THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Big Data with ADAMS

Extending ADAMS

Peter Reutemann

How to extend?

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

- From easy to hard
  - Scripting
    - Groovy
    - Jython
  - Java Project
    - simply add ADAMS jars to CLASSPATH
  - Maven Project
    - create custom *pom.xml* and build your own project
    - check out ADAMS source code and compile yourself

API basics (2)

- dynamic class discovery
- Actor life cycle
  - `setUp(): String`

    configures the actor/flow, may return error message
  - `execute(): String`

    executes the actor/flow, may return error message
  - `wrapUp()`

    finishes up execution, leaves graphical output untouched
  - `cleanUp()`

    destructive, removes graphical output

NB: Skipping *adams.flow* package prefix in following slides

API basics

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

- Interfaces
  - `core.Actor` - all actors
  - `core.InputConsumer` - processes input
    - `input(Token)`
  - `core.OutputProducer` - generates output
    - `hasPendingOutput(): boolean`
    - `output(): Token`
- abstract classes
  - `core.AbstractActor`
  - `standalone.AbstractStandalone`
  - `source.Abstract`*`[Simple]`*`Source`
  - `transformer.AbstractTransformer`
  - `sink.AbstractSink`

API basics (3)

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

- graphical sinks
  - `sink.AbstractDisplay`
  - `sink.ComponentSupplier`
    - `sink.AbstractGraphicalDisplay`
  - `sink.TextSupplier`
    - `sink.AbstractTextualDisplay`
- interactive actors
  - Interfaces
    - `core.InteractiveActor`
    - `core.AutomatableInteractiveActor`
  - abstract classes
    - `source.AbstractInteractiveSource`
    - `transformer.AbstractInteractiveTransformer`

API basics (4)

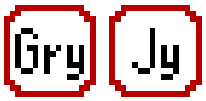
- Actor handlers manage data flow
  - interface: `control.ActorHandler`
  - some use “directors” to direct the data stream, e.g.:
    -  Sequence
    -  IfThenElse
    -  Switch

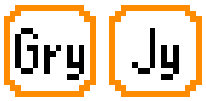
Scripting

- Available through these modules
  -  adams-groovy
  -  adams-jython
- Advantages
  - no compilation required
  - access to all libraries on CLASSPATH
  - fast prototyping
- Limitations
  - code executed in same JVM
  - only pure-Python code will execute (no numpy!)

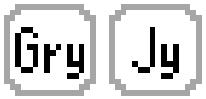
Scripting

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

- Actors available
  - 

Standalones
  - 

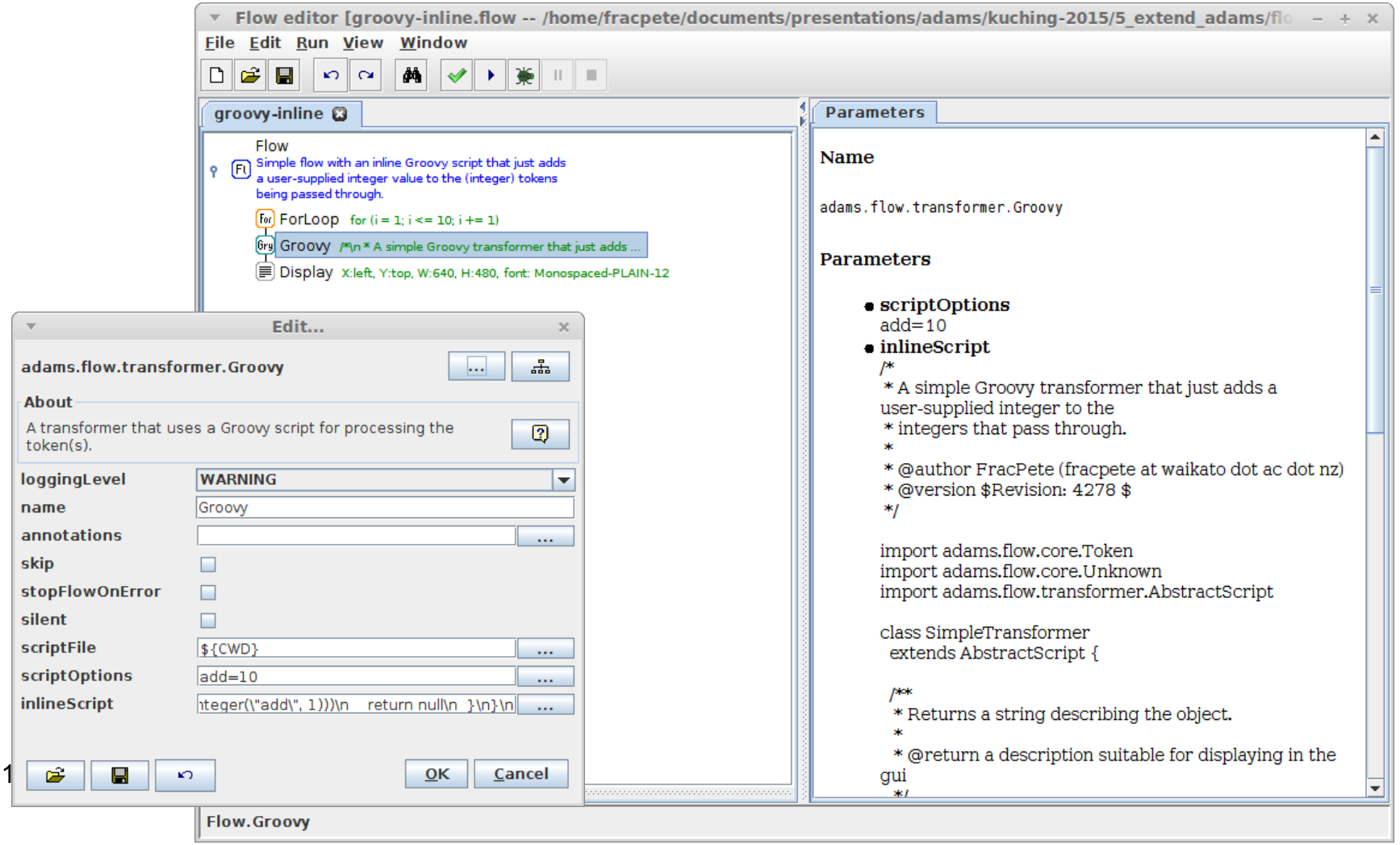
Sources
  - 

Transformers
  - 

Sink

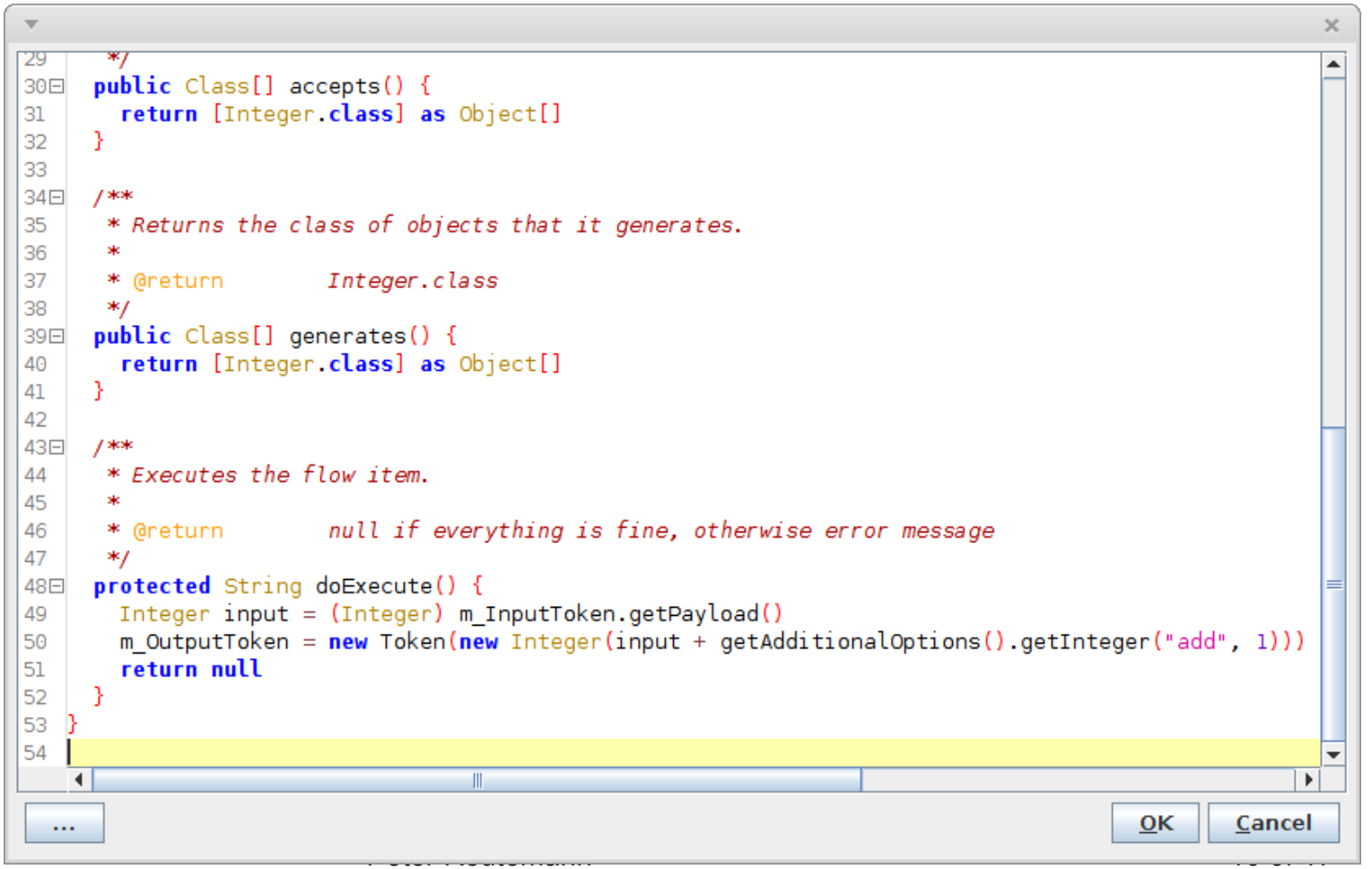
Scripting - Groovy

- Inline script



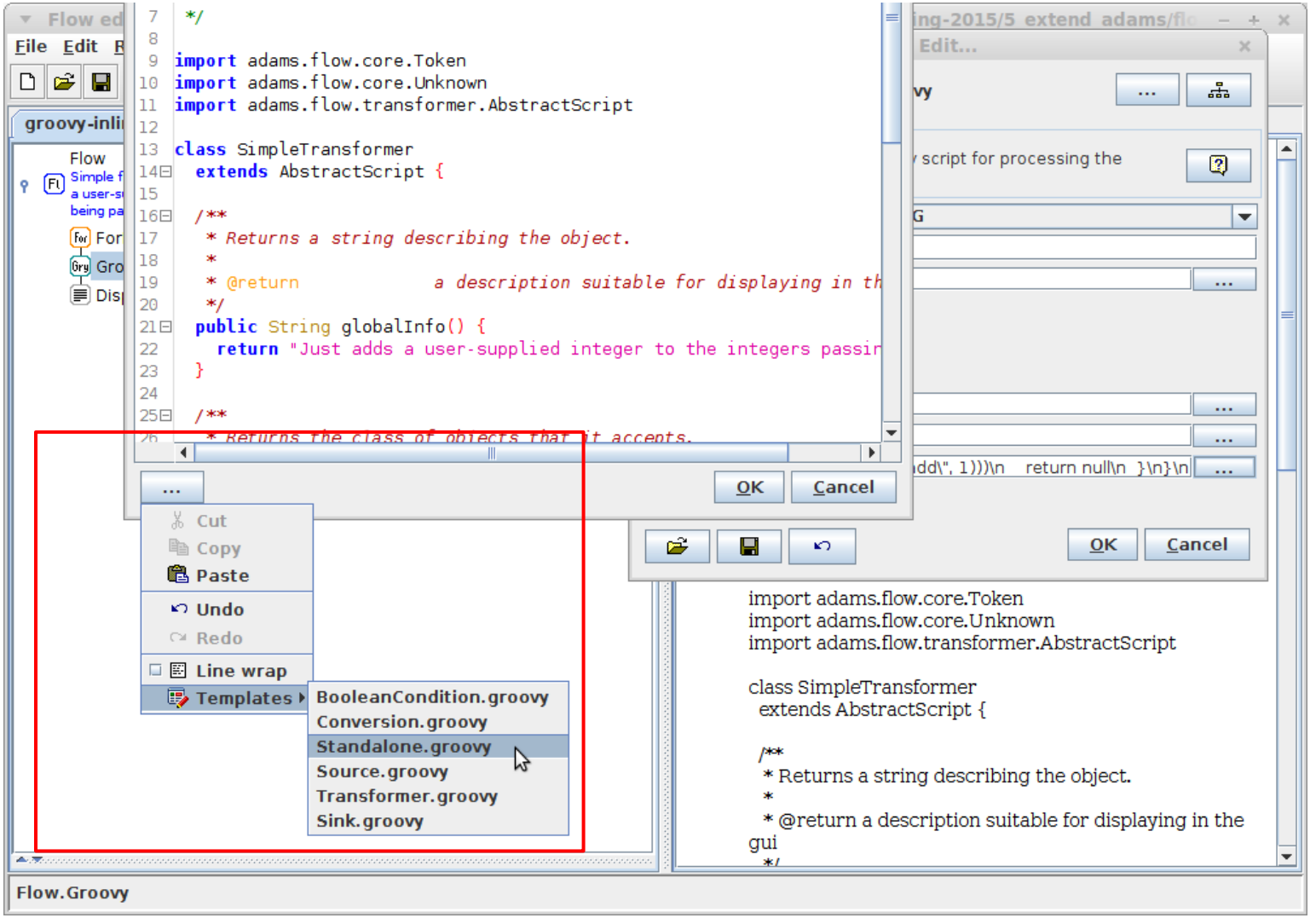
Scripting - Groovy

- Inline editor



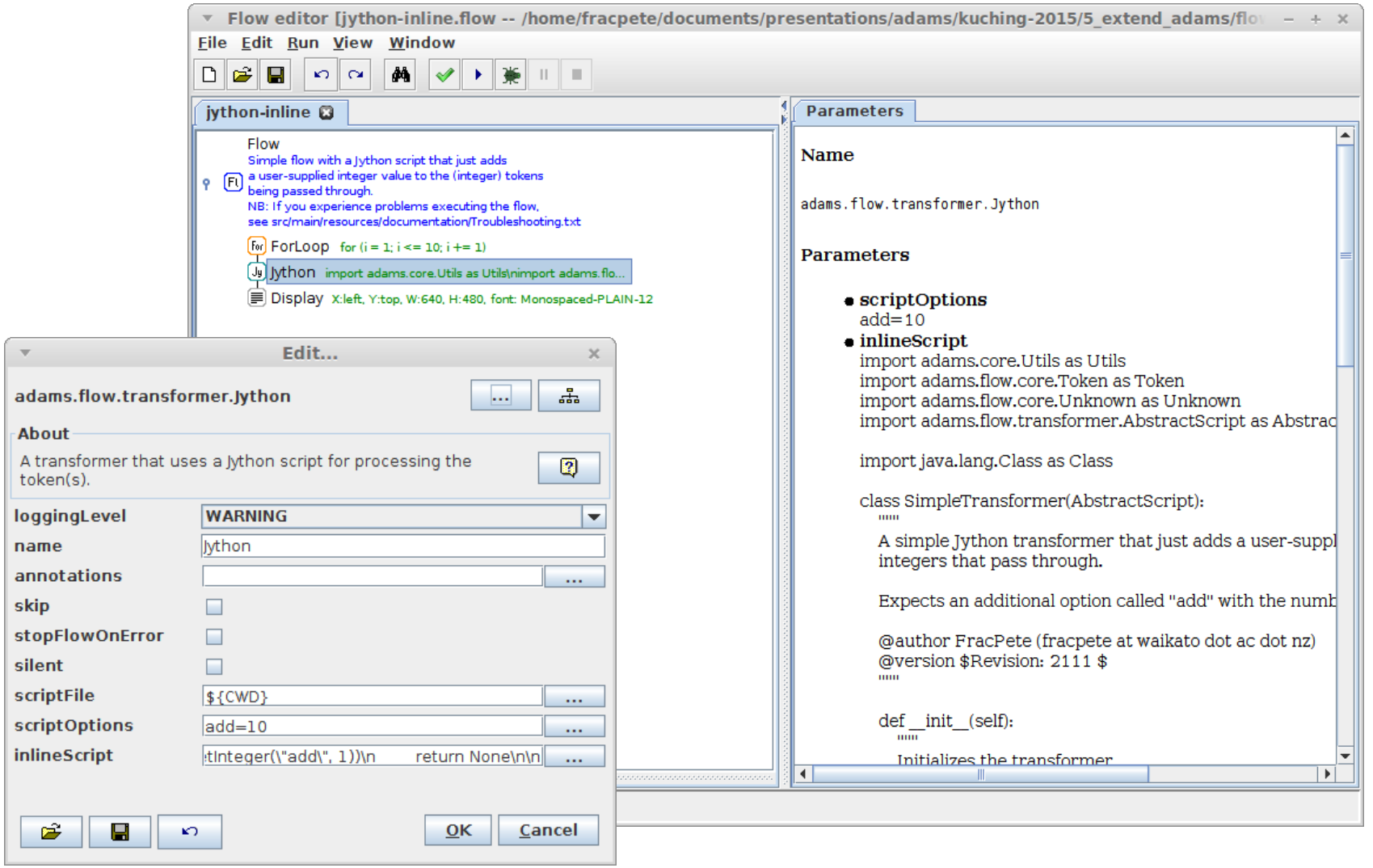
Scripting - Groovy

- Templates



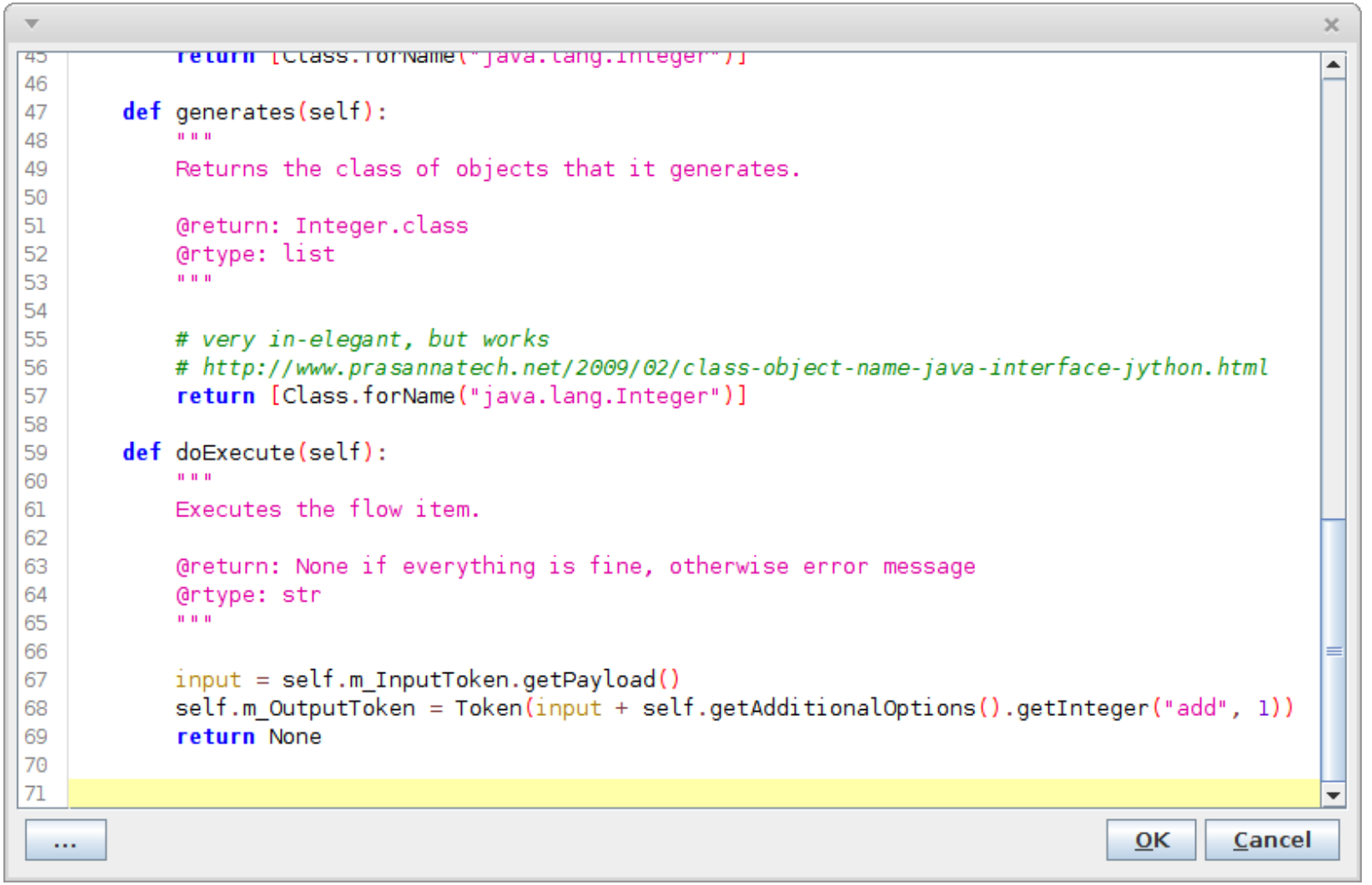
Scripting - Jython

- Inline script



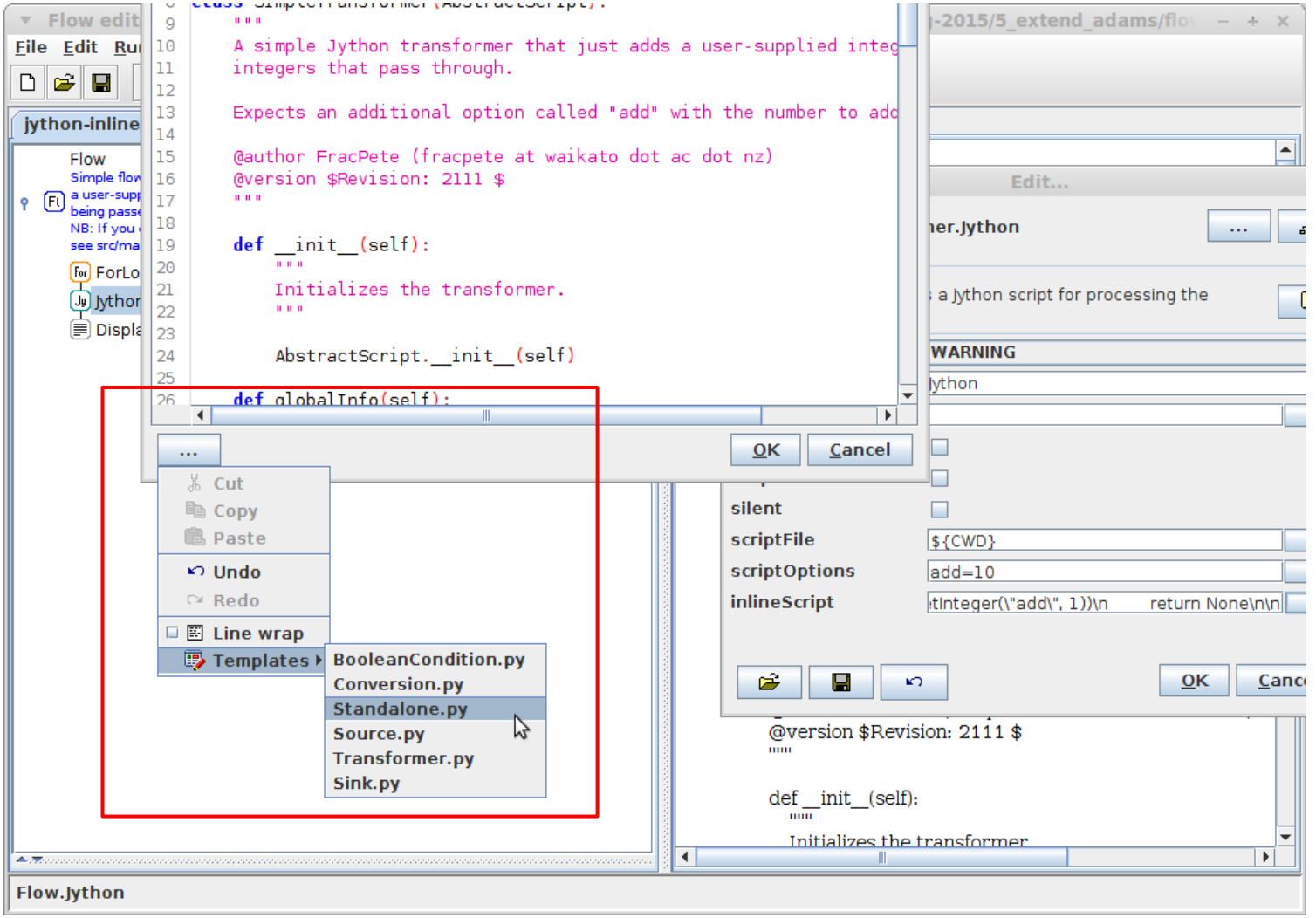
Scripting - Jython

- Inline editor



Scripting - Jython

- Templates



Java Project

- Download release or snapshot, decompress
- Start your favorite IDE
- Add jars from “lib” directory to the project's CLASSPATH
- Ensure that “java-cup-11b-2015.03.26.jar” is listed first
- Done!

Maven Project

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

- Requires Maven 3.0+

  http://maven.apache.org/

- Two options
  - use “roll your own” feature, use existing artifacts *

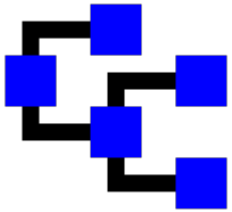
    https://adams.cms.waikato.ac.nz/rollyourown.html
  - check out from subversion, compile yourself

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

    NB: also requires LaTeX installed

* *Maven needs to use the ADAMS Nexus server as mirror (settings.xml)*

Questions?



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

@TheAdamsFlow