

Journey Data Service

Zubin Dang

Product Management – Journey Data Service

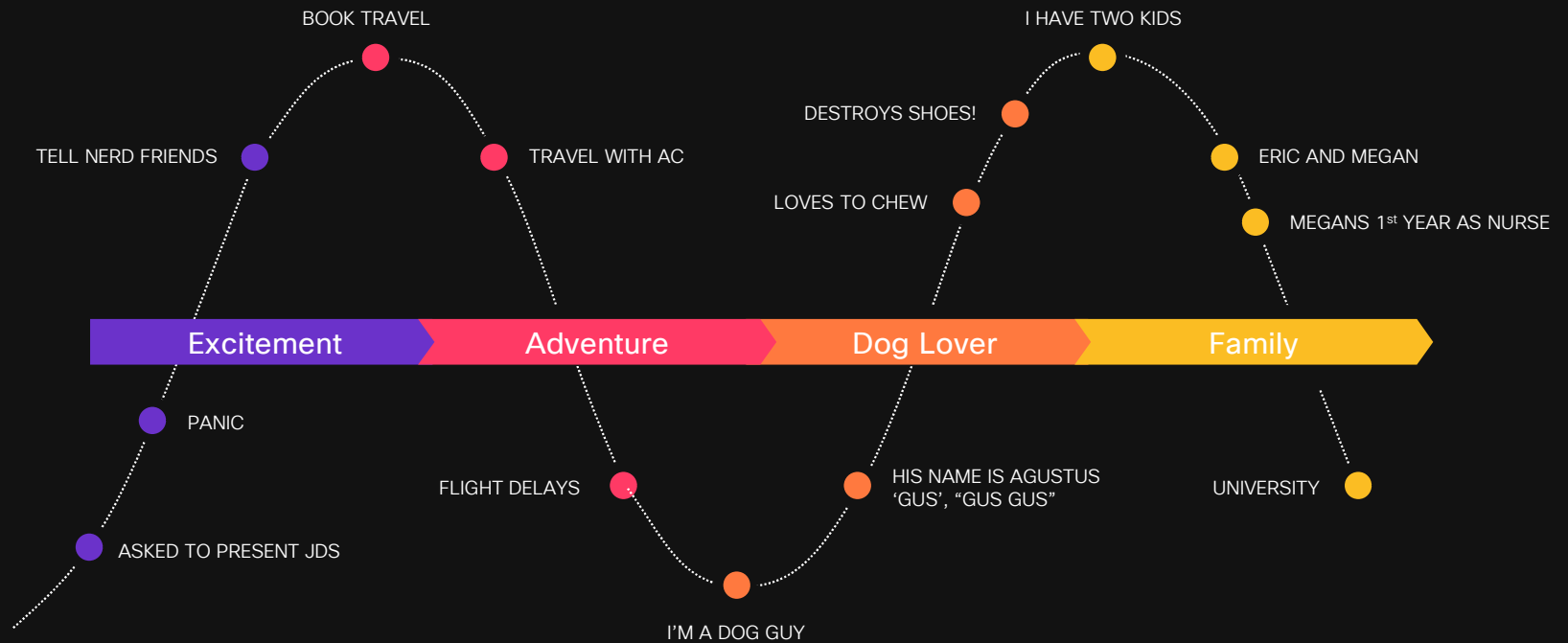
Barry McLellan

USC and SLED CC TSA, US & Americas CC TSA

© 2021 Cisco and/or its affiliates. All rights reserved. Cisco Confidential



Me, the customer



Power of personalization

71%

consumers expect companies to deliver personalized interactions

76%

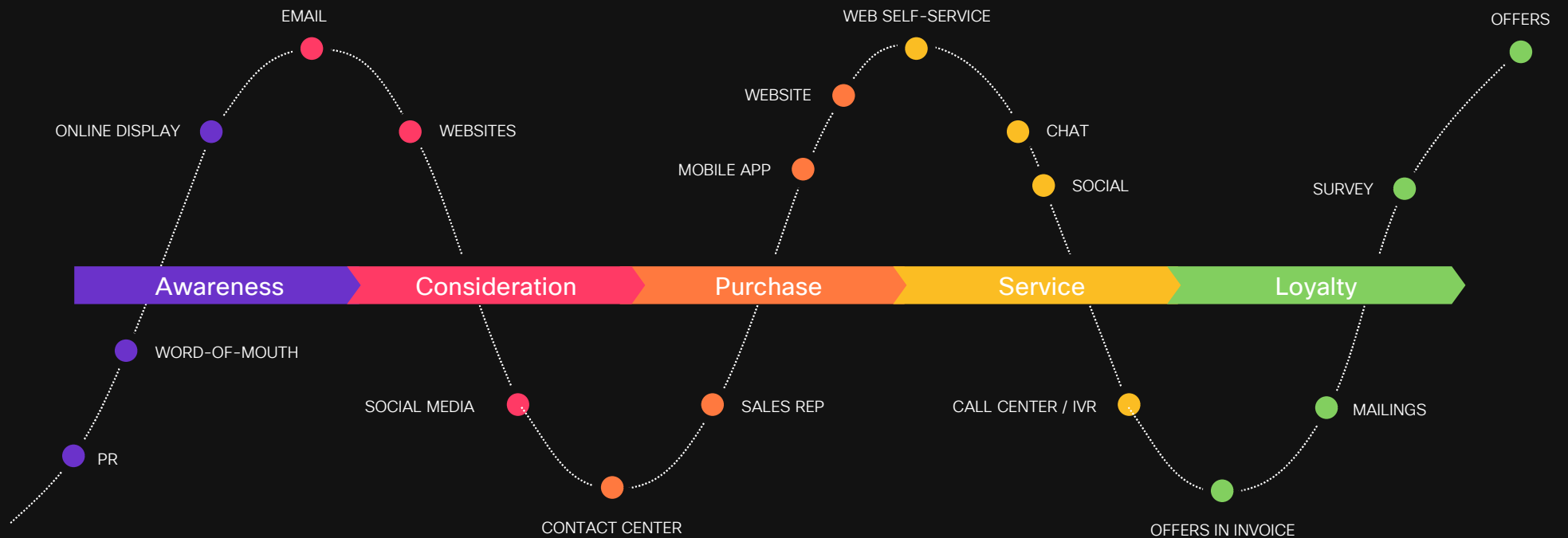
consumers get upset when that doesn't happen

Agenda

- ☐ Problem Statement
- ☐ Overview
- ☐ Journey Data Service
- ☐ Customer Journey Widget
- ☐ Use Case Deep Dive
- ☒ 06 Key Concepts and FAQ's
- ☒ 07 Timelines, Milestones, and Q&A

Problem Statement

Inside the disconnect



Disconnect between business and customers

80%

of businesses believe
they deliver a superior
experience

8%

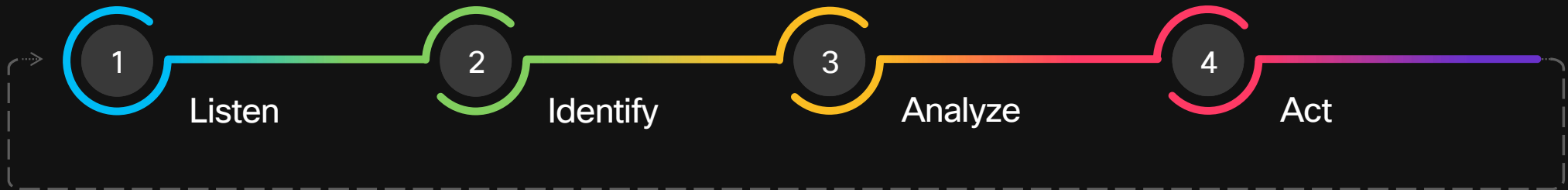
of their customers
agreed

Overview

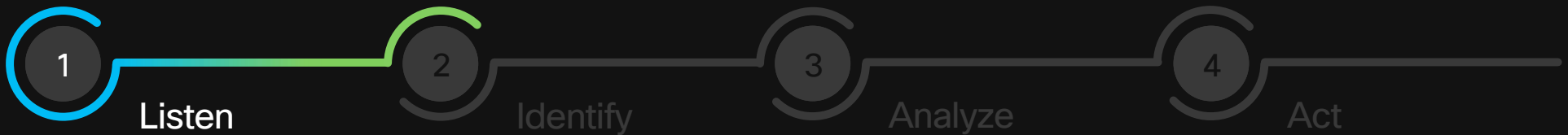
CX Challenges for Evolving Customer Expectation



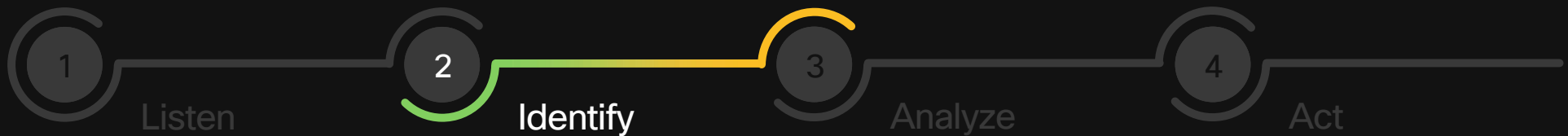
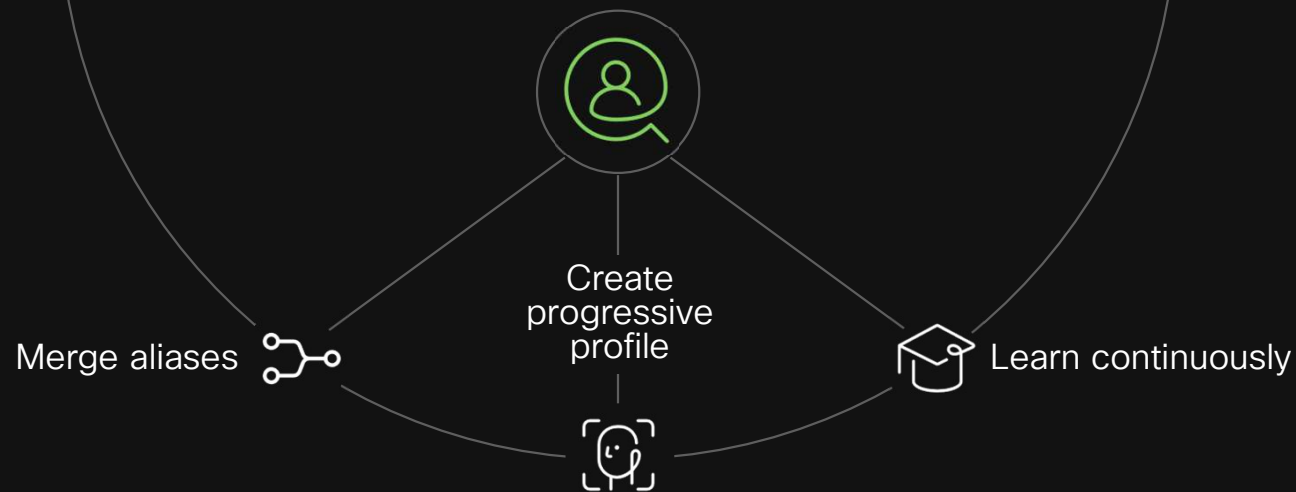
Customer journey lifecycle for customer experience leaders



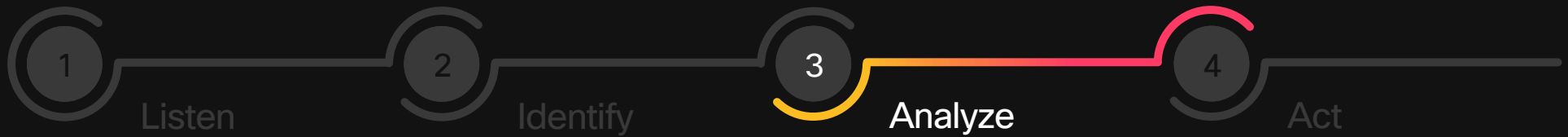
Listen: real-time journey tracking



Identify: dynamic profiling

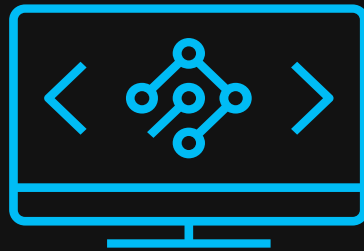


Analyze: advanced analytics

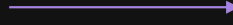


Act: powerful orchestration

Real-time
activities



Trigger
action



Offer

Webex App video connect



Human agent



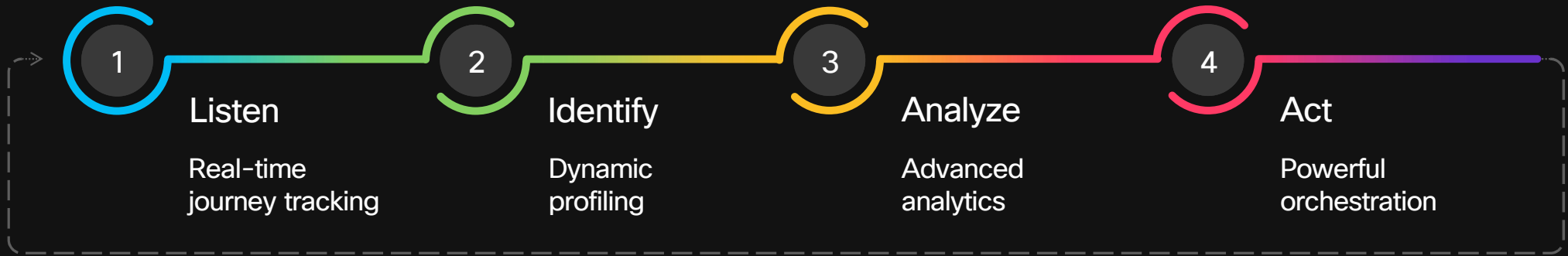
AI assistance



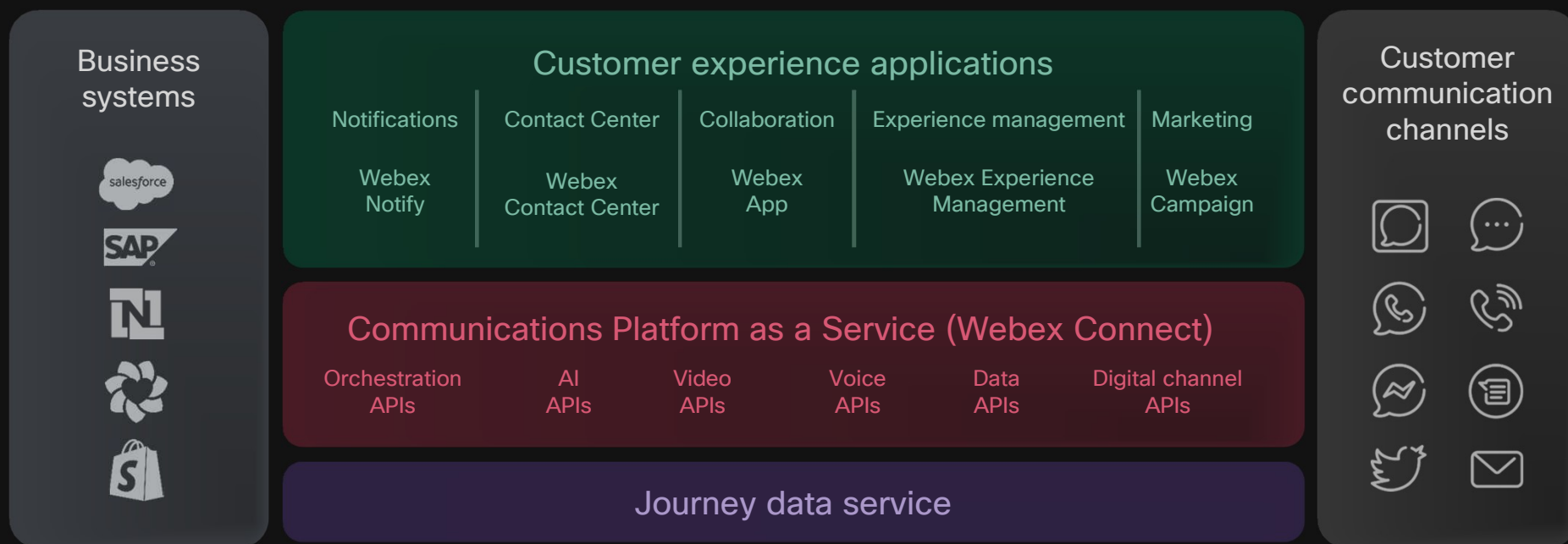
Offer / Voice of Customer



Customer journey lifecycle for customer experience leaders



Overall Portfolio

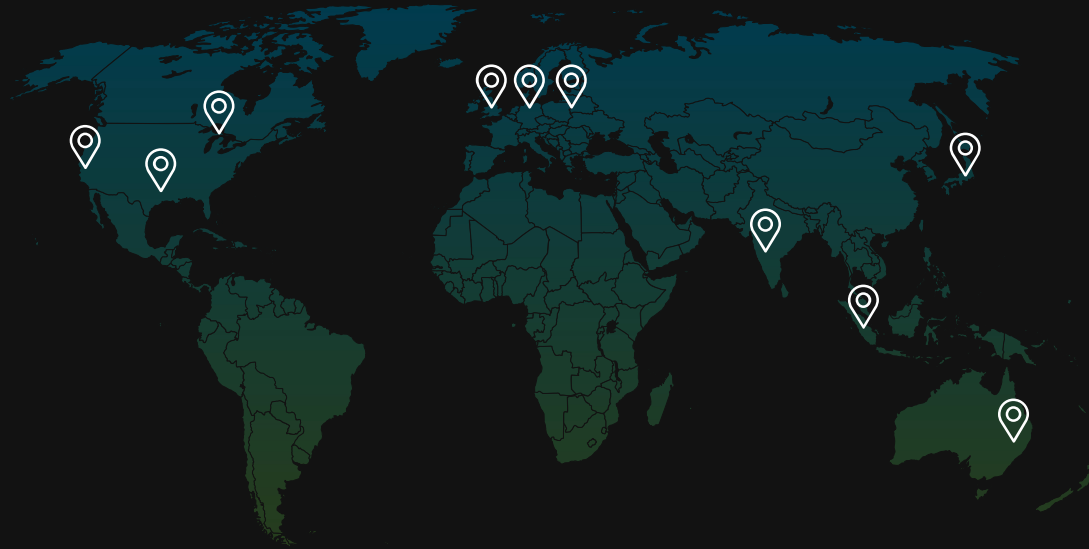


WAP

IL5, FedRamp, Private Cloud
and Public cloud

Federated

GDPR compliant



Calling



Meeting



Devices



Messaging



Slido



CPaaS

Data Variety



Client

- Media Quality
- Operational
- Diagnostic
- Feature Usage

webex



Server

- Media Quality
- Operational
- Diagnostic
- Cascade reports
- Usage
- Clickstream



User

- Identifier
- Relations
- Usage



Business

- Billing Usage
- Subscription



23TB/day



43TB/day



30GB/day



5GB/day

Data Volume

Data Velocity



10K
messages/sec



100K
messages/sec

Empowers

- End User Analytics e.g. People Insight
- IT analytics
- Fraud analytics
- Troubleshooting
- Business Intelligence
- Partner Analytics
- API – Partners

Journey Data Service

JDS

Collects data from various sources

Creates one customer view

Tracks customer touchpoints

Provides insights into customer experience

Optimizes marketing and sales

CRM

Manages customer data and interactions.

Centralized database for customer information.

Segments customers for personalized outreach.

Enhances customer relationships and drives sales.

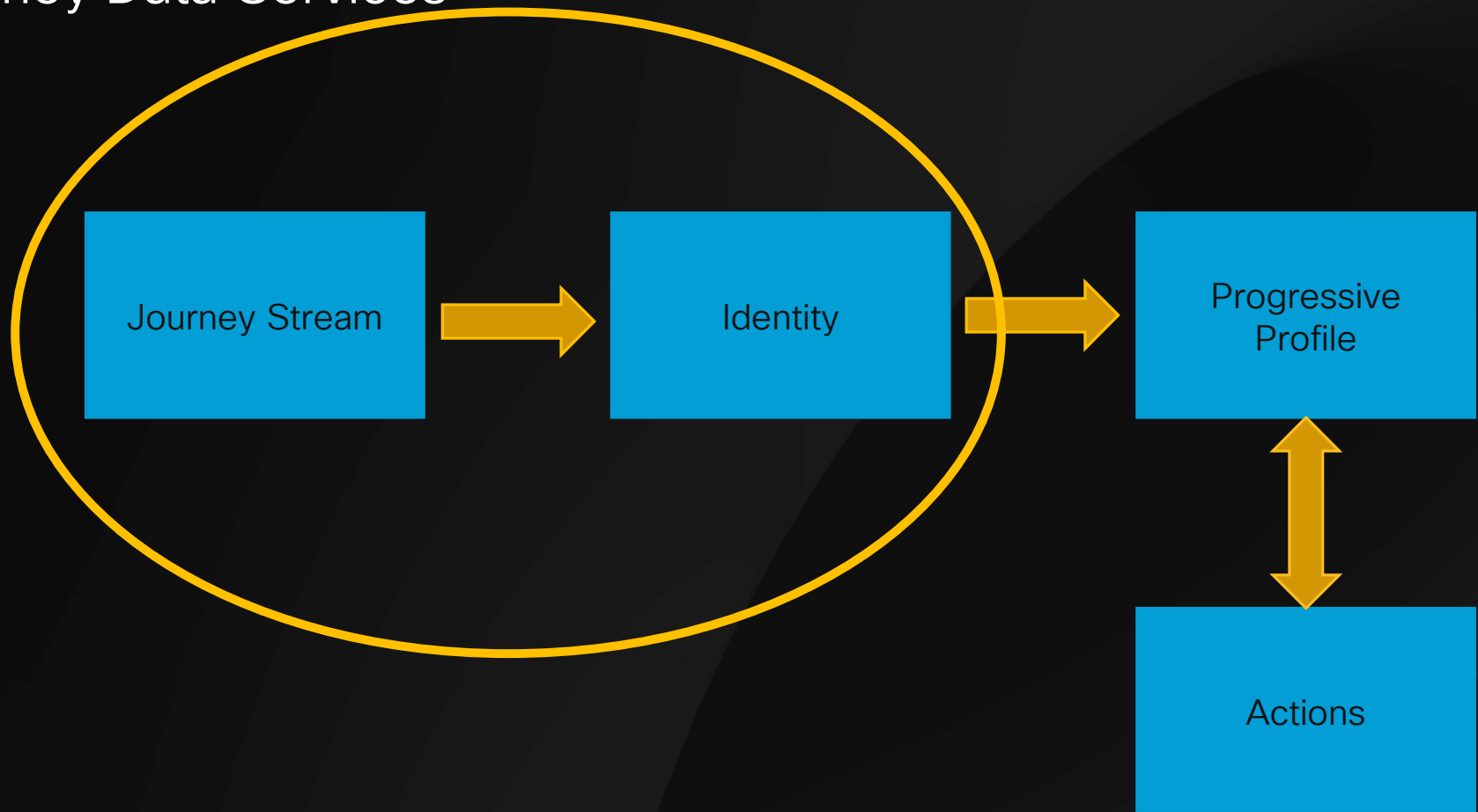
Manages customer interactions for all teams.

API-First

Cisco Advances API-First Strategy to Empower
Developers in the Digital Economy

[Sample Files](#): JDS Tech and Live Postman Examples

Journey Data Services



Identity and Event Feed

First: Barry

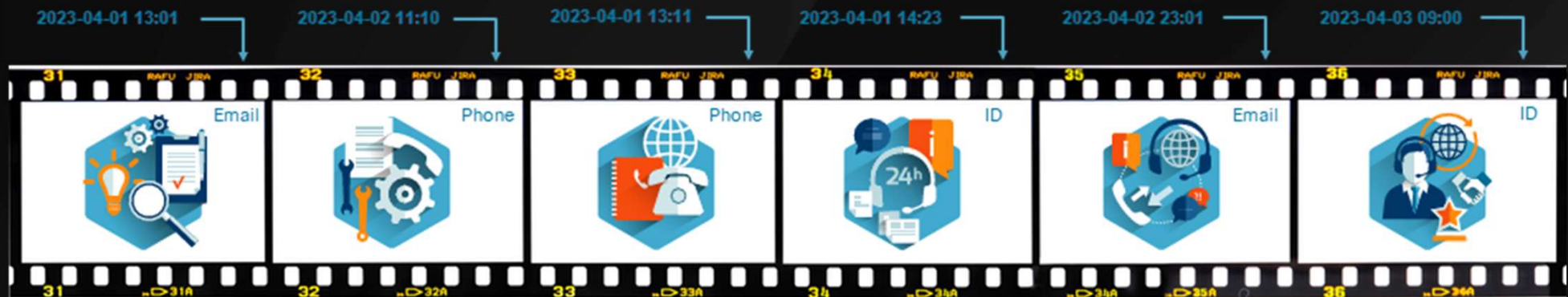
Last: McLellan

Phone: 111-222-3333
226-376-2555

JDS chronologically links records together based on identity aliases

Email: bmclella@cisco.com
barry@home.com

CustomerID: 2266445



Event Feed

The screenshot displays the 'Journey' event feed interface. At the top, there are filters for 'All', 'Last 24 Hours', 'Last 7 Days', and 'Last 30 Days'. Below these, there is a 'Livestream' toggle and a 'Show 1 new events' button. A date filter for '4/8/2023' is also present. The main feed shows four events, each with a custom icon, title, subtitle, and timestamp.

Event Type	Details	Timestamp
Voice	Callback phone: 12263762555 agent: Agent1 Demo origin: Voice channelType: Callback channelBreakout: SMS	4/8/2023 11:45 AM
SendNotes	Notes	4/8/2023 11:44 AM
OTP	OTP-pass	4/8/2023 11:44 AM
covid instructions	Information	4/8/2023 11:44 AM

webex

- Chronological journey
- Customizable data fields
- Clickable URL's
- Custom Icons and Colors
- Insert events from any platform

POST Event Explained

HEADER:

Content-Type
Authorization

application/json

[SharedAccessSignature](#) so=cctsa&sn=sandbox&ss=ds (datasink-write-token)

{		
"id": "xxxx-xxxx-xxxx-xxxx",	Unique UUID	Required
"time": "",	Timestamp ISO 8601 format 2023-03-31T00:39:46.917Z	Not Required
"specversion": "1.0",	Version number	Required
"type": "Payment",	Free form text	Required
"source": "WxCC%20Desktop",	URI-reference (no spaces allowed must use hex %20)	Required
"identity": "12263762555",	Number used to identify stream (phone, email, customerId, UUID)	Required
"identitytype": "phone",	Categorize identity (phone, email, customerId, temporaryId)	Required
"datacontenttype": "application/json",		
"data": {		
"taskid": "",	Can be added to auto collapse multi-widgets	Not Required
"origin": "Bill Payment",	Used for Widget Title	Required
"channelType": "Webex",	Widget subtitle	Required
"agent": "Agent1 Demo",	Any additional Fields you want to add	Not Required
"channelBreakout": "SMS"	Any additional Fields you want to add	Not Required
}		
}		

Inspired by the CloudEvent Spec

<https://github.com/cloudevents/spec/blob/v1.0.2/cloudevents/spec.md>

temporaryId Events to track unknown users

Website Visit

```
{  "id": "{{ $guid }}",
  "specversion": "1.0",
  "type": "webvisit",
  "source": "home.com:website",
  "identity": "954fcd6c-d-047ddb909",
  "identitytype": "temporaryId",
  "datacontenttype": "application/json",
  "data": {
    "notes": "Website Visit",
    "origin": "Website",
    "channelType": "NewVisit",
    "channelBreakout": "Web"  }
```

Need to keep track of identity
New type: identitytype as
temporaryId

Website Review Mortgage Rates

```
{  "id": "{{ $guid }}",
  "specversion": "1.0",
  "type": "webvisitMortgage",
  "source": "home.com:web:mortgage",
  "identity": "954fcd6c-d-047ddb9029",
  "identitytype": "temporaryId",
  "datacontenttype": "application/json",
  "data": {
    "notes": "Website Mortgage Rates",
    "origin": "Mortgage Rates",
    "channelType": "MortgageRates",
    "channelBreakout": "Web"  }
```

Need to keep track of identity
New type: identitytype as
temporaryId

Log into Website

```
{  "id": "{{ $guid }}",
  "specversion": "1.0",
  "type": "webvisitLogin",
  "source": "home.com:login",
  "previousidentity": "954fcd6c-d-047ddb909",
  "identity": "bmclella@cisco.com",
  "identitytype": "email",
  "datacontenttype": "application/json",
  "data": {
    "notes": "Website Login",
    "origin": "Website Login",
    "channelType": "LoginUser",
    "channelBreakout": "Web"  }
```

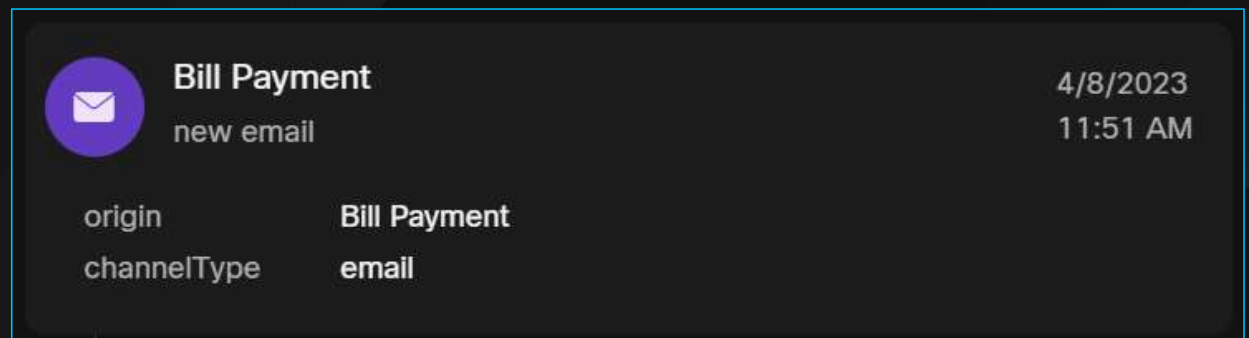
Once use is recognized next post
must contain the previous key
(identity) which will link the 2
previous journeys

Basic Entry required to insert into JDS

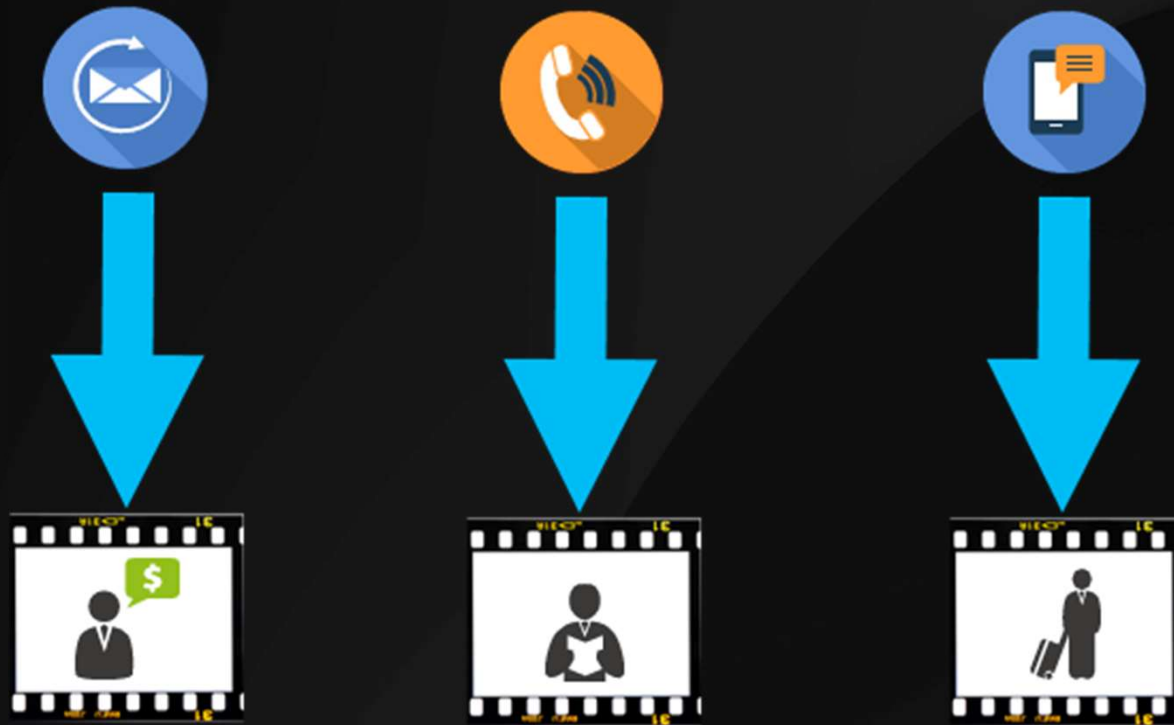
HEADER:

Content-Type application/json
 Authorization [SharedAccessSignature](#) so=cctsa&sn=sandbox&ss=ds (datasink-write-token)

```
{
  "id": "27fcb3bc-0e7d-47c8-9bca-639daa326c76",
  "specversion": "1.0",
  "type": "Payment",
  "source": "Desktop",
  "identity": "12263762555",
  "identitytype": "phone",
  "datacontenttype": "application/json",
  "data": {
    "origin": "Bill Payment",
    "channelType": "Voice"
  }
}
```



Events inserted will automatically create identities



Events from different sources are not linked automatically

Manual Process to manage identities

- Create identity or find an existing alias
- Merge Identities
- Add Aliases
- See video for an example and demonstration
<https://app.vidcast.io/share/f6ff72c4-9091-4508-aa89-65c070dfcd93>

<https://developer.webex-cx.com/documentation/journey>

Or with a little orchestration...

- Create a secure webhook for a Webex Connect flow
- Call Webhook with customer details for association
- Webhook Process
 - Create identity or find an existing alias
 - Merge Identities
 - Add Aliases
- Sample files: JDS Merge Identities and Add Aliases Webex Connect
- See video for an example and demonstration
<https://app.vidcast.io/share/03894bf5-1206-47d8-be1b-bda3c6d07ec3>

Event Feed Filtering to alter flow

- Filter based on Source in the URL

historic/12263762555?filter=type eq 'Payment'

- Filter based on Data in the URL

historic/12263762555?data=category eq 'Water'

- Combine Filters

.../filter=source eq 'Payment'&data=category eq 'Water'

- Return last X

historic/12263762555?top=1

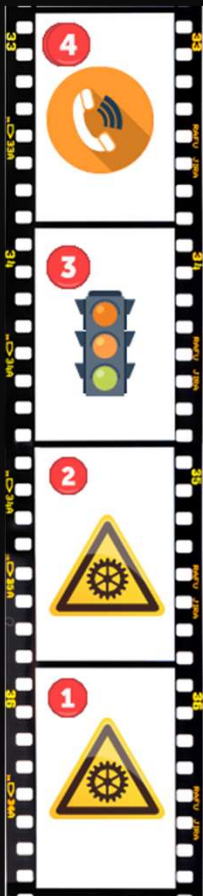
- See video for an example and demonstration

<https://app.vidcast.io/share/3fa5ceae-27cb-4698-ad72-9f65c3db3753>

```
{ "id": "{{ $guid }}",
  "specversion": "1.0",
  "type": "Payment",
  "source": "SMS%20Notification",
  "identity": "+12263762555",
  "identitytype": "phone",
  "datacontenttype": "application/json",
  "data": {
    "phone": "+12263762555",
    "Reason": "Late Payment Notification Sent - Notification",
    "MessageForUser" : "We have noticed your Waterbill payment is overdue. The
    payment of $544.73 was due Thursday, March 23rd.",
    "LinkSent": "https://buy.stripe.com/test_4gw8AnflUblBcbC8wB",
    "Notification": "1",
    "Past Due Amount": "$544.73",
    "category": "Water",
    "status": "overdue",
    "origin": "Past Due - 1st Notification",
    "channelType": "Payment Notify",
    "channelBreakout": "sms"
  }
}
```

<https://jds-us1.cjaas.cisco.com/v1/journey/streams/historic/{{encodedPerson}}?filter=type eq 'Payment'&data=category eq 'Water'&top=1>

Event Feed Filtering to alter flow



2023-04-02 23:16
source: Voice
category: Watermain
status: Customer Issue

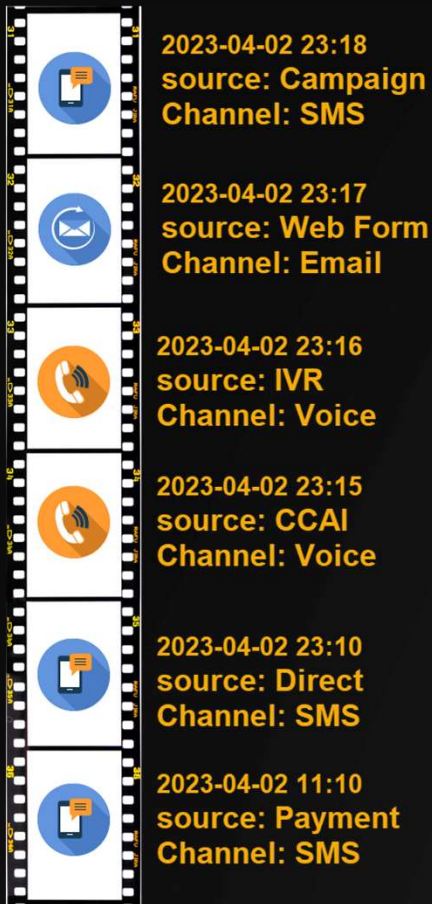
2023-04-02 23:15
source: Outage
category: Watermain
status: Fixed

2023-04-03 12:01
source: Outage
category: Watermain
status: Notification

2023-04-02 11:10
source: Outage
category: Watermain
status: Information

1. Customer sent information about up-and-coming watermain work
2. Customer notification about watermain work to be done
3. Customer notified watermain has been fixed
4. Customer calls into IVR less than a day later.
** Most likely due to watermain outage **

Progressive Profile

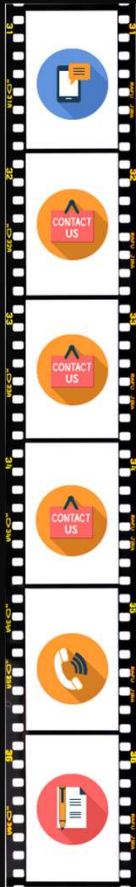


webex

- Creates rolling data based on a series of events when applied against an Identity/Alias
- Used in a flow to consider conditional routing
- Examples:
 - How many times have they called into our queues today?
 - Has the customer ever used email before?
 - Has the customer tried out our voice bots before?
 - Have we sent them a campaign SMS in the last 5 days?

<https://jds-us1.cjaas.cisco.com/v1/journey/views?personId=12263762555&templateId=EmailCheck>

Journey Actions



2023-04-02 23:18
source: SMS
category: Call back

2023-04-02 23:17
source: Voice
category: Closed

2023-04-02 23:16
source: Voice
category: Closed

2023-04-02 23:15
source: Voice
category: Closed

2023-04-02 23:10
source: Voice
category: No Agents

2023-04-02 11:10
source: Payment
category: Late

- Auto execute a rules-based action based off of values calculated in a progressive profile
- Examples:
 - Customer received 3 late payment notifications...
 - Customer has called and abandoned 10 times in 1 day...
 - Customer has added 3 items into their shopping cart....

Progressive Profile and Journey Actions

- Supported Data Types
 - string, integer, double, utc_datetime
- Create aggregations based on conditions
 - Value, Count, Sum, Max, Min, Average, Distinct
- Loop back period
 - days, hours, minutes
- Conditional branch AND / OR Logic
- Logical operators used for decisioning
 - EQ, NEQ, GTE, GT, LTE, LET

Customer Journey Widget

Customer Journey Widget

Lookup User

Profile

☐

No of times contacted in the last 24 hours	3
No of times contacted in the last 7 days	60
No of times contacted in the last 7 days via email	2
No of times contacted in the last 7 days via telephony	55

Aliases ⓘ

Email	bmclella@cisco.com	<input type="button" value="🗑"/>
Phone	18863762888	<input type="button" value="🗑"/>
Phone	12263762555	<input type="button" value="🗑"/>
Phone	+12263762555	<input type="button" value="🗑"/>

Journey

☐ Livestream

Bill Payment

new email

+1 (516) 517-9309

3 events

Voice

Callback

SendNotes

Notes

OTP

OTP-pass

- Lookup any user in the event an identity is not found
- Progressive Profile with counts
- Aliases associated to this identity
- Journey Filtering
- Livestream Refresh
- Expandable Journey Events
- Requires WxCC Adapter for events

Coding Language

- Widget uses lit-element JS framework for development
- lit-element is a lightweight library for web components
- Responsive and fast with lit-element
- CSS used for widget's styling and layout
- Widget Source Code:
<https://github.com/CiscoDevNet/cjaas-widgets>

Custom Icons

38



- All icons that can be used are located at <https://momentum.design/icons>
 - Click the icon and copy the CSS Class Name. We've found size 16 looks the best.
- Host your icon file on any accessible site.
 - Ensure you are handling any CORS configurations with your cloud provider.
- ChannelType in your event feed must equal the name you give your icon.
- Pick from 18 different colors:
blue, green, black, olive, cyan, violet, red, gray, orange, lime, cobalt, purple, yellow, gold, mint, slate, pink

ChannelType	->	"Telephony": {
CSS Class Name	->	"name": "icon-handset-active_16",
Color	->	"color": "green"
		}

Sample file: Sample icon file - icons

See video for an example and demonstration

<https://app.vidcast.io/share/bf3db261-29cb-436d-a983-d7e1a2eeb373>

JSON Layout

- See XXXX for full sample layout to be added into your desktop config

```
"icon-data-path": "https://wxcc-dem.glitch.me/icons.json",  
"logs-on": "true",  
"limit": "50",  
"time-frame": "All",  
"live-stream": "true"  
"collapse-timeline-section": "true",  
"collapse-profile-section": "true",  
"collapse-alias-section": "true",
```

Hard coding search string

Change the line

```
"interactionData": "$STORE.agentContact.taskSelected"
```

To what you would like to search one:

```
"customer" : "XXXXXXX",
```

Sample File: Agent Desktop json configs - jsonJDS

See video for an example and demonstration

<https://app.vidcast.io/share/4565c92e-e7d0-44e0-ab22-8c20990cf4c3>

Getting Started

- To get started with JDS

<https://developer.webex-cx.com/documentation/guides/journey---getting-started>

- This will give you a link to a smartsheet you need to fill
- Location of the latest desktop widget
- Authentication for JDS API's
- JDS SAS tokens and how to create them
- Event Schemas

Use Cases Deep Dive

Directing Customer Input

Customer – Has received a late payment notification but has forgotten.

Use Cases for Today's discussion:

1. **Directing Customer Input** – Last minute customers who remember it's time to pay a bill, don't need to wait for menus to take them to their options. Sometimes businesses need to take control as soon as the customer calls.
2. **Repeat Customer** - Callers repeatedly hitting the queues are most likely very frustrated and in a hurry to talk with an agent.
3. **Saving the Customer** – Things happen in call centers. Late callers, close to end of shift and eager to leave agents could leave a caller with no where to go other than to bed, mad!

Use Case 1 – Directing Customer Input

1. **Directing Customer Input** – Last minute customers who remember it's time to pay a bill don't need to wait for menu's to take them to their options. Sometimes businesses need to take control as soon as the customer calls.

Postman Examples:

- | | |
|---------------------------------------|---|
| UC1 - Payment Past Due Notify 1 | - Set up the late payment |
| UC1 - Check Payment Past Due Notify 1 | - Query JDS and look for event type and category |
| UC1 - Payment Past Due Notify 2 | - 2 nd payment reminder |
| UC1 - Payment Made | - Payment made, close JDS loop so no more inquiries |

Sample files: Flowbuilder Import - Voice_JDS_1

See video for an example and demonstration

<https://app.vidcast.io/share/3f1153c4-199f-4b6a-896d-c7f1d0faf973>

Directing Customer Input

Forgetful customer

- Customer forgets about a payment notification
- Days later, customer calls IVR system for an unrelated transaction
- IVR system reminds customer of overdue payment
- Customer quickly takes care of payment to avoid fees/damage to credit score
- Proactive approach shows company's commitment to customer's financial well-being

Directing Customer Input

Forgetful customer continued

- Busy customer forgets about a payment notification sent earlier
- Days later, the customer calls into an IVR system for an unrelated transaction
- The IVR system reminds the customer of the overdue payment
- ~~• Customer can quickly take care of the payment and avoid late fees or damage to their credit score~~
- Customer ignores payment options and continues
- Busy customer forgets about a payment notification sent AGAIN
- Days later, the customer calls into an IVR system
- IVR routes customer directly to agent
- Proactive approach saves company money by being on top of late payments

Use Case 2 – Repeat Caller

Call and hang up, call and hang up

- Busy customer calls and waits in queue, then hangs up
- Call multiple times throughout the day
- Upon 6th callback, offer them a scheduled courtesy callback
 - Or route differently based on the amount of times they have called

Postman Examples:

- UC2 - Create Profile Template Repeat Caller - Create Progressive Profile
- UC2 - Get All Profile View Templates - Get profile view
- UC2 - POST Queued Caller Info - Used script example, but this is the base
- UC2 - Progressive Profile View Repeat Caller - Get the values being accumulated

Sample files: Flowbuilder Import - Voice_JDS_2

See video for an example and demonstration

<https://app.vidcast.io/share/0f1ae863-ef34-4a4f-b53f-07cb321894c3>

Use Case 3 – Saving the Customer

Late night caller

- Busy customer forgets about a payment notification sent earlier
- Customer calls into IVR close to closing time and waits in queue
- All the agent's logoff for the night abandoning the call
- Customer calls back and greeted with a closed message
- Instantly receives SMS messaging asking them if they can call them first thing in the morning, or near a better time of their preference

Use Case 3 – Saving the Customer – Continued

Late night caller

Postman Examples:

- | | |
|---|------------------------------------|
| UC3 - Create Profile Template Closed Caller | – Create Progressive Profile |
| UC3 - Progressive Profile View Closed Caller | – Get profile view |
| UC3 - Create Journey Actions For Closed Callers | – Action Rules required to execute |
| UC3 - Get Journey Actions For Closed Callers | – View the rules |
| UC3 - POST Unstaffed Closed Caller Info | – Post Unstaffed Closed to JDS |
| UC3 - POST Closed Caller Info | – Post Unstaffed Closed to JDS |

**** Individual Event broadcasting coming soon****

See video for an example and demonstration

<https://app.vidcast.io/share/946a6f80-e7ba-4387-adc3-66249cac0d43>

Key Concepts and FAQs

Key Concepts & FAQs

Questions	Details
<i>What is the Cost of using JDS</i>	JDS will be available to all WXCC customers as part of the Standard agent SKU. There are NO additional cost.
<i>What data format does JDS use for storing Journey data?</i>	JDS is inspired by the CloudEvent Spec to store Data
<i>What is an event?</i>	Events are the smallest unit of data processed. They describe what occurred - where and by whom
<i>How do we identify new customers? How do we create a progressive customer profile?</i>	<p>Data is tracked on tape against a known/unique temporaryId, which can be anything that is unique, validated, and common across the sources.</p> <p>Fresh or Anonymous customers not yet known can also be posted using UUIDv4 as placeholder ID, which is held on to but not posted on-to tape until these profiles become known within a 24hrs span, so that the complete journey can be stitched that includes events before they identified themselves, so no interactions falls between the cracks (ex: a fresh customer calling in who later authenticates via CRM Dip, a fresh visitor to a website adding a item to cart later registers to checkout)</p> <p>Two or more PersonIDs can be merged by API where newer events will be logged on the newest ID.</p>

Key Concepts & FAQs

Questions	Details
<i>Are there any Day 0/Day 1 Operation before using JDS</i>	Yes, we recommend customers to upload their known set of end-user details (Name, Phone, Email etc) to ensure we track all the events across all channels with one customer profile.
<i>Is JDS API only service?</i>	Although JDS is API-first, we do have a UI component (JDS admin portal) for admin to leverage this low code/no-code platform to implement their use cases. As part of our GA release, we will have all the JDS APIs available and JDS admin portal (with limited functionality) available. JDS admin portal will evolve to include more functionalities in the upcoming quarters
<i>What are different API types within JDS</i>	All JDS APIs are listed on the CC developer portal. JDS APIs can be broadly classified into 4 buckets – Listen, Identify, Analyze & Act.
<i>Do we have pre-build connectors?</i>	<p>We currently have pre-build connector within the JDS admin portal for WxCC and will have connectors for other cisco service in our upcoming releases. We are exploring the option to enable Partners to bring their own connectors that meets their specific customer need.</p> <p>Connectors are a low code/no code avenue for JDS to ingest data without leveraging our APIs. However, with our currently available APIs, we can ingest data from ANY application in ANY format</p>
<i>Does JDS have widget? Can it be customized?</i>	Yes, we have a widget that will be available out-of-the-box with the WxCC agent desktop before end of Q4 FY23. However, we have also open sourced the code of this widget for Partners/Customers to customize & use this widget as part of their custom desktop layout.

Key Concepts & FAQs

Questions	Details
<i>Will widget only accept ANI as a search parameter</i>	Widget today will only accept ANI as a parameter, but update will happen to allow passing a specific CAD variable for the search query.
<i>Why do I need to use channelType for my custom icon</i>	This allows for you to use filters in the widget. A custom icon name can still contain the channel type, i.e. "Voice – Write Notes", or you can simply make a new field in the data output "Channel": "Voice"
<i>Hide Alias field</i>	New widget will not have this displayed by default. Code is provided so you can remove this if not required.
<i>On Callbacks we see the wrong ANI sent into the search bar.</i>	This process will be changed to use a CAD variable instead of the ANI when it is invoked.
<i>How do I update First/Last Name fields</i>	These fields can be updated using the https://developer.webex-cx.com/documentation/journey/v1/replace-all-aliases-of-an-individual API, however this requires completely rebuilding the entire profile. A simpler interface to update First/Lastname is on the roadmap
<i>Is there an API call that can update profiles and aliases in one shot</i>	In the slides above it is talked about using an orchestration tool to do this, however this has been added into the roadmap to perform a create/update/merge Identity and Alias in a single API call.

Key Concepts & FAQs

Questions	Details
<i>Why do I need to encode ANI's with a '+' sign for a voice call.</i>	URLs are composed of a limited set of characters, and some characters, such as the plus sign (+), have special meanings in URLs. Therefore, when a URL contains a plus sign, it must be properly encoded to avoid any misinterpretation by web servers or browsers. URL encoding replaces special characters with a percentage sign followed by their ASCII code representation, which ensures that the URL is correctly interpreted and processed by web applications.
<i>Action Triggers to not pass the event ID that triggered them.</i>	Originally an Action trigger was developed to execute on a global event. This is an immediate item action on our roadmap to add this feature.
<i>How can I extract the data to review</i>	There currently isn't an OOB method to extract data, however you can use the history API's to pull out journey information for offline viewing.
<i>Widget format to add a phone number doesn't look right.</i>	The format example is not correct for the widget. Instead, users should add the alias with no spaces or brackets. +12224446666. A plus is required to add in the phone number this way, but not required if you manually add it.
<i>How do I modify or delete existing events.</i>	Events are an append-only, immutable data ledger. You cannot modify or delete events already in the data stream.

Key Concepts & FAQs

Questions	Details
<i>How long will you store data</i>	Data will be stored for one year. In the event data needs to be stored longer, there will be a charge for this.
<i>How much data can we store?</i>	Specifics will come out soon, however for now we will stick with a fair use policy.

Milestones & Timelines

Milestone & Timelines

Milestone	Details	Scope
GA (with WxCC) – US Only	FY23 (May 15 th)	JDS APIs, Admin Portal & Custom Widget
GA (with WxCC) – EMEA&AJPC		JDS APIs, Admin Portal & Out-of-Box Widget
GA (with WxCC) – Global		JDS APIs, Admin Portal & Out-of-Box Widget
EA (With WxCCE) – US Only		JDS APIs, Admin Portal & Custom Widget

Feedback/Questions



webex
by CISCO

Appendix

Important Resources

JDS APIs - <https://developer.webex-cx.com/documentation/journey>

JDS Overview Vidcast - <https://app.vidcast.io/share/889c2cbf-51b2-4cc9-94f8-9143078dca83>

JDS Use Case and example Vidcast

- <https://app.vidcast.io/collections/share/a3d5d3c9-faf8-4773-a16d-1abc27361113>

CloudEvent Spec Details - <https://github.com/cloudevents/spec/blob/v1.0.2/cloudevents/spec.md>

JDS Widget GitHub Link - <https://github.com/CiscoDevNet/cjaas-widgets>

Cisco Momentum Design (icons) - <https://momentum.design/icons>

Handy Resources

Javascript Tester- <https://webtoolkitonline.com/javascript-tester.html>

JSON Beautifier- <https://jsonbeautifier.org/>

JSON UnEscape - <https://www.freeformatter.com/json-escape.html#before-output>

JSONtoString Format to 1 line - <https://jsontostring.com/convert-json-to-one-line>

Test Database - <https://mockapi.io/projects>

Webhook Testing - <https://webhook.site/>

GraphQL to JSON - <https://datafetcher.com/graphql-json-body-converter> or
<https://insomnia.rest/> (free for one cloud project)