

AWS Canvas Connector Installation Instructions

September 2022

Introduction

This slide deck will walk you through installing and configuring the AWS Canvas Connector. The instructions are broken down into three sections:

- Installing the Canvas Live Events Connector
- Installing the Canvas API Amazon AppFlow Connector
- Installing the AWS Glue and Glue Databrew resources

Download Template Files

Before you begin, you will need to download the configuration files by cloning Unicon's GitHub **AWS.CanvasLiveEventsConnector** repository:

<https://github.com/Unicon/AWS.CanvasLiveEventsConnector>

to your local environment.

Please see the README for further information on the project goals and components.

Installing the AWS Canvas Live Events Connector

Step 1 - Download the CloudFormation template

1. Use the AWS Canvas Live Events Connector CloudFormation template: **canvas-live-events.cfn.yaml**
2. If you wish to make any changes to the CloudFormation template (e.g., change resource names to match your naming conventions) do that at this time.
3. Log in to the AWS Management Console. Note, the user you are logging in with must have sufficient privileges to create CloudFormation stacks and the AWS resources needed for the Canvas Live Events Connector (list of resources below).
4. Navigate to CloudFormation

The CloudFormation stack will create the following resources:

- One Lambda function (CanvasLiveEventsLambda)
- One IAM role (CanvasLiveEventsLambdaRole)
- One Amazon S3 bucket (\${Organization}-\${Project}-canvas-events)
- One SQS queue (canvas-live-events-\${Project}-\${Organization})

Step 2 - Create Stack

1. Click on the Create stack button

The screenshot shows the AWS CloudFormation console interface. At the top, the text 'Management & Governance' is visible. The main heading is 'AWS CloudFormation' followed by 'Model and provision all your cloud infrastructure'. Below this, a paragraph states: 'AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.' To the right, a white box titled 'Create a CloudFormation stack' contains the text 'Use your own template or a sample template to quickly get started.' and an orange 'Create stack' button. A red arrow points from the 'Create stack' button in the white box to the 'Create stack' button in the orange box. Below the main heading, there is a section titled 'How it works' which features a video player with the title 'Simplify Your Infrastructure Management Using AWS CloudFormation' and a 'Copy link' button. At the bottom of the video player, it says 'Watch on YouTube'. To the right of the video player, there are two sections: 'Getting started' with links for 'What is AWS CloudFormation', 'Getting started with CloudFormation', 'Learn template basics', and 'Quick starts'; and 'More resources' with links for 'Documentation', 'API reference', and 'FAQs'.

Management & Governance

AWS CloudFormation

Model and provision all your cloud infrastructure

AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.

Create a CloudFormation stack

Use your own template or a sample template to quickly get started.

Create stack

How it works

Simplify Your Infrastructure Management Using AWS CloudFormation

Copy link

AWS Management and Governance

Watch on **YouTube**

Getting started

- [What is AWS CloudFormation](#)
- [Getting started with CloudFormation](#)
- [Learn template basics](#)
- [Quick starts](#)

More resources

- [Documentation](#)
- [API reference](#)
- [FAQs](#)

Step 3 - Create stack

1. Select "Template is ready"
2. Select "Upload a template file"
3. Click Choose file and then choose the yaml file that you downloaded at Step 1
4. Click Next

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file

No file chosen

JSON or YAML formatted file

S3 URL: Will be generated when template file is uploaded

View in Designer

Next

Step 4 - Specify stack details

The Organization and Project parameters are used as substitution variables in AWS resource names (e.g., Amazon S3 bucket - calpoly-qa-canvas-events). Organization should be a short abbreviation for the institution. Project should be a short abbreviation for the initiative or environment. All AWS resources are also tagged with the Project value.

1. Enter a Stack name (e.g., CanvasLiveEventsStack)
2. Enter a value for the Organization parameter (e.g., calpoly)
3. Enter a value for the Project parameter (e.g., qa)
4. Click Next

Specify stack details

Stack name

Stack name

Enter a stack name

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.

Organization

unicon

Project

devers

Cancel

Prev

Next

Step 5 - Configure stack options

No changes are required on this screen. However, if you have your own AWS best practices or conventions feel free to apply them now.

If you do not select IAM role to use, CloudFormation will create one for you.

1. Click Next

Configure stack options

Tags

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](#)

Key	Value	Remove
Add tag		

Permissions

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](#)

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

IAM role name	Sample-role-name	Remove
---------------	------------------	--------

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. [Learn more](#)

☒ Roll back all stack resources

Roll back the stack to the last known stable state.

☐ Preserve successfully provisioned resources

Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Advanced options

You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

► Stack policy

Defines the resources that you want to protect from unintentional updates during a stack update.

► Rollback configuration

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. [Learn more](#)

► Notification options

► Stack creation options

Cancel



Next

Step 6 - Review stack

Review the stack configuration.

1. Click the checkbox (I acknowledge that AWS CloudFormation might create IAM resources with custom names.) in the blue box
2. Click Create stack

This will initiate the process of creating all of the required resources. This process can take a couple of minutes.

Create stack

Review ConsumeLiveEventsStack

Step 1: Specify template

Template

Template URL:
https://aws-est-2.amazonaws.com/cloudformation-us-east-2/2022/10/04/cfn-console-live-events-04.yaml
Stack description:
-
[Learn more](#)

Step 2: Specify stack details

Parameters (2)

Key	▲	Value	▼
Organization		amazon	
Project		blueskywork	

Step 3: Configure stack options

Tags (0)

Key	▲	Value	▼
No tags			
There are no tags defined for this stack.			

Permissions

No permissions

There is no IAM role associated with this stack.

Stack failure options

Rollback on failure:
Enabled

Stack policy

No stack policy

There is no stack policy defined.

Rollback configuration

Handling time:
-
CloudWatch alarm ARN:
-

Notification options

No notification options

There are no notification options defined.

Stack creation options

Timeout:
-
Termination protection:
Disabled

Quick-create link

Capabilities

☒ I acknowledge that AWS CloudFormation might create IAM resources with custom names.

Cancel

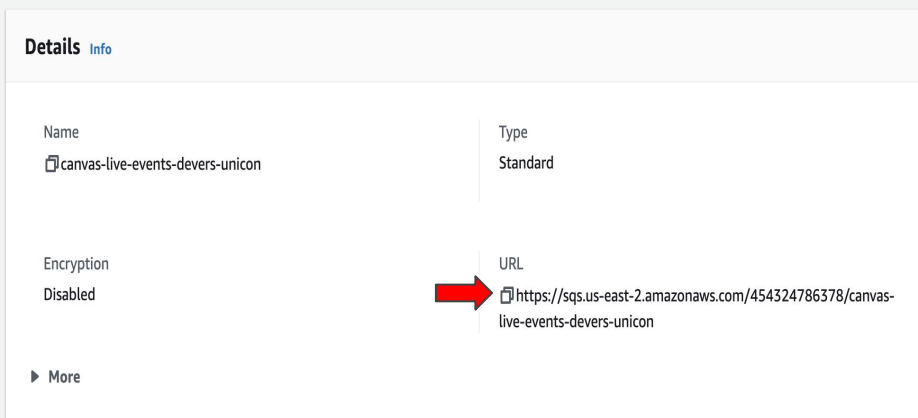
Previous

Create Change Set

Create Stack

Step 7 - Setup AWS Canvas Live Events

1. In the AWS Management Console, navigate to SQS. Click on the link for the queue that was just created and copy the URL value. You will need this URL when configuring the Live Events in Canvas.
2. Open this link <https://community.canvaslms.com/t5/Admin-Guide/How-do-I-subscribe-to-Live-Events-using-Canvas-Data-Services/ta-p/227> and follow the instructions to configure Live Events.
 - a. For Delivery Method select SQS
 - b. For URL, enter the value you copied in step 1
 - c. For Authentication select None (by default the queue is locked down to only Instructure and your AWS account); if you prefer to use AWS Credentials you can
 - d. For Message Type select Caliper 1.1
 - e. For Application Type select Data Streaming
 - f. Check the Subscription box (this will send all messages, if you would prefer only certain messages, select those now)
3. Click Save & Exit



After completing these steps event data will be flowing into Amazon S3. To verify, navigate to S3 in the AWS management console, click on the bucket that ends with -canvas-events. You should see json files partitioned by date. If you don't see any events, log in and log out of Canvas, wait a minute or so and check S3 again.

Installing the Canvas API Amazon AppFlow Connector

Step 1 - Download the CloudFormation template

1. Use the Canvas API Connector CloudFormation template: **canvas-api-lambda.cfn.yaml**
2. If you wish to make any changes to the CloudFormation template (e.g., change resource names to match your naming conventions) do that at this time.
3. Log in to the AWS Management Console. Note, the user you are logging in with must have sufficient privileges to create CloudFormation stacks and the AWS resources needed for the Canvas API Connector (list of resources below).
4. Navigate to CloudFormation

The CloudFormation stack will create the following resources:

- Three Amazon S3 buckets
 - `${Organization}-${Project}-canvas-assignments`
 - `${Organization}-${Project}-canvas-courses`
 - `${Organization}-${Project}-canvas-users`
- One Lambda function (CanvasApiAppFlowLambda)
- One IAM role

Step 2 - Create Stack

1. Click on the Create stack button

The screenshot shows the AWS CloudFormation console's 'Create a CloudFormation stack' page. At the top, the text 'Management & Governance' is visible. The main heading is 'AWS CloudFormation' followed by 'Model and provision all your cloud infrastructure'. Below this, a paragraph states: 'AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.' To the right, a white box contains the title 'Create a CloudFormation stack', a subtext 'Use your own template or a sample template to quickly get started.', and an orange 'Create stack' button. A red arrow points from the text 'cloud infrastructure' to this button. Below the main content, there is a 'How it works' section featuring a video player with the title 'Simplify Your Infrastructure Management Using AWS CloudFormation' and a 'Watch on YouTube' button. To the right of the video player is a 'Copy link' button. On the far right, there are two sections: 'Getting started' with links for 'What is AWS CloudFormation', 'Getting started with CloudFormation', 'Learn template basics', and 'Quick starts'; and 'More resources' with links for 'Documentation', 'API reference', and 'FAQs'.

Management & Governance

AWS CloudFormation

Model and provision all your cloud infrastructure

AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.

Create a CloudFormation stack

Use your own template or a sample template to quickly get started.

Create stack

How it works

Simplify Your Infrastructure Management Using AWS CloudFormation

Watch on YouTube

Copy link

Getting started

- [What is AWS CloudFormation](#)
- [Getting started with CloudFormation](#)
- [Learn template basics](#)
- [Quick starts](#)

More resources

- [Documentation](#)
- [API reference](#)
- [FAQs](#)

Step 3 - Create stack

1. Select "Template is ready"
2. Select "Upload a template file"
3. Click Choose file and then choose the yaml file that you downloaded at Step 1
4. Click Next

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file

No file chosen

JSON or YAML formatted file

S3 URL: Will be generated when template file is uploaded

View in Designer

Next

Step 4 - Specify stack details

The Organization and Project parameters are used as substitution variables in AWS resource names (e.g., Amazon S3 bucket - calpoly-qa-canvas-events). Organization should be a short abbreviation for the institution. Project should be a short abbreviation for the initiative or environment. All AWS resources are also tagged with the Project value.

1. Enter a Stack name (e.g., CanvasAppFlowStack)
2. Enter a value for the Organization parameter (e.g., calpoly)
3. Enter a value for the Project parameter (e.g., qa)
4. Click Next

Specify stack details

Stack name

Stack name

Enter a stack name

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.

Organization

unicon

Project

devers

Cancel

Prev

Next

Step 5 - Configure stack options

No changes are required on this screen. However, if you have your own AWS best practices or conventions feel free to apply them now.

If you do not select IAM role to use, CloudFormation will create one for you.

1. Click Next

Configure stack options

Tags

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](#)

Key	Value	Remove
Add tag		

Permissions

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](#)

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

IAM role name	Sample-role-name	Remove
---------------	------------------	--------

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. [Learn more](#)

☒ Roll back all stack resources

Roll back the stack to the last known stable state.

☐ Preserve successfully provisioned resources

Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Advanced options

You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

► Stack policy

Defines the resources that you want to protect from unintentional updates during a stack update.

► Rollback configuration

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. [Learn more](#)

► Notification options

► Stack creation options

Cancel



Next

Step 6 - Review stack

Review the stack configuration.

1. Click the checkbox (I acknowledge that AWS CloudFormation might create IAM resources with custom names.) in the blue box
2. Click Create stack

This will initiate the process of creating all of the required resources. This process can take a couple of minutes.

Review CnveEventsStack Stack

Step 1: Specify template Exit

Template

Template ID:
[aws-elasticbeanstalk-ec2-with-elasticloadbalancing-us-east-2-20221108/aws-elasticbeanstalk-events-stack-joint](#)

Stack description
 +
[Estimate cost](#)

Step 2: Specify stack details Exit

Parameters (2)

Key	Value
Organization	amazon
Project	Microservices

Step 3: Configure stack options Exit

Tags (0)

Key	Value
No tags	

There are no tags defined for this stack.

Permissions

No permissions

There is no IAM role associated with this stack.

Stack failure actions

Rollback on failure
 Disabled

Stack policy

No stack policy

There is no stack policy defined.

Rollback configuration

Monitoring time
 -

CloudWatch alarms alert
 -

Notification options

No notification options

There are no notification options defined.

Stack creation options

Timeout
 -

Termination protection
 Disabled

+ Quick-create link

Capabilities

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

This template requests identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. Learn more

☐ I acknowledge that AWS CloudFormation might create IAM resources with custom names.

Cancel Previous Create Change Set **Create Stack**

Canvas Access Token

You will need a long lived Canvas API Access token in order to use the Canvas API with Amazon AppFlow. The easiest way to get that token is to follow the steps listed here: <https://community.canvaslms.com/t5/Admin-Guide/How-do-I-manage-API-access-tokens-as-an-admin/ta-p/89>

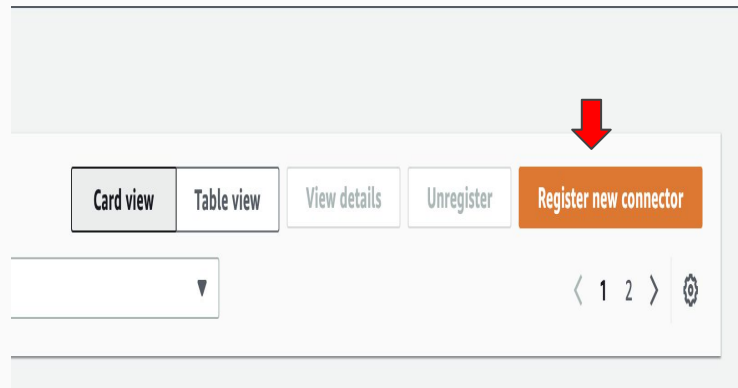
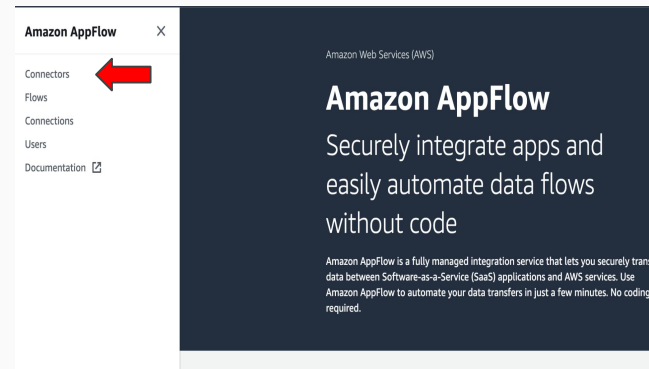
A couple of caveats:

- If you want full access to the data for a Canvas account the user that you are generating the token for must have Canvas Admin privileges
- It is likely that we will change this in the future to use a more traditional OAuth2 approach

Step 1 - Create the Amazon AppFlow Connector

Unfortunately, due to the fact that Amazon AppFlow is a newer service, not everything is available via CloudFormation.

1. In the AWS Console, navigate to Amazon AppFlow
2. Click on Connectors in the left navigation pane
3. Click on Register new connector



Step 1 cont. - Create AppFlow Connector

1. Select Lambda function: CanvasAppFlowLambda
2. Give the Connector a name (e.g., CanvasAppFlowConnector)
3. Optionally add any tags that you want to the Connector
4. Click the Register button

Register a new connector [X]

Amazon AppFlow will add resource-based policies to the Lambda function on your behalf, so that AppFlow can invoke the function when executing flows that use your custom connector.

Provisioning type
Lambda

Lambda function
A Lambda function is needed to implement Amazon AppFlow connector builder SDK.
CanvasAppFlowLambda [Refresh]

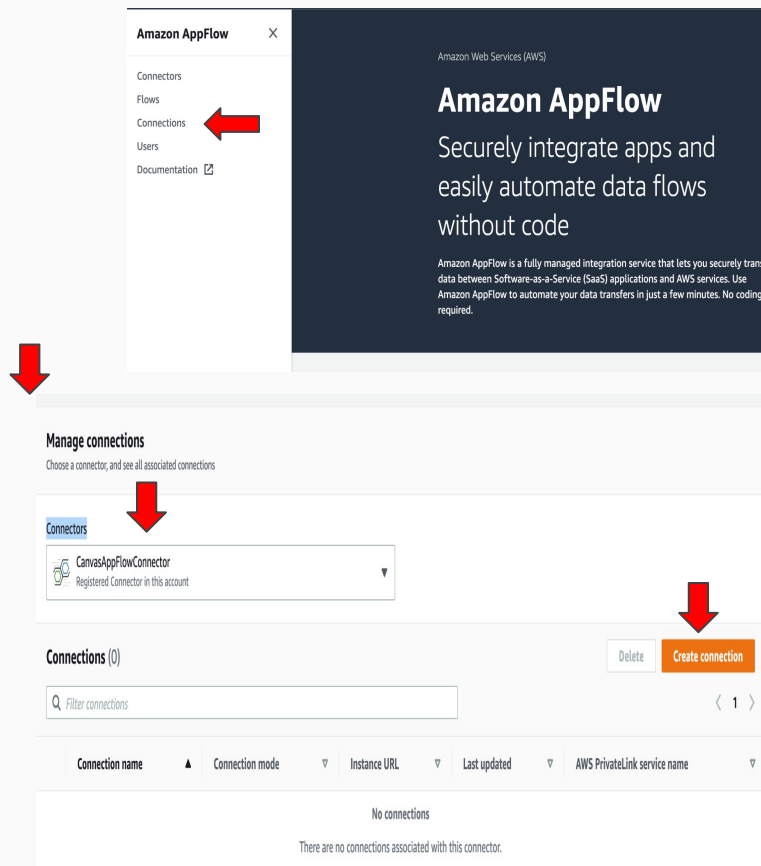
Connector label
Specify a new connector label.
CanvasAppFlowConnector

Tags - optional
Choose key-value pairs to tag your connector.
No tags associated with this connector.
[Add new tag]

[Cancel] [Register]

Step 2 - Create AppFlow Connection

1. In the AWS Console, navigate to Amazon AppFlow
2. Click on Connections in the left navigation pane
3. Click on Register new connector
4. From the Connectors dropdown, select CanvasAppFlowConnector (or whatever you named the connector in step 1)
5. Click Create connection



Step 2 cont. - Create the Amazon AppFlow Connection

1. In the Canvas OAuth Token field enter the Access token that you created in Canvas
2. In the Canvas Base URL field enter the URL of your Canvas instance in http(s)://domain/ format (e.g., https://unicon.instructure.com/)
3. You can safely ignore Data encryption (configuration is encrypted by default) if you choose.
4. Enter a Connection name (e.g., CanvasAppFlowConnection)
5. Click the Connect button

Connect to CanvasAppFlowConnector

Allow Amazon AppFlow to access your CanvasAppFlowConnector account.

Custom authentication inputs

Canvas OAuth Token

Canvas Base URL
Required String field: (e.g., https://unicon.instructure.com/)

Data encryption

Your data is encrypted by default with a key that AWS owns and manages for you. To choose a different key, customize your encryption settings.

☐ Customize encryption settings (advanced)

Connection name

Specify a new connection name

Cancel Connect

Step 3 - Create User Flow


1. Click Flow in the left navigation pane
2. Click the Create flow button
3. Enter Flow name (e.g., CanvasUserFlow) - note the flow name is important, if you don't use CanvasUserFlow remember the value you enter as you will need to supply it to Cloudformation later
4. Data encryption can safely be ignored
5. Optionally add any tags
6. Click the Next button


The screenshot displays the Amazon AppFlow console interface. On the left, the navigation pane shows 'Flows' selected. The main content area has a top bar with a 'Create flow' button. Below this is a search bar and a table with columns: Flow name, Source, Destination, Trigger, Status, Date created, Last run, and Run status. The table is currently empty, showing 'No flows' and 'No flows to display.' Below the table is a 'Create flow' button. The 'Flow details' section contains a 'Flow name' input field and a 'Flow description - optional' text area. The 'Data encryption' section has a checkbox for 'Customize encryption settings (advanced)'. The 'Tags - optional' section has an 'Add new tag' button. At the bottom right, there are 'Cancel' and 'Next' buttons.


Step 3 - Create User Flow

1. For Source name, select your Canvas Connector (e.g., CanvasAppFlowConnector)
2. Select the appropriate connection (e.g., CanvasAppFlowConnection)
3. Choose API Version 1.0
4. Choose Users object


Source details [Info](#)

Source name 


 **CanvasAppFlowConnector**
Registered Connector in this account ▼

Choose CanvasAppFlowConnector connection [Info](#) 

CanvasAppFlowConnector created: 1/5/2022 ▼

Choose API Version 

1.0 ▼

Choose CanvasAppFlowConnector object 

Users ▼

Step 3 - Create User Flow

1. For Destination name, choose Amazon S3
2. For Bucket details, select the Canvas users bucket (e.g., {org}-{project}-canvas-users); leave bucket prefix empty
3. Expand Advanced settings
4. Leave all of the defaults except Aggregate all records
5. Scroll to the bottom of the page
6. Leave Run on demand as the Flow trigger (you can change this later)
7. Click the Next button

Destination details [Info](#)

Destination name

Amazon S3
Amazon Simple Storage Service (Amazon S3) is a service that provides object storage through a web service interface.

Bucket details

unicon-devers2-canvas-users

Your S3 URL is `s3://unicon-devers2-canvas-users/CanvasUserFlow`.

Additional settings

Data format preference [Info](#)
You can specify your preferred format for writing data to S3.

☒ **JSON format (default)**
Format your data in a text-based data exchange file that is human-readable and easy to parse.

☐ **CSV format**
Format your data in a plain text file that structures data in tabular form.

☐ **Parquet format**
Format your data in a flat columnar storage file that is optimized for fast data retrieval and used in AWS analytical applications.

Data transfer preference [Info](#)

☐ No aggregation

☒ **Aggregate all records**
You can combine all records into a single file per flow run.

Filename preference [Info](#)

☒ No timestamp

☐ Add a timestamp to the file name
Your file name in S3 will end with the creation date in YYYY-MM-DDTHH:mm:ss format.

Folder structure preference [Info](#)

☒ No timestamped folder

☐ Place the file in a timestamped folder
You can create a folder that is named according to the flow frequency and date.
Choose the naming convention for your folder
The date granularity that you choose will determine the name of your folder.

Step 3 - Create User Flow

1. Select Manually map fields as the Mapping method
2. For Source field name, choose Map all fields directly
3. Scroll to the bottom of the page
4. Click the Next button
5. On the Add filters page, just click the Next button
6. On the Review and create page, scroll to the bottom and click the Create Flow button

At this point, the Canvas User API flow is ready to run.

Map data fields [Info](#)

Mapping method

☒ Manually map fields
Select one or more source fields and map them to selected destination field.

CanvasAppFlowConnector

Source field name [Info](#)

Choose source fields

Q

Bulk actions

Map all fields directly

Source fields

27


Step 4 - Create Course Flow

1. Click Flow in the left navigation pane
2. Click the Create flow button
3. Enter Flow name (e.g., CanvasCourseFlow) - note the flow name is important, if you don't use CanvasCourseFlow remember the value you enter as you will need to supply it to Cloudformation later
4. Data encryption can safely be ignored
5. Optionally add any tags
6. Click the Next button


The screenshot displays the Amazon AppFlow console interface. On the left, the navigation pane shows 'Flows' highlighted with a red arrow. The main content area shows the 'Flows (0) Info' section with a 'Create flow' button in the top right, also indicated by a red arrow. Below this is a table with columns: Flow name, Source, Destination, Trigger, Status, Date created, Last run, and Run status. The table is empty, showing 'No flows' and 'No flows to display.' with a 'Create flow' button. Below the table is the 'Flow details' section, which includes a 'Flow name' input field with a red arrow pointing to it, a 'Flow description - optional' input field, a 'Data encryption' section with a 'Customize encryption settings (advanced)' checkbox, and a 'Tags - optional' section with an 'Add new tag' button. At the bottom right of the 'Flow details' section, there is a 'Next' button with a red arrow pointing to it, and a 'Cancel' button.


Step 4 - Create Course Flow


1. For Source name, select your Canvas Connector (e.g., CanvasAppFlowConnector)
2. Select the appropriate connection (e.g., CanvasAppFlowConnection)
3. Choose API Version 1.0
4. Choose Courses object




Source details [Info](#)

Source name 
CanvasAppFlowConnector
Registered Connector in this account

Choose CanvasAppFlowConnector connection [Info](#) 
CanvasAppFlowConnector created: 1/5/2022

Choose API Version 
1.0

Choose CanvasAppFlowConnector object 
Courses

Step 4 - Create Course Flow

1. For Destination name, choose Amazon S3
2. For Bucket details, select the Canvas courses bucket (e.g., {org}-{project}-canvas-courses); leave bucket prefix empty
3. Expand Advanced settings
4. Leave all of the defaults except Aggregate all records
5. Scroll to the bottom of the page
6. Leave Run on demand as the Flow trigger (you can change this later)
7. Click the Next button

Destination details [Info](#)

Destination name

Amazon S3
Amazon Simple Storage Service (Amazon S3) is a service that provides object storage through a web service interface.

Bucket details

unicon-devers2-canvas-courses

Enter bucket prefix - optional

Your S3 URL is s3://unicon-devers2-canvas-courses/Test.

Additional settings

Data format preference [Info](#)
You can specify your preferred format for writing data to S3.

☒ **JSON format (default)**
Format your data in a text-based data exchange file that is human-readable and easy to parse.

☐ **CSV format**
Format your data in a plain text file that structures data in tabular form.

☐ **Parquet format**
Format your data in a flat columnar storage file that is optimized for fast data retrieval and used in AWS analytical applications.

Data transfer preference [Info](#)

☐ No aggregation

☒ **Aggregate all records**
You can combine all records into a single file for the job flow run.

Filename preference [Info](#)

☒ **No timestamp**

☐ Add a timestamp to the file name
Your file name in S3 will end with the creation date in YYYY-MM-DDTHH:mm:ss format.

Folder structure preference [Info](#)

☒ **No timestamped folder**

☐ Place the file in a timestamped folder
You can create a folder that is named according to the flow frequency and date.
Choose the naming convention for your folder
The date granularity that you choose will determine the name of your folder.

Choose folder structure

Step 4 - Create Course Flow

1. Select Manually map fields as the Mapping method
2. For Source field name, choose Map all fields directly
3. Scroll to the bottom of the page
4. Click the Next button
5. On the Add filters page, just click the Next button
6. On the Review and create page, scroll to the bottom and click the Create flow button

At this point, the Canvas Course API flow is ready to run.

Map data fields [Info](#)

Mapping method



Manually map fields

Select one or more source fields and map them to selected destination field.

CanvasAppFlowConnector

Source field name [Info](#)



Choose source fields ▲



Bulk actions

Map all fields directly



Source fields

Step 5 - Create Assignment Flow

1. Click Flow in the left navigation pane
2. Click the Create flow button
3. Enter Flow name (e.g., CanvasAssignmentFlow)
- note the flow name is important, if you don't use CanvasAssignmentFlow remember the value you enter as you will need to supply it to Cloudformation later
4. Data encryption can safely be ignored
5. Optionally add any tags
6. Click the Next button

The screenshot displays the Amazon AppFlow console interface. On the left, the navigation pane shows 'Flows' selected. The main content area shows the 'Create flow' button. Below this, the 'Flow details' section is expanded, showing the 'Flow name' field. The 'Data encryption' section is also visible, along with the 'Tags - optional' section. At the bottom right, the 'Next' button is highlighted with a red arrow.

Step 5 - Create Assignment Flow

1. For Source name, select your Canvas Connector (e.g., CanvasAppFlowConnector)
2. Select the appropriate connection (e.g., CanvasAppFlowConnection)
3. Choose API Version 1.0
4. Choose Assignment object

Source details [Info](#)

Source name 

 CanvasAppFlowConnector
Registered Connector in this account ▼

Choose CanvasAppFlowConnector connection [Link](#) 

CanvasAppFlowConnector created: 1/5/2022 ▼

Choose API Version 

1.0 ▼

Choose CanvasAppFlowConnector object 

Assignments ▼

Step 5 - Create Assignment Flow

1. For Destination name, choose Amazon S3
2. For Bucket details, select the Canvas assignments bucket (e.g., {org}-{project}-canvas-assignments); leave bucket prefix empty
3. Expand Advanced settings
4. Leave all of the defaults except Aggregate all records
5. Scroll to the bottom of the page
6. Leave Run on demand as the Flow trigger (you can change this later)
7. Click the Next button

Destination details [Info](#)

Destination name



Amazon S3

Amazon Simple Storage Service (Amazon S3) is a service that provides object storage through a web service interface.

Bucket details

unicon-devers2-canvas-assignments

Enter bucket prefix - optional



Your S3 URL is `s3://unicon-devers2-canvas-assignments/Test`.

Additional settings

Data format preference [Info](#)

You can specify your preferred format for writing data to S3.

☒ JSON format (default)

Format your data in a text-based data exchange file that is human-readable and easy to parse.

☐ CSV format

Format your data in a plain text file that structures data in tabular form.

☐ Parquet format

Format your data in a flat columnar storage file that is optimized for fast data retrieval and used in AWS analytical applications.

Data transfer preference [Info](#)

☐ No aggregation

☒ Aggregate all records

You can combine all records into one file per flow run.

Filename preference [Info](#)

☒ No timestamp

☐ Add a timestamp to the file name

Your file name in S3 will end with the creation date in YYYY-MM-DDTHH:mm:ss format.

Folder structure preference [Info](#)

☒ No timestamped folder

☐ Place the file in a timestamped folder

You can create a folder that is named according to the flow frequency and date.

Choose the naming convention for your folder

The date granularity that you choose will determine the name of your folder.

Choose folder structure

Step 5 - Create Assignment Flow

1. Select Manually map fields as the Mapping method
2. For Source field name, choose Map all fields directly
3. Scroll to the bottom of the page
4. Click the Next button
5. On the Add filters page, just click the Next button
6. On the Review and create page, scroll to the bottom and click the Create flow button

At this point, the Canvas Assignment API flow is ready to run.

Map data fields [Info](#)

Mapping method

☒ **Manually map fields**
Select one or more source fields and map them to selected destination field.

CanvasAppFlowConnector

Source field name [Info](#)

Choose source fields

Q

Bulk actions

Map all fields directly

Source fields

Installing the Glue Components

Step 1 - Download the CloudFormation template

1. Use the Canvas Glue CloudFormation template: **canvas-data-product-glue.cfn.yaml**
2. If you wish to make any changes to the CloudFormation template (e.g., change resource names to match your naming conventions) do that at this time.
3. Log in to the AWS Management Console. Note, the user you are logging in with must have sufficient privileges to create CloudFormation stacks and the AWS resources needed for the AWS Canvas Live Events Connector (list of resources below).
4. Navigate to CloudFormation

The CloudFormation stack will create the following resources:

- One Glue database
- One Glue table
- One Glue crawler
- One IAM role

Step 2 - Create Stack

1. Click on the Create stack button

The screenshot shows the AWS CloudFormation console interface. At the top, the header reads 'Management & Governance' followed by 'AWS CloudFormation' in large bold letters. Below this, the text says 'Model and provision all your cloud infrastructure'. A red arrow points from this text to an orange 'Create stack' button. To the right of the main content, there is a sidebar with a 'Create a CloudFormation stack' section containing the text 'Use your own template or a sample template to quickly get started.' and the 'Create stack' button. Below this, the 'Getting started' section lists links: 'What is AWS CloudFormation', 'Getting started with CloudFormation', 'Learn template basics', and 'Quick starts'. At the bottom, the 'More resources' section lists 'Documentation', 'API reference', and 'FAQs'. In the center, there is a video player titled 'Simplify Your Infrastructure Management Using AWS CloudFormation' with a play button icon. The video player also includes the text 'Simplify Your Infrastructure Management Using A...', 'Copy link', 'AWS Management and Governance', and 'Watch on YouTube'.

Management & Governance

AWS CloudFormation

Model and provision all your cloud infrastructure

AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.

Create a CloudFormation stack

Use your own template or a sample template to quickly get started.

Create stack

Getting started

- [What is AWS CloudFormation](#)
- [Getting started with CloudFormation](#)
- [Learn template basics](#)
- [Quick starts](#)

More resources

- [Documentation](#)
- [API reference](#)
- [FAQs](#)

How it works

Simplify Your Infrastructure Management Using A...

Copy link

Simplify Your Infrastructure Management Using AWS CloudFormation

AWS Management and Governance

Watch on YouTube

Step 3 - Create stack

1. Select "Template is ready"
2. Select "Upload a template file"
3. Click Choose file and then choose the yaml file that you downloaded at Step 1
4. Click Next

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file

No file chosen

JSON or YAML formatted file

S3 URL: Will be generated when template file is uploaded

View in Designer

Next

Step 4 - Specify stack details

The Organization and Project parameters are used as substitution variables in AWS resource names (e.g., Amazon S3 bucket - calpoly-qa-canvas-events). Organization should be a short abbreviation for the institution. Project should be a short abbreviation for the initiative or environment. All AWS resources are also tagged with the Project value.

1. Enter a Stack name (e.g., CanvasGlueStack)
2. Enter a value for the Organization parameter (e.g., calpoly)
3. Enter a value for the Project parameter (e.g., qa)
4. Click Next

Specify stack details

Stack name

Stack name

Enter a stack name

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.

Organization

unicon

Project

devers

Cancel

Previous

Next

Step 5 - Configure stack options

No changes are required on this screen. However, if you have your own AWS best practices or conventions feel free to apply them now.

If you do not select IAM role to use, CloudFormation will create one for you.

1. Click Next

Configure stack options

Tags

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](#)

Key	Value	Remove
Add tag		

Permissions

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](#)

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

IAM role name	Sample-role-name	Remove
---------------	------------------	--------

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. [Learn more](#)

☒ Roll back all stack resources

Roll back the stack to the last known stable state.

☐ Preserve successfully provisioned resources

Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Advanced options

You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

► Stack policy

Defines the resources that you want to protect from unintentional updates during a stack update.

► Rollback configuration

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. [Learn more](#)

► Notification options

► Stack creation options

Cancel



Next

Step 6 - Review stack

Review the stack configuration.

1. Click the checkbox (I acknowledge that AWS CloudFormation might create IAM resources with custom names.) in the blue box
2. Click Create stack

This will initiate the process of creating all of the required resources. This process can take a couple of minutes.

Create stack

Review **ConvasLiveEventsStack**

Step 1: Specify template

Template

Template ID: `aws-2-amazonaws.com:tf-templates-1:cloudtrail-gp-v1-eat-2/2021-11-04:aws-cloud-trail-events-oh-park`

Stack description: -

Estimate cost: [View](#)

Step 2: Specify stack details

Parameters (2)

Key

Value

Organization

us-east-1

Project

ml-solutions

Step 3: Configure stack options

Tags (0)

Key

Value

No tags

There are no tags defined for this stack.

Permissions

No permissions

There is no IAM role associated with this stack.

Stack failure options

Rollback on failure

Disabled

Stack policy

No stack policy

There is no stack policy defined.

Rollback configuration

Monitoring time

-

CloudWatch alarm ARN

-

Notification options

No notification options

There are no notification options defined.

Stack creation options

Timeout

-

Termination protection

Disabled

Quick-start link

Capabilities

The following resource(s) require capability: **[AWS-IAM::Role]**

This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. [Learn more](#)

I acknowledge that AWS CloudFormation might create IAM resources with custom names.

Cancel

Previous

Create change set

Create stack

Installing the AWS Glue DataBrew Components

Step 1 - Download the CloudFormation template

1. Use the AWS Canvas Live Events Connector CloudFormation template:
canvas-data-product-databrew.cfn.yaml
2. If you wish to make any changes to the CloudFormation template (e.g., change resource names to match your naming conventions) do that at this time.
3. Log in to the AWS Management Console. Note, the user you are logging in with must have sufficient privileges to create CloudFormation stacks and the AWS resources needed for the Canvas Live Events Connector (list of resources below).
4. Navigate to CloudFormation

The CloudFormation stack will create the following resources:

- One Amazon S3 bucket
- One AWS Glue DataBrew project
- Four AWS Glue DataBrew datasets
- One IAM Role

Step 2 - Create Stack

1. Click on the Create stack button

The screenshot shows the AWS CloudFormation console interface. At the top, the header reads 'Management & Governance' followed by 'AWS CloudFormation' and the tagline 'Model and provision all your cloud infrastructure'. Below this, a description states: 'AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.' On the right side, there is a white box titled 'Create a CloudFormation stack' which contains the text 'Use your own template or a sample template to quickly get started.' and an orange 'Create stack' button. A red arrow points from the 'Create stack' button in the white box to the 'Create stack' button in the orange box. Below the main content, there is a 'How it works' section featuring a video player with the title 'Simplify Your Infrastructure Management Using AWS CloudFormation' and a 'Watch on YouTube' button. On the right side, there are two sections: 'Getting started' with links for 'What is AWS CloudFormation', 'Getting started with CloudFormation', 'Learn template basics', and 'Quick starts'; and 'More resources' with links for 'Documentation', 'API reference', and 'FAQs'.

Management & Governance

AWS CloudFormation

Model and provision all your cloud infrastructure

AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.

Create a CloudFormation stack

Use your own template or a sample template to quickly get started.

Create stack

How it works

Simplify Your Infrastructure Management Using AWS CloudFormation

Watch on YouTube

Getting started

- [What is AWS CloudFormation](#)
- [Getting started with CloudFormation](#)
- [Learn template basics](#)
- [Quick starts](#)

More resources

- [Documentation](#)
- [API reference](#)
- [FAQs](#)

Step 3 - Create stack

1. Select "Template is ready"
2. Select "Upload a template file"
3. Click Choose file and then choose the yaml file that you downloaded at Step 1
4. Click Next

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file

No file chosen

JSON or YAML formatted file

S3 URL: Will be generated when template file is uploaded

View in Designer

Next

Step 4 - Specify stack details

The Organization and Project parameters are used as substitution variables in AWS resource names (e.g., Amazon S3 bucket - calpoly-qa-canvas-events). Organization should be a short abbreviation for the institution. Project should be a short abbreviation for the initiative or environment. All AWS resources are also tagged with the Project value.

The three *Flow parameters must match the names that you used when creating the Amazon AppFlows.

1. Enter a Stack name (e.g., CanvasDatabrewStack)
2. Leave the default values for the three Flow parameters (unless you used different names when creating the Amazon AppFlows - **they must match**)
3. Enter a value for the Organization parameter (e.g., calpoly)
4. Enter a value for the Project parameter (e.g., qa)
5. Click Next

Stack name

Stack name

Enter a stack name

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.

CanvasAssignmentFlow

CanvasAssignmentFlow

CanvasCourseFlow

CanvasCourseFlow

CanvasUserFlow

CanvasUserFlow

Organization

unicon

Project

devers

Step 5 - Configure stack options

No changes are required on this screen. However, if you have your own AWS best practices or conventions feel free to apply them now.

If you do not select IAM role to use, CloudFormation will create one for you.

1. Click Next

Configure stack options

Tags

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](#)

Key	Value	Remove
Add tag		

Permissions

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](#)

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

IAM role name	Sample-role-name	Remove
---------------	------------------	--------

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. [Learn more](#)

☒ Roll back all stack resources

Roll back the stack to the last known stable state.

☐ Preserve successfully provisioned resources

Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Advanced options

You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

Stack policy

Defines the resources that you want to protect from unintentional updates during a stack update.

Rollback configuration

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. [Learn more](#)

Notification options

Stack creation options

Cancel



Next

Step 6 - Review stack

Review the stack configuration.

1. Click the checkbox (I acknowledge that AWS CloudFormation might create IAM resources with custom names.) in the blue box
2. Click Create stack

This will initiate the process of creating all of the required resources. This process can take a couple of minutes.

Review ElasticSearchEventsStack Stack

Step 1: Specify template Exit

Template

Template URL
<https://aws-elasticsearch.amazonaws.com/templates/1/idxmappings-us-east-2/2012118/index-canonical-events-stack.json>

Stack description
 +
[Elasticsearch logs](#)

Step 2: Specify stack details Exit

Parameters (2)

Key	Value
Organization	amazon
Project	indexlogs

Step 3: Configure stack options Exit

Tags (0)

+ Add new tag

Key	Value
No tags	

There are no tags defined for this stack.

Permissions

No permissions

There is no IAM role associated with this stack.

Stack failure actions

Rollback on failure
 Disabled

Stack policy

No stack policy

There is no stack policy defined.

Rollback configuration

Monitoring time
 -

CrossWatch alarm state
 -

Notification options

No notification options

There are no notification options defined.

Stack creation options

Timeout
 -

Termination protection
 Disabled

+ Quick-create link

Capabilities

☒ The following resource(s) require capabilities [AWS::IAM::Role]
 This template creates identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions; or, add them as custom roles. Check that the custom names are unique within your AWS account. Learn more >

☐ I acknowledge that AWS CloudFormation might create IAM resources with custom names.

Create Cancel Previous Create Change Set Create Stack

You're done! At this point all of the solution components are in place. Feel free to update the Glue Crawler, AppFlows or DataBrew job to run on a schedule.

Rolling Back

With the exception of the Amazon AppFlow components, the easiest way to rollback any parts of the solution is to delete the Cloudformation stack. This is one of the reasons we broke up the Cloudformation templates - so that you can manage each part of the solution independently.

With Amazon AppFlow, you will need to manually delete any of the components to remove them.