the outbound call.

Parameter object

```

{
  "CallerId": {
    Optional, an override of the caller ID to present when calling. Is
    ignored when using VoiceConnectors
    "Number": The caller ID number to present when calling. Can either be fully static or
    a single valid JSONPath identifier
  }
  "VoiceConnector": {
    Optional, Configuration of the voice connector
    "VoiceConnectorType": Only support "ChimeConnector". Must be defined statically.
    "VoiceConnectorArn": The ARN of the Voice Connector. Can be set statically or
    dynamically.
    "FromUser": The user who makes the call. Can be set statically or dynamically.
    "ToUser": The user who receives the call. Can be set statically or dynamically.
    "UserToUserInformation": Optional, SIP user-to-user information. Can be set
    statically or dynamically.
  }
  "ConnectionTimeLimitSeconds": Optional, Only used for Voice Connector use case. An
  integer between 1 and 600
                                (inclusive) representing the
  number of seconds
                                to wait for the voice connector
  to answer before
                                canceling the call. Can be set
  statically or dynamically.

```

Results and conditions

None.

Errors

None.

Restrictions

This action can be used only when the contact is in the process of making an outbound call, but has not yet called the outbound number.

Corresponding block in the UI

[Call phone number](#)

CreateCase

Creates a new case using an existing case template. Templates come with predefined fields, some of which are required and will appear in a side panel when you start. You can review and fill in the required fields, and optionally update any other available fields based on your needs. These fields are set up in your instance ahead of time. You can also choose to link a contact to the new case if needed.

Parameter object

```
"Parameters": {
  "LinkContactToCase": "true" or "false", If set to true, cases will open
  automatically when the agent accepts the contact.
  "CaseTemplateId": A templateId aligned with the existing case templateName
  "CaseRequestFields": An optional map of case fields to be set. Keys should be
  fields from Cases domain. Values can be static or dynamic.
}
```

Results and conditions

None.

Errors

- **ContactNotLinked** - If you specify to link the contact to case, then this error branch will appear. It might be that the contact was not linked after the case is retrieved (partial success/partial failure). If this happens, then the flow will follow this branch.
- **NoMatchingError** - An error was encountered while trying to find the case. This may be due to a system error or how CreateCase is configured.

Restrictions

None. This can be used in any type of flow and any channel.

Corresponding block in the UI

[Flow block: Cases](#)

CreateTask

Creates a new task to run an assigned flow.

Parameter object

```
"Parameters":
{
    "ContactFlowId": A flow ID or flow ARN. *Must be fully static or a single valid
JSONPath identifier*,
    "Attributes": [Optional] { an Object that holds the attributes to be set.
        "Key": "Value" Both the key and value may be defined statically or
dynamically.
    },
    "Name": Name of the task that needs to be created. This is a string.,
    "Description": [Optional] Description of the task that needs to be created.
This is a string.,
    "References": [Optional] { an Object that holds the references to be set.
        "Type": "Value" Both the key and value may be defined statically or
dynamically.
    },
    "DelaySeconds": [Optional] Time in seconds after which the task should be
created. This is used to schedule the task by the agent.
    The integer value between 1 and 518400 (6 days). If ScheduledTime is specified,
this parameter may not be specified.
    "ScheduledTime": [Optional] The date and time at which the task should be
created. If DelaySeconds is specified, this parameter may not be specified.
    "TaskTemplateId": [Optional] ID of the task template that will be used to
create the task. This must be defined statically.
}
}
```

Results and conditions

None. No conditions are supported.

Errors

NoMatchingError - if no other Error matches.

Restrictions

This action is supported on all channels and in all contact flow types.

Corresponding block in the UI

[Flow block: Create task](#)

CreateWisdomSession

Associates a Wisdom domain to a contact that is being executed in a Flow to enable real-time recommendations on the current contact.

Parameter object

```
{
  "WisdomAssistantArn": ARN for the Wisdom Assistant. May be specified statically or
  dynamically.
}
```

Results and conditions

None. No conditions are supported.

Errors

NoMatchingError - if no other Error matches.

Restrictions

This action is only supported on the voice channel. This action can be used in all contact flow types.

Corresponding block in the UI

[Flow block: Amazon Q in Connect](#)

DequeueContactAndTransferToQueue

This action is a combination of a "Dequeue" action and a "TransferContactToQueue" action. This means that a contact in a queue is removed from the queue, a new contact segment is created with the existing contact as its previous contact, and the new contact is placed into the specified queue (referred to as "Queue-to-queue transfer"). If this contact has not been queued, is actively being joined to an agent, or has been routed to an agent, this action fails.

Parameter object

```
{
  "QueueId": [Optional] A queue ID or queue ARN. If AgentId is specified, this may
not be specified. Must be either fully statically defined or a single, valid JSONPath
identifier.
  "AgentId": [Optional] An agent ID or agent ARN, representing an agent queue. If
QueueId is specified, this may not be specified. Must be either fully statically
defined or a single, valid JSONPath identifier.
}
```

Results and conditions

None.

Errors

- QueueAtCapacity - if the destination queue is at capacity and the contact cannot be queued within it.
- NoMatchingError - if no other Error matches.

Restrictions

This action is only supported in the customer queue flow. It is not supported in any other type of flow.

Corresponding block in the UI

Maps to [Transfer to queue](#) block but only when used in a Customer queue flow.

EndFlowModuleExecution

Ends the current module execution without disconnecting the contact.

Parameter object

```
{
}
```

Results and conditions

None. No conditions are supported.

Errors

None

Restrictions

This action is available only in flow modules.

Corresponding block in the UI

Return Block (not yet documented)

GetCase

Searches all existing cases with the provided customer ID. Add request fields to filter by case fields. Specify the case fields to be returned in the response to persist in the context.

Parameter object

```
{
  "LinkContactToCase": "true" or "false", If set to true, cases will open
  automatically when the agent accepts the contact.
  "GetLastUpdatedCase": "true" or "false", You can specify to get only the last
  updated case for any search criteria.
  "CustomerId": "Customer's Id", Search all cases for this customer Id.
  "CaseRequestFields": An optional map of cases request fields. Keys should be fields
  from Cases domain. Values can be static or dynamic.
  "CaseResponseFields": [ ] An optional list of field names that should be persisted
  in the case namespace.
}
```

Results and conditions

None.

Errors

- **NoMatchingError** - An error was encountered while trying to find the case. This may be due to a system error or how Get case is configured.

- **ContactNotLinked** - If you specify to link the contact to case, then this error branch will appear. It might be that the contact was not linked after the case is retrieved (partial success/partial failure). If this happens, then the flow will follow this branch.
- **MultipleFound** - Multiple cases are found with the search criteria.
- **NoneFound** - No cases are found with the search criteria.

Restrictions

None. This can be used in any type of flow and any channel.

Corresponding block in the UI

[Flow block: Cases](#)

InvokeFlowModule

Invokes a flow module. *Flow modules* are reusable sections of a flow. You use them to extract repeatable logic across your flows, and create common functions. For more information about flow modules, see [Flow modules for reusable functions](#), in the *Connect Customer Administrator Guide*.

Parameter object

```
{
  "FlowModuleId": The flow module ID or flow module ARN to be invoked. May be defined
  statically or dynamically.
}
```

Results and conditions

None.

Errors

`NoMatchingError` if no other Error matches.

Restrictions

This action is supported by all channels and only supports Inbound flow types.

Corresponding block in the UI

[Flow block: Invoke module.](#)

StartOutboundChatContact

Initiate an outbound chat contact to a customer. Only SMS chats are supported. For more information, see the [StartOutboundChatContact](#) in the *Connect Customer API Reference*.

Parameter object

```
{
  "SourceEndpoint": {
    "Address": ConnectPhoneNumberArn of the SourceEndpoint,
    "Type": Type of the SourceEndpoint, currently only supports
CONNECT_PHONENUMBER_ARN
  },
  "DestinationEndpoint": {
    "Address": E164 phone number of the DestinationEndpoint,
    "Type": Type of the DestinationEndpoint, currently only supports
TELEPHONE_NUMBER
  },
  "ContactFlowArn": ContactFlowArn of the flow to be executed for the new outbound
chat,
  "ContactSubtype": Subtype of the new chat contact, currently only supports
connect:SMS,
  "InitialSystemMessage": [Optional] {
    "Content": content of the initial system message to be sent to the
DestinationEndpoint
  },
  "RelatedContact": [Optional] Only supported value is CURRENT
}
```

Results and conditions

None. No conditions are supported.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

- This action only supports connect:SMS as the **ContactSubtype** currently.
- This action only supports CONNECT_PHONENUMBER_ARN as the Type of **SourceEndpoint** currently.

- This action only supports TELEPHONE_NUMBER as the Type of **DestinationEndpoint** currently.

Corresponding block in the UI

[Send message](#)

TagContact

Sets a collection of tag to the current contact. With this type of operation, either all tags are set or none are set.

Each tag on a contact is a key-value pair, that is, it is composed of a key and a value. There are two types of tags: user-defined tags and system tags. You create a user-defined tag. A system tag is automatically created by AWS services. A system tag is prefixed with `aws :`. You cannot change it.

The TagContact action parameters correspond to user-defined tags that are applied on a contact.

Parameter object

```
{
  "Tags": { an Object that holds the tags to be set.
    "Key1":"Value1" Both the key and value may be defined statically or dynamically.
  }
}
```

Results and conditions

None.

Errors

None.

Restrictions

None. This can be used in any type of flow and any channel.

Corresponding block in the UI

[Contact tags](#)

TransferContactToAgent

Ends the current flow and transfers the customer to an agent. If the agent is already with someone else, the contact is disconnected. Transfer contact to agent works only for voice interactions.

Parameter object

No parameters are expected.

Results and conditions

None.

Errors

None.

Restrictions

This action is supported in only transfer to agent and transfer to queue flows.

Corresponding block in the UI

[Transfer to agent](#)

TransferContactToQueue

This action places a contact that is not already in a queue into the contact's TargetQueue. If the contact has already been put into a queue (meaning that it is currently being routed to an agent, being joined to an agent, or is connected to an agent), the action fails.

Parameter object

No parameters are expected.

Results and conditions

None.

Errors

- QueueAtCapacity - if the destination queue is at capacity and the contact cannot be queued within it.

- NoMatchingError - if no other Error matches.

Restrictions

This action is supported in inbound contact flows and transfer flows. It is not supported in whisper flows, customer queue flows, or hold flows.

Corresponding block in the UI

[Transfer to queue](#)

UntagContact

Removes a collection of tags on the current contact. With this type of operation, either all tags are set or none are set.

You cannot remove system-defined tags. You can only remove already existing user-defined tags from a contact.

Parameter object

```
{
  "TagKeys": [ an Object that holds the tag-keys for the tags to be removed.
    "Key1" Key(s) can only be set statically.
  ]
}
```

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches

Restrictions

This action is supported across all the Connect Customer media channels.

This action can be used in flows of all types.

Corresponding block in the UI

[Contact tags](#)

UpdateCase

Updates an existing case by providing the case's id and the fields that should be updated.

Parameter object

```
{
  "LinkContactToCase" : "true" or "false", If set to true, cases open automatically
  when the agent accepts the contact.
  "CaseId": the unique identifier of the case
  "CaseRequestFields": An optional map of case fields to be updated. Keys should be
  fields from the Cases domain. Values can be static or dynamic.
}
```

Results and conditions

None.

Errors

- **ContactNotLinked** - If you specify to link the contact to case, then this error branch will appear. It might be that the contact was not linked after the case is retrieved (partial success/partial failure). If this happens, then the flow will follow this branch.
- **NoMatchingError** - An error was encountered while trying to find the case. This may be due to a system error, or how Update case is configured.

Restrictions

None. This can be used in any type of flow and any channel.

Corresponding block in the UI

[Flow block: Cases](#)

UpdateContactAttributes

Sets a collection of contact attributes on either the current contact or the related contact. With this type of operation, either all attributes are set or none are set.

Parameter object

```
{
  "Attributes": { an Object that holds the attributes to be set.
    "Key": "Value" Both the key and value may be defined statically or dynamically.
  },
  "TargetContact": either "Current" or "Related". Must be defined statically.
}
```

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

None. This can be used in any type of flow and any channel.

Corresponding block in the UI

[Set contact attributes](#)

UpdateContactCallbackNumber

Updates the contact callback number, which is the number used by the CreateCallbackContact action. This value defaults to the customer participant caller ID if this action is never used.

Parameter object

```
{
  "CallbackNumber": The callback number to set. Must be a single, valid JSONPath
    reference, and cannot be set statically.
}
```

Results and conditions

None.

Errors

- **InvalidCallbackNumber** - The callback number specified was not a valid (e.164) phone number.
- **CallbackNumberNotDialable** - The callback number specified is not dialable by the instance.

Restrictions

This is supported only in contact flows, transfer flows, and customer queue flows. This is not supported in whispers or hold flows.

Corresponding block in the UI

[Set callback number](#)

UpdateContactData

Sets a collection of connect defined attributes on specified contact. With this type of operation, either all attributes are set or none are set.

Parameter object

```
"Parameters":
{
    "Name": [Optional] The name of the contact. It is a string. May be set
statically or dynamically.
    "Description": [Optional] The description of the contact. It is a string.
    "LanguageCode": [Optional] The language to use for current contact.
    "CustomerId": [Optional] The customer id associated with the contact.
    "References": [Optional] { an Object that holds the references to be set.
        "Type": "Value" Both the key and value may be defined statically or
dynamically.
    },
    "IsVoiceIdStreamingEnabled": [Optional] Enable to start streaming audio from
customer channel to Voice ID. It is a string. "TRUE" and "FALSE" are the only valid
values.
    "IsVoiceAuthenticationEnabled": [Optional] Enable authentication by comparing
voiceprint of the caller to the enrolled voiceprint of the claimed identity. It is a
string. "TRUE" and "FALSE" are the only valid values.
    "IsFraudDetectionEnabled": [Optional] Enable detection for impersonation
attempts and presence of known fraudsters. It is a string. "TRUE" and "FALSE" are the
only valid values.
```

```
"VoiceAuthenticationThreshold": [Optional] Threshold to validate against
confidence score of a voice match. It is a string. Value must be between 0 and 100.
"VoiceAuthenticationResponseTime": [Optional] Define required minimum caller
speech seconds for voice authentication. It is a string. Value must be between 5 and
10.
"FraudDetectionThreshold": [Optional] The threshold you set for fraud detection
is used to measure risk. Scores higher than the threshold are reported as higher risk.
Scores lower than the threshold are reported as lower risk. Raising the threshold
lowers false positive rates (makes result more certain), but raises false negative
rates. It is a string. Value must be between 0 and 100.
"WatchlistId": [Optional] Identifier of watchlist to use when evaluating the
voice session. It is a string. Value must be between 0 and 100.
"WisdomSessionArn": [Optional] A session ARN provided by Connect Customer
Wisdom for agent assistance. It is a string.
"TargetContact": [Required] Target contact on which given attributes should be
set. It is a string. "Current" or "Related" are the only valid values.
}
```

Results and conditions

None. No conditions are supported.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is supported on all channels and in all flow types.

Corresponding blocks in the UI

- [Set contact attributes](#)

UpdateContactEventHooks

Sets one or more contact event hooks, which are flows associated with contact events, such as customer whisper or agent hold. For more information, see [Set event flow](#). The following event hooks are valid:

- AgentHold

- AgentWhisper
- CustomerHold
- CustomerQueue
- CustomerRemaining
- CustomerWhisper
- DefaultAgentUI
- DisconnectAgentUI
- PauseContact
- ResumeContact

Parameter object

```
{
  "EventHooks": { an Object that holds the event hooks to be set. Only one entry may
    be present in this map.
    "Key": "Value" - the event hook to be set where the key is the event type and
    the value is the flow ID or ARN to run when that event occurs. Keys must be defined
    statically.
  }
}
```

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This is supported in all types of flows.

Corresponding blocks in the UI

- [Set customer queue flow](#)
- [Set event flow](#)
- [Set hold flow](#)

- [Set whisper flow](#)

UpdateContactMediaProcessing

Allows customers to configure their own Lambda processor, which will be applied to in-flight messages.

Parameter object

```
{
  "ChatProcessor": { // Configuration for bring-your-own-processor for chat channel.
    "ProcessingEnabled": either "True" or "False". Must be set statically.
    Determines whether to enable custom Lambda processing for chat messages.
    "LambdaProcessorARN": The ARN of the Lambda function to process chat messages.
    Must be set statically. Format: arn:aws:lambda:region:account-id:function:function-
    name
    "ChatProcessorSettings": { // An object that holds chat processor behavior
    settings
      "DeliverUnprocessedMessages": either "True" or "False". Must be set
      statically. Determines whether to deliver messages that fail Lambda processing.
    }
  }
}
```

Results and conditions

None.

Errors

- **NoMatchingError** - if no other Error matches. Must always be defined.
- **ChannelMismatch** - if the media channel that initiated the contact is not the same as the one defined in the action. As of now, only chat is supported in this action.

Corresponding block in the UI

[Set recording, analytics and processing behavior](#)

UpdateContactMediaStreamingBehavior

Enables or disables contact media streaming for a set of participants.

Parameter object

```
{
  "MediaStreamingState": One of "Enabled" or "Disabled". Must be specified
  statically.
  "Participants": [ A list of participants to include in the stream if enabling the
  stream, or disable if disabling the stream
    {
      "ParticipantType": The type of participant to stream. Currently, only
      "Customer" is supported. Must be defined statically.
      "MediaDirections": [ ] A list of the directions of media to include in the
      stream - "From" and "To". Must be defined statically.
    }
  ],
  "MediaStreamType": The type of media to enable or disable from the stream.
  Currently, only "Audio" is supported. Must be defined statically.
}
```

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This is supported in contact flows, customer queue flows, transfer flows, and whisper flows. It is not supported in hold flows.

This is supported only by the voice channel.

Corresponding block in the UI

[Start media streaming](#) and [Stop media streaming](#)

UpdateContactRecordingAndAnalyticsBehavior

Sets contact recording behavior, including analysis behavior and which participants of the contact to record.

Parameter object

```
{
  // Only ONE of the following channel behavior objects can be defined per
  configuration

  "ChatBehavior": { // Configuration for chat channel interactions
    "ChatAnalyticsBehavior": {
      "Enabled": either "True" or "False". Must be set statically.
      "AnalyticsLanguage": Must be one of languages supported by Contact Lens
      analysis. Can be set dynamically. Use the format xx-XX, for example, en-US for US
      English.
      "ConversationalAnalyticsRedactionConfiguration": {
        "Enabled": either "True" or "False". Must be set statically. Determines
        whether to redact sensitive data in Contact Lens output.
        "RedactionResults": either "RedactedAndOriginal" or "RedactedOnly".
        Can be set dynamically. Determines whether to provide both versions or only redacted
        output.
        "RedactionMaskMode": either "EntityType" or "PII". Must be set
        statically.
        "RedactionEntities": [ ] a list of entity types to redact from the
        analytics output. Must be set statically.
      },
      "InFlightChatRedactionConfiguration": { // In-flight redaction for chat
      messages as they are sent. Only available for chat channel.
        "Enabled": either "True" or "False". Must be set statically. Determines
        whether to redact sensitive data in real-time chat messages.
        "RedactionMaskMode": either "EntityType" or "PII". Must be set
        statically.
        "RedactionEntities": [ ] a list of entity types to redact from chat
        messages in-flight. Must be set statically.
        "DeliverUnprocessedMessages": either "True" or "False". Must be set
        statically. Determines whether to deliver messages that fail redaction.
      },
      "AnalyticsModes": ["ContactLens"]. Can only be set to ContactLens for chat
      channel.
      "SentimentConfiguration": { // Optional parameter to enable sentiment
      analysis
        "Enabled": either "True" or "False". Must be set statically.
      },
      "SummaryConfiguration": { // Optional parameter to enable post-contact
      summary
        "SummaryModes": ["PostContact", "AutomatedInteraction"]. Can be set to
        "PostContact" or "AutomatedInteraction", or both for chat channel.
      }
    }
  }
}
```

```

    }
  }
},

"VoiceBehavior": { // Configuration for voice channel interactions
  "VoiceRecordingBehavior": {
    "RecordedParticipants": [ ] a list of participants to record, chosen from
"Agent" and "Customer". An empty list disables recording. Must be set statically.
    "IVRRecordingBehavior": Can be either "Enabled" or "Disabled". Must be set
statically.
  },
  "VoiceAnalyticsBehavior": { // An object that holds voice analytics settings.
Can only be set if RecordedParticipants contains both Agent and Customer.
    "Enabled": either "True" or "False". Must be set statically.
    "AnalyticsLanguage": Must be one of languages supported by Contact Lens
analysis. Must be set statically. Use the format xx-XX, for example, en-US for US
English.
    "ConversationalAnalyticsRedactionConfiguration": { // Redaction settings
for analytics output (transcripts, audio files)
      "Enabled": either "True" or "False". Must be set statically. Determines
whether to redact sensitive data in Contact Lens output.
      "RedactionResults": either "RedactedAndOriginal" or "RedactedOnly".
Can be set dynamically. Determines whether to provide both versions or only redacted
output.
      "RedactionMaskMode": either "EntityType" or "PII". Must be set
statically.
      "RedactionEntities": [ ] a list of entity types to redact from the
analytics output. Must be set statically.
    },
    "AnalyticsModes": ["RealTime", "AutomatedInteraction"]. Can be set to one
of "RealTime" or "PostContact" and also can include "AutomatedInteraction" for voice
channel.
    "SentimentConfiguration": { // Optional parameter to enable sentiment
analysis
      "Enabled": either "True" or "False". Must be set statically.
    },
    "SummaryConfiguration": { // Optional parameter to enable post-contact
summary
      "SummaryModes": ["PostContact", "AutomatedInteraction"]. Can be set
to "PostContact" or "AutomatedInteraction", or both for voice channel. Must be set
statically.
    }
  }
},

```

```

    "ScreenRecordingBehavior": { // Optional configuration for agent screen recording.
    Can be defined independently or alongside any channel behavior.
        "ScreenRecordedParticipants": [ ] a list of participants for which to record
        their screen, can only include "Agent". An empty list disables screen recording. Must
        be set statically.
    }
}

```

Notes

- **AnalyticsRedactionMaskMode:** Optional, String. Allowed values:
 - **PII:** All PII data is replaced with [PII]. For example, Jane Doe is replaced with [PII]
 - **EntityType:** Each PII entity is replaced with its type. For example, Jane Doe is replaced with [NAME].
 - If no value is provided, the default PII is used.
- **AnalyticsRedactionEntities:** Optional, Array of strings.
 - Valid values include: "BANK_ACCOUNT_NUMBER", "BANK_ROUTING", "CREDIT_DEBIT_NUMBER", "CREDIT_DEBIT_CVV", "CREDIT_DEBIT_EXPIRY", "INTERNATIONAL_BANK_ACCOUNT_NUMBER", "PIN", "SWIFT_CODE", "CA_HEALTH_NUMBER", "UK_NATIONAL_HEALTH_SERVICE_NUMBER", "CA_SOCIAL_INSURANCE_NUMBER", "SSN", "UK_NATIONAL_INSURANCE_NUMBER", "PASSPORT_NUMBER", "DRIVER_ID", "IN_AADHAAR", "NAME", "AGE", "EMAIL", "PHONE", "ADDRESS", "US_INDIVIDUAL_TAX_IDENTIFICATION_NUMBER", "UK_UNIQUE_TAXPAYER_REFERENCE_NUMBER", "IN_PERMANENT_ACCOUNT_NUMBER", "IN_NREGA", "AWS_ACCESS_KEY", "AWS_SECRET_KEY", "IP_ADDRESS", "MAC_ADDRESS", "PASSWORD", "URL", "USERNAME", "LICENSE_PLATE", "VEHICLE_IDENTIFICATION_NUMBER", "IN_VOTER_NUMBER", "DATE_TIME", "AGENT_DISPLAY_NAME", "CUSTOMER_DISPLAY_NAME", "ATTACHMENT_NAME".
 - An empty array is not allowed.
 - If **AnalyticsRedactionEntities** is not present, the default "redact all PII data" is used.
 - The following redaction entities are not supported for chat in-flight redaction:
 - IN_PERMANENT_ACCOUNT_NUMBER
 - IN_NREGA
 - IN_VOTER_NUMBER
 - IN_AADHAAR

- DATE_TIME
- CUSTOMER_DISPLAY_NAME
- AGENT_DISPLAY_NAME
- ATTACHMENT_NAME

For more information on sensitive data redaction, see [Enable redaction of sensitive data](#) in the *Connect Customer Administrator's Guide*.

For more information on in-flight chat redaction, see [Enable in-flight sensitive data redaction and message processing](#).

For a list of languages supported by Contact Lens post-call analysis, see [Contact Lens supported languages](#). For the 4-character language code to use, see [Supported languages](#) in the *Amazon Transcribe Developer Guide*.

Results and conditions

None.

Errors

- **NoMatchingError** - if no other Error matches. Must always be defined.
- **ChannelMismatch** - if the media channel that initiated the contact is not the same as the one defined in the action. For screen recording, any channel other than voice, chat or tasks would result in this branch being taken. Must always be defined.
- **InFlightRedactionConfigurationFailed** - if starting/stopping in-flight chat redaction fails. Must be defined if chat behavior is defined in action.

Corresponding block in the UI

[Set recording, analytics and processing behavior](#)

UpdateContactRecordingBehavior

Sets contact recording behavior, including analysis behavior and which participants of the contact to record.

Parameter object

```
{
  "RecordingBehavior": { an object that holds the recording behavior
    "RecordedParticipants": [ ] a list of participants to record, chosen from
    "Agent" and "Customer". An empty list disables recording. Must be set statically.
    "ScreenRecordedParticipants": [ ] a list of participants for which to record
    their screen, can only include "Agent". Must be set statically.
    "IVRRecordingBehavior": Can be either "Enabled" or "Disabled". Must be set
    statically.
  },
  "AnalyticsBehavior": { an object that holds the analytics behavior. Can only be set
  if the RecordedParticipants contains both Agent and Customer
    "Enabled": either "True" or "False". Must be set statically.
    "AnalyticsLanguage": Must be one of languages supported by Contact Lens post-
    call analysis. Must be set statically. Use the format xx-XX, for example, en-US for US
    English.
    "AnalyticsRedactionBehavior": either Enabled or Disabled. Defaults to Disabled
    if not set. Determines whether to redact sensitive data, such as personal information,
    in the Contact Lens output file and audio recording.
    "AnalyticsRedactionResults": either "RedactedAndOriginal" or "RedactedOnly".
    Can be set dynamically. Determines whether the customer gets both the redacted and
    the original transcripts and audio files, or just the redacted transcripts and audio
    files.
    "AnalyticsRedactionMaskMode": either "EntityType" or "PII". Must be set
    statically.
    "AnalyticsRedactionEntities": ["BANK_ACCOUNT_NUMBER", "BANK_ROUTING",
    "CREDIT_DEBIT_NUMBER", "CREDIT_DEBIT_CVV", "CREDIT_DEBIT_EXPIRY",
    "INTERNATIONAL_BANK_ACCOUNT_NUMBER", "PIN", "SWIFT_CODE", "CA_HEALTH_NUMBER",
    "UK_NATIONAL_HEALTH_SERVICE_NUMBER", "CA_SOCIAL_INSURANCE_NUMBER", "SSN",
    "UK_NATIONAL_INSURANCE_NUMBER", "PASSPORT_NUMBER", "DRIVER_ID", "IN_AADHAAR", "NAME",
    "AGE", "EMAIL", "PHONE", "ADDRESS", "US_INDIVIDUAL_TAX_IDENTIFICATION_NUMBER",
    "UK_UNIQUE_TAXPAYER_REFERENCE_NUMBER", "IN_PERMANENT_ACCOUNT_NUMBER", "IN_NREGA",
    "AWS_ACCESS_KEY", "AWS_SECRET_KEY", "IP_ADDRESS", "MAC_ADDRESS", "PASSWORD", "URL",
    "USERNAME", "LICENSE_PLATE", "VEHICLE_IDENTIFICATION_NUMBER", "IN_VOTER_NUMBER",
    "DATE_TIME", "AGENT_DISPLAY_NAME", "CUSTOMER_DISPLAY_NAME", "ATTACHMENT_NAME"],
    "ChannelConfiguration": {
      "Chat": {
        "AnalyticsModes": ["ContactLens"] Can only be set to ContactLens
      },
      "Voice": {
        "AnalyticsModes": ["RealTime"] Can be set to RealTime and PostContact
      }
    }
  },
},
```

```

    "SummaryConfiguration": {
        "SummaryModes": ["PostContact"] Optional parameter to enable post-contact
summary. At present we only support "PostContact".
    },
    "SentimentConfiguration": {
        "Enabled": either "True" or "False". Must be set statically.
    }
}
}

```

Notes

- **AnalyticsRedactionMaskMode:** Optional, String. Allowed values:
 - **PII:** All PII data is replaced with [PII]. For example, Jane Doe is replaced with [PII]
 - **EntityType:** Each PII entity is replaced with its type. For example, Jane Doe is replaced with [NAME].
 - If no value is provided, the default PII is used.
- **AnalyticsRedactionEntities:** Optional, Array of strings.
 - Valid values include: "BANK_ACCOUNT_NUMBER", "BANK_ROUTING", "CREDIT_DEBIT_NUMBER", "CREDIT_DEBIT_CVV", "CREDIT_DEBIT_EXPIRY", "INTERNATIONAL_BANK_ACCOUNT_NUMBER", "PIN", "SWIFT_CODE", "CA_HEALTH_NUMBER", "UK_NATIONAL_HEALTH_SERVICE_NUMBER", "CA_SOCIAL_INSURANCE_NUMBER", "SSN", "UK_NATIONAL_INSURANCE_NUMBER", "PASSPORT_NUMBER", "DRIVER_ID", "IN_AADHAAR", "NAME", "AGE", "EMAIL", "PHONE", "ADDRESS", "US_INDIVIDUAL_TAX_IDENTIFICATION_NUMBER", "UK_UNIQUE_TAXPAYER_REFERENCE_NUMBER", "IN_PERMANENT_ACCOUNT_NUMBER", "IN_NREGA", "AWS_ACCESS_KEY", "AWS_SECRET_KEY", "IP_ADDRESS", "MAC_ADDRESS", "PASSWORD", "URL", "USERNAME", "LICENSE_PLATE", "VEHICLE_IDENTIFICATION_NUMBER", "IN_VOTER_NUMBER", "DATE_TIME", "AGENT_DISPLAY_NAME", "CUSTOMER_DISPLAY_NAME", "ATTACHMENT_NAME".
 - An empty array is not allowed.
 - If **AnalyticsRedactionEntities** is not present, the default "redact all PII data" is used.

For more information on sensitive data redaction, see [Enable redaction of sensitive data](#) in the *Connect Customer Administrator's Guide*.

For a list of languages supported by Contact Lens post-call analysis, see [Contact Lens supported languages](#). For the 4-character language code to use, see [Supported languages](#) in the *Amazon Transcribe Developer Guide*.

Results and conditions

None.

Errors

None.

Restrictions

This is supported only in contact flows, transfer flows, outbound whispers, and customer queue flows. This is not supported in agent/customer whispers or hold flows.

Analytics is only supported by the voice channel.

Corresponding block in the UI

[Set recording and analytics behavior](#)

ResumeContact

Resumes a contact from a paused state.

Parameter object

No parameters are expected.

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

None. This action can be used everywhere.

UpdateContactRoutingBehavior

Updates the contact's routing details. This can move the contact forward or backward in queue, or specify a queue priority.

Parameter object

```
{
  "QueuePriority": An integer that represents the queue priority to be applied to
the contact (lower priorities are routed preferentially). Cannot be specified if
QueueTimeAdjustmentSeconds is specified. Must be statically defined, must be larger
than zero, and a valid integer value.
  "QueueTimeAdjustmentSeconds": An integer that represents the queue time adjust
to be applied to the contact, in seconds (longer / larger queue times are routed
preferentially). Cannot be specified if QueuePriority is specified. Must be statically
defined and a valid integer value.
}
```

Results and conditions

None.

Errors

None.

Restrictions

This is supported only in inbound contact flows. It is not supported in transfer flows, whisper flows, customer queue flows, or hold flows.

Corresponding block in the UI

[Change routing priority / age](#)

UpdateContactTargetQueue

Sets the contact's TargetQueue. This is the queue is used by all other instructions that check a queue implicitly, and for TransferContactToQueue.

Parameter object

```
{
```

```
"QueueId": [Optional] A queue ID or queue ARN. If AgentId is specified, this may
not be specified. This must be either defined fully statically or as a single valid
JSONPath identifier.
"AgentId": [Optional] An agent ID or agent ARN, representing an agent queue. If
QueueId is specified, this may not be specified. This must be either defined fully
statically or as a single valid JSONPath identifier.
}
```

Results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is supported only in inbound contact flows and transfer flows. It is not supported in whisper flows, hold flows, or customer queue flows.

Corresponding block in the UI

[Set working queue](#)

UpdateContactTextToSpeechVoice

Updates the Amazon Polly voice used by text-to-speech for voice contacts (message with text-to-speech, or Amazon Lex bots). This defaults to Joanna if this action is never run.

Parameter object

```
{
  "TextToSpeechVoice": A string holding the name of an Amazon Polly voice. May be
defined statically or dynamically. If this is an invalid text to speech voice, text to
speech is no longer function for this contact.
  "TextToSpeechEngine": The engine associated with the Amazon Polly voice. May be
defined statically or dynamically.
  "TextToSpeechStyle" : The speech style associated with the Amazon Polly Voice. It
could be None, Conversational, or Newscaster. May be defined statically or dynamically.
```

```
}
```

Results and conditions

Results in error if voice or engine are invalid, or if the selected voice does not support the selected engine.

Errors

- NoMatchingError - if no other Error matches. Must always be defined.

Restrictions

None. This action is supported in all flow types, and across all channels.

Corresponding block in the UI

[Set voice](#)

UpdatePreviousContactParticipantState

This action is primarily used to prevent previous participants on the contact from observing the contact. Common use cases are disconnecting the agent that initiates a transfer when they transfer a contact to a secure destination, or putting the agent on hold when transferring to a quick connect that securely gathers customer input such as credit card numbers.

Parameter object

```
{
  "PreviousContactParticipantState": One of ["AgentOnHold", "CustomerOnHold",
  "OffHold"], which are only supported for voice contacts.
}
```

Execution results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is supported only in inbound contact flows and transfer flows.

Corresponding block in the UI

[Hold customer or agent](#)

Flow control actions in the Connect Customer Flow language

These actions don't have any side effects and are only used to determine the path through a flow. Certain data may not be available (such as contact data, if the action is determining its path based on contact data). These actions generally work in every circumstance.

A flow control action is an action that:

- Does not need a contact or a participant to succeed.
- Controls the behavior of the flow, by either enabling or disabling flow behavior (such as logging) or by choosing a branch when the flow runs.

Contents

- [CheckHoursOfOperation](#)
- [CheckMetricData](#)
- [CheckOutboundCallStatus](#)
- [CheckVoiceld](#)
- [Compare](#)
- [DistributeByPercentage](#)
- [EndFlowExecution](#)
- [GetMetricData](#)
- [Loop](#)
- [StartVoiceldStream](#)
- [TransferToFlow](#)
- [UpdateFlowAttributes](#)
- [UpdateFlowLoggingBehavior](#)
- [UpdateRoutingCriteria](#)
- [Wait](#)

CheckHoursOfOperation

Returns whether the specified hours of operation object (or the hours of operation object associated with the current queue if no hours of operation is referenced) is in hours or out of hours as its result, allowing comparisons against it.

Parameter object

```
{
  "HoursOfOperationId": [Optional] An hours of operation ID or hours of operation ARN.
  *Must be either fully static or fully dynamic*. If not specified, the TargetQueue's
  hours of operation for the contact are used
}
```

Results and conditions

True or **False** based on whether the hours of operation object specified is in hours or out of hours. There must be a Condition provided for Equals **True** and a Condition for Equals **False**, and no other conditions.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is available in inbound flows, transfer flows, and customer queue flows. It is not available to hold flows or to whisper flows.

Corresponding block in the UI

[Check hours of operation](#)

CheckMetricData

A shortcut single action to avoid using GetMetricData and Compare for a set of simple metrics. This action loads the specified metric data for the specified queue, and allows comparisons to the loaded value. For example, it loads number of contacts in queue, age of oldest contact in queue, number of agents staffed on the queue, number of agents available on the queue, or number of agents online on the queue.

Parameter object

```
{
  "MetricType": One of [NumberOfAgentsAvailable, NumberOfAgentsStaffed,
    NumberOfAgentsOnline, OldestContactInQueueAgeSeconds, NumberOfContactsInQueue].
  **Dynamic values are not supported**,
  "QueueId": [Optional] A queue ID or queue ARN. If AgentId is specified, this may not
    be specified. *Dynamic values are supported*,
  "AgentId": [Optional] An agent ID or agent ARN, representing an agent queue. If
    QueueId is specified, this may not be specified. *Dynamic values are supported*. If
    neither this nor QueueId are specified, the contact TargetQueue is used
}
```

Execution results and conditions

A number, representing the value of the metric that was requested. This can be used for conditions. If the MetricType is NumberOfAgents* then the only supported condition is "NumberGreaterThan 0", otherwise Equals and any Number* Operands are allowed.

Errors

- NoMatchingError - if no other Error matches.
- NoMatchingCondition - if no other Condition matches (only supported if the MetricType is OldestContactInQueueAgeSeconds or NumberOfContactsInQueue).

Restrictions

This action is only usable in flows, queue and agent transfers, and customer queue flows. It is not available in any type of whisper or hold flows.

Corresponding block in the UI

- [Check staffing](#)
- [Check queue status](#)

CheckOutboundCallStatus

Engages with the output provided by an answering machine, and provides branches to route the contact accordingly.

Parameter object

```
{  
  
}
```

Execution results and conditions

- "CallAnswered" if the call has been answered by a person.
- "VoicemailBeep" if Connect Customer identifies that the call ended in a voice mail and it detects a beep.
- "VoicemailNoBeep" if Connect Customer identifies that call ended in a voicemail, but it doesn't detect a beep, or the beep is unknown.
- "NotDetected" if Connect Customer could not detect whether there is a voicemail. This happens when Connect Customer is unable to make a positive determination of whether a call was answered by a live voice or an answering machine. Typical situations that result in this state include long silences or excessive background noise.

Conditions are supported, but only the "Equals" operator is supported. "CallAnswered", "VoicemailBeep", "VoicemailNoBeep" and "NotDetected" are the only supported operands.

Errors

- NoMatchingError if no condition matches.

Restrictions

This action works with [Connect Customer outbound campaigns](#) only.

Corresponding block in the UI

[Check call progress](#)

CheckVoicemail

Checks the enrollment status, voice authentication or fraud detection results of the voice analysis returned by Voice ID.

Parameter object

```
{
  "CheckVoiceIdOption": "enrollmentStatus"
}
```

```
{
  "CheckVoiceIdOption": "voiceAuthentication"
}
```

```
{
  "CheckVoiceIdOption": "fraudDetection"
}
```

Execution results and conditions

The `checkVoiceId` action returns results of the voice analysis and the status returned by Voice ID. The following is returned when `CheckVoiceIdOption` input is set as:

- **enrollmentStatus:**
 - **Enrolled:** The caller is enrolled in voice authentication.
 - **Not enrolled:** The caller has not yet been enrolled in voice authentication. When this status is returned, for example, you may want to directly route the call to an agent for enrollment.
 - **Opted out:** The caller has opted out of voice authentication.

You are not charged for checking enrollment status.

- **voiceAuthentication:**
 - **Authenticated:** The caller's identity has been verified. That is, the authentication score is greater than or equal to the threshold (default threshold of 90 or your custom threshold).
 - **Not authenticated:** The authentication score is lower than threshold that you configured.
 - **Inconclusive:** Unable to analyze a caller's speech for authentication. This is usually because Voice ID did not get the required 10 seconds to provide a result for authentication.
 - **Not enrolled:** The caller has not yet been enrolled in voice authentication. When this status is returned, for example, you may want to directly route the call to an agent for enrollment.
 - **Opted out:** The caller has opted out of voice authentication.

You are not charged if the result is **Inconclusive**, **Not enrolled** or **Opted out**.

- **fraudDetection:**

- **High risk:** The risk score meets or exceeds the set threshold.
- **Low risk:** The risk score did not meet the set threshold.
- **Inconclusive:** Unable to analyze a caller's voice for detection of fraudsters in a watchlist.

Errors

- NoMatchingError if no condition matches.

Restrictions

Only supported for voice channel. If used with the chat or task channels, the action takes the **Error** branch.

Corresponding block in the UI

[Check Voice ID](#)

Compare

Allows comparisons against the specified value.

Parameter object

```
{
  "ComparisonValue": Any **single** JSONPath identifier that is valid for the flow data
  object
}
```

Execution results and conditions

The value specified for comparison. This can be used for conditions.

Errors

- NoMatchingCondition - if no other Condition matches.

Restrictions

This action is available in every type of flow.

Corresponding block in the UI

[Check contact attributes](#)

DistributeByPercentage

Returns a random number between 1 and 100 (inclusive) as its result, allowing comparisons against it.

Parameter object

```
{  
  
}
```

Results and conditions

A number between 1 and 100, inclusive, chosen randomly. Comparisons are supported, but they must be a chain of NumericLessThan comparisons, with each subsequent comparison checking the previous value, plus the percentage that is desired to go down this next action, and no Comparison comparing a value larger than 100.

Errors

- NoMatchingCondition if no Condition matches. This is the default option in the flow editor.

Restrictions

This action is available in inbound flows, transfer flows, and customer queue flows. It is not available to hold flows or to whisper flows.

Corresponding block in the UI

[Distribute by percentage](#)

EndFlowExecution

Finishes flow, but does not explicitly disconnect the participant. The participant may be disconnected by contact logic after this. For example, if a flow ends before the contact is put into queue, ending the flow results in the contact being ended.

Parameter object

```
{  
  
}
```

Results and conditions

None. No conditions are supported.

Errors

None. This is always a terminal action.

Restrictions

This action is available only in whisper flows and customer queue flows. It is not available in flows, hold flows, or transfer flows.

Corresponding block in the UI

[End flow / Resume](#)

GetMetricData

Loads real time queue metrics for the queue specified by queue ID, agent ID (for agent queues), or the target queue, and makes them available on the flow run data. May be extended in the future to allow getting historical metric data in addition to current metric data, and to getting agent metrics in addition to queue metrics.

Parameter object

```
{  
  "QueueId": [Optional] A queue ID or queue ARN. If AgentId is specified, this may not  
  be specified. *Dynamic values are supported*,  
  "AgentId": [Optional] An agent ID or agent ARN, representing an agent queue. If  
  QueueId is specified, this may not be specified. *Dynamic values are supported*  
  "QueueChannel": [Optional] Either "Voice" or "Chat". Can be set dynamically.  
  Determines the channel for which metrics are returned. If not specified, metrics are  
  returned for all channels.  
}
```

Execution results and conditions

None. No conditions are supported.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is available in every type of flow.

Corresponding block in the UI

[Get queue metrics](#)

Loop

When the same action (the same Action Identifier) is run multiple times, this block returns a result of "NotDone" a number of times equal to the specified loop count, then "Done" once, then reset.

Parameter object

```
{
  "LoopCount": Number of times to loop, must be between 0 and 100 (inclusive). Must
  either be fully static or fully dynamic.
}
```

Execution results and conditions

"ContinueLooping" if the loop should continue. "DoneLooping" if the loop should finish. Conditions are supported, there must be a Condition provided for Equals ContinueLooping and for Equals DoneLooping, and no other Conditions can be specified.

Errors

None.

Restrictions

This is supported in every type of flow.

Corresponding block in the UI

[Loop](#)

StartVoiceIdStream

Sends audio to Connect Customer Voice ID to verify the caller's identity and match against fraudsters in watchlist, as soon as the call is connected to a flow.

Parameter object

```
{  
  
}
```

Execution results and conditions

None. No conditions are supported.

Errors

- NoMatchingError if no condition matches.

Restrictions

Only supported for the voice channel. If used with the chat or task channels, the action takes the **Error** branch. Not supported in hold flows.

Corresponding block in the UI

[Set Voice ID](#)

TransferToFlow

Execution jumps to a different flow, and continues running at that flow's beginning.

Parameter object

```
{  
  "ContactFlowId": A flow ID or flow ARN. *Must be either fully static or a single  
  valid JSONPath identifier*
```

```
}
```

Execution results and conditions

None.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is available in inbound flows and transfer flows. It is not available to hold flows, customer queue flows, or whisper flows.

Corresponding block in the UI

[Transfer to flow](#)

UpdateFlowAttributes

Sets a collection of attributes on the current flow. These attributes are not carried over to the subsequent flows. With this type of operation, either all attributes are set or none are set.

Parameter object

```
{
  "FlowAttributes": { An Object that holds the attributes to be set. Keys are of
    type String, Values are of type FlowAttribute
    "Type": {
      FlowAttribute" : "Value"
    }
  }
}
```

Results and conditions

None. No conditions are supported.

Errors

None.

Restrictions

This action is supported on all channels and in all flow types.

Corresponding block in the UI

[Set contact attributes](#)

UpdateFlowLoggingBehavior

Enables or disables flow logging. If this is a flow, this same behavior remains unless it is overridden for the rest of the contact segment. It is also automatically inherited by new segments in the chain.

Parameter object

```
{
  "FlowLoggingBehavior": One of [Enabled,Disabled]. *Dynamic values are not supported*
}
```

Results and conditions

None. No conditions are supported.

Errors

None.

Restrictions

This action is available in every type of flow.

Corresponding block in the UI

[Set logging behavior](#)

UpdateRoutingCriteria

Sets the routing criteria for the contact.

Parameter object

```
{
  "RoutingCriteria": { Required. RoutingCriteria is a JSON object.
    "Steps": [{ Required. List of routing steps. When Connect Customer not
      find an available agent meeting the requirements in a step for a given step duration,
```

the routing criteria will move on to the next step sequentially until a join is completed with an agent. When all steps are exhausted, the contact will be offered to any agent in the queue. May be set statically or dynamically

"Expression": { Required. An object to specify the expression of a routing step. It is a tagged union to specify expression for a routing step.

"AttributeCondition": {An object to specify the predefined attribute condition.

"Name": The name of predefined attribute. It is a string and has length constraints between 1-64.

"Value": The value of predefined attribute. It is a string and has length constraints between 1-64.

"ProficiencyLevel": The proficiency level of the condition. It is a float value. Valid values are: 1.0, 2.0, 3.0, 4.0 and 5.0

"ComparisonOperator": The operator of the condition. It is a string. Valid values: NumberGreaterOrEqualTo

}

"AndExpression": [List of routing expressions (attribute conditions) which will be AND-ed together.]

},

"Expiry": { An object to specify the expiration of a routing step.

"DurationInSeconds": The number of seconds to wait before expiring the routing step. Can be set only statically

}

}]

}

}

Results and conditions

None. No conditions are supported.

Errors

NoMatchingError - if no other Error matches.

Restrictions

This action is supported on all channels and in only in Inbound flow, Customer Queue flow, Transfer to Agent flow, and Transfer to Queue flow types.

Corresponding block in the UI

[Set routing criteria](#)

Wait

Pauses the flow for a specified duration, or until a specified event happens, whichever happens first.

Parameter object

```
{
  "TimeoutSeconds": The amount of time to wait before the action finishes with the
  "WaitCompleted" result. This can be either statically defined, or a single valid
  JSONPath identifier. If defined statically, this must be a positive integer value no
  greater than 604800 (seven days),
  "Events": An optional list of all events that can trigger an interrupt. The
  supported events currently are "CustomerReturned" and "BotParticipantDisconnected".
  This must be defined statically.
}
```

Execution results and conditions

If an event interrupts the wait, the run result is the event that interrupted. If no event interrupts the Wait and the time elapses, the run result is WaitCompleted. Conditions are supported, but only the "Equals" operator is supported. "WaitCompleted" is always required operand, and every specified event is also required to be present as a condition operand.

Errors

- NoMatchingError - If no other Error matches.
- ParticipantNotFound - The supported event currently is "BotParticipantDisconnected".

Restrictions

This is supported in every type of flow, but is supported only by the chat channel.

Corresponding block in the UI

[Wait](#)

Interactions in the Connect Customer Flow language

Interactions actions have side effects, but they don't require a contact or a participant. They include actions such as invoking an AWS Lambda function. They generally work in every circumstance.

Contents

- [AssociateContactToCustomerProfile](#)
- [CreateCallbackContact](#)
- [CreateCustomerProfile](#)
- [InvokeLambdaFunction](#)
- [GetCustomerProfile](#)
- [GetCustomerProfileObject](#)
- [GetCalculatedAttributesForCustomerProfile](#)
- [UpdateCustomerProfile](#)

AssociateContactToCustomerProfile

Associate a contact to a customer profile. Customer Profiles must be enabled for your Connect Customer instance.

See [AddProfileKey](#) in the *Connect Customer Customer Profiles API Reference*.

Parameter object

A ProfileId and ContactId must be present.

```
{
  "ProfileRequestData": {
    "ProfileId": Profile being associated
    "ContactId": ContactId being associated
  },
  "ProfileResponseData": {
    None.
  }
}
```

Results and conditions

None. Conditions are not supported.

Errors

- NoMatchingError - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

CreateCallbackContact

Creates a new callback contact. If no customer number is specified, and this is run in context of a contact, the contact's CustomerCallbackNumber is used as the customer number. If you specify a ContactFlowId, then InitialCallDelaySeconds parameter is ignored.

Parameter object

```
{
  "QueueId": [Optional] A queue ID or queue ARN. The callback contact is routed with
  this queue, or if this is not specified, the contact's current TargetQueue. Must be
  specified fully statically or with a single valid JSONPath identifier.
  "AgentId": [Optional] An agent ID or agent ARN, representing an agent queue. If
  QueueId is specified, this may not be specified. This must be either defined fully
  statically or as a single valid JSONPath identifier.
  "InitialCallDelaySeconds": The amount of time, in seconds, to wait at a minimum
  before routing the callback contact. This gives the customer enough time to end their
  existing contact before being called back. Must be larger than 0, no greater than
  259,200 (three days), and an integer. Must be defined statically.
  "MaximumConnectionAttempts": The number of attempts at a maximum to connect this
  contact to a customer, if the callback is not answered. Must be larger than zero, and
  an integer. Must be defined statically.
  "RetryDelaySeconds": The minimum amount of time to wait, in seconds, between
  an unanswered callback attempt is made and the next attempt to reach the customer.
  Must be larger than 0, no greater than 259,200 (three days), and an integer. Must be
  defined statically.
  "ContactFlowId": [Optional] A contactFlow ID or contactFlow ARN. Callback contact
  created will execute this flow post creation, if specified. This must be either
  defined fully statically or as a single valid JSONPath identifier.
  "CallerId": [Optional] A caller ID representing the phone number to use for
  the callback. This is what the customer sees when dialed. Must be a valid phone
  number claimed in your Connect Customer instance. This must be either defined fully
  statically or as a single valid JSONPath identifier.
}
```

Results and conditions

None. No conditions are supported.

Errors

- `NoMatchingError` - if no other Error matches.

Restrictions

This action is supported in contact flows, transfer flows, and customer queue flows. It is not supported in whisper flows or hold flows.

Corresponding block in the UI

[Set callback number](#)

CreateCustomerProfile

Create a customer profile. Customer Profiles must be enabled for your Connect Customer instance.

See [CreateProfile](#) in the *Connect Customer Customer Profiles API Reference*.

Parameter object

```
{
  "ProfileRequestData": {
    All of these fields are optional.
    "FirstName",
    "MiddleName",
    "LastName",
    "PhoneNumber",
    "EmailAddress",
    "AccountNumber",
    "AdditionalInformation",
    "PartyType",
    "BusinessName",
    "BirthDate",
    "Gender",
    "MobilePhoneNumber",
    "HomePhoneNumber",
    "BusinessPhoneNumber",
    "BusinessEmailAddress",
    "Address1",
    "Address2",
    "Address3",
    "Address4",
```

```

    "City",
    "County",
    "Country",
    "PostalCode",
    "Province",
    "State",
    "ShippingAddress1",
    "ShippingAddress2",
    "ShippingAddress3",
    "ShippingAddress4",
    "ShippingCity",
    "ShippingCounty",
    "ShippingCountry",
    "ShippingPostalCode",
    "ShippingProvince",
    "ShippingState",
    "MailingAddress1",
    "MailingAddress2",
    "MailingAddress3",
    "MailingAddress4",
    "MailingCity",
    "MailingCounty",
    "MailingCountry",
    "MailingPostalCode",
    "MailingProvince",
    "MailingState",
    "BillingAddress1",
    "BillingAddress2",
    "BillingAddress3",
    "BillingAddress4",
    "BillingCity",
    "BillingCounty",
    "BillingCountry",
    "BillingPostalCode",
    "BillingProvince",
    "BillingState",
    "Attributes.x"
  },
  "ProfileResponseData": {
    All of these fields are optional.
    Newly created profile ID is persisted under the Customer -> ProfileID attribute
+ $.Customer.ProfileId
    "FirstName",
    "MiddleName",

```

```
"LastName",
"PhoneNumber",
"EmailAddress",
"AccountNumber",
"AdditionalInformation",
"PartyType",
"BusinessName",
"BirthDate",
"Gender",
"MobilePhoneNumber",
"HomePhoneNumber",
"BusinessPhoneNumber",
"BusinessEmailAddress",
"Address1",
"Address2",
"Address3",
"Address4",
"City",
"County",
"Country",
"PostalCode",
"Province",
"State",
"ShippingAddress1",
"ShippingAddress2",
"ShippingAddress3",
"ShippingAddress4",
"ShippingCity",
"ShippingCounty",
"ShippingCountry",
"ShippingPostalCode",
"ShippingProvince",
"ShippingState",
"MailingAddress1",
"MailingAddress2",
"MailingAddress3",
"MailingAddress4",
"MailingCity",
"MailingCounty",
"MailingCountry",
"MailingPostalCode",
"MailingProvince",
"MailingState",
"BillingAddress1",
```

```

    "BillingAddress2",
    "BillingAddress3",
    "BillingAddress4",
    "BillingCity",
    "BillingCounty",
    "BillingCountry",
    "BillingPostalCode",
    "BillingProvince",
    "BillingState",
    "Attributes.x"
  }

```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.Customer path based on the attributes included in ProfileResponseData.

Errors

- NoMatchingError - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

InvokeLambdaFunction

Invokes an AWS Lambda function with a collection of optional parameters. This AWS Lambda function is also given a copy of the flow run data if there is an associated contact with the flow.

Parameter object

```

{
  "LambdaFunctionARN": The ARN of the AWS Lambda function to be invoked. May be
    defined statically or dynamically.
  "InvocationTimeLimitSeconds": The number of seconds to wait for a response from the
    AWS Lambda function. Must be greater than 0, no larger than 8, and an integer. Must be
    set statically.
  "InvocationType": Specifies the invocation type, allowed values: "SYNCHRONOUS" |
    "ASYNCHRONOUS".
  "LambdaInvocationAttributes" { A map of additional data to send to the AWS Lambda
    function when invoking it. Keys and values may be set statically or dynamically.

```

```

    }
    "ResponseValidation": {
      "ResponseType": "STRING_MAP" or "JSON", If response validation is set to
      STRING_MAP, then the Lambda function should return a flat object of key/value pairs of
      the string type. Otherwise, if response validation is set to JSON, the Lambda function
      can return any valid JSON, including nested JSON. Must be set statically.
    }
  }
}

```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.External path.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

None. This action is supported by all channels and in all types of flows.

Corresponding block in the UI

[AWS Lambda function](#)

GetCustomerProfile

Retrieve a customer profile based any search identifier, up to five total. Customer Profiles must be enabled for your Connect Customer instance.

See [SearchProfiles](#) in the *Connect Customer Customer Profiles API Reference*.

Parameter object

At least one search identifier must be present.

```

{
  "ProfileRequestData": {
    Requires either IdentifierName and IdentifierValue, or SearchCriteria.

    "IdentifierName": Name to search for profiles with one identifier,
    "IdentifierValue": Value to search for profiles with one identifier,
  }
}

```

```

    "SearchCriteria": [
      {
        "IdentifierName": Name to search for profiles with multiple
identifiers,
        "IdentifierValue": Value to search for profiles with multiple
identifiers
      }
    ],
    Required when using SearchCriteria
    "LogicalOperator": AND or OR

  },
  "ProfileResponseData": {
    All of these fields are optional.
    Newly created profile ID is persisted under the Customer -> ProfileID attribute
+ $.Customer.ProfileId
    "FirstName",
    "MiddleName",
    "LastName",
    "PhoneNumber",
    "EmailAddress",
    "AccountNumber",
    "AdditionalInformation",
    "PartyType",
    "BusinessName",
    "BirthDate",
    "Gender",
    "MobilePhoneNumber",
    "HomePhoneNumber",
    "BusinessPhoneNumber",
    "BusinessEmailAddress",
    "Address1",
    "Address2",
    "Address3",
    "Address4",
    "City",
    "County",
    "Country",
    "PostalCode",
    "Province",
    "State",
    "ShippingAddress1",
    "ShippingAddress2",
    "ShippingAddress3",

```

```
"ShippingAddress4",
"ShippingCity",
"ShippingCounty",
"ShippingCountry",
"ShippingPostalCode",
"ShippingProvince",
"ShippingState",
"MailingAddress1",
"MailingAddress2",
"MailingAddress3",
"MailingAddress4",
"MailingCity",
"MailingCounty",
"MailingCountry",
"MailingPostalCode",
"MailingProvince",
"MailingState",
"BillingAddress1",
"BillingAddress2",
"BillingAddress3",
"BillingAddress4",
"BillingCity",
"BillingCounty",
"BillingCountry",
"BillingPostalCode",
"BillingProvince",
"BillingState",
"Attributes.x"
}
```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.Customer path based on the attributes included in ProfileResponseData.

Errors

- **MultipleFoundError** - if multiple profiles were found for the associated profile search key.
- **NoneFoundError** - if no profiles were found for the associated profile search key.
- **NoMatchingError** - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

GetCustomerProfileObject

Retrieve a customer profile object of the desired type, based on recency or any search identifier. Customer Profiles must be enabled for your Connect Customer instance.

See [ListProfileObjects](#) in the *Connect Customer Customer Profiles API Reference*.

Parameter object

A `ProfileId` and `ObjectType` must be present. Either `UseLatest`, or `IdentifierName` and `IdentifierValue` must be present.

```
{
  "ProfileRequestData": {
    "ProfileId": Profile owning the object,
    "ObjectType": Type of object being retrieved,

    "IdentifierName": Optional name of search identifier,
    "IdentifierValue": Optional value of search identifier,
    "UseLatest": true / false
  },
  "ProfileResponseData": {
    All of these fields are optional.
    Asset ID, if a single asset is found, is always persisted under the Customer ->
    Asset -> AssetID attribute + $.Customer.Asset.AssetId
    Order ID, if a single order is found, is always persisted under the Customer ->
    Order -> OrderId attribute + $.Customer.Order.OrderId
    Case ID, if a single case is found, is always persisted under the Customer ->
    Case -> CaseID attribute + $.Customer.Case.CaseId
    "AssetAssetId",
    "AssetProfileId",
    "AssetAssetName",
    "AssetSerialNumber",
    "AssetModelNumber",
    "AssetModelName",
    "AssetProductSKU",
    "AssetPurchaseDate",
    "AssetUsageEndDate",
    "AssetStatus",
    "AssetPrice",
  }
}
```

```
"AssetQuantity",
"AssetDescription",
"AssetAdditionalInformation",
"AssetDataSource",
"AssetAttributes.x",
"OrderOrderId",
"OrderProfileId",
"OrderCustomerEmail",
"OrderCustomerPhone",
"OrderCreatedDate",
"OrderUpdatedDate",
"OrderProcessedDate",
"OrderClosedDate",
"OrderCancelledDate",
"OrderCancelReason",
"OrderName",
"OrderAdditionalInformation",
"OrderGateway",
"OrderStatus",
"OrderStatusCode",
"OrderStatusUrl",
"OrderCreditCardNumber",
"OrderCreditCardCompany",
"OrderFulfillmentStatus",
"OrderTotalPrice",
"OrderTotalTax",
"OrderTotalDiscounts",
"OrderTotalItemsPrice",
"OrderTotalShippingPrice",
"OrderTotalTipReceived",
"OrderCurrency",
"OrderTotalWeight",
"OrderBillingName",
"OrderBillingAddress1",
"OrderBillingAddress2",
"OrderBillingAddress3",
"OrderBillingAddress4",
"OrderBillingCity",
"OrderBillingCounty",
"OrderBillingCountry",
"OrderBillingPostalCode",
"OrderBillingProvince",
"OrderBillingState",
"OrderShippingName",
```

```
    "OrderShippingAddress1",
    "OrderShippingAddress2",
    "OrderShippingAddress3",
    "OrderShippingAddress4",
    "OrderShippingCity",
    "OrderShippingCounty",
    "OrderShippingCountry",
    "OrderShippingPostalCode",
    "OrderShippingProvince",
    "OrderShippingState",
    "OrderAttributes.x",
    "CaseCaseId",
    "CaseProfileId",
    "CaseTitle",
    "CaseSummary",
    "CaseStatus",
    "CaseReason",
    "CaseCreatedBy",
    "CaseCreatedDate",
    "CaseUpdatedDate",
    "CaseClosedDate",
    "CaseAdditionalInformation",
    "CaseDataSource",
    "CaseAttributes.x",
    "ObjectAttributes.x"
  }
}
```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.Customer path based on the attributes included in ProfileResponseData.

Errors

- NoneFoundError - if no profiles were found for the associated profile search key.
- NoMatchingError - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

GetCalculatedAttributesForCustomerProfile

Retrieve calculated attributes for a customer profile. Customer Profiles must be enabled for your Connect Customer instance.

Parameter object

A `ProfileId` must be present.

```
{
  "ProfileRequestData": {
    "ProfileId": Profile owning the calculated attribute
  },
  "ProfileResponseData": {
    All of these fields are optional.
    "CalculatedAttributes._average_hold_time",
    "CalculatedAttributes._frequent_caller",
    "CalculatedAttributes.x",
  }
}
```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.Customer path based on the attributes included in ProfileResponseData.

Errors

- `NotFoundError` - if no profiles were found for the associated profile search key.
- `NoMatchingError` - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

UpdateCustomerProfile

Update a customer profile that was previously created or retrieved in the flow. Customer Profiles must be enabled for your Connect Customer instance.

See [UpdateProfile](#) in the *Connect Customer Customer Profiles API Reference*.

Parameter object

```
{
  "ProfileRequestData": {
    All of these fields are optional.
    "FirstName",
    "MiddleName",
    "LastName",
    "PhoneNumber",
    "EmailAddress",
    "AccountNumber",
    "AdditionalInformation",
    "PartyType",
    "BusinessName",
    "BirthDate",
    "Gender",
    "MobilePhoneNumber",
    "HomePhoneNumber",
    "BusinessPhoneNumber",
    "BusinessEmailAddress",
    "Address1",
    "Address2",
    "Address3",
    "Address4",
    "City",
    "County",
    "Country",
    "PostalCode",
    "Province",
    "State",
    "ShippingAddress1",
    "ShippingAddress2",
    "ShippingAddress3",
    "ShippingAddress4",
    "ShippingCity",
    "ShippingCounty",
    "ShippingCountry",
    "ShippingPostalCode",
    "ShippingProvince",
    "ShippingState",
    "MailingAddress1",
    "MailingAddress2",
    "MailingAddress3",
    "MailingAddress4",
```

```

    "MailingCity",
    "MailingCounty",
    "MailingCountry",
    "MailingPostalCode",
    "MailingProvince",
    "MailingState",
    "BillingAddress1",
    "BillingAddress2",
    "BillingAddress3",
    "BillingAddress4",
    "BillingCity",
    "BillingCounty",
    "BillingCountry",
    "BillingPostalCode",
    "BillingProvince",
    "BillingState",
    "Attributes.x"
  },
  "ProfileResponseData": {
    All of these fields are optional.
    Newly created profile ID is persisted under the Customer -> ProfileID attribute
+ $.Customer.ProfileId
    "FirstName",
    "MiddleName",
    "LastName",
    "PhoneNumber",
    "EmailAddress",
    "AccountNumber",
    "AdditionalInformation",
    "PartyType",
    "BusinessName",
    "BirthDate",
    "Gender",
    "MobilePhoneNumber",
    "HomePhoneNumber",
    "BusinessPhoneNumber",
    "BusinessEmailAddress",
    "Address1",
    "Address2",
    "Address3",
    "Address4",
    "City",
    "County",
    "Country",

```

```
"PostalCode",
"Province",
"State",
"ShippingAddress1",
"ShippingAddress2",
"ShippingAddress3",
"ShippingAddress4",
"ShippingCity",
"ShippingCounty",
"ShippingCountry",
"ShippingPostalCode",
"ShippingProvince",
"ShippingState",
"MailingAddress1",
"MailingAddress2",
"MailingAddress3",
"MailingAddress4",
"MailingCity",
"MailingCounty",
"MailingCountry",
"MailingPostalCode",
"MailingProvince",
"MailingState",
"BillingAddress1",
"BillingAddress2",
"BillingAddress3",
"BillingAddress4",
"BillingCity",
"BillingCounty",
"BillingCountry",
"BillingPostalCode",
"BillingProvince",
"BillingState",
"Attributes.x"
}
```

Results and conditions

None. Conditions are not supported. If an error does not occur, the response's attributes are available dynamically under the \$.Customer path based on the attributes included in ProfileResponseData.

Errors

- `NoMatchingError` - if no other Error matches.

Corresponding block in the UI

[Customer profiles block](#)

Participant actions in the Connect Customer Flow language

Participant actions are attempted only when the flow is run in context of a participant. They generally result in an action that the participant experiences, such as playing a prompt or disconnecting.

Contents

- [ConnectParticipantWithLexBot](#)
- [DisconnectParticipant](#)
- [GetParticipantInput](#)
- [MessageParticipant](#)
- [MessageParticipantIteratively](#)
- [ShowView](#)

ConnectParticipantWithLexBot

Connects the participant with the specified Amazon Lex bot. When the interaction is over, the Intent and Slots of the bot are available to the flow during its run.

Action syntax

```
{
  "Parameters": {
    "PromptId": "string",
    "Text": "string",
    "SSML": "string",
    "Media": {
      "Uri": "string",
      "SourceType": "string",
      "MediaType": "string"
    }
  },
}
```

```
    "LexV2Bot": {
      "AliasArn": "string"
    },
    "LexBot": {
      "Name": "string",
      "Region": "string",
      "Alias": "string"
    },
    "LexSessionAttributes": {
      "string": "string"
    },
    "LexInitializationData": {
      "InitialMessage": "string"
    },
    "LexTimeoutSeconds": {
      "Text": "number"
    }
  },
  "Identifier": "string",
  "Type": "ConnectParticipantWithLexBot",
  "Transitions": {
    "NextAction": "string",
    "Errors": [
      {
        "NextAction": "string",
        "ErrorType": "InputTimeLimitExceeded"
      },
      {
        "NextAction": "string",
        "ErrorType": "NoMatchingError"
      },
      {
        "NextAction": "string",
        "ErrorType": "NoMatchingCondition"
      }
    ]
  }
}
```

Parameter object

Provide either LexBot or LexV2Bot object depending on the Amazon Lex version in the following format.

PromptId

A prompt ID or prompt ARN to play to the participant along with gathering input. This is an optional property and is not required if Text or SSML is specified. This must be either statically defined or a single valid JSONPath identifier.

Type: String

Required: No

Text

An optional string that defines text to send to the participant along with gathering input. May not be specified if PromptId or SSML is also specified. May be defined statically or dynamically.

Type: String

Required: No

SSML

An optional string that defines SSML to send to the participant along with gathering input. May not be specified if Text or PromptId is also specified. May be defined statically or dynamically.

Type: String

Required: No

Media

An optional object that defines an external media source.

- **Uri:** Location of the message.
- **SourceType:** The source from which the message will be fetched. The only supported type is S3.
- **MediaType:** The type of the message to be played. The only supported type is Audio.

LexV2Bot

The details of the LexV2 bot to invoke.

- **AliasArn:** The alias ARN of the LexV2 bot to invoke. May be specified statically or dynamically.

Required: Yes

LexSessionAttributes

A map of session attributes to pass to the Amazon LexV2 bot when it is invoked. The keys and values may be static or dynamic.

Required: No

LexInitializationData

A mapping that defines the initialization data which will be passed to Lex to prime the bot to improve end-customer experience. Lex initialization experience is only supported for Chat channel today. The Voice channel is not supported.

- **InitialMessage:** An optional string that defines the initial message parsed to Lex to prime the bot. The value for InitialMessage parameter is always serialized to \$.Media.InitialMessage which resolves to the customer's initial chat message.

Required: No

Note

If the initial message attribute is not included as part of the contact, it will result in the Get customer input block taking the error branch in the flow. To have separate flow configurations for different messaging types, such as web chat, SMS, or Apple Messages for Business, you can use the Check contact attributes block prior to using the Get customer input block to verify the initial message is available.

LexTimeoutSeconds

A mapping that defines the length of Lex timer in second, which will timeout inactive customers in a Lex interaction.

- **Text:** An optional string that defines the Lex timer length for chat.

Required: No

Transitions

If the Amazon Lex interaction succeeds, the result is the Intent of the bot. Conditions are supported, but only the Equals operator is supported within these conditions.

Errors

- **NoMatchingCondition:** If no specified condition evaluated to True.
- **NoMatchingError:** If an error occurred and no other error matched.
- **InputTimeLimitExceeded:** if there is no response before the configured LexTimeoutSeconds.

Conditions

- **NextAction:** A string that contains the Identifier of the Action that should be run after this Action, if no error or condition is preferentially chosen.
- **Supported Conditions:** Conditions are supported but only the EQUALS operator may be used on conditions.

Restrictions

This action is supported by all channels.

This action is available only in contact flows, transfer flows, and customer queue flows. It is not available in whisper flows or hold flows.

Corresponding block in the UI

[Get customer input](#)

DisconnectParticipant

Disconnects the participant from the contact and stops this flow from running.

Parameter object

No parameters are expected.

Results and conditions

None. Conditions are not supported.

Errors

None.

Restrictions

This action is supported for all channels and in contact flows, transfer flows, and customer queue flows.

Corresponding block in the UI

[Disconnect / hang up](#)

GetParticipantInput

Gathers customer input (a DTMF collection for voice contacts, or an entered string for other channels). There are many optional behaviors after gathering this: encryption, validation, storing to a "LastParticipantInput" section on the flow run data, specifying a custom DTMF terminator for voice contacts and so on. Details are in the parameter object section.

Parameter object

```
{
  "PromptId": [Optional] A prompt ID or prompt ARN to play to the participant along
  with gathering input. May not be specified if Text or SSML is also specified. Must be
  either statically defined or a single valid JSONPath identifier.
  "Text": An optional string that defines text to send to the participant along with
  gathering input. May not be specified if PromptId or SSML is also specified. May be
  defined statically or dynamically.
  "SSML": An optional string that defines SSML to send to the participant along with
  gathering input. May not be specified if Text or PromptId is also specified. May be
  defined statically or dynamically.
  "Media": { An optional object that defines an external media source
    "Uri": Location of the message
    "SourceType": The source from which the message will be fetched. The only
    supported type is S3
    "MediaType": The type of the message to be played. The only supported type is
    Audio
  }
  "InputTimeLimitSeconds": The number of seconds to wait for input to be collected
  before proceeding with a timeout error. For the Voice channel this is the timeout
  until the *first* DTMF digit is entered. Must be defined statically, and must be a
  valid integer larger than zero.
  "StoreInput": "True" or "False". Must be statically defined.
```

```

    "InputValidation": { An object that defines how to validate customer inputs,
    required if and only if StoreInput is True
        "PhoneNumberValidation": { Optional, one of the ways to validate inputs, make
        sure that it's a valid phone number. May not be specified if CustomValidation is
        specified.
            "NumberFormat": "Local" or "E164". If "Local" is specified, it is
            validated to be a local number (without the + and the country code), "E164" enforces
            that the customer input is a fully defined e.164 phone number. Must be defined
            statically.
            "CountryCode": If the number format is "Local", this must be defined. This
            is the two letter country code to be associated with the input number when validating.
            Must be defined statically.
        }
        "CustomValidation": { Optional, the other way to validate inputs. May not be
        specified if PhoneNumberValidation is specified.
            "MaximumLength": A number representing the maximum length of the input. May
            be defined statically or dynamically.
        }
    },
    "InputEncryption": { An optional object that defines how to encrypt the customer
    input. May only be specified if "CustomValidation" is provided.
        "EncryptionKeyId": The identifier of a key that has been uploaded in the AWS
        console for the purposes of customer input encryption. May be specified statically or
        dynamically.
        "Key": The PEM definition of the public key to use to encrypt this data. This
        key must be signed with the encryption key identified by the EncryptionKeyId. May be
        specified statically or dynamically.
    },
    "DTMFConfiguration": { An optional object to override default DTMF behavior for
    voice calls
        "InputTerminationSequence": Up to five digits to serve as the terminating
        sequence when gathering DTMF
        "DisableCancelKey": "True" or "False". If "True", the "*" key doesn't cancel
        gathering DTMF digits.
        "InterdigitTimeLimitSeconds": The number of seconds to wait between each DTMF
        digit entry before proceeding with a timeout error. This timeout applies after the
        first digit has been entered and resets after each subsequent digit. May be defined
        statically or dynamically, and must be a valid integer between 1 and 20 seconds.
    }
}

```

Results and conditions

If the "StoreInput" option is "True", there is no run result and conditions are not supported. If the "StoreInput" option is not defined or is "False", the run result is the participant input, and conditions are supported but only the Equals operator may be used on conditions. The values being compared must be static and be a single character - 0-9 numeric, *, or #.

Errors

- **NoMatchingCondition** - None of the specified conditions evaluated to true. Must be defined only if StoreInput is False.
- **NoMatchingError** - if no other Error matches. Must always be defined.
- **InvalidPhoneNumber** - the stored input was not a valid phone number according to the specified PhoneNumberValidation. Must be defined only if StoreInput is true, and PhoneNumberValidation is specified.
- **InputTimeLimitExceeded** - if there is no response before the configured InputTimeLimitSeconds.

Restrictions

This action is only supported on the voice channel.

This action can be used in contact flows, transfer flows, and customer queue flows but not in whisper flows or hold flows.

Corresponding block in the UI

[Get customer input](#)

MessageParticipant

Sends a message to the participant. This is an audio prompt or text-to-speech for voice contacts, or a text message for other channels.

Parameter object

```
{
  "PromptId": [Optional] A prompt ID or prompt ARN to play to the participant along
  with gathering input. May not be specified if Text or SSML is also specified. Must be
  specified either statically or as a single valid JSONPath identifier.
```

"Text": An optional string that defines text to send to the participant along with gathering input. May not be specified if PromptId or SSML is also specified. May be specified statically or dynamically.

"SSML": An optional string that defines SSML to send to the participant along with gathering input. May not be specified if Text or PromptId is also specified. May be specified statically or dynamically.

"Media": { An optional object that defines an external media source

"Uri": Location of the message

"SourceType": The source from which the message will be fetched. The only supported type is S3

"MediaType": The type of the message to be played. The only supported type is

Audio

}

}

Results and conditions

None. No conditions are supported.

Errors

NoMatchingError - If an error occurred and no other error matched.

Restrictions

This action is supported in contact flows, transfer flows, whisper flows, and customer queue flows. It is not supported in hold flows.

"PromptId" and "SSML" are only supported for the voice channel. All other channels support only the "Text" option.

Corresponding block in the UI

[Play](#)

MessageParticipantIteratively

Loops a sequence of prompts while a customer or agent is on hold or in queue. This block can be configured with an interruption timeout when in a Queue flow that interrupts the message loop to run other flow logic. The message loop can include entries for both Text and Prompts.

Parameter object

```
{
```

```

    "Messages" : [ A List of messages to be played in a loop. These are defined with
either TTS or a Prompt
    {
        "Text" : An optional string that defines text to send to the participant
    },
    {
        "PromptId" : A prompt ID or prompt ARN to play to the participant
    },
    {
        "SSML" : An optional string that defines the ssml
    },
    {
        "Media": { An optional object that defines an external media source
            "Uri": Location of the message
            "SourceType": The source from which the message will be fetched. The only
supported type is S3
            "MediaType": The type of the message to be played. The only supported type
is Audio
        }
    }

    ],
    "InterruptFrequencySeconds" : [Optional] Time to elapse before the action completes
with "MessagesInterrupted" run result
}

```

Results and conditions

When the timeout elapses, the action completes with the result as "MessagesInterrupted". Conditions are supported, but only the "Equals" operator is supported. The only supported operand is MessagesInterrupted.

Errors

- NoMatchingError - if no other Error matches.

Restrictions

This action is supported in Customer Queue, Customer Hold, and Agent Hold flows.

"PromptId" is supported only for the Voice channel, all other channels support only the "Text" option.

If this action is used on the chat channel, it immediately takes the error branch. If no error branch is available, the flow stop running and the contact is routed to next available agent.

Corresponding block in the UI

[Loop prompt](#)

ShowView

Initiates a UI-based workflow that can be surfaced to users of front end applications. This action can be used to create [step-by-step guides](#) for agents who are using the Connect Customer agent workspace.

Parameter object

```
{
  "ViewResource": {
    "Id": "Id of the View Resource that will be shown in the UI.",
    "Version": "Version of the View Resource that will be shown in the UI."
  },
  "InvocationTimeLimitSeconds": 400,
  "ViewData": {
    "Description": "An optional map of data that will be passed to the View Resource. Keys and values may be set statically or dynamically."
  },
  "SensitiveDataConfiguration": {
    "HideResponseOn": ["TRANSCRIPT"]
  }
}
```

Results and conditions

The result that the user selects when interacting with the View. The available conditions will be dependent on the View resource specified in action parameters.

Errors

- NoMatchingError - if no other Error matches.
- NoMatchingCondition - if no other Condition matches.

- `TimeLimitExceeded` - if there is no response before the configured `InvocationTimeLimitSeconds`.

Restrictions

This action is only supported on the chat channel.

This action can be used in inbound flows and customer queue flows.

To ensure reliable show view rendering, limit combined inputs and contact attributes to 16KB or less.

Corresponding block in the UI

[Show View](#)

This action routes step-by-step guides that are to be displayed to agents in the agent workspace. It routes the guides as chat contacts. This type of chat contact is different from the customer-based contact that the agent is handling.

This action can only be used in inbound contact flows.

Connect Customer Rules Function language

This section describes the Connect Customer Rules Function language. You can use it to add conditions to rules programmatically. The function language is a JSON-based representation of a series of rule conditions.

Contents

- [Concepts](#)
- [Example rule function in Connect Customer Rule Function language](#)
- [Conditions in Connect Customer Rules Function language](#)

Concepts

The following terms are used in the Rules Function language.

Operator

A function that is used to evaluate Operands. The very top Operator has to be either AND or OR.

Operands

An array of objects that Operator is evaluating on. The length the array and the type of each object depends on the Operator defined on the same level.

ComparisonValue

A JSON path string that specifies the value field the rule is comparing.

FilterClause

An object that defines additional criteria that the rule is evaluating against. Depending on **ComparisonValue**, the value of this field varies.

Negate

A Boolean value that indicates if negation should be applied to the operator.

Example rule function in Connect Customer Rule Function language

The following example shows a simple rule that would be evaluated to true if "refund" is mentioned by the customer who is in a specific queue.

To learn how to use conditions correctly, we recommend creating a new rule in the Connect Customer console, and then calling the [DescribeRule](#) API for it.

```
{
  "Version": "2022-11-25", // A string representing the version of the rule. Currently
                           // the only supported version is 2022-11-25.
  "RuleFunction": { // RuleFunction object body
    "Operator": "AND",
    "Operands": [
      {
        "Operator": "CONTAINS_ANY",
        "Operands": [
          "refund"
        ],
        "ComparisonValue": "$.ContactLens.RealTimeCall.ExactMatch.Transcript",
        "FilterClause": {
          "LogicOperator": "AND",
          "Filters": [
            {
              "Type": "ParticipantRole",
              "Data": "CUSTOMER"
            }
          ]
        }
      },
      {
        "Negate": false // If Negate were set to true, it would mean the word "refund"
                        // was NOT mentioned
      },
      {
        "Operator": "CONTAINS_ANY",
        "Operands": [
          "11111111-1234-5678-9123-12345678012" // QueueId
        ],
        "ComparisonValue": "$.ContactLens.RealTimeCall.Queue.QueueId",
        "Negate": false
      }
    ]
  }
}
```

```
}  
}
```

Conditions in Connect Customer Rules Function language

A rule function needs to start with either an AND or OR Operator. The Operands of AND or OR Operators is a list of conditions. Conditions vary depending on the TriggerEventSource.

Following are the conditions that you can use.

Contents

- [OnMetricDataUpdate](#)
- [OnContactEvaluationSubmit](#)
- [OnPostCallAnalysisAvailable](#)
- [OnRealTimeCallAnalysisAvailable](#)
- [OnPostChatAnalysisAvailable](#)
- [OnEmailAnalysisAvailable](#)
- [OnZendeskTicketCreate](#)
- [OnZendeskTicketStatusUpdate](#)
- [OnSalesforceCaseCreate](#)
- [OnCaseCreate](#)
- [OnCaseUpdate](#)
- [OnSlaBreach](#)
- [PatternMatch Operands](#)

OnMetricDataUpdate

Metric Data Update - Rule Conditions

Parameters

- Operator - "AND" or "OR"
- Operands – An array of MetricGroup, can support up to two groups.
- FilterClause – Rule level metric filter for **End Dimension** or grouping. Choosing an End Dimension restricts the usage of metrics, as described in [Metric Data Update - MetricCondition](#).

```

"Operator": "AND", // "AND or "OR"
"Operand": MetricGroup[], // Upto 2 array elements
"FilterClause" {
    "LogicOperator": "AND", // Only "AND" is supported
    "Filters": [
        {
            "Type": "MetricDataGrouping",
            "Data": ["AGENT" | "QUEUE" | "ROUTING_PROFILE" | "FLOW"] // Only one is
supported
        },
    ]
}

```

Metric Data Update - MetricGroup

Each MetricGroup can have up to compatible 10 MetricConditions grouped by an "AND" or "OR". Up to two MetricGroups are supported within the same Rule Condition.

```

"Operands": [
    {
        "Operator": "AND", // "AND or "OR"
        "Operands": MetricCondition[], // Upto 10 array elements
    },
    ...
]

```

Metric Data Update - Metric Filters

Every MetricCondition has a compound filter format, and requires a list of mandatory filters and some optional filters.

Required filters

These filters are always required in MetricCondition for the when supported, but not supported in others.

MetricDataPeriodSeconds

The trailing time period window, in seconds, for which the metric is calculated. All MetricCondition within the same MetricGroup are required to have a consistent MetricDataPeriodSeconds, that is, the exact same filter in all of them, or none of them should have these filters. The time window has to less than 24 hours (< 86400).

```
{
  "Type": "MetricDataPeriodSeconds", // Required for Historical metrics
  "Data": { "Past": 300}
}
```

Supported metrics: AVG_HANDLE_TIME, AVG_QUEUE_ANSWER_TIME, AVG_INTERACTION_TIME, AVG_HOLD_TIME, SERVICE_LEVEL_X, AGENT_OCCUPANCY. These metrics cannot be clubbed with other metrics in the same MetricGroup.

AgentActivityDurationSeconds

This filter is only required for AGENT_ACTIVITY metric, and determines the time period window, in seconds, for which the Agent Activity status is held. It has to be less than 24 hours (< 86400).

```
{
  "Type": "AgentActivityDurationSeconds",
  "Data": { "Past": 300}
}
```

MetricDataThresholdSeconds

This filter is only required for SERVICE_LEVEL_X, and is defined [here](#).

```
{
  "Type": "MetricDataThresholdSeconds",
  "Data": { "Comparison": "LT", "ThresholdValue": 300} // Comparison == "LT" only
}
```

End Dimension filters

Exactly one of the below filters is required for every MetricCondition. If the MetricConditions are in the same group, then they are required to have the same End Dimension Filter. For more information about these filters, see [GetMetricDataV2 request filters](#).

MetricDataFilterByAgent

Provide the list of Agent IDs in MetricCondition. Up to 25 values can be provided.

```
{
  "Type": "MetricDataFilterByAgent",
```

```
"Data": ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"]
}
```

MetricDataFilterByAgentHierarchyLevelX

Agent Hierarchy filters can be any of the five types:

1. MetricDataFilterByAgentHierarchyLevelOne
2. MetricDataFilterByAgentHierarchyLevelTwo
3. MetricDataFilterByAgentHierarchyLevelThree
4. MetricDataFilterByAgentHierarchyLevelFour
5. MetricDataFilterByAgentHierarchyLevelFive

Provide the Agent Hierarchy ID in the filter.

```
{
  "Type": "MetricDataFilterByAgentHierarchyLevelFive",
  "Data": ["b99ba702-b84b-4279-abad-5ceda9ff2640"] // One array element only
}
```

Note

Only one value is supported in the filter.

MetricDataFilterByQueue

Provide the list of Queue IDs in MetricCondition. Up to 25 values can be provided.

```
{
  "Type": "MetricDataFilterByQueue",
  "Data": ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"]
}
```

MetricDataFilterByRoutingProfile

Provide the list of Routing Profile IDs in MetricCondition. Up to 25 values can be provided.

```
{
  "Type": "MetricDataFilterByRoutingProfile",
  "Data": ["b653beae-0a34-4f0c-83d3-067c9b8370bf" , "60a4c89a-
b485-4bf2-8396-2a151b4e05a7"]
}
```

MetricDataFilterByFlow

Provide the list of Flow IDs in MetricCondition. Up to 25 values can be provided.

```
{
  "Type": "MetricDataFilterByFlow",
  "Data": ["a04377e2-zb01-g182-nc02-a285a66279en", "ae5a6965-zb64-g20d-n8a8-
a83d2a15081n"]
}
```

MetricDataFilterByFlowModule

Provide the list of flow module IDs in MetricCondition. Up to 25 values can be provided.

```
{
  "Type": "MetricDataFilterByFlowModule",
  "Data": ["a04377e2-zb01-g182-nc02-a285a66279en", "ae5a6965-zb64-g20d-n8a8-
a83d2a15081n"]
}
```

MetricDataFilterByFlowOutcomeType

Flow Outcome Type filter values can be any of the following depending on the filter resource is flow/module.

- DROPPED
- DISCONNECTED_PARTICIPANT
- ENDED_FLOW_EXECUTION
- TRANSFERRED_TO_QUEUE
- TRANSFERRED_TO_AGENT
- TRANSFERRED_TO_FLOW
- TRANSFERRED_TO_PHONE_NUMBER
- RETURNED_TO_FLOW

```
{
  "Type": "MetricDataFilterByFlowOutcomeType",
  "Data": ["DROPPED", "TRANSFERRED_TO_QUEUE"] // Multiple values can be provided
}
```

Optional filters

The following filters are optional in MetricCondition.

MetricDataFilterByChannel

This is an optional filter that is supported in MetricCondition for all metrics. However, if used, it is required to be consistent for every MetricCondition.

```
{
  "Type": "MetricDataFilterByChannel",
  "Data": ["VOICE", "CHAT", "TASK"], // any subset of this is valid
}
```

Exception: It cannot be used with AGENT_ACTIVITY and AGENT_OCCUPANCY metrics, but can be used for other metrics in the same RuleCondition but within a different MetricGroup.

Metric Data Update - MetricCondition

Contents

- [AGENT_ACTIVITY](#)
- [OLDEST_CONTACT_AGE](#)
- [AGENTS_AVAILABLE](#)
- [CONTACTS_IN_QUEUE](#)
- [AVG_QUEUE_ANSWER_TIME](#)
- [AVG_INTERACTION_TIME](#)
- [AVG_HOLD_TIME](#)
- [AVG_HANDLE_TIME](#)
- [AGENT_OCCUPANCY](#)
- [SERVICE_LEVEL](#)
- [FLOWS_STARTED](#)


```
},
```

Supported End Dimensions (MetricDataGrouping): AGENT

Required Filters: AgentActivityDurationSeconds

Supported End Dimension Filters: MetricDataFilterByAgent,
MetricDataFilterByAgentHierarchyLevelX, MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

OLDEST_CONTACT_AGE

For more information about the OLDEST_CONTACT_AGE metric, see [GetCurrentMetricData](#) and [Oldest](#).

Rule: OLDEST_CONTACT_AGE metric value, filtered by listed queues and channels ≥ 900 (15 minutes), then perform a [RuleAction](#).

```
{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.OLDEST_CONTACT_AGE",
  Operands: [900],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
    ],
  },
  Negate: false, // Fixed value
}
```

Supported End Dimensions (MetricDataGrouping): QUEUE

Required filters: MetricDataFilterByQueue, MetricDataFilterByRoutingProfile

Supported optional filters: `MetricDataFilterByChannel`

AGENTS_AVAILABLE

For more information about the AGENTS_AVAILABLE metric, see [GetCurrentMetricData](#) and [Available](#).

Rule: AGENTS_AVAILABLE metric value, filtered by listed queues and channels ≤ 5 , then perform a [RuleAction](#).

```
{
  Operator: "NumberLessOrEqualTo", // also supports "NumberGreaterOrEqualTo"
  ComparisonValue: "$.MetricData.AGENTS_AVAILABLE",
  Operands: [5],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
    ],
  },
  Negate: false, // fixed value
},
```

Supported End Dimensions (MetricDataGrouping): QUEUE, ROUTING_PROFILE

Supported End Dimension filters: `MetricDataFilterByQueue`,
`MetricDataFilterByRoutingProfile`

Supported optional filters: `MetricDataFilterByChannel`

CONTACTS_IN_QUEUE

For more information about the CONTACTS_IN_QUEUE metric, see [GetCurrentMetricData](#) and [In queue](#).

Rule: CONTACTS_IN_QUEUE metric value, filtered by listed queues and channels ≥ 100 , then perform a [RuleAction](#).

```
{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.CONTACTS_IN_QUEUE",
  Operands: [100],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
    ],
  },
  Negate: false, // fixed value
},
```

Supported End Dimensions (MetricDataGrouping): QUEUE

Supported End Dimension filters: MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

Supported optional filters: MetricDataFilterByChannel

AVG_QUEUE_ANSWER_TIME

For more information about the AVG_QUEUE_ANSWER_TIME metric, see [GetCurrentMetricData](#) and [Avg queue answer time](#).

Rule: AVG_QUEUE_ANSWER_TIME metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels ≥ 900 (15 minutes), then perform a [RuleAction](#).

```
{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.AVG_QUEUE_ANSWER_TIME",
```

```

    Operands: [900],
    FilterClause: {
      LogicOperator: "AND", // Always "AND"
      Filters: [
        {
          Type: "MetricDataFilterByQueue",
          Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
        },
        {
          Type: "MetricDataFilterByChannel",
          Data: ["CHAT"],
        },
        {
          Type: "MetricDataDurationSeconds",
          Data: { Past: 28800 },
        },
      ],
    },
    Negate: false, // fixed value
  }

```

Supported End Dimensions (MetricDataGrouping): QUEUE

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

Supported optional filters: MetricDataFilterByChannel

AVG_INTERACTION_TIME

For more information about the AVG_INTERACTION_TIME metric, see [GetMetricDataV2](#).

Rule: AVG_INTERACTION_TIME metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels $\geq$ 1800 (30 minutes), then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.AVG_INTERACTION_TIME",
  Operands: [1800],
  FilterClause: {

```

```

    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
      {
        Type: "MetricDataDurationSeconds",
        Data: { Past: 28800 }, // 8 hours
      },
    ],
  },
  Negate: false, // fixed value
},

```

Supported End Dimensions (MetricDataGrouping): QUEUE

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

Supported optional filters: MetricDataFilterByChannel

AVG_HOLD_TIME

For more information about the AVG_HOLD_TIME metric, see [GetMetricDataV2](#).

Rule: AVG_HOLD_TIME metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels $\geq$ 1800 (30 minutes), then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.AVG_HOLD_TIME",
  Operands: [1800],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [

```

```

    {
      Type: "MetricDataFilterByQueue",
      Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
    },
    {
      Type: "MetricDataFilterByChannel",
      Data: ["CHAT"],
    },
    {
      Type: "MetricDataDurationSeconds",
      Data: { Past: 28800 },
    },
  ],
},
Negate: false, // fixed value
},

```

Supported End Dimensions (MetricDataGrouping): QUEUE

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

Supported optional filters: MetricDataFilterByChannel

AVG_HANDLE_TIME

For more information about the AVG_HANDLE_TIME metric, see [GetMetricDataV2](#).

Rule: AVG_HANDLE_TIME metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels $\geq$ 1800 (30 minutes), then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.AVG_HANDLE_TIME",
  Operands: [1800],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",

```

```

        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-
a83d2a15081d"],
    },
    {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
    },
    {
        Type: "MetricDataDurationSeconds",
        Data: { Past: 28800 },
    },
],
},
Negate: false, // fixed value
},

```

Supported End Dimensions (MetricDataGrouping): QUEUE, AGENT

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: For QUEUE end dimension, MetricDataFilterByQueue and MetricDataFilterByRoutingProfile are supported. For AGENT end dimensions, MetricDataFilterByQueue, MetricDataFilterByRoutingProfile, MetricDataFilterByAgent, and MetricDataFilterByAgentLevelX are supported

Supported optional filters: MetricDataFilterByChannel

AGENT_OCCUPANCY

For more information about the AGENT_OCCUPANCY metric, see [GetMetricDataV2](#).

Rule: AGENT_OCCUPANCY metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels $\geq 80\%$, then perform a [RuleAction](#).

```

{
    Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
    ComparisonValue: "$.MetricData.AGENT_OCCUPANCY",
    Operands: [80],
    FilterClause: {
        LogicOperator: "AND", // Always "AND"
        Filters: [
            {
                Type: "MetricDataFilterByAgent",

```

```

        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-
a83d2a15081d"],
      },
      {
        Type: "MetricDataDurationSeconds",
        Data: { Past: 300 },
      },
    ],
  },
  Negate: false, // fixed value
},

```

Supported End Dimensions (MetricDataGrouping): AGENT

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByAgent,
MetricDataFilterByAgentLevelX, MetricDataFilterByRoutingProfile

SERVICE_LEVEL

For more information about the SERVICE_LEVEL metric, see [GetMetricDataV2](#).

Rule: SERVICE_LEVEL metric value, over the given duration of 600 (10 minutes) metric value, over the given duration of 28800 seconds (8 hours), filtered by listed queues and channels $\leq 50\%$, then perform a [RuleAction](#).

```

{
  Operator: "NumberLessOrEqualTo", // also supports "NumberGreaterOrEqualTo"
  ComparisonValue: "$.MetricData.SERVICE_LEVEL",
  Operands: [50],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByQueue",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-
a83d2a15081d"],
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
    ],
  },
}

```

```

    {
      Type: "MetricDataDurationSeconds",
      Data: { Past: 28800 },
    },
    {
      Type: "MetricDataThresholdSeconds",
      Data: { Comparison: MetricThresholdComparison.LT, ThresholdValue: 600},
    },
  ],
},
Negate: false, // fixed value
},

```

Supported End Dimensions (MetricDataGrouping): QUEUE

Required filters: MetricDataDurationSeconds, MetricDataThresholdSeconds

Supported End Dimension filters: MetricDataFilterByQueue,
MetricDataFilterByRoutingProfile

Supported optional filters: MetricDataFilterByChannel

FLOWS_STARTED

For more information about the FLOWS_STARTED metric, see [GetMetricDataV2](#).

Rule : FLOWS_STARTED metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows and channels 10, then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.FLOWS_STARTED",
  Operands: [10],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByFlow",
        Data: ["c04377e2-db01-4182-bc02-4285a66279e7", "1e5a6965-7b64-420d-88a8-a83d2a15081d"],
      },
      {
        Type: "MetricDataDurationSeconds",

```

```

        Data: { Past: 300 }
    },
    {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
    },
],
},
Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule

Supported optional filters: MetricDataFilterByChannel

AVG_FLOW_TIME

For more information about the AVG_FLOW_TIME metric, see [GetMetricDataV2](#).

Rule : AVG_FLOW_TIME metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows, flow outcomes and channels >= 600 (10 minutes), then perform a [RuleAction](#).

```

{
    Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
    ComparisonValue: "$.MetricData.AVG_FLOW_TIME",
    Operands: [600],
    FilterClause: {
        LogicOperator: "AND", // Always "AND"
        Filters: [
            {
                Type: "MetricDataFilterByFlow",
                Data: ["5f873afb-f359-4c54-b626-ecad1bd3f774"],
            },
            {
                Type: "MetricDataFilterByFlowOutcomeType",
                Data: ["DROPPED", "TRANSFERRED_TO_QUEUE"],
            },
        ],
    },
}

```

```

    {
      Type: "MetricDataDurationSeconds",
      Data: {
        "Past": 300
      },
    },
    {
      Type: "MetricDataFilterByChannel",
      Data: ["CHAT"],
    },
  ],
},
Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule, MetricDataFilterByFlowOutcomeType

Supported optional filters: MetricDataFilterByChannel

MAX_FLOW_TIME

For more information about the MAX_FLOW_TIME metric, see [GetMetricDataV2](#).

Rule : MAX_FLOW_TIME metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows, flow outcomes and channels >= 600 (10 minutes), then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
  ComparisonValue: "$.MetricData.MAX_FLOW_TIME",
  Operands: [600],
  FilterClause: {
    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByFlow",
        Data: ["5f873afb-f359-4c54-b626-ecad1bd3f774"],
      },
      {

```

```

        Type: "MetricDataFilterByFlowOutcomeType",
        Data: ["DROPPED", "TRANSFERRED_TO_QUEUE"],
    },
    {
        Type: "MetricDataDurationSeconds",
        Data: {
            "Past": 300
        },
    },
    {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
    },
],
Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule, MetricDataFilterByFlowOutcomeType

Supported optional filters: MetricDataFilterByChannel

MIN_FLOW_TIME

For more information about the MIN_FLOW_TIME metric, see [GetMetricDataV2](#).

Rule : MIN_FLOW_TIME metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows, flow outcomes and channels >= 600 (10 minutes), then perform a [RuleAction](#).

```

{
    Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
    ComparisonValue: "$.MetricData.MIN_FLOW_TIME",
    Operands: [600],
    FilterClause: {
        LogicOperator: "AND", // Always "AND"
        Filters: [
            {
                Type: "MetricDataFilterByFlow",
            }
        ]
    }
}

```

```

        Data: ["5f873afb-f359-4c54-b626-ecad1bd3f774"],
    },
    {
        Type: "MetricDataFilterByFlowOutcomeType",
        Data: ["DROPPED", "TRANSFERRED_TO_QUEUE"],
    },
    {
        Type: "MetricDataDurationSeconds",
        Data: {
            "Past": 300
        },
    },
    {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
    },
],
},
Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule, MetricDataFilterByFlowOutcomeType

Supported optional filters: MetricDataFilterByChannel

FLows_OUTCOME

For more information about the FLOWS_OUTCOME metric, see [GetMetricDataV2](#).

Rule : FLOWS_OUTCOME metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows, flow outcomes and channels >= 10, then perform a [RuleAction](#).

```

{
    Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"
    ComparisonValue: "$.MetricData.FLOWS_OUTCOME",
    Operands: [10],
    FilterClause: {

```

```

    LogicOperator: "AND", // Always "AND"
    Filters: [
      {
        Type: "MetricDataFilterByFlow",
        Data: ["5f873afb-f359-4c54-b626-ecad1bd3f774"],
      },
      {
        Type: "MetricDataFilterByFlowOutcomeType",
        Data: ["DROPPED", "TRANSFERRED_TO_QUEUE"],
      },
      {
        Type: "MetricDataDurationSeconds",
        Data: {
          "Past": 300
        },
      },
      {
        Type: "MetricDataFilterByChannel",
        Data: ["CHAT"],
      },
    ],
  },
  Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule, MetricDataFilterByFlowOutcomeType

Supported optional filters: MetricDataFilterByChannel

PERCENT_FLOWS_OUTCOME

For more information about the PERCENT_FLOWS_OUTCOME metric, see [GetMetricDataV2](#).

Rule : PERCENT_FLOWS_OUTCOME metric value, over the given duration of 300 seconds (5 minutes), filtered by listed flows, flow outcomes and channels $\geq 50\%$, then perform a [RuleAction](#).

```

{
  Operator: "NumberGreaterOrEqualTo", // also supports "NumberLessOrEqualTo"

```

```

ComparisonValue: "$.MetricData.PERCENT_FLOWS_OUTCOME",
Operands: [50],
FilterClause: {
  LogicOperator: "AND", // Always "AND"
  Filters: [
    {
      Type: "MetricDataFilterByFlow",
      Data: ["5f873afb-f359-4c54-b626-ecad1bd3f774"],
    },
    {
      Type: "MetricDataFilterByFlowOutcomeType",
      Data: ["DROPPED", "TRANSFERRED_TO_QUEUE"],
    },
    {
      Type: "MetricDataDurationSeconds",
      Data: {
        "Past": 300
      },
    },
    {
      Type: "MetricDataFilterByChannel",
      Data: ["CHAT"],
    },
  ],
},
Negate: false, // Fixed value
}

```

Supported End Dimensions (MetricDataGrouping): FLOW

Required Filters: MetricDataDurationSeconds

Supported End Dimension filters: MetricDataFilterByFlow,
MetricDataFilterByFlowModule, MetricDataFilterByFlowOutcomeType

Supported optional filters: MetricDataFilterByChannel

Full examples

Example 1

A rule with AGENTS_AVAILABLE, OLDEST_CONTACT_AGE, CONTACTS_IN_QUEUE in
MetricGroup 1 and AVG_QUEUE_ANSWER_TIME, AVG_HOLD_TIME, AVG_INTERACTION_TIME,

AVG_HANDLE_TIME, and SERVICE_LEVEL in Metric Group 2, both of which have End Dimension = QUEUE.

```
{
  "Operator": "OR",
  "Operands": [{
    "Operator": "AND",
    "Operands": [{
      "Operator": "NumberGreaterOrEqualTo",
      "ComparisonValue": "$.MetricData.AGENTS_AVAILABLE",
      "Operands": [0],
      "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [{
          "Type": "MetricDataFilterByQueue",
          "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
        }, {
          "Type": "MetricDataFilterByChannel",
          "Data": ["VOICE"]
        }
      ]
    },
    "Negate": false
  }, {
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.CONTACTS_IN_QUEUE",
    "Operands": [0],
    "FilterClause": {
      "LogicOperator": "AND",
      "Filters": [{
        "Type": "MetricDataFilterByQueue",
        "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
      }, {
        "Type": "MetricDataFilterByChannel",
        "Data": ["VOICE"]
      }
    ]
  },
  "Negate": false
}, {
  "Operator": "NumberGreaterOrEqualTo",
  "ComparisonValue": "$.MetricData.OLDEST_CONTACT_AGE",
  "Operands": [0],
```

```

        "FilterClause": {
            "LogicOperator": "AND",
            "Filters": [{
                "Type": "MetricDataFilterByQueue",
                "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
            }, {
                "Type": "MetricDataFilterByChannel",
                "Data": ["VOICE"]
            }]
        },
        "Negate": false
    ]
}, {
    "Operator": "OR",
    "Operands": [{
        "Operator": "NumberGreaterOrEqualTo",
        "ComparisonValue": "$.MetricData.AVG_HANDLE_TIME",
        "Operands": [0],
        "FilterClause": {
            "LogicOperator": "AND",
            "Filters": [{
                "Type": "MetricDataFilterByQueue",
                "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
            }, {
                "Type": "MetricDataDurationSeconds",
                "Data": {
                    "Past": 300
                }
            }, {
                "Type": "MetricDataFilterByChannel",
                "Data": ["VOICE"]
            }]
        },
        "Negate": false
    ]
}, {
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.AVG_QUEUE_ANSWER_TIME",
    "Operands": [0],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [{
            "Type": "MetricDataFilterByQueue",

```

```

        "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
      }, {
        "Type": "MetricDataDurationSeconds",
        "Data": {
          "Past": 300
        }
      }, {
        "Type": "MetricDataFilterByChannel",
        "Data": ["VOICE"]
      }
    ]
  },
  "Negate": false
}, {
  "Operator": "NumberGreaterOrEqualTo",
  "ComparisonValue": "$.MetricData.AVG_INTERACTION_TIME",
  "Operands": [0],
  "FilterClause": {
    "LogicOperator": "AND",
    "Filters": [{
      "Type": "MetricDataFilterByQueue",
      "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
    }, {
      "Type": "MetricDataDurationSeconds",
      "Data": {
        "Past": 300
      }
    }, {
      "Type": "MetricDataFilterByChannel",
      "Data": ["VOICE"]
    }
  ]
},
  "Negate": false
}, {
  "Operator": "NumberGreaterOrEqualTo",
  "ComparisonValue": "$.MetricData.AVG_HOLD_TIME",
  "Operands": [0],
  "FilterClause": {
    "LogicOperator": "AND",
    "Filters": [{
      "Type": "MetricDataFilterByQueue",
      "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]

```

```

        }, {
            "Type": "MetricDataDurationSeconds",
            "Data": {
                "Past": 300
            }
        }, {
            "Type": "MetricDataFilterByChannel",
            "Data": ["VOICE"]
        }
    ]
},
"Negate": false
}, {
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.SERVICE_LEVEL",
    "Operands": [0],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [{
            "Type": "MetricDataFilterByQueue",
            "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
        }, {
            "Type": "MetricDataDurationSeconds",
            "Data": {
                "Past": 300
            }
        }, {
            "Type": "MetricDataThresholdSeconds",
            "Data": {
                "Comparison": "LT",
                "ThresholdValue": 60
            }
        }, {
            "Type": "MetricDataFilterByChannel",
            "Data": ["VOICE"]
        }
    ]
},
"Negate": false
}]
}],
"FilterClause": {
    "LogicOperator": "AND",
    "Filters": [{
        "Type": "MetricDataGrouping",

```

```

        "Data": ["QUEUE"]
    }
}

```

Example 2

A rule with AGENT_OCCUPANCY in MetricGroup 1 and AGENT_ACTIVITY and MetricGroup 2, with End Dimension = AGENT.

```

{
  "Operator": "OR",
  "Operands": [{
    "Operator": "AND",
    "Operands": [{
      "Operator": "NumberGreaterOrEqualTo",
      "ComparisonValue": "$.MetricData.AGENT_OCCUPANCY",
      "Operands": [0],
      "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [{
          "Type": "MetricDataFilterByAgent",
          "Data": ["e7a11ddd-041d-449a-ab1e-6a493cb83626", "26b9c7f4-
ed5c-4cd1-b751-2cbfb4c54ea2"]
        }, {
          "Type": "MetricDataDurationSeconds",
          "Data": {
            "Past": 100
          }
        }
      ]
    }, {
      "Operator": "EQUALS",
      "ComparisonValue": "$.MetricData.AGENT_ACTIVITY",
      "Operands": ["ee8c71b9-49c2-4d00-ab6d-244f2d9a5c58"],
      "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [{
          "Type": "MetricDataFilterByAgentHierarchyLevelOne",

```

```

        "Data": ["b99ba702-b84b-4279-abad-5ceda9ff2640"]
      }, {
        "Type": "MetricDataDurationSeconds",
        "Data": {
          "Past": 600
        }
      }
    ]
  },
  "Negate": false
}]
}],
"FilterClause": {
  "LogicOperator": "AND",
  "Filters": [{
    "Type": "MetricDataGrouping",
    "Data": ["AGENT"]
  }]
}
}

```

Example 3

A rule with FLOWS_STARTED, AVG_FLOW_TIME, MAX_FLOW_TIME in MetricGroup 1 and FLOWS_OUTCOME, PERCENT_FLOWS_OUTCOME and MIN_FLOW_TIME in Metric Group 2, both of which have End Dimension = FLOW.

```

{
  "Operator": "OR",
  "Operands": [
    {
      "Operator": "AND",
      "Operands": [
        {
          "Operator": "NumberGreaterOrEqualTo",
          "ComparisonValue": "$.MetricData.FLOWS_STARTED",
          "Operands": [
            10
          ],
        },
        "FilterClause": {
          "LogicOperator": "AND",
          "Filters": [
            {
              "Type": "MetricDataFilterByFlow",

```

```

        "Data": [
            "z2a11ddd-041d-449a-ab1e-6a493cb83621",
            "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
        ]
    },
    {
        "Type": "MetricDataFilterByChannel",
        "Data": [
            "VOICE"
        ]
    },
    {
        "Type": "MetricDataDurationSeconds",
        "Data": {
            "Past": 300
        }
    }
]
},
"Negate": false
},
{
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.AVG_FLOW_TIME",
    "Operands": [
        600
    ],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
            {
                "Type": "MetricDataFilterByFlow",
                "Data": [
                    "z2a11ddd-041d-449a-ab1e-6a493cb83621",
                    "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
                ]
            },
            {
                "Type": "MetricDataFilterByFlowOutcomeType",
                "Data": [
                    "DROPPED",
                    "TRANSFERRED_TO_QUEUE"
                ]
            }
        ]
    }
},

```

```

        {
            "Type": "MetricDataFilterByChannel",
            "Data": [
                "VOICE"
            ]
        },
        {
            "Type": "MetricDataDurationSeconds",
            "Data": {
                "Past": 300
            }
        }
    ],
    "Negate": false
},
{
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.MAX_FLOW_TIME",
    "Operands": [
        600
    ],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
            {
                "Type": "MetricDataFilterByFlow",
                "Data": [
                    "z2a11ddd-041d-449a-ab1e-6a493cb83621",
                    "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
                ]
            },
            {
                "Type": "MetricDataFilterByFlowOutcomeType",
                "Data": [
                    "DROPPED",
                    "TRANSFERRED_TO_QUEUE"
                ]
            }
        ]
    },
    {
        "Type": "MetricDataFilterByChannel",
        "Data": [
            "VOICE"
        ]
    }
}

```

```

        },
        {
            "Type": "MetricDataDurationSeconds",
            "Data": {
                "Past": 300
            }
        }
    ]
},
"Negate": false
}
]
},
{
    "Operator": "OR",
    "Operands": [
        {
            "Operator": "NumberGreaterOrEqualTo",
            "ComparisonValue": "$.MetricData.FLOWS_OUTCOME",
            "Operands": [
                10
            ],
            "FilterClause": {
                "LogicOperator": "AND",
                "Filters": [
                    {
                        "Type": "MetricDataFilterByFlow",
                        "Data": [
                            "z2a11ddd-041d-449a-ab1e-6a493cb83621",
                            "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
                        ]
                    },
                    {
                        "Type": "MetricDataFilterByFlowOutcomeType",
                        "Data": [
                            "DROPPED",
                            "TRANSFERRED_TO_QUEUE"
                        ]
                    }
                ]
            },
            "Type": "MetricDataFilterByChannel",
            "Data": [
                "VOICE"
            ]
        }
    ]
}
]

```

```

        },
        {
            "Type": "MetricDataDurationSeconds",
            "Data": {
                "Past": 300
            }
        }
    ]
},
"Negate": false
},
{
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.PERCENT_FLOWS_OUTCOME",
    "Operands": [
        50
    ],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
            {
                "Type": "MetricDataFilterByFlow",
                "Data": [
                    "z2a11ddd-041d-449a-ab1e-6a493cb83621",
                    "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
                ]
            },
            {
                "Type": "MetricDataFilterByFlowOutcomeType",
                "Data": [
                    "DROPPED",
                    "TRANSFERRED_TO_QUEUE"
                ]
            }
        ],
        {
            "Type": "MetricDataFilterByChannel",
            "Data": [
                "VOICE"
            ]
        }
    ],
    {
        "Type": "MetricDataDurationSeconds",
        "Data": {
            "Past": 300
        }
    }
}

```

```

        }
    }
]
},
"Negate": false
},
{
    "Operator": "NumberGreaterOrEqualTo",
    "ComparisonValue": "$.MetricData.MIN_FLOW_TIME",
    "Operands": [
        600
    ],
    "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
            {
                "Type": "MetricDataFilterByFlow",
                "Data": [
                    "z2a11ddd-041d-449a-ab1e-6a493cb83621",
                    "y1b9c7f4-ed5c-4cd1-b751-2cbfb4c54ea3"
                ]
            },
            {
                "Type": "MetricDataFilterByFlowOutcomeType",
                "Data": [
                    "DROPPED",
                    "TRANSFERRED_TO_QUEUE"
                ]
            },
            {
                "Type": "MetricDataFilterByChannel",
                "Data": [
                    "VOICE"
                ]
            },
            {
                "Type": "MetricDataDurationSeconds",
                "Data": {
                    "Past": 300
                }
            }
        ]
    },
    "Negate": false
}

```

```
    }
  ]
}
],
"FilterClause": {
  "LogicOperator": "AND",
  "Filters": [
    {
      "Type": "MetricDataGrouping",
      "Data": [
        "FLOW"
      ]
    }
  ]
}
]
```

OnContactEvaluationSubmit

Agent Hierarchy

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Agent hierarchy ARNs.
- ComparisonValue - "\$.ContactLens.ContactEvaluation.Agent.HierarchyGroup.ARN"
- Negate - false

Initiation Method

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'INBOUND'
 - 'OUTBOUND'
 - 'TRANSFER'
 - 'QUEUE_TRANSFER'

- 'CALLBACK'
- 'API'
- 'DISCONNECT'
- ComparisonValue - "\$.ContactLens.ContactEvaluation.InitiationMethod"
- Negate - false

DisconnectReason

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'TELECOM_BUSY'
 - 'TELECOM_NUMBER_INVALID'
 - 'TELECOM_POTENTIAL_BLOCKING'
 - 'TELECOM_UNANSWERED'
 - 'TELECOM_TIMEOUT'
 - 'TELECOM_ORIGINATOR_CANCEL'
 - 'TELECOM_PROBLEM'
 - 'CUSTOMER_NEVER_ARRIVED'
 - 'THIRD_PARTY_DISCONNECT'
 - 'CUSTOMER_DISCONNECT'
 - 'AGENT_DISCONNECT'
 - 'BARGED'
 - 'CONTACT_FLOW_DISCONNECT'
 - 'OTHER'
 - 'OUTBOUND_DESTINATION_ENDPOINT_ERROR'
 - 'OUTBOUND_RESOURCE_ERROR'
 - 'OUTBOUND_ATTEMPT_FAILED'
 - 'EXPIRED'
 - 'AGENT_NETWORK_DISCONNECT'
 - 'CUSTOMER_CONNECTION_NOT_ESTABLISHED'

- 'API'
- 'IDLE_DISCONNECT'
- 'SYSTEM_ERROR'
- 'AGENT_COMPLETED'
- 'TRANSFERRED'
- 'DISCARDED'
- ComparisonValue - "\$.ContactLens.ContactEvaluation.DisconnectReason"
- Negate - false

Routing Profile

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Routing profile ARNs.
- ComparisonValue - "\$.ContactLens.ContactEvaluation.Agent.RoutingProfile"
- Negate - false

PotentialDisconnectIssue

Parameters

- Operator - "CONTAINS_ANY" or "EQUALS"
- Operands - A validated enum of possible values.
- ComparisonValue - "\$.ContactLens.ContactEvaluation.PotentialDisconnectIssue"
- Negate - false

Custom User-Defined Segment Attribute

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of segment attribute values. The values must be a value in the pre-defined attribute resource.

- ComparisonValue - "\$.ContactLens.ContactEvaluation.SegmentAttributes.UserDefined.[KEY]"

The KEY must be an instance pre-defined attribute resource.

- Negate - false or true

ContactEvaluation - Results available condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the evaluation form ID.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Form.FormId"
- Negate - false

ContactEvaluation - Form score condition

ContactEvaluation form score condition has a compound condition format where Operator is an AND condition and its operands consist of two conditions that represent the form and the form score.

Parameters that represent the form: See [ContactEvaluation - Results available condition](#)

Parameters that represent the form score:

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is the form score.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Form.Score"
- Negate - false

Following is an example.

```
{
  "Operator": "AND",
  "Operands": [
    {
      "Operator": "EQUALS",
      "Operands": ["11111111-1234-5678-9123-12345678012"],
    }
  ]
}
```

```

        "ComparisonValue": "$.ContactLens.ContactEvaluation.Form.FormId",
        "Negate": false
    },
    {
        "Operator": "NumberLessOrEqualTo",
        "Operands": [50],
        "ComparisonValue": "$.ContactLens.ContactEvaluation.Form.Score",
        "Negate": false
    },
]
}

```

ContactEvaluation - Section Score

ContactEvaluation section score condition has a compound condition format where Operator is an AND condition and its operands consist of three conditions that represent the form, section, and section score respectively.

Parameters that represent the form: See [ContactEvaluation - Results available condition](#)

Parameters that represent the section:

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the section reference ID.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Section.SectionRefId"
- Negate - false

Parameters that represent the section score:

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is the section score.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Section.Score"
- Negate - false

Following is an example.

```

{
    "Operator": "AND",
    "Operands": [

```

```

    {
      "Operator": "EQUALS",
      "Operands": ["11111111-1234-5678-9123-12345678012"],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Form.FormId",
      "Negate": false
    },
    {
      "Operator": "EQUALS",
      "Operands": ["s12345678"],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Section.SectionRefId",
      "Negate": false
    },
    {
      "Operator": "NumberLessOrEqualTo",
      "Operands": [50],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Section.Score",
      "Negate": false
    },
  ],
}

```

ContactEvaluation - Question and Answer

ContactEvaluation question and answer condition has a compound condition format where Operator is an AND condition and its operands consist of three conditions that represent the form, question, and answer value respectively.

Parameters that represent the form: See [ContactEvaluation - Results available condition](#)

Parameters that represent the question:

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the question reference ID.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Question.QuestionRefId"
- Negate - false

Parameters that represent a numeric answer:

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is the answer value.

- ComparisonValue – "\$.ContactLens.ContactEvaluation.Question.Answer.Value"
- Negate - false

Following is an example for a numeric question type.

```
{
  "Operator": "AND",
  "Operands": [
    {
      "Operator": "EQUALS",
      "Operands": ["11111111-1234-5678-9123-12345678012"],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Form.FormId",
      "Negate": false
    },
    {
      "Operator": "EQUALS",
      "Operands": ["s12345678"],
      "ComparisonValue":
        "$.ContactLens.ContactEvaluation.Question.QuestionRefId",
      "Negate": false
    },
    {
      "Operator": "NumberLessOrEqualTo",
      "Operands": [5],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Question.Answer.Value",
      "Negate": false
    }
  ]
}
```

Parameters that represent a single select answer:

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the answer reference ID.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Question.Answer.ValueRefId"
- Negate - true/false. If set to true, it means *The answer is not equal to the answer reference ID specified in the Operands.*

Following is an example for single select question type.

```
{
  "Operator": "AND",
  "Operands": [
    {
      "Operator": "EQUALS",
      "Operands": ["11111111-1234-5678-9123-12345678012"],
      "ComparisonValue": "$.ContactLens.ContactEvaluation.Form.FormId",
      "Negate": false
    },
    {
      "Operator": "EQUALS",
      "Operands": ["q12345678"],
      "ComparisonValue":
        "$.ContactLens.ContactEvaluation.Question.QuestionRefId",
      "Negate": false
    },
    { // for single select question type
      "Operator": "EQUALS",
      "Operands": ["o12345678"],
      "ComparisonValue":
        "$.ContactLens.ContactEvaluation.Question.Answer.ValueRefId",
      "Negate": false
    }
  ]
}
```

ContactEvaluation - agent condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of agent IDs.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Agent.AgentId"
- Negate - false

ContactEvaluation - queue condition

Parameters

- Operator – "EQUALS"

- Operands – No value
- ComparisonValue – "\$.ContactLens.ContactEvaluation.Queue.QueueId"
- Negate - false

ContactEvaluation - contact attributes condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the contact attribute value.
- ComparisonValue – "\$.ContactLens.ContactEvaluation.ContactAttribute.*YOUR_ATTRIBUTE_KEY*"
- Negate - true/false. If set to true, it means *YOUR_ATTRIBUTE_KEY* does not equal to the attribute value specified in the Operands.

OnPostCallAnalysisAvailable

Agent Hierarchy

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Agent hierarchy ARNs.
- ComparisonValue - "\$.ContactLens.PostCall.Agent.HierarchyGroup.ARN"
- Negate - false

NonTalkTimePct

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands - An array of numbers, array length can only be 1. Value is an integer between 0 and 100.
- ComparisonValue - "\$.ContactLens.PostCall.Agent.NonTalkTimePct"
- Negate - false

Customer Hold Duration Pct

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands - An array of number, array length can only be 1. Value is an integer between 0 and 100.
- ComparisonValue - "\$.ContactLens.PostCall.Agent.CustomerHoldDurationPct"
- Negate - false

Initiation Method

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'INBOUND'
 - 'OUTBOUND'
 - 'TRANSFER'
 - 'QUEUE_TRANSFER'
 - 'CALLBACK'
 - 'API'
 - 'DISCONNECT'
- ComparisonValue - "\$.ContactLens.PostCall.InitiationMethod"
- Negate - false

DisconnectReason

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'TELECOM_BUSY'
 - 'TELECOM_NUMBER_INVALID'

- 'TELECOM_POTENTIAL_BLOCKING'
- 'TELECOM_UNANSWERED'
- 'TELECOM_TIMEOUT'
- 'TELECOM_ORIGINATOR_CANCEL'
- 'TELECOM_PROBLEM'
- 'CUSTOMER_NEVER_ARRIVED'
- 'THIRD_PARTY_DISCONNECT'
- 'CUSTOMER_DISCONNECT'
- 'AGENT_DISCONNECT'
- 'BARGED'
- 'CONTACT_FLOW_DISCONNECT'
- 'OTHER'
- 'OUTBOUND_DESTINATION_ENDPOINT_ERROR'
- 'OUTBOUND_RESOURCE_ERROR'
- 'OUTBOUND_ATTEMPT_FAILED'
- 'EXPIRED'
- 'AGENT_NETWORK_DISCONNECT'
- 'CUSTOMER_CONNECTION_NOT_ESTABLISHED'
- 'API'
- 'IDLE_DISCONNECT'
- 'SYSTEM_ERROR'
- 'AGENT_COMPLETED'
- 'TRANSFERRED'
- 'DISCARDED'
- ComparisonValue - "\$.ContactLens.PostCall.DisconnectReason"
- Negate - false

Absolute Talk Time

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"

- Operands - An integer.
- ComparisonValue - "\$.ContactLens.PostCall.TalkTime.TotalTimeSecs"
- FilterClause -

```
{ // required!!
  LogicOperator: 'AND', // has to be 'AND'
  Filters: [ // only allows one of the object
    {
      Type: 'ParticipantRole',
      Data: 'AGENT', // 'AGENT' or 'CUSTOMER'
    },
  ],
},
```

- Negate - false

Maximum (Highest) Loudness Score

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands - an integer (decibels).
- ComparisonValue - "\$.ContactLens.PostCall.Loudness.HighestLoudnessScore"
- FilterClause -

```
{
  LogicOperator: 'AND', // has to be AND
  Filters: [ // only one object
    {
      Type: 'ParticipantRole',
      Data: 'CUSTOMER', // or 'AGENT'
    },
  ],
},
```

- Negate - false

After Contact Work Duration

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands - A positive integer or zero.
- ComparisonValue - "\$.ContactLens.PostCall.Agent.AfterContactWorkDurationSecs"
- Negate - false

Routing profile

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Routing profile ARNs:
- ComparisonValue - "\$.ContactLens.PostCall.Agent.RoutingProfile"
- Negate - false

PotentialDisconnectIssue

Parameters

- Operator - "CONTAINS_ANY" | "EQUALS"
- Operands - A validated enum of possible values.
- ComparisonValue - "\$.ContactLens.PostCall.PotentialDisconnectIssue"
- Negate - false

Custom User-Defined Segment Attribute

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of segment attribute values. The values must be a value in the pre-defined attribute resource.
- ComparisonValue - "\$.ContactLens.PostCall.SegmentAttributes.UserDefined.[KEY]"

The KEY must be an instance pre-defined attribute resource.

- Negate - false or true

Connect AI Agent

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of strings. Strings are either a list of different UUIDs by themselves or a list of strings prefixed with the same UUID, with different possible version numbers uuid:2 and uuid:3. For example:
 - Different UUIDs by themselves: ['30682191-2312-4c27-9fc1-752e7a399d52', '2b81c7d5-85bc-459f-b605-0f5280a4301e']
 - Prefixed with the same UUID with different possible version numbers: ['30682191-2312-4c27-9fc1-752e7a399d52:1', '30682191-2312-4c27-9fc1-752e7a399d52:2']
- ComparisonValue - "\$.ContactLens.PostCall.AiAgent.IdWithVersion"
- FilterClause -

```
[
  { // REQUIRED filter
    Type: 'AiAgentUseCase',
    Data: {
      Value: 'SelfService', // or 'AgentAssistance'
    },
  },
  { // OPTIONAL filter
    Type: 'AiAgentEscalatedToHuman',
    Data: 'TRUE', // or 'FALSE'
  },
],
},
```

- Negate - false

PostCall words or phrases - Exact match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of strings
- ComparisonValue – "\$.ContactLens.PostCall.ExactMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "PostCallContactPeriodSeconds", // Optional filter type. If
      omitted, the rule applies to the full contact
      "Data": { // Either "First" or "Last" needs to be specified.
        "First": number
      }
    },
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    }
  ]
}
```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostCall words or phrases - Semantic match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of strings
- ComparisonValue – "\$.ContactLens.PostCall.SemanticMatch.Transcript"
- FilterClause –

```
{
```

```

    "LogicOperator": "AND", // Only "AND" is supported
    "Filters": [
      {
        "Type": "PostCallContactPeriodSeconds", // Optional filter type. If
omitted, the rule applies to the full contact
        "Data": { // Either "First" or "Last" needs to be specified.
          "First": number
        }
      },
      {
        "Type": "ParticipantRole",
        "Data": "CUSTOMER" | "AGENT" | "ANY"
      }
    ]
  }
}

```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostCall natural language - Semantic match

Parameters

- Operator - "EQUALS"
- Operands – A list with a single string
- ComparisonValue – "\$.ContactLens.PostCall.SemanticMatch.Phrase"
- FilterClause –

```

{
  Operator: "EQUALS",
  Operands: [
    "Customer called to change their address and agent proceeded with the
address change during the call."
  ],
  ComparisonValue: "$.ContactLens.PostCall.SemanticMatch.Phrase",
  Negate: true or false
},

```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostCall words or phrases - Pattern match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of pattern match objects. See [PatternMatch Operands](#).
- ComparisonValue – "\$.ContactLens.PostCall.PatternMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "PostCallContactPeriodSeconds", // Optional filter type. If
      omitted, the rule applies to the full contact
      "Data": { // Either "First" or "Last" needs to be specified.
        "First": number
      }
    },
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
    {
      "Type": "PatternMatchLanguageFilter",
      "Data": "EN" | "ES" | "AR" | "DE" | "FR" | "HI" | "IT" | "PT" | "KO" | "JA" | "ZH"
    }
  ]
}
```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostCall agent condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of agent IDs
- ComparisonValue – "\$.ContactLens.PostCall.Agent.AgentId"

- Negate - false

PostCall queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue IDs.
- ComparisonValue – "\$.ContactLens.PostCall.Queue.QueueId"
- Negate - false

PostCall no queue condition

Parameters

- Operator - "EQUALS"
- Operands – No value.
- ComparisonValue – "\$.ContactLens.PostCall.Queue.QueueId"
- Negate - false

PostCall contact attributes condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the contact attribute value.
- ComparisonValue – "\$.ContactLens.PostCall.ContactAttribute.*YOUR_ATTRIBUTE_KEY*"
- Negate - true/false. If set to true, it means *YOUR_ATTRIBUTE_KEY* does not equal to the attribute value specified in the Operands. .

PostCall sentiment state condition

Parameters

- Operator - "EQUALS"

- Operands – An array of string, array length can only be 1. Value is one of "POSITIVE", "NEGATIVE", "NEUTRAL".
- ComparisonValue – "\$.ContactLens.PostCall.Sentiment.State"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}
```

- Negate - false

PostCall sentiment overall score condition

Parameters

- Operator - "NumberLessOrEqualTo | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostCall.Sentiment.OverallScore"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}
```

- Negate - false

PostCall sentiment score shift condition

PostCall sentiment score shift condition has a compound condition format where Operator is an AND condition and its operands consist of two conditions that represent beginning and end score respectively.

Parameters that represent the beginning score

- Operator - "NumberLessOrEqualTo | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostCall.Sentiment.Score.Beginning"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}
```

- Negate - false

Parameters that represent the end score

- Operator - "NumberLessOrEqualTo | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostCall.Sentiment.Score.End"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
```

```

        "Data": "CUSTOMER" | "AGENT" | "ANY"
      },
    ]
  }

```

- Negate - false

Following is an example:

```

{
  "Operator": "AND",
  "Operands": [
    {
      "Operator": "NumberGreaterOrEqualTo",
      "Operands": [
        2
      ],
      "ComparisonValue": "$.ContactLens.PostCall.Sentiment.Score.Beginning",
      "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
          {
            "Type": "ParticipantRole",
            "Data": "AGENT"
          }
        ]
      },
      "Negate": false
    },
    {
      "Operator": "NumberGreaterOrEqualTo",
      "Operands": [
        3
      ],
      "ComparisonValue": "$.ContactLens.PostCall.Sentiment.Score.End",
      "FilterClause": {
        "LogicOperator": "AND",
        "Filters": [
          {
            "Type": "ParticipantRole",
            "Data": "AGENT"
          }
        ]
      }
    }
  ]
}

```

```

        },
        "Negate": false
    }
]
}

```

Parameters for overall score

- Operator - "NumberLessOrEqualTo | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostCall.Sentiment.OverallScore"
- FilterClause –

```

{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}

```

- Negate - false

PostCall total non-talk time condition

Parameters

- Operator - "NumberLessOrEqualTo | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.NonTalkTime.TotalTimeSecs"
- Negate - false

PostCall longest non-talk time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.NonTalkTime.LongestTimeSecs"
- Negate - false

PostCall interruption condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.Interruptions.Instances"
- Negate - false

PostCall total hold time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.Agent.CustomerHoldDurationSecs"
- Negate - false

PostCall longest hold time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"

- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.Agent.LongestHoldDurationSecs"
- Negate - false

PostCall number of holds condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.Agent.NumberOfHolds"
- Negate - false

PostCall agent interaction duration condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostCall.Agent.AgentInteractionDurationSecs"
- Negate - false

OnRealTimeCallAnalysisAvailable

RealTimeCall words or phrases - Exact match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of strings
- ComparisonValue – "\$.ContactLens.RealTimeCall.ExactMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    }
  ]
}
```

- Negate - false.

RealTimeCall words or phrases - Pattern match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of pattern match objects. See [PatternMatch Operands](#).
- ComparisonValue – "\$.ContactLens.RealTimeCall.PatternMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
    {
      "Type": "PatternMatchLanguageFilter",
      "Data": "EN" | "ES" | "AR" | "DE" | "FR" | "HI" | "IT" | "PT" | "KO" | "JA" | "ZH"
    }
  ]
}
```

- Negate - false

RealTimeCall agent condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of agent IDs
- ComparisonValue – "\$.ContactLens.RealTimeCall.Agent.AgentId"
- Negate - false

RealTimeCall queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue IDs
- ComparisonValue – "\$.ContactLens.RealTimeCall.Queue.QueueId"
- Negate - true/false. If set to true, it means *If queue is not any of the queues mentioned in the Operands.*

RealTimeCall contact attributes condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the contact attribute value.
- ComparisonValue – "\$.ContactLens.RealTimeCall.ContactAttribute.*YOUR_ATTRIBUTE_KEY*"
- Negate - true/false. If set to true, it means *YOUR_ATTRIBUTE_KEY does not equal to the attribute value specified in the Operands .*

RealTimeCall sentiment state condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "POSITIVE", "NEGATIVE", "NEUTRAL".

- ComparisonValue – "\$.ContactLens.RealTimeCall.Sentiment.State"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
    {
      "Type": "RealTimeCallContactPeriodSeconds",
      "Data": {
        "Past": number
      }
    }
  ]
}
```

- Negate - false

OnPostChatAnalysisAvailable

Agent Hierarchy

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Agent hierarchy ARNs.
- ComparisonValue - "\$.ContactLens.PostChat.Agent.HierarchyGroup.ARN"
- Negate - false

Initiation Method

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:

- 'INBOUND'
- 'OUTBOUND'
- 'TRANSFER'
- 'QUEUE_TRANSFER'
- 'CALLBACK'
- 'API'
- 'DISCONNECT'
- ComparisonValue - "\$.ContactLens.PostChat.InitiationMethod"
- Negate - false

DisconnectReason

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'TELECOM_BUSY'
 - 'TELECOM_NUMBER_INVALID'
 - 'TELECOM_POTENTIAL_BLOCKING'
 - 'TELECOM_UNANSWERED'
 - 'TELECOM_TIMEOUT'
 - 'TELECOM_ORIGINATOR_CANCEL'
 - 'TELECOM_PROBLEM'
 - 'CUSTOMER_NEVER_ARRIVED'
 - 'THIRD_PARTY_DISCONNECT'
 - 'CUSTOMER_DISCONNECT'
 - 'AGENT_DISCONNECT'
 - 'BARGED'
 - 'CONTACT_FLOW_DISCONNECT'
 - 'OTHER'
 - 'OUTBOUND_DESTINATION_ENDPOINT_ERROR'
 - 'OUTBOUND_RESOURCE_ERROR'

- 'OUTBOUND_ATTEMPT_FAILED'
- 'EXPIRED'
- 'AGENT_NETWORK_DISCONNECT'
- 'CUSTOMER_CONNECTION_NOT_ESTABLISHED'
- 'API'
- 'IDLE_DISCONNECT'
- 'SYSTEM_ERROR'
- 'AGENT_COMPLETED'
- 'TRANSFERRED'
- 'DISCARDED'
- ComparisonValue - "\$.ContactLens.PostChat.DisconnectReason"
- Negate - false

After Contact Work Duration

Parameters

- Operator - "NumberGreaterOrEqualTo" | "NumberLessOrEqualTo"
- Operands - A positive integer or zero.
- ComparisonValue - "\$.ContactLens.PostChat.Agent.AfterContactWorkDurationSecs"
- Negate - false

Routing Profile

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Routing profile ARNs.
- ComparisonValue - "\$.ContactLens.PostChat.Agent.RoutingProfile"
- Negate - false

PotentialDisconnectIssue

Parameters

- Operator - "CONTAINS_ANY" where it is matched by a possible operand or "EQUALS" where the operand has to have no value.
- Operands - A validated enum of possible values or no value.
- ComparisonValue - "\$.ContactLens.PostChat.PotentialDisconnectIssue"
- Negate - false

Custom User-Defined Segment Attribute

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of segment attribute values. The values must be a value in the pre-defined attribute resource.
- ComparisonValue - "\$.ContactLens.PostChat.SegmentAttributes.UserDefined.[KEY]"

The KEY must be an instance pre-defined attribute resource.

- Negate - false

Connect AI Agent

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of strings. Strings are either a list of different UUIDs by themselves or a list of strings prefixed with the same UUID, with different possible version numbers uuid:2 and uuid:3. For example:
 - Different UUIDs by themselves: ['30682191-2312-4c27-9fc1-752e7a399d52', '2b81c7d5-85bc-459f-b605-0f5280a4301e']
 - Prefixed with the same UUID with different possible version numbers: ['30682191-2312-4c27-9fc1-752e7a399d52:1', '30682191-2312-4c27-9fc1-752e7a399d52:2']
- ComparisonValue - "\$.ContactLens.PostChat.AiAgent.IdWithVersion"

- FilterClause -

```
{
  LogicOperator: 'AND', // has to be 'AND'
  Filters: [
    { // REQUIRED filter
      Type: 'AiAgentUseCase',
      Data: {
        Value: 'SelfService', // or 'AgentAssistance'
      },
    },
    { // OPTIONAL filter
      Type: 'AiAgentEscalatedToHuman',
      Data: 'TRUE', // or 'FALSE'
    },
  ],
},
```

- Negate - false

PostChat words or phrases - Exact match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of strings
- ComparisonValue – "\$.ContactLens.PostChat.ExactMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "PostChatContactPeriodSeconds", // Optional filter type. If
      omitted, the rule applies to the full contact
      "Data": { // Either "First" or "Last" needs to be specified.
        "First": number
      }
    },
    {
```

```

        "Type": "ParticipantRole",
        "Data": "CUSTOMER" | "AGENT" | "ANY"
      }
    ]
  }

```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostChat words or phrases - Semantic match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of strings
- ComparisonValue – "\$.ContactLens.PostChat.SemanticMatch.Transcript"
- FilterClause –

```

{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "PostChatContactPeriodSeconds", // Optional filter type. If
      omitted, the rule applies to the full contact
      "Data": { // Either "First" or "Last" needs to be specified.
        "First": number
      }
    },
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    }
  ]
}

```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostChat natural language - Semantic match

Parameters

- Operator - "EQUALS"
- Operands – A list with a single string
- ComparisonValue – "\$.ContactLens.PostChat.SemanticMatch.Phrase"
- FilterClause –

```
{
  Operator: "EQUALS",
  Operands: [
    "Customer called to change their address and agent proceeded with the
    address change during the call."
  ],
  ComparisonValue: "$.ContactLens.PostChat.SemanticMatch.Phrase",
  Negate: true or false
},
```

- Negate - true or false. If set to true, it means *If transcript does not contain any of the words mentioned in the Operands.*

PostChat words or phrases - Pattern match

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of pattern match objects. See [PatternMatch Operands](#).
- ComparisonValue – "\$.ContactLens.PostChat.PatternMatch.Transcript"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "PostChatContactPeriodSeconds", // Optional filter type. If
      omitted, the rule applies to the full contact
      "Data": { // Either "First" or "Last" needs to be specified.
        "First": number
      }
    }
  ]
}
```

```

    },
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
    {
      "Type": "PatternMatchLanguageFilter",
      "Data": "EN" | "ES" | "AR" | "DE" | "FR" | "HI" | "IT" | "PT" | "KO" | "JA" | "ZH"
    }
  ]
}

```

- Negate - true or false. If set to true, it means *If the transcript does not contain any of the words mentioned in the Operands.*

PostChat agent condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of agent IDs
- ComparisonValue – "\$.ContactLens.PostChat.Agent.AgentId"
- Negate - false

PostChat queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue IDs.
- ComparisonValue – "\$.ContactLens.PostChat.Queue.QueueId"
- Negate - false

PostChat no queue condition

Parameters

- Operator - "EQUALS"

- Operands – No value.
- ComparisonValue – "\$.ContactLens.PostChat.Queue.QueueId"
- Negate - false

PostChat contact attributes condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the contact attribute value.
- ComparisonValue – "\$.ContactLens.PostChat.ContactAttribute.*YOUR_ATTRIBUTE_KEY*"
- Negate - true/false. If set to true, it means *YOUR_ATTRIBUTE_KEY* does not equal to the attribute value specified in the Operands .

PostChat sentiment state condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "POSITIVE", "NEGATIVE", "NEUTRAL".
- ComparisonValue – "\$.ContactLens.PostChat.Sentiment.State"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}
```

- Negate - false

PostChat sentiment overall score condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostChat.Sentiment.OverallScore"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ],
}
```

- Negate - false

Postchat sentiment score shift condition

PostChat sentiment score shift condition has a compound condition format where Operator is an AND condition and its operands consist of two conditions that represent beginning and end score respectively.

Parameters that represent the beginning score

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostChat.Sentiment.Score.Beginning"
- FilterClause –

```
{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
```

```

    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ]
}

```

- Negate - false

Parameters that represent the end score

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer between -5 and 5, inclusive.
- ComparisonValue – "\$.ContactLens.PostChat.Sentiment.Score.End"
- FilterClause –

```

{
  "LogicOperator": "AND", // Only "AND" is supported
  "Filters": [
    {
      "Type": "ParticipantRole",
      "Data": "CUSTOMER" | "AGENT" | "ANY"
    },
  ]
}

```

- Negate - false

Following is an example:

```

{
  "Operator": "AND",
  "Operands": [
    {
      "Operator": "NumberGreaterOrEqualTo",
      "Operands": [
        2
      ],
      "ComparisonValue": "$.ContactLens.PostChat.Sentiment.Score.Beginning",
    }
  ]
}

```

```

    "FilterClause": {
      "LogicOperator": "AND",
      "Filters": [
        {
          "Type": "ParticipantRole",
          "Data": "AGENT"
        }
      ]
    },
    "Negate": false
  },
  {
    "Operator": "NumberGreaterOrEqualTo",
    "Operands": [
      3
    ],
    "ComparisonValue": "$.ContactLens.PostChat.Sentiment.Score.End",
    "FilterClause": {
      "LogicOperator": "AND",
      "Filters": [
        {
          "Type": "ParticipantRole",
          "Data": "AGENT"
        }
      ]
    },
    "Negate": false
  }
]
}

```

PostChat agent first response time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostChat.ResponseTime.Agent.FirstResponseTimeSecs"
- Negate - false

PostChat agent average response time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostChat.ResponseTime.Agent.AverageTimeSecs"
- Negate - false

PostChat agent longest response time condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostChat.ResponseTime.Agent.LongestTimeSecs"
- Negate - false

PostChat agent interaction duration condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.PostChat.Agent.AgentInteractionDurationSecs"
- Negate - false

OnEmailAnalysisAvailable

Agent Hierarchy

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Agent hierarchy ARNs.
- ComparisonValue - "\$.ContactLens.Email.Agent.HierarchyGroup.ARN"
- Negate - false

Initiation Method

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'INBOUND'
 - 'OUTBOUND'
 - 'AGENT_REPLY'
 - 'FLOW'
- ComparisonValue - "\$.ContactLens.Email.InitiationMethod"
- Negate - false

DisconnectReason

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A validated enum set of possible values. Possible values are as follows:
 - 'AGENT_DISCONNECT'
 - 'CONTACT_FLOW_DISCONNECT'
 - 'EXPIRED'
 - 'DISCARDED'

- 'OTHER'
- 'API'
- 'TRANSFERRED'
- ComparisonValue - "\$.ContactLens.Email.DisconnectReason"
- Negate - false

After Contact Work Duration

Parameters

- Operator - "NumberGreaterOrEqualTo" | "NumberLessOrEqualTo"
- Operands - A positive integer or zero.
- ComparisonValue - "\$.ContactLens.Email.Agent.AfterContactWorkDurationSecs"
- Negate - false

Routing Profile

Parameters

- Operator - "CONTAINS_ANY"
- Operands - Routing profile ARNs.
- ComparisonValue - "\$.ContactLens.Email.Agent.RoutingProfile"
- Negate - false

Custom User-Defined Segment Attribute

Parameters

- Operator - "CONTAINS_ANY"
- Operands - A list of segment attribute values. The values must be a value in the pre-defined attribute resource.
- ComparisonValue - "\$.ContactLens.Email.SegmentAttributes.UserDefined.[KEY]"

The KEY must be an instance pre-defined attribute resource.

- Negate - false or true

Email agent condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of agent IDs
- ComparisonValue – "\$.ContactLens.Email.Agent.AgentId"
- Negate - false

Email queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue IDs.
- ComparisonValue – "\$.ContactLens.Email.Queue.QueueId"
- Negate - false

Email no queue condition

Parameters

- Operator - "EQUALS"
- Operands – No value.
- ComparisonValue – "\$.ContactLens.Email.Queue.QueueId"
- Negate - false

Email contact attributes condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is the contact attribute value.
- ComparisonValue – "\$.ContactLens.Email.ContactAttribute.*YOUR_ATTRIBUTE_KEY*"

- Negate - true/false. If set to true, it means ***YOUR_ATTRIBUTE_KEY** does not equal to the attribute value specified in the Operands*.

Email agent interaction duration condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of number, array length can only be 1. Value is an integer with minimum value 1.
- ComparisonValue – "\$.ContactLens.Email.Agent.AgentInteractionDurationSecs"
- Negate - false

OnZendeskTicketCreate

Type condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "question", "incident", "problem", "task"
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.Type"
- Negate - true or false. If set to true, it means *If ticket type does not equal to the type specified in the Operands.*

Priority condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "low", "normal", "high", "urgent"
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.Priority"
- Negate - true or false. If set to true, it means *If ticket priority does not equal to the priority specified in the Operands.*

Status condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "open", "pending", "solved", "close", "hold"
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.Status"
- Negate - true or false. If set to true, it means *If ticket status does not equal to the status specified in the Operands..*

RequesterId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a requester id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.RequesterId"
- Negate - true or false. If set to true, it means *If requester id does not equal to the requester id specified in the Operands..*

SubmitterId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a submitter id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.SubmitterId"
- Negate - true or false. If set to true, it means *If submitter id does not equal to the submitter id specified in the Operands..*

AssigneeId condition

Parameters

- Operator - "EQUALS"

- Operands – An array of string, array length can only be 1. Value is an assignee id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.AssigneeId"
- Negate - true or false. If set to true, it means *If assignee id does not equal to the assignee id specified in the Operands..*

OrganizationId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an organization id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.OrganizationId"
- Negate - true or false. If set to true, it means *If organization id does not equal to the organization id specified in the Operands..*

BrandId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a brand id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.BrandId"
- Negate - true or false. If set to true, it means *If brand id does not equal to the brand id specified in the Operands.*

FormId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a form id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.FormId"
- Negate - true or false. If set to true, it means *If form id does not equal to the form id specified in the Operands.*

ExternalId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an external id.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.ExternalId"
- Negate - true or false. If set to true, it means *If external id does not equal to the external id specified in the Operands.*

Channel condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a channel string.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.Channel"
- Negate - true or false. If set to true, it means *If channel value does not equal to the channel string specified in the Operands.*

Tags condition

Parameters

- Operator - "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is a tag string.
- ComparisonValue – "\$.ThirdParty.Zendesk.TicketCreate.Tags"
- Negate - true or false. If set to true, it means *If tag value does not equal to the tag string specified in the Operands.*

OnZendeskTicketStatusUpdate

Type condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "question", "incident", "problem", "task"
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.Type"
- Negate - true or false. If set to true, it means *If ticket type does not equal to the type specified in the Operands.*

Priority condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "low", "normal", "high", "urgent"
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.Priority"
- Negate - true or false. If set to true, it means *If ticket priority does not equal to the priority specified in the Operands.*

Status condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is one of "open", "pending", "solved", "close", "hold"
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.Status"
- Negate - true or false. If set to true, it means *If ticket status does not equal to the status specified in the Operands..*

RequesterId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a requester id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.RequesterId"
- Negate - true or false. If set to true, it means *If requester id does not equal to the requester id specified in the Operands..*

SubmitterId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a submitter id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.SubmitterId"
- Negate - true or false. If set to true, it means *If submitter id does not equal to the submitter id specified in the Operands..*

AssigneeId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an assignee id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.AssigneeId"
- Negate - true or false. If set to true, it means *If assignee id does not equal to the assignee id specified in the Operands..*

OrganizationId condition

Parameters

- Operator - "EQUALS"

- Operands – An array of string, array length can only be 1. Value is an organization id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.OrganizationId"
- Negate - true or false. If set to true, it means *If organization id does not equal to the organization id specified in the Operands..*

BrandId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a brand id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.BrandId"
- Negate - true or false. If set to true, it means *If brand id does not equal to the brand id specified in the Operands.*

FormId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a form id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.FormId"
- Negate - true or false. If set to true, it means *If form id does not equal to the form id specified in the Operands.*

ExternalId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an external id.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.ExternalId"
- Negate - true or false. If set to true, it means *If external id does not equal to the external id specified in the Operands.*

Channel condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a channel string.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.Channel"
- Negate - true or false. If set to true, it means *If channel value does not equal to the channel string specified in the Operands.*

Tags condition

Parameters

- Operator - "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is a tag string.
- ComparisonValue – "\$.ThirdParty.Zendesk.StatusUpdate.Tags"
- Negate - true or false. If set to true, it means *If tag value does not equal to the tag string specified in the Operands.*

OnSalesforceCaseCreate

AccountId condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an account Id string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.AccountId"
- Negate - true or false. If set to true, it means *If account id does not equal to the account id specified in the Operands.*

Name condition

Parameters

- Operator - "EQUALS" or "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is an account Id string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Name"
- Negate - true or false. If set to true, it means *If name does not equal to or does not contain the priority specified in the Operands.*

Email condition

Parameters

- Operator - "EQUALS" or "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is an email string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Email"
- Negate - true or false. If set to true, it means *If email does not equal to or does not contain the email specified in the Operands.*

Phone condition

Parameters

- Operator - "EQUALS" or "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is a phone string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Phone"
- Negate - true or false. If set to true, it means *If phone does not equal to or does not contain the phone specified in the Operands.*

Company condition

Parameters

- Operator - "EQUALS" or "CONTAINS"

- Operands – An array of string, array length can only be 1. Value is a company string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Company"
- Negate - true or false. If set to true, it means *If company does not equal to or does not contain the company specified in the Operands.*

Type condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a type string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Type"
- Negate - true or false. If set to true, it means *If type does not equal to the type specified in the Operands.*

Reason condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a reason string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Reason"
- Negate - true or false. If set to true, it means *If reason does not equal to the reason specified in the Operands.*

Origin condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is an origin string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Origin"
- Negate - true or false. If set to true, it means *If origin does not equal to the origin specified in the Operands.*

Subject condition

Parameters

- Operator - "EQUALS" or "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is a subject string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Subject"
- Negate - true or false. If set to true, it means *If subject does not equal to or does not contain the subject specified in the Operands.*

Priority condition

Parameters

- Operator - "EQUALS"
- Operands – An array of string, array length can only be 1. Value is a priority string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Priority"
- Negate - true or false. If set to true, it means *If priority does not equal to the priority specified in the Operands.*

Description condition

Parameters

- Operator - "EQUALS" or "CONTAINS"
- Operands – An array of string, array length can only be 1. Value is a description string.
- ComparisonValue – "\$.ThirdParty.Salesforce.CaseCreate.Description"
- Negate - true or false. If set to true, it means *If description does not equal to or does not contain the description specified in the Operands.*

CustomAttributes condition

Parameters

- Operator - "EQUALS"

- Operands – An array of string, array length can only be 1. Value is the custom attribute value.
- ComparisonValue –
"\$\$.ThirdParty.Salesforce.CaseCreate.CustomAttribute.*YOUR_ATTRIBUTE_KEY*"
- Negate - true or false. If set to true, it means *If custom attribute does not equal to the custom attribute specified in the Operands.*

OnCaseCreate

Assigned queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue ids
- ComparisonValue – "\$.Case.Fields.assigned_queue"
- Negate - false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["queueId"],
  "ComparisonValue": "$.Case.Fields.assigned_queue",
  "Negate": false
}
```

Assigned user condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue ids
- ComparisonValue – "\$.Case.Fields.assigned_user"
- Negate - false

```
{
```

```
"Operator": "CONTAINS_ANY",
"Operands": ["userId1, userId2"],
"ComparisonValue": "$.Case.Fields.assigned_user",
"Negate": false
}
```

Case reason condition

Parameters

- Operator - "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case reason.
- ComparisonValue – "\$.Case.Fields.case_reason"
- Negate - true/false

```
{
  "Operator": "EQUALS",
  "Operands": ["Refund"],
  "ComparisonValue": "$.Case.Fields.case_reason",
  "Negate": false
}
```

Case status condition

Parameters

- Operator - "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case status.
- ComparisonValue – "\$.Case.Fields.status"
- Negate - false

```
{
  "Operator": "EQUALS",
  "Operands": ["open"],
  "ComparisonValue": "$.Case.Fields.status",
}
```

```
"Negate": false
}
```

Case summary condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case summary.
- ComparisonValue – "\$.Case.Fields.summary"
- Negate - false

```
{
  "Operator": "EQUALS",
  "Operands": ["open"],
  "ComparisonValue": "$.Case.Fields.summary",
  "Negate": false
}
```

Case date/time last closed condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.last_closed_datetime"
- Negate - false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.last_closed_datetime",
  "Negate": false
}
```

```
}
```

Case date/time updated condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.last_updated_datetime"
- Negate - false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.last_updated_datetime",
  "Negate": false
}
```

Case date/time opened condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.created_datetime"
- Negate - true/false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.created_datetime",
  "Negate": false
}
```

Case reference number condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.reference_number"
- Negate - true

```
{
  "Operator": "CONTAINS",
  "Operands": ["11111111"],
  "ComparisonValue": "$.Case.Fields.reference_number",
  "Negate": false
}
```

Case customer fields condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1.
- ComparisonValue – "\$.Case.Fields.custom_case_field_id"
- Negate - true/false

```
{
  "Operator": "EQUALS",
  "Operands": ["vip"],
  "ComparisonValue": "$.Case.Fields.custom_case_field_id",
  "Negate": false
}
```

Case template condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of template ids.
- ComparisonValue – "\$.Case.TemplateId"
- Negate - true/false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["templateId"],
  "ComparisonValue": "$.Case.TemplateId",
  "Negate": false
}
```

OnCaseUpdate

Assigned queue condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue ids
- ComparisonValue – "\$.Case.Fields.assigned_queue"
- Negate - false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["queueId"],
  "ComparisonValue": "$.Case.Fields.assigned_queue",
  "Negate": false
}
```

Assigned user condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of queue ids
- ComparisonValue – "\$.Case.Fields.assigned_user"
- Negate - false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["userId1, userId2"],
  "ComparisonValue": "$.Case.Fields.assigned_user",
  "Negate": false
}
```

Case reason condition

Parameters

- Operator - "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case reason.
- ComparisonValue – "\$.Case.Fields.case_reason"
- Negate - true/false

```
{
  "Operator": "EQUALS",
  "Operands": ["Refund"],
  "ComparisonValue": "$.Case.Fields.case_reason",
  "Negate": false
}
```

Case status condition

Parameters

- Operator - "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case status.
- ComparisonValue – "\$.Case.Fields.status"
- Negate - false

```
{
  "Operator": "EQUALS",
  "Operands": ["open"],
  "ComparisonValue": "$.Case.Fields.status",
  "Negate": false
}
```

Case summary condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is a case summary.
- ComparisonValue – "\$.Case.Fields.summary"
- Negate - false

```
{
  "Operator": "EQUALS",
  "Operands": ["open"],
  "ComparisonValue": "$.Case.Fields.summary",
  "Negate": false
}
```

Case date/time last closed condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.last_closed_datetime"
- Negate - false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.last_closed_datetime",
  "Negate": false
}
```

Case date/time updated condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.last_updated_datetime"
- Negate - false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.last_updated_datetime",
  "Negate": false
}
```

Case date/time opened condition

Parameters

- Operator - "NumberLessOrEqualTo" | "NumberGreaterOrEqualTo"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.created_datetime"
- Negate - true/false

```
{
  "Operator": "NumberLessOrEqualTo",
  "Operands": ["2023-11-11T11:11:11.111111Z"],
  "ComparisonValue": "$.Case.Fields.created_datetime",
  "Negate": false
}
```

Case reference number condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1. The value is interpreted as a numeric value.
- ComparisonValue – "\$.Case.Fields.reference_number"
- Negate - true

```
{
  "Operator": "CONTAINS",
  "Operands": ["11111111"],
  "ComparisonValue": "$.Case.Fields.reference_number",
  "Negate": false
}
```

Case customer fields condition

Parameters

- Operator - "CONTAINS" | "EQUALS"
- Operands – An array of strings in which the array length can only be 1.
- ComparisonValue – "\$.Case.Fields.custom_case_field_id"
- Negate - true/false

```
{
  "Operator": "EQUALS",
  "Operands": ["vip"],
  "ComparisonValue": "$.Case.Fields.custom_case_field_id",
  "Negate": false
}
```

Case template condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of template ids.
- ComparisonValue – "\$.Case.TemplateId"
- Negate - true/false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["templateId"],
  "ComparisonValue": "$.Case.TemplateId",
  "Negate": false
}
```

OnSlaBreach

Cases SLA name condition

Parameters

- Operator - "CONTAINS_ANY"
- Operands – A list of SLA names.
- ComparisonValue – "\$.RelatedItem.SlaConfiguration.Name"
- Negate - false

```
{
  "Operator": "CONTAINS_ANY",
  "Operands": ["highPrioritySla"],
  "ComparisonValue": "$.RelatedItem.SlaConfiguration.Name",
  "Negate": false
}
```

PatternMatch Operands

Each operand is an array of Pattern match objects.

PatternMatch Object

Type

- Description: The type of the pattern match object.
- Type: String
- Valid values: PLAIN | LIST | PROXIMITY | NUMERICAL
- Required: Yes

Value

- Description: Depending on the type, value type varies.
 - If type is PLAIN, Value is a string.
 - If type is LIST, Value is an array of PLAIN PatternMatch object.

- If type is PROXIMITY, Value is an object for format.

```
{  
  "Distance": number,  
  "IsWithin": boolean  
}
```

- If type is NUMERICAL, Value is an object for format.

```
{  
  "Decimal": boolean  
}
```

Connect Customer testing language

This section describes the Connect Customer Testing language and how to use it. The Testing language is a JSON-based representation of testing observation, events and, actions.

Use the Testing language to do the following:

- Create test cases for simulating your flows.
- Write test cases instead of using the test designer.

Connect Customer Testing language concepts

The following terms are used in the Testing language.

Observations

Observations represent each complete interaction that includes one observed event expected from the system and many actions to validate or simulate system behaviors.

Events

Events represent expected behaviors that would come from the system, such as a prompt, a bot message, or a Lambda call.

Actions

Actions represent what the testing framework should do in response to an event, such as sending DTMF, responding with text, asserting attribute values, or ending the test.

Actors

Actors represent roles to be played in the testing framework. When observing events, actors can be the system or agent, such as a play prompt coming from the system or an agent accepting the call. When simulating actions, actors can be the customer, system, or agent, such as simulating a customer input DTMF or utterance, or simulating a system response from a Lambda function.

Example test

The following example demonstrates a JSON string representing the content of a test case configuration. It illustrates a simple test case that mocks the hours of operation when the test is initiated and then validates that a welcome prompt is played. Additionally, it simulates a contact's DTMF input to be placed in a queue and connected to an agent. The test will then verify if the queue placement is correct before concluding the test case. Overall, this test case configuration consists of three interactions with three simulated actions. Following is an example of the JSON representation for this test case.

```
{
  "Version": "2019-10-30",
  "Metadata": {},
  "Observations": [
    {
      "Identifier": "TriggerHoursCheck",
      "Event": {
        "Identifier": "evt-hours",
        "Type": "TestInitiated",
        "Actor": "System",
        "Properties": {}
      },
      "Actions": [
        {
          "Identifier": "ActionId",
          "Type": "OverrideSystemBehavior",
          "Parameters": {
            "ActionType": "OverrideSystemBehavior",
            "Behavior": {
              "Type": "FlowAction",
              "Properties": {
                "ActionType": "CheckHoursOfOperation",
                "ActionParameters": {
                  "HoursOfOperationId": "arn:aws:connect:us-west-2:123456789012:instance/abc123/flow/BasicHours"
                }
              },
              "Strategy": {
                "Type": "SubstituteResource",
                "SubstituteArn": "arn:aws:connect:us-west-2:123456789012:instance/abc123/flow/AlwaysOnHours"
              }
            }
          }
        }
      ]
    }
  ]
}
```

```

        }
      },
      "Transitions": {}
    }
  ],
  "Transitions": {
    "NextObservations": [
      "Welcome Message"
    ]
  }
},
{
  "Identifier": "Welcome Message",
  "Event": {
    "Identifier": "evt-welcome",
    "Type": "MessageReceived",
    "Actor": "System",
    "Properties": {
      "Text": "Press 1 to be connected to an agent",
      "MatchingCriteria": { "Type": "Similarity" }
    }
  },
  "Actions": [
    {
      "Identifier": "Send dtmf input",
      "Type": "SendInstruction",
      "Parameters": {
        "ActionType": "SendInstruction",
        "Actor": "Customer",
        "Instruction": {
          "Type": "DtmfInput",
          "Properties": {
            "Value": "1"
          }
        }
      }
    },
    {
      "Identifier": "Trigger Queue Announcement",
      "Type": "SendInstruction",
      "Parameters": {
        "ActionType": "SendInstruction",
        "Actor": "Customer",
        "Instruction": {
          "Type": "DtmfInput",
          "Properties": {
            "Value": "1"
          }
        }
      }
    }
  ],
  "Transitions": {}
}
],
"Transitions": {
  "NextObservations": [
    "Trigger Queue Announcement"
  ]
}
}

```

```

    },
    {
      "Identifier": "Trigger Queue Announcement",
      "Event": {
        "Identifier": "evt-queue",
        "Type": "MessageReceived",
        "Actor": "System",
        "Properties": {
          "Text": "Thank you for calling. Your call is very important to us
and will be answered in the order it was received.",
          "MatchingCriteria": { "Type": "Similarity" }
        }
      },
      "Actions": [
        {
          "Identifier": "assertqueue",
          "Type": "Assert",
          "Parameters": {
            "Namespace": "$.Queue.Name",
            "Operator": "Equals",
            "Operand": "Basic Queue"
          },
          "Transitions": {
            "NextAction": "endthetest"
          }
        },
        {
          "Identifier": "endthetest",
          "Type": "TestControl",
          "Parameters": {
            "ActionType": "TestControl",
            "Command": {
              "Type": "EndTest"
            }
          },
          "Transitions": {}
        }
      ],
      "Transitions": {
        "NextObservations": []
      }
    }
  ]

```

```
}
```

Observations in the Connect Customer Testing language

Observations represent each complete interaction that includes one observed event expected from the system and many actions to validate or simulate system behaviors.

Parameters

- Version - The API version for the testing language, such as 2019-10-30.
- Metadata - Optional object containing UI-specific or non-functional data
- Observations - An array of observation objects that define the test flow

Observation object

Each observation consists of an event to observe and actions to execute when that event occurs.

- Identifier - Unique identifier for the observation
- Event - Defines the expected event from the system to observe
- Actions - Array of actions to execute when the event is observed
- Usage - Defines how many times this observation should be matched
 - Type: Specifies how many times the observation must match. Valid values are Exactly, AtLeast, AtMost, or Always (case-sensitive).
 - Times: An integer value for the count. This field is required for all types except Always.
- Transitions - Optional object defining flow control to next observations
 - NextObservations: Array of observation IDs to transition to

```
{
  "Version": "2019-10-30",
  "Metadata": { ... }, // Metadata to be used for data which is used for UI or any non-
runtime impacting data as required.
  "Observations": [
    {
      "Identifier": "unique identifier",
      "Event": { ... },
      "Actions": [
```

```

        {
            "Identifier": "ActionId",
            "Type": "ActionType", // Action type could be of any type mentioned in
recap (ObserveEvent, SendInstruction, Assertion, OverrideSystemBehavior, EndTest)
            "Parameters": {...},
            "Transitions" : {...}
        },
        ...
    ],
    "Usage": { "Type": "Exactly", "Times": 1 },
    "Transitions" : {
        "NextObservations": ["string-id", "string-id", "string-id"]
    }
},
// Additional observations...
]
}

```

Events in the Connect Customer Testing language

Events represent expected behaviors from the system that the test framework observes.

Contents

- [Test initiated](#)
- [Test completed](#)
- [Message received](#)
- [Flow action started](#)

Test initiated

Triggered when the test execution begins. This is typically used to set up initial conditions such as override system behaviors before the actual flow execution starts.

Parameters

- Identifier - Unique identifier for the event. (API need to specify this identifier in order for the UI to render properly)
- Type - Must always be TestInitiated.
- Actor - Must always be System. This indicates that the event originates from the testing system.

- Properties - Empty object. No additional properties are required.

```
{
  "Identifier": "unique identifier",
  "Type": "TestInitiated",
  "Actor": "System",
  "Properties": {}
}
```

Test completed

Triggered when the test execution ends. This event is observed when the test is terminated.

Parameters

- Type - Must always be TestCompleted.
- Actor - Must always be System. This indicates that the event originates from the testing system.
- Properties - Empty object. No additional properties are required.

```
{
  "Identifier": "unique identifier",
  "Type": "TestCompleted",
  "Actor": "System",
  "Properties": {}
}
```

Message received

Observes when the system plays a prompt or send any voice response to the simulated customer. This event can match messages using different criteria.

Parameters

- Identifier - Unique identifier for the event (API need to specify this identifier in order for the UI to render properly)
- Type - Must always be MessageReceived.
- Actor - Must always be System. This indicates the event originate from the testing system.

- **Properties** - Empty object. Object containing the following message details:
 - **PromptId** (Optional): Specific prompt ID or ARN to match
 - **Text** (Optional): Text content of the message to match
 - **SSML** (Optional): SSML content to match
 - **Media** (Optional): External media source details
 - **Uri**: Location of the media file
 - **SourceType**: Source from which media is fetched such as S3
 - **MediaType**: Type of media such as "audio/mpeg"
- **MatchingCriteria** - Defines how to match the message:
 - **Similarity**: Uses semantic matching to find similar messages
 - **Inclusion**: Checks if the observed message contains the specified text

```
{
  "Identifier": "unique identifier",
  "Type": "MessageReceived",
  "Actor": "System",
  "Properties": {
    "PromptId": "string", // [Optional] A prompt ID or prompt ARN to play to
    the participant along with gathering input. May not be specified if Text or SSML is
    also specified. Must be specified either statically or as a single valid JSONPath
    identifier.
    "Text": "string", // An optional string that defines text to send to the
    participant along with gathering input. May not be specified if PromptId or SSML is
    also specified. May be specified statically or dynamically.
    "SSML": "string", // An optional string that defines SSML to send to the
    participant along with gathering input. May not be specified if Text or PromptId is
    also specified May be specified statically or dynamically.
    "Media": { // An optional object that defines an external media source
      "Uri": "string", // Location of the message
      "SourceType": "string", // The source from which the message will be
      fetched. The only supported type is S3
      "MediaType": "string" // The type of the message to be played. The only
      supported type is Audio
    },
    "MatchingCriteria": { "Type": "Similarity" } // Alternatively, use "Inclusion".
  }
}
```

Flow action started

Observes when a specific flow action begins execution. This allows you to detect when particular flow actions are executed during the simulation.

Invoke Lambda function

Observes when a Lambda function invocation action starts.

Parameters

- Identifier - Unique identifier for the event (In preview, API need to specify this identifier in order for the UI to render properly)
- Type - Must always be `FlowActionStarted`.
- Actor - Must always be `System`. This indicates that the event originates from the testing system.
- Properties:
 - ActionType - Must always be `InvokeLambdaFunction`.
 - ActionParameters:
 - LambdaFunctionARN: The ARN of the Lambda function being invoked

```
{
  "Identifier": "unique identifier",
  "Type": "FlowActionStarted",
  "Actor": "System",
  "Properties": {
    "ActionType": "InvokeLambdaFunction", // V2 JSON action type.
    "ActionParameters": { // V2 JSON action parameters.
      "LambdaFunctionARN": "string"
    }
  }
}
```

Check hours of operation

Observes when the flow checks hours of operation.

Parameters

- Identifier - Unique identifier for the event (API need to specify this identifier in order for the UI to render properly)
- Type - Must always be `FlowActionStarted`.
- Actor - Must always be `System`. This indicates that the event originates from the testing system.
- Properties:
 - ActionType - Must always be `CheckHoursOfOperation`.
 - ActionParameters:
 - HoursOfOperationId: The ID or ARN of the hours of operation resource being checked

```
{
  "Identifier": "unique identifier",
  "Type": "FlowActionStarted",
  "Actor": "System",
  "Properties": {
    "ActionType": "CheckHoursOfOperation", // V2 JSON action type.
    "ActionParameters": { // V2 JSON action parameters.
      "HoursOfOperationId": "string"
    }
  }
}
```

Transfer contact to queue

Observes when a contact is being transferred to a queue.

Parameters

- Identifier - Unique identifier for the event (API need to specify this identifier in order for the UI to render properly)
- Type - Must always be `FlowActionStarted`.
- Actor - Must always be `System`. This indicates that the event originates from the testing system.
- Properties:
 - ActionType - Must always be `TransferContactToQueue`.
 - ActionParameters:

- **QueueId:** The ID or ARN of the target queue
- **AgentId (Optional):** Specific agent ID if transferring to a particular agent

```
{
  "Identifier": "unique identifier",
  "Type": "FlowActionStarted",
  "Actor": "System",
  "Properties": {
    "ActionType": "TransferContactToQueue", // V2 JSON action type.
    "ActionParameters": { // V2 JSON action parameters.
      "QueueId": "string",
      "AgentId": "string"
    }
  }
}
```

Connect participant with Lex bot

Observes when a participant is being connected to a Lex bot.

Parameters

- **Identifier** - Unique identifier for the event (API need to specify this identifier in order for the UI to render properly)
- **Type** - Must always be `FlowActionStarted`.
- **Actor** - Must always be `System`. This indicates that the event originates from the testing system.
- **Properties:**
 - **ActionType** - Must always be `ConnectParticipantWithLexBot`.
 - **ActionParameters:**
 - **LexV2Bot:** Object containing bot details
 - **AliasArn:** The ARN of the Lex V2 bot alias

```
{
  "Identifier": "unique identifier",
  "Type": "FlowActionStarted",
  "Actor": "System",
  "Properties": {
```

```
    "ActionType": "ConnectParticipantWithLexBot",
    "ActionParameters": {
      "LexV2Bot": {
        "AliasArn": "string"
      }
    }
  }
}
```

Actions in the Connect Customer Testing language

Actions represent what the testing framework executes in response to observed events.

Assertion

Validates that a specific attribute or value in the flow matches expected criteria.

Parameters

- Identifier: Unique identifier for the action
- Type: Must be Assert
- Parameters:
 - Namespace: JSON path to the attribute value to check
 - Operator: Comparison operator to use (see [Supported operators](#))
 - Operand: Expected value to compare against
- Transitions
 - NextAction: The unique identifier for the next action

Supported operators

- Equals: Exact match
- TextStartsWith: Text begins with the operand
- TextEndsWith: Text ends with the operand
- TextContains: Text contains the operand
- NumberGreaterThan: Numeric value is greater than operand
- NumberGreaterOrEqualTo: Numeric value is greater than or equal to operand

- **NumberLessThan:** Numeric value is less than operand
- **NumberLessOrEqualTo:** Numeric value is less than or equal to operand
- **Exists:** Checks if the attribute exists (operand not required)

```
{
  "Identifier": "ActionId",
  "Type": "Assert",
  "Parameters": {
    "Namespace": "string", // Data path to fetch.
    "Operator": "string", // Comparison operator.
    "Operand": "string" // Expected value to compare.
  },
  "Transitions": { "NextAction": "string" }
}
```

Send Instruction

Simulates customer input during the test execution. This action allows you to send DTMF tones or voice input as if a customer were interacting with the flow.

DTMF input

Simulates a customer pressing keys on their phone keypad.

Parameters

- **Identifier:** Unique identifier for the action
- **Type:** Must be `SendInstruction`
- **Parameters:**
 - **ActionType:** Must be `SendInstruction`
 - **Actor:** `Customer` indicates this simulates customer behavior
 - **Instruction:** Object defining the instruction type
 - **Type:** Must be `DtmfInput`
 - **Properties:**
 - **Value:** String or number representing the DTMF input (e.g., "1", "#", "*")
- **Transitions:**

- **NextAction:** The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "SendInstruction",
  "Parameters": {
    "ActionType": "SendInstruction",
    "Actor" : "Customer",
    "Instruction": {
      "Type": "DtmfInput",
      "Properties": {
        "Value": "string"
      }
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Text and Utterance Input

Simulates customer voice or text input. This is used for Lex bot or AI agent interactions.

Parameters

- **Identifier:** Unique identifier for the action
- **Type:** Must be `SendInstruction`
- **Parameters:**
 - **ActionType:** Must be `SendInstruction`
 - **Actor:** `Customer` indicates this simulates customer behavior
 - **Instruction:** Object defining the instruction
 - **Type:** Must be `Utterance`
 - **Properties:**
 - **Text (Optional):** Plain text input to send
 - **SSML (Optional):** SSML-formatted input to send
 - **LanguageCode:** Language code for the input (e.g., "en-US")
- **Transitions:**
 - **NextAction:** The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "SendInstruction",
  "Parameters": {
    "ActionType": "SendInstruction",
    "Actor" : "Customer",
    "Instruction": {
      "Type": "Utterance",
      "Properties": {
        "Text": "string", // An optional string that defines text to send to
        the participant along with gathering input. May not be specified if PromptId or SSML
        is also specified. May be specified statically or dynamically.
        "SSML": "string", // An optional string that defines SSML to send to
        the participant along with gathering input. May not be specified if Text or PromptId
        is also specified May be specified statically or dynamically.,
        "LanguageCode": "en-US"
      }
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Disconnect

Simulates customer ending the call.

Parameters

- Identifier: Unique identifier for the action
- Type: Must be `SendInstruction`
- Parameters:
 - ActionType: Must be `SendInstruction`
 - Actor: `Customer` indicates this simulates customer behavior
 - Instruction: Object defining the instruction
 - ActionType: Must be `SendInstruction`
 - Type: Must be `Disconnect`
- Transitions:
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "SendInstruction",
  "Parameters": {
    "ActionType": "SendInstruction",
    "Actor" : "Customer",
    "Instruction": {
      "Type": "Disconnect"
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Override system behavior

Modifies how specific flow actions behave during test execution. This allows you to mock external dependencies, substitute resources, or simulate specific scenarios without modifying the actual flow.

InvokeLambdaFunction

Override Lambda function behavior by substituting with a different function or mocking the response.

Substitute resource strategy

Redirects Lambda invocations to a different function ARN.

Parameters

- Identifier: Unique identifier for the action
- Type: Must be `OverrideSystemBehavior`
- Parameters:
 - ActionType: Must be `OverrideSystemBehavior`
 - Behavior: Object defining the behavior to override
 - Type: Must be `FlowAction`
 - Properties:
 - ActionType: Must be `InvokeLambdaFunction`
 - ActionParameters:

- LambdaFunctionARN: The ARN of the Lambda function to override
- Strategy: Object defining the override strategy
 - Type: Must be SubstituteResource
 - SubstituteArn: ARN of the replacement Lambda function to use
- Transitions:
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType" : "InvokeLambdaFunction",
        "ActionParameters": {
          "LambdaFunctionARN" : "string"
        },
        "Strategy": {
          "Type": "SubstituteResource",
          "SubstituteArn": "string"
        }
      }
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Mock response strategy - Success

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType" : "InvokeLambdaFunction",
```

```

        "ActionParameters": {
            "LambdaFunctionARN" : "string"
        },
        "Strategy": {
            "Type": "MockResponse",
            "Response": {
                "Type" : "ExecutionResult",
                "ExecutionResult" : {
                    "DelaySeconds" : Number,
                    "LoadedData" : "serialized JSON"
                }
            }
        }
    },
    "Transitions": { "NextAction": "string" }
}

```

Mock response strategy - Error

Simulates a Lambda function error without actual invocation.

```

{
    "Identifier": "ActionId",
    "Type": "OverrideSystemBehavior",
    "Parameters": {
        "ActionType": "OverrideSystemBehavior",
        "Behavior": {
            "Type": "FlowAction",
            "Properties": {
                "ActionType" : "InvokeLambdaFunction",
                "ActionParameters": {
                    "LambdaFunctionARN" : "lambda-arn-to-mock"
                }
            },
            "Strategy": {
                "Type": "MockResponse",
                "Response": {
                    "Type" : "Error",
                    "Error" : {
                        "DelaySeconds" : Number,
                        "Value" : "TimeLimitExceeded|NoMatchingError"
                    }
                }
            }
        }
    }
}

```

```

        }
    }
}
},
"Transitions": { "NextAction": "string" }
}

```

CheckHoursOfOperation

Override hours of operation checks to test different time-based scenarios.

Substitute resource strategy

Redirects hours of operation checks to a different hours of operation configuration.

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `OverrideSystemBehavior`
- Parameters:
 - ActionType: Must be `OverrideSystemBehavior`
 - Behavior: Object defining the behavior to override
 - Type: Must be `FlowAction`
 - Properties
 - ActionType: Must be `CheckHoursOfOperation`
 - ActionParameters:
 - HoursOfOperationId: The ID/ARN of the hours of operation to override
 - Strategy:
 - Type: Must be `SubstituteResource`
 - SubstituteArn: ARN of the replacement hours of operation resource
- Transitions:
 - NextAction: The unique identifier for the next action

```

{
    "Identifier": "ActionId",

```

```

    "Type": "OverrideSystemBehavior",
    "Parameters": {
      "ActionType": "OverrideSystemBehavior",
      "Behavior": {
        "Type": "FlowAction",
        "Properties": {
          "ActionType" : "CheckHoursOfOperation",
          "ActionParameters": {
            "HoursOfOperationId" : "string"
          },
          "Strategy": {
            "Type": "SubstituteResource",
            "SubstituteArn": "string"
          }
        }
      }
    },
    "Transitions": { "NextAction": "string" }
  }
}

```

Mock response strategy - Success

Returns a predefined hours of operation check result.

```

{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType": "CheckHoursOfOperation",
        "ActionParameters": {
          "HoursOfOperationId": "string"
        },
        "Strategy": {
          "Type": "MockResponse",
          "Response": {
            "Type": "ExecutionResult",
            "ExecutionResult": {
              "Value": "InHours|OutOfHours"
            }
          }
        }
      }
    }
  }
}

```

```

        }
    }
},
"Transitions": {
    "NextAction": "string"
}
}

```

Mock response strategy - Error

Simulates an error during hours of operation check.

```

{
    "Identifier": "ActionId",
    "Type": "OverrideSystemBehavior",
    "Parameters": {
        "ActionType": "OverrideSystemBehavior",
        "Behavior": {
            "Type": "FlowAction",
            "Properties": {
                "ActionType": "CheckHoursOfOperation",
                "ActionParameters": {
                    "HoursOfOperationId": "arn:aws:connect:us-
west-2:489887583805:instance/0b6b36d4-add8-4fc2-bb5b-72b8e4042838/operating-
hours/90d61ae9-79b5-4940-91e5-3e38f8383958"
                },
                "Strategy": {
                    "Type": "MockResponse",
                    "Response": {
                        "Type": "Error",
                        "Error": {
                            "Value": "NoMatchingError"
                        }
                    }
                }
            }
        }
    },
    "Transitions": {
        "NextAction": "string"
    }
}

```

ConnectParticipantWithLexBot

Override Lex bot behaviors to use a different bot for testing or mock responses.

Substitute resource strategy

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `OverrideSystemBehavior`
- Parameters
 - ActionType - Must be `OverrideSystemBehavior`
 - Behavior:
 - Type: Must be `FlowAction`
 - Properties:
 - ActionType - Must be `ConnectParticipantWithLexBot`
 - ActionParameters:
 - LexV2Bot: Object containing the bot to override
 - AliasArn: ARN of the Lex bot alias to override
 - Strategy:
 - Type: Must be `SubstituteResource`
 - SubstituteArn: ARN of the replacement Lex bot alias
- Transitions:
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType" : "ConnectParticipantWithLexBot",
        "ActionParameters" : {
          "LexV2Bot": {
```

```

        "AliasArn": "string"
      }
    },
    "Strategy": {
      "Type": "SubstituteResource",
      "SubstituteArn": "string"
    }
  }
},
"Transitions": { "NextAction": "string" }
}

```

Mock response strategy - Success

Returns a predefined successful response without invoking the actual Lex bot.

```

{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType": "ConnectParticipantWithLexBot",
        "ActionParameters": {
          "LexV2Bot": {
            "AliasArn": "string"
          }
        }
      },
      "Strategy": {
        "Type": "MockResponse",
        "Response": {
          "Type": "ExecutionResult",
          "ExecutionResult": {
            "DelaySeconds": Number,
            "LoadedData": "serialized JSON"
          }
        }
      }
    }
  }
},
}

```

```
"Transitions": { "NextAction": "string" }
}
```

Mock response strategy - Error

Simulates a Lex bot with error without actual invocation.

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType": "ConnectParticipantWithLexBot",
        "ActionParameters": {
          "LexV2Bot": {"AliasArn": "string"}
        },
        "Strategy": {
          "Type": "MockResponse",
          "Response": {
            "Type": "Error",
            "Error": {
              "DelaySeconds": Number,
              "Value": "TimeLimitExceeded|NoMatchingError"
            }
          }
        }
      }
    }
  }
},
  "Transitions": { "NextAction": "string" }
}
```

TransferContactToQueue

Override queue transfer behavior to substitute queues or simulate transfer failures.

Substitute resource strategy

Redirects queue transfers to a different queue.

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `OverrideSystemBehavior`
- Parameters
 - ActionType - Must be `OverrideSystemBehavior`
 - Behavior
 - Type: `FlowAction`
 - Properties
 - ActionType - Must be `TransferContactToQueue`
 - ActionParameters:
 - QueueId: ID/ARN of the queue to override
 - Strategy:
 - Type: Must be `SubstituteResource`
 - SubstituteArn: ARN of the replacement queue
- Transitions:
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType" : "TransferContactToQueue",
        "ActionParameters" : {
          "QueueId": "string"
        },
        "Strategy": {
          "Type": "SubstituteResource",
          "SubstituteArn": "string"
        }
      }
    }
  }
}
```

```

    },
    "Transitions": { "NextAction": "string" }
  }

```

Mock response strategy - Error

Simulates queue transfer failures for testing error paths.

```

{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType": "TransferContactToQueue",
        "ActionParameters": {
          "QueueId": "string"
        },
        "Strategy": {
          "Type": "MockResponse",
          "Response": {
            "Type": "Error",
            "Error": {
              "Value": "QueueAtCapacity|NoMatchingError"
            }
          }
        }
      }
    }
  },
  "Transitions": {
    "NextAction": "string"
  }
}

```

DequeueAndTransferToQueue

Override behavior when dequeuing a contact and transferring to another queue.

Substitute resource strategy

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `OverrideSystemBehavior`
- Parameters
 - ActionType - Must be `OverrideSystemBehavior`
 - Behavior:
 - Type: `FlowAction`
 - Properties:
 - ActionType - Must be `DequeueAndTransferToQueue`
 - ActionParameters:
 - QueueId: ID/ARN of the queue to override
 - Strategy:
 - Type: Must be `SubstituteResource`
 - SubstituteArn: ARN of the replacement queue
- Transitions
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType" : "DequeueAndTransferToQueue",
        "ActionParameters" : {
          "QueueId": "string"
        },
        "Strategy": {
          "Type": "SubstituteResource",
          "SubstituteArn": "string"
        }
      }
    }
  }
}
```

```

    }
  }
},
"Transitions": { "NextAction": "string" }
}

```

Mock response strategy - Error

Simulates dequeue and transfer failures.

```

{
  "Identifier": "ActionId",
  "Type": "OverrideSystemBehavior",
  "Parameters": {
    "ActionType": "OverrideSystemBehavior",
    "Behavior": {
      "Type": "FlowAction",
      "Properties": {
        "ActionType": "DequeueAndTransferToQueue",
        "ActionParameters": {
          "QueueId": "string"
        },
        "Strategy": {
          "Type": "MockResponse",
          "Response": {
            "Type": "Error",
            "Error": {
              "Value": "QueueAtCapacity|NoMatchingError"
            }
          }
        }
      }
    }
  },
  "Transitions": {
    "NextAction": "string"
  }
}

```

Test Control Actions

Special actions that control test execution flow and debugging.

End Test

Explicitly terminates the test execution. Use this when you want to end the test at a specific point rather than waiting for natural completion.

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `TestControl`
- Parameters:
 - Action Type: Must be `TestControl`
 - Command: Object defining the control command
 - Type: Must be `EndTest`
- Transitions
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "TestControl",
  "Parameters": {
    "ActionType": "TestControl",
    "Command": {
      "Type": "EndTest"
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Log Data

Captures and logs specific attribute values during test execution for debugging and validation purposes. The logged data appears in the test execution results.

Parameters

- Identifier - Unique identifier for the action
- Type - Must be `TestControl`
- Parameters:

- Action Type: Must be `TestControl`
- Command: Object defining the control command
 - Type: Must be `LogData`
 - Properties:
 - Expressions: Object containing key-value pairs where:
 - Key: A descriptive label for the logged value
 - Value: JSON path to extract the attributes (e.g., `$.Queue.Name`)
- Transitions
 - NextAction: The unique identifier for the next action

```
{
  "Identifier": "ActionId",
  "Type": "TestControl",
  "Parameters": {
    "ActionType": "TestControl",
    "Command": {
      "Type": "LogData",
      "Properties": {
        "Expressions": {
          "string": "string", //"myContactId": "$.ContactId"
          ...
        }
      }
    }
  },
  "Transitions": { "NextAction": "string" }
}
```

Amazon CX Designer SDK

Agentic CX Designer SDK is a programmatic API for managing ACXD workspace resources like applications, flows, knowledge bases, and more without using the agentic CX designer workspace.

Agentic CX Designer (ACXD) SDK is a RESTful API that provides programmatic access to all workspace resources. It enables developers to build, deploy, and manage conversational AI applications through code. ACXD SDK is authenticated by API key that can be generated by Account Admin.

Contents

- [Design Philosophy](#)
- [Getting Started](#)
- [API Reference](#)
- [Concepts](#)
- [Common Errors](#)

Design Philosophy

As conversational AI scales from prototypes to production fleets, teams need automation: repeatable deployments, and programmatic testing. ACXD SDK brings infrastructure-as-code workflows to Agentic CX Designer. The same resources you design visually in the agentic CX designer workspace can be created, updated, and deployed through code.

Everything you do in the agentic CX designer workspace, you can do through the SDK. The same permissions apply, the same resources are accessible, and changes made through either path are immediately visible in the other.

Getting Started

This guide walks you through setting up authentication and making your first Agentic CX Designer SDK API call.

Prerequisites

- An Agentic CX Designer workspace with an Administrator role

- Access to Agentic CX Designer workspace (Admin Hub)

Authentication

ACXD SDK uses API key authentication. To get an API key you need a programmatic user.

Step 1: Create a Programmatic User

Programmatic users are machine identities that can authenticate with the ACXD SDK. Only account administrators can create them.

1. In the Agentic CX Designer workspace, navigate to Admin Hub > Programmatic Users
2. Click **Create Programmatic User**
3. Provide a name and an optional description
4. Assign permissions via Role Configuration

Role Configuration

Each programmatic user has a `roleConfig` that determines their access:

Account-level role

Grants full access across all workspaces in the account.

```
{
  "roleConfig": {
    "accountRole": "administrator"
  }
}
```

Workspace-scoped roles

Grants specific permissions per workspace

```
{
  "roleConfig": {
    "workspaceRoles": [
      { "workspaceId": "your-workspace-id", "roleId": "role-uuid" }
    ]
  }
}
```

```
}  
}
```

Workspace-scoped users can be assigned pre-defined roles (administrator, developer, content manager, read-only) or a custom role configured under Roles in Admin Hub.

Step 2: Generate an API Key

Once a programmatic user exists, generate an API key for it:

1. In Admin Hub → Programmatic Users, select your user
2. Click Generate API Key
3. Copy the full key immediately, it is shown only once

The key format is: `acxd_live_<prefix>.<secret>`

Important

Store your API key securely. It cannot be retrieved after creation. You can generate up to 2 keys per programmatic user.

Step 3: Install the SDK

```
npm install amazon-connect-acxd-sdk
```

Step 4: Make your First Call

With your API key and workspace ID, you can make requests to the ACXD SDK.

```
import { AgenticCXDesignerClient, ListContextVariablesCommand } from 'amazon-connect-acxd-sdk';  
  
const client = new AgenticCXDesignerClient({  
  apiKey: 'acxd_live_...',  
  workspaceId: 'your-workspace-uuid', // required for workspace-scoped operations  
});
```

```
const response = await client.send(new ListContextVariablesCommand({}));
console.log(response.items);
```

For account-level operations (e.g., managing programmatic users, workspaces), `workspaceId` is not required.

Permissions

The API key is just a credential - it carries no permissions itself. Permissions are resolved at request time from the programmatic user's assigned role. If the role is updated in Admin Hub, the change takes effect immediately.

Next Steps

- **API Reference:** Explore all available operations
- **Concepts:** Understand workspaces, applications, and flows
- **Error Handling:** Handle errors and retries gracefully

API Reference

All operations follow the Command pattern. Import the client and the command, then call `client.send(command)`.

```
import { AgenticCXDesignerClient, <CommandName> } from "amazon-connect-acxd-sdk";

const client = new AgenticCXDesignerClient({
  apiKey: 'acxd_live_...',
  workspaceId: 'your-workspace-uuid',
});

const response = await client.send(new <CommandName>({ ... }));
```

Most List operations use token-based pagination:

```
const response = await client.send(new ListApplicationsCommand({
  maxResults: 50,
  nextToken: '...',
}));
```

API Operations

The following section lists the API operations by resource.

API Token Operations

- ListApiTokens
- CreateApiToken
- DeleteApiToken

Application Operations

- ListApplications
- CreateApplication
- GetApplication
- UpdateApplication
- DeleteApplication

Application Build Operations

- ListApplicationBuilds
- CreateApplicationBuild
- GetApplicationBuild
- GetApplicationBuildDiff

Application Deployment Operations

- ListApplicationDeployments
- CreateApplicationDeployment
- GetApplicationDeployment
- UpdateApplicationDeployment
- DeleteApplicationDeployment

Analytics Tag Operations

- `ListAnalyticsTags`
- `CreateAnalyticsTag`
- `UpdateAnalyticsTag`
- `DeleteAnalyticsTag`

Context Variables Operations

- `ListContextVariables`
- `CreateContextVariable`
- `UpdateContextVariable`
- `DeleteContextVariable`

Conversation Operations

- `ListConversations`
- `GetConversation`

Data Request Operations

- `ListDataRequests`
- `CreateDataRequest`
- `GetDataRequest`
- `UpdateDataRequest`
- `DeleteDataRequest`

Downloads Operations

- `GetDownload`

Flow Operations

- `ListFlows`
- `CreateFlow`
- `GetFlow`

- UpdateFlow
- DeleteFlow

Guardrail Operations

- ListGuardrails
- CreateGuardrail
- GetGuardrail
- UpdateGuardrail
- DeleteGuardrail
- TestGuardrail
- ListGuardrailEvents

Knowledge Base Operations

- ListKnowledgeBases
- CreateKnowledgeBase
- GetKnowledgeBase
- UpdateKnowledgeBase
- DeleteKnowledgeBase
- CloneKnowledgeBase
- PublishKnowledgeBase
- GetKnowledgeBasePublication
- ListKnowledgeBasePublications

Knowledge Base Article Operations

- ListKnowledgeBaseArticles
- CreateKnowledgeBaseArticle
- GetKnowledgeBaseArticle
- UpdateKnowledgeBaseArticle
- DeleteKnowledgeBaseArticle

Knowledge Base Document Operations

- `ListKnowledgeBaseDocuments`
- `GetKnowledgeBaseDocument`
- `PutKnowledgeBaseDocument`
- `DeleteKnowledgeBaseDocument`

Live Sync Script Operations

- `ListLiveSyncScripts`
- `CreateLiveSyncScript`
- `GetLiveSyncScript`
- `UpdateLiveSyncScript`
- `DeleteLiveSyncScript`
- `ListLiveSyncScriptBuilds`
- `CreateLiveSyncScriptBuild`
- `GetLiveSyncScriptBuild`
- `ListLiveSyncScriptDeployments`
- `CreateLiveSyncScriptDeployment`
- `GetLiveSyncScriptDeployment`
- `UpdateLiveSyncScriptDeployment`
- `DeleteLiveSyncScriptDeployment`

Log Operations

- `QueryLogs`

Modality Operations

- `ListModalities`
- `CreateModality`
- `GetModality`
- `UpdateModality`

- DeleteModality

Programmatic User Operations

- ListProgrammaticUsers
- CreateProgrammaticUser
- GetProgrammaticUser
- UpdateProgrammaticUser
- DeleteProgrammaticUser

Role Operations

- ListRoles
- CreateRole
- GetRole
- UpdateRole
- DeleteRole
- GetRolePermissions

Secret Operations

- ListSecrets
- CreateSecret
- GetSecret
- UpdateSecret
- DeleteSecret

Slot Type Operations

- ListSlotTypes
- CreateSlotType
- GetSlotType
- UpdateSlotType

- DeleteSlotType

Team Operations

- GetTeam

Trail Operations

- StartTrailQuery
- GetTrailQueryResults

User Operations

- ListUsers
- CreateUser
- GetUser
- UpdateUser
- DeleteUser

Version Operations

- ListResourceVersions
- GetResourceVersion

Workspace Operations

- ListWorkspaces
- CreateWorkspace
- GetWorkspace
- UpdateWorkspace
- DeleteWorkspace

Contents

- [Analytics Tags](#)

- [API Tokens](#)
- [Applications](#)
- [Application Builds](#)
- [Application Deployments](#)
- [Context Variables](#)
- [Conversations](#)
- [Data Requests](#)
- [Downloads](#)
- [Flows](#)
- [Guardrails](#)
- [Knowledge Bases](#)
- [Knowledge Base Articles](#)
- [KnowledgeBase Documents](#)
- [Live Sync Scripts](#)
- [Logs](#)
- [Modalities](#)
- [Programmatic Users](#)
- [Roles](#)
- [Secrets](#)
- [Slot Types](#)
- [Team](#)
- [Trails](#)
- [Users](#)
- [Versions](#)
- [Workspaces](#)
- [Common Types](#)

Analytics Tags

Tag conversation events with sentiment classifications for reporting and analysis.

Contents

- [CreateAnalyticsTag](#)
- [ListAnalyticsTags](#)
- [UpdateAnalyticsTag](#)
- [DeleteAnalyticsTag](#)
- [Request Parameters](#)

CreateAnalyticsTag

Creates a new analytics tag.

Input

Parameter	Type	Required
name	string	Yes
type	enum	Yes
description	string	Yes
metadata	object	No

Sample Request

```
await client.send(new CreateAnalyticsTagCommand({
  name: 'AnalyticsTagName',
  type: 'positive',
  description: 'Analytics Tag Description',
  metadata: { path: '/support', tags: ['support'] },
}));
```

Output

Returns the analytics tags entry that was created.

```
{
  "name": "AnalyticsTagName",
  "type": "positive",
  "description": "Analytics Tag Description",
```

```
"isSystemTag": false,
"metadata": { "path": "/support", "tags": ["support"] },
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

ListAnalyticsTags

Lists all analytics tags in the workspace.

Input

No parameters.

Sample Request

```
await client.send(new ListAnalyticsTagsCommand({}));
```

Output

Returns the full analytics tags collection:

```
{
  "items": [
    {
      "name": "resolved_issue",
      "type": "positive",
      "description": "Customer issue was resolved",
      "isSystemTag": false,
      "metadata": { "path": "/support", "tags": ["support"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "lastUpdatedBy": "ci-deploy-bot"
    }
  ]
}
```

```
]
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

UpdateAnalyticsTag

Updates an existing analytics tag by name.

Input

Parameter	Type	Required
name	string	Yes
type	enum	No
description	string	No
metadata	object	No

Sample Request

```
await client.send(new UpdateAnalyticsTagCommand({
    name: 'AnalyticsTagName',
    type: 'negative',
    description: 'Updated description',
    metadata: { path: '/support', tags: ['support'] },
}));
```

Output

Returns the analytics tags entry that was updated.

```
{
  "name": "AnalyticsTagName",
  "type": "negative",
```

```
"description": "Updated description",
"isSystemTag": false,
"metadata": { "path": "/support", "tags": ["support"] },
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteAnalyticsTag

Deletes an analytics tag by name.

Input

Parameter	Type	Required
name	string	Yes

Sample Request

```
await client.send(new DeleteAnalyticsTagCommand({
    name: 'AnalyticsTagName',
}));
```

Output

No response body.

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

Request Parameters

name

Type: String

Description: The tag identifier. Alphanumeric characters and underscores only. Max 36 characters.

type

Type: String

Description: The sentiment classification of the tag. One of: `positive` , `negative` , `neutral` .

description

Type: String

Description: A short human-readable description of the tag. Max 64 characters.

isSystemTag

Type: Boolean

Description: Whether this tag is system-managed (read-only). System tags cannot be modified or deleted.

metadata

Type: Object

Description: Organizational metadata for categorizing the resource. See [Common Types](#).

createdAt

Type: String

Description: When the tag was created (ISO 8601).

updatedAt

Type: String

Description: When the tag was last modified (ISO 8601).

lastUpdatedBy

Type: String

Description: The identity of who last modified the tag.

API Tokens

Manage ACXD SDK API tokens for programmatic users. Tokens are the credentials used to authenticate with the ACXD SDK.

Contents

- [CreateApiToken](#)
- [ListApiTokens](#)
- [DeleteApiToken](#)
- [Request Parameters](#)

CreateApiToken

Creates a new API token for a programmatic user. The full token is returned only once.

Input

Parameter	Type	Required
userId	string	Yes
description	string	No

Sample Request

```
await client.send(new CreateApiTokenCommand({
  userId: 'programmatic-user-uuid',
  description: 'CI/CD pipeline',
}));
```

Output

```
{
  "token": "acxd_live_mPj4Y4hfXKg5NIuhX24d.K9mN2pQ4rS6tU8vW0xY1zA3bC5dE7fG9",
```

```
"keyPrefix": "mPj4Y4hfXKg5NIuhX24d",
"description": "CI/CD pipeline",
"createdAt": "2026-08-01T12:00:00.000Z"
}
```

Important

The token field is only returned on creation. Store it securely, it cannot be retrieved again.

Errors

- ValidationException (400)
- ConflictException (409)
- InternalServerErrorException (500)

ListApiTokens

Lists all API tokens for a programmatic user. Returns metadata only, secrets are never exposed.

Input

Parameter	Type	Required
userId	string	Yes

Sample Request

```
await client.send(new ListApiTokensCommand({
    userId: 'programmatic-user-uuid',
}));
```

Output

```
{
  "items": [
    {
```

```
    "keyPrefix": "mPj4Y4hfXKg5NIuhX24d",
    "description": "CI/CD pipeline",
    "createdAt": "2026-08-01T12:00:00.000Z"
  }
]
```

Errors

- `ValidationException` (400)
- `InternalServerError` (500)

DeleteApiToken

Permanently revokes an API token.

Input

Parameter	Type	Required
<code>userId</code>	string	Yes
<code>keyPrefix</code>	string	Yes

Sample Request

```
await client.send(new DeleteApiTokenCommand({
  userId: 'programmatic-user-uuid',
  keyPrefix: 'mPj4Y4hfXKg5NIuhX24d',
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)

- `InternalServerError` (500)

Request Parameters

`userId`

Type: String

Description: The programmatic user ID that owns the token.

`description`

Type: String

Description: An optional human-readable label for the token (e.g., "CI/CD pipeline", "staging environment").

`token`

Type: String

Description: The full API token value. Format: `acxd_live_<prefix>.<secret>` . Only returned once on creation, store it securely.

`keyPrefix`

Type: String

Description: The 20-character public prefix of the token. Used to identify tokens in list and delete operations. This is not a secret.

`createdAt`

Type: String

Description: When the token was created (ISO 8601).

Applications

Manage conversational AI applications, the top-level container for flows, builds, and deployments. Each application bundles your flows, knowledge bases, language settings, guardrails, and integrations into a single deployable package within your workspace.

Contents

- [ListApplications](#)
- [CreateApplication](#)
- [GetApplication](#)
- [UpdateApplication](#)
- [DeleteApplication](#)
- [Request Parameters](#)
- [Application Settings](#)
- [Deployment Settings](#)

ListApplications

Lists all applications in the workspace.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListApplicationsCommand({}));
```

Output

```
{
  "items": [
    {
      "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
      "name": "MyFirstApp",
      "description": "My first application",
      "metadata": { "path": "/production", "tags": ["support"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T14:30:00.000Z"
    }
  ]
}
```

```
    },
    {
      "applicationId": "a1b2c3d4-5678-90ab-cdef-1234567890ab",
      "name": "My Support Bot",
      "description": "Handles customer support inquiries",
      "metadata": { "path": "/production", "tags": ["support"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T14:30:00.000Z"
    }
  ],
  "nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateApplication

Creates a new application. An application can start empty and you can attach flows, configure settings, and build it to make it runnable.

Input

Parameter	Type	Required
name	string	Yes
flows	array	No
settings	object	Yes
description	string	No
metadata	object	No
deploymentSettings	object	No

Sample Request

```
await client.send(new CreateApplicationCommand({
    name: "MyFirstApp",
    description: "My first application",
    settings: {
        conversationTTL: 5,
        thresholds: { incomprehensionCount: 2 }
    }
}));
```

Output

```
{
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "name": "MyFirstApp",
  "flows": [],
  "settings": {
    "languageCode": "en-US",
    "languageCodes": [
      "en-US"
    ],
    "languageSettings": [
      {
        "languageCode": "en-US",
        "useNativeLanguage": true
      }
    ],
    "defaultFlows": {},
    "thresholds": {
      "incomprehensionCount": 2
    },
    "conversationTTL": 5,
    "repeatOnIncomprehension": false,
    "clusters": {
      "enabled": false,
      "frequency": {
        "count": 1,
        "resolution": "MONTH"
      },
      "phraseThreshold": {
        "count": 100
      },
      "retention": {
        "count": 30,
```

```
        "resolution": "DAY"
      }
    },
    "deploymentSettings": {
      "oneClickDeployEnabled": true,
      "environment": "production",
      "contextAttributes": []
    },
    "description": "My first application",
    "createdAt": "2026-08-07T22:36:43.411Z",
    "updatedAt": "2026-08-07T22:36:43.411Z",
    "metadata": {},
    "updatedBy": "ci-deploy-bot"
  }
}
```

Errors

- ValidationException (400)
- ConflictException (409)
- InternalServerErrorException (500)

GetApplication

Gets full application details including flows, settings, guardrails, and deployment configuration.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes

Sample Request

```
await client.send(new GetApplicationCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
}));
```

Output

```
{
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "name": "MyFirstApp",
  "flows": [],
  "settings": {
    "languageCode": "en-US",
    "languageCodes": [
      "en-US"
    ],
    "languageSettings": [
      {
        "languageCode": "en-US",
        "useNativeLanguage": true
      }
    ],
    "defaultFlows": {},
    "thresholds": {
      "incomprehensionCount": 2
    },
    "conversationTTL": 5,
    "repeatOnIncomprehension": false,
    "clusters": {
      "enabled": false,
      "frequency": {
        "count": 1,
        "resolution": "MONTH"
      },
      "phraseThreshold": {
        "count": 100
      },
      "retention": {
        "count": 30,
        "resolution": "DAY"
      }
    }
  },
  "deploymentSettings": {
    "oneClickDeployEnabled": true,
    "environment": "production",
    "contextAttributes": []
  },
  "description": "My first application",
}
```

```
"createdAt": "2026-08-07T22:36:43.411Z",
"updatedAt": "2026-08-07T22:36:43.411Z",
"metadata": {},
"updatedBy": "ci-deploy-bot"
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

UpdateApplication

Updates an application. Only include fields you want to change.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
name	string	No
flows	array	No
settings	object	No
description	string	No
metadata	object	No
deploymentSettings	object	No

Sample Request

```
await client.send(new UpdateApplicationCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
```

```
description: "Updated via SDK",
}});
```

Output

Returns the full updated application.

```
{
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "name": "MyFirstApp",
  "flows": [],
  "settings": {
    "languageCode": "en-US",
    "languageCodes": [
      "en-US"
    ],
    "languageSettings": [
      {
        "languageCode": "en-US",
        "useNativeLanguage": true
      }
    ],
    "defaultFlows": {},
    "thresholds": {
      "incomprehensionCount": 2
    },
    "conversationTTL": 5,
    "repeatOnIncomprehension": false,
    "clusters": {
      "enabled": false,
      "frequency": {
        "count": 1,
        "resolution": "MONTH"
      },
      "phraseThreshold": {
        "count": 100
      },
      "retention": {
        "count": 30,
        "resolution": "DAY"
      }
    }
  },
  "deploymentSettings": {
```

```
{
  "oneClickDeployEnabled": true,
  "environment": "production",
  "contextAttributes": []
},
"createdAt": "2026-08-07T22:36:43.411Z",
"updatedAt": "2026-08-07T23:01:19.732Z",
"description": "Updated via SDK",
"metadata": {},
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteApplication

Deletes an application. This removes the application and all its builds and deployments.

Input

Parameter	Type	Required
<code>applicationIdentifier</code>	string	Yes

Sample Request

```
await client.send(new DeleteApplicationCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

Request Parameters

applicationIdentifier

Type: String

The unique identifier for the application (assigned on creation) that is used in Get, Update, and Delete operations.

name

Type: String

Application name. 1–100 characters.

description

Type: String

Application description. Max 200 characters.

flows

Type: Array

List of flow references attached to this application. Each entry is an object with a `flowId` field.

```
[  
  {"flowId": "MainFlow"},  
  {"flowId": "MainFlowTwo"}  
]
```

settings

Type: Object

Application settings controlling language, NLP, guardrails, and runtime behavior. See Application Settings.

deploymentSettings

Type: Object

One-click deploy configuration. See Deployment Settings.

metadata

Type: Object

Organizational metadata. See Common Types.

createdAt

Type: String

When the application was created (ISO 8601).

updatedAt

Type: String

When the application was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the application.

Application Settings

Field	Type	Required
languageCode	string	No
languageCodes	array	No
languageSettings	array	No
defaultFlows	object	No
guardrails	array	No

Field	Type	Required
lifecycleHooks	object	No
thresholds	object	No
conversationTTL	integer	No
autoCorrection	boolean	No
childDirected	boolean	No
repeatOnIncomprehe nsion	boolean	No
clusters	object	No

languageCode

Type: String

Primary language for the application. See Common Types.

languageCodes

Type: Array

Supported languages for the application. See Common Types.

languageSettings

Type: Array

Per-language configuration. Each entry contains:

Field	Type	Required
languageCode	string	Yes
useNativeLanguage	boolean	No

Field	Type	Required
useLex3pAsr	boolean	No
projectId	string	No
voice	string	No

languageSettings.languageCode

Type: String

The language this setting applies to. See Common Types.

languageSettings.useNativeLanguage

Type: Boolean

Whether to use native language processing for this language.

languageSettings.useLex3pAsr

Type: Boolean

Whether to use Lex third-party ASR for this language.

languageSettings.projectId

Type: String

External project ID for this language configuration.

languageSettings.voice

Type: String

Voice identifier for text-to-speech output in this language.

defaultFlows

Type: Object

Default flow assignments for system events. Each field is an object with `flowId` and optional `quickReplies`.

`defaultFlows.welcome`

Type: Object

Flow triggered on conversation start. { `"flowId": "...", "quickReplies": ["Yes", "No"] }`

`defaultFlows.fallback`

Type: Object

Flow triggered when no intent matches.

`defaultFlows.unknown`

Type: Object

Flow triggered on unknown input. Supports optional `knowledgeBaseId`: { `"flowId": "...", "knowledgeBaseId": "...", "quickReplies": [...]` }

`defaultFlows.escalation`

Type: Object

Flow triggered when escalation is requested.

`defaultFlows.frustration`

Type: Object

Flow triggered when frustration is detected.

`defaultFlows.help`

Type: Object

Flow triggered when user asks for help.

`defaultFlows.repeat`

Type: Object

Flow triggered when user asks to repeat.

defaultFlows.resume

Type: Object

Flow triggered when a conversation resumes.

guardrails

Type: Array

Guardrail references attached to the application: [{ "guardrailId": "..."}].

lifecycleHooks

Type: Object

Flow IDs invoked at specific lifecycle events.

lifecycleHooks.conversationStart

Type: String

Flow ID invoked when a conversation begins.

lifecycleHooks.conversationEnd

Type: String

Flow ID invoked when a conversation ends.

lifecycleHooks.escalation

Type: String

Flow ID invoked on escalation.

lifecycleHooks.stateModification

Type: String

Flow ID invoked when state is modified externally.

lifecycleHooks.messageReceived

Type: String

Flow ID invoked on each incoming message.

thresholds

Type: Object

Threshold configuration for conversation behavior.

thresholds.incomprehensionCount

Type: Integer

Number of consecutive incomprehensions before triggering fallback. Default: 2.

conversationTTL

Type: Integer

Session timeout in minutes (1–60). Default: 5.

autoCorrection

Type: Boolean

Enable auto-correction of user input. Default: false.

childDirected

Type: Boolean

COPPA compliance flag. Default: false.

repeatOnIncomprehension

Type: Boolean

Repeat last message on incomprehension. Default: false.

clusters

Type: Object

Configuration for clustering similar user messages for analytics.

clusters.enabled

Type: Boolean

Whether clustering is active. Default: false.

clusters.frequency

Type: Object

How often clusters are computed. { "count": 1, "resolution": "HOUR|DAY|WEEK|MONTH" }

clusters.phraseThreshold

Type: Object

Minimum phrase count to form a cluster. { "count": 100 } (minimum 30).

clusters.retention

Type: Object

How long to retain cluster data. { "count": 30, "resolution": "DAY" } (1–90 days).

Deployment Settings

oneClickDeployEnabled

Type: Boolean

Enable automatic deployment on build success.

environment

Type: String

Target environment for automatic deployments. One of: development, qa, staging, production.

contextAttributes

Type: Array

Context attribute values injected into automatic deployments: [{ "key": "attribute_name", "value": ... }]. Max 50 entries.

Application Builds

A build creates an immutable package of your application's flow logic, language settings, and configuration. A build compiles an application's flows and configuration into a deployable artifact.

Contents

- [Quick Start](#)
- [ListApplicationBuilds](#)
- [CreateApplicationBuild](#)
- [GetApplicationBuild](#)
- [GetApplicationBuildDiff](#)
- [Request Parameters](#)

Quick Start

This example creates an application, attaches a flow, builds it, and checks build status.

```
// 1. Create an Application
const app = await client.send(new CreateApplicationCommand({
  name: "CustomerSupport",
  settings: { conversationTTL: 5, thresholds: { incomprehensionCount: 2 }
}
}));
// 2. Create a Flow
const flow = await client.send(new CreateFlowCommand({
  flowId: "GreetingFlow",
  description: "GreetingFlow",
  nodes: {},
}));
// 3. Attach flow to app
const updatedApp = await client.send(new UpdateApplicationCommand({
  applicationIdentifier: app.applicationId,
  flows: [{ flowId: "GreetingFlow" }]
```

```
    }));  
    // 4. Create a build  
    const build = await client.send(new CreateApplicationBuildCommand({  
      applicationIdentifier: app.applicationId,  
    }));  
    // 5. Get Build Status  
    const status = await client.send(new GetApplicationBuildCommand({  
      applicationIdentifier: app.applicationId,  
      buildIdentifier: build.buildId,  
    }));  
    console.log(status.status); // "PENDING" | "BUILT" | "FAILED"
```

ListApplicationBuilds

Lists builds for an application.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListApplicationBuildsCommand({  
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",  
}));
```

Output

```
{  
  "items": [  
    {  
      "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",  
      "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",  
    },  
  ],  
}
```

```
    "status": "BUILT",
    "version": "1.0",
    "description": "Initial build",
    "createdAt": "2026-08-01T15:00:00.000Z",
    "updatedAt": "2026-08-01T15:02:00.000Z",
    "updatedBy": "ci-deploy-bot",
    "languageSettings": [
      {
        "languageCode": "en-US",
        "useNativeLanguage": true,
        "region": "global"
      }
    ]
  }
]
```

Errors

- ResourceNotFoundException (404)
- ValidationException (400)
- InternalServerErrorException (500)

CreateApplicationBuild

Starts a new build. A validation check runs automatically, review the results to catch errors before deploying. Builds are immutable once created.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
version	string	No
description	string	No
languageSettings	array	No

Sample Request

```
await client.send(new CreateApplicationBuildCommand({
  applicationIdentifier: app.applicationId,
  version: "1.0",
  description: "Release build",
  languageSettings: [
    { languageCode: "en-US", useNativeLanguage: true, region: "global" },
  ],
}));
```

Output

```
{
  "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "status": "PENDING",
  "version": "1.0",
  "description": "Release build",
  "createdAt": "2026-08-01T15:00:00.000Z",
  "updatedAt": "2026-08-01T15:00:00.000Z",
  "updatedBy": "ci-deploy-bot",
  "languageSettings": [
    {
      "languageCode": "en-US",
      "useNativeLanguage": true,
      "region": "global"
    }
  ]
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

GetApplicationBuild

Gets build details and status. A build transitions from PENDING → BUILT (success) or FAILED. Failed builds include details on what caused the error.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
buildIdentifier	string	Yes

Sample Request

```
await client.send(new GetApplicationBuildCommand({
  applicationIdentifier: app.applicationId,
  buildIdentifier: build.buildId,
}));
```

Output

```
{
  "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "status": "BUILT",
  "version": "1.0",
  "description": "Release build",
  "createdAt": "2026-08-01T15:00:00.000Z",
  "updatedAt": "2026-08-01T15:00:00.000Z",
  "updatedBy": "ci-deploy-bot",
  "languageSettings": [
    {
      "languageCode": "en-US",
      "useNativeLanguage": true,
      "region": "global"
    }
  ]
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)

- `InternalServerError (500)`

GetApplicationBuildDiff

Gets the diff between two builds showing what changed.

Input

Parameter	Type	Required
<code>applicationIdentifier</code>	string	Yes
<code>buildIdentifier</code>	string	Yes
<code>previousBuildIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetApplicationBuildDiffCommand({
  applicationIdentifier: APP_ID,
  buildIdentifier: SECOND_BUILD_ID,
  previousBuildIdentifier: FIRST_BUILD_ID,
}));
```

Output

```
{
  "application": {
    "properties": {},
    "settings": {},
    "modifiedSlotTypes": {},
    "modifiedDataRequests": {},
    "modifiedActions": {},
    "attachedFlows": {},
    "detachedFlows": {},
    "modifiedFlows": {}
  }
}
```

```
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

Request Parameters

`applicationIdentifier`

Type: String

The application ID that owns the builds.

`buildIdentifier`

Type: String

The unique identifier for a build (assigned on creation) and used in Get and Diff operations.

`previousBuildIdentifier`

Type: String

The build ID to compare against when generating a diff.

`status`

Type: String

The current build status. One of: PENDING, BUILT, FAILED.

`version`

Type: String

A version label for the build. Max 16 characters.

`description`

Type: String

Build description. Max 200 characters.

languageSettings

Type: Array

Per-language build configuration. Each entry: { "languageCode": "en-US", "useNativeLanguage": true, "region": "global" }. See Application languageSettings.

createdAt

Type: String

When the build was created (ISO 8601).

updatedAt

Type: String

When the build was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the build.

Application Deployments

Deploying an application pushes a successful build to the channels where users interact: chat, voice, IVR, website, mobile, or MCP clients. Only one build is live per application at a time. Deploying a new build deactivates the previous one. You can roll back to any previous build at any time.

Contents

- [Quick Start](#)
- [ListApplicationDeployments](#)
- [CreateApplicationDeployment](#)
- [GetApplicationDeployment](#)
- [UpdateApplicationDeployment](#)
- [DeleteApplicationDeployment](#)

- [Request Parameters](#)

Quick Start

```
// 1. Create Application
const app = await client.send(new CreateApplicationCommand({
  name: "DeploymentTestApp",
  settings: { conversationTTL: 5, thresholds: { incomprehensionCount: 2 }
}
}));

// 2. Create Flow
const flow = await client.send(new CreateFlowCommand({
  flowId: "DeployTestFlow",
  description: "Flow for deployment testing",
  nodes: {}
}));

// 3. Attach Flow
const updatedApp = await client.send(new UpdateApplicationCommand({
  applicationIdentifier: app.applicationId,
  flows: [{ flowId: "DeployTestFlow" }]
}));

// 4. Create Build
const build = await client.send(new CreateApplicationBuildCommand({
  applicationIdentifier: app.applicationId,
}));

// 5. Wait for build
await new Promise(r => setTimeout(r, 5000));
const buildStatus = await client.send(new GetApplicationBuildCommand({
  applicationIdentifier: app.applicationId,
  buildIdentifier: build.buildId,
}));

if (buildStatus.status !== "BUILT") {
  console.log("Build failed, cannot deploy. Stopping.");
  return;
}

// 6. Create Deployment
const deployment = await client.send(new CreateApplicationDeploymentCommand({
```

```
    applicationIdentifier: app.applicationId,  
    buildIdentifier: build.buildId,  
    environment: "production",  
    languageCodes: ["en-US"],  
  }));  
  
// 7. Wait and Get Deployment  
await new Promise(r => setTimeout(r, 5000));  
const deployStatus = await client.send(new GetApplicationDeploymentCommand({  
  applicationIdentifier: app.applicationId,  
  deploymentIdentifier: deployment.deploymentId,  
}));  
  
// 8. List Deployments  
const deployments = await client.send(new ListApplicationDeploymentsCommand({  
  applicationIdentifier: app.applicationId,  
}));
```

ListApplicationDeployments

Lists deployments for an application.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListApplicationDeploymentsCommand({  
  applicationIdentifier: app.applicationId,  
}));
```

Output

```
{
  "items": [
    {
      "deploymentId": "d1e2f3a4-5678-90ab-cdef-1234567890ab",
      "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
      "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
      "description": "Production release v1.0",
      "deploymentStatus": "deployed",
      "environment": "production",
      "analyticsTags": [{ "label": "resolved_issue" }],
      "contextAttributes": [{ "key": "region", "value": "us-west-2" }],
      "createdAt": "2026-08-01T16:00:00.000Z",
      "updatedAt": "2026-08-01T16:00:00.000Z",
      "updatedBy": "ci-deploy-bot"
    }
  ],
  "nextToken": null
}
```

Errors

- ResourceNotFoundException (404)
- ValidationException (400)
- InternalServerErrorException (500)

CreateApplicationDeployment

Deploys a build to an environment, making it live for end users on all configured channels.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
buildIdentifier	string	Yes
description	string	No

Parameter	Type	Required
environment	string, enum	No
languageCodes	array	No
analyticsTags	array	No
contextAttributes	array	No

Sample Request

```
await client.send(new CreateApplicationDeploymentCommand({
  applicationIdentifier: app.applicationId,
  buildIdentifier: build.buildId,
  description: "Production release v1.0",
  environment: "production",
  languageCodes: ["en-US"],
  analyticsTags: [{ label: "resolved_issue" }],
  contextAttributes: [{ key: "region", value: "us-west-2" }],
}));
```

Output

```
{
  "deploymentId": "d1e2f3a4-5678-90ab-cdef-1234567890ab",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  "description": "Production release v1.0",
  "deploymentStatus": "pending",
  "environment": "production",
  "analyticsTags": [{ "label": "resolved_issue" }],
  "contextAttributes": [{ "key": "region", "value": "us-west-2" }],
  "createdAt": "2026-08-01T16:00:00.000Z",
  "updatedAt": "2026-08-01T16:00:00.000Z",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException (404)`

- `ValidationException` (400)
- `InternalServerError` (500)

GetApplicationDeployment

Gets deployment details and status.

Input

Parameter	Type	Required
<code>applicationIdentifier</code>	string	Yes
<code>deploymentIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetApplicationDeploymentCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  deploymentIdentifier: "d1e2f3a4-5678-90ab-cdef-1234567890ab",
}));
```

Output

```
{
  "deploymentId": "d1e2f3a4-5678-90ab-cdef-1234567890ab",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  "description": "Production release v1.0",
  "deploymentStatus": "deployed",
  "environment": "production",
  "analyticsTags": [{ "label": "resolved_issue" }],
  "contextAttributes": [{ "key": "region", "value": "us-west-2" }],
  "createdAt": "2026-08-01T16:00:00.000Z",
  "updatedAt": "2026-08-01T16:00:00.000Z",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateApplicationDeployment

Updates a deployment (e.g., deploy a different build, change environment, or roll back to a previous version).

Input

Parameter	Type	Required
<code>applicationIdentifier</code>	string	Yes
<code>deploymentIdentifier</code>	string	Yes
<code>buildIdentifier</code>	string	Yes
<code>description</code>	string	No
<code>environment</code>	string, enum	No
<code>languageCodes</code>	array	No
<code>analyticsTags</code>	array	No
<code>contextAttributes</code>	array	No

Sample Request

```
await client.send(new UpdateApplicationDeploymentCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  buildIdentifier: "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  deploymentIdentifier: "d1e2f3a4-5678-90ab-cdef-1234567890ab",
  description: "Production release v2.0",
```

```
environment: "production",
analyticsTags: [{ label: "resolved_issue" }],
contextAttributes: [{ key: "region", value: "us-west-2" }],
}});
```

Output

```
{
  "deploymentId": "d1e2f3a4-5678-90ab-cdef-1234567890ab",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "buildId": "b1c2d3e4-5678-90ab-cdef-1234567890ab",
  "description": "Production release v2.0",
  "deploymentStatus": "deployed",
  "environment": "production",
  "analyticsTags": [{ "label": "resolved_issue" }],
  "contextAttributes": [{ "key": "region", "value": "us-west-2" }],
  "createdAt": "2026-08-01T16:00:00.000Z",
  "updatedAt": "2026-08-01T16:00:00.000Z",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- ResourceNotFoundException (404)
- ValidationException (400)
- InternalServerErrorException (500)

DeleteApplicationDeployment

Removes a deployment, taking the application offline. The application stays offline until you redeploy.

Input

Parameter	Type	Required
applicationIdentifier	string	Yes
deploymentIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteApplicationDeploymentCommand({
  applicationIdentifier: "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  deploymentIdentifier: "d1e2f3a4-5678-90ab-cdef-1234567890ab"
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Request Parameters

applicationIdentifier

Type: String

The application ID that owns the deployments.

deploymentIdentifier

Type: String

The unique identifier for a deployment (assigned on creation) and is used in Get, Update, and Delete operations.

buildIdentifier

Type: String

The build ID to deploy.

deploymentStatus

Type: String

The current deployment status (e.g., pending, deployed).

environment

Type: String

The target environment. One of: development, qa, staging, production.

description

Type: String

Deployment description. Max 200 characters.

languageCodes

Type: Array

Languages to deploy (e.g., ["en-US", "es-ES"]).

analyticsTags

Type: Array

Analytics tag references to attach to this deployment: [{ "label": "tag_name" }].

contextAttributes

Type: Array

Context attribute values for this deployment: [{ "key": "attribute_name", "value": "... " }]. Max 50 entries.

createdAt

Type: String

When the deployment was created (ISO 8601).

updatedAt

Type: String

When the deployment was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the deployment.

Context Variables

Define typed variables available across conversations within a workspace. Context variables can be referenced in flows and set by external systems.

Contents

- [ListContextVariables](#)
- [CreateContextVariable](#)
- [UpdateContextVariable](#)
- [DeleteContextVariable](#)
- [Request Parameters](#)

ListContextVariables

Lists all context variables in the workspace.

Input

No parameters.

Sample Request

```
await client.send(new ListContextVariablesCommand({}));
```

Output

```
{
  "items": [
    {
      "name": "sampleContextVariable",
      "type": "string",
      "schema": {
        "type": "string",
        "isSensitive": false
      },
      "disallowExternalModification": false,
      "metadata": { "path": "/crm", "tags": ["segmentation"] }
    }
  ]
}
```

```
}  
]  
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateContextVariable

Creates a new context variable.

Input

Parameter	Type	Required
name	string	Yes
schema	object	No
disallowExternalModification	boolean	No
metadata	object	No

Sample Request

```
await client.send(new CreateContextVariableCommand({  
  name: "customer_tier",  
  schema: { type: "string", isSensitive: false },  
  disallowExternalModification: false,  
  metadata: { path: "/crm", tags: ["segmentation"] },  
}));
```

Output

```
{  
  "name": "customer_tier",
```

```
"type": "string",
"schema": {
  "type": "string",
  "isSensitive": false
},
"disallowExternalModification": false,
"metadata": { "path": "/crm", "tags": ["segmentation"] }
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

UpdateContextVariable

Updates an existing context variable.

Sample Request

```
await client.send(new UpdateContextVariableCommand({
  contextVariableIdentifier: "customer_tier",
  schema: { type: "string", isSensitive: true },
  disallowExternalModification: true,
  metadata: { path: "/crm", tags: ["segmentation"] },
}));
```

Input

Parameter	Type	Required
contextVariableIdentifier	string	Yes
schema	object	No
disallowExternalModification	boolean	No

Parameter	Type	Required
metadata	object	No

Output

```
{
  "name": "customer_tier",
  "type": "string",
  "schema": { "type": "string", "isSensitive": true },
  "disallowExternalModification": true,
  "metadata": { "path": "/crm", "tags": ["segmentation"] }
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteContextVariable

Deletes a context variable by name.

Input

Parameter	Type	Required
contextVariableIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteContextVariableCommand({
  contextVariableIdentifier: "customer_tier",
}));
```

Output

No response body.

Errors

- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

Request Parameters

name

Type: String

The variable name. Letters and underscores only, cannot start with `nlx_context`. Max 64 characters.

contextVariableIdentifier

Type: String

The context variable name used in Update and Delete operations.

type

Type: String

The data type of the variable. One of: `text`, `string`, `number`, `boolean`.

schema

Type: Object

A JSON Schema object defining the variable's structure and validation rules. Must be an object with at least one of: `type`, `$ref`, or `anyOf` at the top level. If omitted, a default schema is used.

Examples:

Simple string:

```
{ "type": "string" }
```

Number with constraints:

```
{ "type": "number", "minimum": 0, "maximum": 100 }
```

Object with properties:

```
{
  "type": "object",
  "properties": {
    "tier": { "type": "string" },
    "score": { "type": "number" }
  }
}
```

Union type:

```
{
  "anyOf": [
    { "type": "string" },
    { "type": "number" }
  ]
}
```

disallowExternalModification

Type: Boolean

If true, the variable cannot be modified by external systems during a conversation. Only flows can update it.

metadata

Type: Object

Organizational metadata. See [Common Types](#).

createdAt

Type: String

When the variable was created (ISO 8601).

updatedAt

Type: String

When the variable was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the variable.

Conversations

Access conversation history and transcripts.

Contents

- [ListConversations](#)
- [GetConversation](#)
- [Request Parameters](#)
- [Response Fields](#)

ListConversations

Lists conversation transcripts with filters.

Input

Parameter	Type	Required
startTimestamp	string	Yes
endTimestamp	string	Yes
userId	string	No
applicationId	string	No
conversationIdentifier	string	No
flowId	string	No
flowIds	string	No

Parameter	Type	Required
languageCode	string	No
utterance	string	No
search	string	No
analyticsTags	string	No
excludeTrivials	string	No
userEngagement	string	No
sortBy	string	No
sortOrder	string	No
includeSilence	boolean	No
includeEvaluations	string	No
timezone	string	No
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListConversationsCommand({
  startTimestamp: sevenDaysAgo.toISOString(),
  endTimestamp: now.toISOString(),
  maxResults: 10,
}));
```

Output

```
{
  "items": [
    {
```

```
    "conversationId": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
    "firstTimestamp": "2026-08-01T12:00:00.000Z",
    "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
    "userId": "user-123",
    "firstUtterance": "I need help with my order",
    "flowIds": ["MainFlow", "SupportFlow"],
    "elapsedSeconds": 120,
    "analyticsTags": ["resolved_issue"],
    "avgSentimentScore": 0.8,
    "avgResponseTime": 1.2,
    "evaluationResults": []
  },
  "nextToken": null
}
```

Errors

- ValidationException (400)
- InternalServerErrorException (500)

GetConversation

Gets a single conversation's full transcript.

Input

Parameter	Type	Required
conversationIdentifier	string	Yes
includeSilence	boolean	No
includeEvaluations	string	No

Sample Request

```
await client.send(new GetConversationCommand({
```

```
conversationIdentifier: "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
}});
```

Output

```
{
  "conversationId": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
  "timestamp": "2026-08-01T12:00:00.000Z",
  "userId": "user-123",
  "applicationId": "05c3fcc2-7900-41c4-adee-b59dc69be8ae",
  "duration": 120.5,
  "flowIds": ["MainFlow", "SupportFlow"],
  "analyticsTags": ["resolved_issue"],
  "responseTime": 1.2,
  "messages": [
    {
      "isApplication": false,
      "text": "I need help with my order",
      "timestamp": "2026-08-01T12:00:01.000Z",
      "flowId": "MainFlow",
      "isEscalation": false,
      "isIncomprehension": false
    },
    {
      "isApplication": true,
      "text": "I'd be happy to help! Can you provide your order number?",
      "timestamp": "2026-08-01T12:00:02.000Z",
      "flowId": "SupportFlow",
      "nodeId": "ask-order-number",
      "isEscalation": false,
      "isIncomprehension": false
    }
  ],
  "evaluationResults": [
    {
      "evaluationId": "eval-uuid",
      "evaluationName": "Quality Check",
      "score": 0.95,
      "result": "pass",
      "feedback": null
    }
  ]
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

Request Parameters

`conversationIdentifier`

Type: String

The conversation ID to retrieve.

`startTimestamp`

Type: String

Start of time range filter (ISO 8601). Required for listing conversations.

`endTimestamp`

Type: String

End of time range filter (ISO 8601). Required for listing conversations.

`userId`

Type: String

Filter by end-user ID.

`applicationId`

Type: String

Filter by application.

`flowId`

Type: String

Filter by a single flow ID (alphabetic characters, 3–64 chars).

flowIds

Type: String

Filter by multiple flow IDs (comma-separated, alphabetic characters).

languageCode

Type: String

Filter by language. See Common Types.

utterance

Type: String

Filter by user utterance text (max 2000 characters).

search

Type: String

Full-text search across conversation content (max 2000 characters).

analyticsTags

Type: String

Filter by analytics tags (comma-separated, alphanumeric + underscores).

excludeTrivials

Type: String

Exclude trivial conversations. true or false.

userEngagement

Type: String

Filter by user engagement. true or false.

sortBy

Type: String

Field to sort results by.

sortOrder

Type: String

Sort direction: asc or desc.

includeSilence

Type: Boolean

Whether to include silence events in the transcript.

includeEvaluations

Type: String

Whether to include evaluation results. true or false.

timezone

Type: String

Timezone for the time range filter.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (10–300). See Common Types.

Response Fields

conversationId

Type: String

The unique conversation identifier.

timestamp

Type: String

When the conversation started (ISO 8601).

`firstTimestamp`

Type: String

When the first message was sent (ISO 8601). Used in list responses.

`duration`

Type: Number

Total conversation duration in seconds.

`elapsedSeconds`

Type: Integer

Total elapsed time in seconds. Used in list responses.

`flowIds`

Type: Array

List of flow IDs traversed during the conversation.

`analyticsTags`

Type: Array

Analytics tags triggered during the conversation.

`responseTime`

Type: Number

Average bot response time in seconds.

`avgSentimentScore`

Type: Number

Average sentiment score across the conversation (0–1).

`avgResponseTime`

Type: Number

Average response time in seconds. Used in list responses.

`firstUtterance`

Type: String

The first user message. Used in list responses.

`messages`

Type: Array

Full list of conversation messages. Each message contains:

`messages.isApplication`

Type: Boolean

Whether this message is from the bot (true) or the user (false).

`messages.text`

Type: String

The message content.

`messages.timestamp`

Type: String

When the message was sent (ISO 8601).

`messages.flowId`

Type: String

The flow that was active when this message was sent.

`messages.nodeId`

Type: String

The node that generated this message (bot messages only).

`messages.correlationId`

Type: String

Correlation ID for tracing.

`messages.isEscalation`

Type: Boolean

Whether this message triggered an escalation.

`messages.isIncomprehension`

Type: Boolean

Whether the bot did not understand this input.

`messages.isStructured`

Type: Boolean

Whether this is a structured (non-text) message.

`messages.analyticsTags`

Type: Array

Analytics tags triggered by this message.

`messages.type`

Type: String

Message type identifier.

`evaluationResults`

Type: Array

Evaluation results for this conversation (if `includeEvaluations` was `true`).

`evaluationResults.evaluationId`

Type: String

The evaluation that scored this conversation.

`evaluationResults.evaluationName`

Type: String

Name of the evaluation.

`evaluationResults.score`

Type: Number

The evaluation score (0–1).

`evaluationResults.result`

Type: String

The evaluation result (e.g., pass, fail).

`evaluationResults.feedback`

Type: String

Optional feedback from the evaluation.

Data Requests

Configure webhook integrations for retrieving or sending data during conversations.

Contents

- [ListDataRequests](#)
- [CreateDataRequest](#)
- [GetDataRequest](#)
- [UpdateDataRequest](#)
- [DeleteDataRequest](#)
- [Request Parameters](#)
- [Webhook Config](#)

ListDataRequests

Lists all data requests in the workspace.

Input

Parameter	Type	Required
<code>nextToken</code>	string	No

Parameter	Type	Required
maxResults	integer	No

Sample Request

```
await client.send(new ListDataRequestsCommand({
  maxResults: 20,
}));
```

Output

```
{
  "items": [
    {
      "dataRequestId": "getCustomerInfo",
      "type": "object",
      "webhook": {
        "implementation": "inline-static",
        "code": "{\"name\":\"John Doe\",\"tier\":\"premium\"}",
        "sendContext": true
      },
      "description": "Returns static customer info for testing",
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "responseSchema": {
        "type": "object",
        "properties": {
          "name": {
            "type": "string"
          },
          "tier": {
            "type": "string"
          }
        }
      },
      "sensitive": false,
      "updatedBy": "ci-deploy-bot"
    }
  ]
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateDataRequest

Creates a new data request.

Input

Parameter	Type	Required
<code>dataRequestId</code>	string	Yes
<code>type</code>	enum	Yes
<code>webhook</code>	object	Yes
<code>requestSchema</code>	object	No
<code>responseSchema</code>	object	No
<code>sensitive</code>	boolean	No
<code>description</code>	string	No
<code>metadata</code>	object	No

Sample Request

```
await client.send(new CreateDataRequestCommand({
  dataRequestId: "getCustomerInfo",
  type: "object",
  webhook: {
    implementation: "inline-static",
    code: JSON.stringify({ name: "John Doe", tier: "premium" }),
  },
  responseSchema: {
    type: "object",
```

```
    properties: {
      name: { type: "string" },
      tier: { type: "string" },
    },
  },
  description: "Returns static customer info for testing",
  sensitive: false,
}));
```

Output

```
{
  "dataRequestId": "getCustomerInfo",
  "type": "object",
  "webhook": {
    "implementation": "inline-static",
    "code": "{\\"name\\":\\"John Doe\\",\\"tier\\":\\"premium\\"}",
    "sendContext": true
  },
  "createdAt": "2026-08-10T01:55:34.146Z",
  "updatedAt": "2026-08-10T01:55:37.733Z",
  "responseSchema": {
    "type": "object",
    "properties": {
      "name": {
        "type": "string"
      },
      "tier": {
        "type": "string"
      }
    }
  },
  "sensitive": false,
  "description": "Returns static customer info for testing",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetDataRequest

Gets a single data request by ID.

Input

Parameter	Type	Required
dataRequestId ier	string	Yes

Sample Request

```
await client.send(new GetDataRequestCommand({
  dataRequestId: "getCustomerInfo",
}));
```

Output

```
{
  "dataRequestId": "getCustomerInfo",
  "type": "object",
  "webhook": {
    "implementation": "inline-static",
    "code": "{\\"name\\":\\"John Doe\\",\\"tier\\":\\"premium\\"}",
    "sendContext": true
  },
  "createdAt": "2026-08-10T01:55:34.146Z",
  "updatedAt": "2026-08-10T01:55:34.146Z",
  "responseSchema": {
    "type": "object",
    "properties": {
      "name": {
        "type": "string"
      },
      "tier": {
        "type": "string"
      }
    }
  },
  "sensitive": false,
```

```
"description": "Returns static customer info for testing",
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateDataRequest

Updates an existing data request. Only include fields you want to change.

Input

Parameter	Type	Required
<code>dataRequestId</code>	string	Yes
<code>type</code>	string	No
<code>webhook</code>	object	No
<code>requestSchema</code>	object	No
<code>responseSchema</code>	object	No
<code>sensitive</code>	boolean	No
<code>description</code>	string	No
<code>metadata</code>	object	No

Sample Request

```
await client.send(new UpdateDataRequestCommand({
  dataRequestId: "getCustomerInfo",
```

```

type: "object",
description: "Updated static customer data",
webhook: {
  implementation: "inline-static",
  code: JSON.stringify({ name: "Jane Smith", tier: "enterprise" }),
},
responseSchema: {
  type: "object",
  properties: {
    name: { type: "string" },
    tier: { type: "string" },
  },
},
sensitive: false,
}));

```

Output

```

{
  "dataRequestId": "getCustomerInfo",
  "type": "object",
  "webhook": {
    "implementation": "inline-static",
    "code": "{\\"name\\":\\"Jane Smith\\",\\"tier\\":\\"enterprise\\"}",
    "sendContext": true
  },
  "createdAt": "2026-08-10T01:55:34.146Z",
  "updatedAt": "2026-08-10T01:55:37.733Z",
  "responseSchema": {
    "type": "object",
    "properties": {
      "name": {
        "type": "string"
      },
      "tier": {
        "type": "string"
      }
    }
  },
  "sensitive": false,
  "description": "Updated static customer data",
  "updatedBy": "ci-deploy-bot"
}

```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteDataRequest

Deletes a data request.

Input

Parameter	Type	Required
<code>dataRequestId</code>	string	Yes

Sample Request

```
await client.send(new DeleteDataRequestCommand({
  dataRequestId: "getCustomerInfo",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Request Parameters

`dataRequestId`

Type: String

The data request identifier. Alphanumeric characters only, 3–100 characters.

dataRequestId

Type: String

The data request ID used in Get, Update, and Delete operations.

type

Type: String

The return type of the data request. One of: text, number, boolean, list<text>, object, list<object>.

webhook

Type: Object

Webhook configuration. See Webhook Config.

requestSchema

Type: Object

A JSON Schema defining the expected request payload shape. Same format as Context Variables schema, must have type, \$ref, or anyOf at the top level.

responseSchema

Type: Object

A JSON Schema defining the expected response payload shape.

sensitive

Type: Boolean

Whether this data request handles sensitive data.

description

Type: String

Description. Max 200 characters.

metadata

Type: Object

Organizational metadata. See Common Types.

createdAt

Type: String

When the data request was created (ISO 8601).

updatedAt

Type: String

When the data request was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the data request.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–500). See Common Types.

Webhook Config

Field	Type	Required
implementation	enum	Yes
method	enum	No

Field	Type	Required
url	string	No
headers	array	No
code	string	No
environments	object	No
provider	object	No
mcp	object	No
sendContext	boolean	No

implementation

Type: String

The webhook implementation type. One of: `inline-static`, `external`, `mcp`.

method

Type: String

HTTP method. One of: `GET`, `POST`, `PUT`, `PATCH`, `DELETE`.

url

Type: String

Webhook URL. Max 2048 characters.

headers

Type: Array

Request headers. Each entry:

headers.key

Type: String

Header name. Max 128 characters.

headers.value

Type: String

Header value. Max 4096 characters. Supports secret references: `{{secrets.my_secret}}`.

headers.sensitive

Type: Boolean

Whether this header contains sensitive data (will be masked in logs).

headers.dynamic

Type: Boolean

Whether this header value is evaluated at runtime.

headers.required

Type: Boolean

Whether this header is required.

code

Type: String

Inline code for `inline-static` implementations. Max 200,000 characters.

environments

Type: Object

Environment-specific configuration (free-form object).

provider

Type: Object

Provider reference for managed integrations.

provider.providerId

Type: String

The provider identifier.

provider.actionId

Type: String

The action identifier within the provider.

mcp

Type: Object

Configuration for mcp implementation. Contains method, url, headers, environments, and tools.

mcp.tools

Type: Array

MCP tool definitions. Each tool has name (string, required), enabled (boolean), requestSchema (object), responseSchema (object). Max 100 tools.

sendContext

Type: Boolean

Whether to send conversation context with the webhook request.

Downloads

Configure data exports one-time downloads and scheduled recurring exports.

Contents

- [GetDownload](#)

GetDownload

Polls the status of an async download job. Returns a pre-signed URL when the download has completed.

Input

Parameter	Type	Required
downloadIdentifier	string	Yes

Output

```
{
  "url": "https://s3.amazonaws.com/bucket/path/file.csv?X-Amz-Signature=..."
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

Request Parameters

downloadIdentifier

Type: String

The download job ID to poll.

Flows

A flow represents the structured path a conversation follows to fulfill a user's intent, whether that's answering FAQs, completing a task, or guiding a user toward an action. Each flow combines logic, prompts, and responses into a graph of connected nodes. Once attached to an application and deployed, the application can execute the flow and automate the corresponding task. A single flow can be shared across multiple applications in your workspace.

Contents

- [ListFlows](#)
- [CreateFlow](#)

- [GetFlow](#)
- [UpdateFlow](#)
- [DeleteFlow](#)
- [Request Parameters](#)
- [Attached Slot](#)
- [Flow Node](#)

ListFlows

Lists all flows in the workspace. Returns summary information without node details.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListFlowsCommand({}));
```

Output

```
{
  "items": [
    {
      "flowId": "MainFlow",
      "description": "Handles customer support inquiries",
      "mainLanguageCode": "en-US",
      "languageCodes": ["en-US", "es-ES"],
      "slotTypes": [],
      "contextVariables": [],
      "metadata": { "path": "/support", "tags": ["production"] },
      "saveId": "save-abc123",
      "createdAt": "2026-08-01T12:00:00.000Z",
    }
  ]
}
```

```
      "updatedAt": "2026-08-01T14:00:00.000Z",
      "updatedBy": "ci-deploy-bot"
    }
  ],
  "nextToken": null
}
```

Errors

- ValidationException (400)
- InternalServerErrorException (500)
- ThrottlingException (429)

CreateFlow

Creates a new flow with nodes (conversation logic), and optional slot types and context variables.

Input

Parameter	Type	Required
flowId	string	Yes
description	string	No
nodes	object	Yes
aiDescription	string	No
untrained	boolean	No
mainLanguageCode	string	No
languageCode	enum	No
languageCodes	array	No
slotTypes	array	No
contextVariables	array	No

Parameter	Type	Required
mcp	object	No
metadata	object	No

Sample Request

```
const created = await client.send(new CreateFlowCommand({
  flowId: "MainFlow",
  description: "Handles customer support inquiries",
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  slotTypes: [],
  contextVariables: [],
  nodes: {
    "a0000000-0000-4000-8000-000000000001": {
      nodeId: "a0000000-0000-4000-8000-000000000001",
      type: "start",
      childNodes: [{ nodeId: "a0000000-0000-4000-8000-000000000002" }],
    },
    "a0000000-0000-4000-8000-000000000002": {
      nodeId: "a0000000-0000-4000-8000-000000000002",
      type: "basic",
      messages: [{ body: "Hello! How can I help you today?", type: "text" }],
      childNodes: [{ nodeId: "a0000000-0000-4000-8000-000000000003" }],
    },
    "a0000000-0000-4000-8000-000000000003": {
      nodeId: "a0000000-0000-4000-8000-000000000003",
      type: "end",
    },
  },
  metadata: { path: "/support", tags: ["production"] },
}));
```

Output

```
{
  "flowId": "MainFlow",
  "description": "Handles customer support inquiries",
  "mainLanguageCode": "en-US",
```

```

"languageCodes": ["en-US"],
"slotTypes": [],
"contextVariables": [],
"nodes": {
  "a0000000-0000-4000-8000-000000000001": {
    "nodeId": "a0000000-0000-4000-8000-000000000001",
    "type": "start",
    "childNodes": [
      {
        "nodeId": "a0000000-0000-4000-8000-000000000002"
      }
    ]
  },
  "a0000000-0000-4000-8000-000000000002": {
    "nodeId": "a0000000-0000-4000-8000-000000000002",
    "type": "basic",
    "messages": [
      {
        "type": "text",
        "body": "Hello! How can I help you today?",
        "messageId": "12a13e04-0bc1-4361-9505-8e9c7a2aeb7b"
      }
    ]
  },
  "a0000000-0000-4000-8000-000000000003": {
    "nodeId": "a0000000-0000-4000-8000-000000000003",
    "type": "end",
    "messages": []
  }
},
"metadata": { "path": "/support", "tags": ["production"] },
"saveId": "save-abc123",
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"updatedBy": "ci-deploy-bot"
}

```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

- `ThrottlingException` (429)

GetFlow

Gets the full flow definition including all nodes, attached slots, and context variables.

Input

Parameter	Type	Required
<code>flowIdentifier</code>	string	Yes
<code>languageCode</code>	string	No

Sample Request

```
const fetched = await client.send(new GetFlowCommand({
  flowIdentifier: "MainFlow",
}));
```

Output

```
{
  "flowId": "MainFlow",
  "description": "Handles customer support inquiries",
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "slotTypes": [],
  "contextVariables": [],
  "nodes": {
    "a0000000-0000-4000-8000-000000000001": {
      "nodeId": "a0000000-0000-4000-8000-000000000001",
      "type": "start",
      "childNodes": [
        {
          "nodeId": "a0000000-0000-4000-8000-000000000002"
        }
      ]
    },
    "a0000000-0000-4000-8000-000000000002": {
```

```
{
  "nodeId": "a0000000-0000-4000-8000-000000000002",
  "type": "basic",
  "messages": [
    {
      "type": "text",
      "body": "Hello! How can I help you today?",
      "messageId": "12a13e04-0bc1-4361-9505-8e9c7a2aeb7b"
    }
  ]
},
"a0000000-0000-4000-8000-000000000003": {
  "nodeId": "a0000000-0000-4000-8000-000000000003",
  "type": "end",
  "messages": []
}
},
"metadata": { "path": "/support", "tags": ["production"] },
"saveId": "save-abc123",
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

UpdateFlow

Updates an existing flow. Only include fields you want to change. Changes do not affect deployed applications until a new build is created.

Input

Parameter	Type	Required
<code>flowIdentifier</code>	string	Yes

Parameter	Type	Required
nodes	object	No
description	string	No
aiDescription	string	No
untrained	boolean	No
mainLanguageCode	string	No
languageCode	enum	No
languageCodes	array	No
slotTypes	array	No
contextVariables	array	No
mcp	object	No
metadata	object	No

Sample Request

```
await client.send(new UpdateFlowCommand({
  flowIdentifier: "MainFlow",
  description: "Updated Primary support flow",
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  slotTypes: [],
  contextVariables: [],
  nodes: {
    "a00000000-0000-4000-8000-000000000001": {
      nodeId: "a00000000-0000-4000-8000-000000000001",
      type: "start",
      childNodes: [{ nodeId: "a00000000-0000-4000-8000-000000000002" }],
    },
    "a00000000-0000-4000-8000-000000000002": {
      nodeId: "a00000000-0000-4000-8000-000000000002",
```

```

        type: "basic",
        messages: [{ body: "Hello! How can I help you today?", type: "text" }],
        childNodes: [{ nodeId: "a0000000-0000-4000-8000-000000000003" }],
    },
    "a0000000-0000-4000-8000-000000000003": {
        nodeId: "a0000000-0000-4000-8000-000000000003",
        type: "end",
    },
},
metadata: { path: "/support", tags: ["production"] },
));

```

Output

```

{
  "flowId": "MainFlow",
  "description": "Updated Primary support flow",
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "slotTypes": [],
  "contextVariables": [],
  "nodes": {
    "a0000000-0000-4000-8000-000000000001": {
      "nodeId": "a0000000-0000-4000-8000-000000000001",
      "type": "start",
      "childNodes": [
        {
          "nodeId": "a0000000-0000-4000-8000-000000000002"
        }
      ]
    },
    "a0000000-0000-4000-8000-000000000002": {
      "nodeId": "a0000000-0000-4000-8000-000000000002",
      "type": "basic",
      "messages": [
        {
          "type": "text",
          "body": "Hello! How can I help you today?",
          "messageId": "12a13e04-0bc1-4361-9505-8e9c7a2aeb7b"
        }
      ]
    },
    "a0000000-0000-4000-8000-000000000003": {

```

```
    "nodeId": "a0000000-0000-4000-8000-000000000003",
    "type": "end",
    "messages": []
  }
},
"metadata": { "path": "/support", "tags": ["production"] },
"saveId": "save-abc123",
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `ConflictException` (409)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

DeleteFlow

Deletes a flow. If the flow is attached to applications, detach it first.

Input

Parameter	Type	Required
<code>flowIdentifier</code>	string	Yes

Sample Request

```
await client.send(new DeleteFlowCommand({
  flowIdentifier: "TestFlowSDKClient",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

Request Parameters

`flowId`

Type: String

The flow identifier. Alphanumeric characters, 3–64 characters.

`flowIdentifier`

Type: String

The flow ID used in Get, Update, and Delete operations.

`description`

Type: String

Flow description. Max 200 characters.

`aiDescription`

Type: String

AI-readable description of what this flow does. Max 1000 characters. Used by generative features to understand flow purpose.

`untrained`

Type: Boolean

Whether to skip NLP training for this flow.

`mainLanguageCode`

Type: String

Primary language. See Common Types.

languageCode

Type: String

Language code. See Common Types.

languageCodes

Type: Array

Supported languages. See Common Types.

nodes

Type: Object

The flow node graph. A map of node IDs to node objects. See Flow Node.

slotTypes

Type: Array

Slot types attached to this flow. See Attached Slot.

contextVariables

Type: Array

Flow-scoped context variables. Each entry: { "name": "varName", "type": "text|number|boolean" }.

mcp

Type: Object

MCP endpoint configuration: { "input": { "name": "...", "schema": {...} }, "output": { "name": "...", "schema": {...} } }.

metadata

Type: Object

Organizational metadata. See Common Types.

`magicLayout`

Type: Boolean

Whether to apply automatic layout during validation.

`saveId`

Type: String

Internal save identifier (read-only).

`createdAt`

Type: String

When the flow was created (ISO 8601).

`updatedAt`

Type: String

When the flow was last modified (ISO 8601).

`updatedBy`

Type: String

The identity of who last modified the flow.

`nextToken`

Type: String

Pagination token. See Common Types.

`maxResults`

Type: Integer

Max items per page (1–500). See Common Types.

Attached Slot

Field	Type	Required
name	string	Yes
type	string	Yes
sensitive	boolean	No
examples	array	No
aiDescription	string	No
regex	string	No

name

Type: String

Slot name. Alphabetic characters only, 3–30 characters.

type

Type: String

The slot type reference.

sensitive

Type: Boolean

Whether this slot captures sensitive data.

examples

Type: Array

Example values for this slot (array of strings).

aiDescription

Type: String

AI-readable description of what this slot captures. Max 1000 characters.

regex

Type: String

Optional regex pattern for validation. Max 300 characters.

Flow Node

Field	Type	Required
nodeId	string	Yes
type	enum	Yes
childNodes	array	No
dataRequests	array	No
messages	array	No
modalities	object	No
canvasMetadata	object	No
metadata	object	No

nodeId

Type: String

Unique node identifier.

type

Type: String

The node type. One of: basic, start, end, user_input, user_choice, choice, data_request, redirect, escalate, split, loop, define, wait, transform, note, knowledge_base, generative_text, generative_task, generative_journey, multimodal, intent_capture, application_handoff.

childNodes

Type: Array

Connected child nodes. Each entry: { "nodeId": "...", "name": "...", "conditions": [...] }.

dataRequests

Type: Array

Data requests triggered by this node.

messages

Type: Array

Messages displayed at this node.

modalities

Type: Object

Modality-specific content (free-form).

canvasMetadata

Type: Object

Visual editor position and display settings: { "x": 100, "y": 200, "width": 300, "height": 150, "color": "#fff", "pageId": "..." }.

metadata

Type: Object

Node-type-specific configuration. The fields available depend on the node type. Contains configuration like `generativeText`, `choice`, `redirect`, `knowledgeBase`, `loop`, `multimodal`, `stateModifications`, `tags`, `name`, `timeout`, etc.

Guardrails

Define safety rules that monitor conversation content and enforce behavior like masking, modifying, rerouting, or flagging when violations are detected.

Contents

- [ListGuardrails](#)
- [CreateGuardrail](#)
- [GetGuardrail](#)
- [UpdateGuardrail](#)
- [DeleteGuardrail](#)
- [TestGuardrail](#)
- [ListGuardrailEvents](#)
- [Request Parameters](#)
- [Guardrail Rule](#)
- [Detection](#)
- [Enforcement](#)
- [Enforcement Behavior](#)
- [Fallback Behavior](#)

ListGuardrails

Lists all guardrails in the workspace.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListGuardrailsCommand({}));
```

Output

```
{
```

```
"items": [
  {
    "guardrailId": "g1a2b3c4-5678-90ab-cdef-1234567890ab",
    "name": "PII Filter",
    "trigger": "output",
    "description": "Masks personally identifiable information in bot responses",
    "active": true,
    "rules": [
      {
        "id": "r1a2b3c4-...",
        "name": "Email detection",
        "detection": { "method": "regex", "pattern": "[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\.[a-zA-Z]{2,}" },
        "enforcement": { "action": "mask", "behavior": {"maskText": "[REDACTED]" } },
        "active": true
      }
    ],
    "metadata": { "path": "/safety", "tags": ["pii"] },
    "createdAt": "2026-08-01T12:00:00.000Z",
    "updatedAt": "2026-08-01T12:00:00.000Z",
    "lastUpdatedBy": "ci-deploy-bot"
  }
]
```

Errors

- ValidationException (400)
- InternalServerErrorException (500)

CreateGuardrail

Creates a new guardrail.

Input

Parameter	Type	Required
name	string	Yes
rules	array	Yes

Parameter	Type	Required
trigger	string	Yes
description	string	No
active	boolean	No
metadata	object	No
fallbackBehavior	object	No

Sample Request

```
await client.send(new CreateGuardrailCommand({
  name: "PII Filter",
  trigger: "output",
  description: "Masks PII in bot responses",
  active: true,
  rules: [
    {
      name: "Email detection",
      detection: {
        method: "regex",
        pattern: "[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\.[a-zA-Z]{2,}",
      },
      enforcement: {
        action: "mask",
        behavior: {maskText: "[REDACTED]"},
      },
      active: true,
    },
  ],
  metadata: { path: "/safety", tags: ["pii"] },
}));
```

Output

```
{
  "guardrailId": "g1a2b3c4-5678-90ab-cdef-1234567890ab",
  "name": "PII Filter",
```

```
    "trigger": "output",
    "description": "Masks PII in bot responses",
    "active": true,
    "rules": [
      {
        "id": "r1a2b3c4-...",
        "name": "Email detection",
        "detection": { "method": "regex", "pattern": "[a-zA-Z0-9._%+~]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}" },
        "enforcement": { "action": "mask", "behavior": {"maskText": "[REDACTED]" } },
        "active": true
      }
    ],
    "metadata": { "path": "/safety", "tags": ["pii"] },
    "createdAt": "2026-08-01T12:00:00.000Z",
    "updatedAt": "2026-08-01T12:00:00.000Z",
    "lastUpdatedBy": "ci-deploy-bot"
  }
```

Errors

- ValidationException (400)
- ConflictException (409)
- InternalServerErrorException (500)

GetGuardrail

Gets a single guardrail by ID.

Sample Request

```
await client.send(new GetGuardrailCommand({
  guardrailIdentifier: "g1a2b3c4-5678-90ab-cdef-1234567890ab"
}));
```

Input

Parameter	Type	Required
guardrailIdentifier	string	Yes

Output

```
{
  "guardrailId": "g1a2b3c4-5678-90ab-cdef-1234567890ab",
  "name": "PII Filter",
  "trigger": "output",
  "description": "Masks personally identifiable information in bot responses",
  "active": true,
  "rules": [
    {
      "id": "r1a2b3c4-...",
      "name": "Email detection",
      "detection": { "method": "regex", "pattern": "[a-zA-Z0-9._%+~]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}" },
      "enforcement": { "action": "mask", "behavior": {"maskText": "[REDACTED]" } },
      "active": true
    }
  ],
  "metadata": { "path": "/safety", "tags": ["pii"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

UpdateGuardrail

Updates an existing guardrail. Only include fields you want to change.

Input

Parameter	Type	Required
guardrailIdentifier	string	Yes

Parameter	Type	Required
name	string	No
rules	array	No
trigger	string	No
description	string	No
active	boolean	No
metadata	object	No
fallbackBehavior	object	No

Sample Request

```
await client.send(new UpdateGuardrailCommand({
  guardrailIdentifier: "g1a2b3c4-5678-90ab-cdef-1234567890ab",
  name: "PII Filter",
  trigger: "output",
  description: "Updated - masks PII patterns",
  active: false,
  rules: [
    {
      name: "Email detection",
      detection: {
        method: "regex",
        pattern: "[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}",
      },
      enforcement: {
        action: "mask",
        behavior: {maskText: "[REDACTED]"},
      },
      active: true,
    },
  ],
  metadata: { path: "/safety", tags: ["pii"] },
}));
```

Output

```
{
  "guardrailId": "g1a2b3c4-5678-90ab-cdef-1234567890ab",
  "name": "PII Filter",
  "trigger": "output",
  "description": "Updated - masks PII patterns",
  "active": false,
  "rules": [
    {
      "id": "r1a2b3c4-...",
      "name": "Email detection",
      "detection": { "method": "regex", "pattern": "[a-zA-Z0-9._%+~]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}" },
      "enforcement": { "action": "mask", "behavior": {"maskText": "[REDACTED]" } },
      "active": true
    }
  ],
  "metadata": { "path": "/safety", "tags": ["pii"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- ResourceNotFoundException (404)
- ValidationException (400)
- InternalServerErrorException (500)

DeleteGuardrail

Deletes a guardrail.

Input

Parameter	Type	Required
guardrailIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteGuardrailCommand({
  guardrailIdentifier: "2be7442d-3cb4-4f45-940a-20335a339f70",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

TestGuardrail

Tests a guardrail against sample input without affecting live conversations.

Input

Parameter	Type	Required
<code>guardrailIdentifier</code>	string	Yes
<code>input</code>	string	Yes
<code>trigger</code>	string	No

Sample Request

```
await client.send(new TestGuardrailCommand({
  guardrailIdentifier: created.guardrailId,
  input: "My email is john@example.com",
}));
```

Output

```
{
  "input": "My email is john@example.com",
  "processedInput": "My email is [REDACTED]",
  "output": "My email is [REDACTED]",
  "blocked": false,
  "terminalRuleId": null,
  "violations": [
    {
      "ruleId": "r1a2b3c4-...",
      "ruleName": "Email detection",
      "action": "mask",
      "behavior": {"maskText": "[REDACTED]" },
      "metadata": { "pattern": "[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\.[a-zA-Z]{2,}" },
      "latencyMs": 12
    }
  ]
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

ListGuardrailEvents

Lists historical guardrail trigger events.

Input

Parameter	Type	Required
<code>startTimestamp</code>	string	Yes
<code>endTimestamp</code>	string	Yes
<code>guardrailIdentifier</code>	string	No

Parameter	Type	Required
behaviorType	enum	No
userId	string	No
applicationId	string	No
conversationId	string	No
languageCode	enum	No
ruleId	string	No
sortBy	enum	No
sortOrder	enum	No
timezone	string	No

Sample Request

```
const now = new Date();
const sevenDaysAgo = new Date(now.getTime() - 7 * 24 * 60 * 60 * 1000);
const events = await client.send(new ListGuardrailEventsCommand({
  startTimestamp: sevenDaysAgo.toISOString(),
  endTimestamp: now.toISOString(),
}));
```

Output

```
{
  "items": [
    {
      "timestamp": "2026-08-01T12:05:00.000Z",
      "applicationId": "05c3fcc2-...",
      "conversationId": "conv-uuid",
      "userId": "user-123",
      "guardrailId": "g1a2b3c4-...",
      "guardrailName": "PII Filter",
      "behaviorType": "mask",
```

```
{
  "ruleId": "r1a2b3c4-...",
  "ruleName": "Email detection",
  "originalResponse": "My email is john@example.com",
  "ruleOutput": "My email is [REDACTED]",
  "responseTime": 15.2
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

Request Parameters

Request Parameters

`guardrailIdentifier`

Type: String

The guardrail ID used in Get, Update, Delete, and Test operations.

`guardrailId`

Type: String

The unique guardrail identifier (assigned on creation).

`name`

Type: String

Guardrail name. 1–100 characters.

`trigger`

Type: String

When to evaluate the guardrail. One of: input (check user messages), output (check bot responses).

description

Type: String

Guardrail description. Max 100 characters.

active

Type: Boolean

Whether the guardrail is active.

rules

Type: Array

List of rules (max 50). See Guardrail Rule.

fallbackBehavior

Type: Object

What to do if guardrail processing fails. See Fallback Behavior.

input

Type: String

Sample input text to test against the guardrail (TestGuardrail only).

metadata

Type: Object

Organizational metadata. See Common Types.

startTimestamp

Type: String

Start of time range for events (ISO 8601).

endTimestamp

Type: String

End of time range for events (ISO 8601).

behaviorType

Type: String

Filter events by enforcement action. One of: mask, modify, route, flag.

userId

Type: String

Filter events by end-user ID. Max 255 characters.

applicationId

Type: String

Filter events by application.

conversationId

Type: String

Filter events by conversation.

languageCode

Type: String

Filter events by language. See Common Types.

ruleId

Type: String

Filter events by a specific rule.

sortBy

Type: String

Sort events by field. timestamp.

sortOrder

Type: String

Sort direction: asc or desc.

timezone

Type: String

Timezone for the time range filter.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–500). See Common Types.

createdAt

Type: String

When the guardrail was created (ISO 8601).

updatedAt

Type: String

When the guardrail was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the guardrail.

Guardrail Rule

Field	Type	Required
id	string	No
name	string	Yes

Field	Type	Required
detection	object	Yes
enforcement	object	Yes
description	string	No
active	boolean	No
stateModifications	array	No
tags	array	No

id

Type: String

Rule identifier. Server-generated if omitted on create.

name

Type: String

Rule name. Max 100 characters.

detection

Type: Object

How to detect violations. See Detection.

enforcement

Type: Object

What to do when triggered. See Enforcement.

description

Type: String

Rule description. Max 100 characters.

active

Type: Boolean

Whether this rule is active.

stateModifications

Type: Array

State changes to apply when the rule triggers.

tags

Type: Array

Analytics tags to apply when triggered: [{ "label": "tag_name" }].

Detection

Field	Type	Required
method	enum	Yes
pattern	string	No
keywords	array	No
prompt	string	No
threshold	float	No

method

Type: String

Detection method. One of: regex, keyword, llmJudge.

pattern

Type: String

Regex pattern (for regex method). Max 200 characters.

keywords

Type: Array

Keyword list (for keyword method). Max 200 keywords, each max 50 characters.

prompt

Type: String

LLM evaluation prompt (for llmJudge method). Max 4000 characters.

threshold

Type: Number

Confidence threshold for llmJudge detection (0–1).

Enforcement

Field	Type	Required
action	enum	Yes
behavior	object	No
tags	array	No

action

Type: String

What to do when the rule triggers. One of: mask, modify, route, flag.

behavior

Type: Object

Action-specific configuration. Provide the variant matching the action. See Enforcement Behavior.

tags

Type: Array

Tags to apply when triggered.

Enforcement Behavior

Fields depend on the enforcement action:

For mask or flag:

```
{ "maskChar": "*", "maskText": "[REDACTED]" }
```

maskChar: Single character to use for masking.

maskText: Text to replace matched content with. Max 50 characters.

For modify:

```
{ "message": "I can't help with that.", "prompt": "...", "flowId": "..." }
```

message: Static replacement message. Max 500 characters.

prompt: LLM prompt to generate a replacement. Max 1000 characters.

flowId: Flow to route to after modification.

For route:

```
{ "flowId": "EscalationFlow" }
```

flowId: Flow to route the conversation to (required).

Fallback Behavior

What to do if guardrail processing itself fails (e.g., LLM timeout).

type

Type: String

One of: `continue` (proceed without guardrail), `routeToFlow` (route to a safe flow).

flowId

Type: String

Flow to route to (required when type is routeToFlow).

Knowledge Bases

Manage knowledge sources for AI-powered responses. Knowledge bases can contain articles (structured Q&A) or documents (uploaded files). They must be published before they are available in conversations.

Contents

- [ListKnowledgeBases](#)
- [CreateKnowledgeBase](#)
- [GetKnowledgeBase](#)
- [UpdateKnowledgeBase](#)
- [DeleteKnowledgeBase](#)
- [CloneKnowledgeBase](#)
- [PublishKnowledgeBase](#)
- [GetKnowledgeBasePublication](#)
- [ListKnowledgeBasePublications](#)
- [Request Parameters](#)
- [Response Config](#)

ListKnowledgeBases

Lists all knowledge bases in the workspace.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListKnowledgeBasesCommand({}));
```

Output

```
{
  "items": [
    {
      "knowledgeBaseId": "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
      "name": "Product FAQ",
      "type": "articles",
      "description": "Common product questions",
      "response": { "summarize": true, "minConfidenceScore": 70, "k": 3 },
      "mainLanguageCode": "en-US",
      "languageCodes": ["en-US"],
      "creationStatus": "SUCCEEDED",
      "metadata": { "path": "/support", "tags": ["faq"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "lastUpdatedBy": "ci-deploy-bot"
    }
  ],
  "nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateKnowledgeBase

Creates a new knowledge base.

Input

Parameter	Type	Required
name	string	Yes

Parameter	Type	Required
type	enum	Yes
description	string	No
response	object	No
mainLanguageCode	enum	No
languageCodes	array	No
metadataSchema	object	No
metadata	object	No

Sample Request

```
await client.send(new CreateKnowledgeBaseCommand({
  name: "Product FAQ",
  type: "articles",
  description: "Common product questions",
  response: {
    summarize: true,
    minConfidenceScore: 70,
    k: 3,
  },
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  metadata: { path: "/support", tags: ["faq"] },
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "name": "Product FAQ",
  "type": "articles",
  "description": "Common product questions",
  "response": { "summarize": true, "minConfidenceScore": 70, "k": 3 },
  "mainLanguageCode": "en-US",
```

```
{
  "languageCodes": ["en-US"],
  "creationStatus": "SUCCEEDED",
  "metadata": { "path": "/support", "tags": ["faq"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Same shape as ListKnowledgeBases item.

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetKnowledgeBase

Gets a single knowledge base by ID.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes

Sample Request

```
await client.send(new GetKnowledgeBaseCommand({
  knowledgeBaseId: "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "name": "Product FAQ",
  "type": "articles",
}
```

```
"description": "Common product questions",
"response": { "summarize": true, "minConfidenceScore": 70, "k": 3 },
"mainLanguageCode": "en-US",
"languageCodes": ["en-US"],
"creationStatus": "SUCCEEDED",
"metadata": { "path": "/support", "tags": ["faq"] },
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateKnowledgeBase

Updates an existing knowledge base. Only include fields you want to change.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>name</code>	string	No
<code>type</code>	enum	No
<code>description</code>	string	No
<code>response</code>	object	No
<code>mainLanguageCode</code>	enum	No
<code>languageCodes</code>	array	No
<code>metadataSchema</code>	object	No

Parameter	Type	Required
metadata	object	No

Sample Request

```
await client.send(new UpdateKnowledgeBaseCommand({
  knowledgeBaseId: "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  name: "Product FAQ",
  type: "articles",
  description: "Updated - product FAQ and troubleshooting",
  response: {
    summarize: true,
    minConfidenceScore: 80,
    k: 5,
  },
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  metadata: { path: "/support", tags: ["faq"] },
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "name": "Product FAQ",
  "type": "articles",
  "description": "Updated - product FAQ and troubleshooting",
  "response": { "summarize": true, "minConfidenceScore": 80, "k": 5 },
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "creationStatus": "SUCCEEDED",
  "metadata": { "path": "/support", "tags": ["faq"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:40:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)

- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteKnowledgeBase

Deletes a knowledge base.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes

Sample Request

```
await client.send(new DeleteKnowledgeBaseCommand({
  knowledgeBaseId: "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
}));
```

Output

No response body.

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

CloneKnowledgeBase

Creates a copy of an existing knowledge base with a new ID.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes

Parameter	Type	Required
name	string	No
portTranslations	boolean	No

Sample Request

```
await client.send(new CloneKnowledgeBaseCommand({
  knowledgeBaseId: created.knowledgeBaseId,
  name: "Product FAQ Clone",
  portTranslations: false,
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "name": "Product FAQ Clone",
  "type": "articles",
  "description": "Updated - product FAQ and troubleshooting",
  "response": { "summarize": true, "minConfidenceScore": 80, "k": 5 },
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "creationStatus": "SUCCEEDED",
  "metadata": { "path": "/support", "tags": ["faq"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:40:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

PublishKnowledgeBase

Publishes a knowledge base, triggering indexing and making it available for conversations.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
deploymentId	string	No
version	string	No
description	string	No

Sample Request

```
await client.send(new PublishKnowledgeBaseCommand({
  knowledgeBaseId: created.knowledgeBaseId,
  version: "1.0",
  description: "Initial publish",
}));
```

Output

```
{
  "deploymentId": "dep-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "knowledgeBaseId": "kb-a1b2c3d4-...",
  "status": "scheduled",
  "version": "1.0",
  "description": "Initial publish",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)

- `InternalServerError` (500)

GetKnowledgeBasePublication

Gets a specific publication record.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>deploymentId</code>	string	Yes

Sample Request

```
await client.send(new GetKnowledgeBasePublicationCommand({
  knowledgeBaseId: created.knowledgeBaseId,
  deploymentId: publications.items[0].deploymentId,
}));
```

Output

Same shape as `PublishKnowledgeBase` output. Status may be scheduled, published, or failed.

```
{
  "deploymentId": "dep-a1b2c3d4-5678-90ab-cdef-1234567890ab",
  "knowledgeBaseId": "kb-a1b2c3d4-...",
  "status": "published",
  "version": "1.0",
  "description": "Initial publish",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

ListKnowledgeBasePublications

Lists all publications for a knowledge base.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListKnowledgeBasePublicationsCommand({
    knowledgeBaseId: "kb-a1b2c3d4-...",
}));
```

Output

```
{
  "items": [
    {
      "deploymentId": "dep-a1b2c3d4-...",
      "knowledgeBaseId": "kb-a1b2c3d4-...",
      "status": "published",
      "version": "1.0",
      "description": "Initial publish",
      "updatedBy": "ci-deploy-bot"
    }
  ]
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)

- `InternalServerError (500)`

Request Parameters

knowledgeBaseId

Type: String

The knowledge base ID.

name

Type: String

Knowledge base name. Alphanumeric + spaces/dashes/underscores, 1–100 characters.

type

Type: String

The type of knowledge base. One of: `articles` or `documents`.

description

Type: String

Description. Max 200 characters.

response

Type: Object

Response behavior configuration. See `Response Config`.

mainLanguageCode

Type: String

Primary language. See `Common Types`.

languageCodes

Type: Array

Supported languages. See Common Types.

metadataSchema

Type: Object

A JSON Schema defining the structure of article/document metadata.

metadata

Type: Object

Organizational metadata. See Common Types.

creationStatus

Type: String

Creation status of the knowledge base. One of: PENDING, SUCCEEDED.

deploymentId

Type: String

The publication/deployment ID.

version

Type: String

Version label for a publication.

status

Type: String

Publication status. One of: scheduled, published, failed.

portTranslations

Type: Boolean

Whether to copy translations when cloning.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

createdAt

Type: String

When the knowledge base was created (ISO 8601).

updatedAt

Type: String

When the knowledge base was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the knowledge base.

Response Config

Field	Type	Required
summarize	boolean	No
minConfidenceScore	float	No
temperature	float	No
topP	float	No
k	integer	No

summarize

Type: Boolean

Whether to summarize retrieved content before returning to the user.

minConfidenceScore

Type: Number

Minimum confidence threshold (0–100). Results below this score are not returned.

temperature

Type: Number

LLM temperature for response generation.

topP

Type: Number

Top-p sampling parameter.

k

Type: Integer

Number of results to retrieve from the knowledge base.

Knowledge Base Articles

Manage structured Q&A content within a knowledge base. Articles contain a question and one or more response messages.

Contents

- [ListKnowledgeBaseArticles](#)
- [CreateKnowledgeBaseArticle](#)
- [GetKnowledgeBaseArticle](#)
- [UpdateKnowledgeBaseArticle](#)

- [DeleteKnowledgeBaseArticle](#)
- [Request Parameters](#)

ListKnowledgeBaseArticles

Lists all articles in a knowledge base.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListKnowledgeBaseArticlesCommand({
    knowledgeBaseId: "kb-a1b2c3d4-...",
}));
```

Output

```
{
  "items": [
    {
      "knowledgeBaseId": "kb-a1b2c3d4-...",
      "articleId": "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
      "question": {
        "text": "How do I reset my password?",
        "messageId": "5701b504-b3ae-4d65-9834-bb65b5dc7fb0"
      },
      "responses": [{ "type": "text", "body": "Go to Settings > Security > Reset Password.", "messageId": "83c542e2-bb98-4f48-b2b9-292af18796e7" }],
      "tags": ["account"],
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
    }
  ]
}
```

```
    "lastUpdatedBy": "ci-deploy-bot"
  }
],
"nextToken": null
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

CreateKnowledgeBaseArticle

Creates a new article in a knowledge base.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>question</code>	object	Yes
<code>responses</code>	array	Yes
<code>articleMetadata</code>	object	No
<code>payload</code>	string	No
<code>tags</code>	array	No

Sample Request

```
await client.send(new CreateKnowledgeBaseArticleCommand({
  knowledgeBaseId: "kb-a1b2c3d4-...",
  question: { text: "How do I reset my password?" },
  responses: [
    { type: "text", body: "Go to Settings > Security > Reset Password." },
```

```
    { type: "text", body: "If you still have trouble, contact support." },
  ],
  tags: ["account", "security"],
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-...",
  "articleId": "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
  "question": {
    "text": "How do I reset my password?",
    "messageId": "5701b504-b3ae-4d65-9834-bb65b5dc7fb0"
  },
  "responses": [
    {
      "type": "text",
      "body": "Go to Settings > Security > Reset Password.",
      "messageId": "83c542e2-bb98-4f48-b2b9-292af18796e7"
    },
    {
      "type": "text",
      "body": "If you still have trouble, contact support.",
      "messageId": "806d8c14-d0c0-4eee-a26f-be7f74fb3ad7"
    }
  ],
  "tags": ["account", "security"],
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

GetKnowledgeBaseArticle

Gets a single article by ID.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
articleId	string	Yes

Sample Request

```
await client.send(new GetKnowledgeBaseArticleCommand({
  knowledgeBaseId: "kb-a1b2c3d4-...",
  articleId: "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
}));
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-...",
  "articleId": "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
  "question": {
    "text": "How do I reset my password?",
    "messageId": "5701b504-b3ae-4d65-9834-bb65b5dc7fb0"
  },
  "responses": [
    {
      "type": "text",
      "body": "Go to Settings > Security > Reset Password.",
      "messageId": "83c542e2-bb98-4f48-b2b9-292af18796e7"
    },
    {
      "type": "text",
      "body": "If you still have trouble, contact support.",
      "messageId": "806d8c14-d0c0-4eee-a26f-be7f74fb3ad7"
    }
  ],
  "tags": ["account", "security"],
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

UpdateKnowledgeBaseArticle

Updates an existing article. Only include fields you want to change.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>articleId</code>	string	Yes
<code>question</code>	object	No
<code>responses</code>	array	No
<code>articleMetadata</code>	object	No
<code>payload</code>	string	No
<code>tags</code>	array	No

Sample Request

```
await client.send(new UpdateKnowledgeBaseArticleCommand({
  knowledgeBaseId: "kb-a1b2c3d4-...",
  articleId: "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
  question: { text: "How do I reset or change my password?" },
  responses: [
    { type: "text", body: "Go to Settings > Security > Reset Password." },
    { type: "text", body: "You can also use the 'Forgot Password' link on the login
page." },
  ],
},
```

```
tags: ["account", "security"],
});
```

Output

```
{
  "knowledgeBaseId": "kb-a1b2c3d4-...",
  "articleId": "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
  "question": {
    "text": "How do I reset or change my password?",
    "messageId": "5f767bcd-462d-4d2e-9651-408459d06b42"
  },
  "responses": [
    {
      "type": "text",
      "body": "Go to Settings > Security > Reset Password.",
      "messageId": "b2edcfab-b436-4c4e-8d0d-7cbc4cba0643"
    },
    {
      "type": "text",
      "body": "You can also use the 'Forgot Password' link on the login page.",
      "messageId": "75c2d1b0-f98c-4db6-abfa-8e0aad41fa56"
    }
  ],
  "tags": ["account", "security"],
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteKnowledgeBaseArticle

Deletes an article.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
articleId	string	Yes

Sample Request

```
await client.send(new DeleteKnowledgeBaseArticleCommand({
  knowledgeBaseId: "kb-a1b2c3d4-...",
  articleId: "art-e5f6g7h8-5678-90ab-cdef-1234567890ab",
}));
```

Output

No response body.

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

Request Parameters

knowledgeBaseld

Type: String

The knowledge base that contains the articles.

articleId

Type: String

The article ID.

question

Type: Object

question.text

Type: String

Required. The article's question or title. This is what gets matched against user queries.

question.messageId

Type: String

Generated upon Create.

question.skipTranslation

Type: Boolean

question.translated

Type: Boolean

responses

Type: Array

Response messages returned when this article matches. Each entry is a message object with a body field.

Example:

```
[
  { "type": "text", "body": "Go to Settings > Security > Reset Password." },
  { "type": "text", "body": "If you still have trouble, contact support." }
]
```

articleMetadata

Type: Object

Free-form metadata for the article (structure defined by the knowledge base's metadataSchema).

payload

Type: String

Raw content for the article. Max 10,000 characters.

tags

Type: Array

Classification tags for the article (max 5, each max 256 characters).

createdAt

Type: String

When the article was created (ISO 8601).

updatedAt

Type: String

When the article was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the article.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

KnowledgeBase Documents

Manage uploaded files within a knowledge base. Documents are registered via the API and uploaded via pre-signed URLs.

Contents

- [ListKnowledgeBaseDocuments](#)
- [GetKnowledgeBaseDocument](#)
- [PutKnowledgeBaseDocument](#)
- [DeleteKnowledgeBaseDocument](#)
- [Request Parameters](#)

ListKnowledgeBaseDocuments

Lists all documents in a knowledge base.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
client.send(new ListKnowledgeBaseDocumentsCommand({
    knowledgeBaseId: KNOWLEDGE_BASE_ID,
}));
```

Output

```
{
  "items": [
    {
      "documentId": "product-guide-v2.pdf",
      "uploadStatus": "UPLOADED",
      "contentType": "application/pdf",
      "customerMetadata": { "category": "guides" },
      "createdAt": "2026-08-01T12:00:00.000Z"
    }
  ]
}
```

```
  ],  
  "nextToken": null  
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

GetKnowledgeBaseDocument

Gets a pre-signed download URL for a document.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>documentId</code>	string	Yes

Sample Request

```
await client.send(new GetKnowledgeBaseDocumentCommand({  
    knowledgeBaseId: KNOWLEDGE_BASE_ID,  
    documentId: "product-guide-v2.pdf",  
}));
```

Output

```
{  
  "url": "https://s3.amazonaws.com/bucket/path/document.pdf?X-Amz-Signature=..."  
}
```

Errors

- `ValidationException` (400)

- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

PutKnowledgeBaseDocument

Registers a document and returns a pre-signed upload URL. Use the returned URL and fields to upload the file content.

Input

Parameter	Type	Required
<code>knowledgeBaseId</code>	string	Yes
<code>documentId</code>	string	Yes
<code>contentType</code>	string	Yes
<code>customerMetadata</code>	object	No

Sample Request

```
await client.send(new PutKnowledgeBaseDocumentCommand({
  knowledgeBaseId: KNOWLEDGE_BASE_ID,
  documentId: "product-guide-v2.pdf",
  contentType: "application/pdf",
  customerMetadata: { category: "guides", version: "2.0" },
}));
```

Output

```
{
  "url": "https://s3.amazonaws.com/bucket/upload-path",
  "fields": {
    "key": "...",
    "policy": "...",
    "x-amz-signature": "..."
  }
}
```

Note

Use the returned url and fields to perform a multipart form upload of your document content.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

DeleteKnowledgeBaseDocument

Deletes a document.

Input

Parameter	Type	Required
knowledgeBaseId	string	Yes
documentId	string	Yes

Sample Request

```
await client.send(new DeleteKnowledgeBaseDocumentCommand({
  knowledgeBaseId: KNOWLEDGE_BASE_ID,
  documentId: "product-guide-v2.pdf",
}));
```

Output

No response body.

Errors

- `ResourceNotFoundException` (404)

- `ValidationException` (400)
- `InternalServerErrorException` (500)

Request Parameters

`knowledgeBaseId`

Type: String

The knowledge base that contains the documents.

`documentId`

Type: String

The document identifier. User-provided, max 255 characters (e.g., a filename like `product-guide-v2.pdf`).

`contentType`

Type: String

MIME type of the document (e.g., `application/pdf` , `text/plain` , `text/html`).

`customerMetadata`

Type: Object

Free-form metadata for the document (structure defined by the knowledge base's `metadataSchema`).

`uploadStatus`

Type: String

The document's upload status. One of: `PENDING` (registered but not yet uploaded), `UPLOADED` (file received), `DELETED` .

`url`

Type: String

Pre-signed S3 URL for upload or download.

fields

Type: Object

Form fields required for multipart upload (returned by PutKnowledgeBaseDocument).

createdAt

Type: String

When the document was registered (ISO 8601).

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

Live Sync Scripts

Contents

- [ListLiveSyncScripts](#)
- [CreateLiveSyncScript](#)
- [GetLiveSyncScript](#)
- [UpdateLiveSyncScript](#)
- [DeleteLiveSyncScript](#)
- [ListLiveSyncScriptBuilds](#)
- [CreateLiveSyncScriptBuild](#)
- [GetLiveSyncScriptBuild](#)
- [ListLiveSyncScriptDeployments](#)
- [CreateLiveSyncScriptDeployment](#)
- [GetLiveSyncScriptDeployment](#)
- [UpdateLiveSyncScriptDeployment](#)

- [DeleteLiveSyncScriptDeployment](#)
- [Request Parameters](#)
- [Metadata](#)
- [Step](#)

ListLiveSyncScripts

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListLiveSyncScriptsCommand({}));
```

Output

```
{
  "items": [
    {
      "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
      "name": "test-script",
      "steps": [
        {
          "stepId": "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
          "body": "Welcome! Can I help you find something?",
          "name": "welcome-step",
          "action": "continue",
          "trigger": {
            "event": "pageLoad",
            "once": true
          }
        }
      ]
    },
    {
      "createdAt": "2026-08-24T18:03:53.046Z",
```

```
    "updatedAt": "2026-08-24T18:03:53.046Z",
    "description": "Created via SDK",
    "metadata": {
      "path": "/sdk-tests",
      "tags": [
        "test"
      ]
    },
    "updatedBy": "ci-DeployBot"
  }
]
```

Errors

- `ValidationException` (400)
- `InternalServerError` (500)

CreateLiveSyncScript

Input

Parameter	Type	Required
name	string	Yes
steps	array	Yes
description	string	No
metadata	object	No

Sample Request

```
await client.send(new CreateLiveSyncScriptCommand({
  name: 'test-script',
  description: 'Created via SDK',
  steps: [
    {
      stepId: "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
```

```

        name: 'welcome-step',
        body: 'Welcome! Can I help you find something?',
        action: 'continue',
        trigger: { event: 'pageLoad', once: true },
    },
],
metadata: { path: '/sdk-tests', tags: ['test'] },
));

```

Output

```

{
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "name": "test-script",
  "steps": [
    {
      "stepId": "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
      "body": "Welcome! Can I help you find something?",
      "name": "welcome-step",
      "action": "continue",
      "trigger": {
        "event": "pageLoad",
        "once": true
      }
    }
  ],
  "createdAt": "2026-08-24T18:03:53.046Z",
  "updatedAt": "2026-08-24T18:03:53.046Z",
  "description": "Created via SDK",
  "metadata": {
    "path": "/sdk-tests",
    "tags": [
      "test"
    ]
  },
  "apiKey": "aB3dEfGh1JkLmN0pQrStU=vWxYz2A4bCd",
  "updatedBy": "ci-DeployBot"
}

```

Errors

- `ValidationException` (400)
- `ConflictException` (409)

- `InternalServerError` (500)

GetLiveSyncScript

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetLiveSyncScriptCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
}));
```

Output

```
{
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "name": "test-script",
  "steps": [
    {
      "stepId": "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
      "body": "Welcome! Can I help you find something?",
      "name": "welcome-step",
      "action": "continue",
      "trigger": {
        "event": "pageLoad",
        "once": true
      }
    }
  ],
  "createdAt": "2026-08-24T18:03:53.046Z",
  "updatedAt": "2026-08-24T18:03:53.046Z",
  "description": "Created via SDK",
  "metadata": {
    "path": "/sdk-tests",
    "tags": [
```

```
    "test"
  ]
},
"apiKey": "aB3dEfGh1JkLmN0pQrStU=vWxYz2A4bCd",
"updatedBy": "ci-DeployBot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

UpdateLiveSyncScript

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes
<code>name</code>	string	No
<code>steps</code>	array	No
<code>description</code>	string	No
<code>metadata</code>	object	No

Sample Request

```
await client.send(new UpdateLiveSyncScriptCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  name: 'test-script',
  description: 'Updated via SDK',
  steps: [
    {
      stepId: "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
```

```

        name: 'welcome-step',
        body: 'Welcome! Can I help you find something?',
        action: 'continue',
        trigger: { event: 'pageLoad', once: true },
    },
],
metadata: { path: '/sdk-tests', tags: ['test'] },
}));

```

Output

```

{
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "name": "test-script",
  "steps": [
    {
      "stepId": "d4e5f6a7-8b9c-4d0e-1f2a-3b4c5d6e7f80",
      "body": "Welcome! Can I help you find something?",
      "name": "welcome-step",
      "action": "continue",
      "trigger": {
        "event": "pageLoad",
        "once": true
      }
    }
  ],
  "createdAt": "2026-08-24T18:03:53.046Z",
  "updatedAt": "2026-08-24T18:03:53.046Z",
  "description": "Updated via SDK",
  "metadata": {
    "path": "/sdk-tests",
    "tags": [
      "test"
    ]
  },
  "apiKey": "aB3dEfGh1JkLmN0pQrStU=vWxYz2A4bCd",
  "updatedBy": "ci-DeployBot"
}

```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)

- `InternalServerError` (500)

DeleteLiveSyncScript

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes

Sample Request

```
await client.send(new DeleteLiveSyncScriptCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

ListLiveSyncScriptBuilds

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes
<code>nextToken</code>	string	No

Parameter	Type	Required
maxResults	integer	No

Output

Returns items (a list of builds) and an optional nextToken.

Sample Request

```
await client.send(new ListLiveSyncScriptBuildsCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  maxResults: 10
}));
```

Sample Response

```
{
  "items": [
    {
      "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
      "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
      "status": "PENDING",
      "createdAt": "2026-08-24T18:35:15.164Z",
      "updatedAt": "2026-08-24T18:35:15.164Z",
      "description": "first build",
      "languageSettings": [
        {
          "languageCode": "en-US"
        }
      ]
    }
  ],
  "nextToken": null
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)

- `InternalServerError (500)`

CreateLiveSyncScriptBuild

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes
<code>description</code>	string	No
<code>languageSettings</code>	array	No

Sample Request

```
await client.send(new CreateLiveSyncScriptBuildCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  description: 'first build',
  languageSettings: [
    { languageCode: 'en-US' }
  ],
}));
```

Output

```
{
  "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "status": "PENDING",
  "description": "first build",
  "languageSettings": [
    {
      "languageCode": "en-US"
    }
  ],
  "createdAt": "2026-08-24T18:35:15.164Z",
  "updatedAt": "2026-08-24T18:35:15.164Z"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerError` (500)

GetLiveSyncScriptBuild

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes
<code>buildIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetLiveSyncScriptBuildCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  buildIdentifier: "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
}));
```

Output

```
{
  "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "status": "PENDING",
  "description": "first build",
  "languageSettings": [
    {
      "languageCode": "en-US"
    }
  ],
  "createdAt": "2026-08-24T18:35:15.164Z",
  "updatedAt": "2026-08-24T18:35:15.164Z"
```

```
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

ListLiveSyncScriptDeployments

Input

Parameter	Type	Required
<code>liveSyncScriptIdentifier</code>	string	Yes
<code>nextToken</code>	string	No
<code>maxResults</code>	integer	No

Sample Request

```
await client.send(new ListLiveSyncScriptDeploymentsCommand({  
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",  
  maxResults: 10  
}));
```

Output

```
{  
  "items": [  
    {  
      "deploymentId": "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",  
      "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",  
      "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",  
      "description": "first deployment",  
      "deploymentStatus": "scheduled",  
      "environment": "development",
```

```
        "createdAt": "2026-08-24T18:45:52.812Z",
        "updatedAt": "2026-08-24T18:46:30.523Z"
      }
    ],
    "nextToken": null
  }
}
```

Errors

- ResourceNotFoundException (404)
- ValidationException (400)
- InternalServerErrorException (500)

CreateLiveSyncScriptDeployment

Input

Parameter	Type	Required
liveSyncScriptIdentifier	string	Yes
buildIdentifier	string	Yes
description	string	No
environment	string	No
languageCodes	array	No
analyticsTags	array	No

Sample Request

```
await client.send(new CreateLiveSyncScriptDeploymentCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  buildIdentifier: "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  description: 'first deployment',
  environment: 'development'
})
```

```
});
```

Output

```
{
  "deploymentId": "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  "description": "first deployment",
  "deploymentStatus": "scheduled",
  "environment": "development",
  "createdAt": "2026-08-24T18:45:52.812Z",
  "updatedAt": "2026-08-24T18:45:52.812Z"
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

GetLiveSyncScriptDeployment

Input

Parameter	Type	Required
liveSyncScriptIdentifier	string	Yes
deploymentIdentifier	string	Yes

Sample Request

```
await client.send(new GetLiveSyncScriptDeploymentCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  deploymentIdentifier: "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f"
}));
```

Output

```
{
  "deploymentId": "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  "description": "first deployment",
  "deploymentStatus": "scheduled",
  "environment": "development",
  "createdAt": "2026-08-24T18:45:52.812Z",
  "updatedAt": "2026-08-24T18:45:52.812Z"
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

UpdateLiveSyncScriptDeployment

Input

Parameter	Type	Required
liveSyncScriptIdentifier	string	Yes
deploymentIdentifier	string	Yes
buildIdentifier	string	Yes
description	string	No
environment	string	No
languageCodes	array	No
analyticsTags	array	No

Output

Returns the updated deployment.

Sample Request

```
await client.send(new UpdateLiveSyncScriptDeploymentCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  deploymentIdentifier: "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",
  buildIdentifier: "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  description: 'updated deployment',
  environment: 'development',
}));
```

Sample Response

```
{
  "deploymentId": "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",
  "liveSyncScriptId": "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  "buildId": "b2c3d4e5-6f7a-4b8c-9d0e-1f2a3b4c5d6e",
  "description": "updated deployment",
  "deploymentStatus": "scheduled",
  "environment": "development",
  "createdAt": "2026-08-24T18:45:52.812Z",
  "updatedAt": "2026-08-24T18:45:52.812Z"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteLiveSyncScriptDeployment

Input

Parameter	Type	Required
liveSyncScriptIdentifier	string	Yes
deploymentIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteLiveSyncScriptDeploymentCommand({
  liveSyncScriptIdentifier: "a1b2c3d4-5e6f-4a7b-8c9d-0e1f2a3b4c5d",
  deploymentIdentifier: "c3d4e5f6-7a8b-4c9d-0e1f-2a3b4c5d6e7f",
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Request Parameters

`liveSyncScriptIdentifier`

Type: String

The unique identifier for the Live Sync Script (a UUIDv4, assigned on creation) used in Get, Update, Delete, and all build/deployment operations.

`buildIdentifier`

Type: String

The unique identifier for a build (a UUIDv4). Referenced when getting a build or creating/ updating a deployment.

`deploymentIdentifier`

Type: String

The unique identifier for a deployment (a UUIDv4).

`name`

Type: String

Live Sync Script name. 1–100 characters; letters, numbers, spaces, underscores, and hyphens.

`description`

Type: String

Live Sync Script, build, or deployment description. Max 200 characters.

`steps`

Type: Array

The ordered body of the script. Each entry is a step object. See [Step](#).

`metadata`

Type: Object

Organizational metadata. See [Metadata](#).

`languageSettings`

Type: Array

Per-language build configuration. Same shape as the application language settings. See [Common Types](#).

`environment`

Type: String

Target environment for the deployment. One of: development, qa, staging, production.

languageCodes

Type: Array

Languages included in the deployment. See [Common Types](#).

analyticsTags

Type: Array

Analytics tag references applied to the deployment: [{ "label": "..."}].

apiKey

Type: String (sensitive)

The script's API key, returned on Create/Get/Update of the full resource. Omitted from list summaries.

createdAt

Type: String

When the resource was created (ISO 8601).

updatedAt

Type: String

When the resource was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the resource.

status

Type: String

Build status.

deploymentStatus

Type: String

Deployment status.

Metadata

Field	Type	Required
path	string	No
tags	array	No

`metadata.path`

Type: String

Organizational path for the script. Max 512 characters.

`metadata.tags`

Type: Array

Classification tags (max 5; each 1–256 characters).

Step

A single step in a Live Sync Script's steps array.

Field	Type	Required
stepId	string	Yes
name	string	No
description	string	No
action	string	No
group	string	No
body	string	Yes

Field	Type	Required
skipTranslation	boolean	No
translated	boolean	No
variations	array	No
trigger	object	No
stateModifications	array	No
tags	array	No

`step.stepId`

Type: String

Unique identifier for the step (a UUIDv4).

`step.name`

Type: String

Step name. Max 100 characters.

`step.description`

Type: String

Step description. Max 200 characters.

`step.action`

Type: String

Terminal action for the step. One of: escalate, end, continue.

`step.group`

Type: String

Optional grouping label for the step. Max 100 characters.

`step.body`

Type: String

The step's message content shown to the user. Max 1000 characters.

`step.skipTranslation`

Type: Boolean

Whether to skip translation of this step's content.

`step.translated`

Type: Boolean

Whether this step's content has been translated.

`step.variations`

Type: Array

Alternative message bodies for A/B testing. Each entry contains body (max 1000 chars), percentage (0–100), and optional tags.

`step.trigger`

Type: Object

Defines when the step fires on the page. Contains:

- `event`: one of `click`, `pageLoad`, `appear`, `enterViewport`.
- `query`: a selector/query document identifying the target element.
- `once`: boolean; fire only once.
- `highlight`: boolean; highlight the target element.
- `urlCondition`: an object { `"operator"`: `"contains"` | `"matches_regex"` | `"smart_match"`, `"value"`: `"..."` } restricting the step to matching URLs.

`step.stateModifications`

Type: Array

State values written when the step runs. Each entry is a state modification object:

- `type`: context.
- `name`: the state variable name.
- `modification`: one of `clear`, `set`, `increment`, `decrement`, `push`, `pop`, `custom`.
- `functionName`: function name (used with the custom modification).
- `value`: an operand object describing the value source.

See [Common Types](#).

`step.tags`

Type: Array

Analytics tag references applied to the step: `[{ "label": "..."}]`.

Logs

Query conversation event logs with time-based and attribute-based filters.

Contents

- [QueryLogs](#)

QueryLogs

Queries logs from the workspace.

Input

Parameter	Type	Required
<code>timeFilter</code>	object	Yes
<code>searchFilter</code>	object	No
<code>sortOrder</code>	string	No
<code>maxResults</code>	integer	No
<code>nextToken</code>	string	No

Sample Request

Query with relative time filter (last 7 days)

```
await client.send(new QueryLogsCommand({
  timeFilter: {
    relative: { span: "604800000" },
  },
  maxResults: 10,
}));
```

Query with absolute time filter

```
const now = new Date();
const oneDayAgo = new Date(now.getTime() - 24 * 60 * 60 * 1000);
const absolute = await client.send(new QueryLogsCommand({
  timeFilter: {
    absolute: {
      startTimestamp: oneDayAgo.toISOString(),
      endTimestamp: now.toISOString(),
    },
  },
  maxResults: 10,
}));
```

Query with search filter (by event type)

```
const filtered = await client.send(new QueryLogsCommand({
  timeFilter: {
    relative: { span: "604800000" },
  },
  searchFilter: {
    eventType: "ConversationStarted",
  },
  maxResults: 10,
}));
```

Output

```
{
  "queryStatus": {
```

```

    "progressPercentage": 100.0,
    "cumulativeBytesScanned": 524288,
    "cumulativeBytesMetered": 1048576
  },
  "items": [
    {
      "eventType": "ConversationStarted",
      "eventTime": "2026-08-01T12:00:01.000Z",
      "commonProperties": [
        { "key": "conversationId", "value": "conv-uuid" },
        { "key": "applicationId", "value": "05c3fcc2-..." },
        { "key": "userId", "value": "user-123" }
      ],
      "eventProperties": [
        { "key": "buildId", "value": "b6c9ccd8-..." },
        { "key": "nodeId", "value": "f2a20ad1-..." },
        { "key": "messages", "value": ["Let me connect you with a specialist who can help."] },
        { "key": "responseTime", "value": 137 },
        { "key": "languageCode", "value": "en-US" }
      ]
    },
    {
      "eventType": "NlpInvoked",
      "eventTime": "2026-08-01T12:00:02.000Z",
      "commonProperties": [
        { "key": "conversationId", "value": "conv-uuid" }
      ],
      "eventProperties": [
        { "key": "utterance", "value": "I need help with my order" },
        { "key": "flowId", "value": "SupportFlow" }
      ]
    },
    ...
  ],
  "nextToken": "eyJvZmZzZXQiOi0jIw=="
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

Request Parameters

`timeFilter`

Type: Object

Time range filter. Provide exactly one of `relative` or `absolute`. See [Time Filter](#).

`searchFilter`

Type: Object

Attribute filters to narrow results. See [Search Filter](#).

`sortOrder`

Type: String

Sort direction: `asc` or `desc`.

`maxResults`

Type: Integer

Max items per page (5–100). See [Common Types](#).

`nextToken`

Type: String

Pagination token. See [Common Types](#).

Response Fields

`queryStatus`

Type: Object

Progress information for the query.

`queryStatus.progressPercentage`

Type: Number

Query completion percentage (0–100).

`queryStatus.cumulativeBytesScanned`

Type: Integer

Total bytes scanned so far.

`queryStatus.cumulativeBytesMetered`

Type: Integer

Total bytes metered for billing.

`items`

Type: Array

List of log entries.

`items.eventType`

Type: String

The event type. See Event Types.

`items.eventTime`

Type: String

When the event occurred (ISO 8601).

`items.commonProperties`

Type: Array

Properties common across events (e.g., `conversationId`, `applicationId`). Each entry: { `"key": "...", "value": "..."` }.

`items.eventProperties`

Type: Array

Properties specific to this event type (e.g., `utterance`, `flowId`). Each entry: { `"key": "...", "value": "..."` }.

Time Filter

Provide exactly one variant:

relative

Type: Object

A relative time span from now.

```
{ "relative": { "span": "86400000" } } // 24hrs = 86400000 ms
```

relative.span

Type: String

absolute

Type: Object

An explicit start and end time.

```
{ "absolute": { "startTimestamp": "2026-08-01T00:00:00.000Z", "endTimestamp":  
  "2026-08-01T23:59:59.000Z" } }
```

absolute.startTimestamp

Type: String

Start of time range (ISO 8601).

absolute.endTimestamp

Type: String

End of time range (ISO 8601).

Search Filter

Field	Type	Required
applicationId	string	No
buildId	string	No
conversationId	string	No

Field	Type	Required
correlationId	string	No
deploymentId	string	No
languageCode	string	No
userId	string	No
eventType	string	No

applicationId

Type: String

Filter by application.

buildId

Type: String

Filter by build.

conversationId

Type: String

Filter by conversation.

correlationId

Type: String

Filter by correlation ID.

deploymentId

Type: String

Filter by deployment.

languageCode

Type: String

Filter by language. See [Common Types](#).

`userId`

Type: String

Filter by end-user ID. Max 256 characters.

`eventType`

Type: String

Filter by event type. See [Event Types](#).

Event Types

Event Type	Description
<code>AutoEscalationState</code>	Conversation auto-escalated to a human
<code>EscalationState</code>	Conversation enters an escalation state
<code>ChoiceSelected</code>	User selected a choice
<code>ConditionEvaluated</code>	A condition was evaluated
<code>DefaultFlowRetrieval</code>	The Default flow was retrieved
<code>FallbackState</code>	Fallback flow triggered, conversation entered a fallback state
<code>FrustrationState</code>	User frustration was detected
<code>IncomprehensionState</code>	Input could not be understood
<code>FlowCaptureState</code>	Flow captured user input
<code>FlowRedirection</code>	Conversation redirected to another flow
<code>NodeMetAllConditions</code>	A node's conditions were all satisfied
<code>NodeTraversal</code>	A flow node was traversed

Event Type	Description
RepeatState	A step was repeated
Redirection	Conversation redirected to another node
ResponseSkippedAwaitingUserTurn	Response skipped, awaiting user turn
StateValuesEjected	State values cleared
NluRequestReceived	An NLU request received
NluResponded	The NLU engine responded
LifecycleHookInvoked	Lifecycle hook was invoked
LifecycleHookResponded	The lifecycle hook responded
LifecycleHookValidationFailed	Lifecycle hook validation failed
WebhookResponseValidationFailed	Webhook response failed validation
AgentStarted	Agent Started
AgentEnded	Agent Ended
AgenticToolStart	A tool wired up to an agent starts

Modalities

Define input/output schemas for different interaction channels. Modalities describe the structured data shapes your application can send and receive.

Contents

- [ListModalities](#)
- [CreateModality](#)
- [GetModality](#)
- [UpdateModality](#)
- [DeleteModality](#)

- [Request Parameters](#)
- [Modality Schema](#)

ListModalities

Lists all modalities in the workspace.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListModalitiesCommand({}));
```

Output

```
{
  "items": [
    {
      "modalityId": "checkout_form",
      "schema": {
        "type": "object",
        "description": "Checkout form data",
        "properties": {
          "productId": { "type": "string", "description": "Product SKU" },
          "quantity": { "type": "number", "description": "Quantity to order" },
          "confirmed": { "type": "boolean", "description": "User confirmed" }
        }
      },
      "metadata": { "path": "/commerce", "tags": ["checkout"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "lastUpdatedBy": "ci-deploy-bot"
    }
  ],
}
```

```
"nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateModality

Creates a new modality.

Input

Parameter	Type	Required
modalityId	string	Yes
schema	object	Yes
metadata	object	No

Sample Request

```
await client.send(new CreateModalityCommand({
  modalityId: "checkout_form",
  schema: {
    type: "object",
    description: "Checkout form data",
    properties: {
      productId: { type: "string", description: "Product SKU" },
      quantity: { type: "number", description: "Quantity to order" },
      confirmed: { type: "boolean", description: "User confirmed" },
      ssn: { type: "string", description: "Social security number", isSensitive:
true },
    },
  },
  metadata: { path: "/commerce", tags: ["checkout"] },
}));
```

Output

```
{
  "modalityId": "checkout_form",
  "schema": {
    "type": "object",
    "description": "Checkout form data",
    "isSensitive": false,
    "properties": {
      "productId": {
        "type": "string",
        "description": "Product SKU",
        "isSensitive": false
      },
      "quantity": {
        "type": "number",
        "description": "Quantity to order",
        "isSensitive": false
      },
      "confirmed": {
        "type": "boolean",
        "description": "User confirmed",
        "isSensitive": false
      },
      "ssn": {
        "type": "string",
        "description": "Social security number",
        "isSensitive": true
      }
    }
  },
  "metadata": {
    "path": "/commerce",
    "tags": [
      "checkout"
    ]
  },
  "createdAt": "2026-08-10T17:56:03.985Z",
  "updatedAt": "2026-08-10T17:56:03.985Z"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerError` (500)

GetModality

Gets a single modality by ID.

Input

Parameter	Type	Required
<code>modalityIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetModalityCommand({
  modalityIdentifier: "checkout_form",
}));
```

Output

```
{
  "modalityId": "checkout_form",
  "schema": {
    "type": "object",
    "description": "Checkout form data",
    "isSensitive": false,
    "properties": {
      "productId": {
        "type": "string",
        "description": "Product SKU",
        "isSensitive": false
      },
      "quantity": {
        "type": "number",
```

```
    "description": "Quantity to order",
    "isSensitive": false
  },
  "confirmed": {
    "type": "boolean",
    "description": "User confirmed",
    "isSensitive": false
  },
  "ssn": {
    "type": "string",
    "description": "Social security number",
    "isSensitive": true
  }
}
},
"metadata": {
  "path": "/commerce",
  "tags": [
    "checkout"
  ]
},
"createdAt": "2026-08-10T17:56:03.985Z",
"updatedAt": "2026-08-10T17:56:03.985Z"
}
```

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

UpdateModality

Updates an existing modality. Only include fields you want to change.

Input

Parameter	Type	Required
modalityIdentifier	string	Yes

Parameter	Type	Required
schema	object	No
metadata	object	No

Sample Request

```
await client.send(new UpdateModalityCommand({
  modalityIdentifier: "checkout_form",
  schema: {
    type: "object",
    description: "Checkout form data - updated",
    properties: {
      productId: { type: "string", description: "Product SKU" },
      quantity: { type: "number", description: "Quantity to order" },
      confirmed: { type: "boolean", description: "User confirmed" },
      ssn: { type: "string", description: "Social security number", isSensitive:
true },
      couponCode: { type: "string", description: "Optional coupon code" },
    },
  },
}));
```

Output

```
{
  "modalityId": "checkout_form",
  "schema": {
    "type": "object",
    "description": "Checkout form data - updated",
    "isSensitive": false,
    "properties": {
      "quantity": {
        "type": "number",
        "description": "Quantity to order",
        "isSensitive": false
      },
      "productId": {
        "type": "string",
        "description": "Product SKU",
```

```
    "isSensitive": false
  },
  "confirmed": {
    "type": "boolean",
    "description": "User confirmed",
    "isSensitive": false
  },
  "couponCode": {
    "type": "string",
    "description": "Optional coupon code",
    "isSensitive": false
  },
  "ssn": {
    "type": "string",
    "description": "Social security number",
    "isSensitive": true
  }
}
},
"metadata": {
  "path": "/commerce",
  "tags": [
    "checkout"
  ]
},
"createdAt": "2026-08-10T17:56:03.985Z",
"updatedAt": "2026-08-10T17:56:06.157Z"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteModality

Deletes a modality.

Input

Parameter	Type	Required
modalityIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteModalityCommand({
  modalityIdentifier: "checkout_form",
}));
```

Output

No output returned. HTTP Status Code: 204.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Request Parameters

modalityId

Type: String

The modality identifier. Alphanumeric + underscores, cannot start with a digit. Max 50 characters.

modalityIdentifier

Type: String

The modality ID used in Get, Update, and Delete operations.

schema

Type: Object

The modality's data schema. A recursive structure defining the shape of data this modality handles. See [Modality Schema](#).

metadata

Type: Object

Organizational metadata. See [Common Types](#).

createdAt

Type: String

When the modality was created (ISO 8601).

updatedAt

Type: String

When the modality was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the modality.

nextToken

Type: String

Pagination token. See [Common Types](#).

maxResults

Type: Integer

Max items per page (1–500). See [Common Types](#).

Modality Schema

A recursive schema definition describing the structure of data for this modality.

Field	Type	Required
type	string	Yes
description	string	No
isSensitive	boolean	No
properties	object	No
items	object	No

type

Type: String

The data type. One of: `string`, `number`, `boolean`, `array`, `object`.

description

Type: String

Human-readable description of this field. Max 255 characters.

isSensitive

Type: Boolean

Whether this field contains sensitive data (will be masked in logs/exports).

properties

Type: Object

Nested field schemas. Required when `type` is `object`. A map of field names to schema objects (recursive).

items

Type: Object

Item schema. Required when type is array. A schema object describing each array element (recursive).

Example:

```
{
  "type": "object",
  "description": "User profile card",
  "properties": {
    "name": { "type": "string", "description": "Display name" },
    "age": { "type": "number" },
    "preferences": {
      "type": "array",
      "items": { "type": "string", "description": "A preference tag" }
    },
    "ssn": { "type": "string", "isSensitive": true }
  }
}
```

Programmatic Users

Manage machine identities that authenticate with the Platform SDK. Programmatic users are assigned role configurations that determine their access level.

Contents

- [ListProgrammaticUsers](#)
- [CreateProgrammaticUser](#)
- [GetProgrammaticUser](#)
- [UpdateProgrammaticUser](#)
- [DeleteProgrammaticUser](#)
- [Request Parameters](#)
- [Role Config](#)

ListProgrammaticUsers

Lists all programmatic users.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListProgrammaticUsersCommand({}));
```

Output

```
{
  "items": [
    {
      "userId": "u1a2b3c4-5678-90ab-cdef-1234567890ab",
      "teamId": "team-uuid",
      "name": "ci-deploy-bot",
      "roleConfig": {
        "accountRole": "administrator"
      },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z"
    }
  ],
  "nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateProgrammaticUser

Creates a new programmatic user.

Input

Parameter	Type	Required
name	string	Yes
roleConfig	object	Yes

Sample Request

```
await client.send(new CreateProgrammaticUserCommand({
  name: "ci-deploy-bot",
  roleConfig: {
    accountRole: "administrator",
  },
}));
```

Output

```
{
  "userId": "u1a2b3c4-5678-90ab-cdef-1234567890ab",
  "teamId": "team-uuid",
  "name": "ci-deploy-bot",
  "roleConfig": {
    "accountRole": "administrator"
  },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetProgrammaticUser

Gets a single programmatic user by ID.

Input

Parameter	Type	Required
userId	string	Yes

Sample Request

```
await client.send(new GetProgrammaticUserCommand({
  userId: "u1a2b3c4-5678-90ab-cdef-1234567890ab",
}));
```

Output

```
{
  "userId": "u1a2b3c4-5678-90ab-cdef-1234567890ab",
  "teamId": "team-uuid",
  "name": "ci-deploy-bot",
  "roleConfig": {
    "accountRole": "administrator"
  },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateProgrammaticUser

Updates a programmatic user's name or role configuration.

Input

Parameter	Type	Required
userId	string	Yes

Parameter	Type	Required
name	string	No
roleConfig	object	No

Sample Request

```
await client.send(new UpdateProgrammaticUserCommand({
  userId: "u1a2b3c4-5678-90ab-cdef-1234567890ab",
  name: "ci-deploy-bot-updated",
  roleConfig: {
    accountRole: "administrator",
  },
}));
```

Output

```
{
  "userId": "u1a2b3c4-5678-90ab-cdef-1234567890ab",
  "teamId": "team-uuid",
  "name": "ci-deploy-bot-updated",
  "roleConfig": {
    "accountRole": "administrator"
  },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

DeleteProgrammaticUser

Deletes a programmatic user. Fails if API keys are still associated, delete all keys first.

Input

Parameter	Type	Required
userId	string	Yes

Sample Request

```
await client.send(new DeleteProgrammaticUserCommand({
  userId: "u1a2b3c4-5678-90ab-cdef-1234567890ab",
}));
```

Output

No response body.

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- ConflictException (409) API keys still exist for this user
- InternalServerErrorException (500)

Request Parameters

userId

Type: String

The programmatic user ID.

teamId

Type: String

The team/account this user belongs to (read-only).

name

Type: String

Display name for the programmatic user. 1–128 characters.

`roleConfig`

Type: Object

Role configuration determining access level. See Role Config.

`createdAt`

Type: String

When the user was created (ISO 8601).

`updatedAt`

Type: String

When the user was last modified (ISO 8601).

`nextToken`

Type: String

Pagination token. See Common Types.

`maxResults`

Type: Integer

Max items per page. See Common Types.

Role Config

Provide exactly one variant:

`accountRole`

Type: String

Account-level role granting full access across all workspaces. Currently the only value is `administrator`.

```
{
  "roleConfig": {
    "accountRole": "administrator"
  }
}
```

```
}  
}
```

workspaceRoles

Type: Array

Workspace-scoped role assignments. Each entry grants access to a specific workspace with a designated role.

```
{  
  "roleConfig": {  
    "workspaceRoles": [  
      { "workspaceId": "ws-uuid-1", "role": "developer" },  
      { "workspaceId": "ws-uuid-2", "roleId": "custom-role-uuid" }  
    ]  
  }  
}
```

Workspace Role Assignment

Field	Type	Required
workspaceId	string	Yes
role	string	No
roleId	string	No

workspaceId

Type: String

The workspace this assignment applies to.

role

Type: String

A pre-defined role name. One of: administrator, developer, contentManager, readOnly. Provide role OR roleId, not both.

roleId

Type: String

A custom role ID (created via the Roles API). Provide role OR roleId, not both.

Roles

Manage permission roles that can be assigned to programmatic users. Roles define what actions a user can perform within a workspace.

Contents

- [ListRoles](#)
- [CreateRole](#)
- [GetRole](#)
- [UpdateRole](#)
- [DeleteRole](#)
- [GetRolePermissions](#)
- [Request Parameters](#)
- [Role Permission](#)
- [Condition Catalog](#)

ListRoles

Lists all roles.

Input

Parameter	Type	Required
type	string	No
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListRolesCommand({}));
```

Output

```
{
  "items": [
    {
      "roleId": "role-a1b2c3d4-5678-90ab-cdef-1234567890ab",
      "name": "Content Editor",
      "type": "studio",
      "description": "Can edit flows and knowledge bases but cannot deploy",
      "permissions": [
        { "permissionId": "ds:ListFlows", "effect": "allow" },
        { "permissionId": "ds:CreateFlow", "effect": "allow" },
        { "permissionId": "ds:UpdateFlow", "effect": "allow" },
        { "permissionId": "ds>DeleteFlow", "effect": "allow" },
        { "permissionId": "ds:CreateApplicationDeployment", "effect": "deny" }
      ],
      "conditionCatalog": [],
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "updatedBy": "ci-deploy-bot"
    }
  ],
  "nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateRole

Creates a new custom role.

Input

Parameter	Type	Required
name	string	Yes
type	string	No
description	string	No
permissions	array	No
conditionCatalog	array	No

Sample Request

```
await client.send(new CreateRoleCommand({
  name: "Content Editor",
  type: "studio",
  description: "Can edit flows but cannot deploy",
  permissions: [
    { permissionId: "ds:ListFlows", effect: "allow" },
    { permissionId: "ds:CreateFlow", effect: "allow" },
    { permissionId: "ds:UpdateFlow", effect: "allow" },
    { permissionId: "ds:DeleteFlow", effect: "allow" },
    { permissionId: "ds:CreateApplicationDeployment", effect: "deny" },
  ],
  conditionCatalog: [],
}));
```

Output

```
{
  "roleId": "a80d56c4-3829-4f63-9500-13948d0fea87",
  "name": "Content Editor",
  "type": "studio",
  "description": "Can edit flows but cannot deploy",
  "permissions": [
    { "permissionId": "ds:ListFlows", "effect": "allow" },
    { "permissionId": "ds:CreateFlow", "effect": "allow" },
    { "permissionId": "ds:UpdateFlow", "effect": "allow" },
```

```
{ "permissionId": "ds>DeleteFlow", "effect": "allow" },
  { "permissionId": "ds>CreateApplicationDeployment", "effect": "deny" }
],
"conditionCatalog": [],
"createdAt": "2026-08-10T20:35:16.003Z",
"updatedAt": "2026-08-10T20:35:16.003Z",
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetRole

Gets a single role by ID.

Input

Parameter	Type	Required
<code>roleId</code>	string	Yes

Sample Request

```
await client.send(new GetRoleCommand({
  roleId: "a80d56c4-3829-4f63-9500-13948d0fea87",
}));
```

Output

```
{
  "roleId": "a80d56c4-3829-4f63-9500-13948d0fea87",
  "name": "Content Editor",
  "type": "studio",
  "description": "Can edit flows but cannot deploy",
}
```

```
"permissions": [
  { "permissionId": "ds:ListFlows", "effect": "allow" },
  { "permissionId": "ds:CreateFlow", "effect": "allow" },
  { "permissionId": "ds:UpdateFlow", "effect": "allow" },
  { "permissionId": "ds>DeleteFlow", "effect": "allow" },
  { "permissionId": "ds:CreateApplicationDeployment", "effect": "deny" }
],
"conditionCatalog": [],
"createdAt": "2026-08-10T20:35:16.003Z",
"updatedAt": "2026-08-10T20:35:16.003Z",
"updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateRole

Updates an existing role. Only include fields you want to change.

Input

Parameter	Type	Required
<code>roleId</code>	string	Yes
<code>name</code>	string	No
<code>type</code>	string	No
<code>description</code>	string	No
<code>permissions</code>	array	No
<code>conditionCatalog</code>	array	No

Sample Request

```
await client.send(new UpdateRoleCommand({
  roleId: "a80d56c4-3829-4f63-9500-13948d0fea87",
  name: "Content Editor",
  type: "studio",
  description: "Updated - content editor with KB access",
  permissions: [
    { permissionId: "ds:ListFlows", effect: "allow" },
    { permissionId: "ds:CreateFlow", effect: "allow" },
    { permissionId: "ds:UpdateFlow", effect: "allow" },
    { permissionId: "ds>DeleteFlow", effect: "allow" },
    { permissionId: "ds:ListKnowledgeBases", effect: "allow" },
    { permissionId: "ds:CreateApplicationDeployment", effect: "deny" },
  ],
  conditionCatalog: [],
}));
```

Output

```
{
  "roleId": "a80d56c4-3829-4f63-9500-13948d0fea87",
  "name": "Content Editor",
  "type": "studio",
  "description": "Updated - content editor with KB access",
  "permissions": [
    { "permissionId": "ds:ListFlows", "effect": "allow" },
    { "permissionId": "ds:CreateFlow", "effect": "allow" },
    { "permissionId": "ds:UpdateFlow", "effect": "allow" },
    { "permissionId": "ds>DeleteFlow", "effect": "allow" },
    { "permissionId": "ds:ListKnowledgeBases", "effect": "allow" },
    { "permissionId": "ds:CreateApplicationDeployment", "effect": "deny" }
  ],
  "conditionCatalog": [],
  "createdAt": "2026-08-10T20:35:16.003Z",
  "updatedAt": "2026-08-10T20:35:16.003Z",
  "updatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException (400)`

- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

DeleteRole

Deletes a custom role. Fails if the role is still assigned to programmatic users.

Input

Parameter	Type	Required
<code>roleId</code>	string	Yes

Sample Request

```
await client.send(new DeleteRoleCommand({
  roleId: created.roleId,
}));
```

Output

No response body.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `ConflictException` (409) role still assigned to users
- `InternalServerError` (500)

GetRolePermissions

Gets the list of all available permissions that can be assigned to roles.

Input

Parameter	Type	Required
type	string	Yes

Sample Request

```
await client.send(new GetRolePermissionsCommand({
  type: "studio",
}));
```

Output

```
{
  "permissions": [
    { "permissionId": "ds:ListApplications", "effect": "allow" },
    { "permissionId": "ds:CreateApplication", "effect": "allow" },
    { "permissionId": "ds:GetApplication", "effect": "allow" },
    { "permissionId": "ds:ListFlows", "effect": "allow" },
    { "permissionId": "ds:CreateFlow", "effect": "allow" },
    { "permissionId": "ds:UpdateFlow", "effect": "allow" },
    { "permissionId": "ds>DeleteFlow", "effect": "allow" },
    { "permissionId": "ds:ListDataRequests", "effect": "allow" },
    { "permissionId": "ds:CreateDataRequest", "effect": "allow" },
    { "permissionId": "ds:ListSlotTypes", "effect": "allow" },
    { "permissionId": "ds:CreateApplicationDeployment", "effect": "allow" },
    { "permissionId": "ds:ListKnowledgeBases", "effect": "allow" }
  ]
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

Request Parameters

`roleId`

Type: String

The role ID.

`name`

Type: String

Role name. Alphanumeric + spaces/dashes, 3–36 characters.

`type`

Type: String

The role type. One of: `studio`, `voicecompass`, `voiceinsights`.

`description`

Type: String

Role description. Max 200 characters.

`permissions`

Type: Array

Permission assignments. See [Role Permission](#).

`conditionCatalog`

Type: Array

Conditional permission rules for fine-grained access control. See [Condition Catalog](#).

`nextToken`

Type: String

Pagination token. See [Common Types](#).

`maxResults`

Type: Integer

Max items per page (1–100). See [Common Types](#).

createdAt

Type: String

When the role was created (ISO 8601).

updatedAt

Type: String

When the role was last modified (ISO 8601).

updatedAtBy

Type: String

The identity of who last modified the role.

Role Permission

Field	Type	Required
permissionId	string	Yes
effect	string	Yes
conditionId	string	No

permissionId

Type: String

The permission identifier (e.g., `ds:CreateFlow`, `ds:ListApplications`). Use `GetRolePermissions` to discover available values.

effect

Type: String

Whether to allow or deny this permission. One of: `allow`, `deny`.

conditionId

Type: String

Optional reference to a condition in the condition catalog. When present, the permission only applies when the condition is met.

Condition Catalog

Conditions enable fine-grained access control (e.g., restrict a user to specific languages or resources).

Field	Type	Required
categoryId	string	No
subcategoryId	string	No
conditionId	string	No
conditions	object	No

categoryId

Type: String

Category grouping for the condition.

subcategoryId

Type: String

Subcategory grouping.

conditionId

Type: String

Unique condition ID. Referenced by permissions via conditionId.

conditions

Type: Object

The condition logic. Either a single condition or a composite:

Single condition:

```
{
  "condition": {
    "left": { "type": "languageCode", "value": "" },
    "right": { "type": "constant", "value": "en-US" },
    "operator": "EQ"
  }
}
```

Composite condition (multiple conditions combined):

```
{
  "composite": {
    "operator": "OR",
    "items": [
      { "condition": { "left": { "type": "languageCode" }, "right": { "type": "constant", "value": "en-US" }, "operator": "EQ" } },
      { "condition": { "left": { "type": "languageCode" }, "right": { "type": "constant", "value": "es-ES" }, "operator": "EQ" } }
    ]
  }
}
```

Condition Operand Types

Type	Description
languageCode	The language of the resource being accessed
resourceId	The ID of the resource being accessed
constant	A literal value to compare against

Condition Operators

Operator	Description
EQ	Equals

Operator	Description
NEQ	Not equals
PREFIX	Starts with
NOT_PREFIX	Does not start with
SUFFIX	Ends with
NOT_SUFFIX	Does not end with
CONTAINS	Contains
NOT_CONTAINS	Does not contain

Boolean Operators

Operator	Description
AND	All conditions must be true
OR	At least one condition must be true

Secrets

Store sensitive values (API keys, credentials, connection strings) encrypted at rest. Secrets can be referenced in data requests and integrations without exposing the value.

Contents

- [ListSecrets](#)
- [CreateSecret](#)
- [GetSecret](#)
- [UpdateSecret](#)
- [DeleteSecret](#)
- [Request Parameters](#)

ListSecrets

Lists all secrets in the workspace. Values are never included in list responses.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListSecretsCommand({}));
```

Output

```
{
  "items": [
    {
      "name": "stripe_api_key",
      "description": "Stripe production API key",
      "isSensitive": true,
      "metadata": { "path": "/integrations", "tags": ["payments"] },
      "createdAt": "2026-08-01T12:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z"
    }
  ],
  "nextToken": null
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateSecret

Creates a new secret.

Input

Parameter	Type	Required
name	string	Yes
secretValue	string	Yes
description	string	No
isSensitive	boolean	No
metadata	object	No

Sample Request

```
await client.send(new CreateSecretCommand({
  name: "stripe_api_key",
  secretValue: "sk_live_test123456789",
  description: "Stripe production API key",
  isSensitive: true,
}));
```

Output

```
{
  "name": "stripe_api_key",
  "secretValue": "*****89",
  "description": "Stripe production API key",
  "isSensitive": true,
  "metadata": {},
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409) a secret with this name already exists

- `InternalServerError` (500)

GetSecret

Gets a single secret by name, including its value.

Input

Parameter	Type	Required
<code>secretIdentifier</code>	string	Yes

Sample Request

```
await client.send(new GetSecretCommand({
  secretIdentifier: "stripe_api_key",
}));
```

Output

```
{
  "name": "stripe_api_key",
  "secretValue": "*****89",
  "description": "Stripe production API key",
  "isSensitive": true,
  "metadata": {},
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `InternalServerError` (500)

UpdateSecret

Updates an existing secret. Only include fields you want to change.

Input

Parameter	Type	Required
secretIdentifier	string	Yes
secretValue	string	No
description	string	No
isSensitive	boolean	No
metadata	object	No

Sample Request

```
const updated = await client.send(new UpdateSecretCommand({
  secretIdentifier: "stripe_api_key",
  description: "Updated - Stripe prod key (rotated Aug 2026)",
  secretValue: "sk_live_rotated_987654321",
}));
```

Output

```
{
  "name": "stripe_api_key",
  "secretValue": "*****1",
  "description": "Updated - Stripe prod key (rotated Aug 2026)",
  "isSensitive": true,
  "metadata": {},
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteSecret

Deletes a secret.

Input

Parameter	Type	Required
secretIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteSecretCommand({
  secretIdentifier: "stripe_api_key",
}));
```

Output

No response body.

Errors

- ResourceNotFoundException (404)
- InternalServerError (500)

Request Parameters

name

Type: String

The secret name. Alphanumeric + underscores, 3–100 characters.

secretIdentifier

Type: String

The secret name used in Get, Update, and Delete operations.

secretValue

Type: String

The secret value. 1–4096 characters. This value is encrypted at rest and never logged.

description

Type: String

Description. Max 200 characters.

isSensitive

Type: Boolean

Whether this secret contains sensitive data.

metadata

Type: Object

Organizational metadata. See Common Types.

createdAt

Type: String

When the secret was created (ISO 8601).

updatedAt

Type: String

When the secret was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the secret.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–500). See Common Types.

Slot Types

Custom entity types for extracting structured data from user input. A slot type defines a list of values (with optional synonyms) that the NLP engine uses to identify entities in conversation.

Contents

- [ListSlotTypes](#)
- [CreateSlotType](#)
- [GetSlotType](#)
- [UpdateSlotType](#)
- [DeleteSlotType](#)
- [Request Parameters](#)
- [Slot Type Value](#)

ListSlotTypes

Lists all slot types in the workspace.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListSlotTypesCommand({}));
```

Output

```
{
  "items": [
    {
      "slotTypeId": "ProductCategory",
```

```
    "values": [
      {
        "value": "Electronics",
        "valueId": "YgpzvBg05U0e-t29GbY_0",
        "synonyms": ["tech", "gadgets"]
      },
      {
        "value": "Clothing",
        "valueId": "Up978eurWeqncYG9bxVof",
        "synonyms": ["apparel", "fashion"]
      }
    ],
    "sensitive": false,
    "mainLanguageCode": "en-US",
    "languageCodes": ["en-US"],
    "description": "Product categories for the catalog",
    "metadata": { "path": "/commerce", "tags": ["catalog"] },
    "createdAt": "2026-08-01T12:00:00.000Z",
    "updatedAt": "2026-08-01T12:00:00.000Z",
    "lastUpdatedBy": "ci-deploy-bot"
  }
],
"nextToken": null
}
```

Errors

- ValidationException (400)
- InternalServerErrorException (500)

CreateSlotType

Creates a new slot type.

Input

Parameter	Type	Required
slotTypeId	string	Yes
values	array	Yes

Parameter	Type	Required
sensitive	boolean	No
mainLanguageCode	string	No
languageCodes	array	No
description	string	No
metadata	object	No

Sample Request

```
await client.send(new CreateSlotTypeCommand({
  slotTypeId: "ProductCategory",
  values: [
    { value: "Electronics", synonyms: ["tech", "gadgets", "devices"] },
    { value: "Clothing", synonyms: ["apparel", "fashion"] },
    { value: "Home", synonyms: ["household", "furniture"] },
  ],
  sensitive: false,
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  description: "Product categories for the catalog",
  metadata: { path: "/commerce", tags: ["catalog"] },
}));
```

Output

```
{
  "slotTypeId": "ProductCategory",
  "values": [
    {
      "value": "Electronics",
      "valueId": "YgpzvBg05U0e-t29GbY_0",
      "synonyms": ["tech", "gadgets", "devices"]
    },
    {
      "value": "Clothing",
      "valueId": "Up978eurWeqncYG9bxVof",
```

```
    "synonyms": ["apparel", "fashion"]
  },
  {
    "value": "Home",
    "valueId": "S8SBNDX0eJvRyQv1Vvkq0G",
    "synonyms": ["household", "furniture"]
  }
],
"sensitive": false,
"mainLanguageCode": "en-US",
"languageCodes": ["en-US"],
"description": "Product categories for the catalog",
"metadata": { "path": "/commerce", "tags": ["catalog"] },
"createdAt": "2026-08-01T12:00:00.000Z",
"updatedAt": "2026-08-01T12:00:00.000Z",
"lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetSlotType

Gets a single slot type by ID.

Input

Parameter	Type	Required
<code>slotTypeIdentifier</code>	string	Yes
<code>languageCode</code>	string	No

Sample Request

```
await client.send(new GetSlotTypeCommand({
```

```
slotTypeIdentifier: "ProductCategory",
}});
```

Output

```
{
  "slotTypeId": "ProductCategory",
  "values": [
    {
      "value": "Electronics",
      "valueId": "YgpzvBg05U0e-t29GbY_0",
      "synonyms": ["tech", "gadgets", "devices"]
    },
    {
      "value": "Clothing",
      "valueId": "Up978eurWeqncYG9bxVof",
      "synonyms": ["apparel", "fashion"]
    },
    {
      "value": "Home",
      "valueId": "S8SBNDX0eJvRyQv1Vkq0G",
      "synonyms": ["household", "furniture"]
    }
  ],
  "sensitive": false,
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "description": "Product categories for the catalog",
  "metadata": { "path": "/commerce", "tags": ["catalog"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateSlotType

Updates an existing slot type. Only include fields you want to change.

Input

Parameter	Type	Required
slotTypeIdentifier	string	Yes
values	array	No
sensitive	boolean	No
mainLanguageCode	string	No
languageCodes	array	No
description	string	No
metadata	object	No

Sample Request

```
await client.send(new UpdateSlotTypeCommand({
  slotTypeIdentifier: "ProductCategory",
  values: [
    { value: "Electronics", synonyms: ["tech", "gadgets", "devices"] },
    { value: "Clothing", synonyms: ["apparel", "fashion"] },
    { value: "Home", synonyms: ["household", "furniture"] },
    { value: "Sports", synonyms: ["fitness", "athletic", "outdoor"] },
  ],
  sensitive: false,
  mainLanguageCode: "en-US",
  languageCodes: ["en-US"],
  description: "Updated - product categories with sports",
  metadata: { path: "/commerce", tags: ["catalog"] },
}));
```

Output

```
{
  "slotTypeId": "ProductCategory",
  "values": [
    {
      "value": "Electronics",
      "valueId": "YgpzvBg05U0e-t29GbY_0",
      "synonyms": ["tech", "gadgets", "devices"]
    },
    {
      "value": "Clothing",
      "valueId": "Up978eurWeqncYG9bxVof",
      "synonyms": ["apparel", "fashion"]
    },
    {
      "value": "Home",
      "valueId": "S8SBNDX0eJvRyQv1Vvkq0G",
      "synonyms": ["household", "furniture"]
    },
    {
      "value": "Sports",
      "valueId": "YmMQgeQ1lEKErJCclW17Z",
      "synonyms": ["fitness", "athletic", "outdoor"]
    }
  ],
  "sensitive": false,
  "mainLanguageCode": "en-US",
  "languageCodes": ["en-US"],
  "description": "Updated - product categories with sports",
  "metadata": { "path": "/commerce", "tags": ["catalog"] },
  "createdAt": "2026-08-01T12:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "lastUpdatedBy": "ci-deploy-bot"
}
```

Errors

- `ResourceNotFoundException` (404)
- `ValidationException` (400)
- `InternalServerErrorException` (500)

DeleteSlotType

Deletes a slot type.

Input

Parameter	Type	Required
slotTypeIdentifier	string	Yes

Sample Request

```
await client.send(new DeleteSlotTypeCommand({
  slotTypeIdentifier: "ProductCategory",
}));
```

Output

No response body.

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

Request Parameters

slotTypeId

Type: String

The slot type identifier. Alphabetic characters only, 3–100 characters.

slotTypeIdentifier

Type: String

The slot type ID used in Get, Update, and Delete operations.

values

Type: Array

Slot type values. See Slot Type Value.

sensitive

Type: Boolean

Whether values contain sensitive data.

mainLanguageCode

Type: String

Primary language. See Common Types.

languageCode

Type: String

Get values in a specific language. See Common Types.

languageCodes

Type: Array

Supported languages. See Common Types.

description

Type: String

Description. Max 200 characters.

metadata

Type: Object

Organizational metadata. See Common Types.

createdAt

Type: String

When the slot type was created (ISO 8601).

updatedAt

Type: String

When the slot type was last modified (ISO 8601).

lastUpdatedBy

Type: String

The identity of who last modified the slot type.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–500). See Common Types.

Slot Type Value

Field	Type	Required
value	string	Yes
valueId	string	No
synonyms	array	No
skipTraining	boolean	No
skipTranslation	boolean	No
choicePayload	string	No

value

Type: String

The canonical value. 1–256 characters.

valueId

Type: String

Optional value identifier.

synonyms

Type: Array

Alternative phrases that map to this value (array of strings, each 1–256 characters).

Example:

```
{ "value": "Electronics", "synonyms": ["tech", "gadgets", "devices"] }
```

skipTraining

Type: Boolean

Exclude this value from NLP training.

skipTranslation

Type: Boolean

Skip auto-translation for this value.

choicePayload

Type: String

Payload sent when this value is selected in a choice node. Max 200 characters.

Team

Manage account-level team settings. Each account has a single team resource.

Contents

- [GetTeam](#)

GetTeam

Gets the team settings for the authenticated account.

Input

No parameters.

Sample Request

```
await client.send(new GetTeamCommand({}));
```

Output

```
{
  "teamId": "t1a2b3c4-5678-90ab-cdef-1234567890ab",
  "name": "Northwind Corp",
  "createdAt": "2026-07-30T18:39:14.741Z",
  "updatedAt": "2026-07-30T18:39:14.741Z"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Trails

Query the audit trail of changes made to workspace resources. Trails use an async query pattern, submit a query, then poll for results.

Contents

- [StartTrailQuery](#)
- [GetTrailQueryResults](#)
- [Request Parameters](#)
- [Response Fields](#)

- [Event Names](#)

StartTrailQuery

Submits an audit trail query. Returns a result ID for polling.

Input

Parameter	Type	Required
startTimestamp	string	Yes
endTimestamp	string	Yes
eventType	string	No
eventName	string	No
principalId	string	No
principalEmail	string	No
sourceIpAddress	string	No
page	integer	No
size	integer	No

Sample Request

```
await client.send(new StartTrailQueryCommand({
  startTimestamp: "2026-08-03T00:00:00.000Z",
  endTimestamp: "2026-08-10T23:59:59.000Z",
}));
```

StartTrailQuery with Filters

```
await client.send(new StartTrailQueryCommand({
  startTimestamp: "2026-08-10T00:00:00.000Z",
  endTimestamp: "2026-08-10T23:59:59.000Z",
```

```
    eventType: "WRITE",  
    size: 5,  
  }));
```

Output

```
{  
  "resultId": "qry-a1b2c3d4e5f6..."  
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

GetTrailQueryResults

Gets results of a previously submitted trail query.

Input

Parameter	Type	Required
resultId	string	Yes

Sample Request

```
await client.send(new GetTrailQueryResultsCommand({  
  resultId: "7eba7313-ad4d-48d9-acba-dbf2a0d492cc",  
}));
```

Output

```
{  
  "status": "SUCCEEDED",  
  "items": [  
    {
```

```
    "eventVersion": "1.0",
    "eventTime": "2026-08-10T18:07:08.008Z",
    "eventSource": "acxd.studio",
    "eventName": "FlowUpdate",
    "eventType": "WRITE",
    "userAgent": "ACXD-SDK/1.0",
    "requestId": "req-uuid",
    "requestParameters": "{\"flowId\":\"MainFlow\", \"customerId\":\"a018836f-...\"}\",
    "responseElements": "{\"statusCode\":\"200\", \"message\":\"Flow updated
successfully.\"}\",
    "userType": "ACXDStudioUser",
    "userId": "user-uuid",
    "tier": ""
  }
]
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)
- `ThrottlingException` (429)

Request Parameters

`resultId`

Type: String

The result ID returned by `StartTrailQuery`. Used to poll for results.

`startTimestamp`

Type: String

Start of time range (ISO 8601).

`endTimestamp`

Type: String

End of time range (ISO 8601).

eventType

Type: String

Filter by event type. One of: WRITE, DELETE.

eventName

Type: String

Filter by specific event name. See Event Names.

principalId

Type: String

Filter by the user who made the change. Alphanumeric, 24–30 characters.

principalEmail

Type: String

Filter by user email.

sourceIpAddress

Type: String

Filter by source IP address.

page

Type: Integer

Page number (0–100).

size

Type: Integer

Page size (1–100).

Response Fields

status

Type: String

Query status. One of: IN_PROGRESS (poll again), SUCCEEDED (results ready), FAILED, CANCELLED.

items

Type: Array

List of trail events. Each event contains:

eventVersion

Type: String

Event schema version.

eventTime

Type: String

When the event occurred (ISO 8601).

eventSource

Type: String

The source system (e.g., `acxd.studio`).

eventName

Type: String

The action that occurred. See Event Names.

eventType

Type: String

WRITE or DELETE.

sourceIpAddress

Type: String

The IP address of the caller.

userAgent

Type: String

The user agent string of the caller.

requestId

Type: String

Unique request identifier.

requestParameters

Type: String

The request parameters (JSON string).

responseElements

Type: String

The response elements (JSON string).

errorCode

Type: String

Error code if the action failed.

errorMessage

Type: String

Error message if the action failed.

userType

Type: String

Type of user who performed the action.

userId

Type: String

User identifier.

email

Type: String

User email.

userName

Type: String

User display name.

tier

Type: String

Account tier.

Event Names

Category	Events
Applications	ApplicationCreate , ApplicationUpdate , ApplicationDelete , ApplicationBatchUpdate
Builds	BuildCreate , BuildUpdate
Deployments	DeploymentCreate , DeploymentUpdate , DeploymentDelete
Conversation Flows	ConversationFlowCreate , ConversationFlowUpdate , ConversationFlowDelete , ConversationFlowCloneCreate , ConversationFlowBatchUpdate
Data Requests	DataRequestCreate , DataRequestUpdate , DataRequestDelete , DataRequestBatchUpdate
Guardrail	GuardrailCreate , GuardrailUpdate , GuardrailDelete , GuardrailBatchUpdate
Knowledge Bases	KnowledgeBaseCreate , KnowledgeBaseUpdate , KnowledgeBaseDelete , KnowledgeBaseCloneCreate , KnowledgeBaseBatchUpdate , KnowledgeBasePublish

Category	Events
KB Articles	KnowledgeBaseArticleCreate , KnowledgeBaseArticleUpdate , KnowledgeBaseArticleDelete
KB Documents	KnowledgeBaseDocumentUpload , KnowledgeBaseDocumentDelete
Secrets	SecretCreate , SecretUpdate , SecretDelete , SecretBatchUpdate
Slot Types	SlotCreate , SlotUpdate , SlotDelete , SlotBatchUpdate
Modalities	ModalityCreate , ModalityUpdate , ModalityDelete , ModalityBatchUpdate
Integrations	IntegrationCreate , IntegrationUpdate , IntegrationDelete
Analytics Tags	AnalyticsTagsCreate , AnalyticsTagsUpdate , AnalyticsTagsDelete , AnalyticsTagsBatchUpdate
Context Variables	ContextVariablesCreate , ContextVariablesUpdate , ContextVariablesDelete
Translations	TranslateContent , RequestTranslation , RequestTranslationExport , SupportedLanguagesUpdate , UpdateResourceSupportedLanguages
Downloads	DownloadConversations , RequestDownload , DownloadFile

Category	Events
Dashboards	DashboardCreate , DashboardUpdate , DashboardDelete
Monitors	AnalyticsMonitorCreate , AnalyticsSubscriptionCreate , ClustersMonitorCreate , MonitorUpdate , SubscriptionDelete
Notifications	NotificationAlertUpdate , NotificationAlertDelete
History Tabs	HistoryTabCreate , HistoryTabUpdate , HistoryTabDelete
Tests	TestCreate , TestUpdate , TestDelete , TestExecutionCreate
LiveSyncScripts	LiveSyncScriptCreate , LiveSyncScriptUpdate , LiveSyncScriptBatchUpdate , LiveSyncScriptDelete , LiveSyncScriptBuildCreate , LiveSyncScriptBuildDelete , LiveSyncScriptDeploymentCreate , LiveSyncScriptDeploymentUpdate , LiveSyncScriptDeploymentDelete
Lifecycle Hooks	LifecycleHookCreate , LifecycleHookUpdate , LifecycleHookDelete
Other	ApplyUpload , BatchDelete , DeleteUpload , UnknownMessagesUpdate , ResourceFoldersUpdate , ResourceTagsUpdate , TimezoneUpdate

Users

Manage users in the account. Users are human identities (as opposed to programmatic users which are machine identities).

Contents

- [ListUsers](#)
- [CreateUser](#)
- [GetUser](#)
- [UpdateUser](#)
- [DeleteUser](#)
- [Request Parameters](#)
- [User Role Assignment](#)
- [Default Role](#)

ListUsers

Lists all users.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListUsersCommand({}));
```

Output

```
{
  "items": [
    {
      "userId": "user-id-string",
```

```
    "cxnRole": "administrator",
    "userArn": "arn:aws:connect:us-west-2:123456789:instance/.../user/...",
    "username": "jane.doe",
    "firstName": "Jane",
    "lastName": "Doe",
    "email": "jane@example.com",
    "applicationIds": ["05c3fcc2-..."],
    "roles": [
      { "applicationId": "05c3fcc2-...", "role": "administrator" }
    ],
    "defaultRole": { "role": "administrator" },
    "createdAt": "2026-08-01T12:00:00.000Z",
    "updatedAt": "2026-08-01T12:00:00.000Z",
    "updatedBy": "admin-user"
  }
],
"nextToken": null
}
```

Errors

- ValidationException (400)
- InternalServerErrorException (500)

CreateUser

Creates a new user.

Input

Parameter	Type	Required
userId	string	Yes
cxnRole	string	Yes
userArn	string	Yes
username	string	Yes
firstName	string	No

Parameter	Type	Required
lastName	string	No
email	string	No
applicationIds	array	No
roles	array	No
defaultRole	object	No

Sample Request

```
await client.send(new CreateUserCommand({
  userId: "000000000-0000-4000-8000-0000000000099",
  cxnRole: "member",
  userArn: "arn:aws:connect:us-west-2:176202286863:instance/cb0ac315-73ce-4def-
a2cc-cf3caf9866cc/agent/000000000-0000-4000-8000-0000000000099",
  username: "sdk.testuser",
  firstName: "SDK",
  lastName: "TestUser",
  email: "sdk-test@example.com",
  applicationIds: ["a018836f-a909-4be0-a28b-53b413f1429c"],
  defaultRole: { role: "developer" },
  roles: [
    { applicationId: "a018836f-a909-4be0-a28b-53b413f1429c", role: "developer" },
  ],
}));
```

Output

```
{
  "userId": "000000000-0000-4000-8000-0000000000099",
  "cxnRole": "member",
  "userArn": "arn:aws:connect:us-west-2:176202286863:instance/cb0ac315-73ce-4def-a2cc-
cf3caf9866cc/agent/000000000-0000-4000-8000-0000000000099",
  "username": "sdk.testuser",
  "firstName": "SDK",
  "lastName": "TestUser",
  "email": "sdk-test@example.com",
```

```
{
  "applicationIds": [
    "a018836f-a909-4be0-a28b-53b413f1429c"
  ],
  "roles": [
    {
      "applicationId": "a018836f-a909-4be0-a28b-53b413f1429c",
      "role": "developer"
    }
  ],
  "defaultRole": {
    "role": "developer"
  },
  "createdAt": "2026-08-10T22:54:16.796Z",
  "updatedAt": "2026-08-10T22:54:16.796Z",
  "updatedBy": "deploy-ci-bot"
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerError` (500)

GetUser

Gets a single user by ID.

Input

Parameter	Type	Required
<code>userId</code>	string	Yes

Sample Request

```
await client.send(new GetUserCommand({
  userId: "00000000-0000-4000-8000-000000000099",
}));
```

Output

```
{
  "userId": "000000000-0000-4000-8000-0000000000099",
  "cxnRole": "member",
  "userArn": "arn:aws:connect:us-west-2:<account-id>:instance/cb0ac315-73ce-4def-a2cc-cf3caf9866cc/agent/000000000-0000-4000-8000-0000000000099",
  "username": "sdk.testuser",
  "firstName": "SDK",
  "lastName": "TestUser",
  "email": "sdk-test@example.com",
  "applicationIds": [
    "a018836f-a909-4be0-a28b-53b413f1429c"
  ],
  "roles": [
    {
      "applicationId": "a018836f-a909-4be0-a28b-53b413f1429c",
      "role": "developer"
    }
  ],
  "defaultRole": {
    "role": "developer"
  },
  "createdAt": "2026-08-10T22:54:16.796Z",
  "updatedAt": "2026-08-10T22:54:16.796Z",
  "updatedBy": "deploy-ci-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

UpdateUser

Updates a user. Only include fields you want to change.

Input

Parameter	Type	Required
userId	string	Yes
cxnRole	string	No
userArn	string	No
firstName	string	No
lastName	string	No
email	string	No
applicationIds	array	No
roles	array	No
defaultRole	object	No

Sample Request

```
await client.send(new UpdateUserCommand({
  userId: "000000000-0000-4000-8000-0000000000099",
  cxnRole: "member",
  userArn: "arn:aws:connect:us-west-2:<account-id>:instance/cb0ac315-73ce-4def-a2cc-
cf3caf9866cc/agent/000000000-0000-4000-8000-0000000000099",
  firstName: "SDKUpdated",
  lastName: "TestUser",
  email: "sdk-test@example.com",
  applicationIds: ["a018836f-a909-4be0-a28b-53b413f1429c"],
  roles: [
    { applicationId: "a018836f-a909-4be0-a28b-53b413f1429c", role: "developer" },
  ],
  defaultRole: { role: "contentManager" },
}));
```

Output

```
{
  "userId": "000000000-0000-4000-8000-0000000000099",
  "cxnRole": "member",
  "userArn": "arn:aws:connect:us-west-2:<account-id>:instance/cb0ac315-73ce-4def-a2cc-cf3caf9866cc/agent/000000000-0000-4000-8000-0000000000099",
  "username": "sdk.testuser",
  "firstName": "SDKUpdated",
  "lastName": "TestUser",
  "email": "sdk-test@example.com",
  "applicationIds": [
    "a018836f-a909-4be0-a28b-53b413f1429c"
  ],
  "roles": [
    {
      "applicationId": "a018836f-a909-4be0-a28b-53b413f1429c",
      "role": "developer"
    }
  ],
  "defaultRole": {
    "role": "contentManager"
  },
  "createdAt": "2026-08-10T22:54:16.796Z",
  "updatedAt": "2026-08-10T22:54:20.530Z",
  "updatedBy": "deploy-ci-bot"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

DeleteUser

Deletes a user.

Input

Parameter	Type	Required
userId	string	Yes

Sample Request

```
await client.send(new DeleteUserCommand({
  userId: created.userId,
}));
```

Output

No response body.

Errors

- ValidationException (400)
- ResourceNotFoundException (404)
- InternalServerErrorException (500)

Request Parameters

userId

Type: String

The user identifier.

cxnRole

Type: String

The user's account-level role. One of: member, administrator, owner.

userArn

Type: String

The user's Amazon Connect ARN.

username

Type: String

Username. 1–128 characters.

firstName

Type: String

First name. 1–36 characters.

lastName

Type: String

Last name. Max 36 characters.

email

Type: String

Email address. 3–256 characters.

applicationIds

Type: Array

Workspace IDs the user can access.

roles

Type: Array

Per-workspace role assignments. See [User Role Assignment](#).

defaultRole

Type: Object

Default role when no workspace-specific role applies. See [Default Role](#).

createdAt

Type: String

When the user was created (ISO 8601).

updatedAt

Type: String

When the user was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the user.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

User Role Assignment

Field	Type	Required
applicationId	string	Yes
role	string	Yes
roleId	string	No

applicationId

Type: String

The workspace this assignment applies to.

role

Type: String

Pre-defined role name. One of: administrator, developer, contentManager, readOnly.
roleId

Type: String

Custom role ID (alternative to role). Created via the Roles API.

Default Role

The fallback role applied when the user accesses a workspace without a specific role assignment.

Field	Type	Required
role	string	Yes
roleId	string	No

role

Type: String

Pre-defined role name. One of: administrator, developer, contentManager, readOnly.
roleId

Type: String

Custom role ID (alternative to role).

Versions

Access version history for workspace resources. Every change to a versioned resource creates a new version that can be retrieved later.

Contents

- [ListResourceVersions](#)
- [GetResourceVersion](#)
- [Request Parameters](#)

ListResourceVersions

Lists the version history for a specific resource.

Input

Parameter	Type	Required
resourceType	string	Yes
resourceId	string	Yes
nextToken	string	No
maxResults	integer	No

Sample Request

```
client.send(new ListResourceVersionsCommand({
    resourceType: "flows",
    resourceId: "DeployTestFlow",
}));
```

Output

```
{
  "items": [
    {
      "versionId": "v-a1b2c3d4",
      "lastUpdatedBy": "ci-deploy-bot",
      "updatedAt": "2026-08-01T14:00:00.000Z",
      "isLatest": true,
      "isPublished": false
    },
    {
      "versionId": "v-e5f6g7h8",
      "lastUpdatedBy": "jane@example.com",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "isLatest": false,
      "isPublished": true
    }
  ]
}
```

```
  ],  
  "nextToken": null  
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

GetResourceVersion

Gets the full content of a specific version.

Input

Parameter	Type	Required
<code>versionId</code>	string	Yes
<code>resourceType</code>	string	Yes
<code>resourceId</code>	string	Yes

Sample Request

```
await client.send(new GetResourceVersionCommand({  
  versionId: "NDjL4SbcnwdK5w3fNBL6A4oFIzwRYeyE",  
  resourceType: "flows",  
  resourceId: "DeployTestFlow",  
}));
```

Output

```
{  
  "data": {  
    "flow": {  
      "flowId": "DeployTestFlow",  
      "description": "Flow for deployment testing",  
    }  
  }  
}
```

```
    "mainLanguageCode": "en-US",  
    "createdAt": "2026-08-08T16:12:06.767Z",  
    "updatedAt": "2026-08-08T16:12:06.767Z",  
    "updatedBy": "ci-deploy-bot"  
  }  
}  
}
```

The data field contains exactly one key matching the resource type, with the full resource content at that version.

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)
- `InternalServerErrorException` (500)

Request Parameters

versionId

Type: String

The version ID.

resourceType

Type: String

The type of versioned resource. One of: `slotTypes`, `flows`, `dataRequests`, `actions`, `lifecycleHooks`, `journeys`, `guardrails`, `feedbackConfigs`.

resourceId

Type: String

The identifier of the resource to retrieve versions for.

lastUpdatedBy

Type: String

Who made this version's change.

updatedAt

Type: String

When this version was created (ISO 8601).

isLatest

Type: Boolean

Whether this is the most recent version.

isPublished

Type: Boolean

Whether this version is currently published (live).

data

Type: Object

The full resource content at this version. Contains exactly one key corresponding to the resource type: `slotType`, `flow`, `dataRequest`, `action`, `lifecycleHook`, `journey`, `guardrail`, or `feedbackConfig`.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

Workspaces

Manage workspaces: isolated environments that contain all resources for a project or team. Workspace operations are account-level and do not require a workspace ID.

Contents

- [ListWorkspaces](#)
- [CreateWorkspace](#)
- [GetWorkspace](#)
- [UpdateWorkspace](#)
- [DeleteWorkspace](#)
- [Request Parameters](#)

ListWorkspaces

Lists all workspaces in the account.

Input

Parameter	Type	Required
nextToken	string	No
maxResults	integer	No

Sample Request

```
await client.send(new ListWorkspacesCommand({}));
```

Output

```
{
  "items": [
    {
      "workspaceId": "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
      "name": "Production",
      "tags": ["prod", "main"],
      "createdAt": "2026-01-15T10:00:00.000Z",
      "updatedAt": "2026-08-01T12:00:00.000Z",
      "updatedBy": "admin-user"
    }
  ]
}
```

```
  ],  
  "nextToken": null  
}
```

Errors

- `ValidationException` (400)
- `InternalServerErrorException` (500)

CreateWorkspace

Creates a new workspace.

Input

Parameter	Type	Required
name	string	Yes
tags	array	No

Sample Request

```
await client.send(new CreateWorkspaceCommand({  
  name: "SDK Test Workspace",  
  tags: ["testing", "sdk"],  
}));
```

Output

```
{  
  "workspaceId": "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",  
  "name": "SDK Test Workspace",  
  "tags": ["testing", "sdk"],  
  "createdAt": "2026-01-15T10:00:00.000Z",  
  "updatedAt": "2026-08-01T12:00:00.000Z",  
  "updatedBy": "admin-user"  
}
```

Errors

- `ValidationException` (400)
- `ConflictException` (409)
- `InternalServerErrorException` (500)

GetWorkspace

Gets a single workspace by ID.

Input

Parameter	Type	Required
<code>workspaceId</code>	string	Yes

Sample Request

```
await client.send(new GetWorkspaceCommand({
  workspaceId: "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
}));
```

Output

```
{
  "workspaceId": "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
  "name": "SDK Test Workspace",
  "tags": ["testing", "sdk"],
  "createdAt": "2026-01-15T10:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "updatedBy": "admin-user"
}
```

Errors

- `ValidationException` (400)
- `ResourceNotFoundException` (404)

- `InternalServerError (500)`

UpdateWorkspace

Updates a workspace.

Input

Parameter	Type	Required
<code>workspaceId</code>	string	Yes
<code>name</code>	string	No
<code>tags</code>	array	No

Sample Request

```
await client.send(new UpdateWorkspaceCommand({
  workspaceId: "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
  name: "SDK Test Workspace Updated",
  tags: ["testing", "sdk", "updated"],
}));
```

Output

```
{
  "workspaceId": "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
  "name": "SDK Test Workspace Updated",
  "tags": ["testing", "sdk", "updated"],
  "createdAt": "2026-01-15T10:00:00.000Z",
  "updatedAt": "2026-08-01T12:00:00.000Z",
  "updatedBy": "admin-user"
}
```

Errors

- `ValidationException (400)`
- `ResourceNotFoundException (404)`

- `InternalServerError (500)`

DeleteWorkspace

Deletes a workspace and all its resources.

Input

Parameter	Type	Required
<code>workspaceId</code>	string	Yes

Sample Request

```
await client.send(new DeleteWorkspaceCommand({
  workspaceId: "67898477-f36f-4b06-bdca-0d9a1f2eb9cc",
}));
```

Output

No response body.

Errors

- `ValidationException (400)`
- `ResourceNotFoundException (404)`
- `InternalServerError (500)`

Request Parameters

`workspaceId`

Type: String

The workspace ID.

`name`

Type: String

Workspace name. Alphanumeric + spaces/punctuation, 1–36 characters.

tags

Type: Array

Classification tags (max 5, each a string).

createdAt

Type: String

When the workspace was created (ISO 8601).

updatedAt

Type: String

When the workspace was last modified (ISO 8601).

updatedBy

Type: String

The identity of who last modified the workspace.

nextToken

Type: String

Pagination token. See Common Types.

maxResults

Type: Integer

Max items per page (1–100). See Common Types.

Common Types

Shared types used across multiple Platform SDK resources.

Contents

- [Metadata Object](#)
- [Pagination](#)
- [languageCode](#)
- [languageCodes](#)

Metadata Object

The metadata object is optional organizational metadata used to categorize and group resources within your workspace. It does not affect runtime behavior it is purely for organization and filtering in the agentic CX designer workspace.

Field	Type	Required	Description
path	string	No	A folder-like path for organizing resources in the agentic CX designer workspace (e.g., /support/tags , /production). Max 512 characters.
tags	string[]	No	Classification labels for filtering and grouping (max 5 tags, each max 256 characters).

Example

```
{
  "metadata": {
    "path": "/customer-support/sentiment",
    "tags": ["production", "support"]
  }
}
```

Pagination

nextToken

Type: String

An opaque pagination token returned in list responses when more results are available. Pass it back in the next request to retrieve the next page. When there are no more pages, `nextToken` is `null` or absent.

maxResults

Type: Integer

The maximum number of items to return per page. Default and maximum vary by resource (typically 1–500).

languageCode

Type: String

A BCP-47 language code identifying a single language (e.g., `en-US`, `es-ES`, `fr-FR`).

Supported Language Codes

Code	Code	Code	Code
ar-AE	ar-KW	ar-MEA	ar-QA
ar-SA	az-AZ	bg-BG	bs-BA
ca-ES	cs-CZ	da-DK	de-AT
de-CH	de-DE	el-GR	en-INT
en-CA	en-IE	en-IN	en-NZ
en-PH	en-PK	en-SG	en-ZA
en-EG	en-KE	en-KW	en-MEA
en-MY	en-NG	en-UAE	en-AE

Code	Code	Code	Code
en-AU	en-GB	en-QA	en-SE
en-US	es-419	es-ES	es-CB
es-US	es-AR	es-CL	es-CO
es-MX	es-PE	et-EE	fi-FI
fr-BE	fr-CA	fr-CH	fr-DZ
fr-FR	hi-IN	hr-HR	hu-HU
is-IS	it-IT	id-ID	ja-JP
kk-KZ	ko-KR	lt-LT	lv-LV
mk-MK	ms-MY	nl-BE	nl-NL
no-NO	pl-PL	pt-BR	pt-PT
ro-RO	ru-KZ	ru-RU	sk-SK
sl-SI	sq-AL	sr-RS	sv-SE
th-TH	tr-TR	uk-UA	vi-VN
zh-CN	zh-HK	zh-TW	

languageCodes

Type: Array

A list of BCP-47 language codes representing the supported languages for a resource (e.g., ["en-US", "es-ES", "fr-FR"]).

Concepts

This section defines the key resources and terminology used in the ACXD SDK.

Workspace

A workspace is an isolated environment containing all the resources for a project or team applications, flows, secrets, knowledge bases, and more. Most SDK operations are scoped to a single workspace, specified via the `workspaceId` client configuration.

Application

An application is the top-level conversational AI unit. It contains flows, settings, and deployment configuration. Applications go through a build and deploy lifecycle before they are live.

Flow

A flow defines the conversational logic, the dialog paths, node connections, messages, and behavior your application follows during a conversation. Flows are authored as a graph of typed nodes.

Build

A build compiles an application's flows and configuration into a deployable artifact. Builds are immutable. Once built, the artifact does not change.

Deployment

A deployment publishes a build to a channel, making it live for end users. You can have multiple deployments per application (e.g., production, staging).

Secret

A secret stores a sensitive value (API key, credential, connection string) encrypted at rest. Secrets can be referenced in data requests and integrations without exposing the value.

Context Variable

A context variable is a typed variable available across conversations within a workspace. Use them to pass information between flows or from external systems.

Slot Type

A slot type defines a custom entity for extracting structured data from user input, for example a list of product names or cities with synonyms.

Data Request

A data request is a webhook integration that retrieves or sends data during a conversation. It connects your application to external APIs and services.

Knowledge Base

A knowledge base is a collection of articles or documents that power AI-generated responses. Knowledge bases must be published before they are available in conversations.

Guardrail

A guardrail is a safety rule that monitors conversation content and enforces behavior, like blocking, masking, or rerouting when violations are detected.

Evaluation

An evaluation runs automated quality checks against your application using defined criteria. Evaluations produce scored results and detailed logs.

Simulation

A simulation runs automated conversations against a deployed application to test behavior at scale.

Scenario

A scenario defines a scripted test conversation, a sequence of user inputs and expected behaviors used for automated testing.

Modality

A modality defines the input/output schema for a specific interaction channel (text, voice, custom).

Analytics Tag

An analytics tag labels conversation events as positive, negative, or neutral for reporting and analysis.

Trail

A trail is the audit log of changes made to workspace resources, who changed what, and when.

Programmatic User

A programmatic user is a machine identity that authenticates with the ACXD SDK via API key. Programmatic users are managed by account administrators.

Role

A role defines a set of permissions. Roles can be pre-defined (administrator, developer, content manager, read-only) or custom. Roles are assigned to programmatic users to control what they can access.

Common Errors

This section lists the error types that the ACXD SDK may return.

Common Errors

ValidationException

The input does not meet the required format or constraints. Check that all required parameters are included and that values are valid.

HTTP Status Code: 400

ResourceNotFoundException

The specified resource does not exist. Verify the identifier and that the resource has not been deleted.

HTTP Status Code: 404

ConflictException

The request conflicts with an existing resource. For example, creating a secret with a name that already exists.

HTTP Status Code: 409

AccessDeniedException

The programmatic user does not have permission to perform this action. Check the user's role configuration.

HTTP Status Code: 403

ThrottlingException

The request rate is too high. The SDK automatically retries with exponential backoff.

HTTP Status Code: 429

InternalServerErrorException

An internal error occurred. This is retryable. The SDK automatically retries these requests.

HTTP Status Code: 500

SerializationException

The request body could not be parsed. Verify that the request body is valid JSON.

HTTP Status Code: 400

Error Response Shape

All errors return a consistent JSON body:

```
{
  "type": "ResourceNotFoundException",
  "message": "Secret 'my-secret' not found.",
  "resourceId": "my-secret",
  "resourceType": "Secret"
}
```

A2A protocol contract for external AI agents

This page walks you through building an external AI agent that collaborates with Connect Customer over the A2A protocol. It covers the wire protocol, session lifecycle, voice and chat channels, error handling, and tracing requirements in the order you encounter them when implementing your agent server.

To collaborate with Connect Customer, you must build and set up an AI agent that is compatible with the Connect Customer A2A extension. This developer documentation describes the protocol contract your agent server must implement. The Connect Customer admin setup page links here so that the team building the agent can find these implementation details.

If you are configuring an agent that someone else built rather than implementing one yourself, see [Set up collaboration with an external AI agent](#) for registration and API setup. This page explains the contract at the level of behavior and sequencing, not individual fields.

Contents

- [Prerequisites](#)
- [Transport and discovery](#)
- [Channels](#)
- [Session initiation \(handoff and immediate handoff\)](#)
- [Channel readiness \(CHANNEL_STATE\)](#)
- [Finishing: hand-back and terminal outcomes](#)
- [Errors](#)
- [Interruptions \(barge-in\)](#)
- [DTMF input](#)
- [Silence input](#)
- [Transport fragmentation \(WebSocket only\)](#)
- [Tracing](#)
- [Note on A2A spec compliance](#)
- [Sample events](#)
- [Related topics](#)

Prerequisites

Before your agent server can receive A2A traffic, the Connect Customer instance must have a Lex V2 bot and contact flow configured. These are setup steps described in [Set up collaboration with an external AI agent](#), Step 6. Your agent server does not receive traffic until both are in place.

Transport and discovery

Your agent server communicates with Connect Customer under the following transport rules:

- Every agent must support JSON-RPC frames over WebSockets. The contract targets A2A protocol v1.0.
- On WebSocket, each event is the same JSON-RPC result frame pushed over the socket.
- **Authentication.** Connect Customer sends your API key as `Authorization: Bearer <apiKey>` on the WebSocket upgrade.
- **Extension activation.** Connect Customer sends the header `A2A-Extensions: https://docs.aws.amazon.com/connect/a2a/ext/v1` on requests and upgrades. Echo it back to confirm activation.

Event conventions

The following conventions apply to every A2A event:

- Extension semantics ride on standard A2A objects. The event type is in metadata under the key `https://docs.aws.amazon.com/connect/a2a/ext/v1/eventType`.
- **Trace context.** Connect Customer sends W3C `traceparent` and `tracestate` (`connect=parentSpanId:{full-uuid}`) as headers on each request (WebSocket: on the connection upgrade). Adopt the propagated `traceId` and parent span in the spans you emit (see [the section called "A.10 TRACING_SPAN \(your agent to Connect\)"](#)).

Channels

Connect Customer supports communication through two channels, voice and chat, on the same contact flow.

For the voice channel, a persistent connection to the third-party server stays open over WebSockets.

For the chat channel, a new WebSocket connection opens for each new customer chat message. The connection uses the same `contextId` every time.

Session initiation (handoff and immediate handoff)

Before your agent receives turns in a handoff, Connect Customer sends `INIT_SESSION` (see [the section called “A.1 INIT_SESSION \(Connect to your agent\)”](#)) and expects an `INIT_SESSION_RESPONSE`. You can receive the following fields:

- **instanceArn, contactArn.** The Connect Customer instance and contact this session belongs to.
- **history.** Prior conversation turns as A2A-native Message objects, oldest first, so the customer never re-states their request. On chat, expect a new connection for each turn, with history re-sent on each `INIT_SESSION`. On voice, one persistent connection serves the whole session. Empty for the first handoff when `immediateHandoff` is set to `true`.
- **supportedFinishTypes.** The FINISH types the caller can route on. Emit only FINISH events whose type appears in this list.
- **subscribeToTracingEvents.** When `true`, emit `TRACING_SPAN` events for each turn (see [the section called “Tracing”](#)).
- **audioInputConfiguration / audioOutputConfiguration.** Voice only. Connect Customer proposes a config (LINEAR_PCM, 16-bit, mono, 8000, 16000, or 24000 Hz). Your `INIT_SESSION_RESPONSE` can counter-propose a different rate from the supported set (a counter-proposal is a full re-declaration). Connect Customer resamples between the negotiated config and the caller side. The config is fixed for the session.

Respond with `INIT_SESSION_RESPONSE` (see [the section called “A.2 INIT_SESSION_RESPONSE \(your agent to Connect\)”](#)).

Channel readiness (CHANNEL_STATE)

On a handoff, your agent gates its output on the channel being live. It must not start producing turn output until it has seen a `CHANNEL_STATE` event with state `READY` for the session. `READY` signals that the customer channel is connected and can receive your output.

CHANNEL_STATE is published only on handoff. Delegate does not transfer the channel, so no READY is sent for delegate interactions.

Finishing: hand-back and terminal outcomes

To conclude your agent's participation, emit a FINISH event. You can carry a finish data part on the terminal status update (see [the section called "A.5 Terminal status update with FINISH \(your agent to Connect\)"](#)) or as a message on WebSockets. The types are:

- **COMPLETE.** Request fulfilled. Control returns to the parent agent.
- **ESCALATE.** The conversation needs a human. Connect Customer routes the contact to a human queue (surfaced to the contact flow through the Tool session attribute).

Emit only finish types that appear in the supportedFinishTypes sent on init. After emitting FINISH, your agent must not send further messages on the session. Connect Customer can close the connection immediately.

Errors

Errors fall into these levels, based on where the failure occurs:

- **Connection level.** Failures on the request or WebSocket upgrade surface as native HTTP status codes (for example, authentication failures, or 429 when Connect Customer throttles). No A2A event is involved.
- **Request level.** A request your agent cannot dispatch (malformed frame, unknown method, unknown task) receives a standard JSON-RPC error object. According to A2A v1.0, the error's data carries a `google.rpc.ErrorInfo` object (`{"@type":"type.googleapis.com/google.rpc.ErrorInfo","reason":"...","domain":"a2a-protocol.org"}`). Connect Customer treats a JSON-RPC error from your agent as a failure.
- **Task level.** The primary structured-error path. When a turn starts but cannot complete, emit a terminal statusUpdate with `status.state: TASK_STATE_FAILED`, carrying the structured error as an error data part on `status.message` (see [the section called "A.6 Task failure with structured error \(your agent to Connect\)"](#)). Connect Customer consumes structured errors from this terminal task status update event. Fields: `errorCode` (HTTP-style: 408 timeout, 424 dependency failure, 400 bad request, 500 internal), `errorReason` (human-readable),

`errorCategory` (DEPENDENCY_FAILURE, INTERNAL_ERROR, TIMEOUT, BAD_REQUEST, extensible; treat unknown values as generic failures).

- **Standalone ERROR messages are informational, not terminal.** An ERROR event sent as its own message (see [the section called “A.6b Informational ERROR message \(Connect to your agent\)”](#)) is a diagnostic signal. It does not change task state or end the turn. Connect Customer sends your agent one to explain a problem it detected. For fatal problems (an idle timeout 408, TIMEOUT, or a protocol violation such as conflicting artifacts 400, BAD_REQUEST), the ERROR precedes Connect Customer closing the connection. For non-fatal problems (such as a malformed trace frame 400, BAD_REQUEST), the connection stays open and the conversation continues. Treat the ERROR as context. The authoritative signals remain the terminal statusUpdate and the connection close.
- **Failure compared to error disposition.** If your agent completes its participation but with an error outcome, that is a completion, not a task failure. Emit FINISH with type: COMPLETE_WITH_ERROR on a TASK_STATE_COMPLETED terminal (see [the section called “Finishing: hand-back and terminal outcomes”](#)).
- Connect Customer treats TASK_STATE_FAILED, TASK_STATE_CANCELED, and TASK_STATE_REJECTED from your agent as turn failures.

Interruptions (barge-in)

When your agent is receiving text over the voice channel and the customer speaks over your agent's audio, Connect Customer sends an INTERRUPTION event with type: USER_AUDIO_INPUT (see [the section called “A.7 INTERRUPTION \(either direction\)”](#)). On receipt, stop generating, discard buffered output, and do not send a `lastChunk` for the interrupted artifact. Open a fresh `artifactId` for your next response.

When your agent is receiving customer audio directly, Connect Customer expects you to publish this event so that Connect Customer can flush any buffered audio. Interruption events also arrive if the customer gives a DTMF input with type: DTMF_INPUT. You can consume that or ignore it based on whether your agent consumes DTMF.

DTMF input

Connect Customer accumulates telephone keypad input (end character #, deletion *, inter-digit timeout). It delivers the input to your agent as a single DTMF_INPUT_COMPLETE event that carries the accumulated `inputString` and a `terminationReason` (END_CHARACTER, TIMEOUT, or

MAX_LENGTH). See [the section called “A.8 DTMF_INPUT_COMPLETE \(Connect to your agent\)”](#).

A companion INTERRUPTION (type: DTMF_INPUT) precedes it if output is in flight. Treat the digit string as the turn's input. Connect Customer fixes the accumulation policy at launch, and you cannot change it for individual sessions.

Silence input

Connect Customer detects extended silence and sends input so that an agent can re-prompt the customer. Your agent receives a payload with the text value <EMPTY_USER_INPUT>.

Transport fragmentation (WebSocket only)

Payloads that exceed the WebSocket frame budget split into FRAGMENT events (see [the section called “A.9 FRAGMENT \(WebSocket only, either direction\)”](#)). Buffer by `fragmentId`, reassemble in `partNumber` order after `totalParts` fragments arrive, then base64-decode into the original JSON payload. Your agent must reassemble inbound fragments and can fragment its own oversized frames. Use FRAGMENT only for complete payloads that exceed frame size. Never use it for progressive output streaming (use `artifactUpdate` chunks), never for audio (emit smaller independent audio chunks), and never on HTTPS.

Tracing

We recommend that your agent publish trace spans for observability through TRACING_SPAN events (a `tracingSpan` data part on an `artifactUpdate`). Traces ride as artifacts on the turn's task: emit each trace as an `artifactUpdate` carrying the turn's `taskId`. This is how Connect Customer attributes the trace to the turn. Connect Customer consumes and persists traces for analytics and multi-agent observability. They never reach the end customer. Emit each turn's traces after the response and before the terminal status update. Keep individual span payloads under the frame budget (split oversized trace messages with FRAGMENT).

- **Wire format.** The `tracingSpan` payload wraps a standard OpenTelemetry OTLP/JSON TracesData document under an `otlp` key: `{"otlp": {"resourceSpans": [...]}}`. Generate spans with any OpenTelemetry SDK and serialize the finished spans with the OTLP/JSON encoding. See [the section called “A.10 TRACING_SPAN \(your agent to Connect\)”](#).
- **Session opt-in.** Connect Customer sets `subscribeToTracingEvents: true` on `INIT_SESSION` when it wants traces.

- **Span naming (OpenTelemetry GenAI semantic conventions).** `invoke_agent` `<agentName>` as the root span for the turn (SpanKind SERVER), `chat` `<model>` around each LLM call (CLIENT), `execute_tool` `<toolName>` around each tool call (CLIENT).
- **Required on every span.** `traceId`, `spanId`, `name`, `startTimeUnixNano`, `endTimeUnixNano`. Use a distinct `traceId` for each turn so that you can distinguish turns on persistent (voice) connections. Use the same `taskId` as the turn response.
- **What to include for each turn.** The customer input, the agent's output, and any tool-call requests and results, carried as `aws.connect.span.message` span events with `role` (input, output, or instruction), `type` (for example, TEXT_MESSAGE), `participant`, and `content` attributes. Set `status.code` = ERROR with a message on failure.
- **Recognized attributes.** OTEL GenAI keys (`gen_ai.agent.name`, `gen_ai.operation.name`, `gen_ai.request.model`, `gen_ai.usage.input_tokens`, `gen_ai.usage.output_tokens`, and others) plus Connect Customer public keys (`aws.connect.tool.id`, `aws.connect.tool.type`, `aws.connect.usage.total_tokens`, `aws.connect.error.type`, and others). You can emit additional keys; unrecognized keys have no effect.

Note on A2A spec compliance

Certain types of A2A task states are currently unsupported. These include the following:

- TASK_STATE_UNSPECIFIED
- TASK_STATE_SUBMITTED
- TASK_STATE_CANCELED
- TASK_STATE_INPUT_REQUIRED
- TASK_STATE_REJECTED
- TASK_STATE_AUTH_REQUIRED

Sample events

`<EXT>` = `https://docs.aws.amazon.com/connect/a2a/ext/v1`. The contract targets A2A protocol v1.0 (JSON-RPC binding). All samples show the full JSON-RPC framing. Requests from Connect Customer are JSON-RPC requests (method and params). Your agent's direct RPC response must echo the request's `id`. Events your agent streams back ride in a JSON-RPC result envelope

whose value is exactly one of `message`, `task`, `statusUpdate`, or `artifactUpdate`. Connect Customer identifies each event by which of those keys is present (plus the `eventType` metadata) and correlates it by `contextId` and `taskId`.

A.1 INIT_SESSION (Connect to your agent)

The audio config is negotiated only when the third-party agent is voice enabled.

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "1",
  "params": {
    "message": {
      "messageId": "m-1",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "data": {
          "initSession": {
            "instanceArn": "<CONNECT_INSTANCE_ARN>",
            "contactArn": "<CONNECT_CONTACT_ARN>",
            "supportedFinishTypes": ["COMPLETE", "COMPLETE_WITH_ERROR",
"ESCALATE"],
            "subscribeToTracingEvents": true,
            "audioInputConfiguration": {
              "encoding": "LINEAR_PCM",
              "sampleRateHertz": 8000,
              "sampleSizeBits": 16,
              "channelCount": 1
            },
            "audioOutputConfiguration": {
              "encoding": "LINEAR_PCM",
              "sampleRateHertz": 16000,
              "sampleSizeBits": 16,
              "channelCount": 1
            },
            "history": [{
              "messageId": "h-1",
              "role": "ROLE_USER",
              "parts": [{
                "text": "I need help with my bill",
                "mediaType": "text/plain"
```

```

        ]]
      }, {
        "messageId": "h-2",
        "role": "ROLE_AGENT",
        "parts": [{
          "text": "I can help with that.",
          "mediaType": "text/plain"
        }]
      }]
    },
    "mediaType": "application/json"
  ]],
  "metadata": {
    "<EXT>/eventType": "INIT_SESSION"
  }
}
}
}

```

Text-only sessions omit the audio configurations. The history is re-sent on every chat reconnection.

A.2 INIT_SESSION_RESPONSE (your agent to Connect)

Your agent sends this in reply to INIT_SESSION.

```

{
  "jsonrpc": "2.0",
  "id": "1",
  "result": {
    "message": {
      "messageId": "m-2",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_AGENT",
      "parts": [{
        "data": {
          "initSessionResponse": {
            "statusCode": "200",
            "audioOutputConfiguration": {
              "encoding": "LINEAR_PCM",
              "sampleRateHertz": 24000,
              "sampleSizeBits": 16,

```

```

        "channelCount": 1
      }
    }
  },
  "mediaType": "application/json"
}],
"extensions": ["<EXT>"],
"metadata": {
  "<EXT>/eventType": "INIT_SESSION_RESPONSE"
}
}
}
}

```

The audio override is present only when counter-proposing. Omit it to accept the config that Connect Customer proposed.

A.2b CHANNEL_STATE READY (Connect to your agent, orchestrated/tool handoff)

On a handoff, your agent treats receipt of READY as the signal that the channel is live and it can begin emitting output.

```

{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "cs-1",
  "params": {
    "message": {
      "messageId": "m-cs-1",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "data": {
          "channelState": {
            "state": "READY"
          }
        },
        "mediaType": "application/json"
      }],
      "metadata": {
        "<EXT>/eventType": "CHANNEL_STATE"
      }
    }
  }
}

```

```

    }
  }
}

```

A.3 Text turn input (Connect to your agent)

After initialization, Connect Customer sends each customer message as a text turn.

```

{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "req-1",
  "params": {
    "message": {
      "messageId": "m-3",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "text": "Why is my bill higher this month?",
        "mediaType": "text/plain"
      }],
      "metadata": {}
    }
  }
}

```

Voice sessions instead stream audio parts (see [the section called “A.3b Audio input chunk \(Connect to your agent, voice, audio enabled 3P agent\)”](#)).

A.3b Audio input chunk (Connect to your agent, voice, audio enabled 3P agent)

Customer audio streams continuously as SendMessage requests, roughly one small PCM slice for each frame, each with its own JSON-RPC request id, at the negotiated audioInputConfiguration. There is no per-turn boundary marker on input audio. Your agent detects turn-taking from the audio itself (voice activity detection). Your spoken response does not correlate to any audio frame's request id: it streams back as artifactUpdate events on the turn's task (see [the section called “A.4b Audio response chunk \(your agent to Connect, voice, audio enabled 3P agent\)”](#)).

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "req-audio-1",
  "params": {
    "message": {
      "messageId": "m-4",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "raw": "<BASE64_PCM_CHUNK>",
        "mediaType": "audio/lpcm"
      }],
      "metadata": {}
    }
  }
}
```

A.4 Text response chunk (your agent to Connect)

Your agent streams text back as artifactUpdate events.

```
{
  "jsonrpc": "2.0",
  "id": "c7d03f92",
  "result": {
    "artifactUpdate": {
      "taskId": "task-1",
      "contextId": "<SESSION_ID>",
      "artifact": {
        "artifactId": "resp-1",
        "parts": [{
          "text": "Your bill increased because ",
          "mediaType": "text/plain"
        }],
        "metadata": {
          "<EXT>/eventType": "TEXT_RESPONSE_CHUNK"
        }
      }
    },
    "append": false,
    "lastChunk": false
  }
}
```

```
}
```

Continuations set `append: true`; the final chunk sets `lastChunk: true`. For audio output, see [the section called “A.4b Audio response chunk \(your agent to Connect, voice, audio enabled 3P agent\)”](#).

A.4b Audio response chunk (your agent to Connect, voice, audio enabled 3P agent)

Spoken output streams as `artifactUpdate` events tagged `AUDIO_RESPONSE_CHUNK`, carrying base64-encoded PCM at the negotiated `audioOutputConfiguration`. All chunks of one spoken response share an `artifactId`; the first frame sets `append: false`, continuations set `append: true`, and the final frame sets `lastChunk: true`:

```
{
  "jsonrpc": "2.0",
  "id": "e15a8d43",
  "result": {
    "artifactUpdate": {
      "taskId": "task-1",
      "contextId": "<SESSION_ID>",
      "artifact": {
        "artifactId": "resp-1-audio",
        "parts": [{
          "raw": "<BASE64_PCM_CHUNK>",
          "mediaType": "audio/lpcm"
        }],
        "metadata": {
          "<EXT>/eventType": "AUDIO_RESPONSE_CHUNK"
        }
      },
      "append": false,
      "lastChunk": false
    }
  }
}
```

Each audio chunk must be independently playable: keep every frame under the transport frame budget rather than relying on `FRAGMENT`. You cannot play fragmented audio progressively. Emit audio at the negotiated output config; Connect Customer resamples toward the caller. If the

customer barges in, stop the stream without sending `lastChunk` and open a fresh `artifactId` for the next response (see [the section called “A.7 INTERRUPTION \(either direction\)”](#)).

A.5 Terminal status update with FINISH (your agent to Connect)

Your agent sends this as the last frame of the turn.

```
{
  "jsonrpc": "2.0",
  "id": "2f6b9c18",
  "result": {
    "statusUpdate": {
      "taskId": "task-1",
      "contextId": "<SESSION_ID>",
      "status": {
        "state": "TASK_STATE_COMPLETED",
        "timestamp": "2026-08-08T16:40:00.000Z",
        "message": {
          "messageId": "m-9",
          "role": "ROLE_AGENT",
          "parts": [{
            "data": {
              "finish": {
                "type": "ESCALATE",
                "reason": "Customer requested a human agent"
              }
            },
            "mediaType": "application/json"
          ]},
          "metadata": {
            "<EXT>/eventType": "FINISH"
          }
        }
      }
    }
  }
}
```

A normal mid-conversation turn ends with the same `statusUpdate` without the finish part.

A.6 Task failure with structured error (your agent to Connect)

The terminal statusUpdate for a failed turn. The structured error rides as an error data part on status.message, tagged with the ERROR eventType:

```
{
  "jsonrpc": "2.0",
  "id": "a3c47e05",
  "result": {
    "statusUpdate": {
      "taskId": "task-1",
      "contextId": "<SESSION_ID>",
      "status": {
        "state": "TASK_STATE_FAILED",
        "message": {
          "messageId": "m-10",
          "role": "ROLE_AGENT",
          "parts": [{
            "data": {
              "error": {
                "errorCode": 424,
                "errorReason": "Order lookup service timed out",
                "errorCategory": "DEPENDENCY_FAILURE"
              }
            }
          ]
        },
        "mediaType": "application/json"
      }
    },
    "metadata": {
      "<EXT>/eventType": "ERROR"
    }
  }
}
```

A.6b Informational ERROR message (Connect to your agent)

Sent as a standalone message to explain a detected problem. Not a turn boundary. For fatal problems (idle timeout, artifact conflict) it precedes a connection close; for non-fatal problems (for example, a malformed trace) the conversation continues:

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "d90e5b72",
  "params": {
    "message": {
      "messageId": "m-11",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "data": {
          "error": {
            "errorCode": 408,
            "errorReason": "No frames received within the idle window; closing
the connection",
            "errorCategory": "TIMEOUT"
          }
        },
        "mediaType": "application/json"
      }],
      "extensions": [<EXT>],
      "metadata": {
        "<EXT>/eventType": "ERROR"
      }
    }
  }
}
```

A.7 INTERRUPTION (either direction)

When Connect Customer detects the interruption (for example, the customer speaks or presses a key while your agent's output is playing), it arrives as a `SendMessage` request:

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "req-3",
  "params": {
    "message": {
      "messageId": "m-11",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
```

```

    "parts": [{
      "data": {
        "interruption": {
          "type": "USER_AUDIO_INPUT",
          "reason": "user_speech"
        }
      },
      "mediaType": "application/json"
    }],
    "metadata": {
      "<EXT>/eventType": "INTERRUPTION"
    }
  }
}

```

When your agent detects the interruption (audio mode, your agent's own barge-in detection), emit the same event as a streamed frame so Connect Customer flushes buffered playback:

```

{
  "jsonrpc": "2.0",
  "id": "8f2d1c07",
  "result": {
    "message": {
      "messageId": "m-11a",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_AGENT",
      "parts": [{
        "data": {
          "interruption": {
            "type": "USER_AUDIO_INPUT",
            "reason": "user_speech"
          }
        },
        "mediaType": "application/json"
      }],
      "extensions": ["<EXT>"],
      "metadata": {
        "<EXT>/eventType": "INTERRUPTION"
      }
    }
  }
}

```

The type is `USER_AUDIO_INPUT` for voice barge-in, `DTMF_INPUT` for keypad, `SYSTEM` otherwise.

A.8 DTMF_INPUT_COMPLETE (Connect to your agent)

Connect Customer delivers the accumulated keypad digits as a single event:

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "req-2",
  "params": {
    "message": {
      "messageId": "m-12",
      "contextId": "<SESSION_ID>",
      "role": "ROLE_USER",
      "parts": [{
        "data": {
          "dtmfInputComplete": {
            "inputString": "1234",
            "terminationReason": "END_CHARACTER"
          }
        },
        "mediaType": "application/json"
      }],
      "metadata": {
        "<EXT>/eventType": "DTMF_INPUT_COMPLETE"
      }
    }
  }
}
```

A.9 FRAGMENT (WebSocket only, either direction)

Connect Customer to third-party agent:

```
{
  "jsonrpc": "2.0",
  "method": "SendMessage",
  "id": "6a1f3d84",
  "params": {
    "message": {
      "messageId": "m-13",
```

```

    "role": "ROLE_AGENT",
    "parts": [{
      "data": {
        "fragment": {
          "fragmentId": "f-1",
          "partNumber": 0,
          "totalParts": 3,
          "payload": "<BASE64_CHUNK>",
          "encoding": "base64"
        }
      },
      "mediaType": "application/json"
    }],
    "metadata": {
      "<EXT>/eventType": "FRAGMENT"
    }
  }
}

```

Third-party agent to Connect Customer:

```

{
  "jsonrpc": "2.0",
  "id": "6a1f3d84",
  "result": {
    "message": {
      "messageId": "m-13",
      "role": "ROLE_AGENT",
      "parts": [{
        "data": {
          "fragment": {
            "fragmentId": "f-1",
            "partNumber": 0,
            "totalParts": 3,
            "payload": "<BASE64_CHUNK>",
            "encoding": "base64"
          }
        },
        "mediaType": "application/json"
      }],
      "metadata": {
        "<EXT>/eventType": "FRAGMENT"
      }
    }
  }
}

```

```

    }
  }
}
}

```

Concatenate payload values in `partNumber` order and base64-decode to recover the original JSON payload. A `FRAGMENT` message contains exactly one data part and no other content parts.

A.10 TRACING_SPAN (your agent to Connect)

The following example shows a trace with a single root span:

```

{
  "jsonrpc": "2.0",
  "id": "b8c2e690",
  "result": {
    "artifactUpdate": {
      "taskId": "task-1",
      "contextId": "<SESSION_ID>",
      "artifact": {
        "artifactId": "trace-1",
        "parts": [{
          "data": {
            "tracingSpan": {
              "otlp": {
                "resourceSpans": [{
                  "resource": {
                    "attributes": [
                      { "key": "service.name", "value":
{ "stringValue": "billing-agent" } }
                    ]
                  },
                  "scopeSpans": [{
                    "scope": { "name": "amazon.connect.ai-agent",
"version": "1.0.0" },
                    "spans": [{
                      "traceId": "550e8400e29b41d4a716446655440000",
                      "spanId": "eee19b7ec3c1b173",
                      "parentSpanId": "6ba7b8109dad11d1",
                      "name": "invoke_agent billing-agent",
                      "kind": 2,
                      "startTimeUnixNano": "1715000000000000000",
                      "endTimeUnixNano": "1715000002500000000",

```

```

        "attributes": [
            { "key": "gen_ai.agent.name", "value":
{ "stringValue": "billing-agent" } },
            { "key": "gen_ai.operation.name", "value":
{ "stringValue": "invoke_agent" } },
            { "key": "gen_ai.usage.input_tokens",
"value": { "intValue": "1200" } },
            { "key": "gen_ai.usage.output_tokens",
"value": { "intValue": "340" } }
        ],
        "events": [
            {
                "name": "aws.connect.span.message",
                "timeUnixNano": "1715000000000000000",
                "attributes": [
                    { "key":
"aws.connect.span.message.role", "value": { "stringValue": "input" } },
                    { "key":
"aws.connect.span.message.type", "value": { "stringValue": "TEXT_MESSAGE" } },
                    { "key":
"aws.connect.span.message.participant", "value": { "stringValue": "user" } },
                    { "key":
"aws.connect.span.message.content", "value": { "stringValue": "What is my
balance?" } }
                ]
            },
            {
                "name": "aws.connect.span.message",
                "timeUnixNano": "1715000002500000000",
                "attributes": [
                    { "key":
"aws.connect.span.message.role", "value": { "stringValue": "output" } },
                    { "key":
"aws.connect.span.message.type", "value": { "stringValue": "TEXT_MESSAGE" } },
                    { "key":
"aws.connect.span.message.participant", "value": { "stringValue": "assistant" } },
                    { "key":
"aws.connect.span.message.content", "value": { "stringValue": "Your current balance is
$2,847.53." } }
                ]
            }
        ],
        "status": { "code": 1 }
    ]
}

```

```

    }
  }
}
},
"mediaType": "application/json"
}],
"metadata": {
  "<EXT>/eventType": "TRACING_SPAN"
}
},
"append": false,
"lastChunk": true
}
}
}

```

The payload under `otlp` is a standard OpenTelemetry OTLP/JSON TracesData document. Generate spans with any OpenTelemetry SDK and serialize finished spans with the OTLP/JSON encoding. The trace rides as an artifact on the turn's task: the `artifactUpdate` carries the turn's `taskId`, which is how Connect Customer attributes the trace to the turn. The trace adopts the propagated context from Connect Customer: `traceId` comes from the request's `traceparent` header, and the root span's `parentSpanId` is the propagated parent span. The root `invoke_agent` span carries the turn's input and output as `aws.connect.span.message` events; add child `chat` and `execute_tool` spans for LLM and tool calls. Emit traces after the turn's response and before the terminal status update.

A.10b TRACING_SPAN with per-segment latency (multiple spans in one trace)

A turn can emit multiple output segments (for example, one sentence at a time). To record the latency of each segment, put one span for each segment inside the same `otlp` envelope, not multiple `TRACING_SPAN` frames. Connect Customer consumes the first `TRACING_SPAN` artifact on the turn and reads every span inside its `otlp.resourceSpans` batch, so all segment spans must ride in that single envelope. Each segment span carries its own `startTimeUnixNano/endTimeUnixNano` (segment latency = end minus start) and parents to the turn's `invoke_agent` root; give every span the turn's single `traceId`.

```

{
  "jsonrpc": "2.0",

```

```

    "id": "b8c2e691",
    "result": {
      "artifactUpdate": {
        "taskId": "task-1",
        "contextId": "<SESSION_ID>",
        "artifact": {
          "artifactId": "trace-1",
          "parts": [{
            "data": {
              "tracingSpan": {
                "otlp": {
                  "resourceSpans": [{
                    "resource": {
                      "attributes": [
                        { "key": "service.name", "value":
{ "stringValue": "billing-agent" } }
                      ]
                    },
                    "scopeSpans": [{
                      "scope": { "name": "amazon.connect.ai-agent",
"version": "1.0.0" },
                      "spans": [
                        {
                          "traceId":
"550e8400e29b41d4a716446655440000",
                          "spanId": "aaaa000000000001",
                          "parentSpanId": "6ba7b8109dad11d1",
                          "name": "invoke_agent billing-agent",
                          "kind": 2,
                          "startTimeUnixNano": "1715000000000000000",
                          "endTimeUnixNano": "1715000001310000000",
                          "attributes": [
                            { "key": "gen_ai.agent.name", "value":
{ "stringValue": "billing-agent" } },
                            { "key": "gen_ai.operation.name",
"value": { "stringValue": "invoke_agent" } }
                          ],
                          "status": { "code": 1 }
                        },
                        {
                          "traceId":
"550e8400e29b41d4a716446655440000",
                          "spanId": "bbbb000000000001",
                          "parentSpanId": "aaaa000000000001",

```

```

"value": { "stringValue": "chat" } }

    "name": "chat sentence-1",
    "kind": 3,
    "startTimeUnixNano": "1715000000000000000",
    "endTimeUnixNano": "1715000000420000000",
    "attributes": [
        { "key": "gen_ai.operation.name",

"aws.connect.span.message.role", "value": { "stringValue": "output" } },
        { "key":
"aws.connect.span.message.type", "value": { "stringValue": "TEXT_MESSAGE" } },
        { "key":
"aws.connect.span.message.participant", "value": { "stringValue": "assistant" } },
        { "key":
"aws.connect.span.message.content", "value": { "stringValue": "Your bill increased by
$40 this month." } }

    ]
    }],
    "status": { "code": 1 }
},
{
    "traceId":
"550e8400e29b41d4a716446655440000",

    "spanId": "bbbb0000000000002",
    "parentSpanId": "aaaa0000000000001",
    "name": "chat sentence-2",
    "kind": 3,
    "startTimeUnixNano": "1715000000420000000",
    "endTimeUnixNano": "1715000000905000000",
    "attributes": [
        { "key": "gen_ai.operation.name",

"value": { "stringValue": "chat" } }

    ],
    "events": [{
        "name": "aws.connect.span.message",
        "timeUnixNano": "1715000000905000000",
        "attributes": [
            { "key":
"aws.connect.span.message.role", "value": { "stringValue": "output" } },

```

```

        { "key":
"aws.connect.span.message.type", "value": { "stringValue": "TEXT_MESSAGE" } },
        { "key":
"aws.connect.span.message.participant", "value": { "stringValue": "assistant" } },
        { "key":
"aws.connect.span.message.content", "value": { "stringValue": "The increase is from
your streaming add-on." } }
    ]
  }],
  "status": { "code": 1 }
},
{
  "traceId":
"550e8400e29b41d4a716446655440000",
  "spanId": "bbbb0000000000003",
  "parentSpanId": "aaaa0000000000001",
  "name": "chat sentence-3",
  "kind": 3,
  "startUnixNano": "1715000000905000000",
  "endUnixNano": "1715000001310000000",
  "attributes": [
    { "key": "gen_ai.operation.name",
"value": { "stringValue": "chat" } }
  ],
  "events": [{
    "name": "aws.connect.span.message",
    "timeUnixNano": "1715000001310000000",
    "attributes": [
      { "key":
"aws.connect.span.message.role", "value": { "stringValue": "output" } },
      { "key":
"aws.connect.span.message.type", "value": { "stringValue": "TEXT_MESSAGE" } },
      { "key":
"aws.connect.span.message.participant", "value": { "stringValue": "assistant" } },
      { "key":
"aws.connect.span.message.content", "value": { "stringValue": "You can remove it
anytime in settings." } }
    ]
  }],
  "status": { "code": 1 }
}
]
}]
}]

```

```
        }
      },
      "mediaType": "application/json"
    ]],
    "metadata": {
      "<EXT>/eventType": "TRACING_SPAN"
    }
  },
  "append": false,
  "lastChunk": true
}
}
```

Three chat segment spans nest under the one `invoke_agent` root; their start and end deltas give per-segment latencies of 420 ms, 485 ms, and 405 ms. All spans share one `traceId` and ride in a single `TRACING_SPAN` artifact. A second `TRACING_SPAN` frame on the same turn is not read, so batch every segment span into this one envelope.

Related topics

- [Observability for collaborating AI agents](#)
- [Trace data requirements for external AI agents](#)
- [Configure voice for collaborating AI agents](#)

Touchpoint

Touchpoint is a drop-in conversational UI and SDK for Amazon Connect Customer — a single widget for chat, voice, and voice-mini, plus **Live Sync**, the native technology that keeps a voice or chat conversation in sync with your on-screen interface.

Install the package from npm:

```
npm i @amazon-connect-touchpoint/web
```

Important

Touchpoint is open source on GitHub at <https://github.com/amazon-connect/touchpoint>. The full SDK reference — every configuration option, type, and modality component — is generated from source and lives in the repository's [docs/](#) directory.

Contents

- [What Touchpoint provides](#)
- [Prerequisites](#)
- [Examples](#)
- [Interactive playground](#)
- [API reference](#)
- [License](#)

What Touchpoint provides

Touchpoint renders a single widget that you configure through the `input` option. Choose the input mode that matches the experience you want:

- `text` — a chat experience. Connects through your `StartChatContact` endpoint (`config.chatEndpoint`).
- `voice` — the full-screen voice experience. Connects through your `StartWebRTCContact` endpoint (`config.voiceEndpoint`).

- `voiceMini` — a compact, floating voice widget. Also connects through your `StartWebRTCContact` endpoint.
- `external` — no rendered UI. Touchpoint opens no chat or voice contact of its own; it only connects Live Sync to an existing live contact so that contact can drive your digital asset in real time.

Live Sync is native Amazon Connect Customer technology that synchronizes a conversation with a digital, on-screen interface. Live Sync is the only feature that requires an Agentic CX Designer (ACXD) application; plain chat and voice work without it. For more information about ACXD, see the [Amazon CX Designer SDK](#) chapter in this guide and [Agentic CX designer \(ACXD\)](#) in the *Amazon Connect Administrator Guide*.

Prerequisites

To run Touchpoint you need an Amazon Connect Customer instance with a contact flow, plus a browser-facing endpoint that mints the contact credentials. Gather the following before you call `create()`:

What you need	Where it comes from
StartChatContact endpoint (chat)	An API Gateway/Lambda route that calls Amazon Connect's <code>StartChatContact</code> and returns participant credentials. Deploy the StartChatContact API . Required for input: <code>"text"</code> . Passed as <code>config.chatEndpoint</code> .
StartWebRTCContact endpoint (voice)	An endpoint that calls <code>StartWebRTCContact</code> and returns the Chime connection data. Deploy the StartWebRTCContact sample . Required for input: <code>"voice"</code> and input: <code>"voiceMini"</code> . Passed as <code>config.voiceEndpoint</code> .
Instance ID	The Amazon Connect Customer instance (UUID) the contact is created in. <code>config.instanceId</code> .
Contact Flow ID	The contact flow (UUID) that handles the contact. <code>config.contactFlowId</code> .

What you need	Where it comes from
Region	AWS region of your instance, for example <code>us-west-2</code> . <code>config.region</code> .

That's everything you need for chat and voice. **Live Sync is optional** — it additionally requires an Agentic CX designer (ACXD) application; see the following procedures.

Setting up Amazon Connect Customer

1. **Create a contact flow** in your Amazon Connect Customer instance that handles the contact; note the **instance ID** and **contact flow ID**.
2. **Stand up the browser endpoints** so the page can create a contact — the [StartChatContact](#) (chat) and [StartWebRTCContact](#) (voice) endpoints listed in the preceding table.

Adding Live Sync (optional)

Live Sync keeps the conversation in sync with your on-screen interface. It's the only feature that requires ACXD; plain chat and voice work without it. To enable it:

1. **Build the ACXD application.** In the ACXD Canvas, wire `Start` → a **Live Sync** node → `Exit` application. On the Live Sync node, declare the actions and scopes the assistant may use (or define them on the fly with this SDK for rapid prototyping — see [the section called “Live Sync”](#)).
2. **Publish it and copy your keys.** Deploy the application, then copy the **deployment key** and **API key** from its settings.
3. **Route your contact flow into it** so contacts reach the ACXD application, and pass the keys as `liveSync.deploymentKey` / `liveSync.apiKey`.

The [the section called “Interactive playground”](#) lets you plug these values into a form and launch a live instance without writing any code.

Examples

The following examples show how to configure Touchpoint for each input mode.

Chat

Configure a chat experience with input: "text".

```
import { create } from "@amazon-connect-touchpoint/web";

const touchpoint = await create({
  config: {
    // Your StartChatContact endpoint (e.g. an API Gateway route) that mints a
    // participant token.
    chatEndpoint: "REPLACE_WITH_START_CHAT_ENDPOINT",
    instanceId: "REPLACE_WITH_INSTANCE_ID",
    contactFlowId: "REPLACE_WITH_CONTACT_FLOW_ID",
    region: "us-east-1",
  },
  input: "text",
});
```

Voice mini

A compact, floating voice widget. Voice modes connect through your StartWebRTCContact endpoint.

```
import { create } from "@amazon-connect-touchpoint/web";

const touchpoint = await create({
  config: {
    voiceEndpoint: "REPLACE_WITH_START_WEBRTC_ENDPOINT",
    instanceId: "REPLACE_WITH_INSTANCE_ID",
    contactFlowId: "REPLACE_WITH_CONTACT_FLOW_ID",
    region: "us-east-1",
  },
  input: "voiceMini", // or "voice" for the full-screen experience
});
```

Live Sync

Live Sync is native Amazon Connect Customer technology that synchronizes a voice (or chat) conversation with a digital, on-screen interface. Customers see options, next steps, and confirmations in real time while an AI agent guides them — no app, no channel switch — unlocking

complex, multi-step tasks that voice or chat alone can't handle, where customers need to see, choose, and complete actions in a single continuous conversation.

In Touchpoint, the AI agent drives the page through **custom actions**: you register a named action with a JSON Schema, the agent resolves what the customer said into typed arguments, and your handler applies them to the page.

```
import { create } from "@amazon-connect-touchpoint/web";

const touchpoint = await create({
  config: {
    chatEndpoint: "REPLACE_WITH_START_CHAT_ENDPOINT",
    instanceId: "REPLACE_WITH_INSTANCE_ID",
    contactFlowId: "REPLACE_WITH_CONTACT_FLOW_ID",
    region: "us-east-1",
  },
  input: "text",
  liveSync: {
    deploymentKey: "REPLACE_WITH_DEPLOYMENT_KEY",
    apiKey: "REPLACE_WITH_API_KEY",
    // `contactId` is optional – defaults to the active session's contact. Set it
    // to bind a separate contact (e.g. an inbound phone call) to this page.
  },
});

// Advertise the actions the AI agent may take (sent once a contact exists):
await touchpoint.sendContext({
  actions: [
    {
      action: "set_cabin_class",
      description: "Choose the cabin class",
      schema: {
        type: "object",
        properties: {
          cabin: { type: "string", enum: ["economy", "business", "first"] },
        },
        required: ["cabin"],
      },
      handler: ({ cabin }) => setCabinClass(cabin),
    },
  ],
});
```

Actions can also be configured in [Agentic CX designer \(ACXD\)](#) on the Live Sync node; defining them here is useful for rapid prototyping. To notify a Live Sync script that the user reached a step, call `touchpoint.sendStep({ stepId, scriptId, apiKey, context })`.

External (no UI)

Use input: "external" to run Touchpoint with **no rendered UI**. It opens no chat or voice contact of its own — it only connects Live Sync to an existing live contact (identified by `liveSync.contactId`) so that contact can drive your digital asset in real time. A common case is a **phone call**: the caller talks to the agent while the page updates alongside them. No `chatEndpoint` or `voiceEndpoint` is needed; only the Live Sync credentials.

```
import { create } from "@amazon-connect-touchpoint/web";

const touchpoint = await create({
  config: { region: "us-east-1" },
  input: "external",
  liveSync: {
    deploymentKey: "REPLACE_WITH_DEPLOYMENT_KEY",
    apiKey: "REPLACE_WITH_API_KEY",
    contactId: "REPLACE_WITH_CONTACT_ID", // the contact to sync with
  },
});

// Advertise your actions, then drive the page as the agent invokes them.
await touchpoint.sendContext({ actions: [/ * ... */ ] });
```

Interactive playground

Running the project locally launches an interactive playground covering every capability (chat, voice, voice-mini, and Live Sync):

```
npm install
npm run dev
# then open http://localhost:5173
```

API reference

The full SDK specification — every configuration option, type, and modality component — is generated from source and lives in the repository's [docs/](#) directory.

Common entry points include:

- `create(config)` — create and mount a Touchpoint instance.
- `TouchpointConfiguration` — top-level options (`config`, `input`, `colorMode`, `theme`, `liveSync`, and more).
- `ConnectConfig` — Amazon Connect Customer connection details.
- `LiveSyncConnection` / `LiveSyncContextInput` — Live Sync connection and the actions, scopes, and destinations you advertise to the AI agent.

License

Touchpoint is released under the [MIT](#) license.

Document history for the Amazon Connect Developer Guide

The following table describes the documentation releases for Amazon Connect.

Change	Description	Date
Initial release	Initial release of the Amazon Connect Developer Guide. This guide provides developer-focused content including API best practices, the Connect flow language, rules function language, and testing language.	May 15, 2026