Snowballing Tool

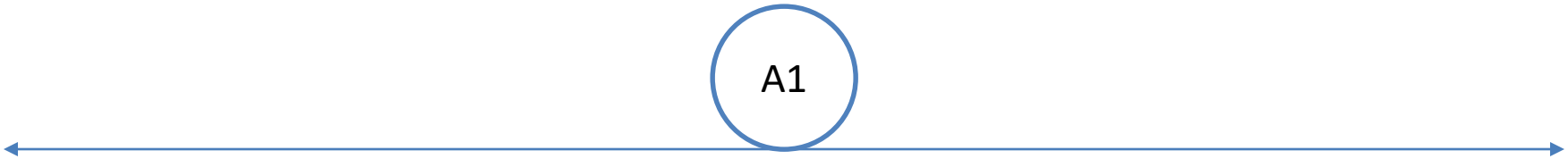
Instituto de Computação

João Felipe Pimentel

ESI
Laboratório de Engenharia de Sistemas e Informação

Snowballing

- Search approach for systematic literature review
- Start set: {A1}



Citations

A1

1 Introduction

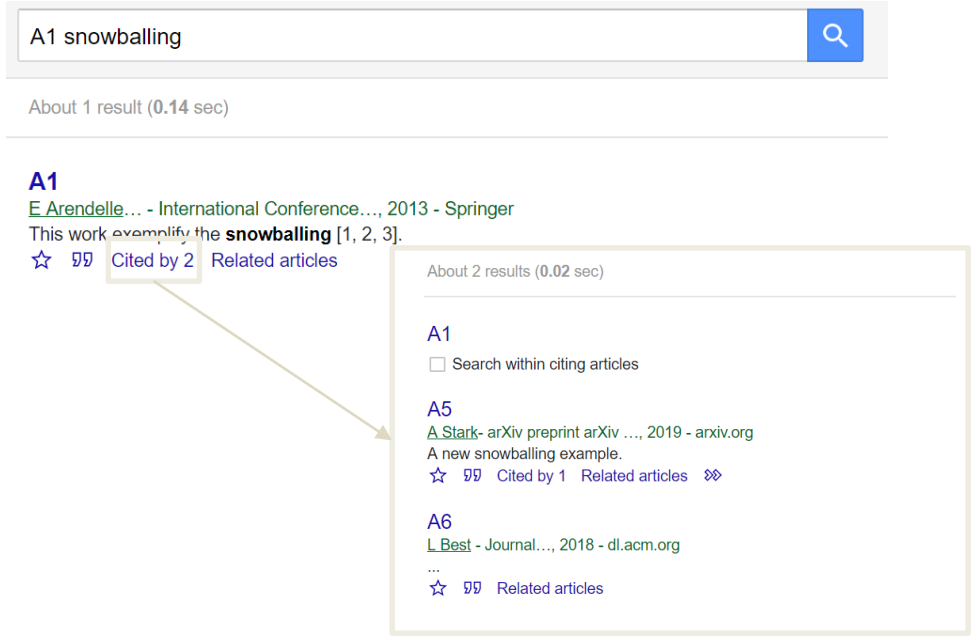
Other work exemplify the snowballing [1, 2, 3].

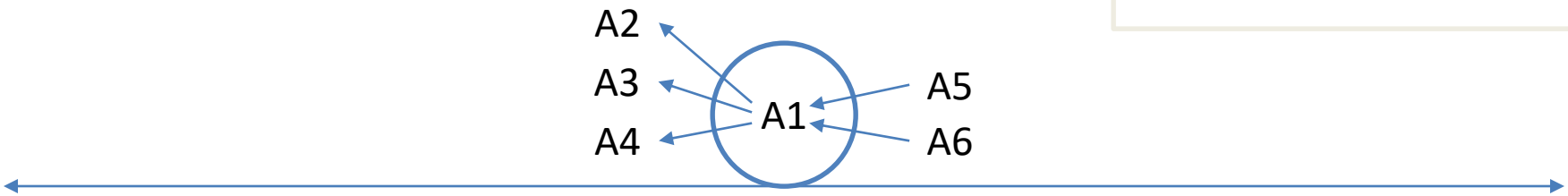
References

[1] A2, 2013.

[2] A3, 2014.

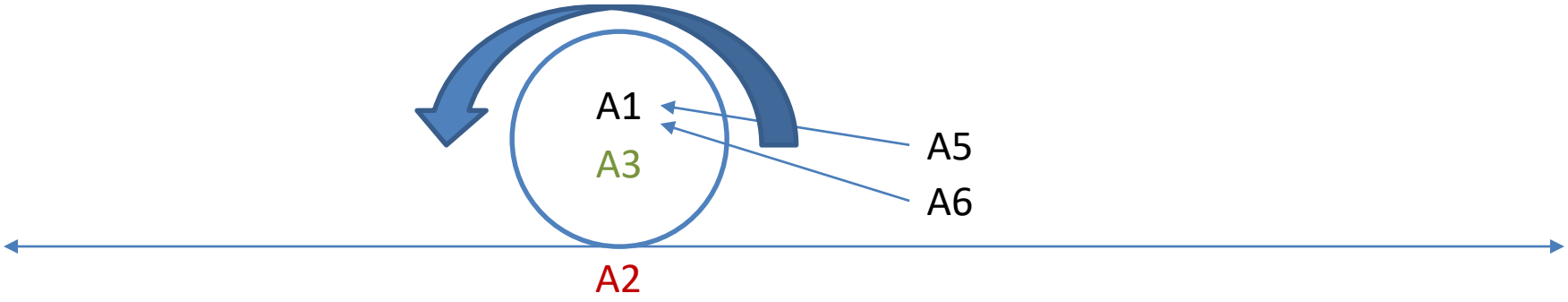
[3] A4, 2018.





Backward Snowballing

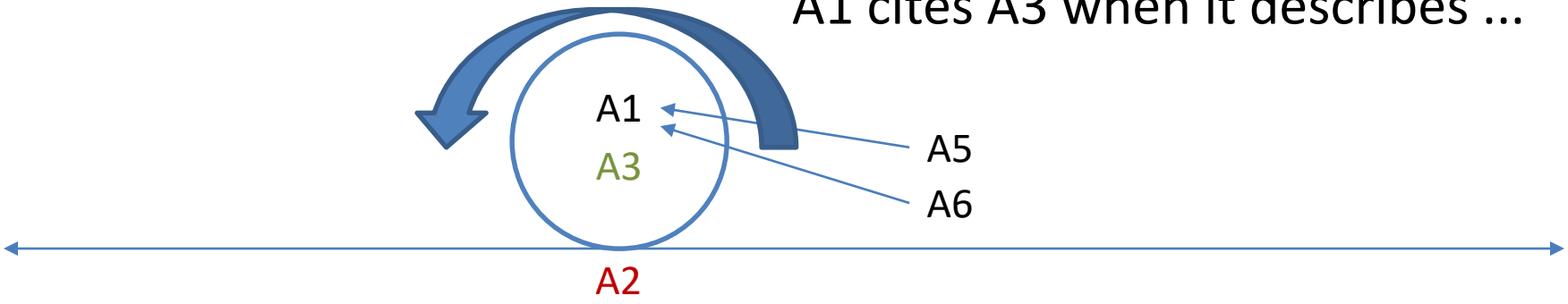
- Which papers where cited by A1 and are related?



Annotations

"A3 approaches the issue by..."

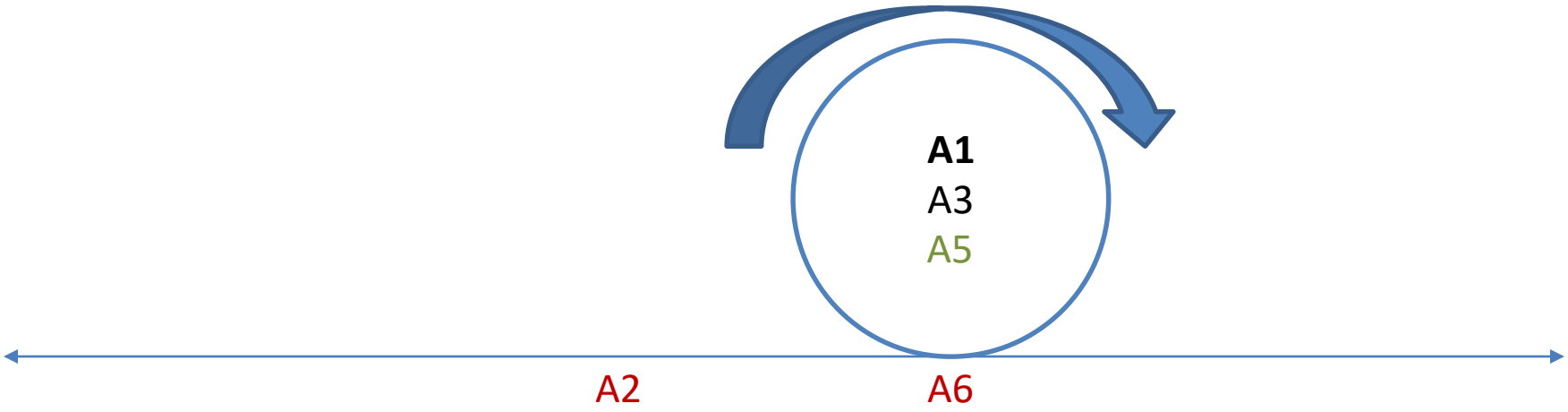
"A1 cites A3 when it describes ..."



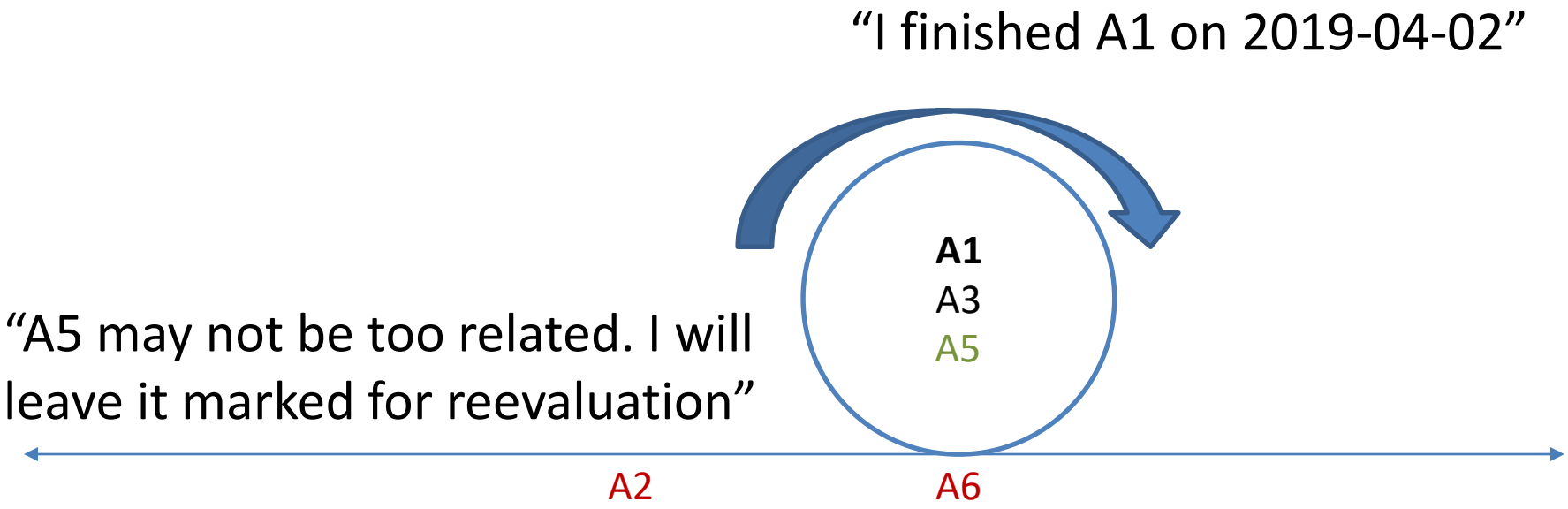
"A2 is not related due to ..."

Forward Snowballing

- Which papers cite A1 and are related?

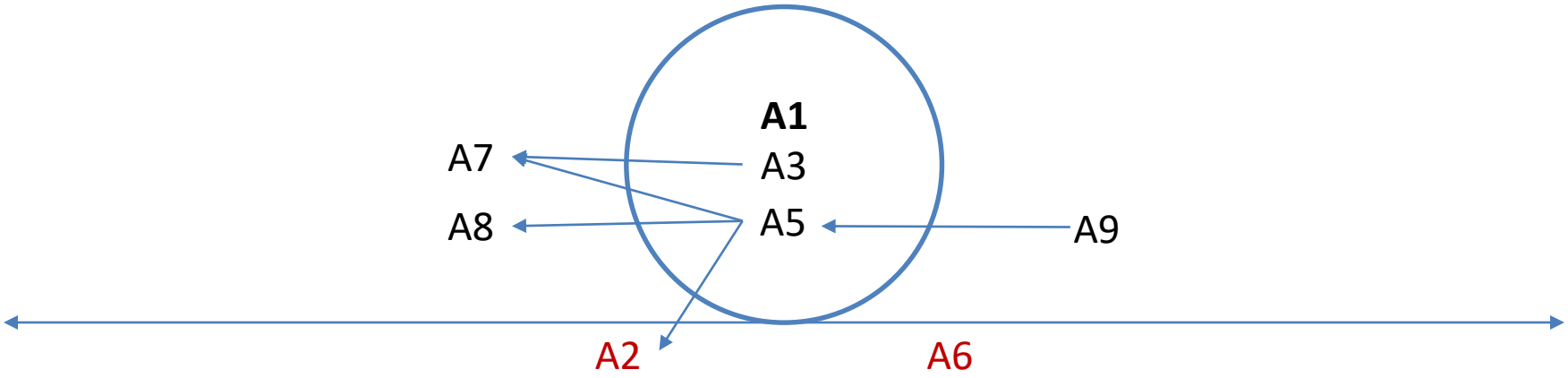


Other Annotation Types

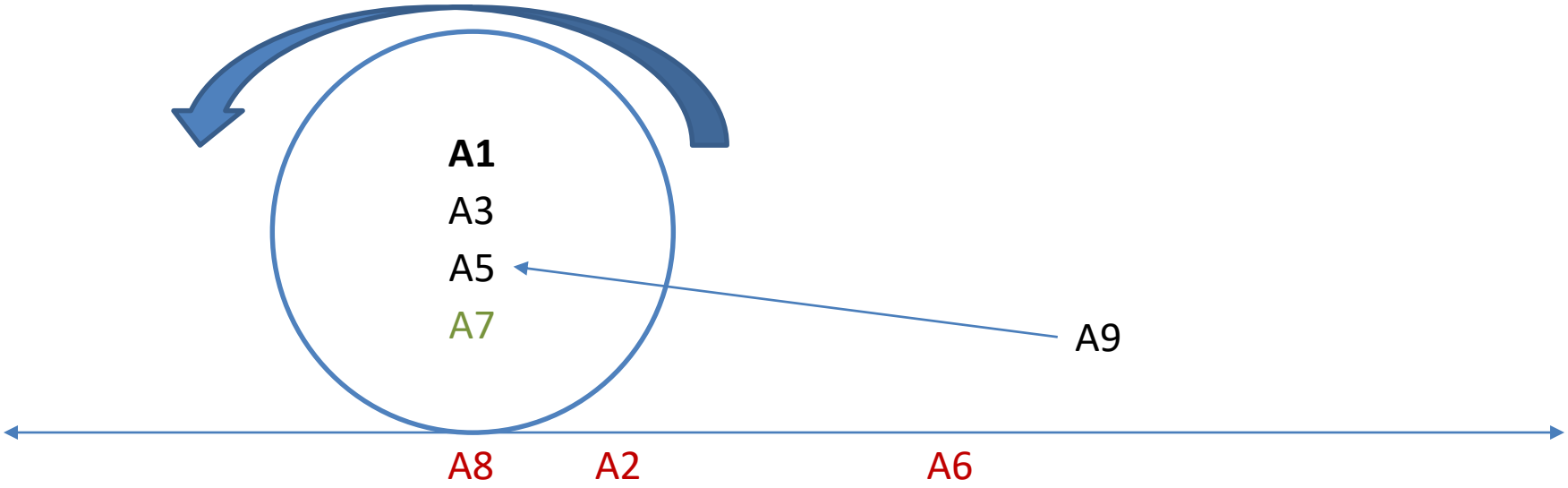


Continuing the process

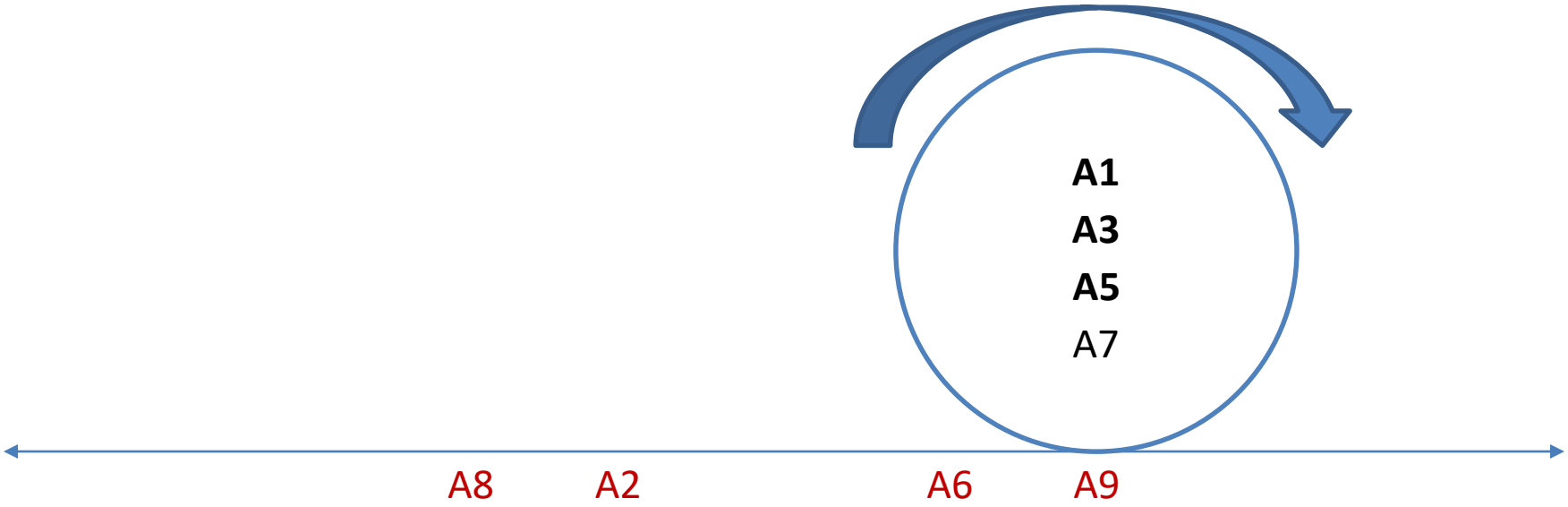
- A1 is complete, but A3 and A5 also have references and citations



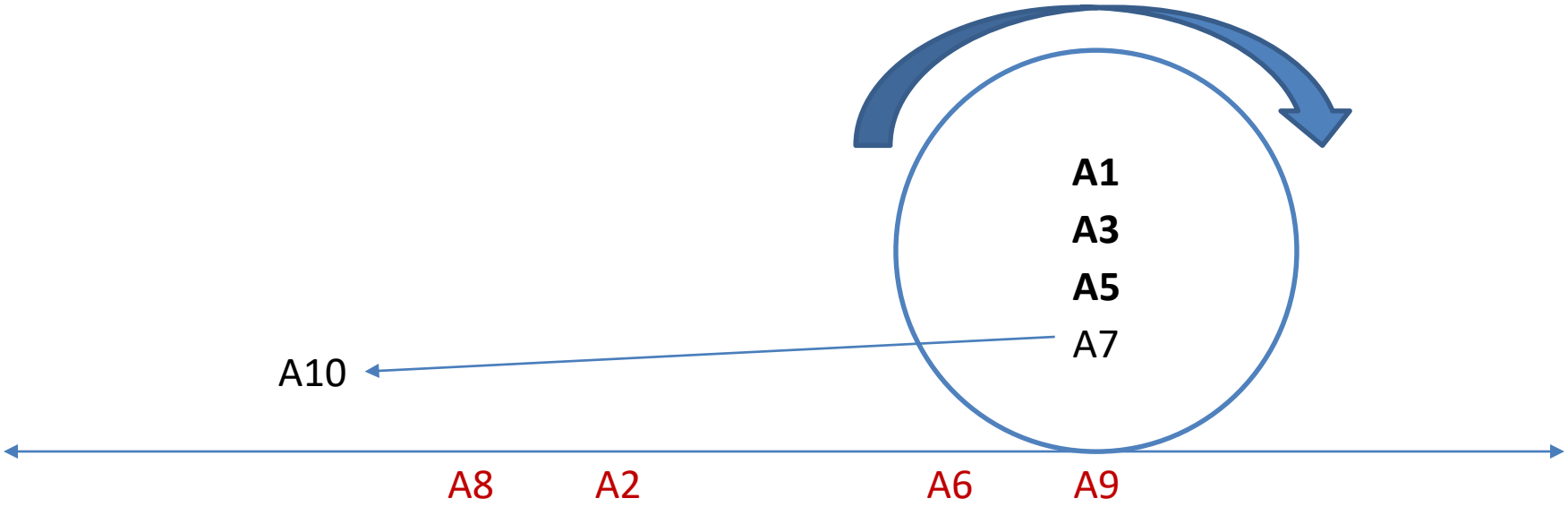
Backward



Forward

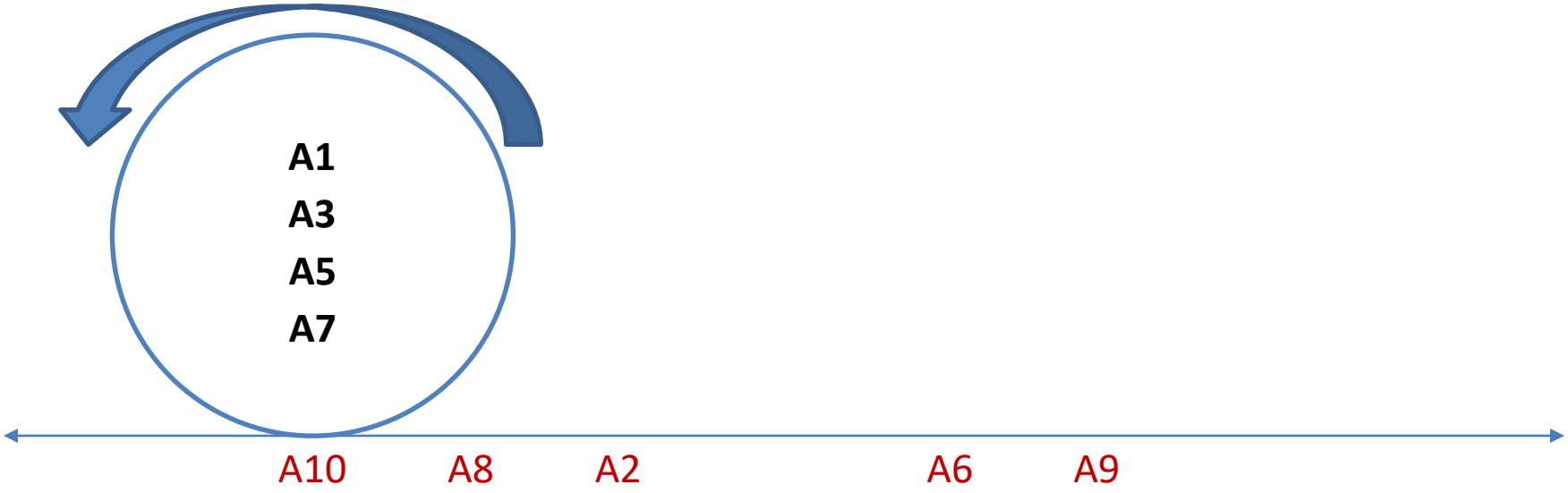


References and Citations



Stop Condition

- No new related reference nor related citation



Challenges

- Laborious and repetitive process
- After hundreds of papers, it gets hard to remember all discarded
  - It is good to record all visited papers and not only the related ones
- The recording must be easy, consistente, and extensible
- Searches and analyses must be flexible for querying the records

Tool

- Set of scripts and Jupyter Notebooks
  - Developed during a snowballing process to automatize and support the process
  - Some scripts are related to the **search/snowballing** process by itself
  - Others are related to the **analysis** process
- The tool is divided into two parts:
  - Generic **library**
  - **Project** workspace

Getting Started

- Install the generic library
  - $ pip install snowballing
- Install geckodriver (https://github.com/mozilla/geckodriver/releases)
  - I extract it in the Firefox directory, but I'm not sure it is mandatory: C:\Program Files\Mozilla Firefox
  - Add the directory to the PATH
- Install graphviz and add it to the PATH
- Start a new project:
  - $ snowballing start <name>

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

database Structure

- __init__.py - Basic configuration
- places.py - Stores publication places
- groups - Stores approaches. Ignore this for the search phase
- work - Stores papers
  - y2008.py - Papers from 2008. Remove it
  - y2014.py - Papers from 2014. Remove it
  - y2015.py - Papers from 2015. Remove it

  Sample Files
  - y9999.py - Papers without year. **Do not** remove it
- citations - Stores citations. A file for each related work
  - murta2014a.py - Sample file. Remove it

places.py Source Code

```
from snowballing.models import Place, DB
from snowballing.common_places import *

IPAW = conference("IPAW", "International
Provenance and Annotation Workshop")
CSUR = journal("CSUR", "ACM Computing Surveys")

arXiv = DB(Place("arXiv", "arXiv", "Archive"))
```

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando
and Koop, David and Freire, Juliana",
    place=IPAW,




))
```

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando and Koop, David and Freire, Juliana",
    place=IPAW,
))
```

var name

snowballing state

config.CLASSES no __init__.py:

- Work: inserted, but with no decision
- WorkNoFile: file not found
- WorkLang: in another language
- WorkUnrelated: unrelated paper
- WorkOk: considered, but before the backward
- WorkSnowball: related, after backward

Name for the analyses

Publication place. Refers a variable from places.py

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando
and Koop, David and Freire, Juliana",
    place=IPAW,
    example="1",
    todo="remove this example",
    analysis="""graph-based: summarization of the activation graph;
                diff-based: basic attributes comparison;
                query-based: queries""",
))
```

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando and Koop, David and Freire, Juliana",
    place=IPAW,
    aliases=[(2015, "noWorkflow: Capturing and Analyzing Provenance of Scripts", "Chirigati, Fernando and Koop, David and Freire, Juliana")],
    snowball=datetime(2017, 3, 6),
    file="murta2014a.pdf",
    citation_file="murta2014a",
))
```

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando
and Koop, David and Freire, Juliana",
    place=IPAW,
    scholar="http://scholar.google.com/scholar?cites=
5458343950729529273&as_sdt=2005&sciodt=0,5&hl=en",
    scholar_id="ucciVefuv0sJ",
    cluster_id="5458343950729529273",
    scholar_ok=True,
))
```

y2014.py Source Code

```
from datetime import datetime
from snowballing.models import *
from ..places import *

murta2014a = DB(WorkSnowball(
    2014, "noWorkflow: capturing and analyzing provenance of scripts",
    display="noWorkflow",
    authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando
and Koop, David and Freire, Juliana",
    place=IPAW,
    local="Cologne, Germany",
    pp="71--83",
    entrytype="inproceedings",
    organization="Springer",
    editor="Ludaescher, Bertram and Plale, Beth",
))
```

citations/*.py Source Code

```
from snowballing.models import *
from snowballing import dbindex
dbindex.last_citation_file = dbindex.this_file(__file__)

from ..work.y2008 import freire2008a
from ..work.y2014 import murta2014a
from ..work.y2015 import pimentel2015a

DB(Citation(
   murta2014a, freire2008a,




))
```

citations/*.py Source Code

```
from snowballing.models import *
from snowballing import dbindex
dbindex.last_citation_file = dbindex.this_file(__file__)

from ..work.y2008 import freire2008a
from ..work.y2014 import murta2014a
from ..work.y2015 import pimentel2015a

DB(Citation(
    murta2014a, freire2008a, ref="5",
    contexts=[
        "There are two types of provenance for scientific workflows: prospective and retrospective [5]. Prospective provenance describes the structure of the experiment and corresponds to the workflow definition, the graph of the activities, and their associated parameters. Retrospective provenance captures the steps taken during the workflow execution, and while it has similar (graph) structure, it is constructed using information collected at runtime, including activities invoked and parameter values used, intermediate data produced, the execution start and end times, etc"
    ],
))
```

Project Structure

database - Snowballing data storage

files - Papers' PDF storage

notebooks - Analysis Notebooks

.gitignore - Ignores the files directory and Python cache files


Backward.ipynb - Supports Backward Snowballing


Forward.ipynb - Supports Forward Snowballing


Index.ipynb - Describes the Project Structure


Insert.ipynb - Supports the insertion of papers from BibTex


Progress.ipynb - Monitors the Snowballing progress


SearchScholar.ipynb - Supports the insertion of papers from Scholar


Validate.ipynb - Validates and updates the insertions

Insert.ipynb 1/4

In [1]:
```
import database
from snowballing.operations import load_work, reload, work_by_varname
from snowballing.snowballing import Converter
from snowballing.snowballing import ArticleNavigator
from snowballing.dbmanager import insert, set_attribute
```

1

In [2]:
```
Converter("bibtex")
```

2

BibTeX | Set article_list variable | 1

3

```
@article{pimentel2017noworkflow,
  title={noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts},
  author={Pimentel, Jo{\~a}o Felipe and Murta, Leonardo and Braganholo, Vanessa and Freire, Juliana},
  journal={Proceedings of the VLDB Endowment},
  volume={10},
  number={12},
  year={2017}
}
```

4

```
[
 {
  "name": "noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts",
  "authors": "Pimentel, Jo\u00e3o Felipe and Murta, Leonardo and Braganholo, Vanessa and Freire, Juliana",
  "year": 2017,
  "place1": "Proceedings of the VLDB Endowment",
  "entrytype": "article",
  "display": "pimentel",
  "pyref": "pimentel2017a",
  "volume": "10",
  "number": "12",
  "ID": "pimentel2017noworkflow",
  "_work_type": "Work"
 }
]
```

5

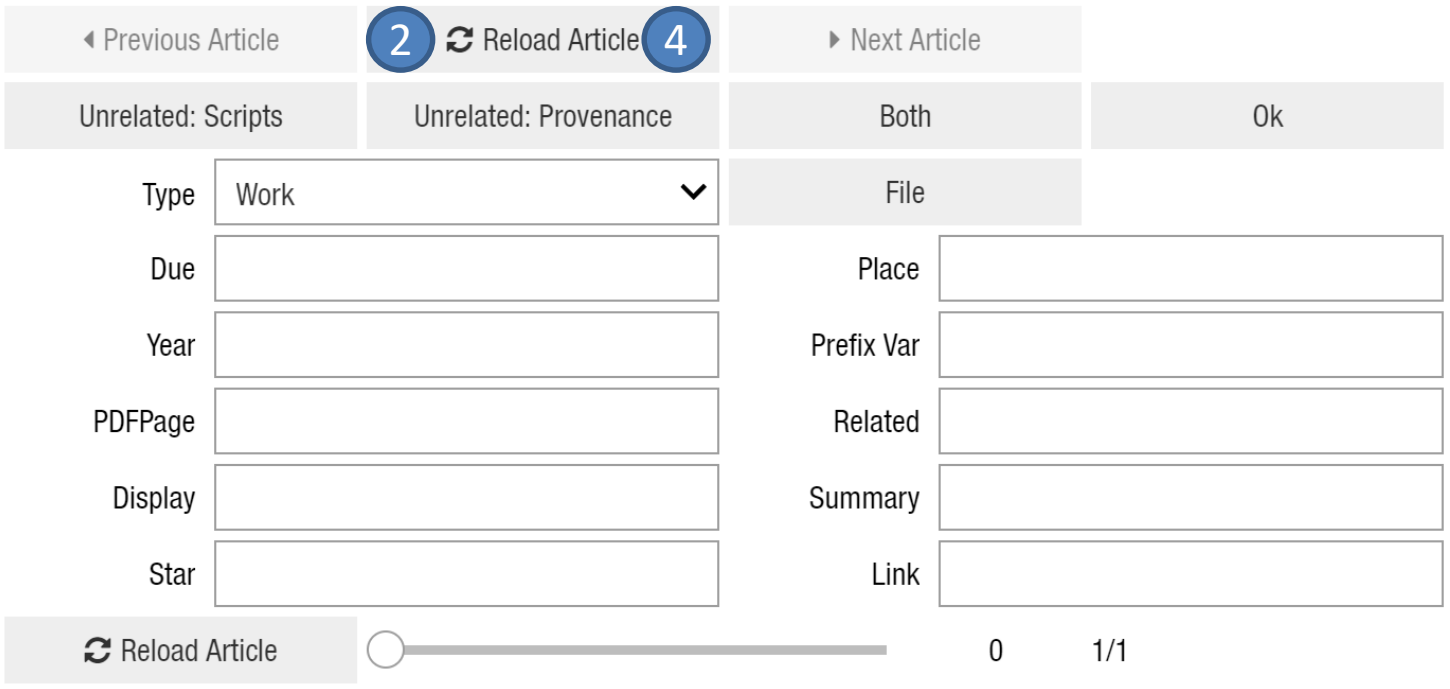
In [3]:
```
len(article_list)
```

Out[3]: 1

Insert.ipynb 2/4

```
In [4]: ArticleNavigator(articles=article_list)
```

1



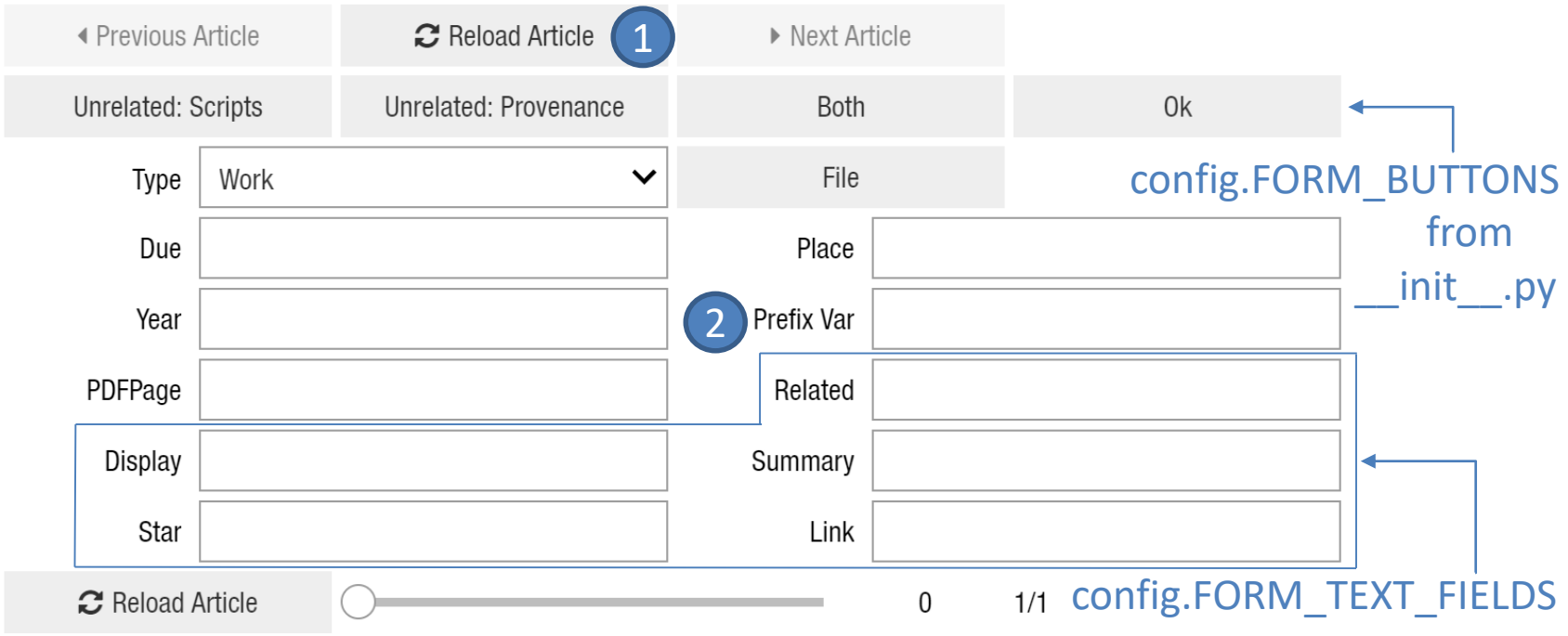
3 If this field shows up, the publication places was not detected
Edit places.py to add it or write the varname on the Place field

places.py Source Code

```
from snowballing.models import Place, DB
from snowballing.common_places import *

IPAW = conference("IPAW", "International
Provenance and Annotation Workshop")
CSUR = journal("CSUR", "ACM Computing Surveys")
VLDB = conference("VLDB", "VLDB Endowment")
arXiv = DB(Place("arXiv", "arXiv", "Archive"))
```

Insert.ipynb 3/4

```
In [4]: ArticleNavigator(articles=article_list)
```



noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts

pimentel2017a.pdf

Insert.ipynb 4/4

In [5]:

1

```
# Temp
insert('''
pimentel2017a = DB(WorkOk(
    2017, "noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts",
    display="pimentel",
    authors="Pimentel, João Felipe and Murta, Leonardo and Braganholo, Vanessa and Freire, Juliana",
    place=VLDB,
    entrytype="article",
    volume="10",
    number="12",
    ID="pimentel2017noworkflow",
))
''');
```

```
-Insert: pimentel2017a
```

In [4]:

```
ArticleNavigator(articles=article_list)
```

◂ Previous Article | ⟳ Reload Article | ▸ Next Article 2

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

SearchScholar.ipynb 1/3

In [1]:
```
import database
from snowballing.operations import load_work, reload, work_by_varname
from snowballing.selenium_scholar import SeleniumScholarQuerier
from snowballing.snowballing import SearchScholar
from snowballing.dbmanager import insert, set_attribute
```
1

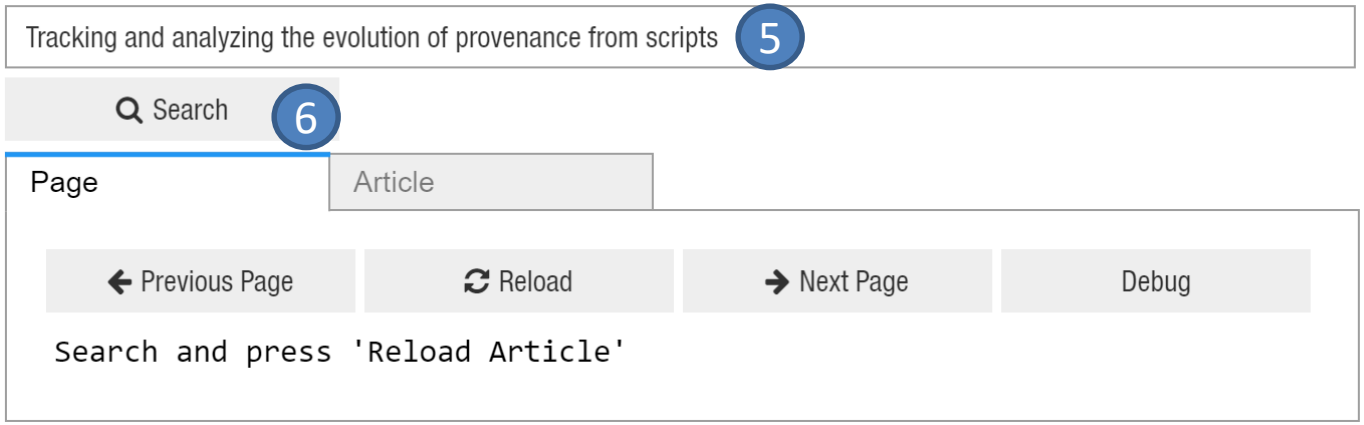
In [2]:
```
querier = SeleniumScholarQuerier()
querier.apply_settings(10, 4)
```
2

```
[ INFO]  settings applied
```

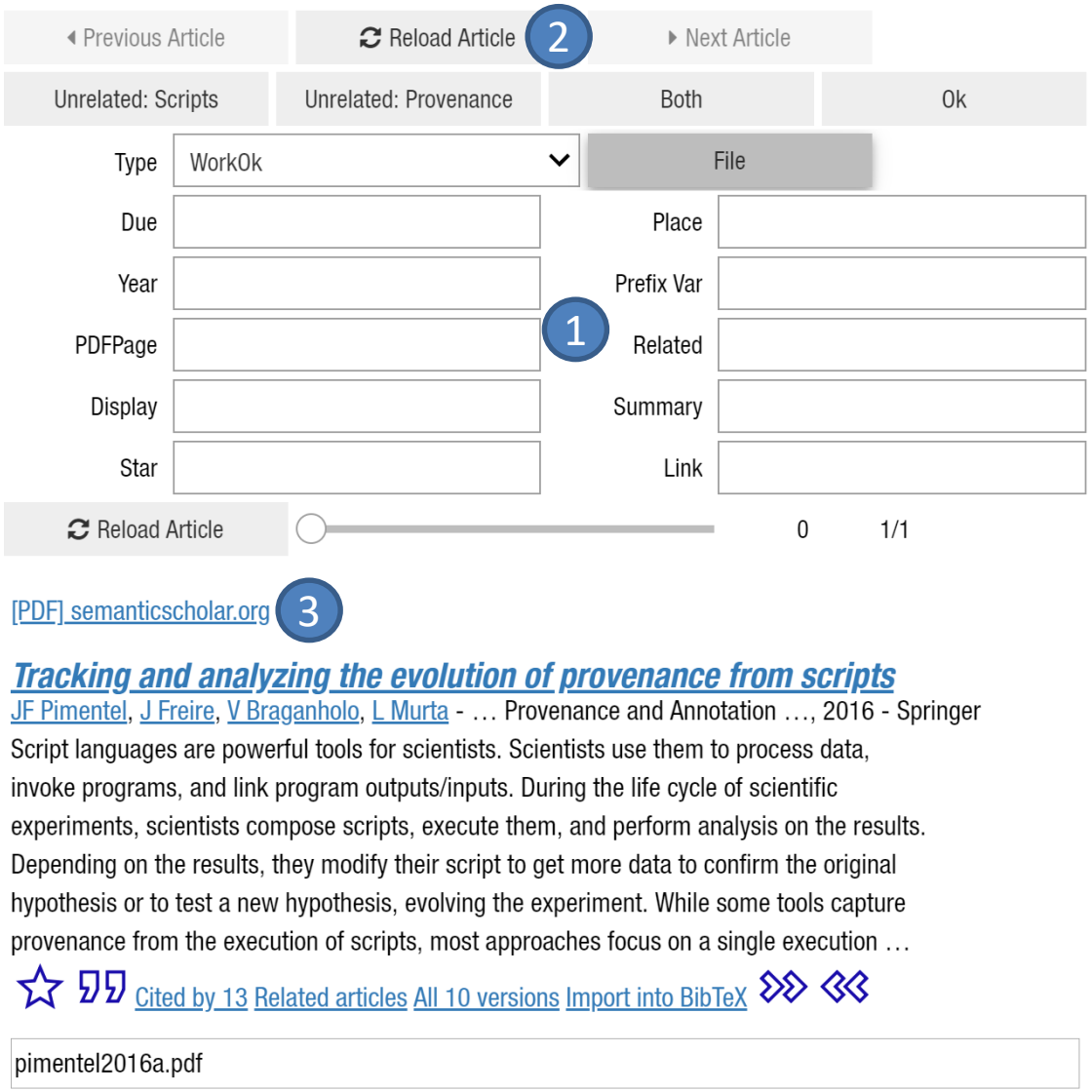
Out[2]: <snowballing.selenium_scholar.SeleniumScholarQuerier at 0x219d9ef2be0>

In [3]:
```
manager = SearchScholar(querier)
```
3

In [4]:
```
manager.browser()
```
4



SearchScholar.ipynb 2/3



SearchScholar.ipynb 3/3

In [5]:

```
# Temp
insert('''
pimentel2016a = DB(WorkOk(
    2016, "Tracking and analyzing the evolution of provenance from scripts",
    display="pimentel",
    authors="Pimentel, João Felipe and Freire, Juliana and Braganholo, Vanessa and Murta, Leonardo",
    place=IPAW,
    pp="16--28",
    entrytype="inproceedings",
    organization="Springer",
    ID="pimentel2016tracking",
    cluster_id="3783364081347190151",
    scholar="http://scholar.google.com/scholar?cites=3783364081347190151&as_sdt=2005&sciodt=0,5&hl=en"
    file="pimentel2016a.pdf",
))
''');
```

```
-Insert: pimentel2016a
```

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Backward.ipynb 1/4

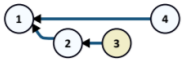


Figure 5: Experiment history with trials as nodes.

aforementioned visualization tool. Figure 5 presents the trial history for this demonstration. Note that trial 4 is based on trial 1 and trial 3 appears with a different color that denotes it is a backup trial. If the derivation history is not important, and the user just wants to list all trials with their command lines and durations, she can run `now list`.

Different from standard version control systems, noWorkflow versions are related to trial executions. This allows users to keep the full history of their experiments, keeping track of arguments, input data, output data, and other provenance information.

4. CONCLUSION

In this demonstration paper, we present *noWorkflow*, a tool that automatically collects provenance from Python scripts, without requiring any modification to the script. During the execution of scripts, noWorkflow collects imported modules, environment variables, function calls, file accesses, and, optionally, variables. While it does not collect network activity or database accesses directly, it collects the functions called for such accesses. noWorkflow also tracks the evolution of experiments and allows users to navigate over different versions. noWorkflow provides support for different kinds of provenance analyses through a command line interface, SQL and Prolog queries, and visualizations. Finally, noWorkflow also supports interactive analyses on Jupyter Notebooks.

noWorkflow is under active development. The system is available as open source software at `http://gems-uff.github.io/noworkflow`. Short videos showcasing the tool are available at `http://github.com/gems-uff/noworkflow/wiki/Videos`.

5. ACKNOWLEDGMENTS

We would like to thank CNPq, FAPERJ and the Moore-Sloan Data Science Environment for their financial support for this project. Juliana Freire is supported by the DARPA Memex and D3M programs, and NSF awards ACI- 1640864, CNS-1229185 and CNS-1405927.

6. REFERENCES

[1] C. Bochner, R. Gude, and A. Schreiber. A python library for provenance recording and querying. In *IPAW*, pages 229–240, 2008.

[2] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger, C. T. Silva, and H. T. Vo. Managing the Evolution of Dataflows with VisTrails. In *ICDE*, pages 71–71, 2006.

[3] F. Chirigati, D. Shasha, and J. Freire. Reprozip: Using provenance to support computational reproducibility. In *TaPP*, pages 977–980, 2013.

[4] S. Dar and R. Agrawal. Extending Sql with generalized transitive closure. *IEEE Transactions on Knowledge and Data Engineering*, 5(5):799–812, 1993.

[5] S. Dey, K. Belhajjame, D. Koop, M. Raul, and B. Ludäscher. Linking prospective and retrospective provenance in scripts. In *TaPP*, pages 1–7, 2015.

[6] J. Freire, D. Koop, E. Santos, and C. T. Silva. Provenance for computational tasks: A survey. *Computing in Science & Engineering*, 10(3):11–21, 2008.

[7] P. J. Guo and D. Engler. Using automatic persistent memoization to facilitate data analysis scripting. In *ISSTA*, pages 287–297, 2011.

[8] P. J. Guo and M. Seltzer. BUrrIto: Wrapping Your Lab Notebook in Computational Infrastructure. In *TaPP*, volume 12, pages 1–7, 2012.

[9] T. McPhillips, T. Song, T. Kolisnik, S. Aulenbach, K. Belhajjame, K. Bocinsky, Y. Cao, F. Chirigati, S. Dey, J. Freire, et al. YesWorkflow: a user-oriented, language-independent tool for recovering workflow information from scripts. *International Journal of Digital Curation*, 10(1):298–313, 2015.

[10] R. Meyer and K. Obermayer. pypet: A python Toolkit for Data Management of Parameter Explorations. *Frontiers in Neuroinformatics*, 10:1–16, 2016.

[11] K.-K. Muniswamy-Reddy, D. A. Holland, U. Braun, and M. I. Seltzer. Provenance-Aware Storage Systems. In *USENIX ATC*, pages 43–56, 2006.

[12] L. Murta, V. Braganholo, F. Chirigati, D. Koop, and J. Freire. noWorkflow: capturing and analyzing provenance of scripts. In *IPAW*, pages 71–83, 2014.

[13] J. F. Pimentel, S. Dey, T. McPhillips, K. Belhajjame, D. Koop, L. Murta, V. Braganholo, and B. Ludäscher. Yin & Yang: demonstrating complementary provenance from noWorkflow & YesWorkflow. In *IPAW*, pages 161–165, 2016.

[14] J. F. Pimentel, J. Freire, V. Braganholo, and L. Murta. Tracking and analyzing the evolution of provenance from scripts. In *IPAW*, pages 16–28, 2016.

[15] J. F. Pimentel, J. Freire, L. Murta, and V. Braganholo. Fine-grained provenance collection over scripts through program slicing. In *IPAW*, pages 199–203, 2016.

[16] J. F. N. Pimentel, V. Braganholo, L. Murta, and J. Freire. Collecting and analyzing provenance on interactive notebooks: when IPython meets noWorkflow. In *TaPP*, pages 1–6, 2015.

[17] C. E. Scheidegger, H. T. Vo, D. Koop, J. Freire, and C. T. Silva. Querying and re-using workflows with Vstrails. In *SIGMOD*, pages 1251–1254, 2008.

[18] M. Stamatogiannakis, P. Groth, and H. Bos. Looking inside the black-box: capturing data provenance using dynamic instrumentation. In *IPAW*, pages 155–167, 2014.

[19] M. Weske, G. Vossen, and C. B. Medeiros. *Scientific workflow management: WASA architecture and applications*. Citeseer, Universität Münster, Angewandte Mathematik und Informatik, 1996.

In [1]:

```
import database
from snowballing.operations import load_work, reload, work_by_varname
from snowballing.snowballing import Converter
from snowballing.snowballing import BackwardSnowballing
from snowballing.dbmanager import insert, set_attribute
```

In [2]:

```
Converter().browser()
```

1

2

3

4

Text | Set article_list variable

[1] C. Bochner, R. Gude, and A. Schreiber. A python library for provenance recording and querying. In IPAW, pages 229–240, 2008.
[2] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger,
C. T. Silva, and H. T. Vo. Managing the Evolution of Dataflows with VisTrails. In ICDE, pages 71–71, 2006.
[3] F. Chirigati, D. Shasha, and J. Freire. Reprozip: Using provenance to support computational reproducibility. In TaPP, pages 977–980, 2013.
[4] S. Dar and R. Agrawal. Extending Sql with generalized transitive closure. IEEE Transactions on Knowledge and Data Engineering, 5(5):799–812, 1993.
[5] S. Dey, K. Belhajjame, D. Koop, M. Raul, and B. Lud¨ascher. Linking prospective and retrospective provenance in scripts. In TaPP, pages 1–7, 2015.
[6] J. Freire, D. Koop, E. Santos, and C. T. Silva. Provenance for computational tasks: A survey. Computing in Science & Engineering, 10(3):11–21, 2008.
[7] P. J. Guo and D. Engler. Using automatic persistent memoization to facilitate data analysis scripting. In ISSTA, pages 287–297, 2011.
[8] P. J. Guo and M. Seltzer. BUrrIto: Wrapping Your Lab Notebook in Computational Infrastructure. In

[1] C. Bochner, R. Gude, and A. Schreiber. A pythonlibrary for provenance recording and querying. InIPAW, pages 229--240, 2008.[2] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger,C. T. Silva, and H. T. Vo. Managing the Evolution ofDataflows with VisTrails. In ICDE, pages 71--71, 2006.[3] F. Chirigati, D. Shasha, and J. Freire. Reprozip:Using provenance to support computationalreproducibility. In TaPP, pages 977--980, 2013.[4] S. Dar and R. Agrawal. Extending Sql withgeneralized transitive closure. IEEE Transactions onKnowledge and Data Engineering, 5(5):799--812, 1993.[5] S. Dey, K. Belhajjame, D. Koop, M. Raul, andB. Lud¨ascher. Linking prospective and retrospectiveprovenance in scripts. In TaPP, pages 1--7, 2015.[6] J. Freire, D. Koop, E. Santos, and C. T. Silva.Provenance for computational tasks: A survey.Computing in Science & Engineering, 10(3):11--21,2008.[7] P. J. Guo and D. Engler. Using automatic persistentmemoization to facilitate data analysis scripting. InISSTA, pages 287--297, 2011.[8] P. J. Guo and M. Seltzer. BUrrIto: Wrapping YourLab Notebook in Computational Infrastructure. InTaPP, volume 12, pages 1--7, 2012.[9] T. McPhillips, T. Song, T. Kolisnik, S. Aulenbach,K. Belhajjame, K. Bocinsky, Y. Cao, F. Chirigati,S. Dey, J. Freire, et al. YesWorkflow: a user-

Backward.ipynb 2/4

In [1]:
```
import database
from snowballing.operations import load_work, reload, work_by_varname
from snowballing.snowballing import Converter
from snowballing.snowballing import BackwardSnowballing
from snowballing.dbmanager import insert, set_attribute
```

In [2]:
```
Converter().browser()
```

[N] author name place other year

Set article_list variable 19

1

[1]
C. Bochner, R. Gude, and A. Schreiber.
A python library for provenance recording and querying.
IPAW
pp=229--240
2008

[2]
S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger,C. T. Silva, and H. T. Vo.
Managing the Evolution of Dataflows with VisTrails
ICDE
pp=71--71
2006

2


```
[
 {
  "citation_id": "[1]",
  "authors": "C. Bochner, R. Gude, and A. Schreiber.",
  "name": "A python library for provenance recording and querying.",
  "place1": "IPAW",
  "year": 2008,
  "_work_type": "Work",
  "pp": "229--240"
 },
 {
  "citation_id": "[2]",
  "authors": "S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger,C. T. Silva, and H. T. Vo."
```

Backward.ipynb 3/4

1

```
In [4]: BackwardSnowballing("pimentel2017a", articles=article_list)
```

2

Backward.ipynb 4/4

In [5]:

```
# Temp
insert('''
bochner2008a = DB(Work(
    2008, "A python library for provenance recording and querying.",
    display="bochner",
    authors="C. Bochner, R. Gude, and A. Schreiber.",
    place=IPAW,
    pp="229--240",
))

DB(Citation(
    pimentel2017a, bochner2008a, ref="[1]",
    contexts=[

    ],
))

''', citations='pimentel2017a');
```

```
-Insert: bochner2008a
-Insert Import: pimentel2017a
-Insert Import: bochner2008a
-Insert Citation: pimentel2017a -> bochner2008a
```

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Validate.ipynb 1/3

- Checks for place1
- Checks for Work type
- Checks for WorkUnrelated without “due=”
- Checks for names ending with “.”
- Verifies author format
- Checks for non-existente files
- Checks for WorkNoFile without “request”
- Updates papers with data from scholar
- Validates forward snowballing data
- Check for WorkSnowball without alerts
- Check for dash in the attributes “pp”, “volume”, “number”

Validate.ipynb 1/3

- Checks for place1
- Checks for Work type
- Checks for WorkUnrelated without “due=”
- Checks for names ending with “.”
- Verifies author format
- Checks for non-existente files
- Checks for WorkNoFile without “request”
- **Updates papers with data from scholar**
- Validates forward snowballing data
- Check for WorkSnowball without alerts
- Check for dash in the attributes “pp”, “volume”, “number”

Validate.ipynb 2/3

8 Scholar

In [15]:
```
querier = None
reload()
worklist = sorted(
    [k for k, w in load_work_map_all_years() if not getattr(w, "scholar_ok", False)],
    key=lambda x: (int(x[-5:-1]), x)
)
len(worklist)
```

Out[15]: 4

In [16]:
```
if worklist and querier is None:
    querier = SeleniumScholarQuerier()
    querier.apply_settings(10, 4)
```

```
[ INFO]  settings applied
```

In [17]:
```
from snowballing.snowballing import ScholarUpdate
supdate = ScholarUpdate(querier, worklist, force=False)
```

Validate.ipynb 3/3

Instituto de Computação

ESI Laboratório de Engenharia de Sistemas e Informação

In [18]:

```
supdate
```

1

Previous Work | Reload 2 | Next Work 4 | Debug | TextArea | 0

```
bochner2008a False
```

	bochner2008a	*Scholar*
name	A python library for provenance recording and querying	A python library for provenance recording and querying
authors	C. Bochner, R. Gude, and A. Schreiber	Bochner, Carsten and Gude, Roland and Schreiber, Andreas
entrytype	Conference	inproceedings
place1	International Provenance and Annotation Workshop (IPAW)	International Provenance and Annotation Workshop
year	2008	2008
organization		Springer
cluster_id		5386952245854088621
scholar		http://scholar.google.com/scholar?cites=5386952245854088621&as_sdt=2005&sciodt=0,5&hl=en

In [19]:

3

5

```
# Temp
set_attribute('bochner2008a', 'authors', 'Bochner, Carsten and Gude, Roland and Schreiber, Andreas')
set_attribute('bochner2008a', 'entrytype', 'inproceedings')
set_attribute('bochner2008a', 'organization', 'Springer')
set_attribute('bochner2008a', 'cluster_id', '5386952245854088621')
set_attribute('bochner2008a', 'scholar', 'http://scholar.google.com/scholar?cites=5386952245854088621&
set_attribute('bochner2008a', 'scholar_ok', True)
None
```

Validate.ipynb 3/3

In [18]:

```
supdate
```

1

Previous Work | Reload 2 | Next Work 4 | Debug | TextArea | 0

```
bochner2008a False
```

	bochner2008a	*Scholar*
name	A python library for provenance recording and querying	A python library for provenance recording and querying
authors	C. Bochner, R. Gude, and A. Schreiber	Bochner, Carsten and Gude, Roland and Schreiber, Andreas
entrytype	Conference	inproceedings
place1	International Provenance and Annotation Workshop (IPAW)	International Provenance and Annotation Workshop
year	2008	2008
organization		Springer
cluster_id		5386952245854088621
scholar		http://scholar.google.com/scholar?cites=5386952245854088621&as_sdt=2005&sciodt=0,5&hl=en

In [19]:

3

5

```
# Temp
set_attribute('bochner2008a', 'authors', 'Bochner, Carsten and Gude, Roland and Schreiber, Andreas')
set_attribute('bochner2008a', 'entrytype', 'inproceedings')
set_attribute('bochner2008a', 'organization', 'Springer')
set_attribute('bochner2008a', 'cluster_id', '5386952245854088621')
set_attribute('bochner2008a', 'scholar', 'http://scholar.google.com/scholar?cites=5386952245854088621&
set_attribute('bochner2008a', 'scholar_ok', True)
None
```

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Forward.ipynb 1/3

Instituto de Computação

ESI Laboratório de Engenharia de Sistemas e Informação

In [1]:
```
import database
from snowballing.operations import load_work, reload, work_by_varname
from snowballing.selenium_scholar import SeleniumScholarQuerier
from snowballing.snowballing import ForwardSnowballing
from snowballing.dbmanager import insert, set_attribute
```

1

In [2]:
```
querier = SeleniumScholarQuerier()
```

2

In [3]:
```
querier.apply_settings(20, 4)
```

3

```
[ INFO]  settings applied
```

Out[3]:
```
<snowballing.selenium_scholar.SeleniumScholarQuerier at 0x2166dae19b0>
```

4

In [5]:
```
manager = ForwardSnowballing(querier, "pimentel2017a", start=0)
```

5

In [6]:
```
manager
```

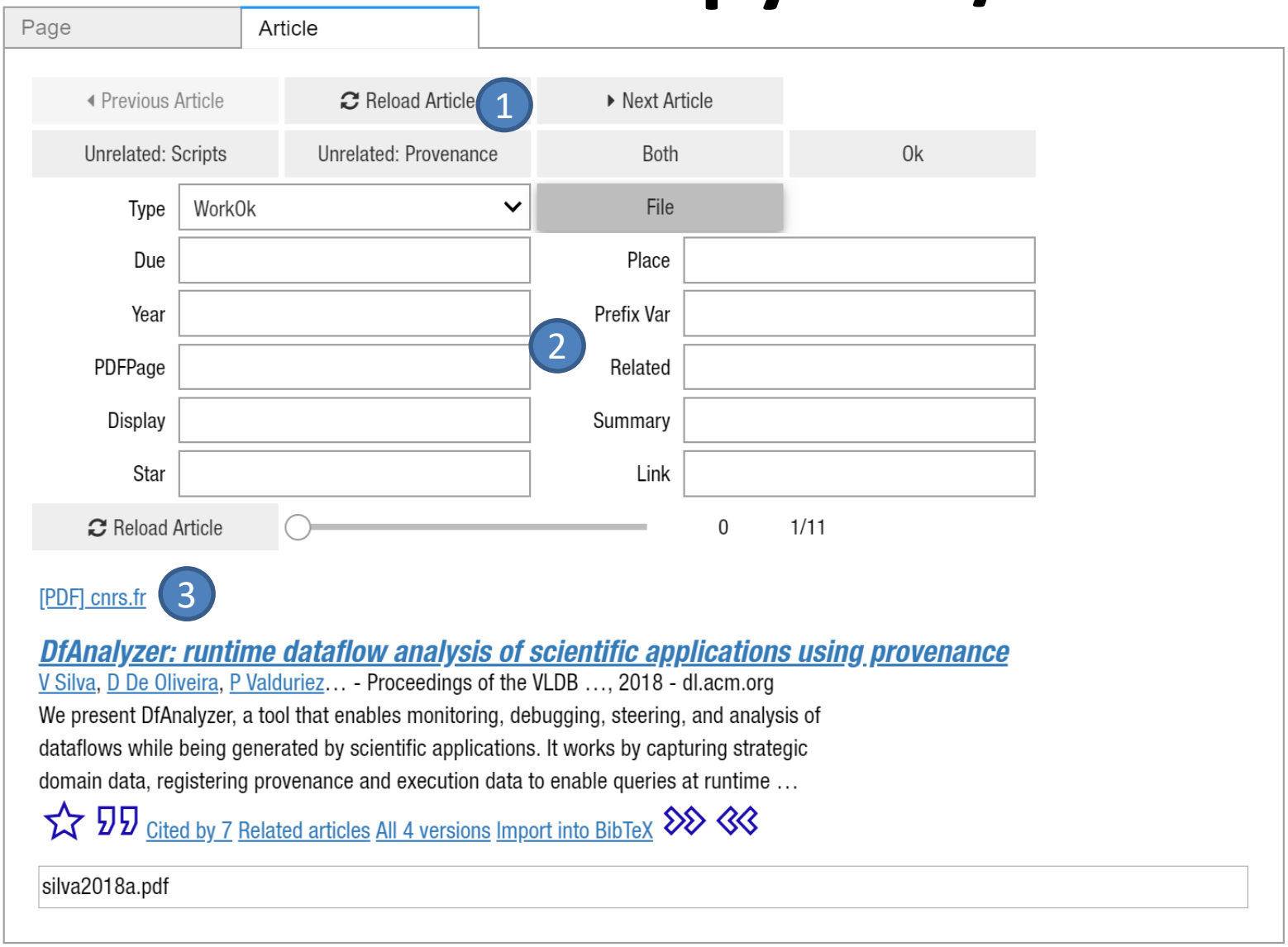
6

Page | Article 7

Previous Page | Reload | Next Page | Debug | 0

Click on 'Article' and press 'Reload Article'

Forward.ipynb 2/3



Forward.ipynb 3/3

In [7]:

1

```
# Temp
insert('''
silva2018a = DB(WorkOk(
    2018, "DfAnalyzer: runtime dataflow analysis of scientific applications using provenance",
    display="silva",
    authors="Silva, Vítor and De Oliveira, Daniel and Valduriez, Patrick and Mattoso, Marta",
    place=VLDB,
    pp="2082--2085",
    entrytype="article",
    volume="11",
    number="12",
    publisher="VLDB Endowment",
    ID="silva2018dfanalyzer",
    cluster_id="7347323264876490372",
    scholar="http://scholar.google.com/scholar?cites=7347323264876490372&as_sdt=2005&sciodt=0,5&hl=en"
    file="silva2018a.pdf",
))

DB(Citation(
    silva2018a, pimentel2017a, ref="",
    contexts=[

    ],
))

''', citations='pimentel2017a');
```

```
-Insert: silva2018a
-Insert Import: silva2018a
-Insert Citation: silva2018a -> pimentel2017a
```

Beware of the Google Scholar terms and conditions!

- The scholar does not allow to use bots
- The tool makes 21 requests to scholar to get the BibTex of papers for each resulting page
- These request may trigger the Scholar bot detection
- Pay attention to the Selenium window to fill the captachas
  - If a text field shows up asking for a captcha in the tool, **fill the captcha in the Selenium window** and type “<ok>” in the text field
- If it is possible, sign in with a google account that is not your main one in the Selenium window

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Project Structure

- database - Snowballing data storage
- files - Papers' PDF storage
- notebooks - Analysis Notebooks
- .gitignore - Ignores the files directory and Python cache files
- Backward.ipynb - Supports Backward Snowballing
- Forward.ipynb - Supports Forward Snowballing
- Index.ipynb - Describes the Project Structure
- Insert.ipynb - Supports the insertion of papers from BibTex
- Progress.ipynb - Monitors the Snowballing progress
- SearchScholar.ipynb - Supports the insertion of papers from Scholar
- Validate.ipynb - Validates and updates the insertions

Notebooks Structure

output - Notebooks outputs


ApproachesHTML.ipynb - Exports HTML from approaches


Bibtex.SearchWork.ipynb - Queries work and exports BibTex


CitationGraph.ipynb - Generates citation graph


Place.ipynb - Generates histogram with publication places


SnowballingProvenance.ipynb - Describes the snowballing process

Notebooks Structure

output - Notebooks outputs


ApproachesHTML.ipynb - Exports HTML from approaches


Bibtex.SearchWork.ipynb - Queries work and exports BibTex


CitationGraph.ipynb - Generates citation graph


Place.ipynb - Generates histogram with publication places


SnowballingProvenance.ipynb - Describes the snowballing process

Bibtex.SearchWork.ipynb 1/2

1 Search Work and Export BibTeX

In [1]:
```
import os, sys
sys.path.insert(1, os.path.join(sys.path[0], '..'))
import database
from snowballing.operations import load_work, work_to_bibtex, reload
from snowballing.models import DB
reload()
```

In [2]:
```
def find(text):
    words = text.split()
    for work in load_work():
        match = True
        for word in words:
            if not any(word.lower() in str(getattr(work, attr)).lower() for attr in dir(work) if not a
                match = False
                break
        if match:
            yield work_to_bibtex(work)
```

In [3]:
```
from ipywidgets import widgets, interactive

def result(text):
    if len(text) > 2:
        for work in find(text):
            print(work.replace("\n ", "\n  "))
interactive(result, text="")
```

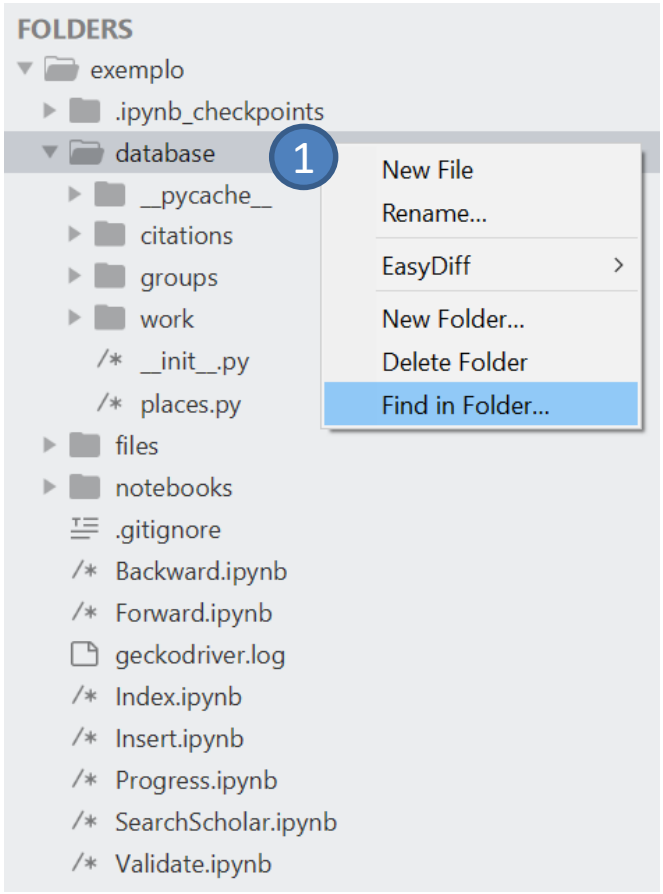
text

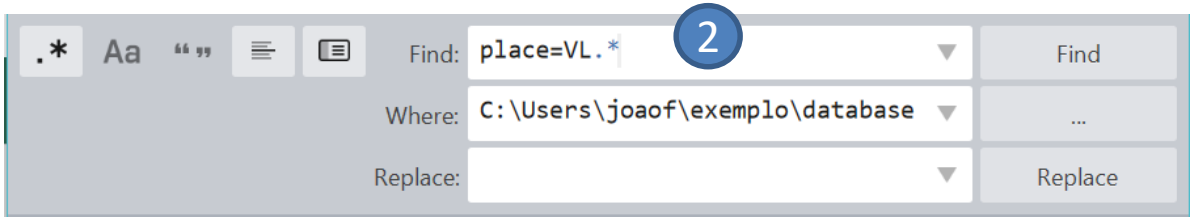
Bibtex.SearchWork.ipynb 2/2

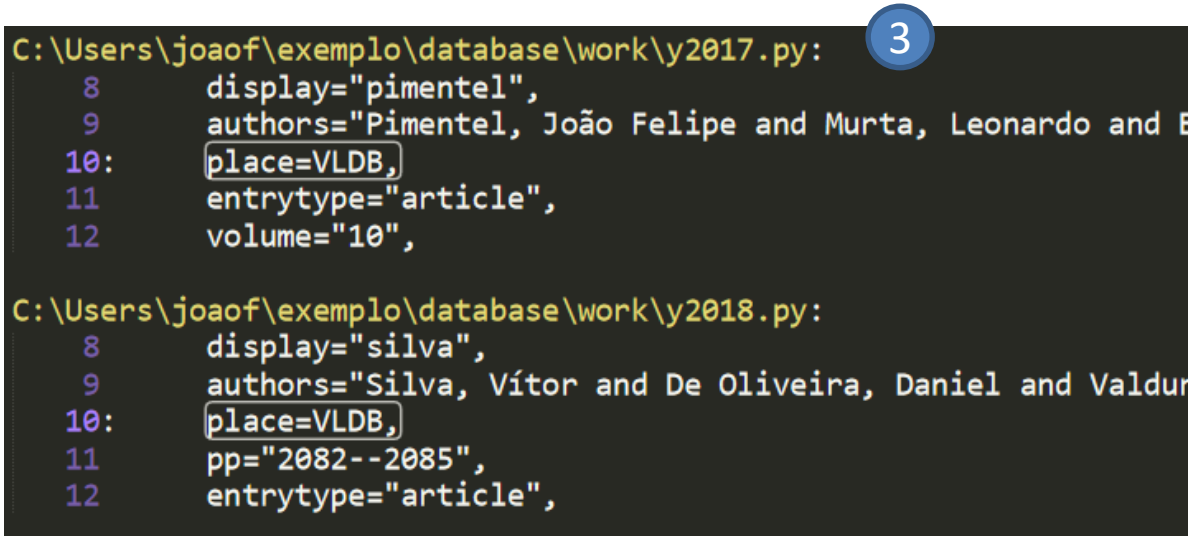
text vldb

```
@article{pimentel2017a,
   author = {Pimentel, Jo{\~a}o Felipe and Murta, Leonardo and Braganholo, Vanessa and Freire, Juli
ana},
   journal = {VLDB Endowment},
   number = {12},
   title = {no{W}orkflow: a tool for collecting, analyzing, and managing provenance from python scr
ipts},
   volume = {10},
   year = {2017}
}


@article{silva2018a,
   author = {Silva, V{\'\i}tor and De Oliveira, Daniel and Valduriez, Patrick and Mattoso, Marta},
   journal = {VLDB Endowment},
   number = {12},
   pages = {2082--2085},
   publisher = {VLDB Endowment},
   title = {{D}fAnalyzer: runtime dataflow analysis of scientific applications using provenance},
   volume = {11},
   year = {2018}
}
```

Busca Alternativa







Notebooks Structure

output - Notebooks outputs


ApproachesHTML.ipynb - Exports HTML from approaches


Bibtex.SearchWork.ipynb - Queries work and exports BibTex


CitationGraph.ipynb - Generates citation graph


Place.ipynb - Generates histogram with publication places


SnowballingProvenance.ipynb - Describes the snowballing process

CitationGraph.ipynb

```
In [2]: Graph("graph", delayed=False)
```

Filter In

Filter Out

Work

WorkSnowball

WorkOk

WorkUnrelated

WorkNoFile

WorkLang

Email

Work	#FFD86E	black	
WorkSnowball	#6DCE9E	white	
WorkOk	#68BDF6	white	

Places

References

R

Margin

M. Left

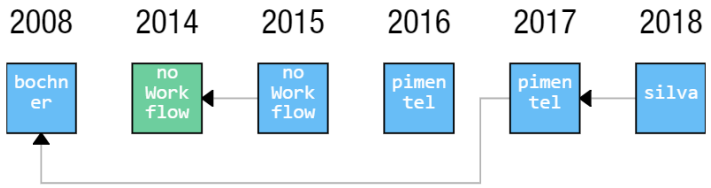
Dist. X

Dist. Y

Letters

By Year

'output\\graph.svg'



Notebooks Structure

output - Notebooks outputs


ApproachesHTML.ipynb - Exports HTML from approaches


Bibtex.SearchWork.ipynb - Queries work and exports BibTex

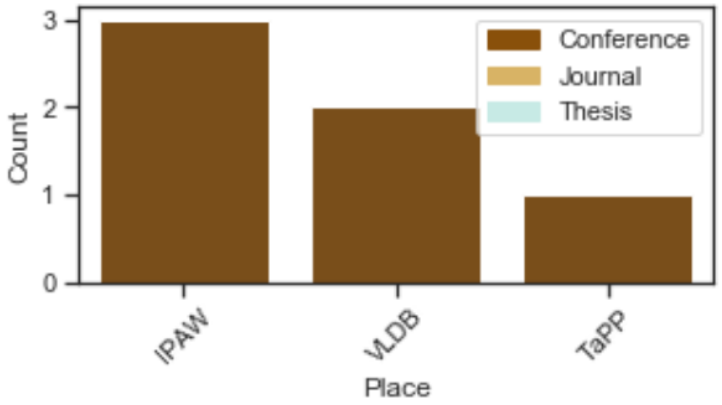

CitationGraph.ipynb - Generates citation graph


Place.ipynb - Generates histogram with publication places


SnowballingProvenance.ipynb - Describes the snowballing process

Place.ipynb

```
df = pd.DataFrame(rename, columns=["Place", "Quantity", "Type"])
ax = sns.barplot(x="Place", y="Quantity", data=df, palette=colors)
plt.xticks(rotation=45)
legend_patches = [
    matplotlib.patches.Patch(color=color_def[label], label=label)
    for label in TYPES
]
ax.set(ylabel='Count')
plt.legend(handles=legend_patches)
plt.gcf().subplots_adjust(left=0.05, right=1, top=0.95, bottom=0.40)
rcParams['figure.figsize'] = 13, 3

#ax.xaxis.labelpad = -15
plt.show()
plt.savefig("output/place.pdf")
```



Notebooks Structure

output - Notebooks outputs


ApproachesHTML.ipynb - Exports HTML from approaches


Bibtex.SearchWork.ipynb - Queries work and exports BibTex


CitationGraph.ipynb - Generates citation graph


Place.ipynb - Generates histogram with publication places


SnowballingProvenance.ipynb - Describes the snowballing process

SnowballingProvenance.ipynb

Instituto de Computação
ESI
Laboratório de Engenharia de Sistemas e Informação

In [1]:
```
import os, sys
sys.path.insert(1, os.path.join(sys.path[0], '..'))
import database
from snowballing.operations import reload, work_by_varname
from snowballing.strategies import Strategy
reload()
```

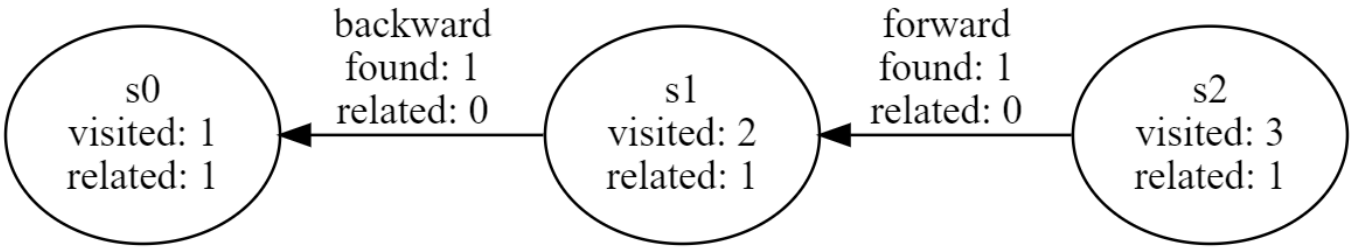
In [2]:
```
reload()
frontier = {work_by_varname(x) for x in (
    "murta2014a",
)}
filter_function = lambda x: x.category == "snowball"
strategy = Strategy(frontier, filter_function).bfbf()

len(strategy.visited)
```

Start set

Out[2]: 3

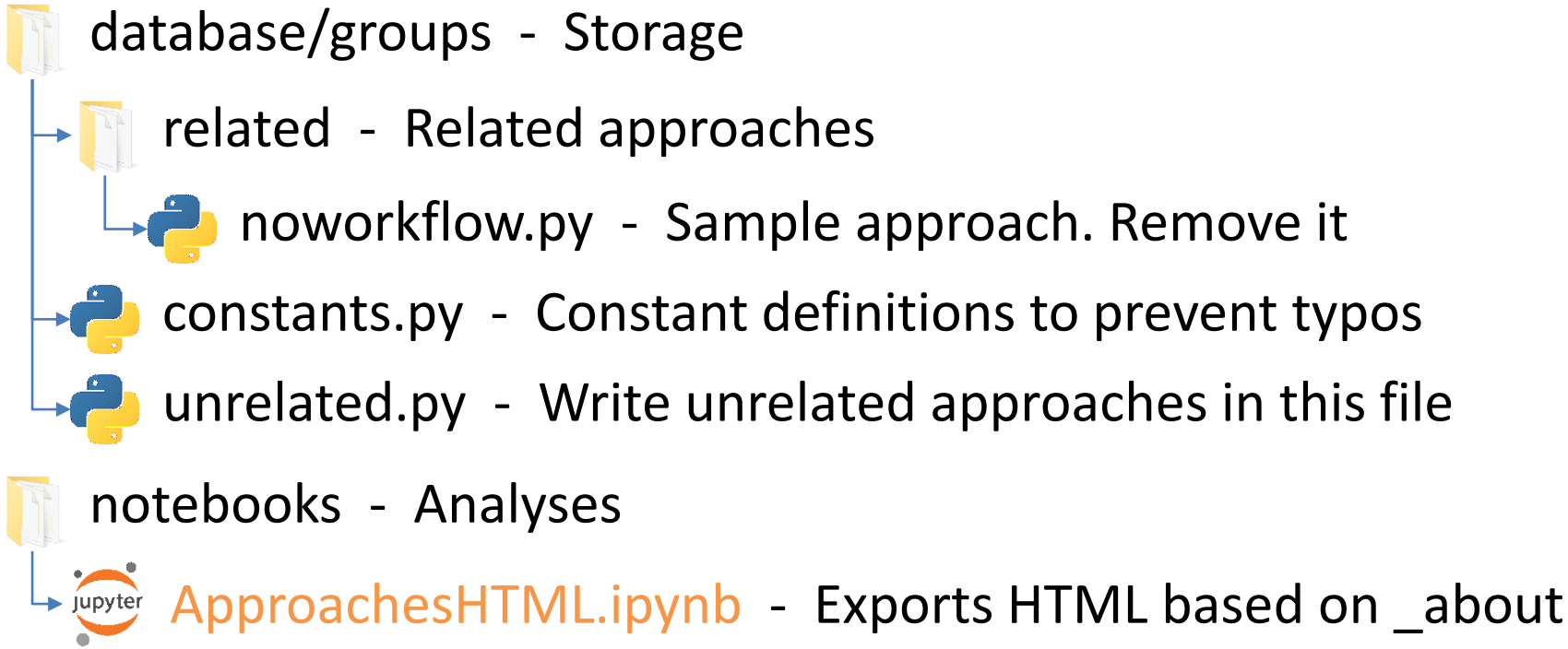
- .bb(): just BS
- .ff(): just FS
- .bbff(): all BS, followed by all FS
- .ffbb(): all FS , followed by all BS
- .s2bbff2u(): all BS and FS, in parallel
- **.bfbf(): alternates BS and FS**
- **.fbfb(): alternates FS and BS**
- **.sbfu(): BS and FS in parallel + union**

In [3]:
```
strategy
```



Approaches

- Groups papers to describe them together



noworkflow.py Source Code

```
approach = Group(
    murta2014a, pimentel2015a, pimentel2016a, pimentel2017a,
    display="no  Work  flow",
    approach_name="noWorkflow",
    _cite=False,
    _meta=[dict(
        target=PYTHON,
    )],
    _about="""
        <p>
            noWorkflow (<a href="#murta2014a" class="reference">murta2014a</a>; <a
href="#pimentel2015a" class="reference">pimentel2015a</a>) captures provenance
from Python scripts.
        </p>
    """,
)
```

noworkflow.py Source Code

```
approach = Group(
    murta2014a, pimentel2015a, pimentel2016a, pimentel2017a,
    display="no  Work  flow",
    approach_name="noWorkflow",
    _cite=False,
    _meta=[dict(
        target=PYTHON,
    )],
    _about="""
        <p>
            noWorkflow (<a href="#murta2014a" class="reference">murta2014a</a>; <a
href="#pimentel2015a" class="reference">pimentel2015a</a>) captures provenance
from Python scripts.
        </p>
    """,
)
```

Work list

Replaces display attribute

Name for the approach analyses

Add \cite{varnames} when it uses approach_name

noworkflow.py Source Code

```
approach = Group(
    murta2014a, pimentel2015a, pimentel2016a, pimentel2017a,
    display="no  Work  flow",
    approach_name="noWorkflow",
    _cite=False,
    _meta=[dict(
        target=PYTHON,
    )],
    _about="""
        <p>
            noWorkflow (<a href="#murta2014a" class="reference">murta2014a</a>; <a
href="#pimentel2015a" class="reference">pimentel2015a</a>) captures provenance
from Python scripts.
        </p>
    """,
)
```

Approach annotations

HTML to be exported by notebook

Library 1/2

from snowballing import

- approaches
  - **Approach analysis functions**
  - Group, GroupUnrelated, Item
  - get_approaches, name, wcite, wlatex_name, wcitea
- dbindex
  - **Functions that indicate where are each date in the "DB"**
  - citation_file, year_file, places_file, this_file, parse_varname
- dbmanager
  - **Insert, rename, remove papers and citations**
  - rename_citation, insert_citation, remove_source_citation, remove_target_citation, insert_work, rename_work, set_atribute, insert
- graph
  - **Configures and creates citation graph**
  - Graph, create_graph
- jupyter_utils
  - **Buttons and cells for Jupyter**
  - display_cell, idisplay, new_button, work_button
- models
  - **Data classes**
  - Place, Work, Site, Email, Citation, Database
- operations
  - **Querying and access operations**
  - **reload,** load_work, load_citations, load_work_map, **work_by_varname**, work_to_bibtex, find_citation, find
  - **metakey, metakey_title**

Library 2/2

from snowballing import

- scholar
  - **Original scholar.py library**
- selenium_scholar
  - **Reimplementation of parts of scholar.py to use it with Selenium**
  - SeleniumScholarQuerier, ScholarSettingsTask
  - get_scholar_url
- utils
  - **Extra functions**
  - text_y, multiline_wrap, Point
  - import_or_reload, import_submodules
  - consume, setitem, match_any
- snowballing
  - **Widgets for the snowballing phase**
  - Converter: Insert.ipynb, Backward.ipynb
  - ArticleNavigator: Todos
  - BackwardSnowballing: Backward.ipynb
  - ForwardSnowballing: Forward.ipynb
  - ScholarUpdate: Validate.ipynb
  - SearchScholar: SearchScholar.ipynb
- strategies
  - **Process analysis**
  - State, Strategy

Snowballing Tool

João Felipe Pimentel

Instituto de Computação

Laboratório de Engenharia de Sistemas e Informação