



Amazon Connect と Amazon Pinpoint による 効果的なマルチチャネルコミュニケーション

AWS Black Belt Online Seminar

中村 達也 / 梅田 裕義

Solutions Architect

2023年8月

AWS Black Belt Online Seminar とは

- 「サービス別」「ソリューション別」「業種別」などのテーマに分け、アマゾンウェブサービスジャパン合同会社が提供するオンラインセミナーシリーズです
- AWS の技術担当者が、AWS の各サービスやソリューションについてテーマごとに動画を公開します
- 動画を一時停止・スキップすることで、興味がある分野・項目だけの聴講も可能、スキマ時間の学習にもお役立ていただけます
- 以下のURLより、過去のセミナー含めた資料などをダウンロードすることができます
- <https://aws.amazon.com/jp/aws-jp-introduction/aws-jp-webinar-service-cut/>

内容についての注意点

- 本資料では2023年8月時点のサービス内容および価格についてご説明しています。最新の情報は AWS 公式ウェブサイト(<https://aws.amazon.com>)にてご確認ください。
- 資料作成には十分注意しておりますが、資料内の価格と AWS 公式ウェブサイト記載の価格に相違があった場合、AWS 公式ウェブサイトの価格を優先とさせていただきます。
- 価格は税抜表記となっております。日本居住者のお客様には別途消費税をご請求させていただきます。

本セミナーの対象者

- 基本的な Amazon Connect の知識や操作経験をお持ちの技術者
- 基本的な Amazon Pinpoint の知識や操作経験をお持ちの技術者
- 電話やメールなど複数のチャネルを使い、効果的な顧客アプローチを検討されている方
- すでに電話やメールなどでアウトバウンド業務を行っており、効率的な運用を検討したい方

アジェンダ

- アウトバウンド業務の課題
- マルチチャネルコミュニケーション実装 DiveDeep
- まとめ

アウトバウンド業務の課題

アウトバウンド業務が直面する課題

キャンペーン案内、定期/緊急通知、他



電話、SMS、Emailの
アプローチが個別

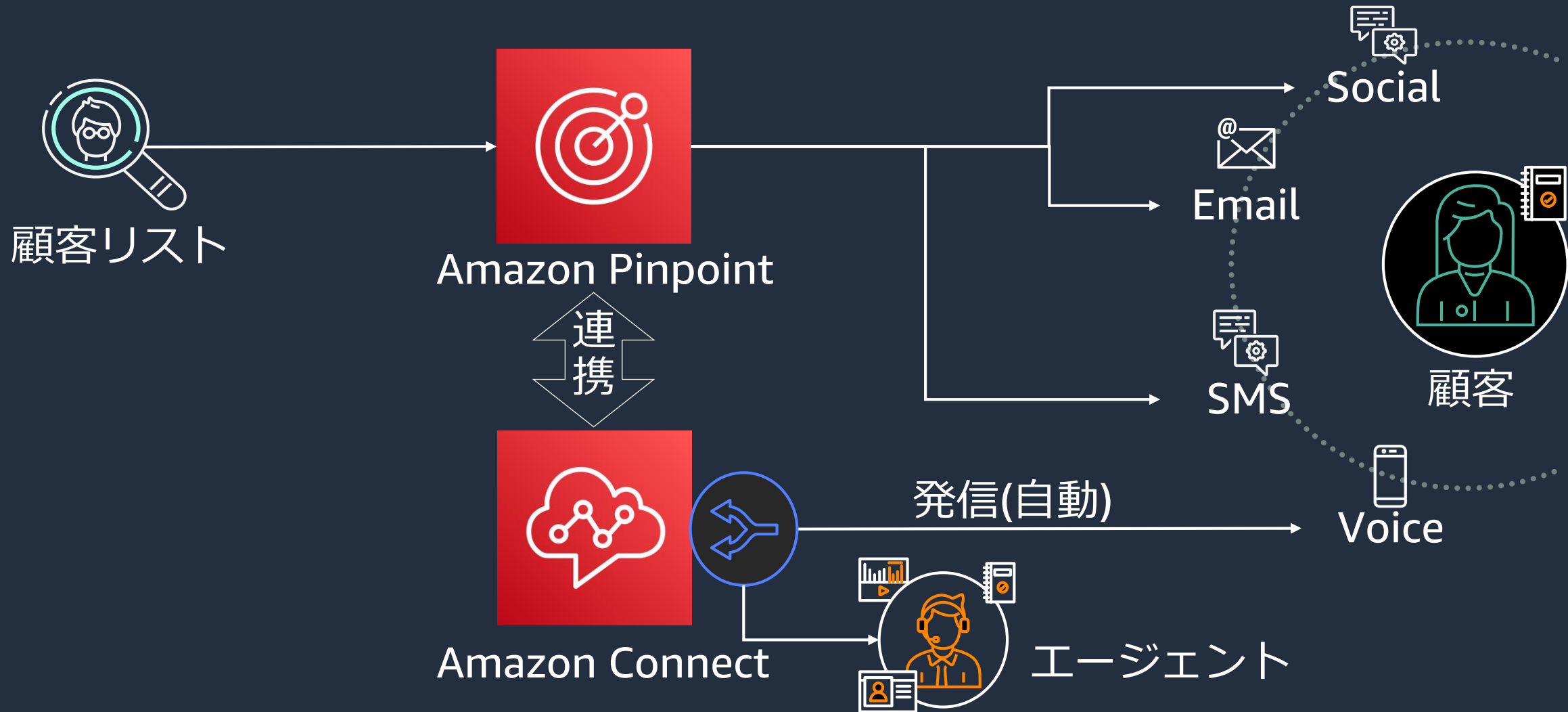
時間帯やチャンネルが管理さ
れていないため効果が薄い

手作業による電話発信、
全顧客への対応が非効率

効果が少なく、作業効率の悪いコミュニケーション

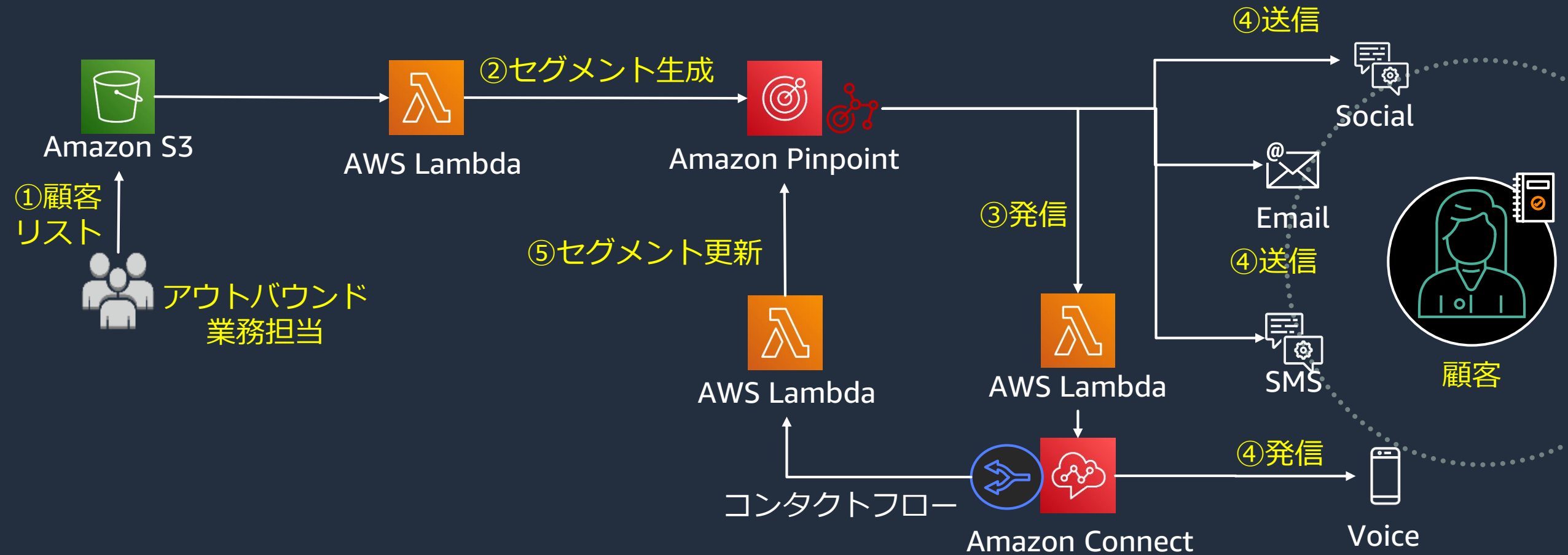
Amazon Connect と Pinpoint の活用

顧客の応答状況に応じて 電話 → SMS → Email の順にコンタクトを実施



アーキテクチャ図：マルチチャネルコミュニケーション

マルチチャネルのアウトバウンド業務をサーバーレスで実現



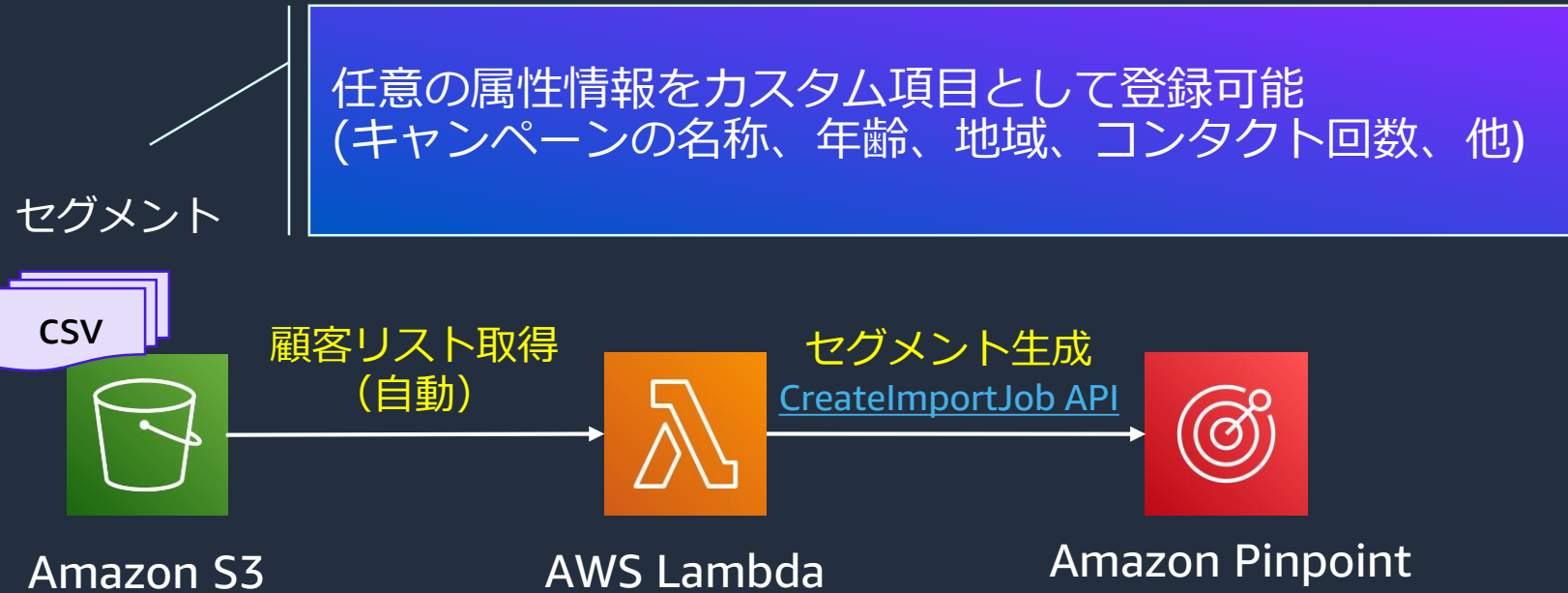
マルチチャネルコミュニケーション 実装 DiveDeep

Amazon Pinpoint へ顧客リストを登録

Amazon Pinpoint へ顧客リストを登録



Amazon Pinpoint へ顧客リストを登録



CSV データ

Channel	Address	User.UserID	Attributes.CallStatus	Attributes.Index
CUSTOM	+819011111111	User1	Pending	User1-Custom
SMS	+819011111111	User1	Pending	User1-SMS
EMAIL	test@example.com	User1	Pending	User1-EMAIL

Amazon Pinpoint へ顧客リストを登録

The screenshot shows the Amazon S3 console interface. The breadcrumb navigation is "Amazon S3 > バケット > pinpoint-connect-demo-calllistbucket". The main heading is "pinpoint-connect-demo-calllistbucket" with a "情報" (Info) link. Below the heading are tabs for "オブジェクト", "プロパティ", "アクセス許可", "メトリクス", "管理", and "アクセスポイント". The "オブジェクト" tab is active, showing "オブジェクト (5)". A description explains that objects are basic entities stored in Amazon S3 and provides instructions on how to access them. Below the description are buttons for "リフレッシュ", "S3 URI をコピー", "URL をコピー", "ダウンロード", "開く", "削除", and "アクション". There are also buttons for "フォルダの作成" and "アップロード". A search bar is present with the text "プレフィックスでオブジェクトを検". Below the search bar is a table of objects.

<input type="checkbox"/>	名前 ▲	タイプ ▼	最終更新日時 ▼	サイズ ▼	ストレージクラス ▼
<input type="checkbox"/>	OLD/	フォルダ	-	-	-
<input type="checkbox"/>	sample_call_sms_e_mail_list_summit_hi	csv	2023/03/10 06:00:21 PM JST	665.0 B	スタンダード

Amazon Pinpoint Pinpoint にインポートされた エンドポイント情報

対象のユーザーを特定するキーとなる情報 ※必須
[Channel]
送信するチャンネルのタイプ
[Address]
対象チャンネルタイプの具体的な情報

インポートされたエンドポイント情報

Endpoint Id	Channel	Address	User.UserID	Attributes.CallStatus	Attributes.Index
abc111XXX	CUSTOM	+819011111111	User1	Pending	User1-Custom
abc112XXX	SMS	+819011111111	User1	Pending	User1-SMS
abc113XXX	EMAIL	test@example.com	User1	Pending	User1-EMAIL

※任意

[Endpoint Id]

- ・対象を特定するユニークな識別子。
- ・指定をしなければランダムな値が採番される。

ユーザーの識別やメール文で使用する付帯情報 ※任意

[User.UserID]

- ・予め用意されている標準属性
 - ・UserIDを指定することで、同一ユーザーとして判別
- [Attributes.CallStatus]
- ・独自で設定したカスタム属性
 - ・対象チャンネルの配信ステータスを管理

REST API Reference > Endpoint

https://docs.aws.amazon.com/ja_jp/pinpoint/latest/apireference/apps-application-id-endpoints-endpoint-id.html

© 2023, Amazon Web Services, Inc. or its affiliates.



Amazon Pinpoint のエンドポイント作成方法

- **ImportJob を用いたバッチ処理**
 - プログラムから [CreateImportJob API](#) を実行、
もしくは [AWS マネージメントコンソール](#)から「セグメントをインポート」実行
 - CSV のリストを元に、一括でエンドポイント情報を登録可能
- **Update Endpoint を用いたリアルタイム処理**
 - プログラムから [Update Endpoint API](#) を実行
 - [AWS Amplify](#) にも統合されている

[Workshop]Amazon Pinpoint で作るカスタマーエクスペリエンス

・ Lab 1: カスタマーデータのインポート

<https://catalog.us-east-1.prod.workshops.aws/workshops/070a93a2-3373-4736-8af0-f7a08c9eb08a/ja-JP/import-customers>

Lab 7.1: エンドポイントの作成・削除

<https://catalog.us-east-1.prod.workshops.aws/workshops/070a93a2-3373-4736-8af0-f7a08c9eb08a/ja-JP/api-operations/create-update-endpoints>

Amplify Documentation > ANALYTICS > Update Endpoint

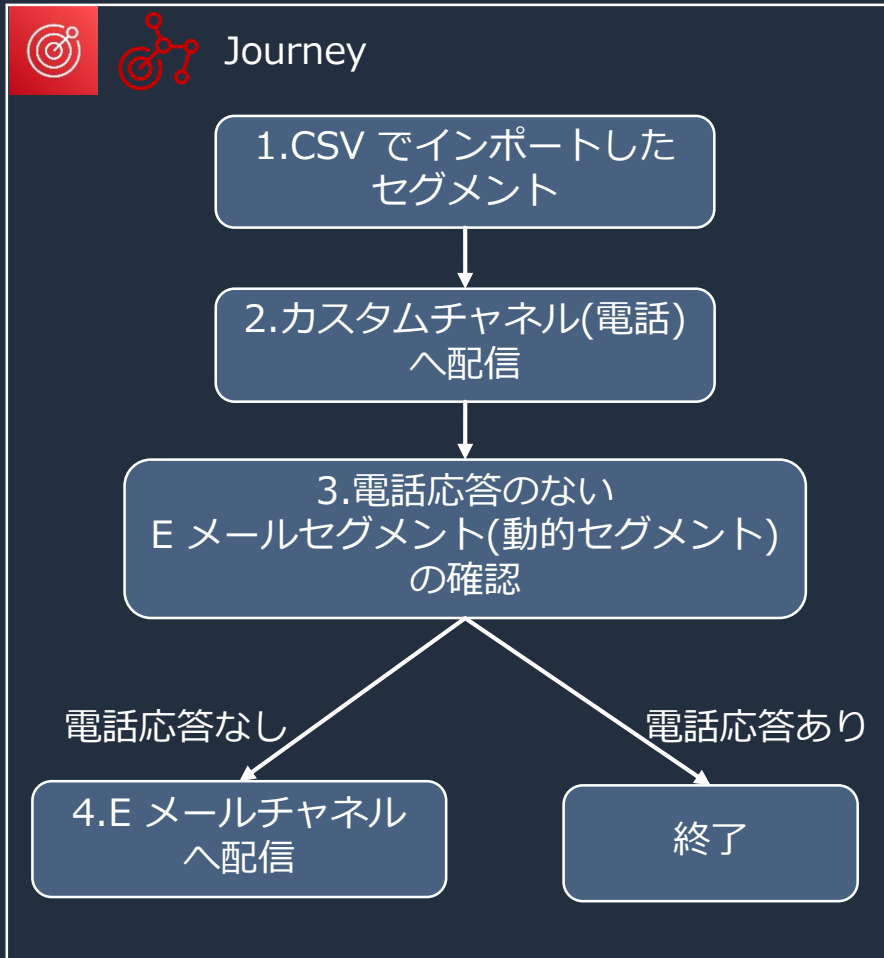
<https://docs.amplify.aws/lib/analytics/update-endpoint/q/platform/js/>

Amazon Pinpoint のジャーニーを使った マルチチャンネル配信

Amazon Pinpoint のジャーニーを使ったマルチチャネル配信



ジャーニーを使ったマルチチャネル配信設定



[ジャーニーの設定情報]

1. CSVでインポートしたセグメントをジャーニーの対象とする
2. カスタムチャネル(電話)へ配信
3. 1時間待機し、電話応答のないEメールのセグメント(動的セグメント)を配信対象として指定
4. Eメールチャネルへを配信

動的セグメントの対象

Endpoint Id	Channel	Address	User.UserID	Attributes.CallStatus
abc111XXX	CUSTOM	+819011111111	User1	PENDING
abc112XXX	SMS	+819011111111	User1	PENDING
abc113XXX	EMAIL	test@example.com	User1	PENDING
abc114XXX	EMAIL	test2@example.com	User2	SUCCESS

動的セグメントで対象となるエンドポイント
[条件]
ChannelType = Eメール
CallStatus = PENDING

ジャーニーを使ったマルチチャネル配信設定

The screenshot shows the Amazon Pinpoint console interface. At the top, there are browser tabs for 'S3 Management Console', 'Amazon Pinpoint | ap-northea...', and 'ジャーニー | Amazon Pinpoint |'. The AWS navigation bar includes the 'aws' logo, 'サービス' (Services), a search bar, and a '[オプション+S]' (Options+S) button. The main navigation pane on the left shows 'Amazon Pinpoint', 'AWS Amplify', 'Lambda', 'IAM', and 'S3'. The breadcrumb trail is 'Pinpoint > すべてのプロジェクト > pinpoint-connect-demo > セグメント'. The page title is 'セグメント Info'. Below the title are tabs for 'セグメント', 'スケジュールされたインポート', and '最近のエクスポート'. The main content area is titled 'セグメント (8)' and contains a search bar, a refresh button, an 'アクション' (Action) dropdown, a 'セグメントをインポート' (Import Segment) button, and a 'セグメントを作成' (Create Segment) button. A table lists 8 segments with columns for 'セグメント名' (Segment Name), 'タイプ' (Type), 'セグメント ID' (Segment ID), and '最終変更日' (Last Modified). The first row is highlighted with a mouse cursor.

<input type="checkbox"/>	セグメント名	タイプ	セグメント ID	最終変更日
<input type="checkbox"/>	sample_call_sms_email_list_blackbelt_20230711.csv	インポート済み	1160075b9a1b41fe8e6cd48025e09e09	July 11th 2023, 1:27, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_shimizyu_dynamic	動的	39247fc7b38a4fbab6cfca1e5f467418	April 10th 2023, 12:07, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_shimizyu.csv	インポート済み	8a648ed3a6ee4e279359bbb2aa0ba1eb	April 10th 2023, 12:03, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_nakamura_dynamic	動的	00a04dd8e64d47eda1c103f310e6fa0d	March 14th 2023, 2:16, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_naktat.csv	インポート済み	4d553f3483394d808ae80063cdd3215b	March 14th 2023, 1:30, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_hiroyou_dynamic	動的	0d6f384207994b889b5a0b94ec7890ae	March 10th 2023, 9:07, UTC
<input type="checkbox"/>	sample_call_sms_email_list_summit_hiroyou.csv	インポート済み	878eadedef5f4aa09c3f0ea6c1d8bef5	March 10th 2023, 9:00, UTC
<input type="checkbox"/>	sample_call_sms_email_list_test_shouzuki_230220_1000.csv	インポート済み	6e1bade107fc4b379157513b93a922f5	February 22nd 2023, 3:40, UTC

ジャーニーとは

ジャーニーの設定例

The screenshot displays the Amazon Pinpoint Journey Builder interface for a journey named "sample_call_sms_email_list_blackbelt_20230711". The journey is currently in "Draft" status. Key details include: Start: "パブリッシュ後ただちに" (Immediately after publish), Stop: "120 か月 (3600 日) 後" (120 months (3600 days) later), and Auto-save: "自動保存済み a few seconds ago (5:19 PM)".

The workflow consists of the following steps:

- ジャーニーエントリ** (Journey Entry): Segment: "sample_call_sms_email_list_blackbelt_20230711.csv".
- カスタムチャネル経由で送信** (Send via Custom Channel): Lambda function: "pinpoint-connect-demo-call-initiator-lambda", End point type: "カスタム" (Custom).
- はい/いいえ分割** (Yes/No Split): Delay: "次の後に評価: 1 hours" (Evaluate 1 hour after next). Segment: "sample_call_sms_email_list_blackbelt_20230711_dynamic".
- Eメールを送信** (Send Email): Includes a "メッセージをプレビュー" (Preview message) button. Details: Template: "Wacem_to_Pinpoint", Version: "有効なバージョン1", Sender address: "sender@example.com".

A dark button at the bottom right asks "ジャーニーはいかがですか?" (How is the journey?).

- ・ 複数ステップの顧客エンゲージメントを実現することができる機能

- ・ セグメントまたはカスタムイベントを使い、どの顧客がジャーニーに参加するかを定義

- ・ 顧客がジャーニーで迎えるアクティビティを設定

[アクティビティの設定項目]

- Eメール・SMS・カスタムチャネルによる

- メッセージの送信

- セグメントまたはカスタムイベントを使い、

- はい/いいえで分岐

- 一定時間の待機、ランダムな分割などが可能

Amazon Pinpoint のジャーニー

https://docs.aws.amazon.com/ja_jp/pinpoint/latest/userguide/journeys.html

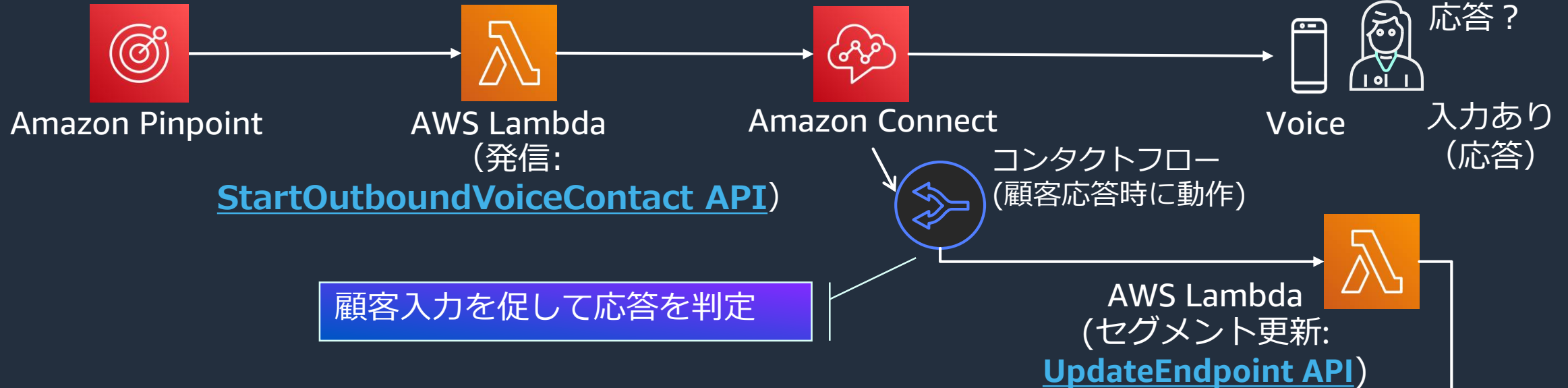
Amazon Connect のアウトバウンド処理

Amazon Connect のアウトバウンド処理



Amazon Connect のアウトバウンド処理

Pinpoint からの発信リクエストに対して顧客にアウトバウンドを実施

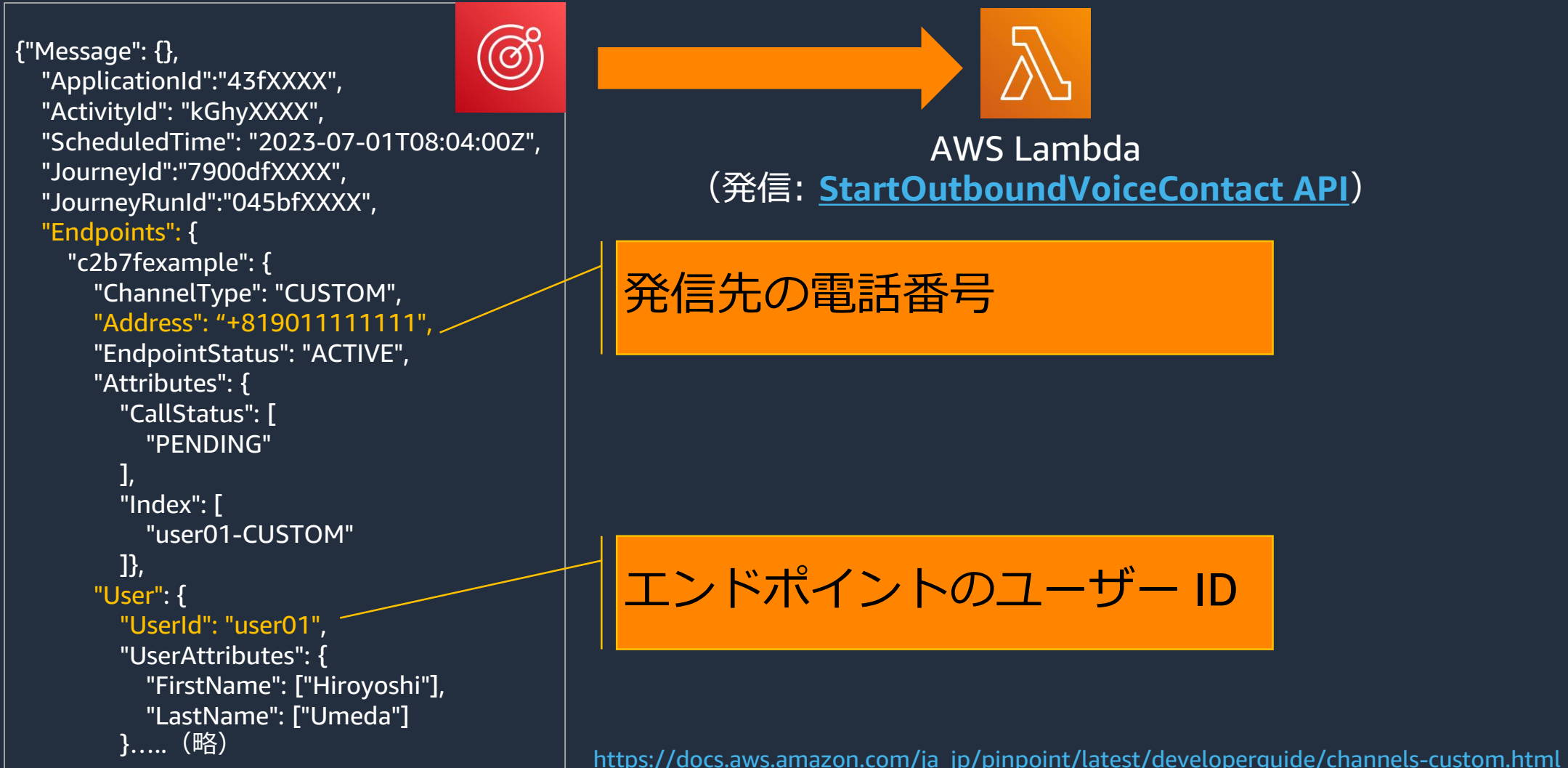


セグメント (顧客リスト)

Channel	Address	User.UserID	Attributes.Index	Attributes.CallStatus
CUSTOM	+819011111111	User1	User1-Custom	Pending → COMPLETED
SMS	+819011111111	User1	User1-SMS	Pending → CANCELED
EMAIL	test@example.com	User1	User1-EMAIL	Pending → CANCELED

StartOutboundVoiceContact API による発信処理

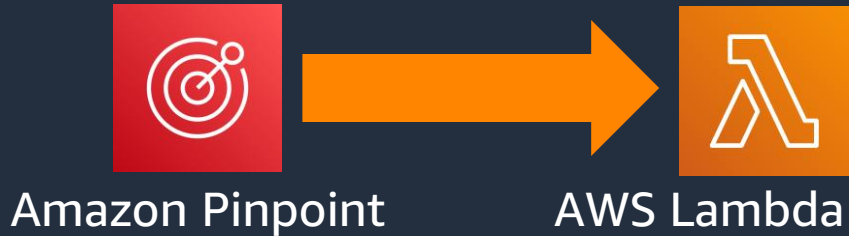
- Amazon Pinpoint から Lambda をトリガーするとイベントデータが送信される



StartOutboundVoiceContact API による発信処理

- API はイベントデータから取得した値をパラメータに指定

サンプルコード(Python 3.9)

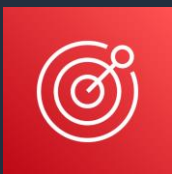


- インスタンス ID
- 発信用の Queue
- コンタクトフロー ID
- 発信先の電話番号
- コンタクト属性 (オプション)

```
connect = boto3.client('connect')
def lambda_handler(event, context):
    for key in event['Endpoints'].keys():
        connectResponse = connect.start_outbound_voice_contact(
            InstanceId = os.environ['INSTANCE_ID'],
            QueueId = os.environ['QUEUE_ID'],
            ContactFlowId = os.environ['CONTACT_FLOW_ID'],
            DestinationPhoneNumber = event['Endpoints'][key]['Address'],
            Attributes = {
                "UserId": event['Endpoints'][key]['User']['UserId'],
            }
        )
```

StartOutboundVoiceContact API による発信処理

サンプルコード(Python 3.9)



Amazon Pinpoint

AWS Lambda

```
connect = boto3.client('connect')
def lambda_handler(event, context):
    for key in event['Endpoints'].keys():
        connectResponse=connect.start_outbound_voice_contact(
            InstanceId = os.environ['INSTANCE_ID'],
            QueueId = os.environ['QUEUE_ID'],
            ContactFlowId = os.environ['CONTACT_FLOW_ID'],
            DestinationPhoneNumber = event['Endpoints'][key]['Address'],
            Attributes = {
                "UserId": event['Endpoints'][key]['User']['UserId'],
            }
        )
```

イベントデータ (抜粋)

```
"Endpoints": {
  "c2b7fexample": {
    "Address": "+819011111111",
    "User": {
      "UserId": "user01",
    }..... (略)
```



```
event['Endpoints'][key]['Address']
→"+819011111111"
```

```
event['Endpoints'][key]['User']['UserId']
→"user01"
```

アウトバウンドで動作するコンタクトフロー

The screenshot displays the Amazon Connect console interface for a contact flow named "pinpoint-connect-demo-flow-BB". The flow is designed for inbound contacts and consists of several sequential steps:

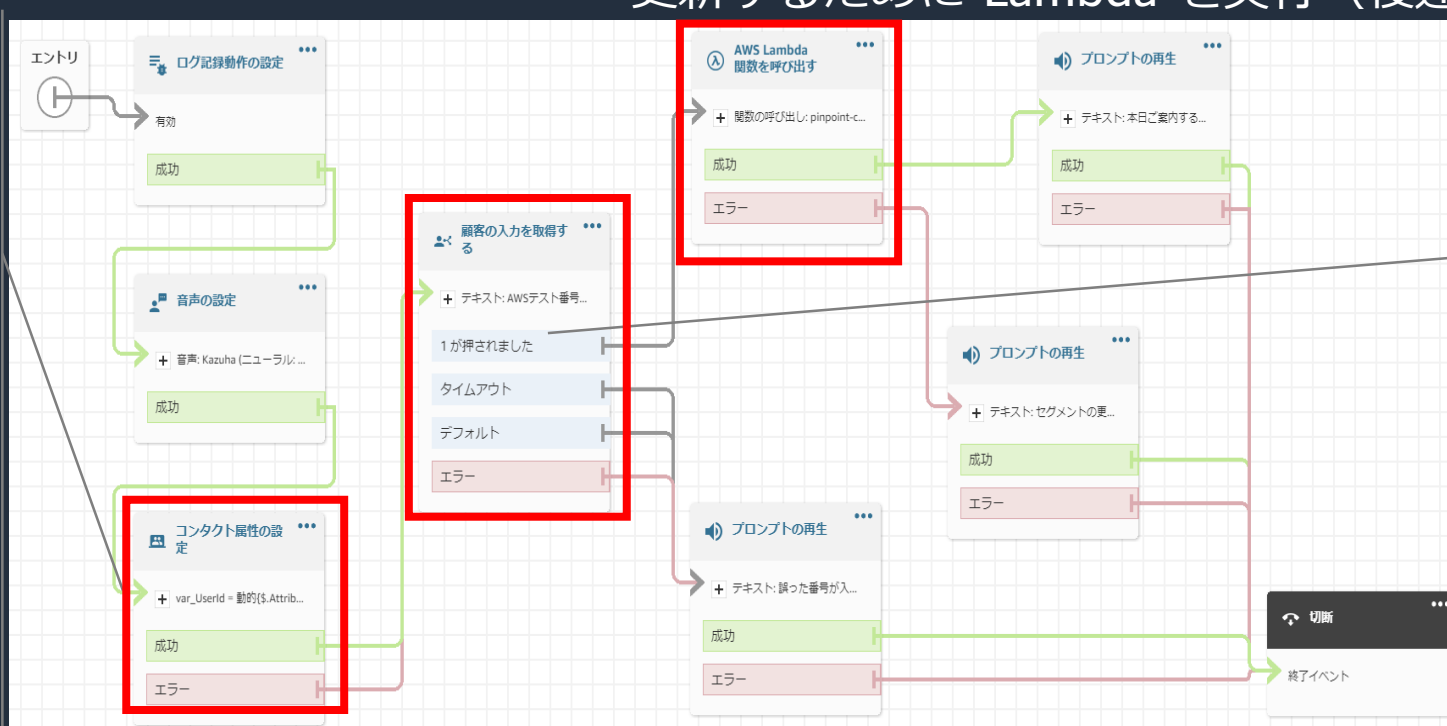
- エントリ (Entry):** The flow begins with an entry point.
- ログ記録動作の設定 (Log Recording Action):** A configuration step for logging actions, currently set to "有効" (Enabled).
- 音声の設定 (Voice Settings):** A step for configuring voice, including a prompt: "音声: Kazuha (ニューラル...)".
- コンタクト属性の設定 (Contact Attribute Settings):** A step for setting contact attributes, with a variable: "var_Userid = 動的({\$.Attrib...)".
- 顧客の入力を取得する (Get Customer Input):** A step to capture customer input, with options for "テキスト: AWSテスト番号...", "1が押されました", "タイムアウト", "デフォルト", and "エラー".
- AWS Lambda 関数を呼び出す (Call AWS Lambda Function):** A step to call an AWS Lambda function, with options for "関数の呼び出し: pinpoint-c...", "成功", and "エラー".
- プロンプトの再生 (Play Prompt):** Two steps for playing prompts, with options for "テキスト: 本日ご案内する...", "成功", "エラー", "テキスト: セグメントの更...", "成功", and "エラー".
- 終了イベント (End Event):** The flow concludes with a "切断" (Disconnect) event leading to a "終了イベント" (End Event).

The interface includes a search bar, a settings icon, and buttons for "最新" (Latest), "公開" (Publish), and "保存" (Save). The flow is currently at 80% zoom.

アウトバウンドで動作するコンタクトフロー（まとめ）

応答が確認できた場合はセグメントを
更新するために Lambda を実行（後述）

このスクリーンショットは「コンタクト属性の設定」ダイアログを示しています。上部には「コンタクト属性として key-value ペアを定義して保存します」という説明があります。下部には「動的に設定」が選択されており、名前空間「ユーザー定義済み」、キー「var_UserId」が設定されています。



このスクリーンショットは「顧客の入力を取得する」ダイアログを示しています。「手動で設定」が選択されており、読み上げるテキストが入力されています。また、「動的に設定」のセクションも表示されています。

Lambda から API に
渡した Attributes は
“コンタクト属性の設定”
ブロックで取得可能

お客様に対する確認アナウンス
から応答可否を判定
（例） 詳細のご案内をしてよろしい
場合は、1を押して下さい。

（ポイント）
後続の Lambda でセグメント
更新のため UserId を取得

アウトバウンドにおけるセグメント更新



セグメント (顧客リスト)

Channel	Address	User.UserID	Attributes.Index	Attributes.CallStatus
CUSTOM	+819011111111	User1	User1-Custom	Pending → COMPLETED
SMS	+819011111111	User1	User1-SMS	Pending → CANCELED
EMAIL	test@example.com	User1	User1-EMAIL	Pending → CANCELED



UpdateEndpoint API が Pinpoint のセグメントを更新

(API 使用時の注意点)

- セグメント内のエンドポイントに対して Update を実施
- 引数に application Id(固定), Endpoint Id(対象レコード) が必要

(実施内容)

- エンドポイントからチャネルタイプがカスタム（電話）, SMS, EMAIL の各レコードの Endpoint Id を特定してから CallStatus を更新

(方法)

- コンタクトフローで特定済みの UserId を使用
- UserId から SMS, EMAIL の Index を使用して、Endpoint Id の該当レコードを検索
- CUSTOM（電話）、SMS、EMAIL のエンドポイントの Status を API によって更新

セグメント（顧客リスト）

	Endpoint Id	Channel	Address	User.UserId	Attributes.Index	Attributes.CallStatus
エンドポイント	abc111XXX	CUSTOM	+819011111111	User1	User1-Custom	Pending
エンドポイント	abc112XXX	SMS	+819011111111	User1	User1-SMS	Pending
エンドポイント	abc113XXX	EMAIL	test@example.com	User1	User1-EMAIL	Pending

(例) 電話発信成功時にセグメント内の SMS レコードを更新



サンプルコード(Python 3.9)

```
# 初期化
client = boto3.client('pinpoint')
applicationId = os.environ['APP_ID']
userId = event['Details']['Parameters']['userId']

# セグメントからエンドポイント取得
response = client.get_user_endpoints(
    ApplicationId = applicationId, UserId=userId)

# 対象のエンドポイントを更新
for endpoint in response['EndpointsResponse']['Item']:
    if endpoint['Attributes']['Index'][0] == userId + '-SMS':
        endpoint['Attributes']['Status'][0] = 'CANCELED'
        result = client.update_endpoint(
            ApplicationId=applicationId,
            EndpointId=endpoint['Id'],
            EndpointRequest={
                'Attributes': endpoint['Attributes']}
        )...略
```

Pinpoint オブジェクトを初期化
アプリケーション ID は Amazon Pinpoint プロジェクトの ID、
固定値のため環境変数から取得
userId は検索キーとなり、コンタクトフロー内の属性値から取得可

セグメント一覧は [get_user_endpoints](#) で取得可能
セグメントからエンドポイントを検索するために使用

セグメント (顧客リスト)

Endpoint Id	Index	UserID	Status
abc111XXX	User1-Custom	User1	Pending
abc112XXX	User1-SMS	User1	Pending
abc113XXX	User1-EMAIL	User1	Pending

Endpoint Id	Index	UserID	Status
abc111XXX	User1-Custom	User1	Pending
abc112XXX	User1-SMS	User1	CANCELED
abc113XXX	User1-EMAIL	User1	Pending

Index で検索する事
で EndpointId を特定

まとめ

まとめ

- Amazon Pinpoint はマルチチャネルコミュニケーションを通じた顧客へのアプローチを自動化する事で、効率的なアウトバウンド業務を実現します
- Amazon Pinpoint のジャーニーは SMS、Email、カスタムチャネルを組み合わせた高度なフロー作成が設定だけで可能になります
- Amazon Connect との連携により電話を含むアウトバウンドの仕組みをサーバーレスに構築する事が可能です

本資料に関するお問い合わせ・ご感想

技術的な内容に関しましては、有料の AWS サポート窓口へお問い合わせください

<https://aws.amazon.com/jp/premiumsupport/>

料金面でのお問い合わせに関しましては、カスタマーサポート窓口へお問い合わせください（マネジメントコンソールへのログインが必要です）

<https://console.aws.amazon.com/support/home#/case/create?issueType=customer-service>

具体的な案件に対する構成相談は、後述する個別相談会をご活用ください



ご感想は Twitter へ！ハッシュタグは以下をご利用ください
#awsblackbelt

その他コンテンツのご紹介

ウェビナーなど、AWS のイベントスケジュールをご参照いただけます

<https://aws.amazon.com/jp/events/>

ハンズオンコンテンツ

<https://aws.amazon.com/jp/aws-jp-introduction/aws-jp-webinar-hands-on/>



Thank you!