

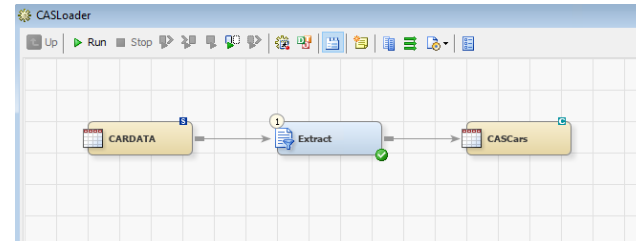
# Custom Transformasjon i DI Studio vs. Custom Step i flows I SAS Studio

Tom Arne Orthe

# Custom Transformasjon i DI Studio

## CAS Save Table

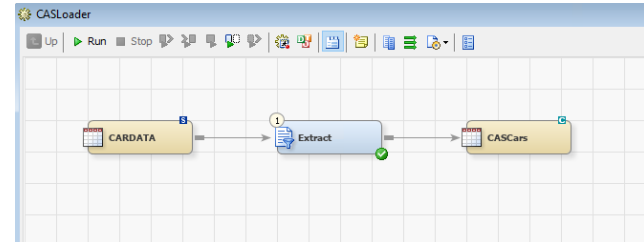
- Vi har en jobb som laster en SAS Base tabell fra SAS 9.4 til CAS vha en «extract»-node.



# Custom Transformasjon i DI Studio

## CAS Save Table

- Vi har en jobb som laster en SAS Base tabell fra SAS 9.4 til et PATH CASLIB vha en «extract»-node.

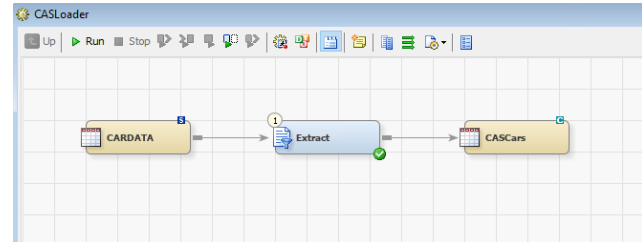


- Vi ønsker å også lagre en .sashdat kopi av tabellen i CASLIB-et

# Custom Transformasjon i DI Studio

## CAS Save Table

- Vi har en jobb som laster en SAS Base tabell fra SAS 9.4 til et PATH CASLIB vha en «extract»-node.



- Vi ønsker å også lagre en .sashdat kopi av tabellen i CASLIB-et
- På en så enkel og rask måte som mulig for ETL-utvikleren.

# Custom Transformasjon i DI Studio

## CAS Save Table

- En Custom Transformasjon som gjør dette med minst mulig innstillinger fra ETL-utvikler ble løsningen.

# Custom Transformasjon i DI Studio

## CAS Save Table

- En Custom Transformasjon som gjør dette med minst mulig innstillinger fra ETL-utvikler ble løsningen.
- En Custom Transformasjon er i prinsippet SAS Base kode pakket inn i en node (litt som UWC-transformasjonen), men den er persistent og man kan definere hvilke «Options» og hvilke «Input»/»Output»-noder utvikleren skal/kan sette.

# Custom Transformasjon i DI Studio

## CAS Save Table

- En Custom Transformasjon som gjør dette med minst mulig innstillinger fra ETL-utvikler ble løsningen.
- En Custom Transformasjon er i prinsippet SAS Base kode pakket inn i en node (litt som UWC-transformasjonen), men den er persistent og man kan definere hvilke «Options» og hvilke «Input»/»Output»-noder utvikleren skal/kan sette.
- Opsjonene man definerer blir til macrovariabler man kan bruke i SAS koden. Det samme blir input- og output-tabeller (&\_INPUT1., &\_INPUT2., &\_OUTPUT1. etc.)

# Custom Transformasjon i DI Studio

## CAS Save Table

- Vår transformasjon måtte gjøre noen oppslag mot SAS9.4 metadataserver for å finne CASLIB fra CASENGINE library på input-tabellen.
- Når den har funnet det er den klar til å kjøre en proc casutil save.

```
%let CASENGINELIB=%scan(&_INPUT_,1,.);
%let CASTABLE=%scan(&_INPUT_,2,.);
%let CASSESSREF=&CASENGINELIB_&CASTABLE.;
%let CASLIB=;
%let LIBREF=%tslit(&CASENGINELIB);
%let OMSQueryString=omsobj:SASLibrary?@Libref=%trim(&LIBREF.);
/* Find the Viya CASLIB name for the CAS Engine library of the input table */
data _null_;
    length name $256 uri $256;
    name="";
    uri="";
    rc=1;
    objrc=1;
    n=1;
    rc=metadata_getnasn("&OMSQueryString.", "Properties", n, uri);
    objrc=metadata_getattr(uri, "DefaultValue", name);
    call symput("CASLIB", name);
run;
proc casutil incaslib="&CASLIB." outcaslib="&CASLIB.";
    save casdata="&CASTABLE."
    replace;
quit;
```



# Custom Transformasjon i DI Studio

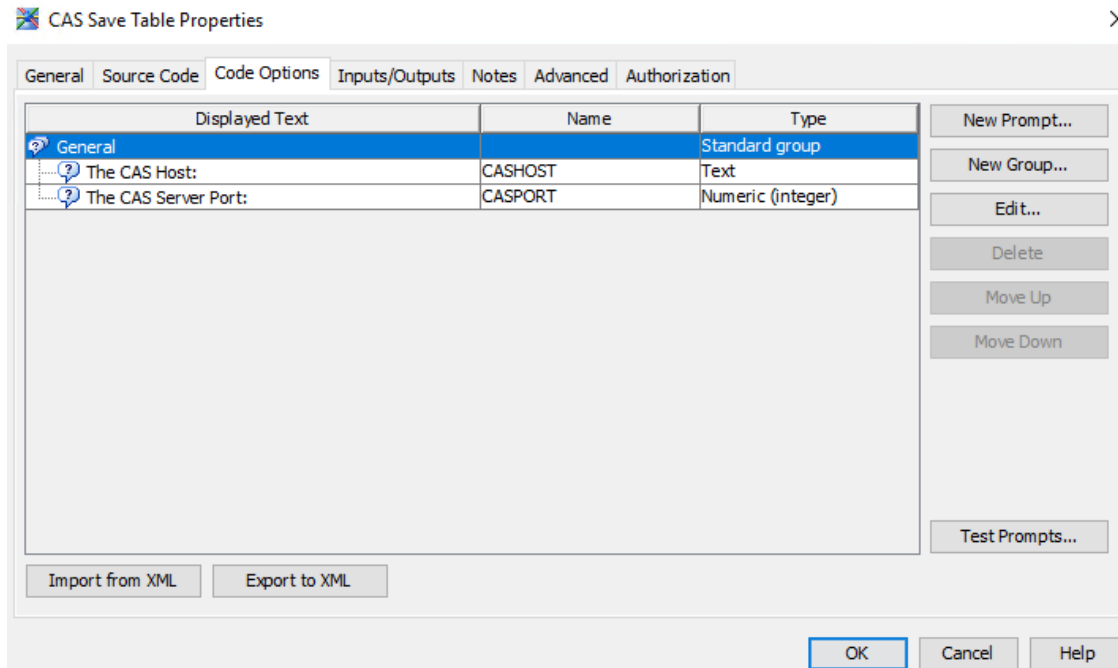
## CAS Save Table

- DEMO

# Custom Transformasjon i DI Studio

## CAS Save Table

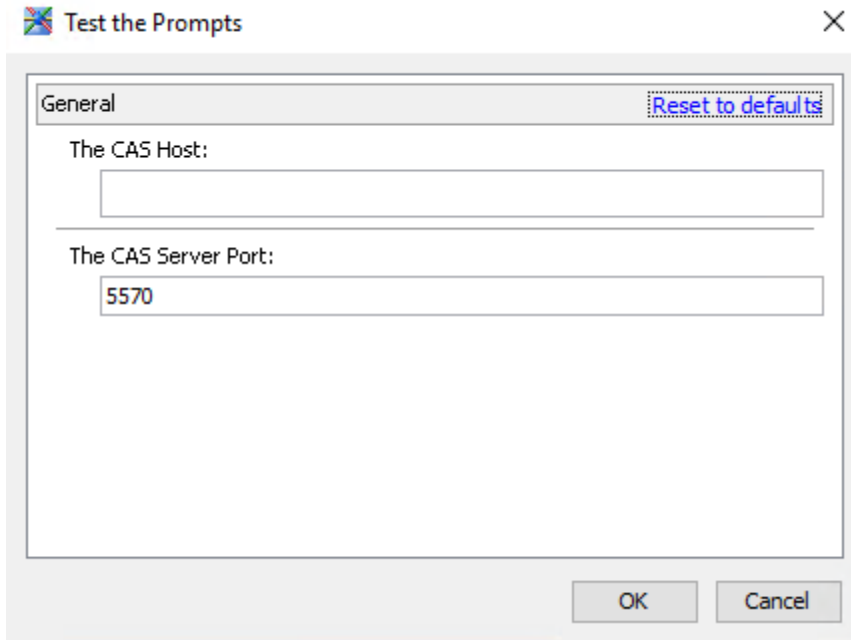
- Opsjonene defineres her:



# Custom Transformasjon i DI Studio

## CAS Save Table

- Man kan teste promptene for opsjonene:



Test the Prompts

General [Reset to defaults](#)

The CAS Host:

---

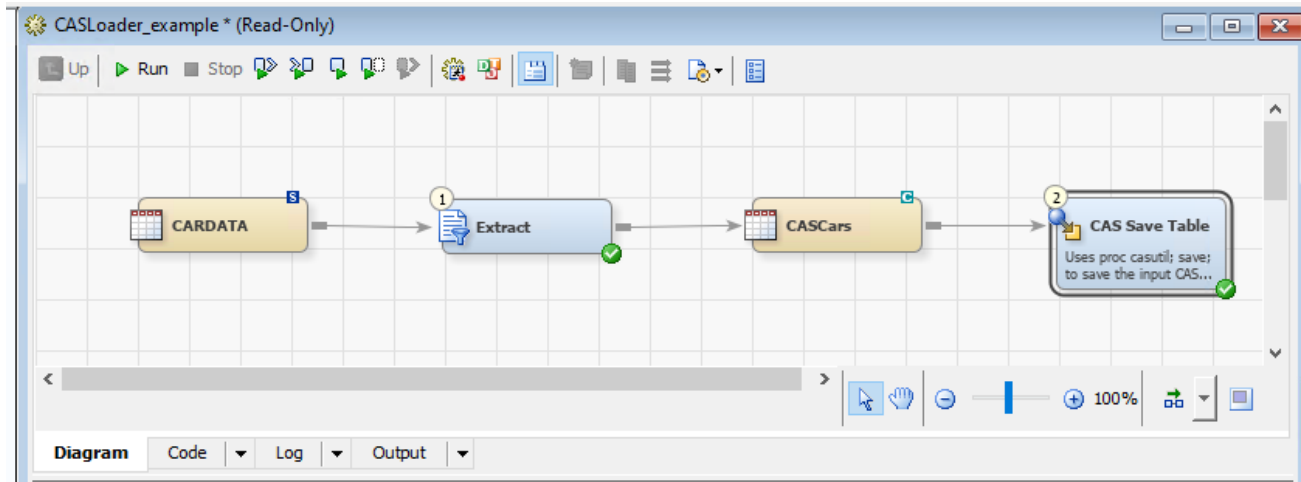
The CAS Server Port:

OK Cancel

# Custom Transformasjon i DI Studio

## CAS Save Table

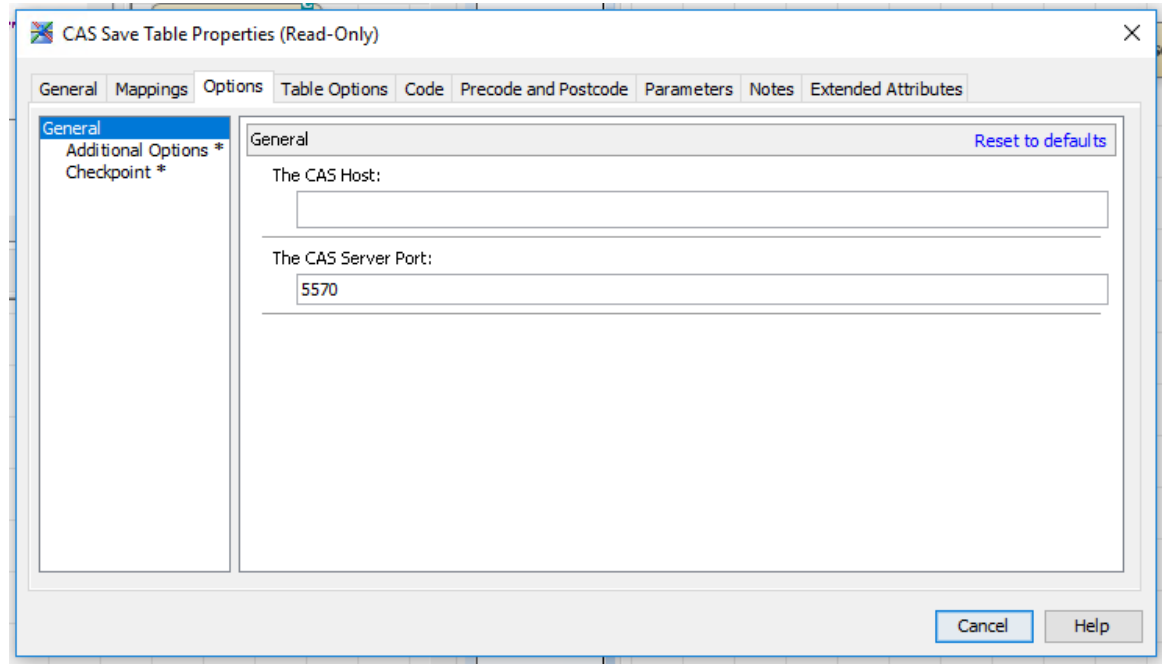
- Man drar «CAS Save Table»-transformasjonen inn i jobben man lager og kobler den på flyten slik som med en vanlig transformasjon.



# Custom Transformasjon i DI Studio

## CAS Save Table

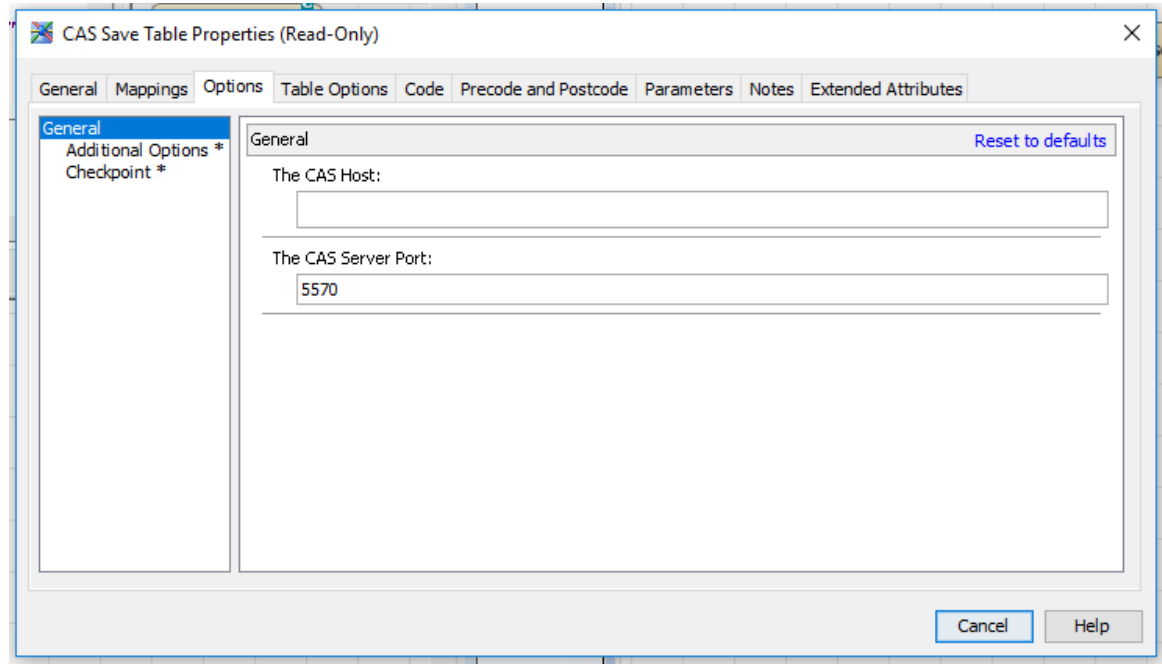
- Når man bruker transformasjonen i en DI Studio jobb så ligger opsjonene her:



# Custom Transformasjon i DI Studio

## CAS Save Table

- Når man bruker transformasjonen i en DI Studio jobb så ligger opsjonene her:



# Custom Transformasjon i DI Studio

## CAS Save Table

NB!

Når man legger til et custom transformasjon objekt i en jobb så blir en kopi av custom transformasjonen lagt til i jobben.

Det vil si at påfølgende endringer i Custom Transformasjonsobjektet *ikke* vil medføre endringer i eksisterende jobber som bruker den transformasjonen.

Da må man evt. fjerne og legge til transformasjonen på nytt i alle jobbene dette gjelder.

# SAS Studio Flow

- Hvordan gjøre tilsvarende på Viya?



# SAS Studio Flow

- Hvordan gjøre tilsvarende på Viya?
- SAS Studio Flow for å lage ETL-flyter som i DI Studio

# SAS Studio Flow

- Hvordan gjøre tilsvarende på Viya?
- SAS Studio Flow for å lage ETL-flyter som i DI Studio
- Custom Step i SAS Studio Flow er motstykke til Custom Transformasjoner i DIS

# SAS Studio Flow

- Hvordan gjøre tilsvarende på Viya?
- SAS Studio Flow for å lage ETL-flyter som i DI Studio
- Custom Step i SAS Studio Flow er motstykke til Custom Transformasjoner i DIS
- Kom med Viya 2020.1.5 (som kom i April 2021)
  - Denne hadde ikke en prompt UI designer og man måtte skrive json for å definere prompt brukergrensesnittet for eventuelle opsjoner som brukeren av custom step-et setter.

# SAS Studio Flow

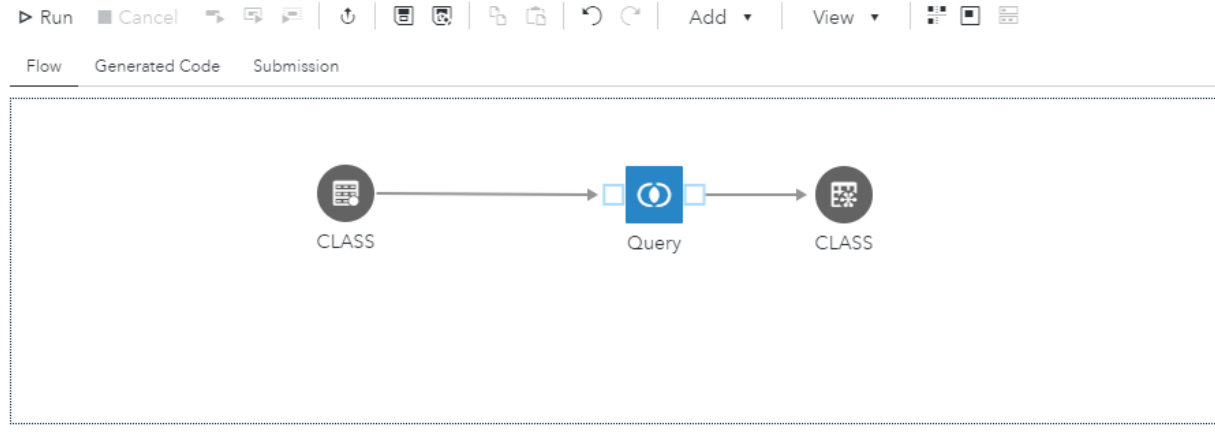
- Hvordan gjøre tilsvarende på Viya?
- SAS Studio Flow for å lage ETL-flyter som i DI Studio
- Custom Step i SAS Studio Flow er motstykke til Custom Transformasjoner i DIS
- Kom med Viya 2020.1.5 (som kom i April 2021)
  - Denne hadde ikke en prompt UI designer og man måtte skrive json for å definere prompt brukergrensesnittet for eventuelle opsjoner som brukeren av custom step-et setter.
- Krever en SAS Studio Analyst lisens.

# SAS Studio Flow

- Hvordan gjøre tilsvarende på Viya?
- SAS Studio Flow for å lage ETL-flyter som i DI Studio
- Custom Step i SAS Studio Flow er motstykke til Custom Transformasjoner i DIS
- Kom med Viya 2020.1.5 (som kom i April 2021)
  - Denne hadde ikke en prompt UI designer og man måtte skrive json for å definere prompt brukergrensesnittet for eventuelle opsjoner som brukeren av custom step-et setter.
- Krever en SAS Studio Analyst lisens.
- Skjermbilder og demo i dag er på Viya 2021.1.5
  - Denne har prompt UI designer der man kan bruke «drag-n-drop» av forskjellige typer input-felter og ledetekster for å lage brukergrensesnittet.


# SAS Studio Flow

- Vi starter med et flow som bruker Query-steget til å laste en SAS tabell (CLASS) til CASUSER caslib-et.



# Custom Step i SAS Studio Flow

## SaveCAS

- Enten via New meny i SAS Studio eller via Steps og  - ikonet.
- Jeg har brukt «Custom Step Quick start», men det finne også noen få start-templater som utgangspunkt.

# Custom Step i SAS Studio Flow

## SaveCAS

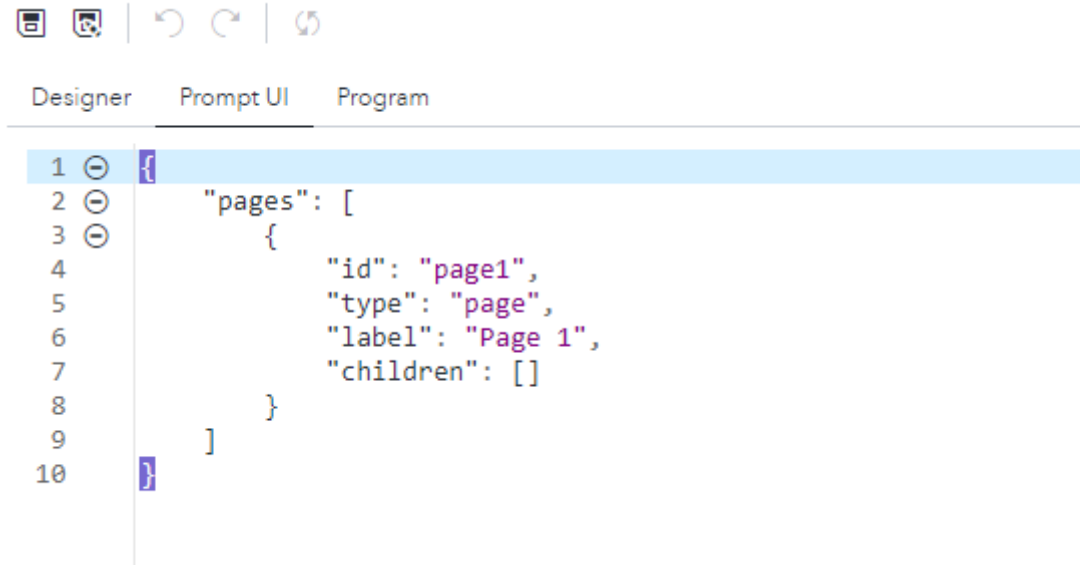
The screenshot displays the SAS Studio Flow Designer interface. At the top, there are navigation icons and a 'Preview' button. Below the navigation bar, the 'Designer' tab is active, showing a central canvas with a message: 'Add controls to the canvas to build your step.' To the left of the canvas is the 'Control Library' panel, which includes an 'Add Page' button, a list containing 'Page 1', a search box labeled 'Filter controls', and a list of control types: Section, Text, Text Area, Text or Numeric Field, Column Selector, Input Table, New Column, and Output Table. To the right of the canvas is the 'Properties: Page' panel, which contains two input fields: 'ID: \*' with the value 'page1' and 'Label: \*' with the value 'Page 1'. At the bottom of the interface, there is a status bar showing 'Recover (1)' and 'Submission (0)'.



# Custom Step i SAS Studio Flow

## SaveCAS

- JSON til prompt UI-et



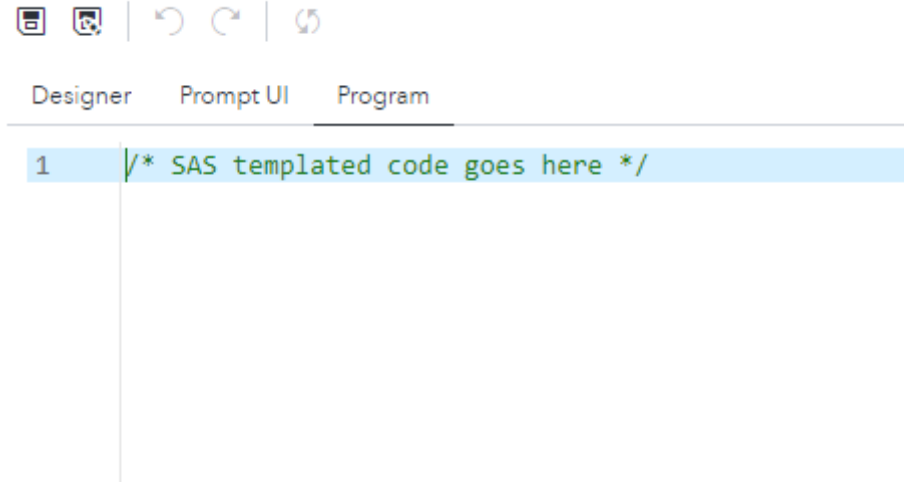
The screenshot shows the SAS Studio interface with the 'Prompt UI' tab selected. The editor displays a JSON object with a 'pages' array containing one page object. The code is as follows:

```
1 {
2   "pages": [
3     {
4       "id": "page1",
5       "type": "page",
6       "label": "Page 1",
7       "children": []
8     }
9   ]
10 }
```

# Custom Step i SAS Studio Flow

## SaveCAS

- Kode-templatet til custom step-et.



The screenshot shows the SAS Studio interface. At the top, there are icons for file operations (save, open, refresh) and a navigation bar with three tabs: "Designer", "Prompt UI", and "Program". The "Program" tab is active. Below the tabs is a code editor window. The first line of code is highlighted in light blue and contains the comment: `1 /* SAS templated code goes here */`. A vertical line is visible on the left side of the code editor, indicating the current cursor position.

# Custom Step i SAS Studio Flow

## SaveCAS

- Definerer først promptene våre

The screenshot displays the SAS Studio Flow Designer interface. At the top, there are navigation icons and tabs for 'Designer', 'Prompt UI', and 'Program'. The main workspace is divided into three vertical panels:

- Control Library:** Located on the left, it contains a search bar labeled 'Filter controls' and a list of common controls: Check Box, Drop-down List, Numeric Stepper, Radio Button Group, Section, Text, Text Area, and Text or Numeric Field. The 'Opsjoner' control is currently selected.
- Step Configuration:** The central panel shows the configuration for the 'SaveCAS' step. It features a label 'Input table label 1 \*' above a text input field containing 'page1'. Below this is a checkbox labeled 'Print file size of table to log', which is currently unchecked.
- Properties Panel:** On the right, the 'Properties: Page' panel shows the 'ID:' field set to 'page1' and the 'Label:' field set to 'Opsjoner'.

# Custom Step i SAS Studio Flow

## SaveCAS

- Så legger vi inn kodetemplatet



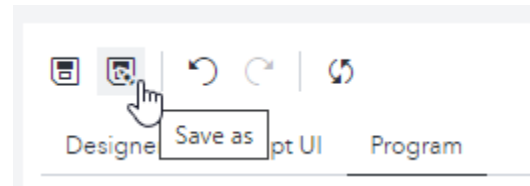
The screenshot shows the SAS Studio interface with the 'Program' tab selected. The code editor contains the following SAS code template:

```
1 /* SAS templated code goes here */
2
3 proc casutil incaslib="&inputtable1_lib." outcaslib="&inputtable1_lib.";
4     save casdata="&inputtable1_name."
5     replace;
6 quit;
7
8 %if &LogFlag = 1 %then %do;
9     %put "Logging size";
10
11 proc cas;
12     table.fileinfo result=res status=rc /
13     caslib="&inputtable1_lib.",
14     kbytes=true,
15     path="&inputtable1_name..sashdat";
16     run;
17     *describe res;
18     output log;
19     print "Filesize &inputtable1: " res.FileInfo[1,"Size"] "kB";
20     output ods;
21 quit;
22 %end;
```

# Custom Step i SAS Studio Flow

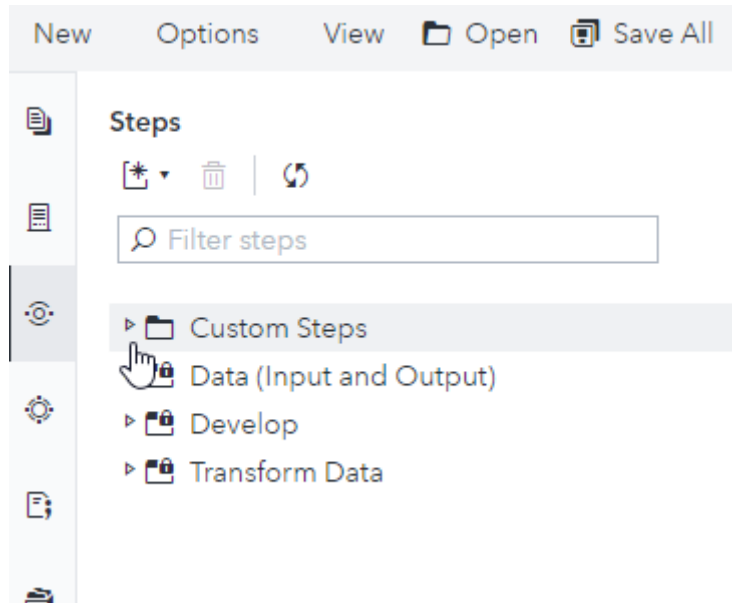
## SaveCAS

- Lagrer Custom Step-et et fornuftig sted der andre utviklere har lesetilgang



# Custom Step i SAS Studio Flow SaveCAS

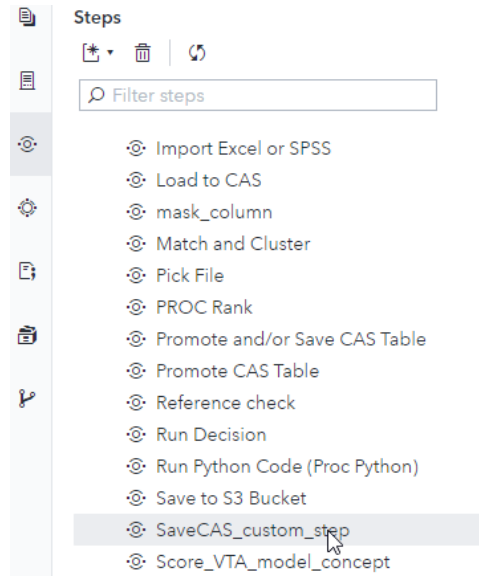
- Det dukker opp i Custom Step mappen i Step-seksjonen



# Custom Step i SAS Studio Flow

## SaveCAS

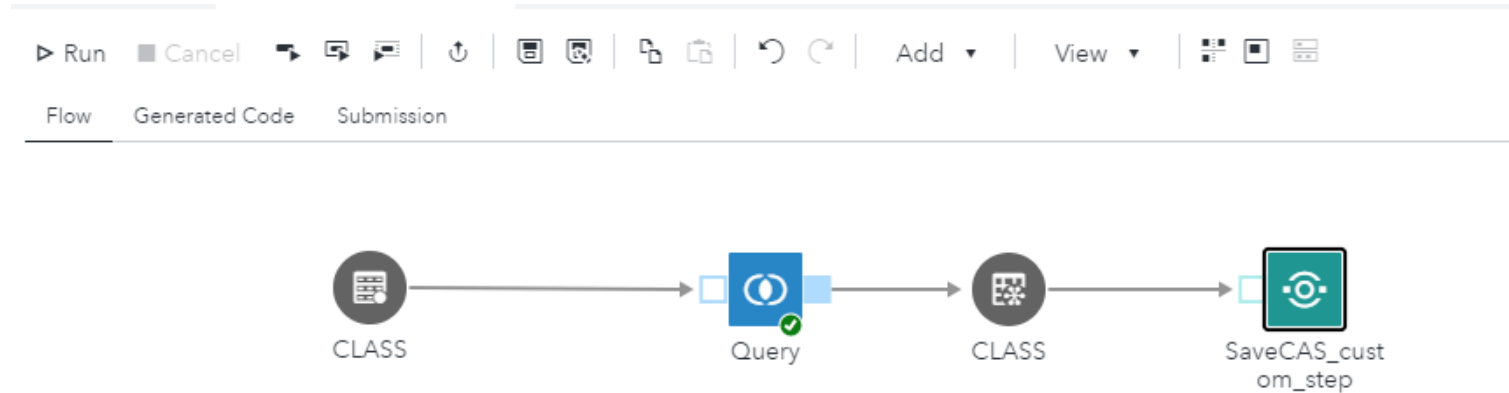
- Man plukker opp vårt Custom Step og legger det til i flow-et vårt



# Custom Step i SAS Studio Flow

## SaveCAS

- Man plukker opp vårt Custom Step og legger det til i flow-et vårt





# Custom Step i SAS Studio Flow

## SaveCAS

- NB!  
Custom Step blir referert fra flow-ene.  
Det vil si at endringer i et Custom Step vil slå til i eksisterende Flows.

# Resources

- <https://communities.sas.com/t5/SAS-Communities-Library/My-First-SAS-Studio-Flow-Custom-Step/ta-p/740204>
- Viya 2021.1.5 Working With Custom Step dokumentasjon:  
[https://go.documentation.sas.com/doc/en/sasstudiocdc/v\\_012/webeditorcdc/webeditorug/n0h4ijiwhbk8uwn15hng2vhepepa.htm](https://go.documentation.sas.com/doc/en/sasstudiocdc/v_012/webeditorcdc/webeditorug/n0h4ijiwhbk8uwn15hng2vhepepa.htm)



# Noen spørsmål?

Tom Arne Orthe <[tomarne.orth@sas.com](mailto:tomarne.orth@sas.com)>

[sas.com](https://sas.com)