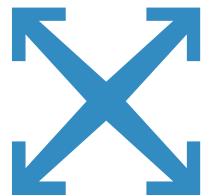




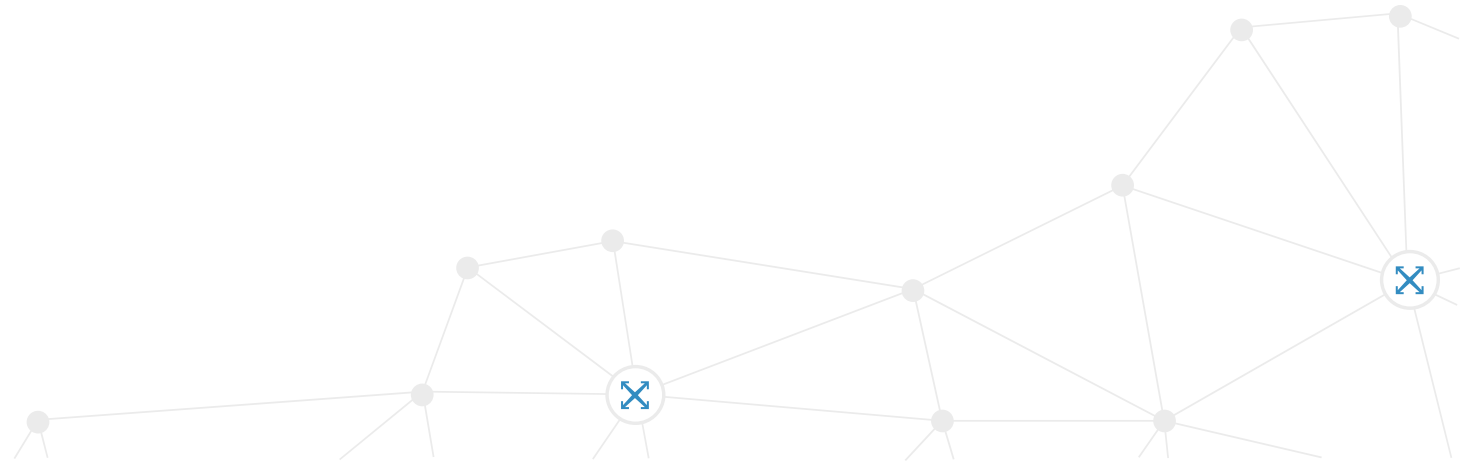
**ACADEMY**



Edge Analytics Online Training

# ADVANCED EDITION

Technical Deep-Dive into  
Streaming Analytics at the Edge



# Lecturer



**Göran Appelquist**

CTO

goran.appelquist@crosser.io




**Mikael Isenberg**


Solution Architect  
Product Owner

mikael.isenberg@crosser.io

Top 10 Edge Analytics Use Cases  
Hosted by Crosser Webcasts

Blank **Slides**

 **crosser**  
EDGE ANALYTICS

 crosser

New speaking: Johan Jonzon

**Chat** Q&A

Public Presenters Private Twitter

Type your message here... **Send**

# COURSE SETUP

- All participants will receive their own login to Crosser Cloud  
Handouts are available in the handout section in the webinar room and on the landing page: <https://crosser.io/academy/advanced-online-training-course-participants-information/>
- Recordings of these sessions will also be available shortly afterwards
- You are expected to work on your own with the exercises
- Support channels to use during the course
  - Chat Support (Slack) → You will get an invite
  - Email support → [support@crosser.io](mailto:support@crosser.io)

# SESSIONS

SESSION  
**01**

Edge Analytics Advanced

## THE UNIVERSAL CONNECTOR

Build reusable modules for  
REST APIs

SESSION  
**02**

Edge Analytics Advanced

## DATABASES

Working with databases

SESSION  
**03**

Edge Analytics Advanced

## FLOW PARAMETERS

Same Setting - multiple Flows  
Same Flow - multiple Nodes

SESSION  
**04**

Edge Analytics Advanced

## CUSTOM CODE MODULES

Add your own logic

SESSION  
**05**

Edge Analytics Advanced

## MODULES

Message/Time Counter, State and  
Delay modules

SESSION  
**06**

Edge Analytics Advanced

## INTER FLOW COMMUNICATION



SESSION  
**01**

Edge Analytics Advanced

**THE UNIVERSAL  
CONNECTOR**

Build reusable modules for REST APIs



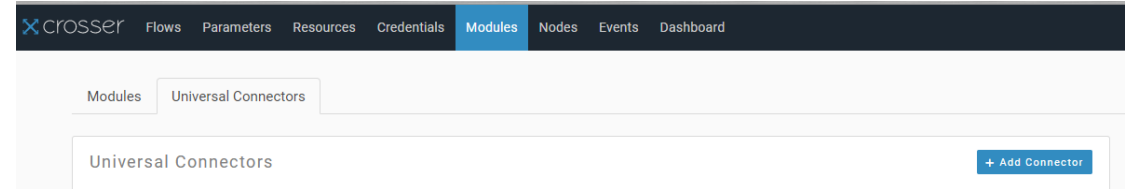
# SESSION - 01

- Introduction to Universal Connector
- Universal Connector Wizard
- Exercises

# Universal Connector

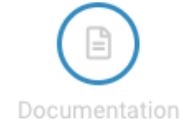
## - Introduction

- Use it to build your own modules that connect to REST API:s
- Available from Modules page => Universal Connector
  - Supports versioning of Universal Connectors
- Input or Output supported
- Step by step Wizard
- Possible to use properties from incoming messages





# Universal Connector

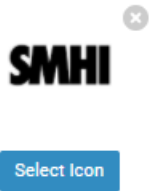


Next

## General (Step 1)

- Name
- Version (x.x.x)
- Description
- Input or Output
- Icon

|  |   |
|--|---|
| <input type="text" value="SMHI"/>                | Module type<br><input type="text" value="Input"/> |
| <input type="text" value="1.0.0"/>               |   |
| <input type="text" value="SMHI weather report"/> |   |



# Universal Connector



## Authentication (Step 2)

- Credentials are added from the Credential page
  - Supported Authentication methods:
    - Basic
    - API keys
    - OAuth (Will be released in Q4)
- Authentication Headers and Query Parameters
  - Header/Parameter added in UI
  - Value can be added
    - In the Wizard UI - E.g.: 12344566
    - As a setting - E.g.: {appKey}

The screenshot shows the 'Configure Universal Connector' wizard. The progress bar at the top highlights the 'Authentication' step. Below the progress bar, there are three sections for configuration:

- Credential types**: A dropdown menu labeled 'Credential'.
- Authentication headers**: A table with two columns: 'Header' and 'Value'. The first row has 'appKey' in the header and '{appKey}' in the value. There are '+' and '-' buttons to the right of the value field.
- Authentication query parameters**: A table with two columns: 'Parameter' and 'Value'. There is a '+' button to the right of the value field.

# Universal Connector



## Configuration and Test (Step 3)

### Settings

- **URL to the REST API endpoint**
  - Settings can be added and used on URL: {property name}
- **Actions**
  - GET, PUT, POST, DELETE, OPTIONS, HEAD, PATCH, TRACE
- **Headers**
- **Query parameters**
- **Content-Type** (only applicable for POST, PUT, DELETE, PATCH)
  - application/json
  - application/xml
  - text/plain
- **Body** (only applicable for POST, PUT, DELETE, PATCH)
  - Entire message property from incoming message
    - Example: {body}
  - Added straight in the UI, possible to also use properties
    - Example: Body={message}&From={from}&To={to}

URL

Action

Headers

|                                     |                                    |                                  |
|-------------------------------------|------------------------------------|----------------------------------|
| <input type="text" value="Header"/> | <input type="text" value="Value"/> | <input type="button" value="+"/> |
|-------------------------------------|------------------------------------|----------------------------------|

Query parameters

|  |                                    |                                  |
|--|------------------------------------|----------------------------------|
| <input type="text" value="Parameter"/> | <input type="text" value="Value"/> | <input type="button" value="+"/> |
|--|------------------------------------|----------------------------------|

# Universal Connector



Request | Response

Url  
https://opendata-download-metfcst.smhi.se/api/category/pmp3g/version/2/geotype/pc

longitude 16

latitude 58

Test Clear

Status:

Close



Test result

Request | Response

Url  
https://opendata-download-metfcst.smhi.se/api/category/pmp3g/version/2/geotype/pc

longitude 16

latitude 58

Test Clear

Status: 200

Curl Copy to clipboard

```
curl -XGET "https://opendata-download-metfcst.smhi.se/api/category/pmp3g/version/2/geotype/point/lon/16/lat/58/data.json"
```

Close

## Configuration and Test (Step 3)

- Possible to test and verify the connection in the wizard
- Possible to use settings when testing
- Shows status and response message

# Universal Connector



| Setting                 | Source              | Display name    | Type   | Requirements          | Default value | Help text                              | Purpose                                |
|-------------------------|---------------------|-----------------|--------|-----------------------|---------------|--|--|
| targetPropertyParameter | User setting        | Target Property | String | min length max length | data          | The property to write the result into. | The property to write the result into. |
| longitude               | User setting if set | longitude       | String | min length max length | 16            | The longitude of the location          | The longitude of the location          |
| latitude                | User setting if set | latitude        | String | min length max length | 58            | The latitude of the location           | The latitude of the location           |

## Usage settings (Step 4)

- Define settings
- All defined settings will show up and be configurable
- targetPropertyParameter – Only available for Input modules
- **Set source**
  - User Setting
  - Message parameter
  - User setting if set
- **Display name** – The name the property will have in Module settings
- **Type**: String, Number, Boolean
- Requirements – length, min/max
- **Default value** – Set the default value for the property
- **Help text** – Property help text in module settings documentation
- **Purpose** – Property purpose text in module settings documentation

# Universal Connector



## Documentation (Step 5)

- Add necessary documentation
- Connector description
  - Short description of the module
  - This is the text that will be seen in the Module library in FlowStudio
- Module documentation
  - Describe the module
  - This text is the top text in the Module – Info tab.

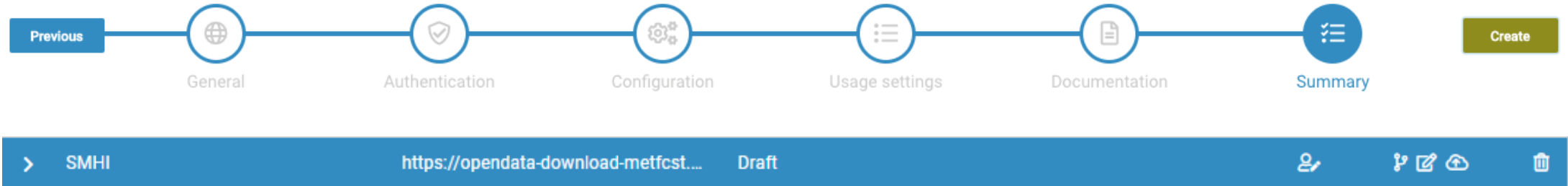
### Connector description

```
# SMHI  
SMHI weather report
```

### Module documentation

```
# SMHI  
This module get weather forecasts based on longitude and latitude values
```

# Universal Connector



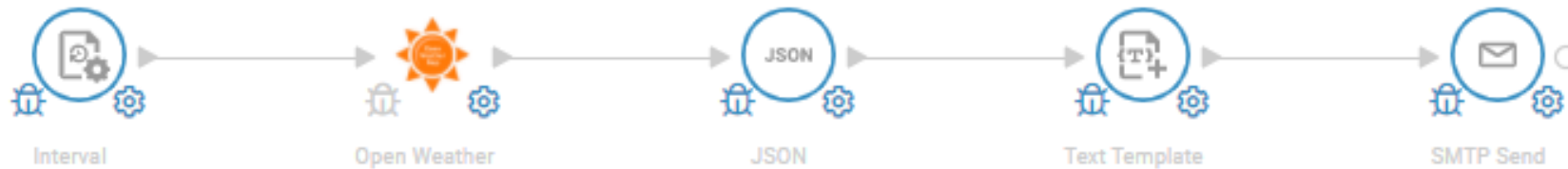
## Summary and Publish (Step 6)

- Review the result of how the module will look like
- Settings and Info tab are available to review
- Create the Universal Connector
- Next step is to publish as a new module
  - Click Publish version to publish it as a new module
  - Now it will be available from FlowStudio as any other module
  - Only available within your organization
  - Once published the version will be read-only
- Make changes
  - It is possible to create a new version of the Universal Connector
  - It is also possible to create a new one from an existing Universal Connector

The screenshot shows the configuration interface for a connector named 'SMHI'. On the left, there is a preview area with the 'SMHI' logo and a sample output: 'SMHI weather report'. On the right, there are two tabs: 'Settings' and 'Info'. The 'Settings' tab is active, showing a 'Target Property' field with the value 'data'. Below this, there are two input fields for 'longitude' (value '16') and 'latitude' (value '58').

# Exercise 1A

## - Get and use weather information



1. Create an Open Weather module that can get weather reports
  1. City should be a setting – either defined in UI or by incoming message
  2. Link to API documentation: <https://openweathermap.org/current>
  3. The API-key is: **9dae47f4b2791c84d9f7c96d3655116d**
2. Create a flow that gets weather information from your favorite city and use the response to send City name, temp\_min and temp\_max in an e-mail to yourself
  1. Use the Interval module
  2. Add your newly created Open Weather module and set your favorite city
  3. Add a JSON module
  4. Add a Text Template module, with city\_name, temp\_min and temp\_max properties
  5. Add a SMTP Send module
  6. Run the flow and make sure you get an email with City, max and min temp



# Exercise 1B

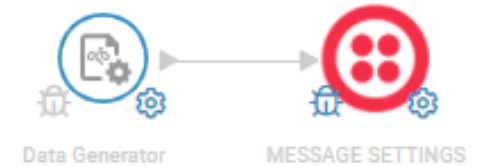
## - Send a SMS using Twilio

### 1. Create a Twilio module that sends messages as SMS

1. Register for a free Twilio account for your cell phone number
  1. <https://www.twilio.com/sms>
  2. Get you credentials (Account SID and Auth Token)
2. Link to API documentation: <https://www.twilio.com/docs/sms/api>
3. Settings that should be available in Twilio module
  1. Account SID
  2. To (number)
  3. From (number)
  4. Message (Text)
  5. Credentials (Username & Password used for SID and Token)

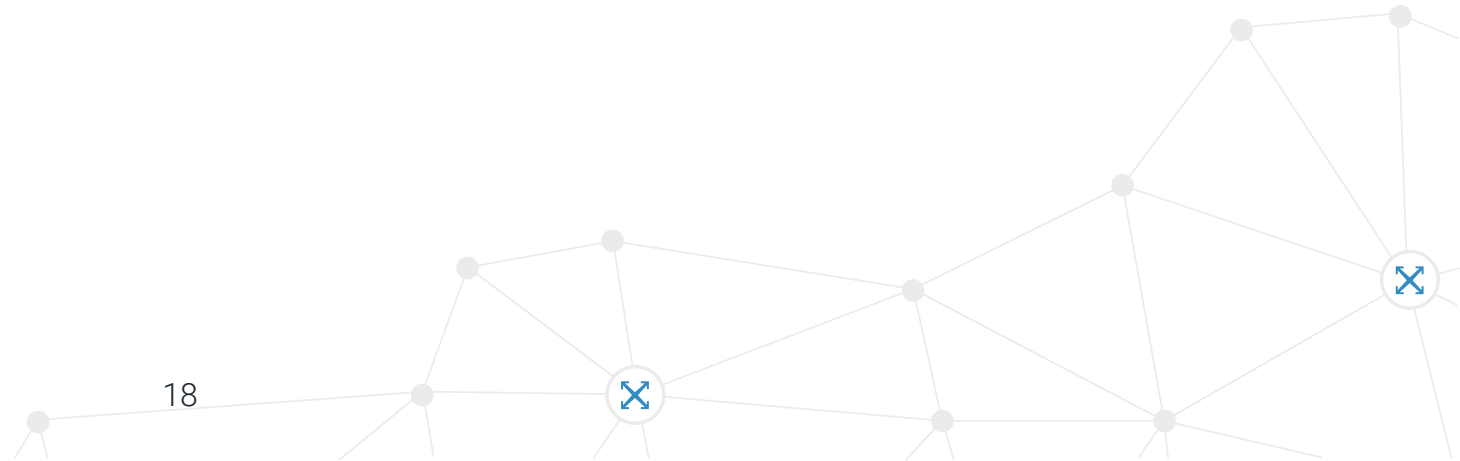
### 2. Create a flow that sends a SMS to your cell phone number

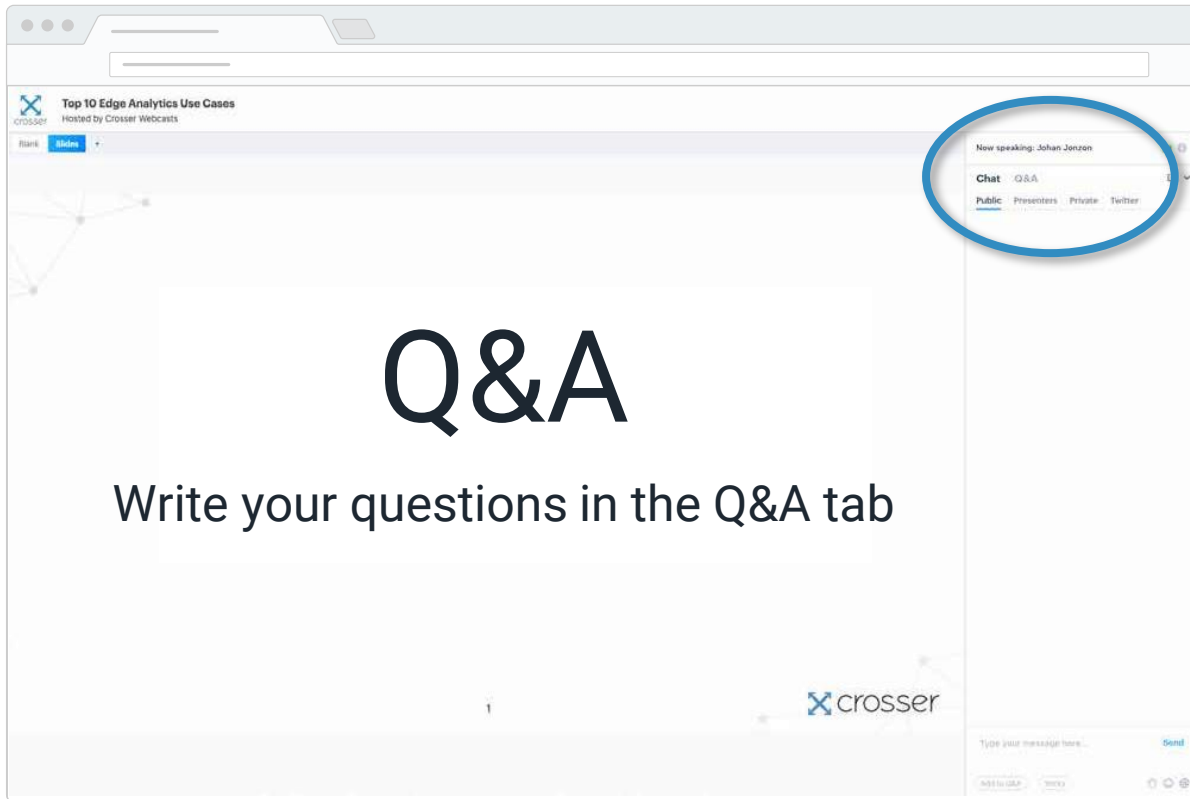
1. Use a Data generator module to generate the needed values (to, from, message) for the Twilio module
2. Add you Twilio module
3. Run the flow and make sure you get an SMS with your message text from the Data generator





# SESSION – 01 END





A profile card for Mikael Isenberg. At the top is a circular portrait photo of a man with a beard and short hair, wearing a blue and white checkered shirt. Below the photo, the name "Mikael Isenberg" is written in a large, bold, black font. Underneath the name, the titles "Solution Architect" and "Product Owner" are listed in a smaller, blue font. At the bottom of the card, the email address "mikael.isenberg@crosser.io" is provided in a black font. The card has a white background and a blue border.