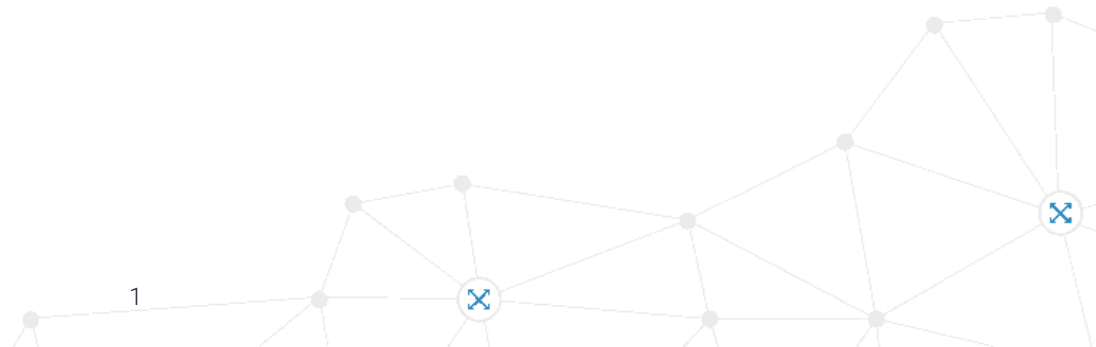SESSION

06

Edge Analytics Fundamentals

WORKING WITH FILES & USER MANAGEMENT

Course examination

Session 6
Agenda

Administration

- Users
- Roles & Permissions
- Authenticating users using external authentication services

Working with files

- CSV Reader module
- File TextReader module
- File TextWriter module
- File StreamWriter module
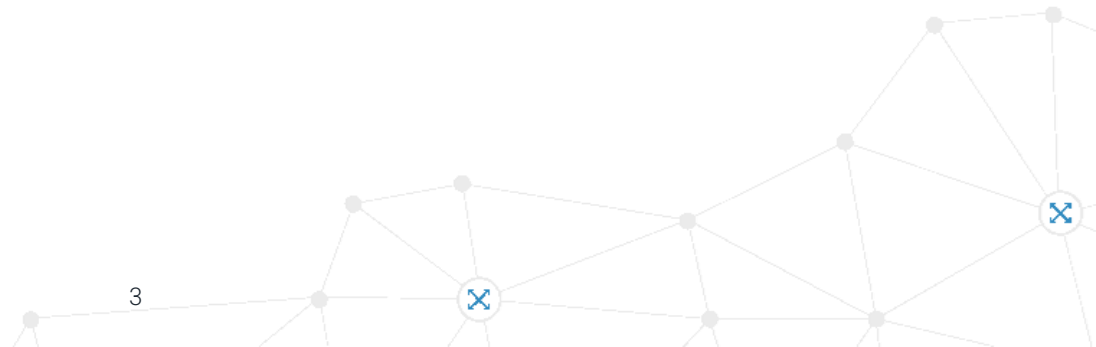- Additional file related modules
- Exercise 6

Exam!

crosser

CROSSER CONTROL CENTER

Administration

User Management

- Users are managed on the *Organization* page (found in the user menu)
- Requires *User, Roles* or *Identity Provider* permissions
  - Not available on free-trial accounts
- Here you can:
  - Add new Users
  - Assign Roles to users
  - Create/Modify Roles
  - Enable two-factor authentication for each user
  - Setup authentication of users through federation with external identity providers using OpenID Connect, e.g. Active Directory

Goran
- Crosser Academy

Settings

Organization

Documentation

LOG OUT

crosser
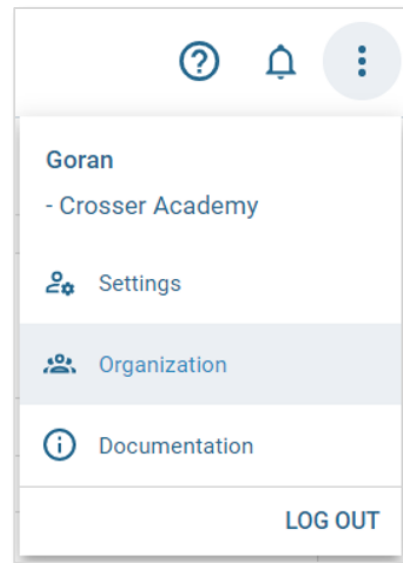
Permissions and Roles

Permissions

Select at least one permission to be used in the role.

Name	Actions						Select All	Labels
› Nodes	☰	⊞	✎	🗑	🔌	⑂	☑	
› Flows	☰	⊞	✎	🗑			☑	
› Credentials	☰	⊞	✎	🗑			☑	
› Resources	☰	⊞	✎	🗑			☑	
› Universal Connector	☰	⊞	✎	🗑	☁		☑	
› Labels	☰	⊞	✎	🗑			☑	
› Modules	☰	⊞	✎	🗑			☑	
› Users	☰	⊞	✎	🗑	👤		☑	
› Roles	☰	⊞	✎	🗑			☑	
› Identity Providers	☰	⊞	✎	🗑			☑	

Not in "Crosser Standard User"

View   Create   Edit   Delete

- Roles give Users permissions to use different parts of the tool
- A User can be assigned one or several Roles
- Custom Roles can be added

- Pre-defined roles:
Super User
  - Can do everything
  - Assigned to account owner

- **Crosser Standard User**
  - Can do everything except user management

crosser

Identity Providers

Identity provider wizard

- Crosser can give your organisation control over specific domains

- These domains can then be setup to use your federation server (through OpenID Connect)

- Roles/Groups in your directory can be mapped against Roles in Crosser

Note

Contact Crosser to enable external authentication for your domains

FILE MODULES
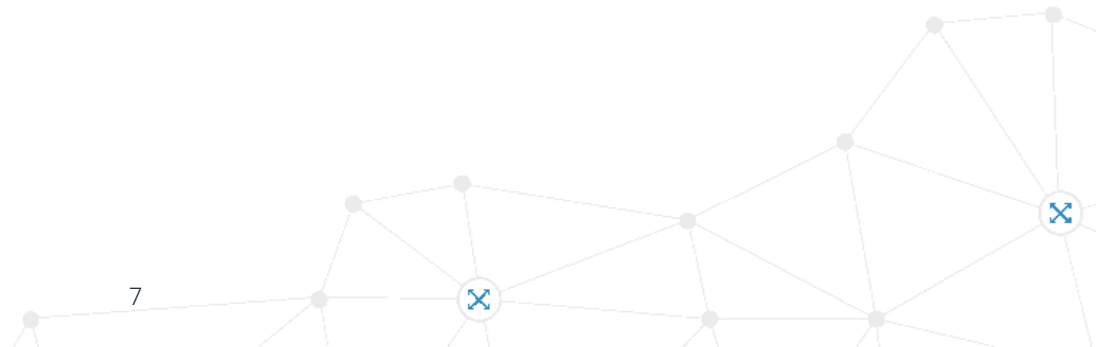
Module
File TextReader

- Reads a whole text file and produces one output message with the whole content assigned as a string on the target property

- Use a static filename (UI setting) or get the filename from the incoming message

- Must be triggered by an incoming message

Module Settings

File TextReader
File TextReader

Settings   Common   Documentation >

Name
File TextReader

Version
2.1.0

Compatible with all Nodes from 2.3

Static Filename
data/tmp/demo.txt

Set this to use same filename all the time

Filename Property

Dynamic filename set by the incoming message

Target Property
data

The property to write the file content into

crosser
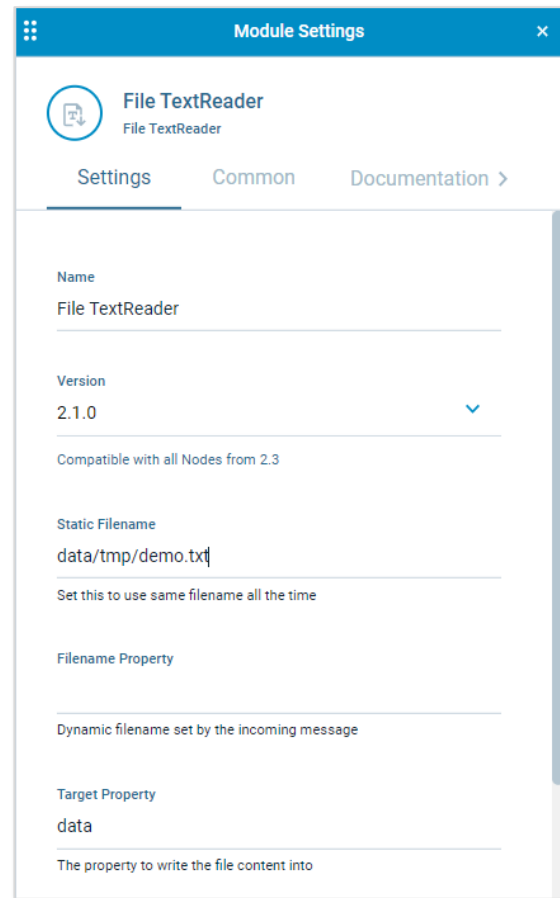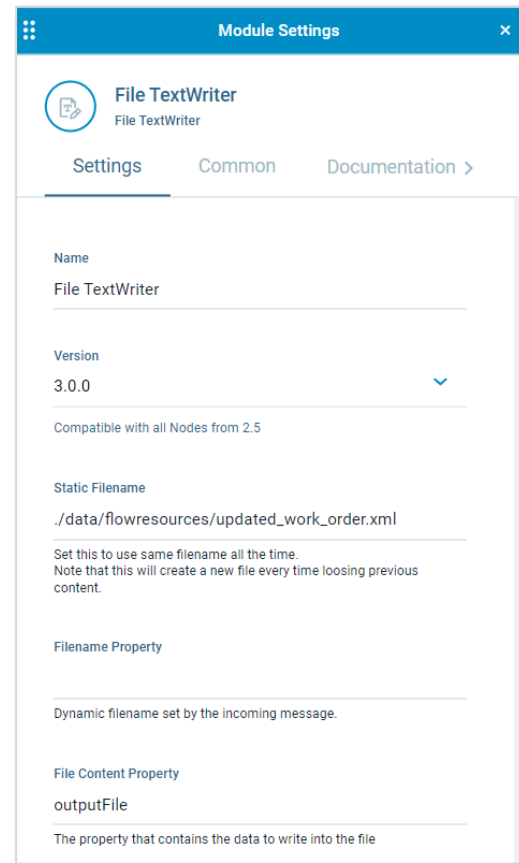
Module
File TextWriter

- Write a single string property to a file

- Existing files will be overwritten

- Use a static filename (UI setting) or get the filename from the incoming message

Module Settings

File TextWriter
File TextWriter

Settings | Common | Documentation >

Name
File TextWriter

Version
3.0.0

Compatible with all Nodes from 2.5

Static Filename
./data/flowresources/updated_work_order.xml

Set this to use same filename all the time.
Note that this will create a new file every time loosing previous content.

Filename Property

Dynamic filename set by the incoming message.

File Content Property
outputFile

The property that contains the data to write into the file

Module
File StreamWriter

- Append text strings to files

- Files rotated on time (configurable)

- Optional header added, use e.g. to create CSV files, see also *CSV StreamWriter*

- Optional compression of rotated files

- Files are named:
  <base-name>-<timestamp>.<ext>

Module Settings

File StreamWriter
File StreamWriter

Settings   Common   Documentation >

Name
File StreamWriter

Version
3.0.0

Compatible with all Nodes from 2.5

Source Property
data
The property that contains the string(s) to write to the file

Target Property
data
The property to write the result to when passing message to next module(s)

Target Path
./data/flowresources
Directory to write to

Base Filename
csv_demo
Add extension below

crosser

Module
CSV Reader

- Read CSV files from local storage and produce one message per row

- Output messages are objects with 'col':'value' pairs

- If a header is available column names will be used as property name, otherwise 'col1', 'col2'…

- Numeric values are converted

- Either the whole file or a specified number of lines read on each trigger

- Use fixed filename (UI setting) or provide the filename with the incoming message

Module Settings

CSV Reader
CSV Reader

Settings   Common   Documentation >

Name
CSV Reader

Version
3.0.1
Compatible with all Nodes from 2.5

File
./data/flowresources/training.csv

Target Property
data

☑ Treat first line as column header

Number of lines to read
5
Max number of lines to read. Leave empty if all lines should be read

Delimiter
,

crosser
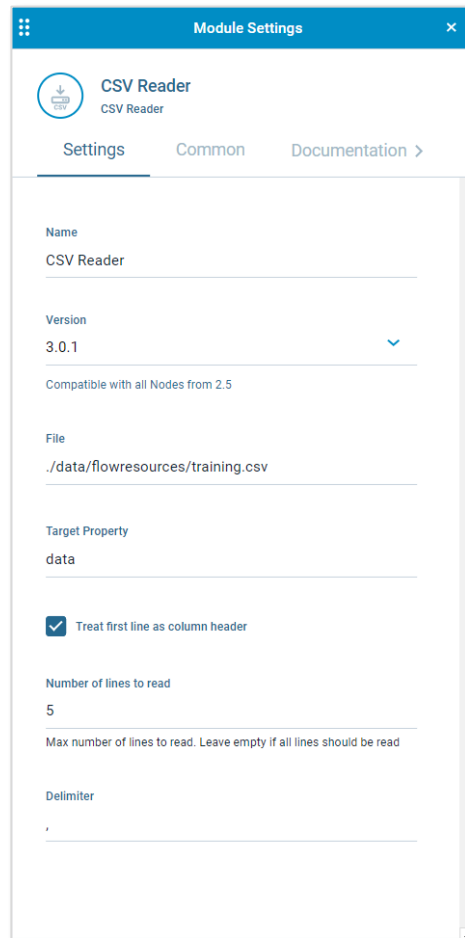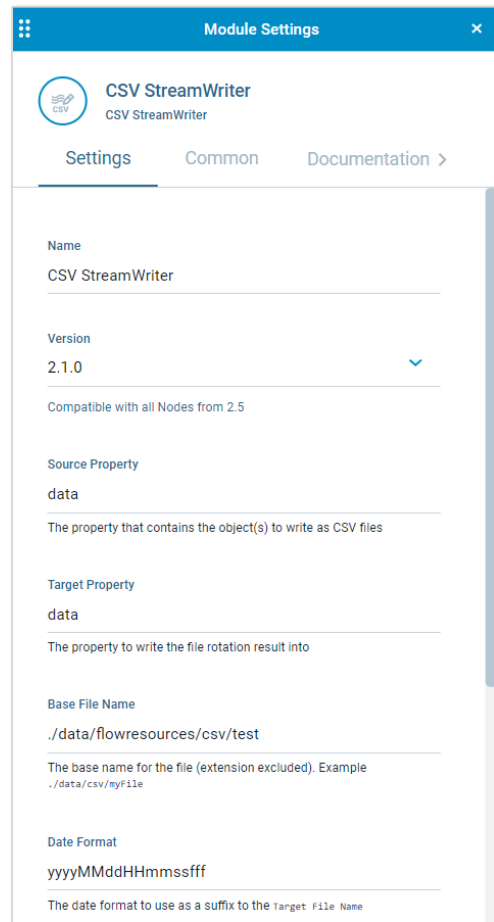
Module
CSV StreamWriter

- Append message data to CSV files
  - Input from an object, or an array of objects
  - Each object should have the same properties
  - Property names used as header

- Files rotated on time or number of rows, or both

- Files are named:
  <base-name>-<timestamp>.csv

- Configurable delimiter and timestamp format

Note

This module only works on Docker Nodes

Module Settings

CSV StreamWriter
CSV StreamWriter

Settings   Common   Documentation >

Name
CSV StreamWriter

Version
2.1.0

Compatible with all Nodes from 2.5

Source Property
data

The property that contains the object(s) to write as CSV files

Target Property
data

The property to write the file rotation result into

Base File Name
./data/flowresources/csv/test

The base name for the file (extension excluded). Example ./data/csv/myFile

Date Format
yyyyMMddHHmmssfff

The date format to use as a suffix to the Target File Name

Modules

Additional modules for working with files in local storage

File Finder

- Scan a directory for files matching pattern
- Output is a message with an array of filenames

File TailReader

- Get a message when new content is added to a file

Files Watcher

- Watch for changes to files or directories
- Get a message when files have been changed, created or deleted

Files Move   Files Delete

- Move or Delete files

crosser

Modules

Modules for working with remote files

SFTP Text Reader

- Get the content of a text file on the SFTP server

SFTP Directory Reader   **SFTP Files Watcher(2)**

- Scan a directory for files matching pattern

- Get a message when files are created or changed

SFTP Files Move

- Move files on the SFTP server

SFTP Download   **SFTP Upload**

- Download files to local storage or upload files from local storage to the SFTP server

crosser

EXERCISES

Working with files
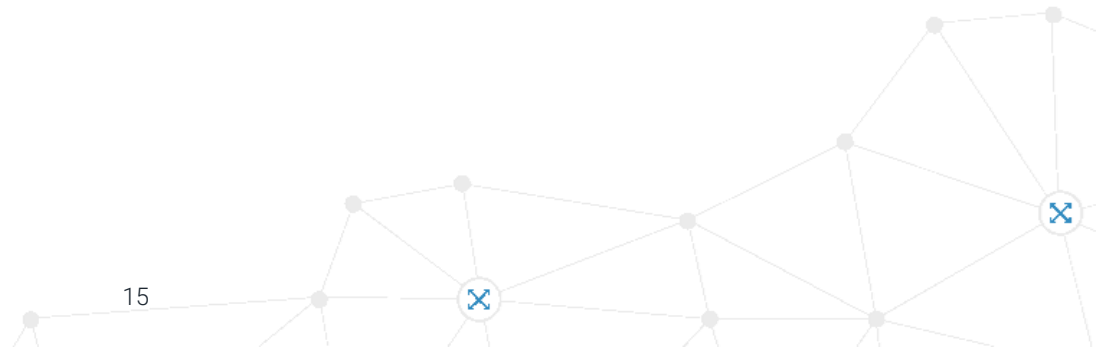
Exercise 6
Creating resources

For these exercises you need to create two resources before you start building flows

1. On the Resources page, add a new 'File' resource, give it a name and select 'Enter data in editor…'

2. Copy the text in the 'training.csv' box to the right, including the header, and paste it into the resource editor. Then click on 'Add content'

3. Set the 'Local name on node' to training.csv

4. Click on 'Add resource'

5. Repeat these steps with the work_order.xml data with another name and set the 'Local name on node' to work_order.xml

training.csv

```
A,B,C
0.656522649,68,714.2343198
0.878196354,23,775.4661639
0.437855321,100,626.7427292
0.077856166,15,65.25553316
0.53325495,60,1.428440405
0.059479536,67,442.5756271
0.204823965,52,254.0578046
0.756972378,37,346.8085203
0.107829569,89,775.8296446
0.705399242,37,312.7642644
0.472487204,11,167.9224729
0.882779035,12,192.078554
0.424580664,59,322.0796454
0.603819557,41,648.9296402
0.953433287,82,258.1882842
0.653187851,66,433.4167383
0.62848724,54,220.6768886
0.096767329,32,785.0659949
0.786741734,45,987.4842342
0.920375056,37,604.4130675
```

work_order.xml

```xml
<?xml version="1.0" encoding="UTF-16"?>
<DOC>
  <HEADER>
    <TO_ADDRESS>CROSSER</TO_ADDRESS>
    <FROM_ADDRESS>STUDENT</FROM_ADDRESS>
    <IDENTITY>FUNDAMENTAL</IDENTITY>
    <TIME>2022-02-21 11:11:52</TIME>
    <MESSAGE_TYPE>WORK_ORDER</MESSAGE_TYPE>
  </HEADER>
  <BODY>
    <HEAD>
      <DIVI>SE</DIVI>
      <FACI>STO</FACI>
      <LINE>
        <EQNO>XYZ.123</EQNO>
        <STRT>STD</STRT>
        <TXT1>UPDATE MEASUREMENTS</TXT1>
        <VALUE1>0</VALUE1>
        <VALUE2>0</VALUE2>
      </LINE>
    </HEAD>
  </BODY>
</DOC>
```

16

Exercise 6.1
Overview

Interval → CSV Reader → Object To Array → Array Split → Aggregate
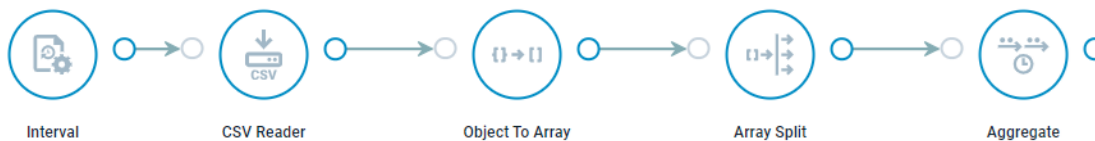
- Read CSV files from local disk (resource) and process the data

crosser

Exercise 6.1
Reading files

Interval   CSV Reader   Object To Array   Array Split   Aggregate
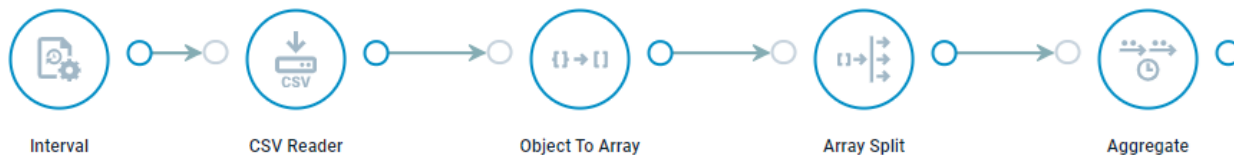
1. Create a new flow called Exercise 6.1 and add the CSV resource you created to the flow on the 'Resources' panel (right-hand menu in the Flow Studio)

2. Use a 'CSV Reader' module to read the file:
   - File: ./data/flowresources/training.csv
   - Use the 'Interval' module to trigger the 'CSV Reader' on startup

3. Run the flow and check the output

4. Build a flow that calculates the average of each column
   Answer: 'A': 0.5420425, 'B': 49.35, 'C': 446.7709

crosser

Exercise 6.1
Calculations with the Math Expression module

Interval → CSV Reader → Object To Array → Array Split → Aggregate → ? → Math Expression
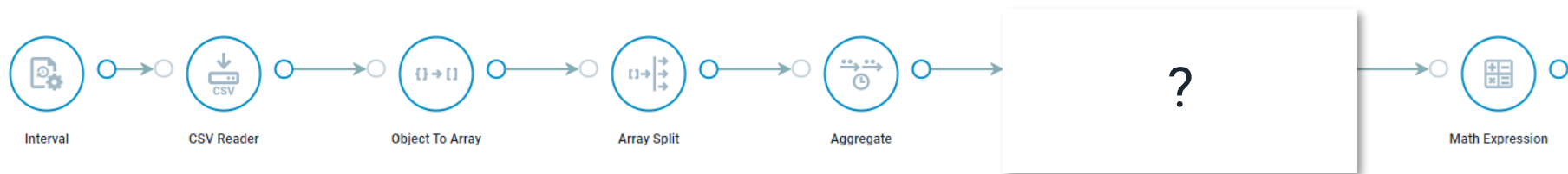
1. Calculate the sum of the results from the previous exercise, ie A + B + C, using the Math module (Answer: 496.6629425)

2. Hint: You must get all values into a single message. The modules from session 4 can come in handy!

crosser

Exercise 6.1
Wrap-up

Things to test/consider:

- Why did we have to use the 'Object to Array' and 'Array Split' modules before the 'Aggregate' module?

- Why did we have to convert messages after the 'Aggregate' module to calculate the sum?

- *Extra: Add a local file in the './data/flowresources' directory and modify your flow to use this file instead (you must run the flow on your local node)*

Interval   CSV Reader   Object To Array   Array Split   Aggregate

crosser
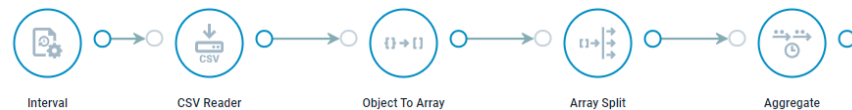
Exercise 6.2
Overview

- Read an XML template from local storage (resource) and modify some values
- Write back an updated XML file
- Conversion between external data formats and Flow Messages
- Note: To see the created file you need to run the flow on your local node

Exercise 6.2
Writing files

Data Generator → File TextReader → XML(2) → Property Mapper → XML → File TextWriter

1. Create a new flow called Exercise 6.2 and add the XML resource you created to the flow
2. Add a Data Generator module with the default template:
   - Sample Interval: 10 seconds
3. Use a File Text Reader module to read the template file:
   - Static Filename: ./data/flowresources/work_order.xml
   - Target Property: inputFile
4. Add an XML module to convert from XML into a Flow Message:
   - Source Property: inputFile
   - Target Property: xml
5. Run the flow and check the output from the 'File TextReader' and the 'XML' modules

Exercise 6.2
Writing files

Data Generator → File TextReader → XML(2) → Property Mapper → XML → File TextWriter

6. Use a 'Property Mapper' module to insert the value 'temp' and 'pressure' values from the 'Data Generator' into the 'VALUE1' and 'VALUE2' properties in the XML template

7. Convert the result back to XML:
   1. Source Property: xml
   2. Target Property: outputFile

8. Add a File Text Writer module to write the updated work order to disk:
   - Static Filename: ./data/flowresources/updated_work_order.xml
   - File Content Property: outputFile

9. Run the flow on your local node and verify that the output file is created

crosser

Exercise 6.2
Wrap-up

Data Generator → File TextReader → XML(2) → Property Mapper → XML → File TextWriter

Things to test/consider:

- Why do we need the XML modules and what is the different operations performed by the first and the second XML module

- Try adding a local file in the './data/flowresources' directory and read it in one of the flows (you must run the flow on your local node)

crosser

Exercise 6.3
Overview

On Docker nodes

Data Generator → Time Stamp → CSV StreamWriter

On Windows nodes

Data Generator → Time Stamp → Text Template → File StreamWriter
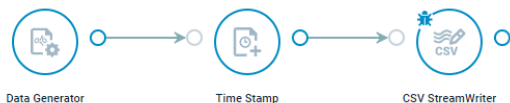
- Create random data, add a timestamp and append rows to a CSV file

- The exercise shows how to use the CSV StreamWriter module, which is the easiest way to create a CSV file.

- You need to run this flow on your local node to see the result

- *Note: The CSV StreamWriter module only works on Docker nodes. If you have installed your local node as a Windows service you can replace it with a Text Template module followed by a File StreamWriter module. You need to figure out how to configure these (hint: create a CSV line with the Text Template and add the header in the File StreamWriter)*

```
timestamp,temp,pressure
2020-02-18T08:16:43.4363073Z,31,78
2020-02-18T08:16:44.4555897Z,44,96
2020-02-18T08:16:45.4579760Z,17,36
2020-02-18T08:16:46.4585630Z,36,32
```

crosser

Exercise 6.3
Write streaming data to a CSV file

On Docker nodes

Data Generator → Time Stamp → CSV StreamWriter

On Windows nodes

Data Generator → Time Stamp → Text Template → File StreamWriter

1. Create a new flow called Exercise 6.3 and add a 'Data Generator' module with the default message template

2. Add a timestamp to the message ('Time Stamp' module):
   - Target Property: data.timestamp

3. Add a CSV StreamWriter module:
   - Base File Name: ./data/flowresources/csv/test
   - Max Lines Rotation: 10

4. Run the flow on your local node for at least 20 seconds and check the result in your output folder

crosser

Exercise 6.3
Wrap-up

On Docker nodes

Data Generator → Time Stamp → CSV StreamWriter

On Windows nodes

Data Generator → Time Stamp → Text Template → File StreamWriter
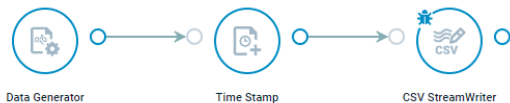
Things to test/consider:

- What happens if you change the rotation time to 5 seconds on the 'CSV StreamWriter' module? How many lines do you get in each file and why?

crosser

EXAM!

Exam
Introduction

- You will receive machine status messages every 5 seconds through MQTT

- Each message contains the machine uptime (seconds) and performance (items per minute), see example on the right.
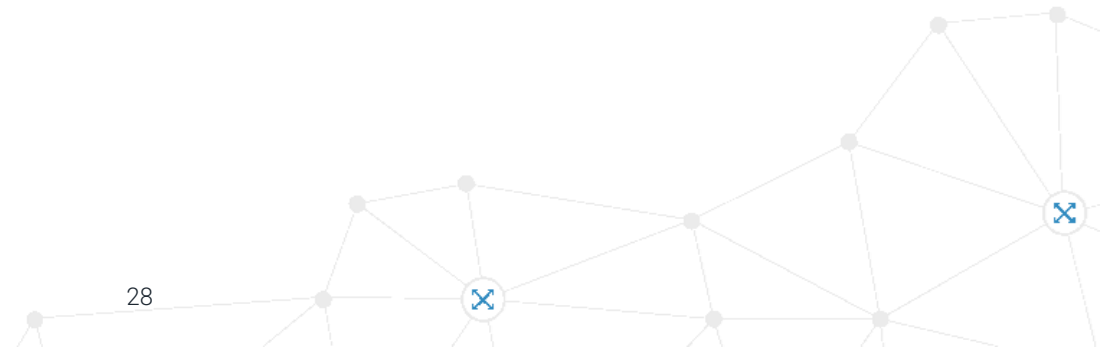
- There is a sequence of 4 messages, then the data repeats. One of the four messages contains invalid data (null)

- Your task is to build a flow that calculates the total number of products produced by the machine over this sequence:
  - The number of produced products, as reported by one message, is uptime * performance
  - Note 1: Keep track of the units used
  - Note 2: Only a whole number of products can be produced in each step, ie if the result from one message is 1.2 it means that only one product has been produced.

```
"data": [
  {
    "name": "uptime",
    "value": 5
  },
  {
    "name": "performance",
    "value": 500
  }
]
```

$$result = \sum_{msgs} Floor(uptime * performance/60)$$

crosser

Exam
Introduction

```
{
    "data": {
        "email": "name@company.com",
        "result": some number
    }
}
```

- You should report your result by sending a message over MQTT, where you include your email address and the number of products produced, see example on the right.

- You will get an email telling you if your response was correct or not.
  - Note: The email service has a rate limit of one message per minute. Since the data repeats you should design your flow so that only one message is sent. The 'Report by Exception' module can come in handy here.

- Delivery of a correct answer to the exam exercise will trigger a review of your account and if everything looks fine you should expect a diploma within a week, sent to the email address provided with the exam.

- Examination criteria:
  - Correct answer from the exam exercise delivered to Crosser
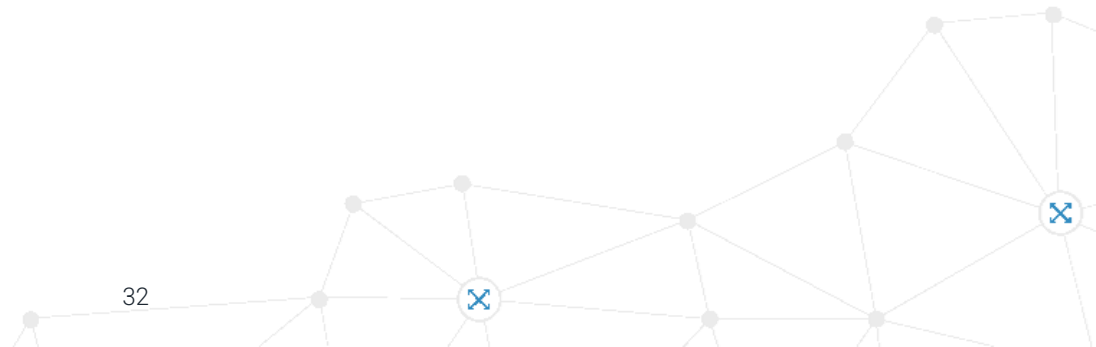 - Valid implementations of the exam flow and exercises

Exam
Setup

- MQTT broker URL: 10.0.48.117
- Topic for input data: training/exam1_input
- Topic for sending the result: training/exam1_result

crosser

COURSE WRAP-UP

Wrap-up

- ## What we hope you have learned:
  - Basic understanding of the Crosser Edge Analytics solution
  - Introduction to flow-based processing
  - Introduction to the Crosser module library
  - How to build, test and deploy use cases (Flows)
  - Hands-on experience of typical use cases:
    - Process streaming machine data
    - Accessing APIs
    - Working with files

Good luck with your future use cases!

Note

The Help Center is a good place to find information on the Crosser tools.

Some articles that may be of interest:
- Analytics Modules Overview
- Complete list of available modules
- Module Updates
- The Flow Studio
- Release Notes

crosser

THANK YOU!