



# Amazon Connect Integration

WEEK 5 PROJECT – HOLIDAY ROUTING



PROPRIETARY & CONFIDENTIAL

1095 Morris Ave, SUITE 101, Union NJ 07083  
646-762-0305 | [INFO@QUINTRIXSOLUTIONS.COM](mailto:INFO@QUINTRIXSOLUTIONS.COM)

# Amazon Connect – Routing

There are times when contact centers need to play messages like holiday, inclement weather, and emergency notifications to callers. For example, in the event of a fire alarm in a contact center, agents would have to evacuate the building immediately and would not be able to answer calls. As a result, the call center supervisor might invoke an emergency message to the callers.

In the case of such events, you may want to set up messages quickly and without changing your Contact Flows. This training will take you through the process of setting up your Amazon Connect instance to invoke message to inbound caller dynamically.

## **You will use the AWS Management Console to:**

1. Create a user interface to set up a holiday message, using an AWS CloudFormation template.
2. Add a holiday message In the user interface and store it in an Amazon DynamoDB table.
3. Set up a Contact Flow in Amazon Connect and test the holiday message created using Amazon Polly.

# Amazon Connect – Routing

## Learning objectives

After completing this training, you will be able to:

- Use Amazon CloudFormation to build a website hosted on Amazon S3.
- Use Amazon Cognito to authenticate a login attempt to the website.
- Create and store emergency and holiday messages using Amazon DynamoDB.
- Create an Amazon Connect Contact Flow and test the holiday message playback using text to speech via Amazon Polly.

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

In this section you will create a web page based user interface to set up holiday and emergency messages. To do this you will use an Amazon CloudFormation template. The template will build the user interface and then you will use Amazon DynamoDB to store the text representation of the prompts.

The Amazon CloudFormation template will create these resources:

1. Amazon DynamoDB table
2. Amazon S3 bucket to store the user interface web contents.
3. Two AWS Lambda Functions.
4. Amazon Cognito User Pool and an Identity Pool

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

Use the corresponding YAML file as a template within CloudFormation

[https://drive.google.com/file/d/1OXrEeMuuqFdc2YftbjgWUe\\_PgCxfGKTu/view?usp=sharing](https://drive.google.com/file/d/1OXrEeMuuqFdc2YftbjgWUe_PgCxfGKTu/view?usp=sharing)

# Amazon Connect – Routing

**Build the User Interface to Set Dynamic Messages**

**Build the User Interface**

**Amazon CloudFormation**

Login to the AWS Management Console and select the **Amazon CloudFormation** service.

Choose your region.

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Create stack

Select the Create stack button and choose the With new resources (standard) option from the dropdown list.

Under Template Source, select Upload a template file.

Click the Choose file button and upload the callcenterhop.yaml file.

Scroll down and click Next.

The screenshot shows the 'Create stack' wizard in the Amazon Connect console. The breadcrumb navigation at the top reads 'Stacks > Create stack'. The main heading is 'Create stack'. Below this, there are two main sections: 'Prerequisite - Prepare template' and 'Specify template'. In the 'Prerequisite' section, there are three radio button options: 'Template is ready' (which is selected), 'Use a sample template', and 'Create template in Designer'. The 'Specify template' section has a sub-heading 'Template source' with the instruction 'Selecting a template generates an Amazon S3 URL where it will be stored.' It contains two radio button options: 'Amazon S3 URL' and 'Upload a template file' (which is selected). Below this, there is a section 'Upload a template file' with a 'Choose file' button and a text input field containing 'callcenterhop.yaml'. Below the input field, it says 'JSON or YAML formatted file'. At the bottom of the 'Specify template' section, it shows the 'S3 URL: https://s3-2822vr-callcenterhop.yaml' and a 'View in Designer' button. At the very bottom of the wizard, there are 'Cancel' and 'Next' buttons.

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Specify stack details

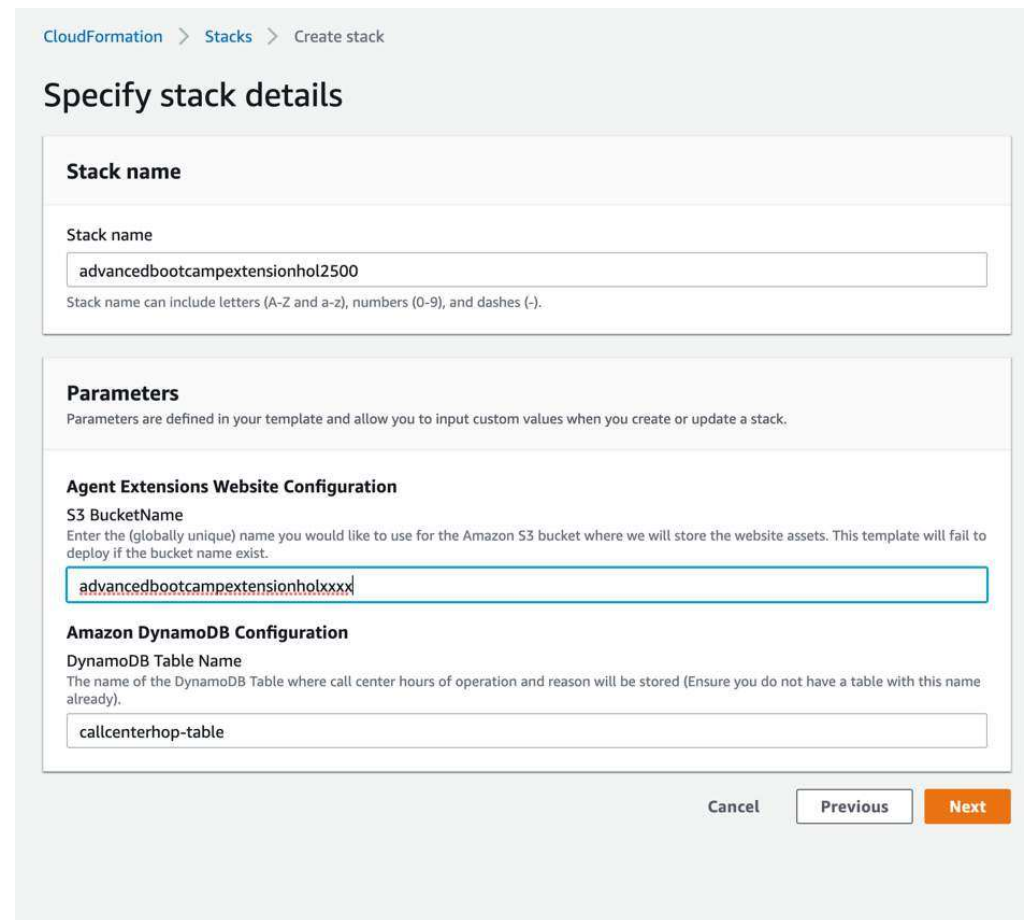
Give both the stack and the Amazon S3 bucket a unique name.

We will use Amazon S3 to host the web user interface.

Leave the Amazon DynamoDB table name as callcenterhop-table.

We will store the emergency and holiday messages and the date / time when they are meant to be played in this table.

Click Next.



The screenshot shows the 'Specify stack details' page in the AWS CloudFormation console. The breadcrumb trail at the top is 'CloudFormation > Stacks > Create stack'. The page title is 'Specify stack details'. There are three main sections: 'Stack name', 'Parameters', and 'Agent Extensions Website Configuration'. The 'Stack name' section has a text input field with the value 'advancedbootcampextensionhol2500' and a note: 'Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-)'. The 'Parameters' section has a note: 'Parameters are defined in your template and allow you to input custom values when you create or update a stack.' The 'Agent Extensions Website Configuration' section has two sub-sections: 'S3 BucketName' with a note 'Enter the (globally unique) name you would like to use for the Amazon S3 bucket where we will store the website assets. This template will fail to deploy if the bucket name exist.' and a text input field with the value 'advancedbootcampextensionholxxxx'; and 'Amazon DynamoDB Configuration' with a note 'The name of the DynamoDB Table where call center hours of operation and reason will be stored (Ensure you do not have a table with this name already).' and a text input field with the value 'callcenterhop-table'. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Next'.

CloudFormation > Stacks > Create stack

### Specify stack details

**Stack name**

Stack name

advancedbootcampextensionhol2500

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

**Parameters**

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

**Agent Extensions Website Configuration**

**S3 BucketName**

Enter the (globally unique) name you would like to use for the Amazon S3 bucket where we will store the website assets. This template will fail to deploy if the bucket name exist.

advancedbootcampextensionholxxxx

**Amazon DynamoDB Configuration**

**DynamoDB Table Name**

The name of the DynamoDB Table where call center hours of operation and reason will be stored (Ensure you do not have a table with this name already).

callcenterhop-table

Cancel Previous Next



# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Configure stack options

Select or create an IAM role appropriate to execute this template. Tags are optional, not required. However, it is a best practice to add a tag for tracking / billing purposes.

The IAM role should have permissions to create:

- Amazon S3 buckets
- AWS Lambda functions
- Amazon Cognito user and identity pools
- Amazon DynamoDB tables

Click Next.

The screenshot shows the 'Configure stack options' page in the AWS CloudFormation console. The breadcrumb trail at the top reads 'CloudFormation > Stacks > Create stack'. The page title is 'Configure stack options'. There are two main sections: 'Tags' and 'Permissions'. The 'Tags' section includes a description: 'You can specify tags (key-value pairs) to apply to resources in your stack. You can add up to 50 unique tags for each stack. [Learn more.](#)'. It features a table with one tag: 'Name' with the value 'advancedbootcampextensionhol2500' and a 'Remove' button. Below the table is an 'Add tag' button. The 'Permissions' section includes a description: 'Choose an IAM role to explicitly define how CloudFormation can create, modify, or delete resources in the stack. If you don't choose a role, CloudFormation uses permissions based on your user credentials. [Learn more.](#)'. It has a sub-section 'IAM role - optional' with the instruction 'Choose the IAM role for CloudFormation to use for all operations performed on the stack.' Below this is a dropdown menu for 'IAM role name' with the selected value 'advancedbootcampextensionvm'. There is a 'Remove' button next to the dropdown. At the bottom of the 'Permissions' section is a warning box with a triangle icon and the text: 'AWS CloudFormation will use this role for all stack operations. Other users that have permissions to operate on this stack will be able to use this role, even if they don't have permission to pass it. Ensure that this role grants least privilege.'

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Create stack

Review your selections and if you are comfortable, select the acknowledgment check box at the bottom of the page.

Click Create stack.

Capabilities

**i** The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

☒ I acknowledge that AWS CloudFormation might create IAM resources.

Cancel

Previous

Create change set

Create stack

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Create in progress




The AWS CloudFormation template starts creating Amazon DynamoDB tables, an Amazon S3 bucket, and hosts the website. It also creates Amazon Cognito user and identity pools.

The creation process will take about 5 minutes. You can click the refresh button to see the tasks as they are executing in the CF Stack. Check the stack status to ensure it says **CREATE\_COMPLETE**.

Advancedbootcampextensionhol2500 [Delete] [Update] [Stack actions ▼] [Create stack]

Stack info | Events | Resources | Outputs | Parameters | Template | Change sets

### Overview

Stack ID	Description
arn:aws:cloudformation:us-east-1:111111111111:stack/advancedbootcampextensionhol2500/01777802331- ea76-11e9-9bb2-1230783fe8ac 	" - Create the basic foundation for streaming customer audio from Amazon Connect by deploying: - S3 Bucket for creating website to manage the agents associated with extension - Dynamo DB tables stores the agents associated with extension - Lambda function copies all the files and creates the default user sjobs
Status	Status reason
 CREATE_COMPLETE	-
Root stack	Parent stack
-	-
Created time	Deleted time
2019-10-09 10:20:26 UTC+0100	-
Updated time	
-	
Drift status	Last drift check time
 NOT_CHECKED	-
Termination protection	IAM role
Disabled	arn:aws:iam::111111111111:role/advancedbootcampextensionhol2500-01777802331- ea76-11e9-9bb2-1230783fe8ac

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Note the website URL

Select the **Outputs** tab and make a note of the **webSiteLocation** value.

This is the URL where the user interface is hosted. Copy this location to a clipboard or open it in a new browser tab.

Stack info

Events

Resources

Outputs

Parameters

Template

Change sets

Outputs (3)

Search outputs

Key	Value	Description	Export name
IAMRole	arn:aws:iam::123456789012:role/callcenterhop-lambda-role	The IAM Role created for the Lambda Functions in this template	-
callcenterhopTable	arn:aws:dynamodb:us-east-1:123456789012:table/callcenterhop-table	The ARN of the DynamoDB table created to store agent info	-
webSiteLocation	https://s3.amazonaws.com/123456789012-us-east-1/callcenterhop/callcenterhop.html	Site can be access with this link.	-

# Amazon Connect – Routing

## Build the User Interface to Set Dynamic Messages

### Summary of resources created

The Amazon CloudFormation template has created a user interface with the following resources:

1. Amazon DynamoDB table: callcenterhop-table
2. Amazon S3 bucket to store the website contents
3. Two Lambda functions ( \*\*\*\*checkHoursOfOperation\*\*\*\* and \*\*\*\*webSiteCreator\*\*\*\*)
4. Amazon Cognito User pool with a user name sjobs and an identity Pool – BootCampCognitoIDPool\*\*\*\*

# Amazon Connect – Routing

## Configure a Holiday Message

You will use the user interface previously created to add a holiday message and store the message in the Amazon DynamoDB table.

## Add a Holiday Message

### Access the website

Browse to the website URL noted in slide 12

A user account has already been created within the Amazon Cognito User Pool. Login using the following details:

- Username: sjobs
- Password: password

# Amazon Connect – Routing

## Configure a Holiday Message

You will use the user interface previously created to add a holiday message and store the message in the Amazon DynamoDB table.

## Add a Holiday Message

### Access the website

Browse to the website URL noted in slide 12

A user account has already been created within the Amazon Cognito User Pool. Login using the following details:

- Username: sjobs
- Password: password

# Amazon Connect – Routing

## Create a holiday message

To create a holiday message, use the date and time picker to select a Start Date Time and End Date Time.

Enter a value in the Reason field.

Click Submit

If necessary, click Refresh to see the data you entered in the table below.

**Amazon Connect Call Center Holiday/Emergency Closed Hours of Operation**  
Use this form to enter data in to DynamoDB

**Holiday/Emergency Hours of Operation**

Start Date

End Date Time

Reason

Start Date Time	End Date Time	Reason
10/09/2019 05:00:00	10/09/2019 11:39:00	Bank Holiday



# Amazon Connect – Routing

## Confirm data has been saved

The message data entered will be stored in the Amazon DynamoDB table `callcenterhop-table`.

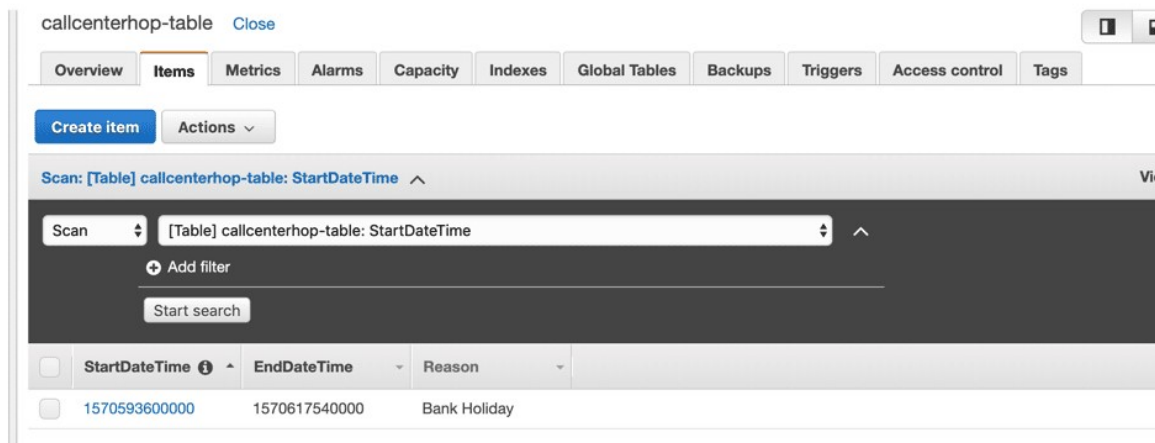
To verify this, open **Amazon DynamoDB (region N. Virginia)**,

In the sidebar menu, select **Tables**.

Click **callcenterhop-table**.

Click on **Items** to view data in the DynamoDB table.

You should see the data you entered in your website.



# Amazon Connect – Routing

## Test the Message in Amazon Connect

In this section, you will use an Amazon Connect instance and Contact Flow to play the holiday message that you previously set. If you do not have a Contact Flow, you will need to create one before continuing. The steps to create one are covered in the Amazon Connect product documentation and in the prerequisite training.

Note: If you are having trouble building your Contact Flow, use the sample flow attached here. Please download it and import it to your Amazon Connect Contact Flow interface.

[https://drive.google.com/file/d/1G9k6U0cur8J\\_wvdORmNSzZwvURNDHJ-9/view?usp=sharing](https://drive.google.com/file/d/1G9k6U0cur8J_wvdORmNSzZwvURNDHJ-9/view?usp=sharing)

# Amazon Connect – Routing


## Access your Contact Flow and test the message

## Allow list the Lambda function

Login to your Amazon Connect instance in N. Virginia, navigate to the Contact Flows page of the Instance configuration and allow the Lambda function: **check hours of operation** (your function name will have your Amazon CloudFormation stack name followed by checkHoursOfOperation and some numbers.)

## AWS Lambda

Amazon Connect can interact with your own systems and take different paths in IVR dynamically. To achieve this, invoke AWS Lambda functions in contact flows to interact with your own systems or other services, then build personalized and dynamic experiences based on data returned.

Note: By adding Lambda functions, you are granting Amazon Connect permission to invoke them [Create a new Lambda function](#) 

**Function** Select  [+ Add Lambda Function](#)

## Lambda Functions

```

hol-
checkHoursOfOperation-

```

# Amazon Connect – Routing

## Create the Contact Flow and invoke Lambda function

Within your Amazon Connect instance, create a contact flow that will invoke the AWS Lambda function: check hours of operation.

**Note:** If you are using the sample Contact Flow provided, update the AWS Lambda function you allowed in step 1.

Once the function has been successfully executed, use the **Check contact attributes** block to check if the attribute **holidayFound** returned by the function equals True.

If there is a No match, the call will be sent to the BasicQueue.

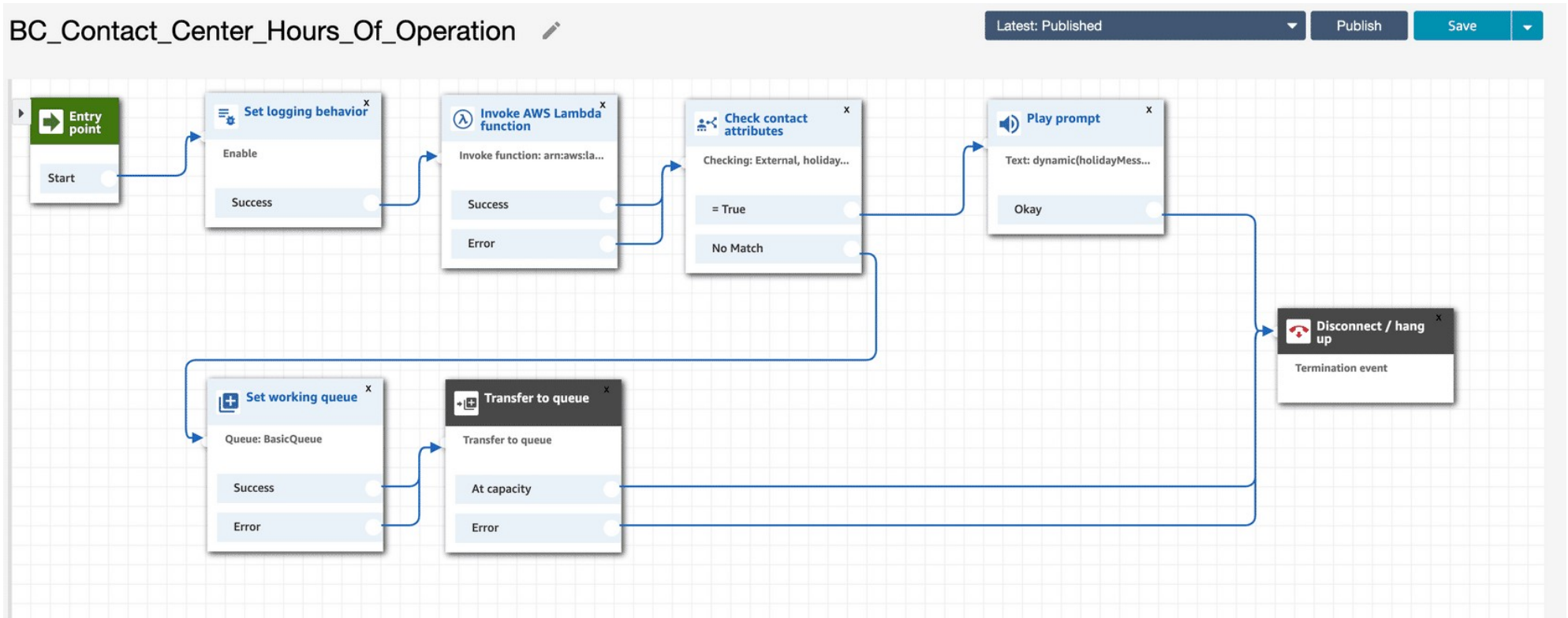
If there is a match, use the **Play prompt** block to play a message using TTS from the function attribute **holidayMessage** and then disconnect the call.

Above is a sample Contact Flow if you wish to follow.

Save and publish your Contact Flow. Assign it to a phone number.

# Amazon Connect – Routing

## Create the Contact Flow and invoke Lambda function



# Amazon Connect – Routing

## Test the message

To place or receive calls in your instance, you need to call the number you claimed.

Call the phone number. The message you set from the previous section plays. You can change this message dynamically by updating the website.

# Amazon Connect – Routing

## Play the message using SSML

You can use Amazon Polly to generate speech from either plain text or from documents marked up with Speech Synthesis Markup Language (SSML).

To play messages using SSML, you have to edit the contact flow, change the play prompt block, and select interpret as SSML.

Save and publish the Contact Flow.

### Play prompt ×

Plays audio.

Prompts can be an audio file, stored in the prompt library, or text-to-speech, which can optionally be specified in a flow via a contact attribute.

Prompt

☐ Select from the prompt library (audio)

☒ Text to speech (Ad hoc)

☐ Enter text

☒ Enter dynamically

Type

External

Attribute

holidayMessage

Interpret as

SSML

# Amazon Connect – Routing

## Test the message

## Enter the message with SSML

Go back to the website and enter a message with SSML tags, for example:

`< speak>The call center is closed today < break time="3s"/> please call back tomorrow </ speak>`

Make a call to test your message is played with SSML.



# Some layout options



## Current Options Are Not Solving the Problem

### STAFFING AGENCY

Do not help with  
shortage

### CAMPUS HIRING

Not enterprise ready

### VISA TALENT

High compliance &  
legal costs; delays

### INTERNAL TRAINING

Not core competency;  
bureaucratic

### BOOTCAMPS

Demand-supply  
mismatch; fragmented