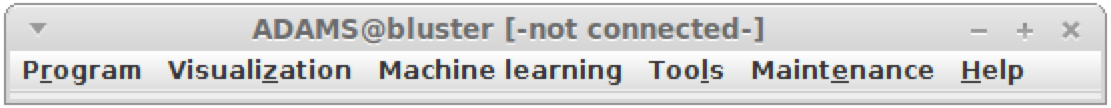
THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

Big Data with ADAMS

The Basics

Peter Reutemann

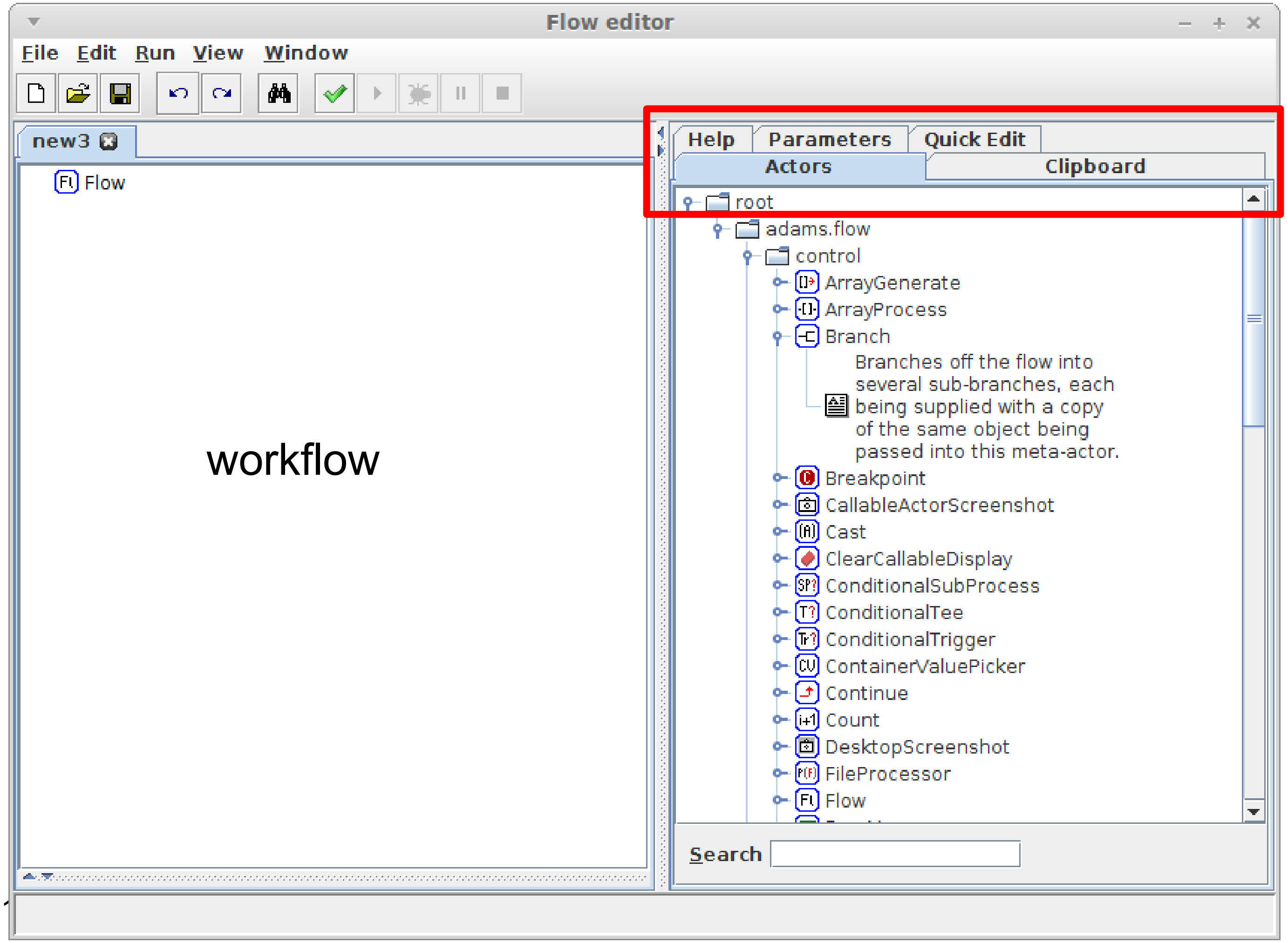
The Interface



- Main menu launches GUIs of tools (eg WEKA)
- “User mode” defines what menu items are visible
- Additional tools for Weka (eg dataset handling)
- Flows
  - edited/executed/debugged with Flow editor
  - executed with Flow runner
  - executed from command-line (eg daemon/service)
    adams.flow.FlowRunner

Flow editor

- Split view: flow and tabs

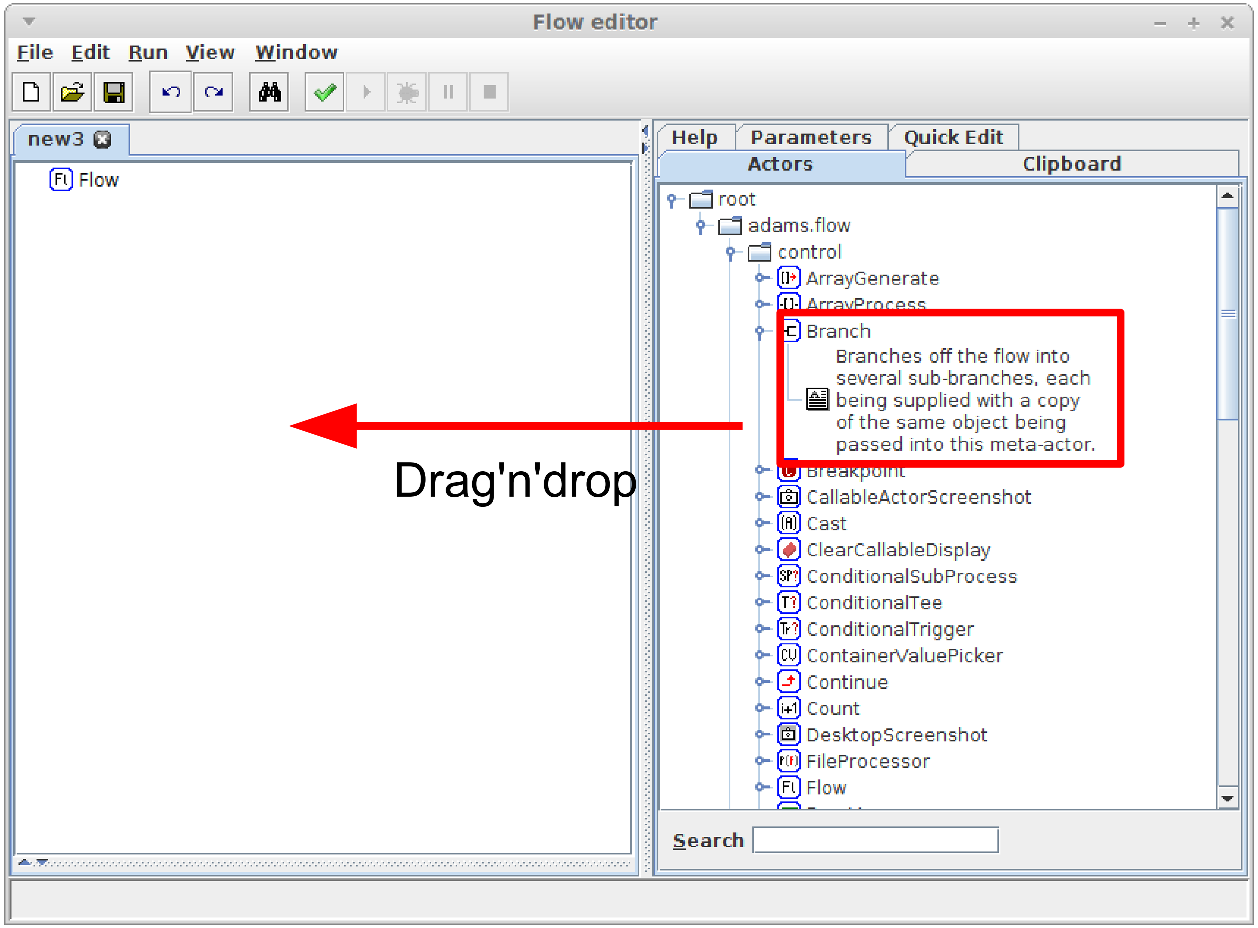


Branches off the flow into several sub-branches, each being supplied with a copy of the same object being passed into this meta-actor.

workflow

optional tabs

Adding actors



Flow

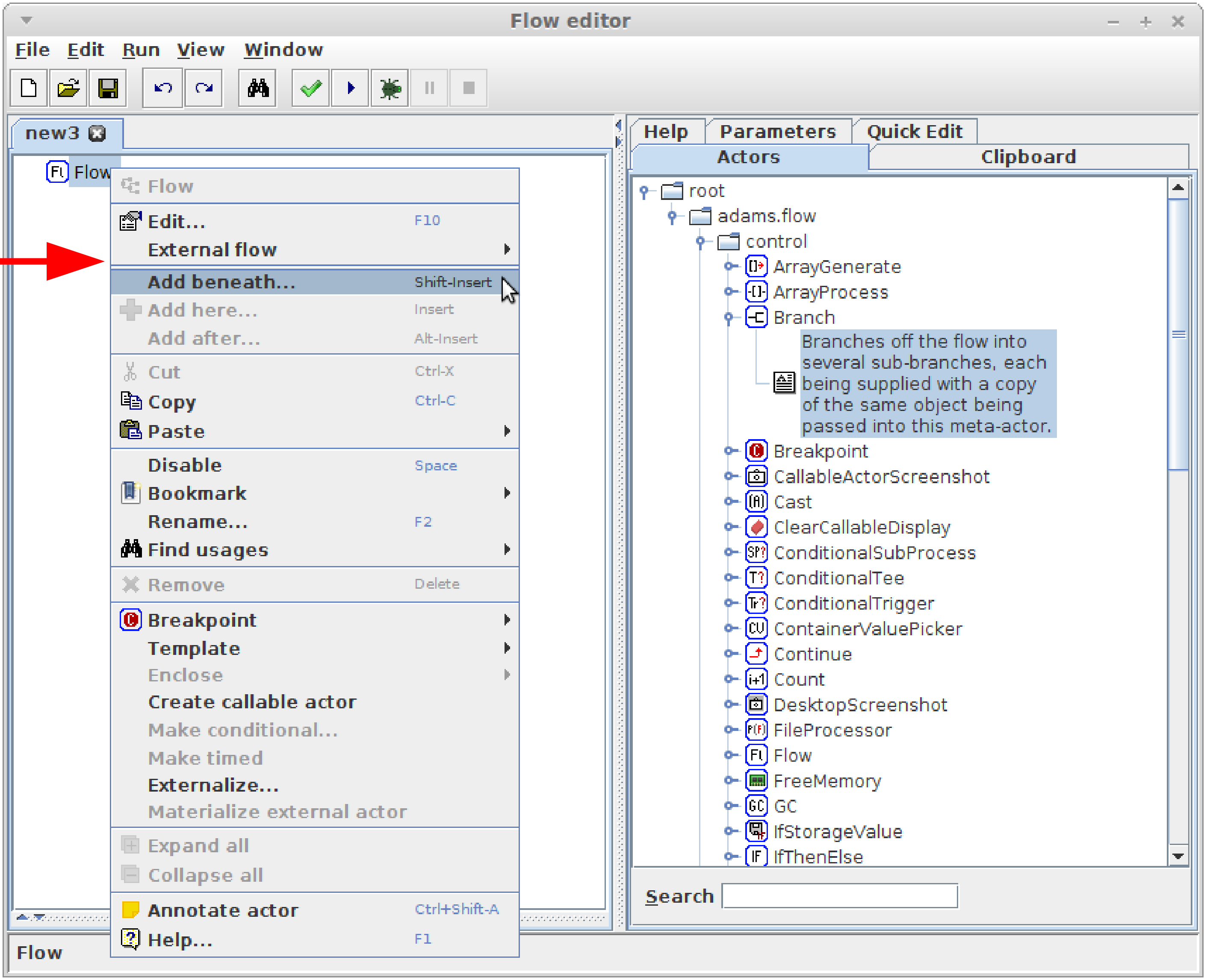
- root
  - adams.flow
    - control
      - ArrayGenerate
      - ArrayProcess
      - Branch
        - Branches off the flow into several sub-branches, each being supplied with a copy of the same object being passed into this meta-actor.
      - Breakpoint
      - CallableActorScreenshot
      - Cast
      - ClearCallableDisplay
      - ConditionalSubProcess
      - ConditionalTee
      - ConditionalTrigger
      - ContainerValuePicker
      - Continue
      - Count
      - DesktopScreenshot
      - FileProcessor
      - Flow

Drag'n'drop

Short description

Adding actors (2)

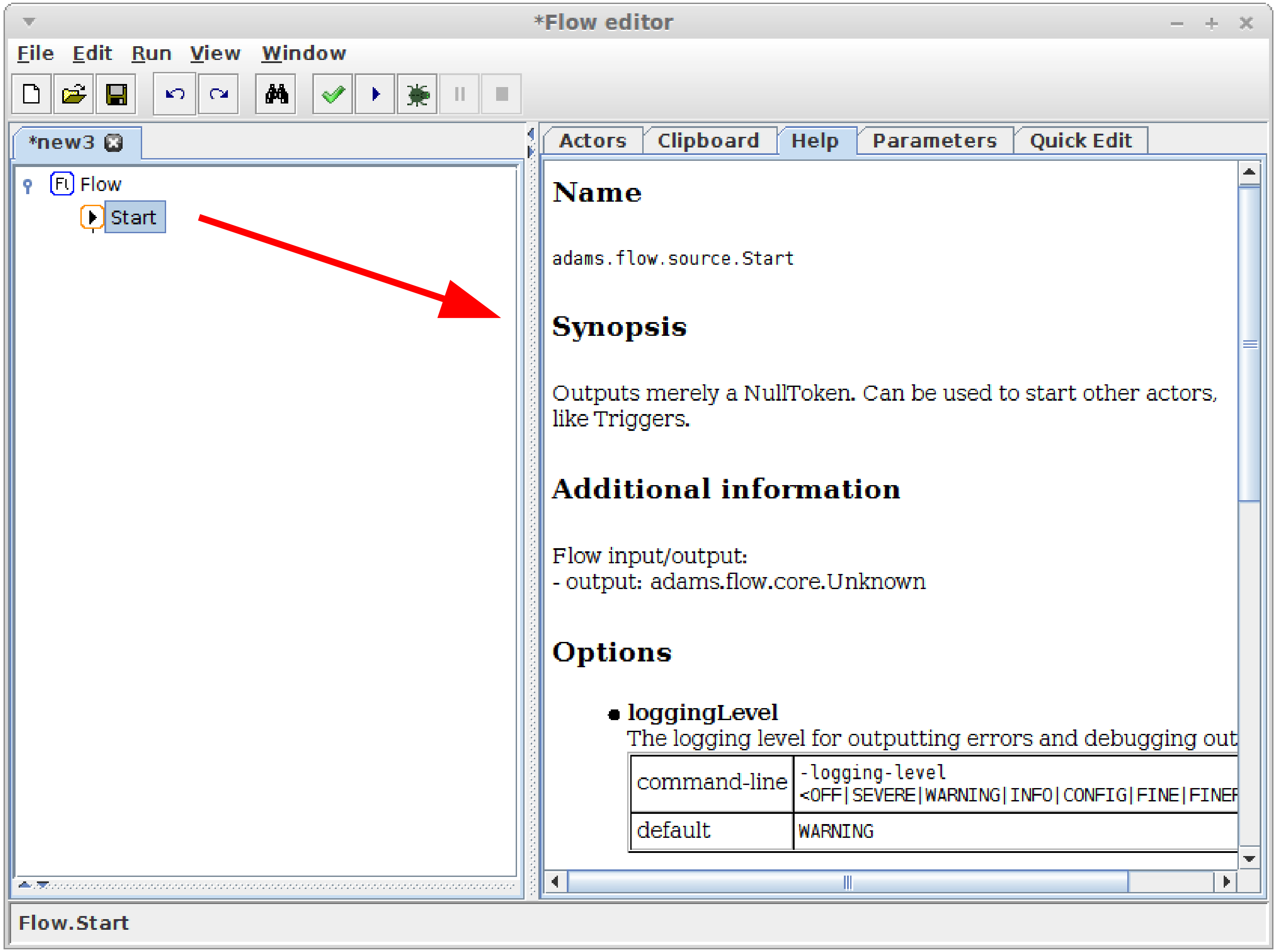
Context-sensitive popup menu



Branches off the flow into several sub-branches, each being supplied with a copy of the same object being passed into this meta-actor.

Actor help

online help
whenever actor
gets selected


*Flow editor
File Edit Run View Window
*new3
Flow
Start
Actors Clipboard Help Parameters Quick Edit
Flow.Start

Name

adams.flow.source.Start

Synopsis

Outputs merely a NullToken. Can be used to start other actors, like Triggers.

Additional information

Flow input/output:
- output: adams.flow.core.Unknown

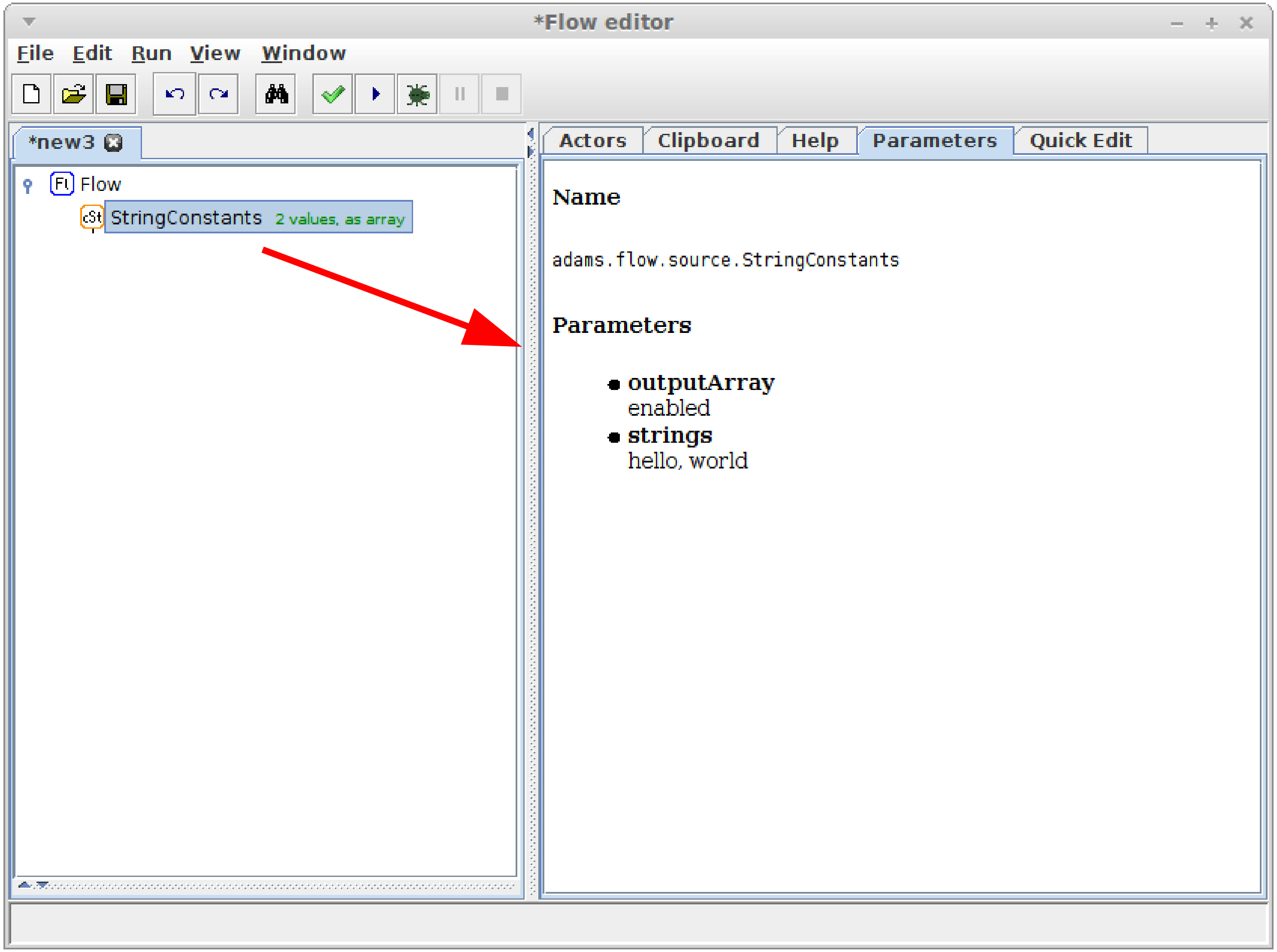
Options

- **loggingLevel**
  The logging level for outputting errors and debugging out

command-line	-logging-level <OFF|SEVERE|WARNING|INFO|CONFIG|FINE|FINEF
default	WARNING

Actor parameters

display of non-default parameters, whenever actor gets selected



Name

adams.flow.source.StringConstants

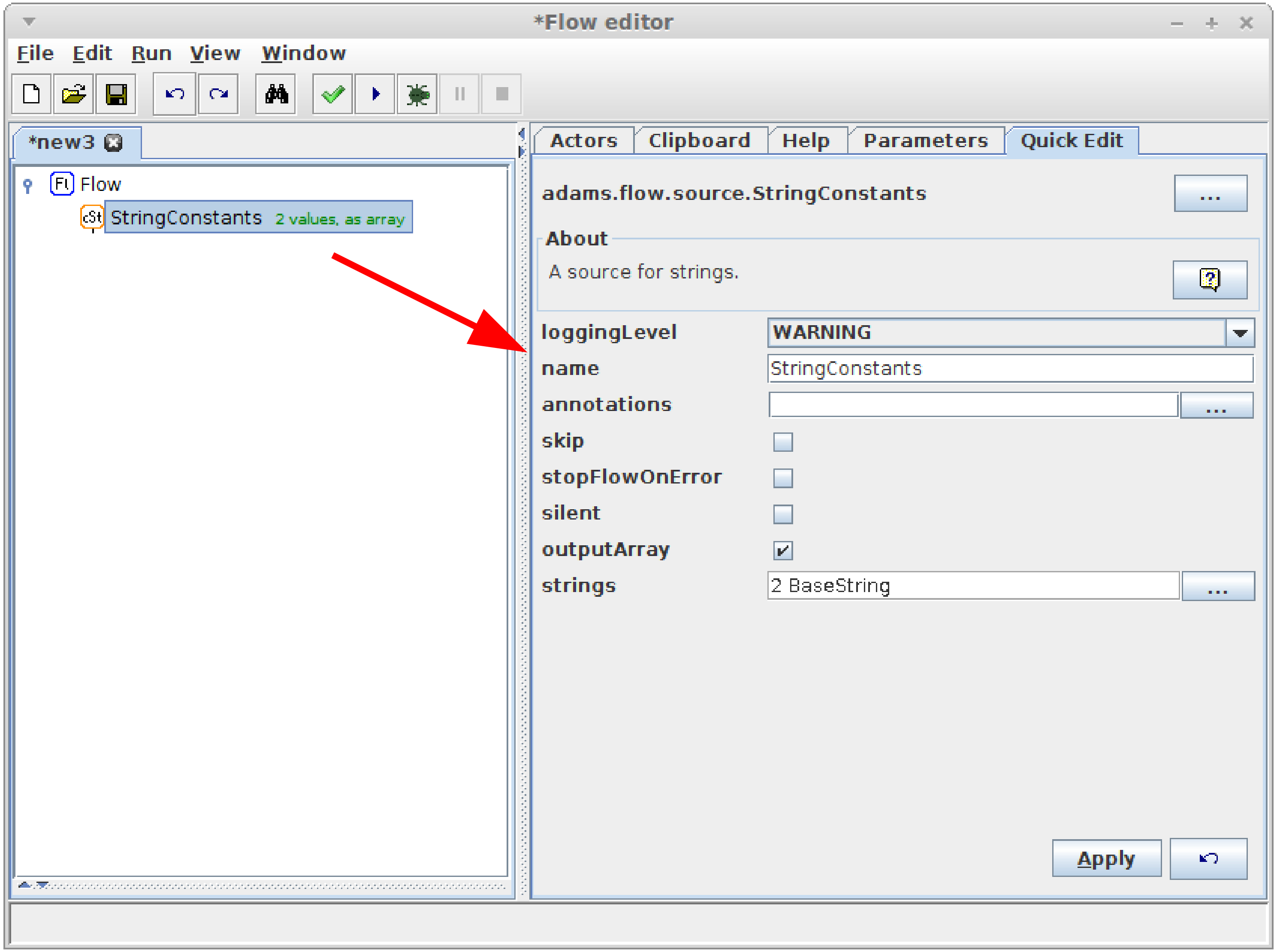
Parameters

- **outputArray**
  enabled
- **strings**
  hello, world

Actor quick edit

quickly changing parameters without bringing up edit dialog, whenever actor gets selected

NB: expensive



Flow
StringConstants 2 values, as array

adams.flow.source.StringConstants

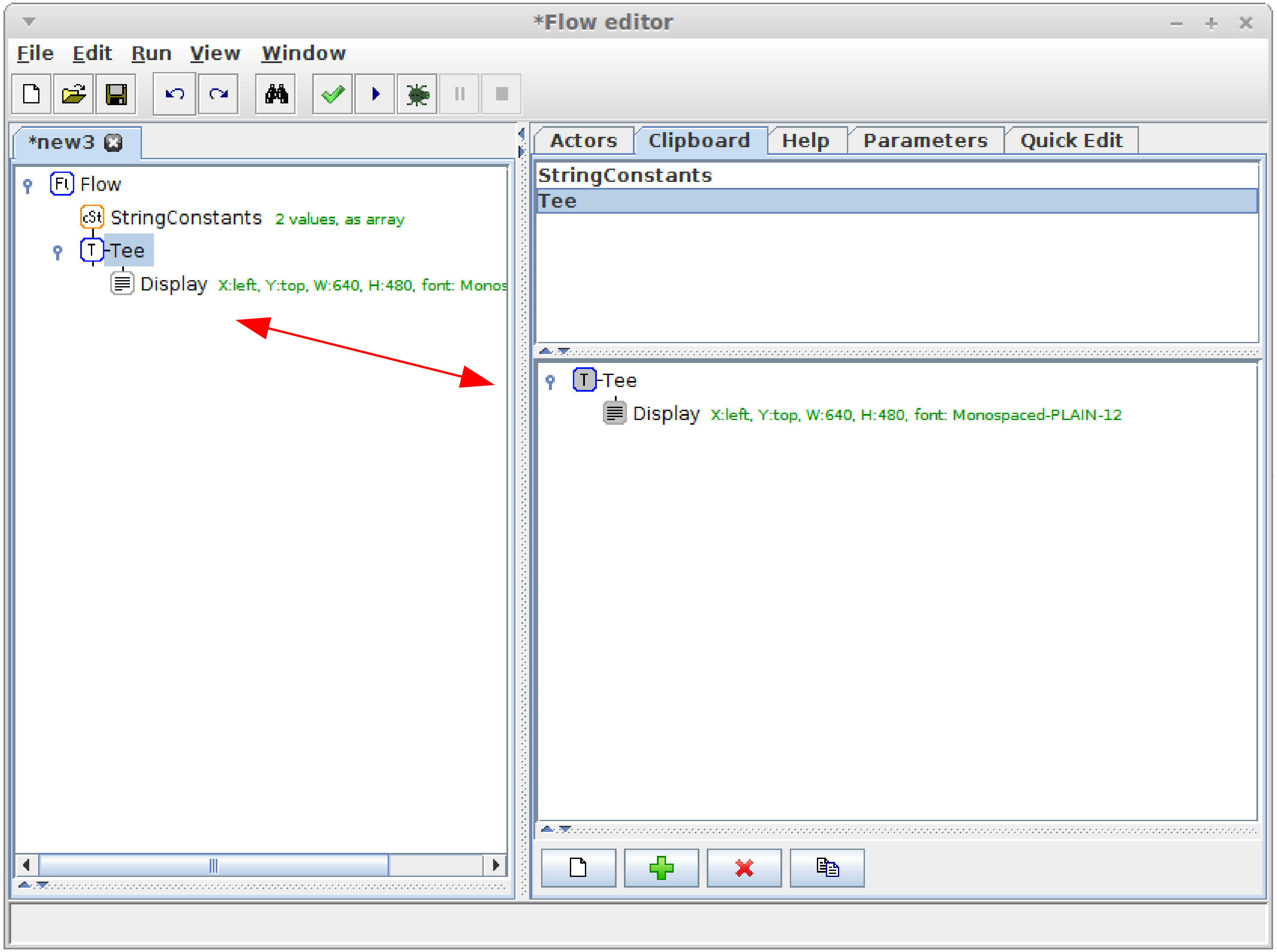
About

A source for strings.

loggingLevel	WARNING
name	StringConstants
annotations	
skip	☐
stopFlowOnError	☐
silent	☐
outputArray	☑
strings	2 BaseString

Clipboard

allows temporary storage of multiple subflows



```
Flow
    StringConstants  2 values, as array
    Tee
        Display  X:left, Y:top, W:640, H:480, font: Monos
```

Clipboard:

```
StringConstants
Tee
```

```
Tee
    Display  X:left, Y:top, W:640, H:480, font: Monospaced-PLAIN-12
```

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

“Small” Data with ADAMS

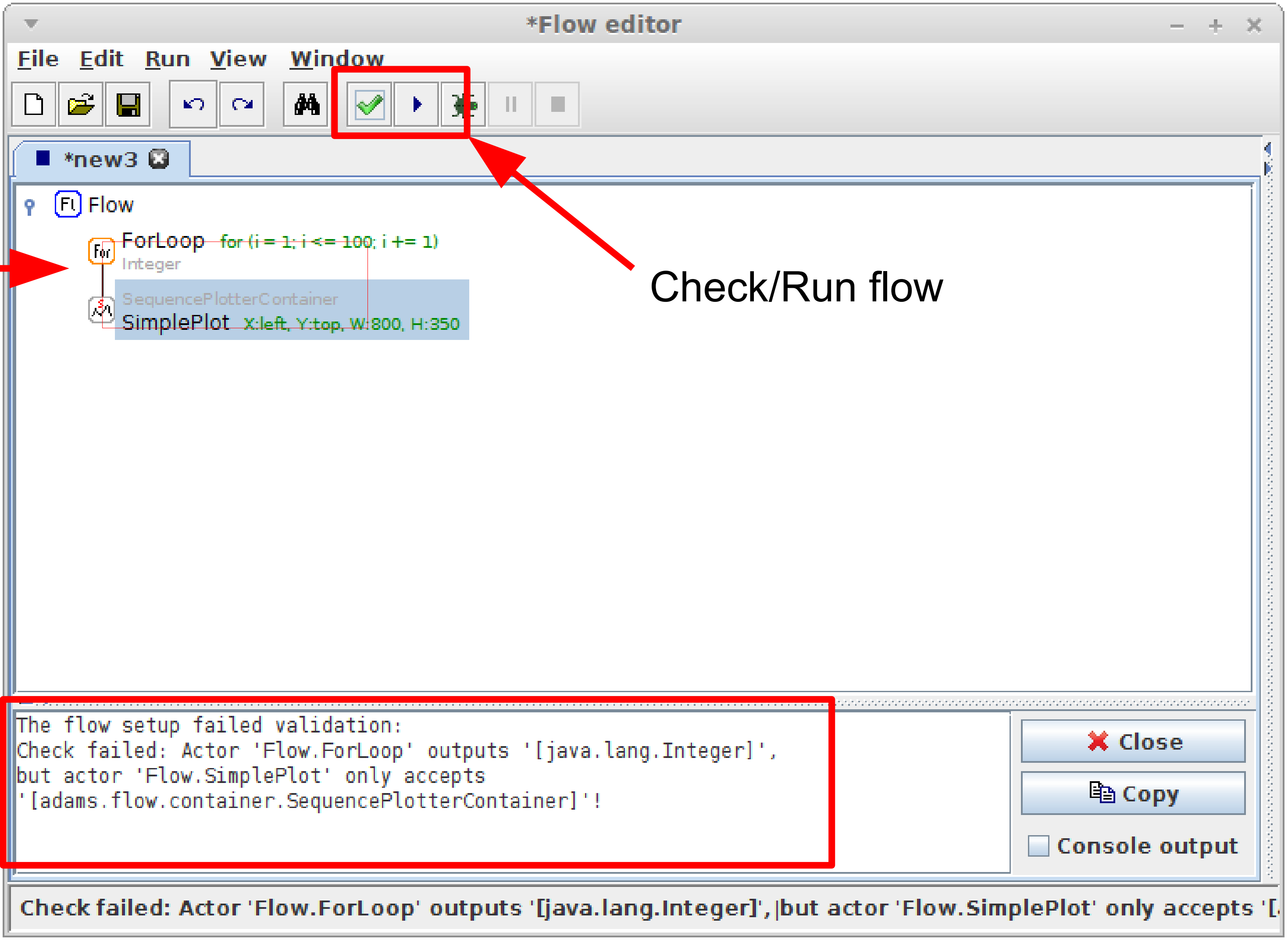
Plotting, Variables,
Callable actors,
Interactive flows

Peter Reutemann

Plotting

- What we need
  - generate some numbers
  - turn them into a plot
- What we use
  - For ForLoop source
  - SimplePlot sink

Plotting



```
The flow setup failed validation:
Check failed: Actor 'Flow.ForLoop' outputs '[java.lang.Integer]',
but actor 'Flow.SimplePlot' only accepts
'[adams.flow.container.SequencePlotterContainer]'!
```

Check failed: Actor 'Flow.ForLoop' outputs '[java.lang.Integer]', |but actor 'Flow.SimplePlot' only accepts '[

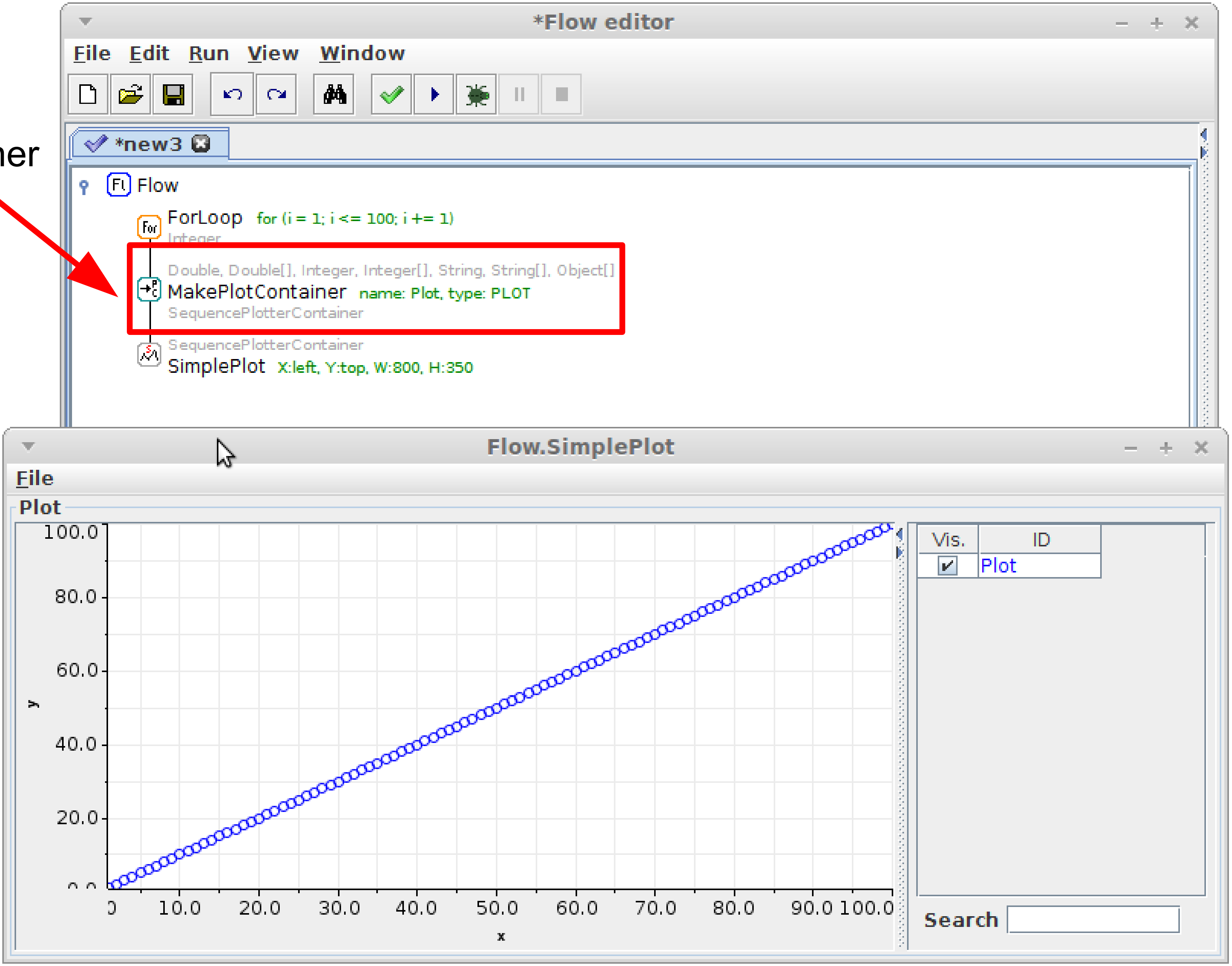
Check/Run flow

Mismatch of input/output *[View → Show input/output]*

Error message

Plotting - fixed

add transformer:
MakePlotContainer



*Flow editor

*new3

Flow
ForLoop for (i = 1; i <= 100; i += 1)
Integer
Double, Double[], Integer, Integer[], String, String[], Object[]
MakePlotContainer name: Plot, type: PLOT
SequencePlotterContainer
SequencePlotterContainer
SimplePlot X:left, Y:top, W:800, H:350

Flow.SimplePlot

Vis.	ID
☑	Plot

Plotting X/Y

- let's plot a sawtooth function

$$x(t)=2\left(\frac{t}{a}-floor\left(\frac{1}{2}+\frac{t}{a}\right)\right); with\ a\ being\ the\ period$$

- how to implement the function?

  MathExpression transformer

- formula

  2*(X/2-floor(0.5 + X/2)) using period of 2

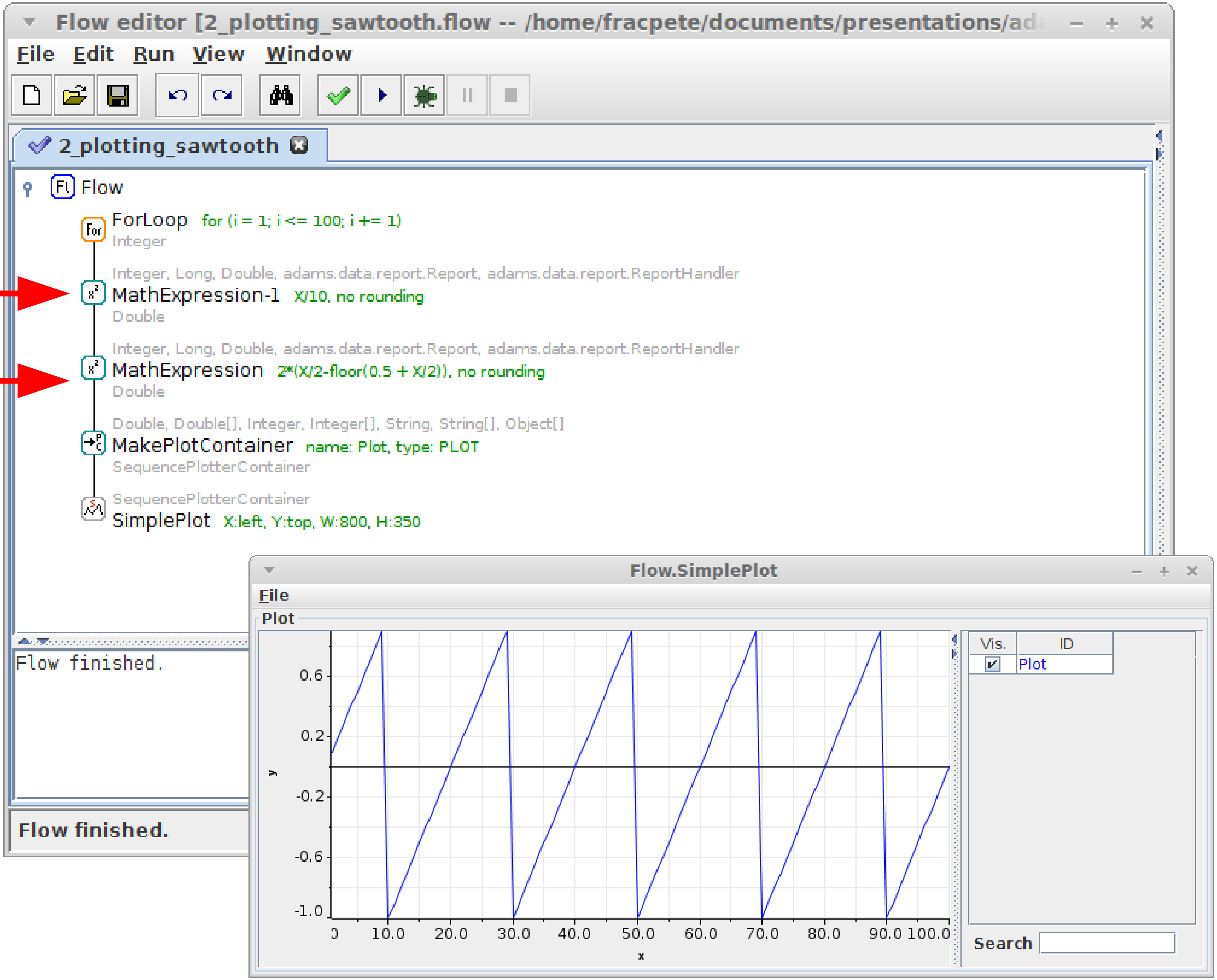
- another MathExpression to create fractions

  X/10

Plotting X/Y

create fraction

sawtooth



Flow editor [2_plotting_sawtooth.flow -- /home/fracpete/documents/presentations/ad

2_plotting_sawtooth

```
Flow
    ForLoop  for (i = 1; i <= 100; i += 1)
    Integer
    Integer, Long, Double, adams.data.report.Report, adams.data.report.ReportHandler
    MathExpression-1  X/10, no rounding
    Double
    Integer, Long, Double, adams.data.report.Report, adams.data.report.ReportHandler
    MathExpression  2*(X/2-floor(0.5 + X/2)), no rounding
    Double
    Double, Double[], Integer, Integer[], String, String[], Object[]
    MakePlotContainer  name: Plot, type: PLOT
    SequencePlotterContainer
    SequencePlotterContainer
    SimplePlot  X:left, Y:top, W:800, H:350
```

```
Flow finished.
```

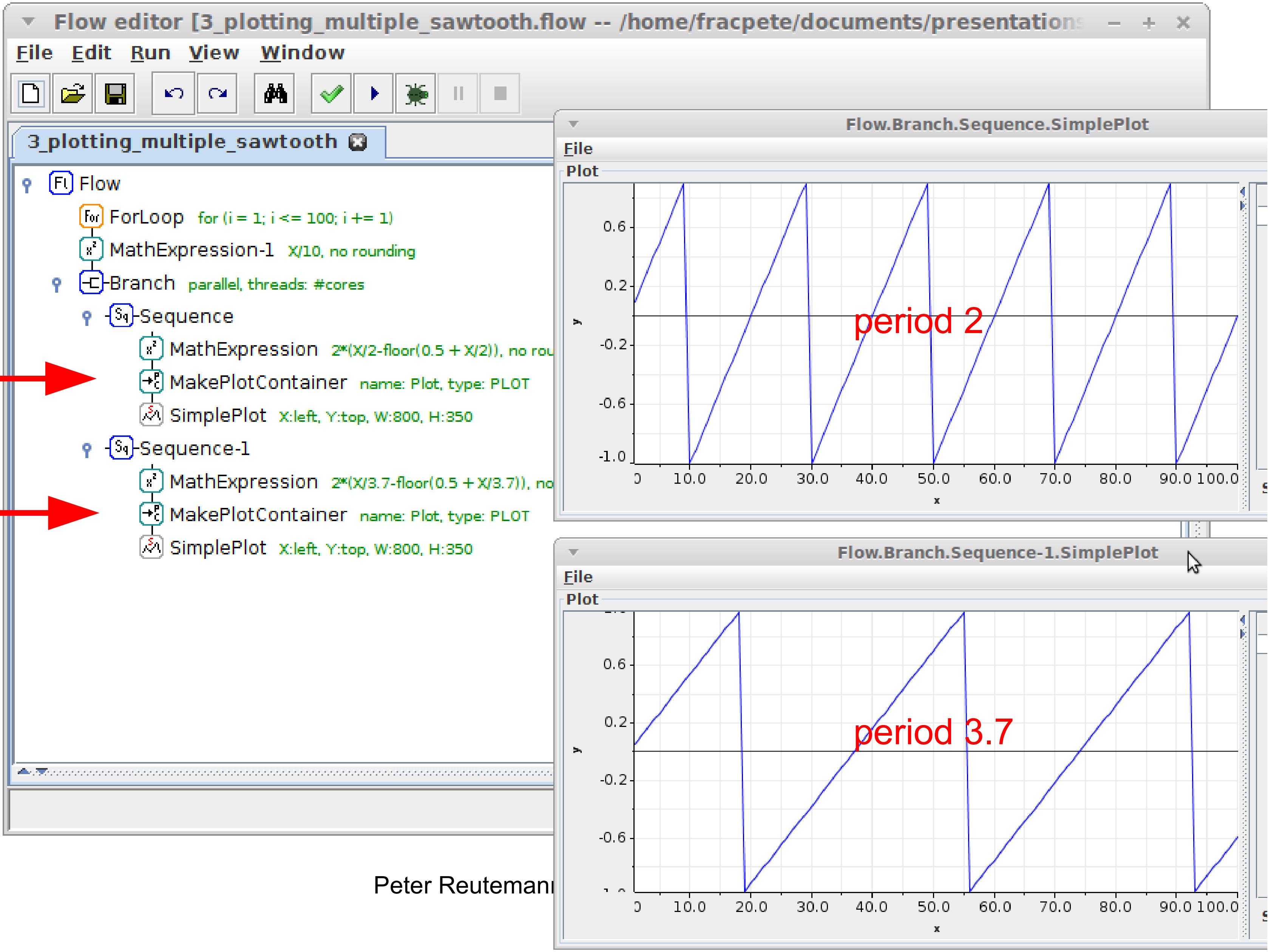
Flow finished.

Flow.SimplePlot

Multiple plots

- Let's create multiple sawtooth plots
- use period 2 and 3.7
- use ⊏ Branch actor
- combine steps in Sq Sequence actor

Multiple plots



Flow editor [3_plotting_multiple_sawtooth.flow -- /home/fracpete/documents/presentations

3_plotting_multiple_sawtooth

```
Flow
    ForLoop  for (i = 1; i <= 100; i += 1)
    MathExpression-1  X/10, no rounding
    Branch  parallel, threads: #cores
        Sequence
            MathExpression  2*(X/2-floor(0.5 + X/2)), no rou
            MakePlotContainer  name: Plot, type: PLOT
            SimplePlot  X:left, Y:top, W:800, H:350
        Sequence-1
            MathExpression  2*(X/3.7-floor(0.5 + X/3.7)), no
            MakePlotContainer  name: Plot, type: PLOT
            SimplePlot  X:left, Y:top, W:800, H:350
```

period 2

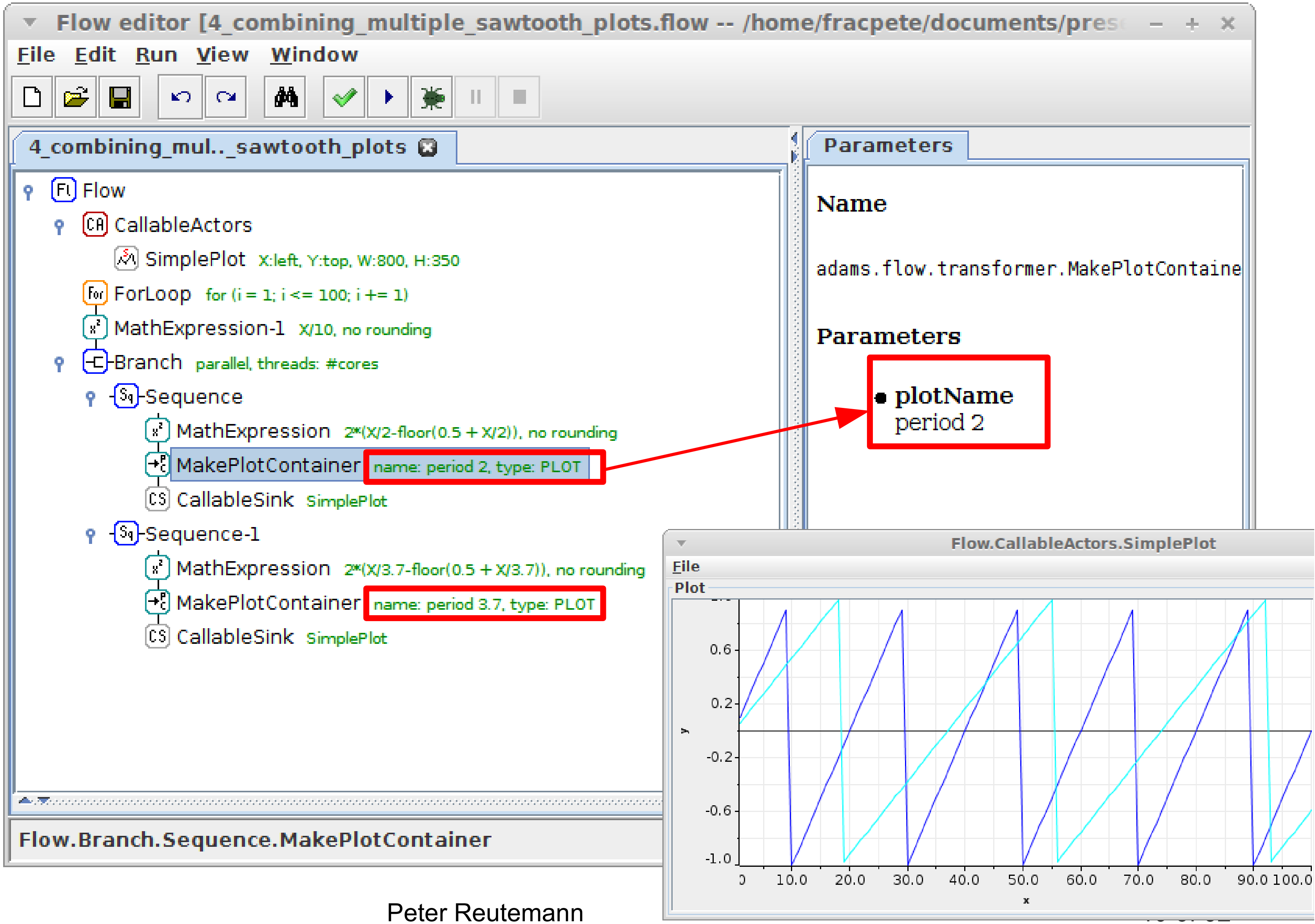
period 3.7

Combining plots

- Multiple plots are great, but...
- Combine them in single plot window?
- Make use of “callable actors”
  - handler: CA CallableActors
  - source: CS CallableSource
  - transformer: CT CallableTransformer
  - sink: CS CallableSink

Combining plots

use different plot names to separate plots in graph



Flow editor [4_combining_multiple_sawtooth_plots.flow -- /home/fracpete/documents/pres

4_combining_mul.._sawtooth_plots

```
Flow
  CallableActors
    SimplePlot X:left, Y:top, W:800, H:350
  ForLoop for (i = 1; i <= 100; i += 1)
  MathExpression-1 X/10, no rounding
  Branch parallel, threads: #cores
    Sequence
      MathExpression 2*(X/2-floor(0.5 + X/2)), no rounding
      MakePlotContainer name: period 2, type: PLOT
      CallableSink SimplePlot
    Sequence-1
      MathExpression 2*(X/3.7-floor(0.5 + X/3.7)), no rounding
      MakePlotContainer name: period 3.7, type: PLOT
      CallableSink SimplePlot
```

Flow.Branch.Sequence.MakePlotContainer

Parameters

Name

adams.flow.transformer.MakePlotContaine

Parameters

- **plotName**
  period 2

Flow.CallableActors.SimplePlot

Reduce duplication?

- Two almost identical formulas
- Use variables (can also be attached to options)
  - SetVariable standalone
  - Variable source
  - VariablesArray source
  - CombineVariables source
  - SetVariable transformer
- format of a variable: @{*<name>*}

Reducing duplication

- Old formula

  2*(X/2-floor(0.5 + X/2))

- New formula using *@{period}* as placeholder

  2*(X/@{period}-floor(0.5 + X/@{period}))

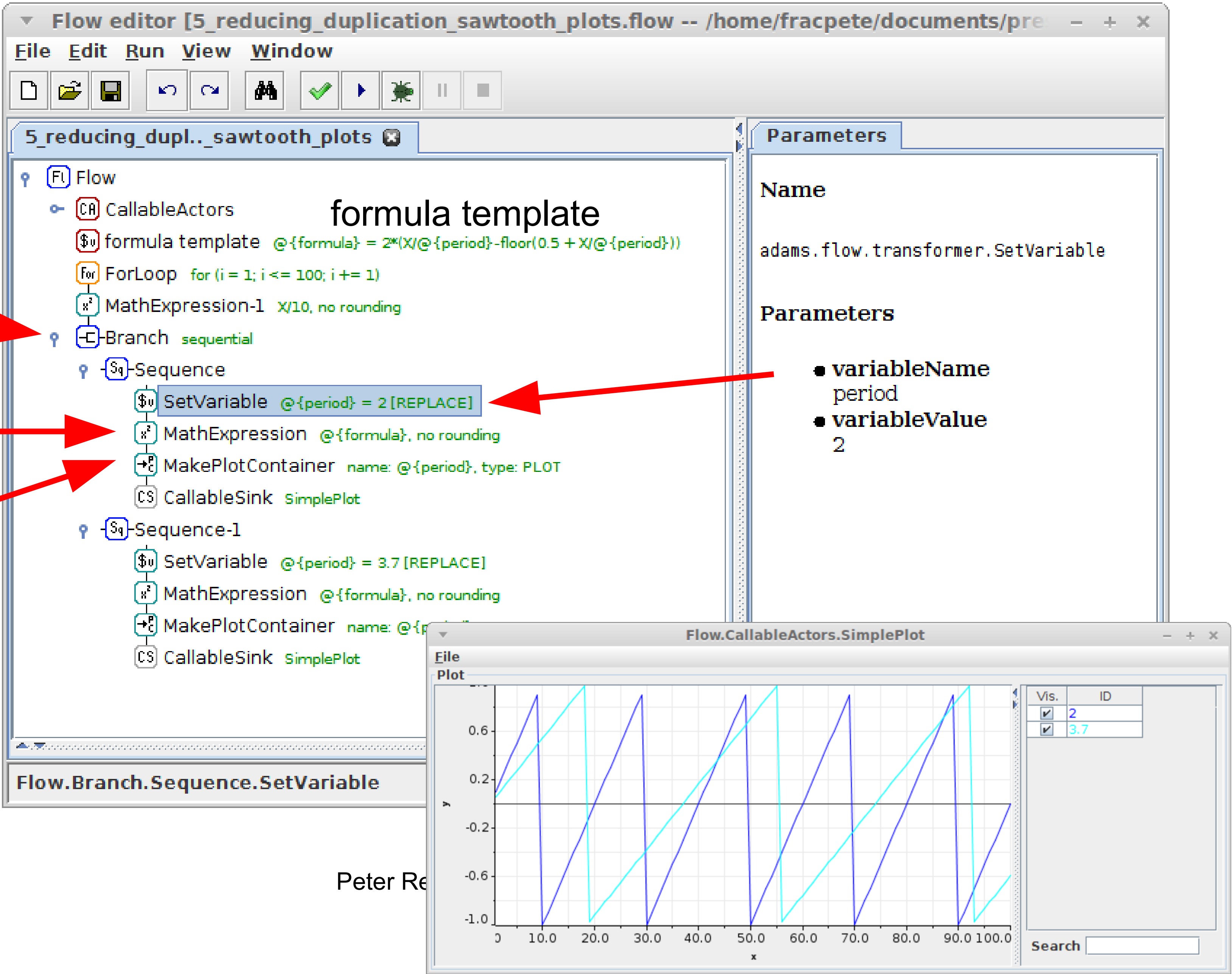
- But that still doesn't solve the duplicate formula
- Turn the formula into another variable and attach it to the MathExpression's expression option

Reducing duplication

sequential execution

variable attached to “expression”

variable attached to “plotName”



formula template

Flow editor [5_reducing_duplication_sawtooth_plots.flow -- /home/fracpete/documents/pre

```
Flow
  CallableActors
    formula template  @{formula} = 2*(X/@{period}-floor(0.5 + X/@{period}))
    ForLoop  for (i = 1; i <= 100; i += 1)
    MathExpression-1  X/10, no rounding
  Branch  sequential
    Sequence
      SetVariable  @{period} = 2 [REPLACE]
      MathExpression  @{formula}, no rounding
      MakePlotContainer  name: @{period}, type: PLOT
      CallableSink  SimplePlot
    Sequence-1
      SetVariable  @{period} = 3.7 [REPLACE]
      MathExpression  @{formula}, no rounding
      MakePlotContainer  name: @{p
      CallableSink  SimplePlot
```

Flow.Branch.Sequence.SetVariable

Name

adams.flow.transformer.SetVariable

Parameters

- **variableName**
  period
- **variableValue**
  2

Flow.CallableActors.SimplePlot

Vis.	ID
☑	2
☑	3.7

Avoid duplicate branch?

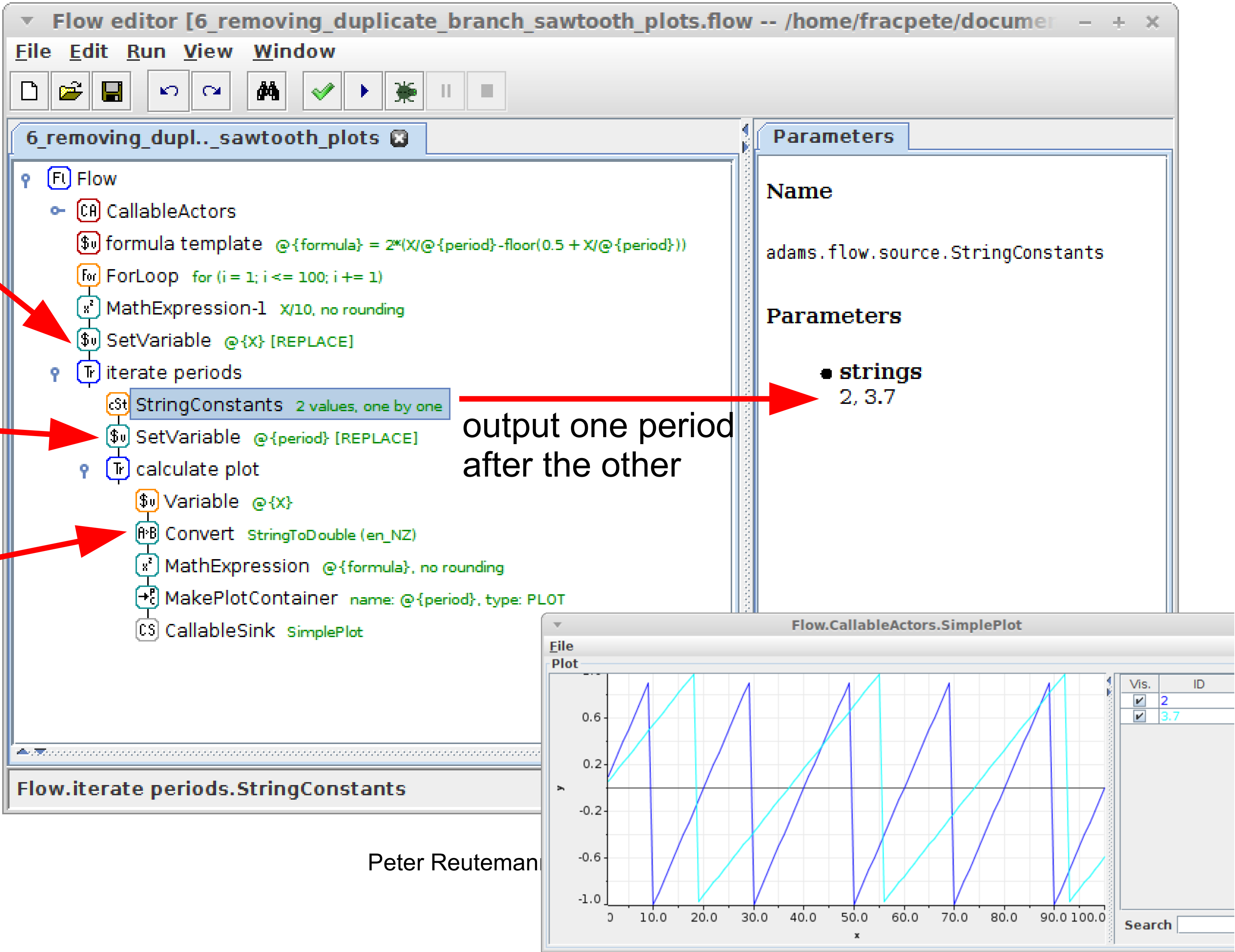
- Can we avoid the duplicate branch?
- Short answer: yes
- Long answer
  - make use of variables
  - use Tr Triggers
  - use cSt StringConstants as storage for the periods

No duplicate branch

store fraction in variable

store current period in var

convert string representation of X into float



Flow editor [6_removing_duplicate_branch_sawtooth_plots.flow -- /home/fracpete/documer
File Edit Run View Window
6_removing_dupl.._sawtooth_plots
Parameters

```
Flow
  CallableActors
    formula template  @{formula} = 2*(X/@{period}-floor(0.5 + X/@{period}))
  ForLoop  for (i = 1; i <= 100; i += 1)
  MathExpression-1  X/10, no rounding
  SetVariable  @{X} [REPLACE]
  iterate periods
    StringConstants  2 values, one by one
    SetVariable  @{period} [REPLACE]
    calculate plot
      Variable  @{X}
      Convert  StringToDouble (en_NZ)
      MathExpression  @{formula}, no rounding
      MakePlotContainer  name: @{period}, type: PLOT
      CallableSink  SimplePlot
```

Name

adams.flow.source.StringConstants

Parameters

- **strings**
  2, 3.7

Flow.iterate periods.StringConstants

output one period after the other

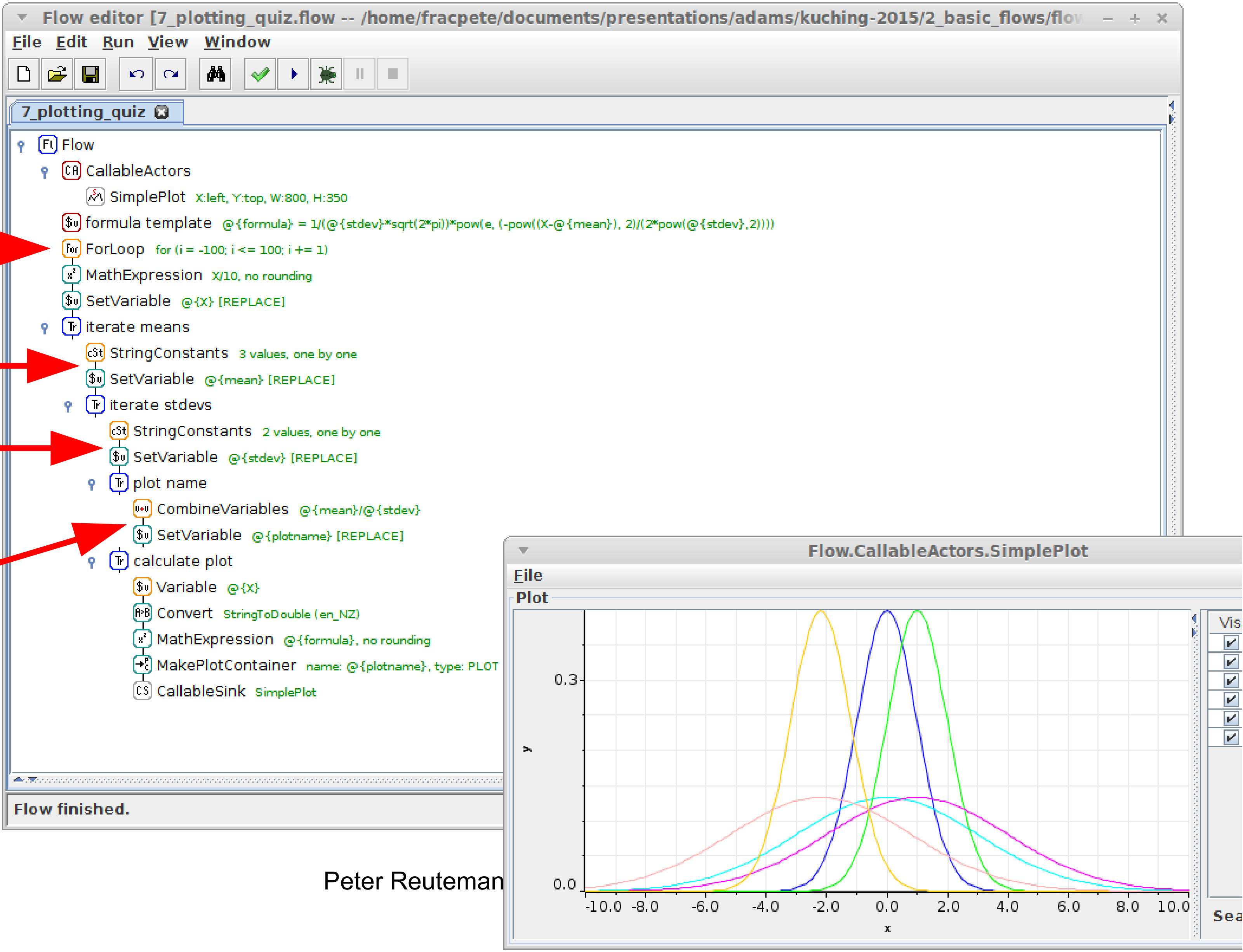
Quiz: Plotting

- implement the following function

$$f(x|\mu,\sigma)=\frac{1}{(\sigma\sqrt{2\pi})}e^{(\frac{-(x-\mu)^2}{(2\sigma^2)})}$$

- plot all combinations of
  - mean: 0, 1, -2.2
  - stdev: 1, 3
- output the functions in the same plot window
- **Hint:** you need the CombineVariables source

Quiz: Plotting


Flow editor [7_plotting_quiz.flow -- /home/fracpete/documents/presentations/adams/kuching-2015/2_basic_flows/flow
File Edit Run View Window
7_plotting_quiz
Flow.CallableActors.SimplePlot
File
Plot
0.3
0.0
y
-10.0 -8.0 -6.0 -4.0 -2.0 0.0 2.0 4.0 6.0 8.0 10.0
x
Vis
Sea

```
Flow
  CallableActors
    SimplePlot X:left, Y:top, W:800, H:350
  formula template @{formula} = 1/(@{stdev}*sqrt(2*pi))*pow(e, (-pow((X-@{mean}), 2)/(2*pow(@{stdev},2))))
  ForLoop for (i = -100; i <= 100; i += 1)
  MathExpression X/10, no rounding
  SetVariable @{X} [REPLACE]
  iterate means
    StringConstants 3 values, one by one
    SetVariable @{mean} [REPLACE]
    iterate stdevs
      StringConstants 2 values, one by one
      SetVariable @{stdev} [REPLACE]
      plot name
        CombineVariables @{mean}/@{stdev}
        SetVariable @{plotname} [REPLACE]
      calculate plot
        Variable @{X}
        Convert StringToDouble (en_NZ)
        MathExpression @{formula}, no rounding
        MakePlotContainer name: @{plotname}, type: PLOT
        CallableSink SimplePlot
```

Flow finished.

from neg
to pos

stdev as var

mean as var

plotname consists
of mean/stdev

Interactive flows

- Changing parameters in flows is tedious, only for experts
- How about asking the user directly?
- Examples
  - EnterValue
  - EnterManyValues
  - SelectFile
  - SelectDirectory

Sawtooth revisited

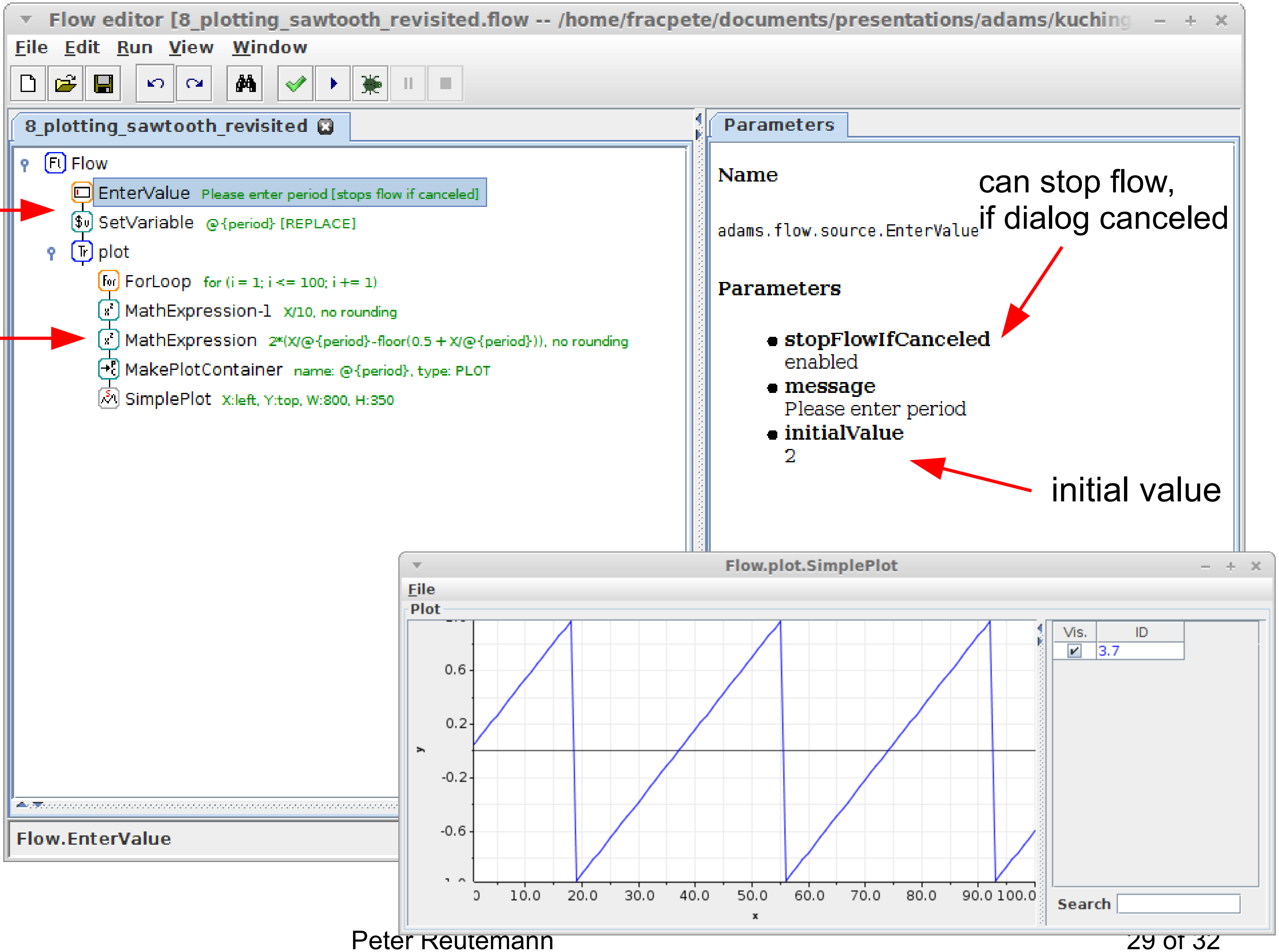
- Let user enter period
  - use ▭ EnterValue source
  - store output in variable
- Use period variable in formula and plotName

```
2*(X/@{period}-floor(0.5 + X/@{period}))
```

Sawtooth revisited

store user input
in variable

use variable
in formula



Flow editor [8_plotting_sawtooth_revisited.flow -- /home/fracpete/documents/presentations/adams/kuching
File Edit Run View Window
8_plotting_sawtooth_revisited

```
Flow
    EnterValue  Please enter period [stops flow if canceled]
    SetVariable  @{period} [REPLACE]
    plot
        ForLoop  for (i = 1; i <= 100; i += 1)
        MathExpression-1  X/10, no rounding
        MathExpression  2*(X/@{period}-floor(0.5 + X/@{period})), no rounding
        MakePlotContainer  name: @{period}, type: PLOT
        SimplePlot  X:left, Y:top, W:800, H:350
```

Parameters

Name

`adams.flow.source.EnterValue`

Parameters

- **stopFlowIfCanceled**
  enabled
- **message**
  Please enter period
- **initialValue**
  2

Flow.EnterValue

can stop flow,
if dialog canceled

initial value

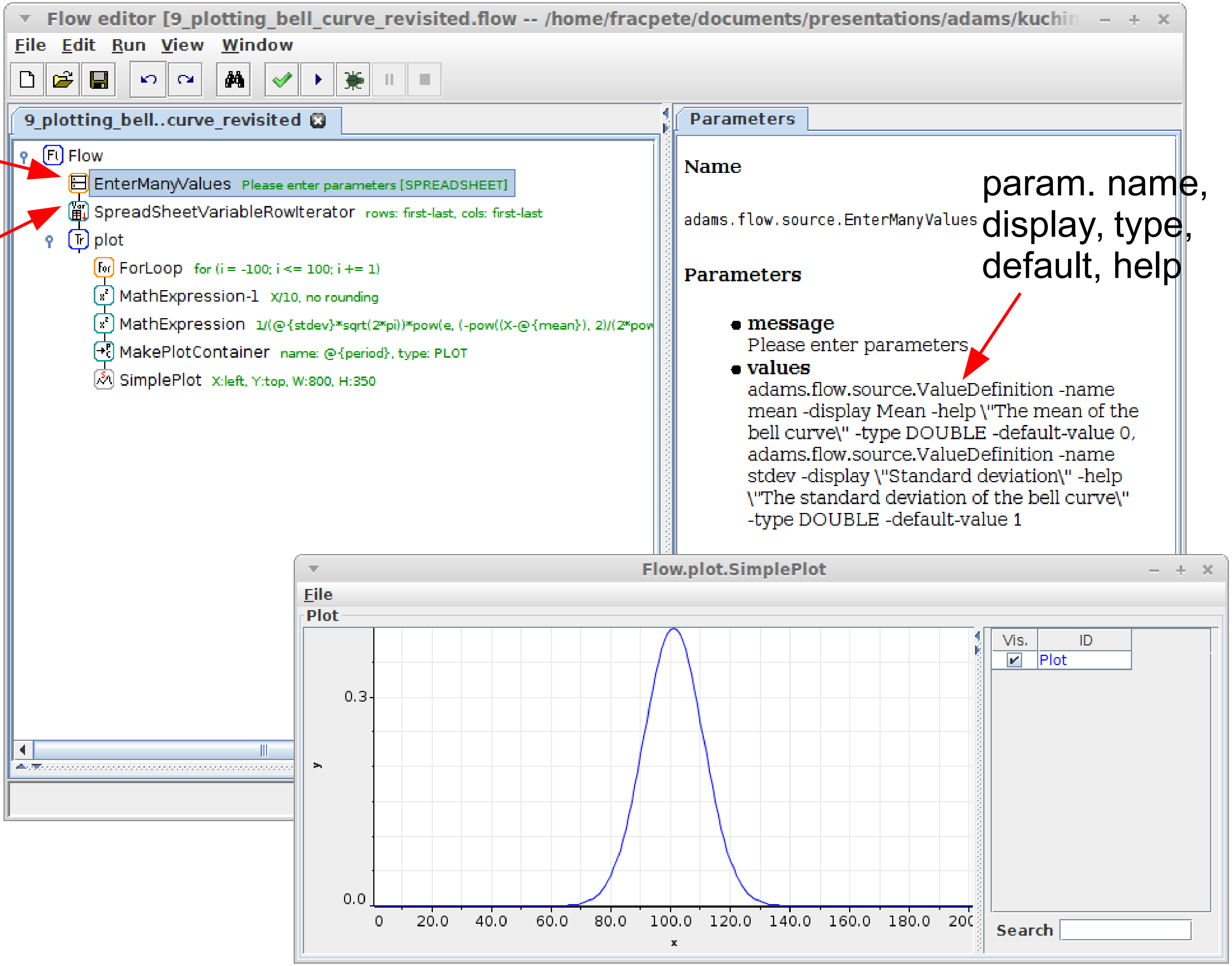
Bell curve revisited

- User enters mean and stdev
- Make use of EnterManyValues
- But how to obtain parameters?
  - outputs spreadsheet (column name = var name)
  - use SpreadSheetVariableRowIterator

Bell curve revisited

enter multiple parameters

turn spreadsheet into variables (col name = var name)



Flow editor [9_plotting_bell_curve_revisited.flow -- /home/fracpete/documents/presentations/adams/kuchin

```
Flow
  EnterManyValues  Please enter parameters [SPREADSHEET]
  SpreadSheetVariableRowIterator  rows: first-last, cols: first-last
  plot
    ForLoop  for (i = -100; i <= 100; i += 1)
    MathExpression-1  X/10, no rounding
    MathExpression  1/(@{stdev}*sqrt(2*pi))*pow(e, (-pow((X-@{mean}), 2)/(2*pow
    MakePlotContainer  name: @{period}, type: PLOT
    SimplePlot  X:left, Y:top, W:800, H:350
```

Name

adams.flow.source.EnterManyValues

Parameters

- **message**
  Please enter parameters
- **values**
  adams.flow.source.ValueDefinition -name mean -display Mean -help \"The mean of the bell curve\" -type DOUBLE -default-value 0, adams.flow.source.ValueDefinition -name stdev -display \"Standard deviation\" -help \"The standard deviation of the bell curve\" -type DOUBLE -default-value 1

Flow.plot.SimplePlot

param. name, display, type, default, help

Questions?

https://adams.cms.waikato.ac.nz/

@TheAdamsFlow