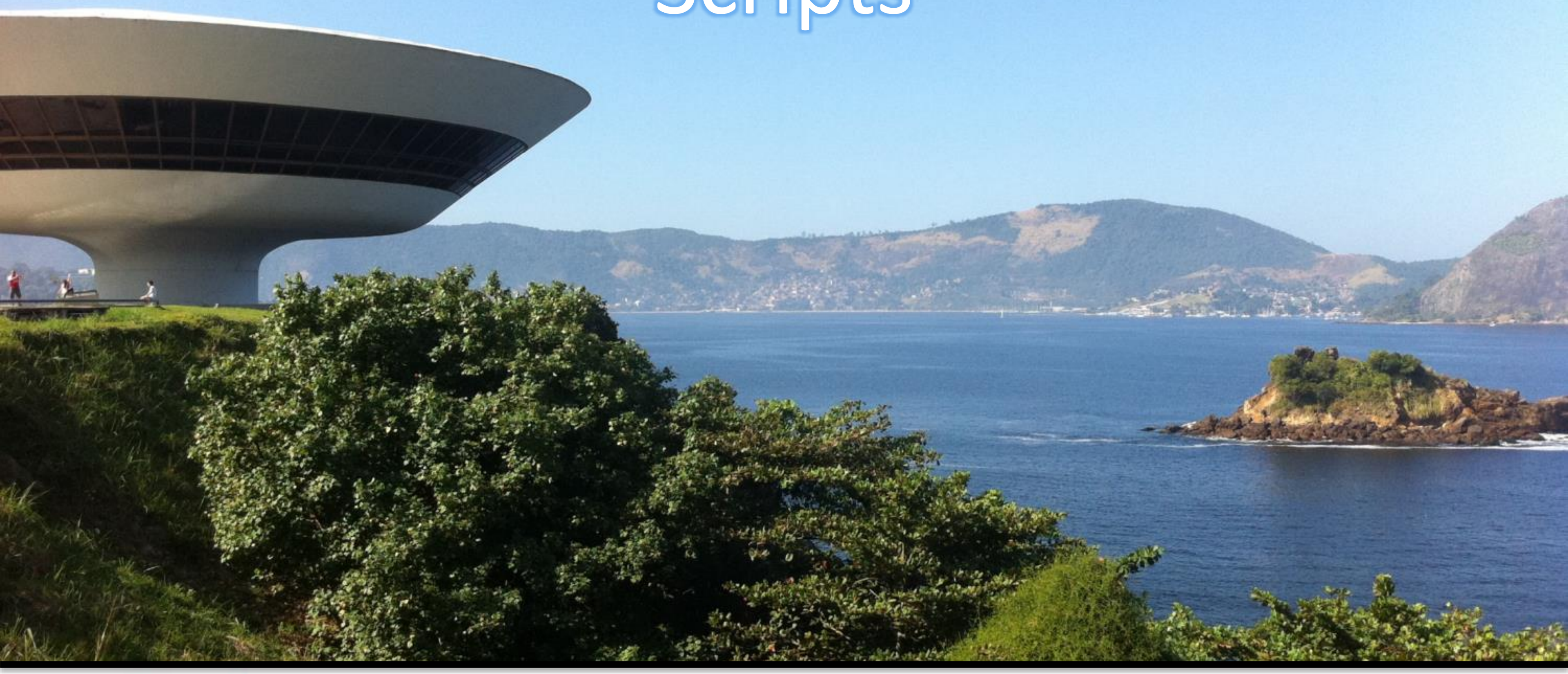
noWorkflow: Capturing, Analyzing, and Managing Provenance from Python Scripts



João Felipe Nicolaci Pimentel (UFF),
Juliana Freire (NYU), Vanessa Braganholo (UFF), Leonardo Murta (UFF)

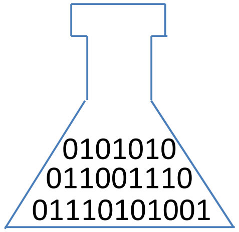
Instituto de computação

NYU POLYTECHNIC SCHOOL OF ENGINEERING

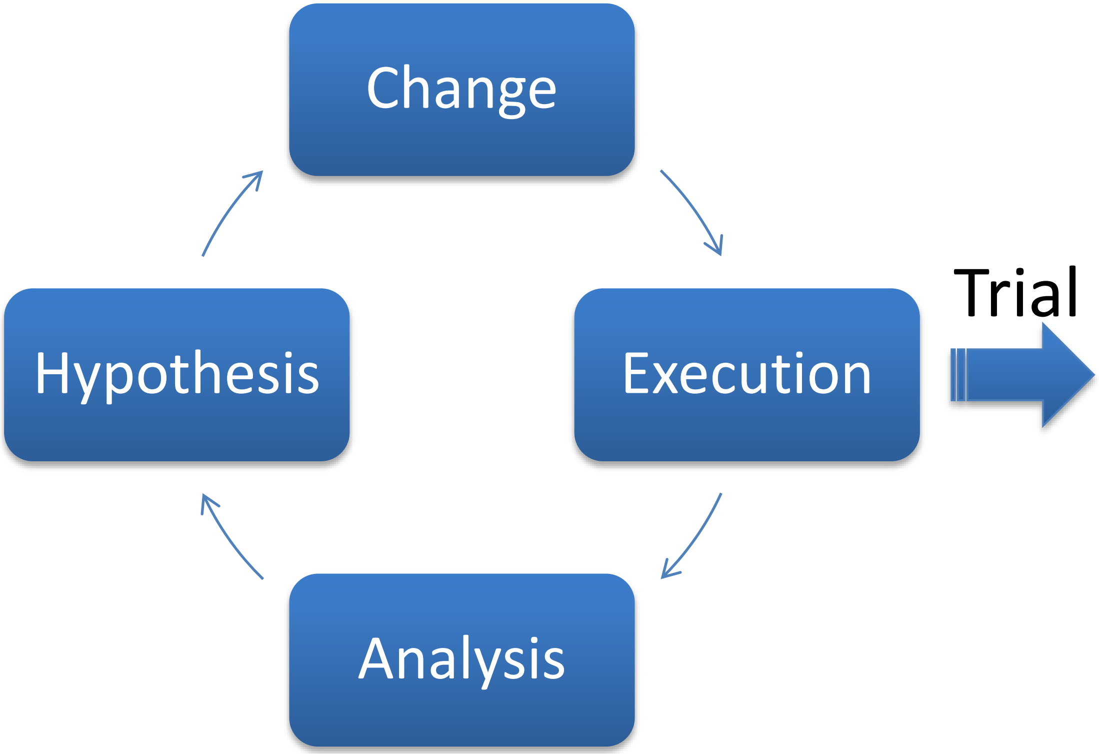
Outline

- Motivation
  - Scientific Experiments
  - Provenance
- noWorkflow
  - Collection
  - Management
  - Analysis
- IPython Notebook

Scientific Experiments



Exploratory Development



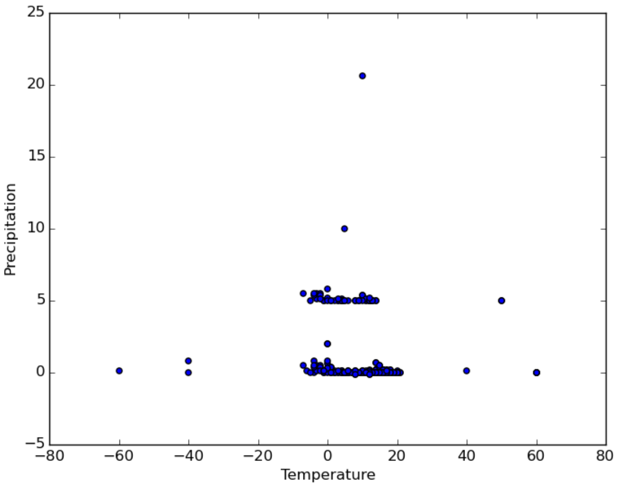
Change

Execution

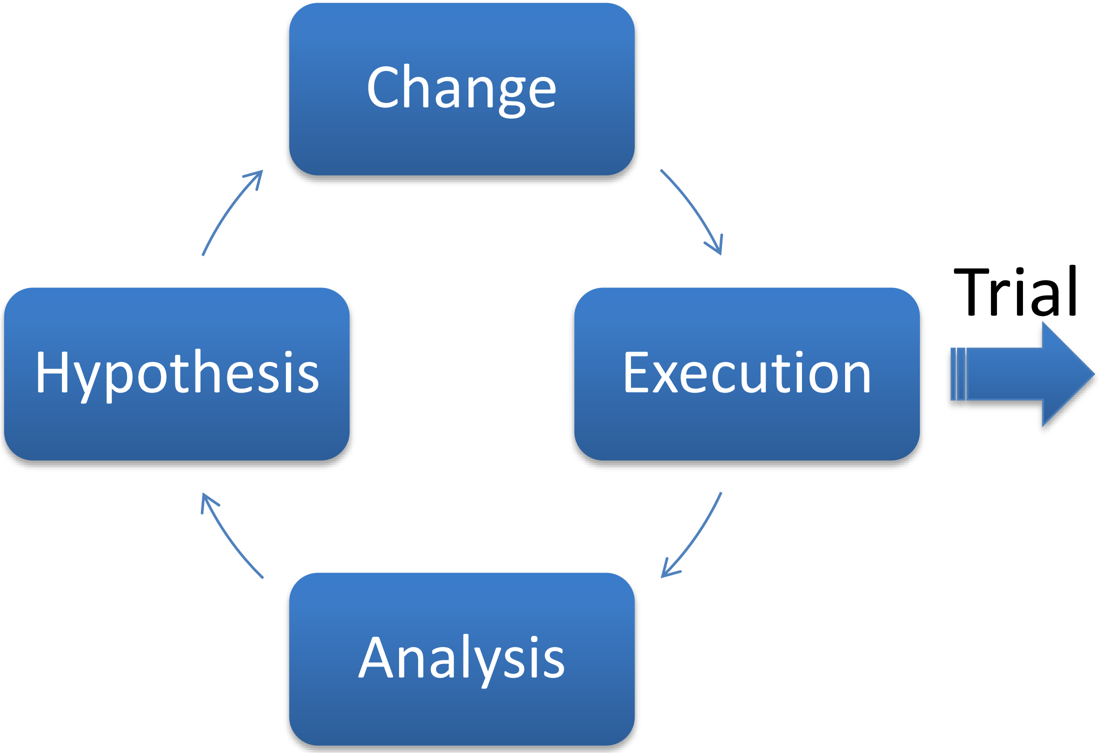
Analysis

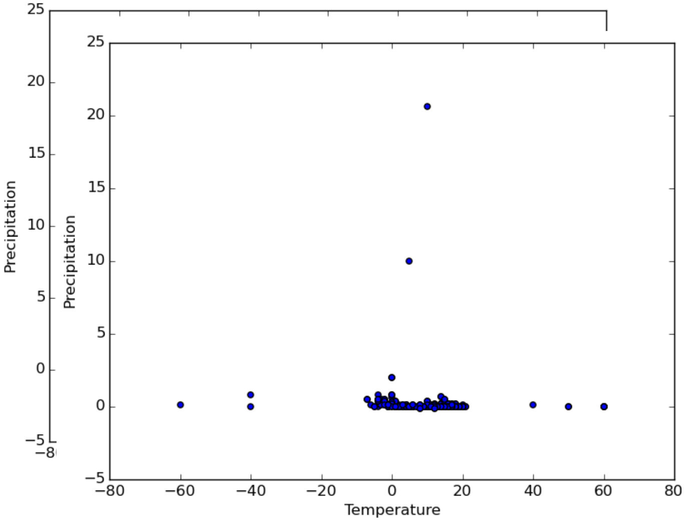
Hypothesis

Trial

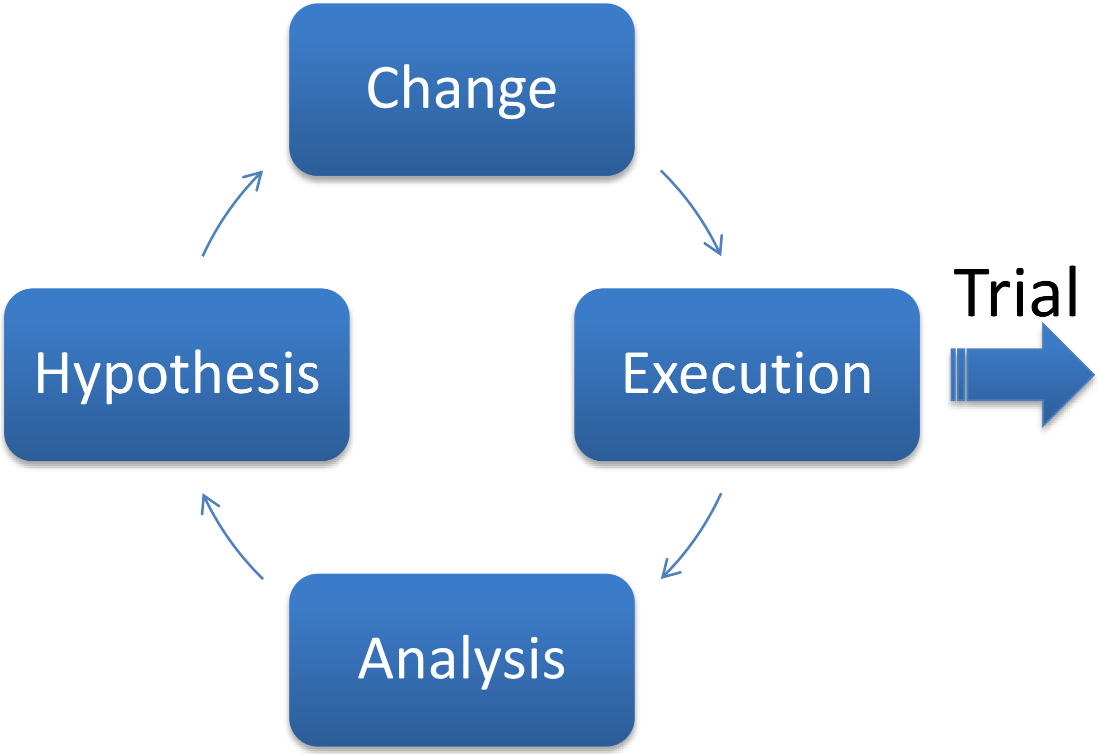


Exploratory Development





Exploratory Development



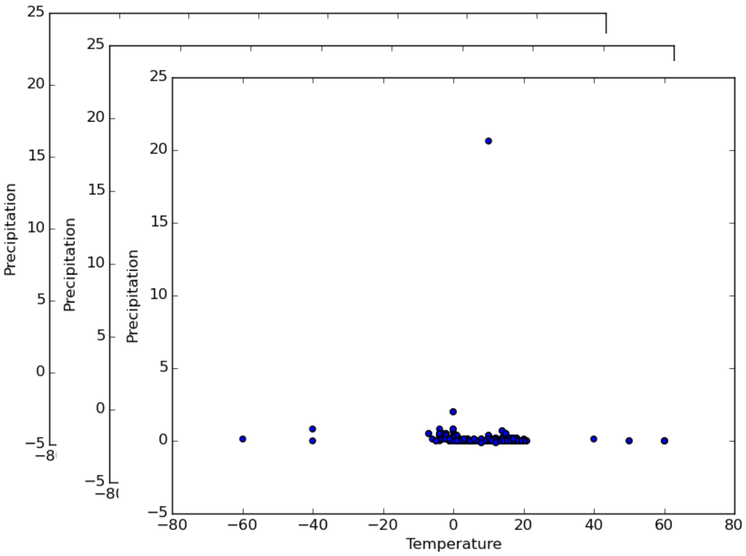
Change

Execution

Analysis

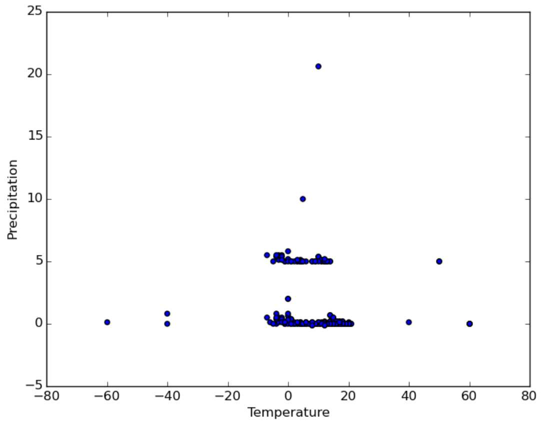
Hypothesis

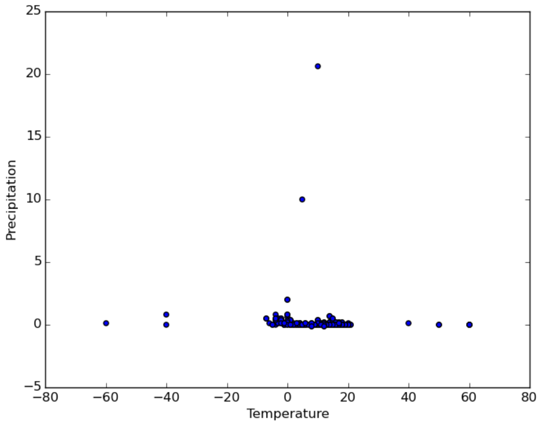
Trial

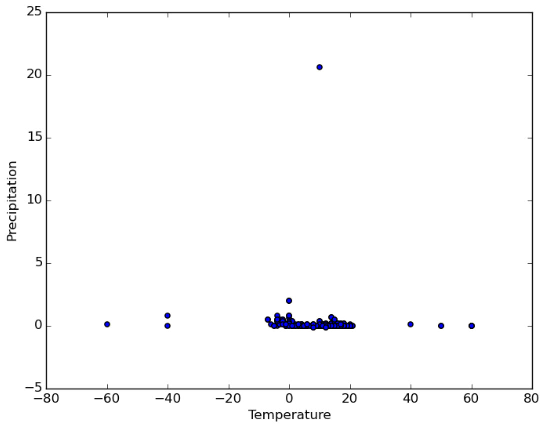


Questions

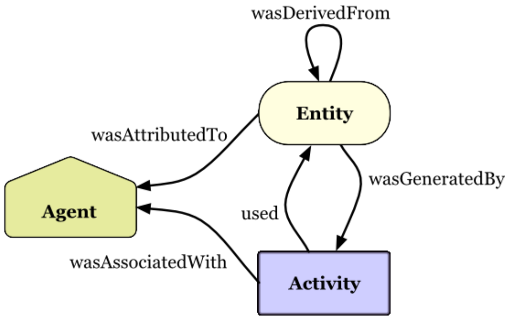
- How long did it take to execute each trial?
- How was the source code for each trial?
- Which data were used?
- Which transformations were performed?
- Can I reproduce it?







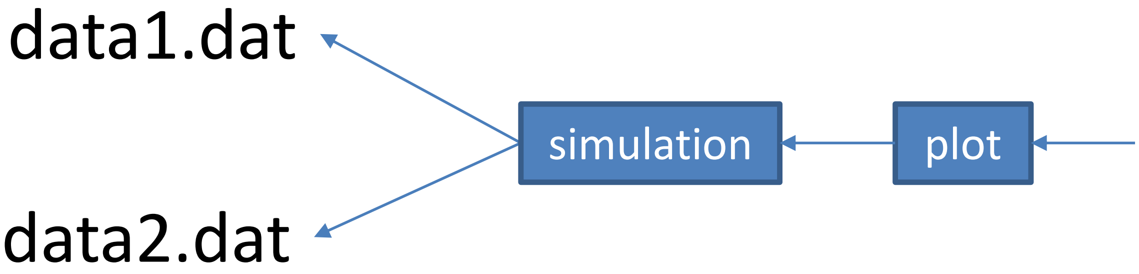
Provenance



http://www.w3.org/TR/prov-primer/

Provenance

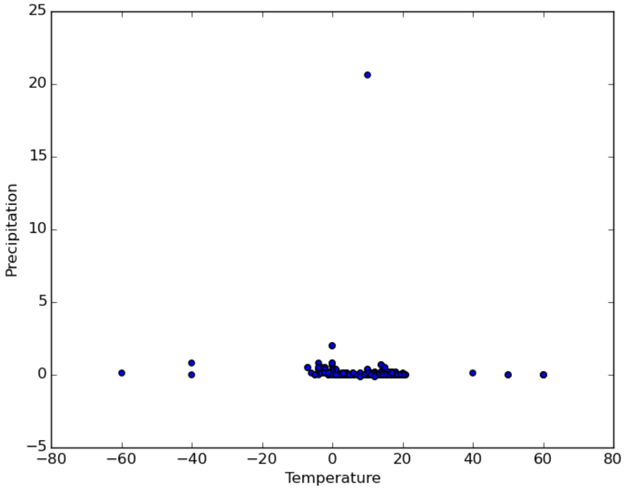
“Refers to the documented history of an art object, or the documentation of processes in a digital object’s life cycle” [Moreau et al., 2008]


data1.dat

data2.dat

simulation

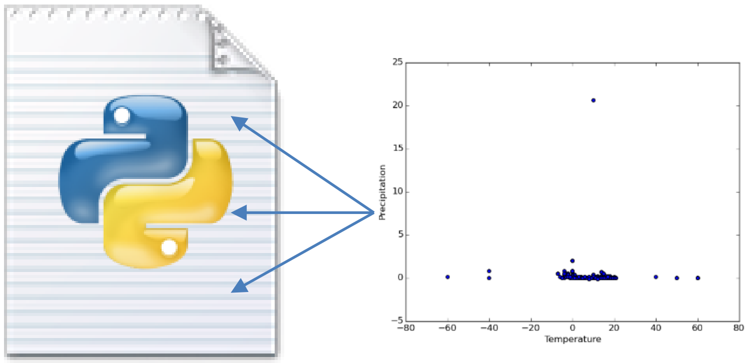
plot



How do I capture it?

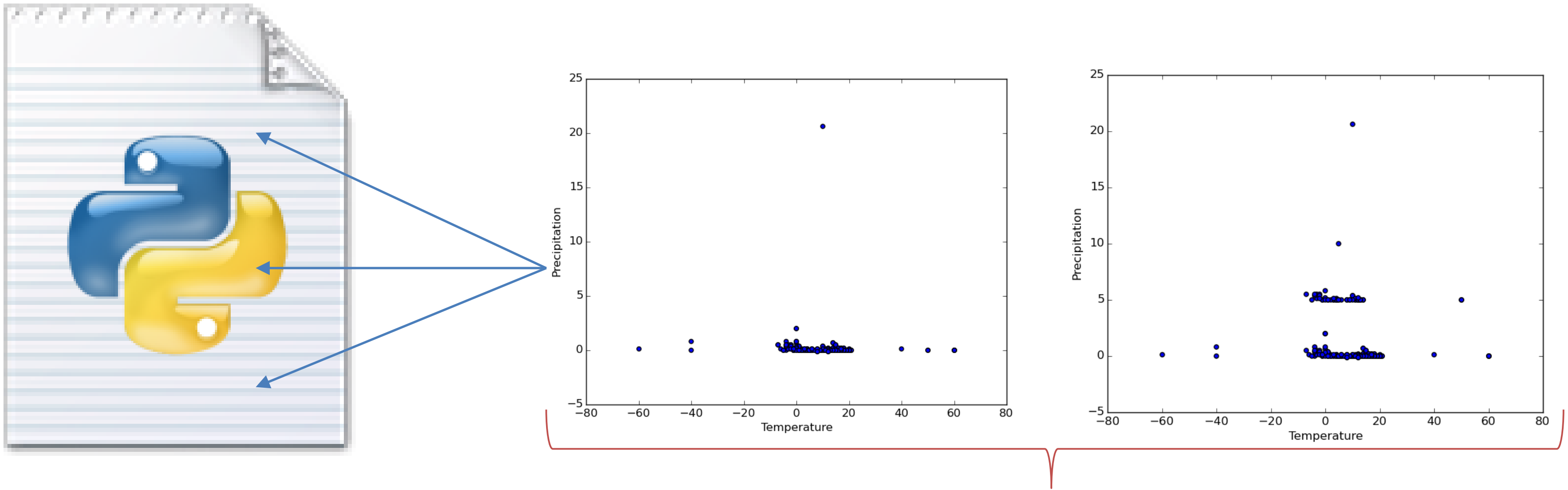
- **Workflow Systems**
  - Transparent
  - Large start-up costs
  - Hard to integrate tools
- **OS-based solutions**
  - Transparent
  - Hard to connect to the semantics of the experiment
- **Script-based solutions**
  - Users must annotate their script with provenance capture directives

noWorkflow


25
20
15
10
5
0
−5
Precipitation
−80
−60
−40
−20
0
20
40
60
80
Temperature

noWorkflow

- **Transparently** captures provenance of Python scripts
  - No changes required!
- Allows users to analyze provenance information



Manage, Assess, and **Reproduce**

Python
vs
noWorkflow

simulation.py

```python
import csv
import sys
import matplotlib.pyplot as plt
from simulator import simulate

def run_simulation(data_a, data_b):
    ...

def csv_read(f):
    ...

def extract_column(data, column):
    ...

def plot(data):
    ...
```

simulation.py

main

```
data_a = sys.argv[1]
data_b = sys.argv[2]
data = run_simulation(data_a, data_b)
plot(data)
```

simulation.py

main

```
data_a = sys.argv[1]
data_b = sys.argv[2]
data = run_simulation(data_a, data_b)
plot(data)
```

simulation.py

```
def run_simulation(data_a, data_b):
    a = csv_read(data_a)
    b = csv_read(data_b)
    data = simulate(a, b)
    return data
```

```
def csv_read(f):
    reader = csv.reader(open(f, 'rU'), delimiter=':')
    data = []
    for row in reader:
        data.append(row)
    return data
```

simulation.py

main

```
data_a = sys.argv[1]
data_b = sys.argv[2]
data = run_simulation(data_a, data_b)
plot(data)
```

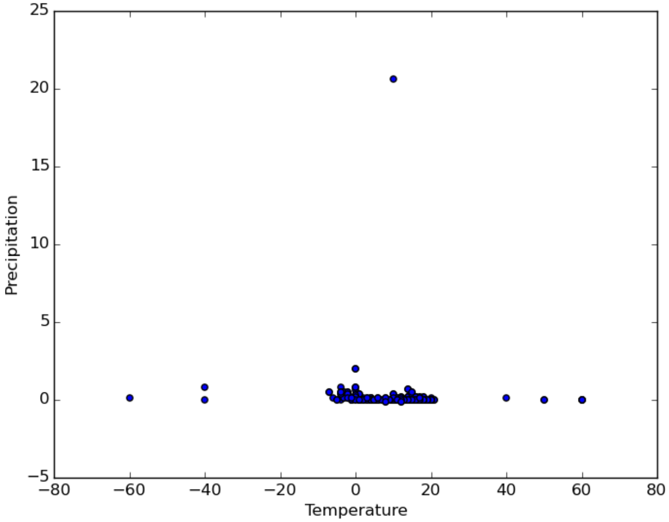
simulation.py

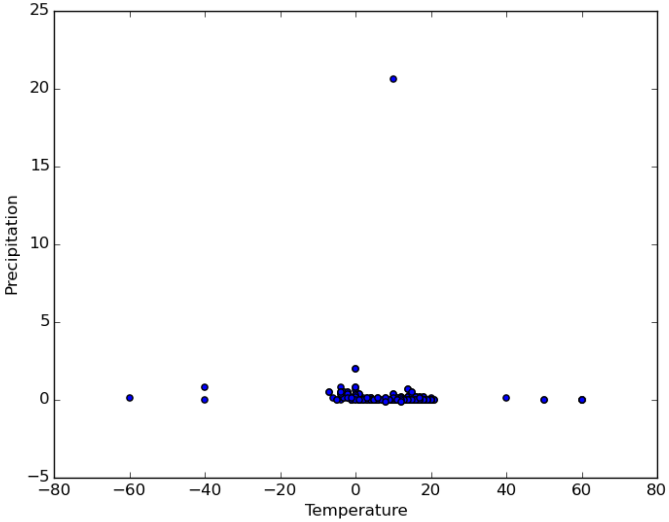
```python
def plot(data):
    # Get Temperature
    t = extract_column(data, 0)
    # Get Precipitation
    p = extract_column(data, 1)
    plt.scatter(t, p, marker='o')
    plt.xlabel('Temperature')
    plt.ylabel('Precipitation')
    plt.savefig('output.png')
```

Comparison

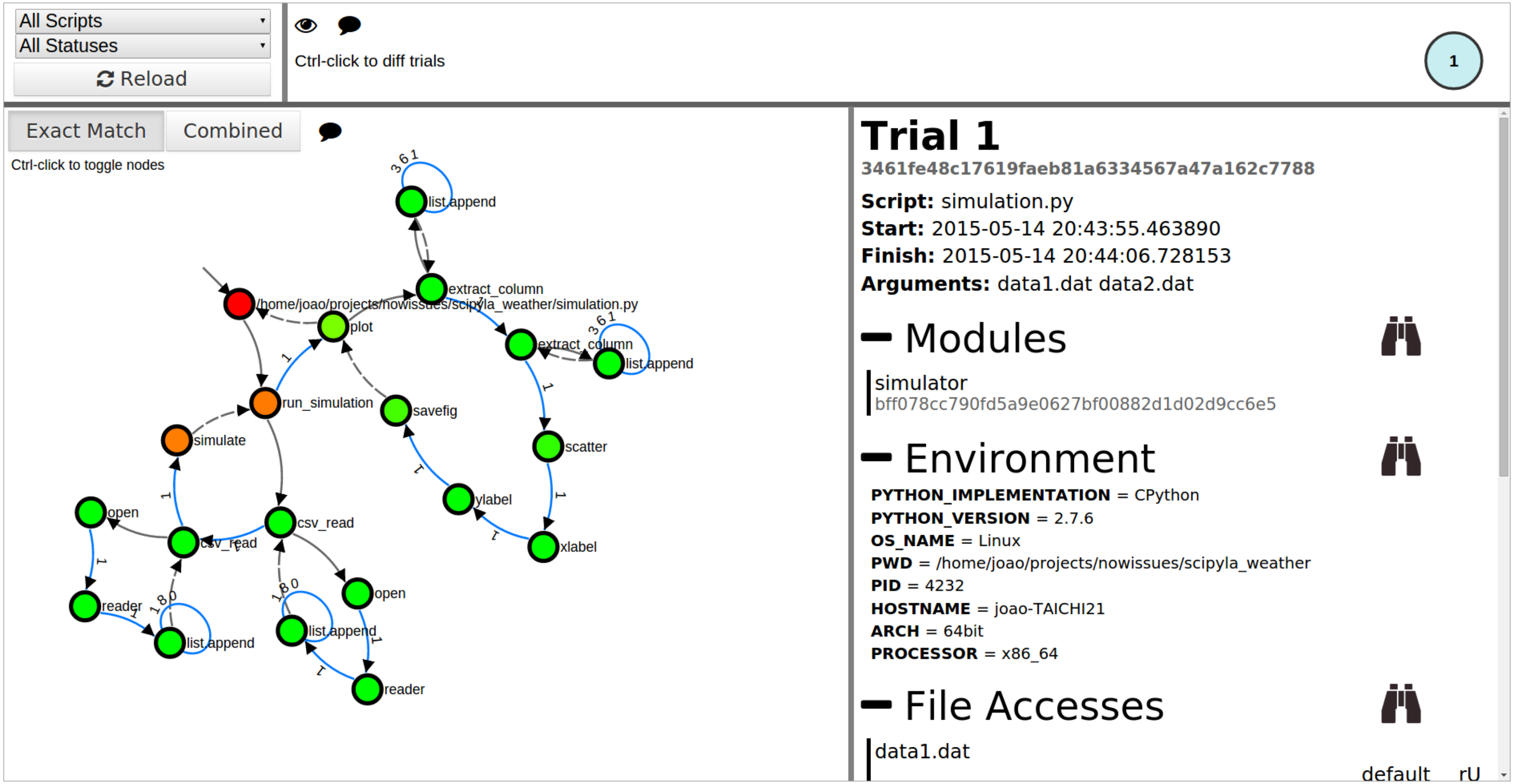
```
$ python simulation.py \
> data1.dat data2.dat
$ display output.png
```

```
$ now run simulation.py \
> data1.dat data2.dat
$ display output.png
```





$ now vis



Trial 1

3461fe48c17619faeb81a6334567a47a162c7788

Script: simulation.py
Start: 2015-05-14 20:43:55.463890
Finish: 2015-05-14 20:44:06.728153
Arguments: data1.dat data2.dat

— Modules

simulator
bff078cc790fd5a9e0627bf00882d1d02d9cc6e5

— Environment

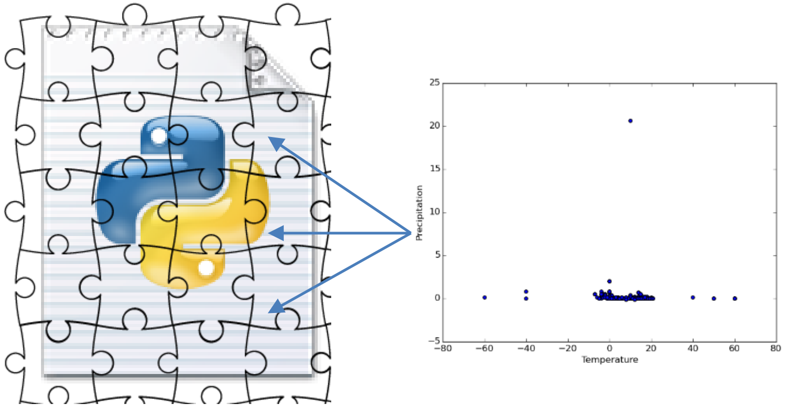
PYTHON_IMPLEMENTATION = CPython
PYTHON_VERSION = 2.7.6
OS_NAME = Linux
PWD = /home/joao/projects/nowissues/scipyla_weather
PID = 4232
HOSTNAME = joao-TAICHI21
ARCH = 64bit
PROCESSOR = x86_64

— File Accesses

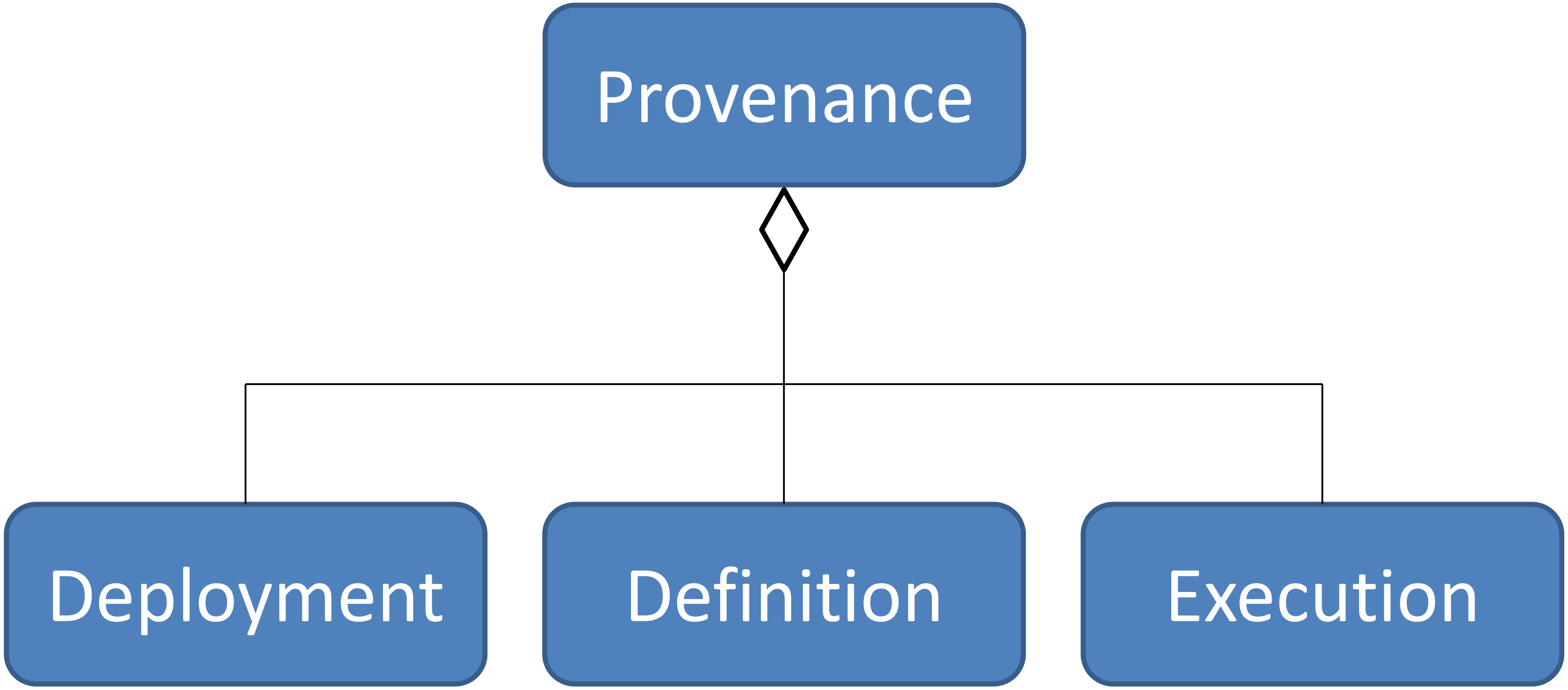
data1.dat

default rU

Collection


25
20
15
10
5
0
−5
Precipitation
−80
−60
−40
−20
0
20
40
60
80
Temperature

$ now run simulation.py data1.dat data2.dat



Provenance

Deployment

Definition

Execution

Deployment

- Environment
- Module dependencies

```
import csv
import sys
import matplotlib.pyplot as plt
from simulator import simulate
```

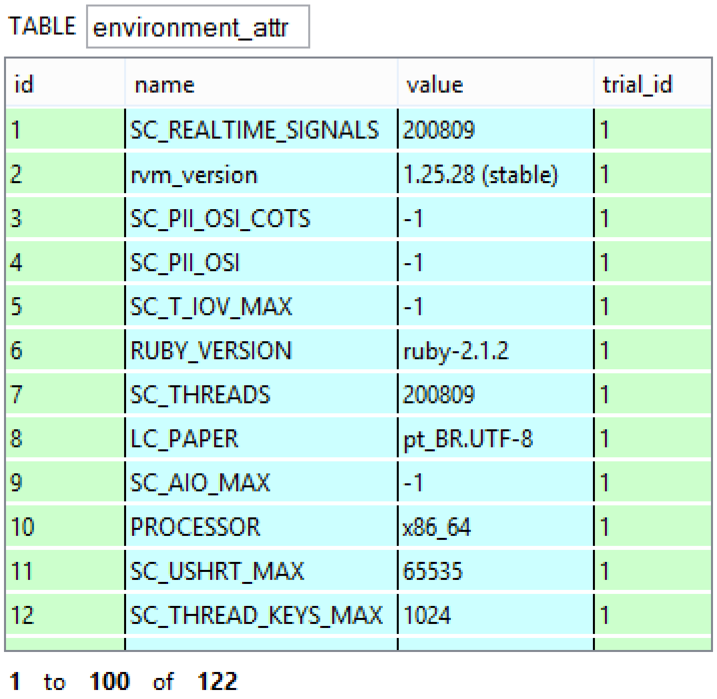


TABLE environment_attr

id	name	value	trial_id
1	SC_REALTIME_SIGNALS	200809	1
2	rvm_version	1.25.28 (stable)	1
3	SC_PII_OSI_COTS	-1	1
4	SC_PII_OSI	-1	1
5	SC_T_IOV_MAX	-1	1
6	RUBY_VERSION	ruby-2.1.2	1
7	SC_THREADS	200809	1
8	LC_PAPER	pt_BR.UTF-8	1
9	SC_AIO_MAX	-1	1
10	PROCESSOR	x86_64	1
11	SC_USHRT_MAX	65535	1
12	SC_THREAD_KEYS_MAX	1024	1

1 to 100 of 122

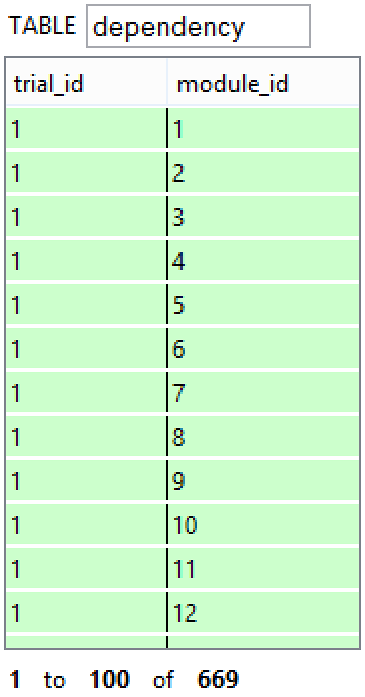


TABLE dependency

trial_id	module_id
1	1
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
1	10
1	11
1	12

1 to 100 of 669

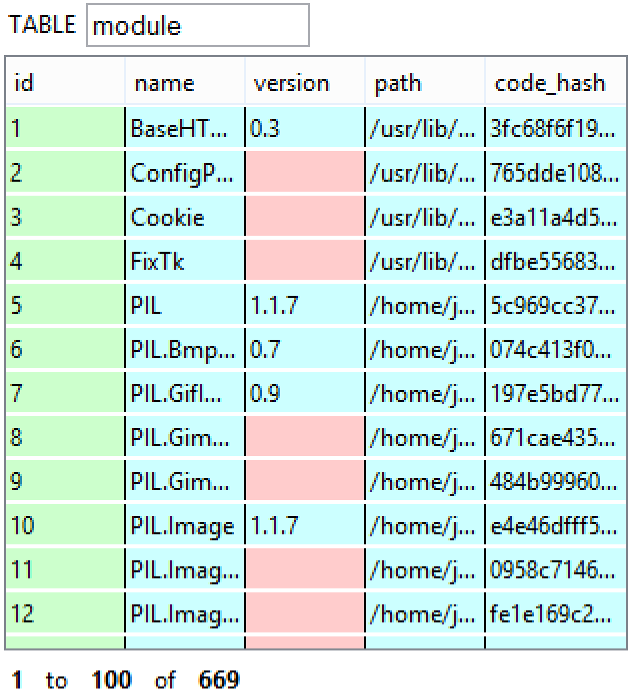


TABLE module

id	name	version	path	code_hash
1	BaseHT...	0.3	/usr/lib/...	3fc68f6f19...
2	ConfigP...		/usr/lib/...	765dde108...
3	Cookie		/usr/lib/...	e3a11a4d5...
4	FixTk		/usr/lib/...	dfbe55683...
5	PIL	1.1.7	/home/j...	5c969cc37...
6	PIL.Bmp...	0.7	/home/j...	074c413f0...
7	PIL.Gifl...	0.9	/home/j...	197e5bd77...
8	PIL.Gim...		/home/j...	671cae435...
9	PIL.Gim...		/home/j...	484b99960...
10	PIL.Image	1.1.7	/home/j...	e4e46dfff5...
11	PIL.Imag...		/home/j...	0958c7146...
12	PIL.Imag...		/home/j...	fe1e169c2...

1 to 100 of 669

Definition

- Script
- Function definitions
- Arguments and Globals

```
def run_simulation(data_a, data_b):
def csv_read(f):
def extract_column(data, column):
def plot(data):
```

Hash: 3461fe48c17619faeb81a6334567a47a162c7788

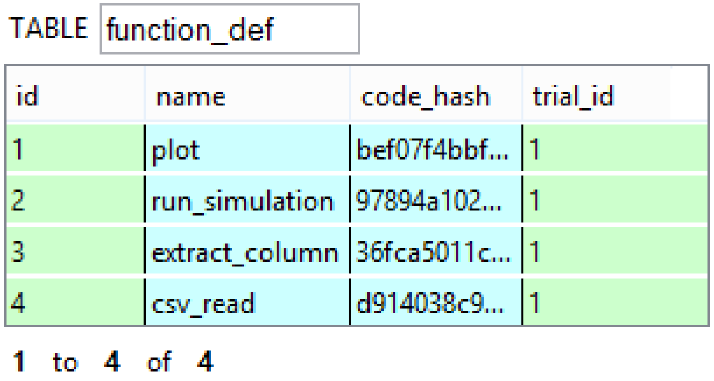


TABLE function_def

id	name	code_hash	trial_id
1	plot	bef07f4bbf...	1
2	run_simulation	97894a102...	1
3	extract_column	36fca5011c...	1
4	csv_read	d914038c9...	1

1 to 4 of 4

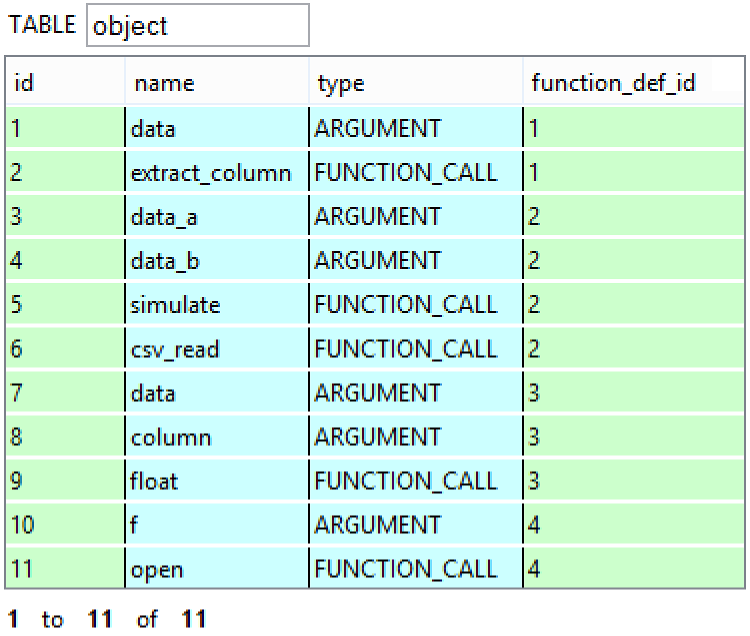


TABLE object

id	name	type	function_def_id
1	data	ARGUMENT	1
2	extract_column	FUNCTION_CALL	1
3	data_a	ARGUMENT	2
4	data_b	ARGUMENT	2
5	simulate	FUNCTION_CALL	2
6	csv_read	FUNCTION_CALL	2
7	data	ARGUMENT	3
8	column	ARGUMENT	3
9	float	FUNCTION_CALL	3
10	f	ARGUMENT	4
11	open	FUNCTION_CALL	4

1 to 11 of 11

Execution

- Files content
- Function calls
- Parameter and global values
- Program arguments

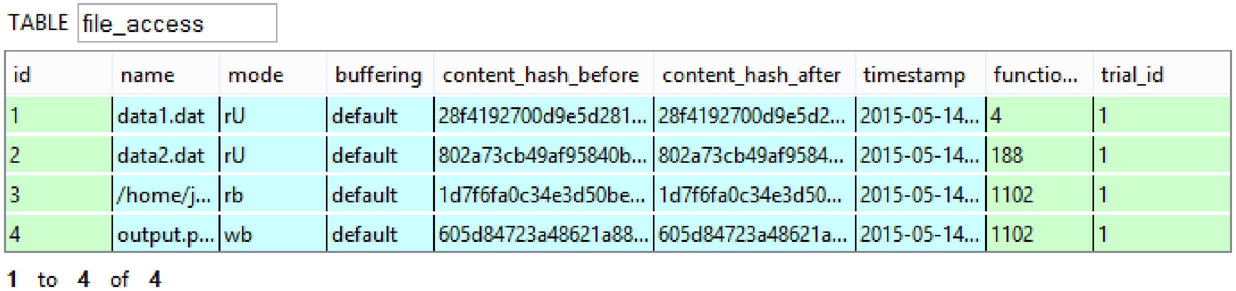


TABLE file_access

id	name	mode	buffering	content_hash_before	content_hash_after	timestamp	functio...	trial_id
1	data1.dat	rU	default	28f4192700d9e5d281...	28f4192700d9e5d2...	2015-05-14...	4	1
2	data2.dat	rU	default	802a73cb49af95840b...	802a73cb49af9584...	2015-05-14...	188	1
3	/home/j...	rb	default	1d7f6fa0c34e3d50be...	1d7f6fa0c34e3d50...	2015-05-14...	1102	1
4	output.p...	wb	default	605d84723a48621a88...	605d84723a48621a...	2015-05-14...	1102	1

1 to 4 of 4

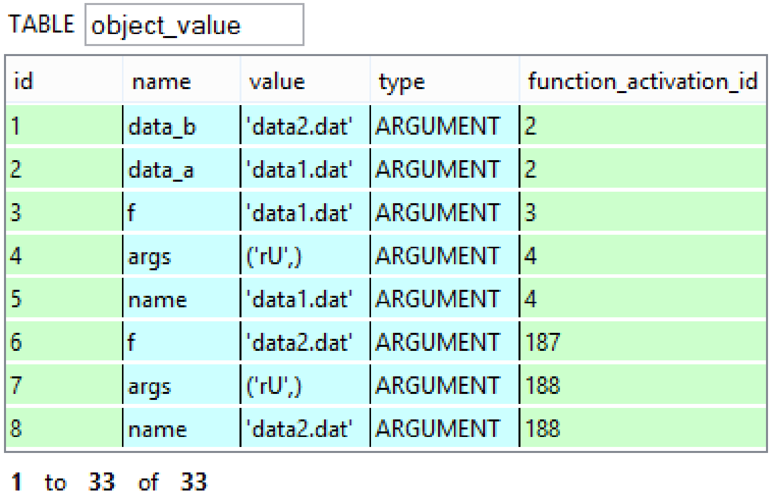


TABLE object_value

id	name	value	type	function_activation_id
1	data_b	'data2.dat'	ARGUMENT	2
2	data_a	'data1.dat'	ARGUMENT	2
3	f	'data1.dat'	ARGUMENT	3
4	args	('rU',)	ARGUMENT	4
5	name	'data1.dat'	ARGUMENT	4
6	f	'data2.dat'	ARGUMENT	187
7	args	('rU',)	ARGUMENT	188
8	name	'data2.dat'	ARGUMENT	188

1 to 33 of 33

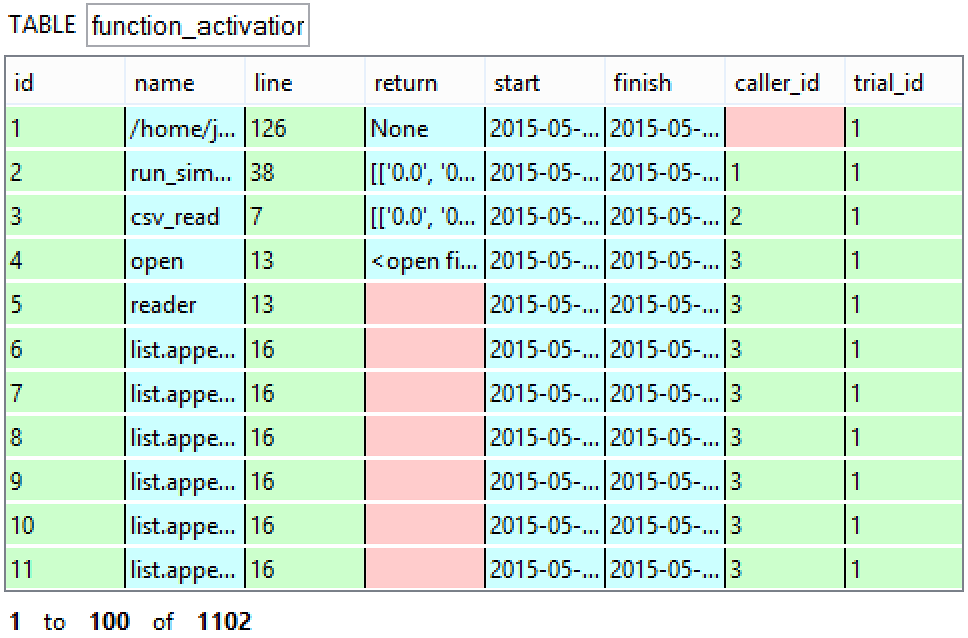


TABLE function_activatior

id	name	line	return	start	finish	caller_id	trial_id
1	/home/j...	126	None	2015-05-...	2015-05-...		1
2	run_sim...	38	[['0.0', '0...	2015-05-...	2015-05-...	1	1
3	csv_read	7	[['0.0', '0...	2015-05-...	2015-05-...	2	1
4	open	13	<open fi...	2015-05-...	2015-05-...	3	1
5	reader	13		2015-05-...	2015-05-...	3	1
6	list.appe...	16		2015-05-...	2015-05-...	3	1
7	list.appe...	16		2015-05-...	2015-05-...	3	1
8	list.appe...	16		2015-05-...	2015-05-...	3	1
9	list.appe...	16		2015-05-...	2015-05-...	3	1
10	list.appe...	16		2015-05-...	2015-05-...	3	1
11	list.appe...	16		2015-05-...	2015-05-...	3	1

1 to 100 of 1102

Trial

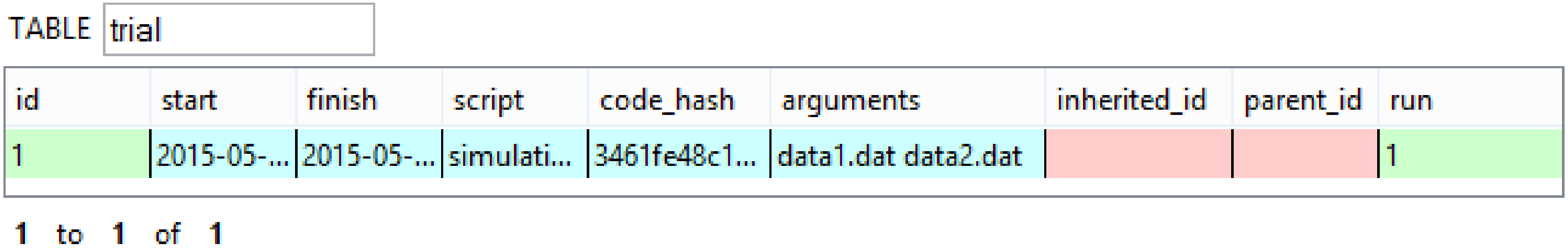


TABLE trial

id	start	finish	script	code_hash	arguments	inherited_id	parent_id	run
1	2015-05-...	2015-05-...	simulati...	3461fe48c1...	data1.dat data2.dat			1

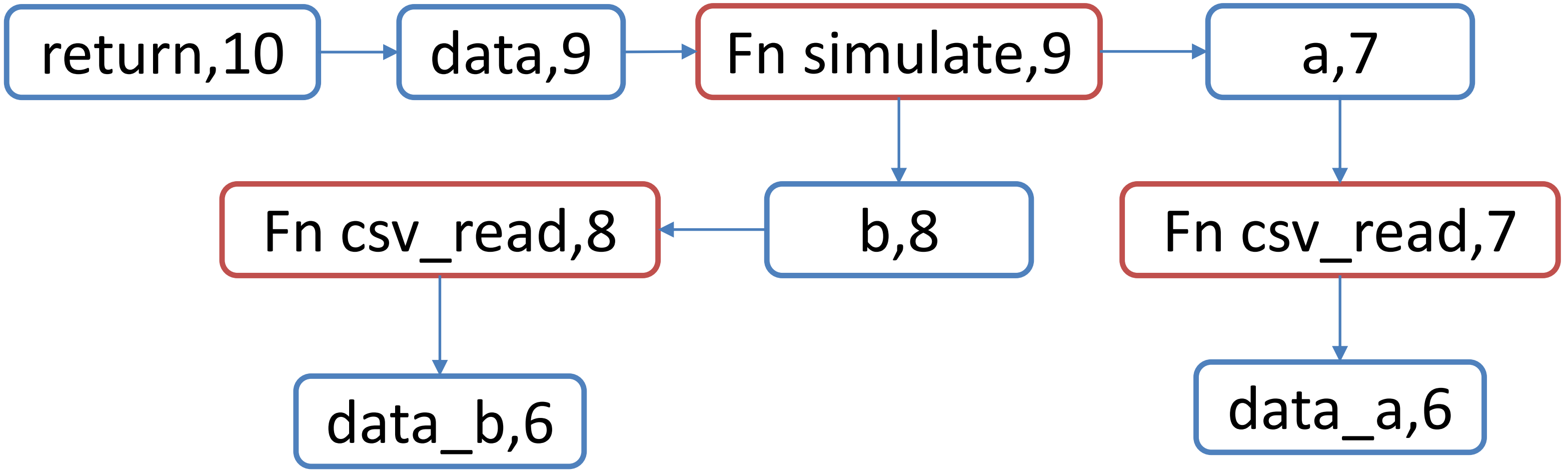
1 to 1 of 1

Fine-grained Collection

$ now run –e Tracer simulation.py data1.dat data2.dat

- Default provenance
- Variable assignments
- Dependencies

```
 6| def run_simulation(data_a, data_b):
 7|     a = csv_read(data_a)
 8|     b = csv_read(data_b)
 9|     data = simulate(a, b)
10|     return data
```



return,10

data,9

Fn simulate,9

a,7

Fn csv_read,8

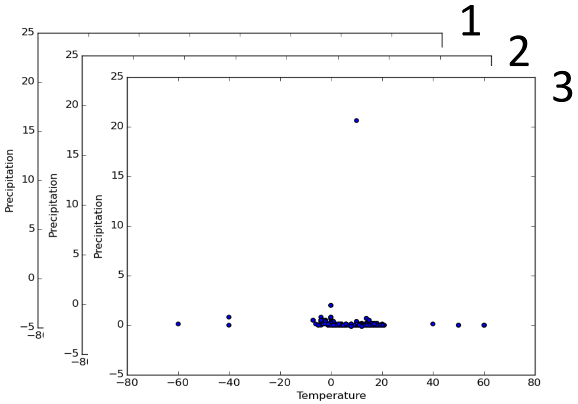
b,8

Fn csv_read,7

data_b,6

data_a,6

Management


1
2
3
Precipitation
Temperature
25
20
15
10
5
0
−5
−80
−60
−40
−20
0
20
40
60
80

Restore

(1) $ now run simulation.py data1.dat data2.dat

(2) $ now run simulation.py data1.dat data3.dat

(3) $ now run simulation.py

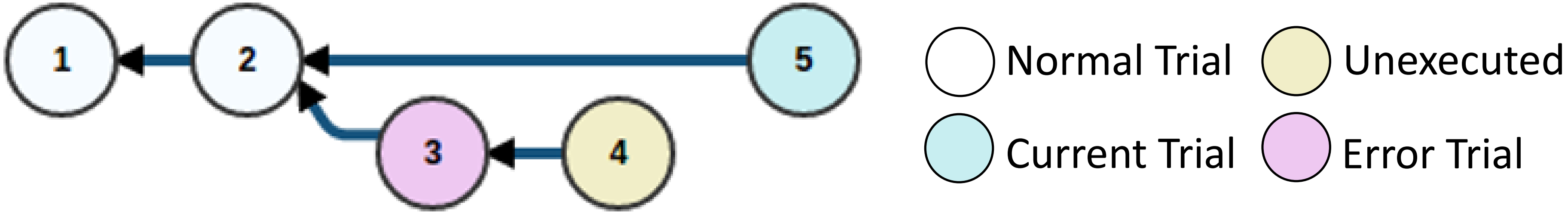
Error

Try to fix simulation.py, save it, but do not run it.

(4) $ now restore -li 2

Restores local modules, input and output files from trial 2

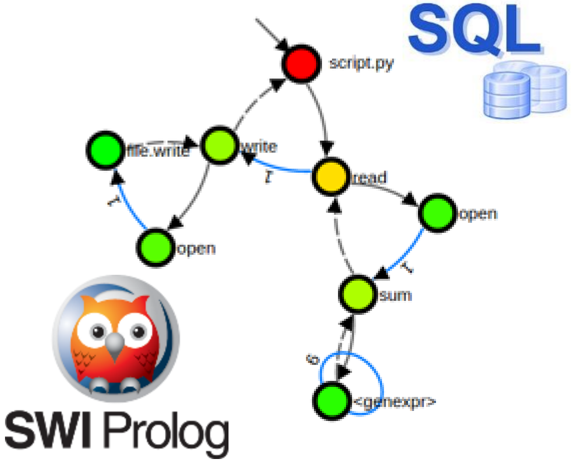
(5) $ now run simulation.py data1.dat data4.dat



$ now restore –li 2

```
[now] Backup Trial 4 created
[now] File simulation.py from trial 2 restored
[now] File
/home/joao/projects/nowissues/scipyla_weather/simulator.py from
trial 2 restored
[now] File output.png from trial 2 restored
[now] File data3.dat from trial 2 restored
[now] File data1.dat from trial 2 restored
```

Analysis



Command line

- $ now list
- $ now show [trial]
- $ now diff [trial1] [trial2]
- $ now export –r [trial]
- $ now export –i [trial]
- $ now export –i history
- $ now export –i diff:[trial1]:[trial2]
- $ now vis

$ now list

```
[now] trials available in the provenance store:
  Trial 1: simulation.py data1.dat data2.dat
           with code hash 3461fe48c17619faeb81a6334567a47a162c7788
           ran from 2015-05-15 00:45:09.908030 to 2015-05-15
00:45:20.602758
  Trial 2: simulation.py data1.dat data3.dat
           ...
  Trial 3: simulation.py
           ...
  Trial 4: simulation.py <restore 2>
           with code hash cf879bd94d8c5942de800a9a17aed2f98acd1300
           ran from 2015-05-15 00:47:31.103699 to None
  Trial 5: simulation.py data1.dat data4.dat
           with code hash 3461fe48c17619faeb81a6334567a47a162c7788
           ran from 2015-05-15 01:03:44.096991 to 2015-05-15
01:03:54.803461
```

$ now show 5 -a

```
[now] trial information:
  Id: 5
  Inherited Id: None
  Script: simulation.py
  Code hash: 3461fe48c17619faeb81a6334567a47a162c7788
  Start: 2015-05-15 01:03:44.096991
  Finish: 2015-05-15 01:03:54.803461
[now] this trial has the following function activation graph:
  126:
/home/joao/projects/nowissues/scipyla_weather/simulation.py
(2015-05-15 01:03:52.165483 - 2015-05-15 01:03:54.803433)
       Globals, Arguments, Return value
    38: run_simulation (2015-05-15 01:03:52.165687 - 2015-05-15
01:03:54.176796)
        Globals:
        Arguments: data_b = 'data4.dat', data_a = 'data1.dat'
```

$ now diff 1 2

```
[now] trial diff:
  duration changed from 10694728 to 10662788
  start changed from 2015-05-15 00:45:09.908030 to 2015-05-15
00:45:37.045604
  finish changed from 2015-05-15 00:45:20.602758 to 2015-05-15
00:45:47.708392
  arguments changed from data1.dat data2.dat to data1.dat
data3.dat
  parent_id changed from None to 1
```

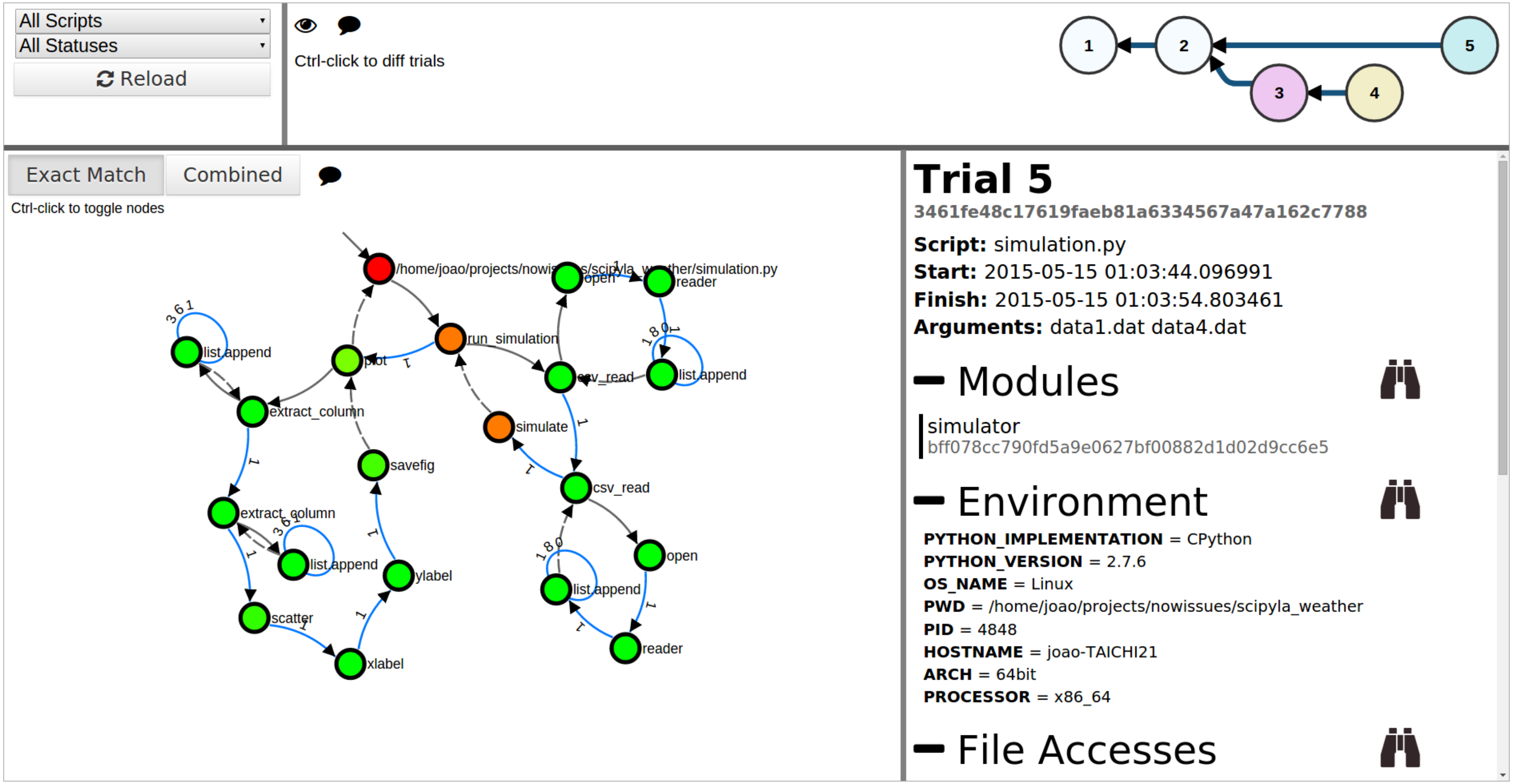
Export

- Trial to Prolog: **$ now export –r 1 > trial1.pl**
- Trial to IPython Notebook: **$ now export –i 1**
- History to Notebook: **$ now export –i history**
- Diff to Notebook: **$ now export –i diff:1:2**

SWI Prolog

IP[y]: IPython Interactive Computing

$ now vis



Trial 5

3461fe48c17619faeb81a6334567a47a162c7788

Script: simulation.py
Start: 2015-05-15 01:03:44.096991
Finish: 2015-05-15 01:03:54.803461
Arguments: data1.dat data4.dat

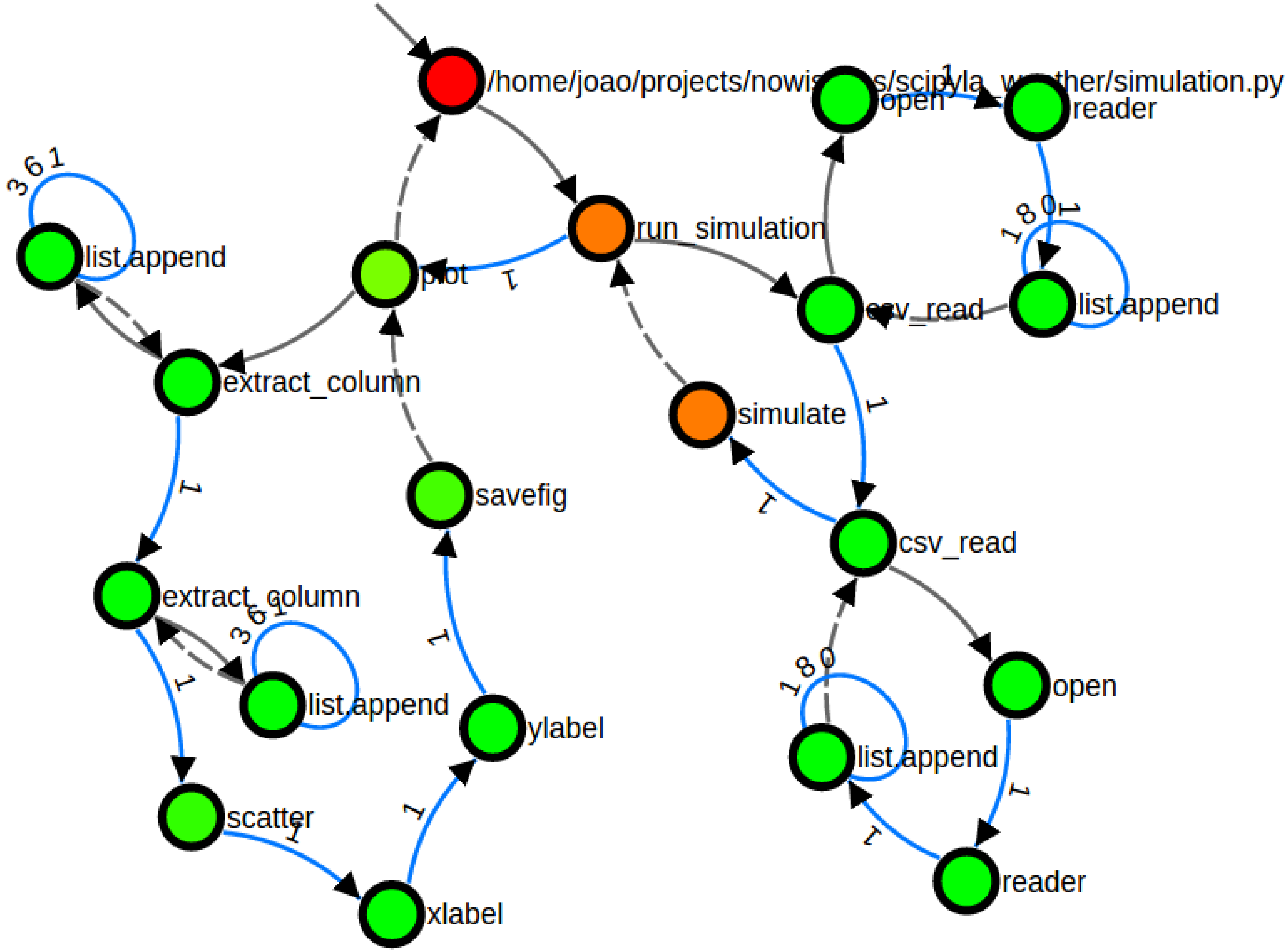
− Modules

simulator
bff078cc790fd5a9e0627bf00882d1d02d9cc6e5

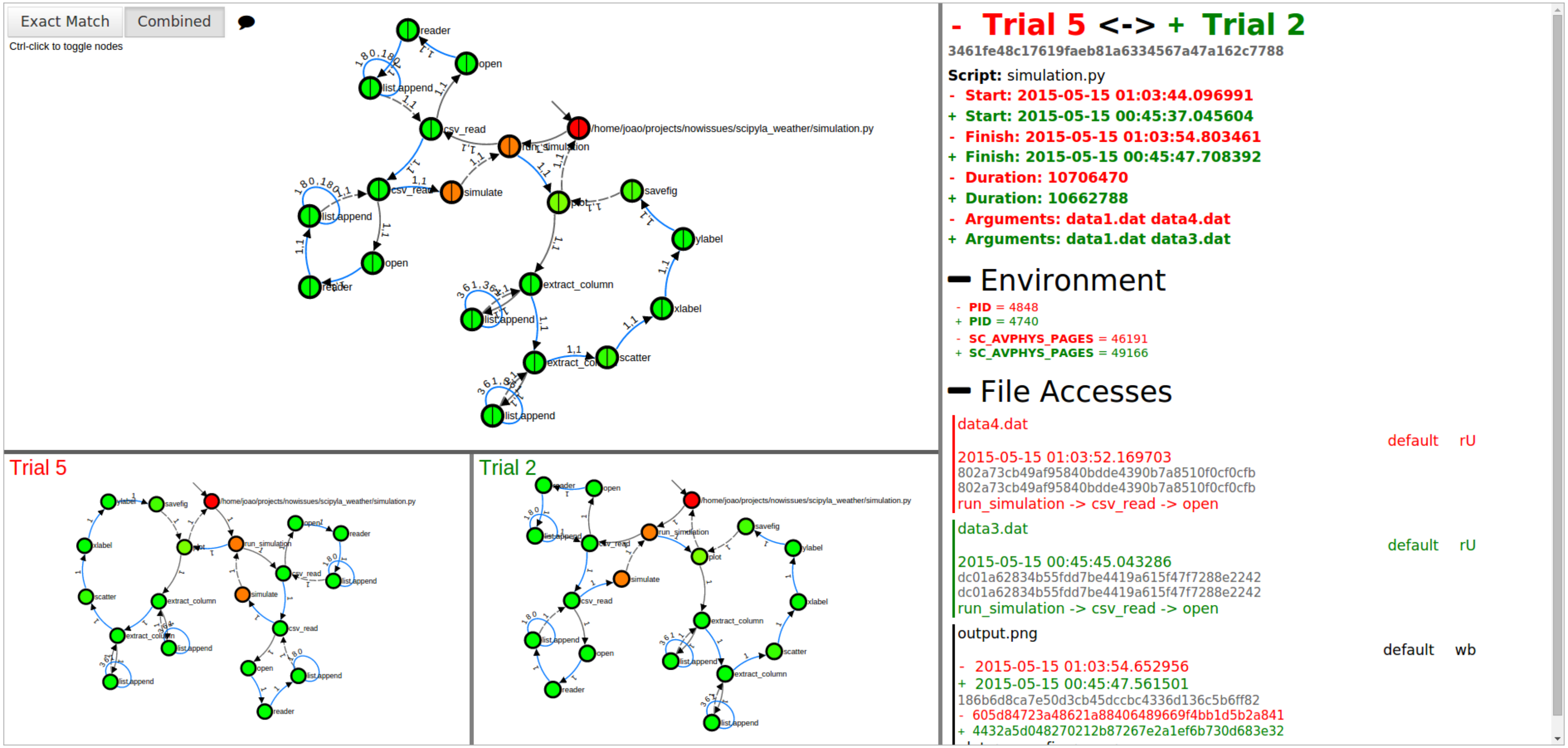
− Environment

PYTHON_IMPLEMENTATION = CPython
PYTHON_VERSION = 2.7.6
OS_NAME = Linux
PWD = /home/joao/projects/nowissues/scipyla_weather
PID = 4848
HOSTNAME = joao-TAICHI21
ARCH = 64bit
PROCESSOR = x86_64

− File Accesses

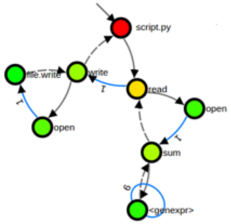

/home/joao/projects/nowis...s/scipyla_weather/simulation.py
open
1
reader
run_simulation
plot
1
list.append
3 6 1
extract_column
csv_read
1 8 0
1
list.append
simulate
1
csv_read
1
savefig
extract_column
3 6 1
1
list.append
1
ylabel
1
scatter
1
xlabel
open
1 8 0
list.append
1
1
reader

$ now vis

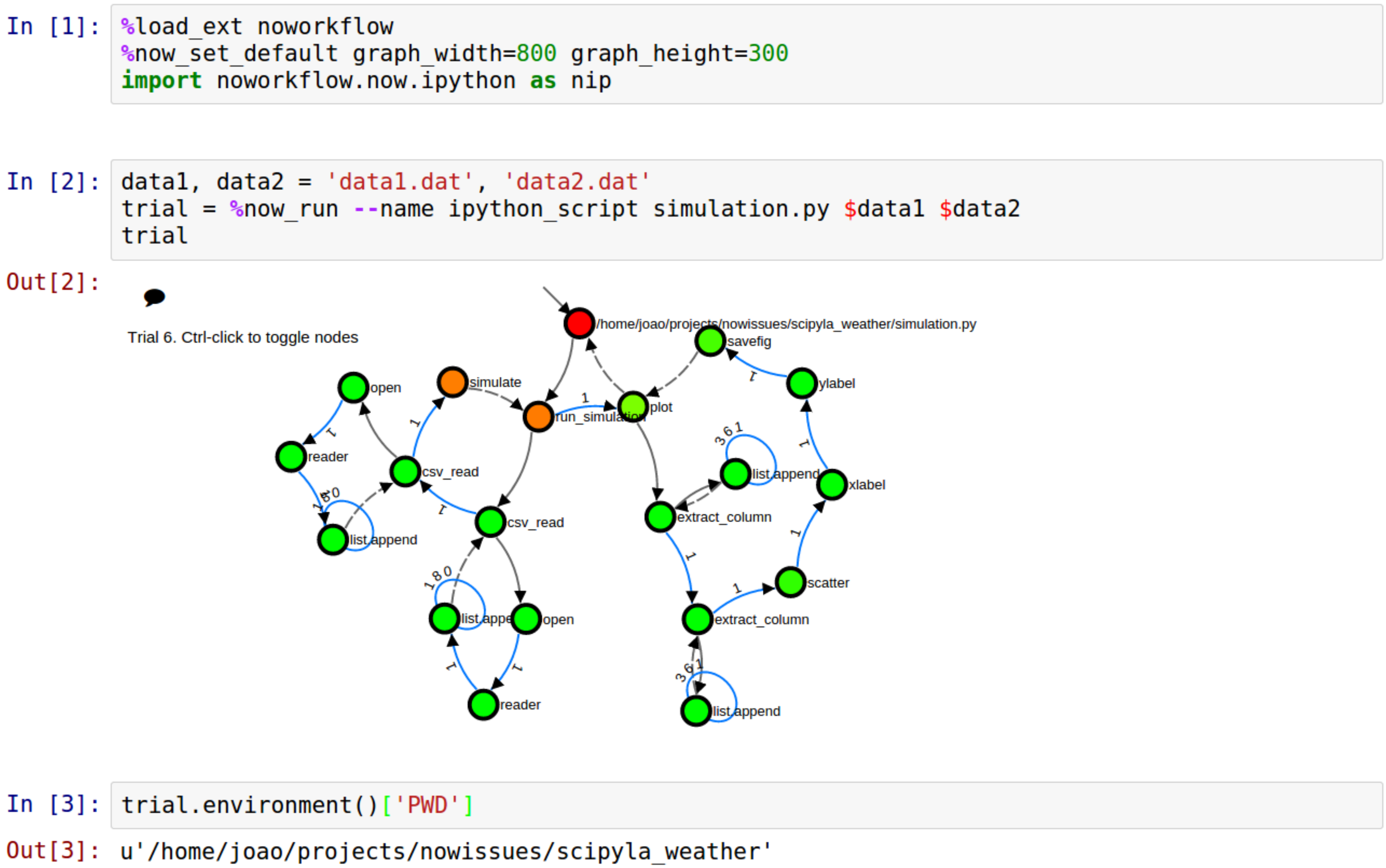


```
-  Trial 5 <-> +  Trial 2
3461fe48c17619faeb81a6334567a47a162c7788
Script: simulation.py
-  Start: 2015-05-15 01:03:44.096991
+  Start: 2015-05-15 00:45:37.045604
-  Finish: 2015-05-15 01:03:54.803461
+  Finish: 2015-05-15 00:45:47.708392
-  Duration: 10706470
+  Duration: 10662788
-  Arguments: data1.dat data4.dat
+  Arguments: data1.dat data3.dat

━ Environment
 -  PID = 4848
 +  PID = 4740
 -  SC_AVPHYS_PAGES = 46191
 +  SC_AVPHYS_PAGES = 49166

━ File Accesses
 data4.dat
                                                    default   rU
 2015-05-15 01:03:52.169703
 802a73cb49af95840bdde4390b7a8510f0cf0cfb
 802a73cb49af95840bdde4390b7a8510f0cf0cfb
 run_simulation -> csv_read -> open
 data3.dat
                                                    default   rU
 2015-05-15 00:45:45.043286
 dc01a62834b55fdd7be4419a615f47f7288e2242
 dc01a62834b55fdd7be4419a615f47f7288e2242
 run_simulation -> csv_read -> open
 output.png
                                                    default   wb
 -  2015-05-15 01:03:54.652956
 +  2015-05-15 00:45:47.561501
 186b6d8ca7e50d3cb45dccbc4336d136c5b6ff82
 -  605d84723a48621a88406489669f4bb1d5b2a841
 +  4432a5d048270212b87267e2a1ef6b730d683e32
```

IPython meets noWorkflow



$ ipython notebook



```
In [1]: %load_ext noworkflow
        %now_set_default graph_width=800 graph_height=300
        import noworkflow.now.ipython as nip
```

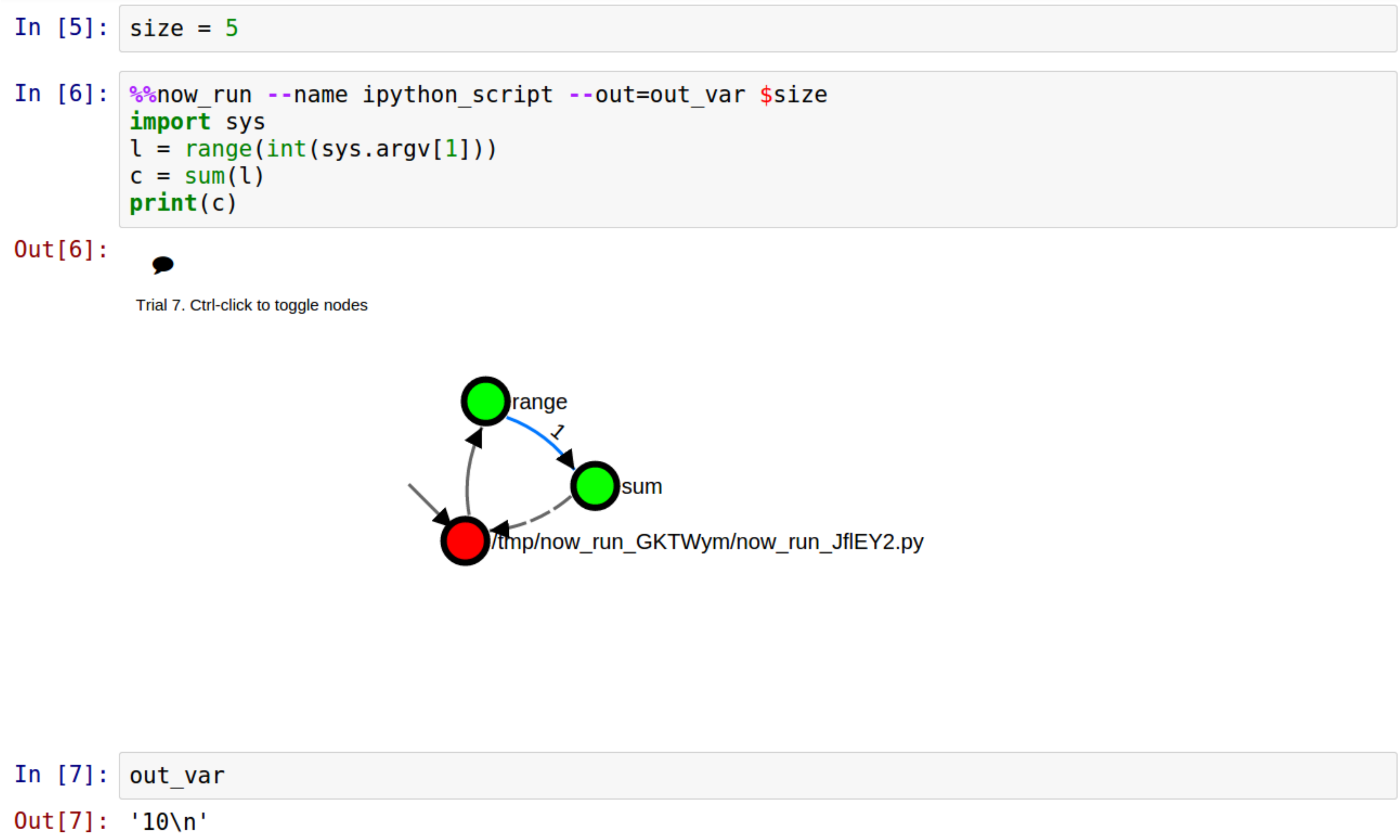
```
In [2]: data1, data2 = 'data1.dat', 'data2.dat'
        trial = %now_run --name ipython_script simulation.py $data1 $data2
        trial
```

Out[2]:

Trial 6. Ctrl-click to toggle nodes

```
In [3]: trial.environment()['PWD']
Out[3]: u'/home/joao/projects/nowissues/scipyla_weather'
```

Collection



```
In [5]: size = 5
```

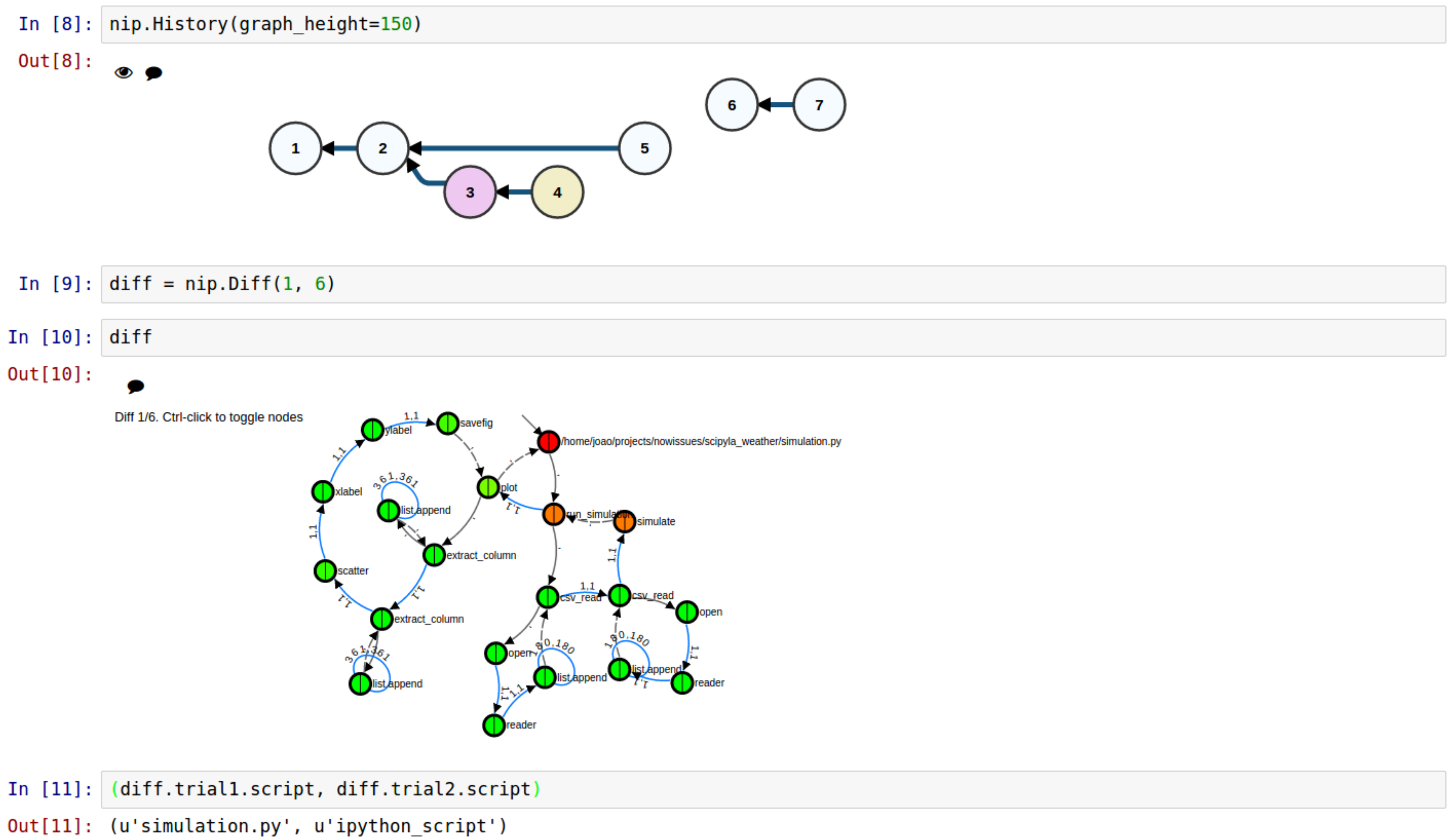
```
In [6]: %%now_run --name ipython_script --out=out_var $size
        import sys
        l = range(int(sys.argv[1]))
        c = sum(l)
        print(c)
```

```
Out[6]:
```

Trial 7. Ctrl-click to toggle nodes

```
In [7]: out_var
Out[7]: '10\n'
```

Analysis



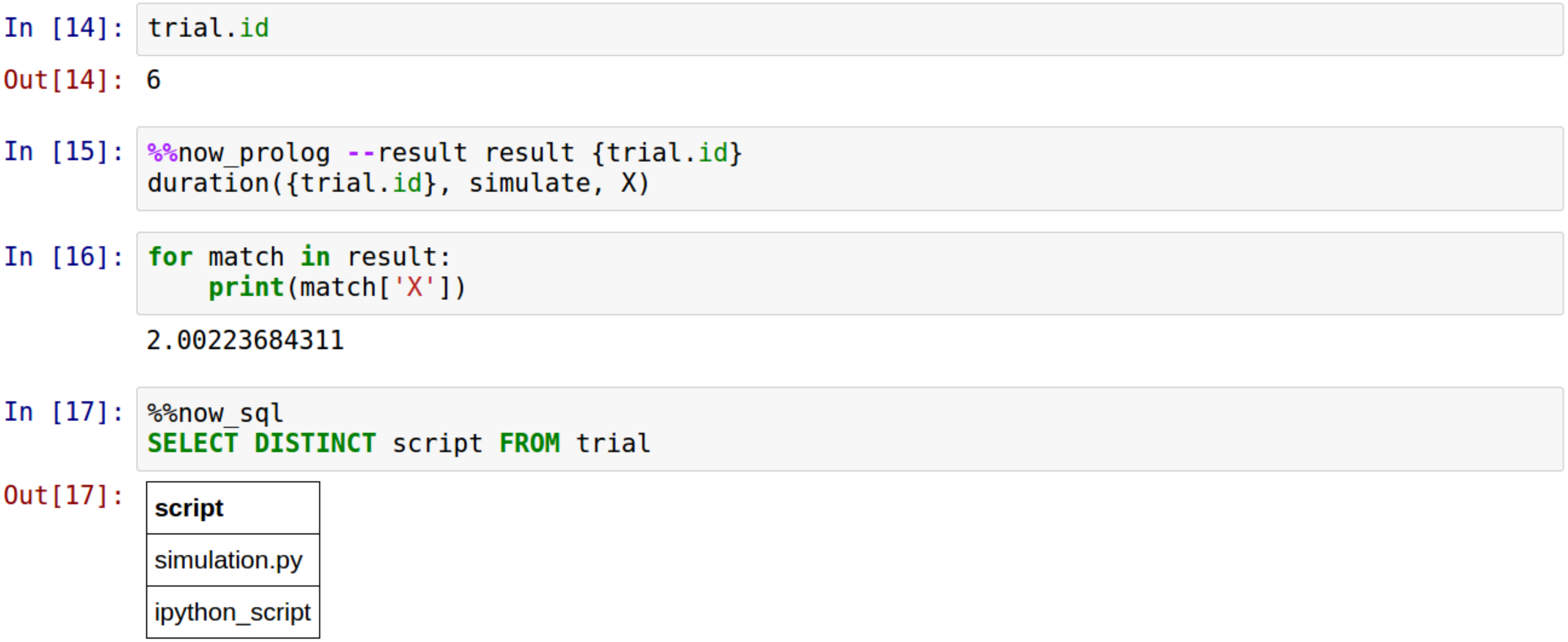
```
In [8]: nip.History(graph_height=150)
Out[8]:
```

```
In [9]: diff = nip.Diff(1, 6)

In [10]: diff
Out[10]:
```

Diff 1/6. Ctrl-click to toggle nodes

```
In [11]: (diff.trial1.script, diff.trial2.script)
Out[11]: (u'simulation.py', u'ipython_script')
```

Queries

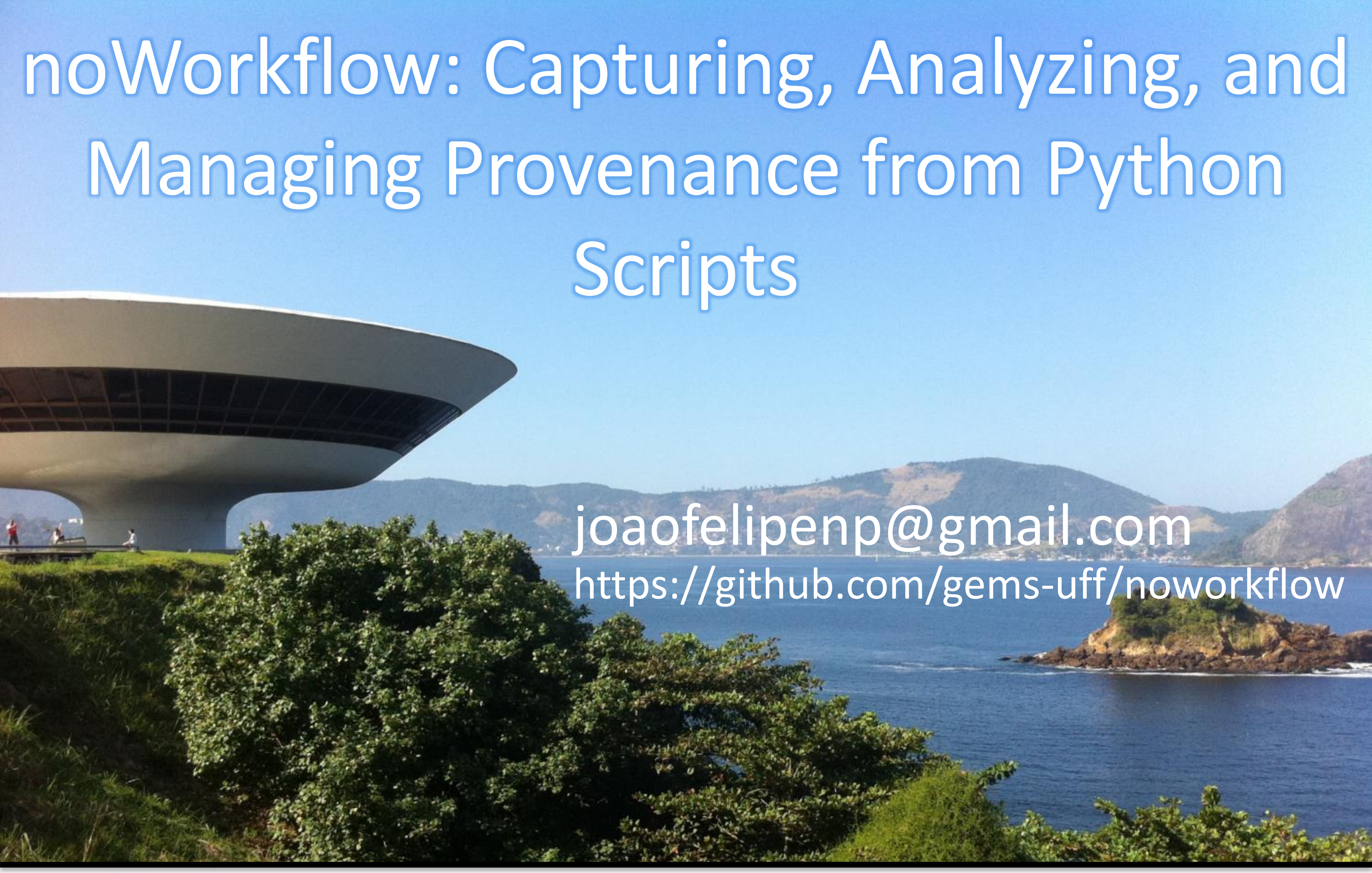


```
In [14]: trial.id
Out[14]: 6

In [15]: %%now_prolog --result result {trial.id}
         duration({trial.id}, simulate, X)

In [16]: for match in result:
             print(match['X'])

         2.00223684311

In [17]: %%now_sql
         SELECT DISTINCT script FROM trial
Out[17]:
```

script
simulation.py
ipython_script

Conclusion

- noWorkflow allows users to capture and analyze provenance from Python Scripts
- It is easy to install and use it
  - **$ pip install noworkflow[all]**
- Open source. Please, submit issues at
  - https://github.com/gems-uff/noworkflow



noWorkflow: Capturing, Analyzing, and Managing Provenance from Python Scripts

joaofelipenp@gmail.com
https://github.com/gems-uff/noworkflow

Instituto de computação uff

João Felipe Nicolaci Pimentel (UFF), Juliana Freire (NYU), Vanessa Braganholo (UFF), Leonardo Murta (UFF)

NYU
POLYTECHNIC SCHOOL OF ENGINEERING