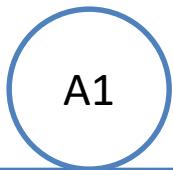


# Ferramenta de Snowballing

<https://github.com/Joaofelipe/snowballing>

# Snowballing

- Abordagem de busca sistemática para revisão da literatura
- Conjunto Inicial: {A1}



# Citações

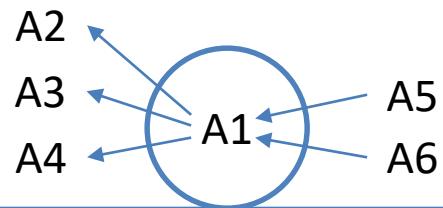
A1

## 1 Introduction

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

## References

- [1] A2, 2013.
- [2] A3, 2014.
- [3] A4, 2018.



A1 snowballing



About 1 result (0.14 sec)

**A1**

[E Arendelle...](#) - International Conference..., 2013 - Springer

This work exemplify the **snowballing** [1, 2, 3].

☆ 99 Cited by 2 Related articles

About 2 results (0.02 sec)

**A1**

Search within citing articles

**A5**

[A Stark-](#) arXiv preprint arXiv ..., 2019 - arxiv.org

A new snowballing example.

☆ 99 Cited by 1 Related articles 99

**A6**

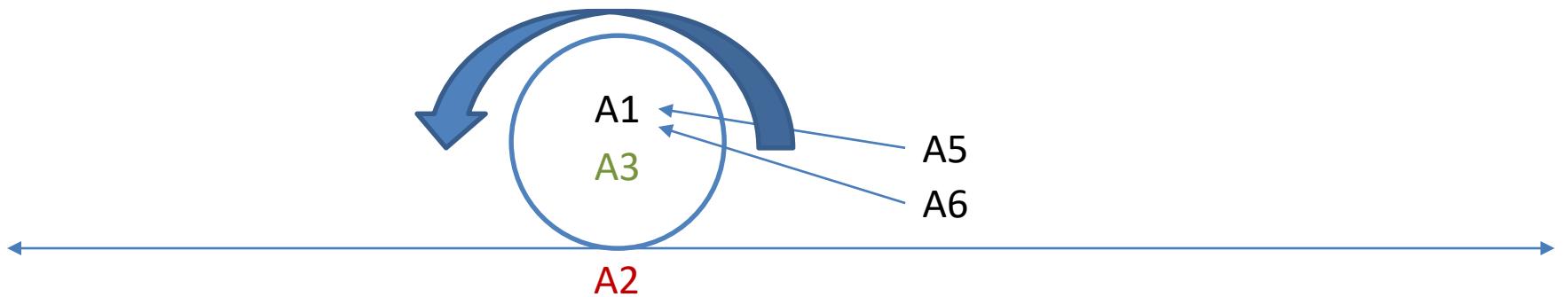
[L Best](#) - Journal..., 2018 - dl.acm.org

...

☆ 99 Related articles

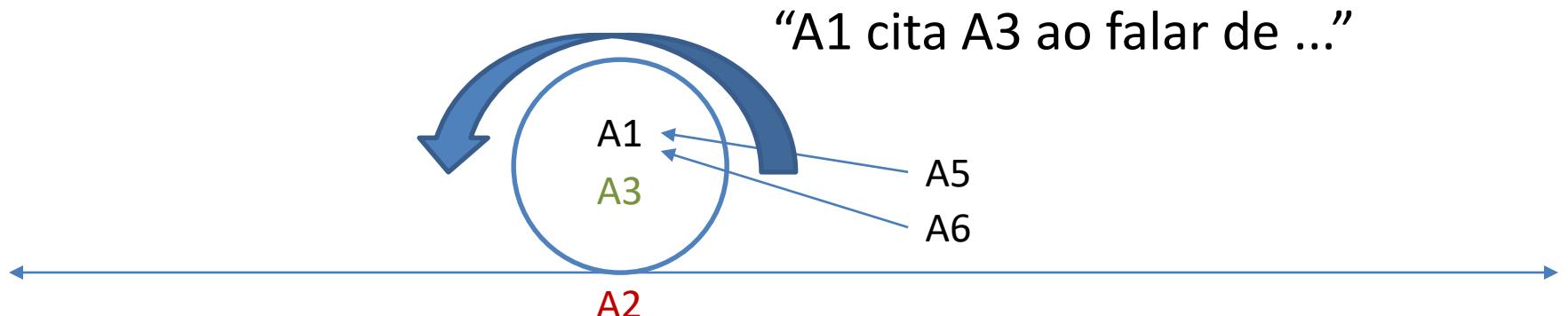
# Backward Snowballing

- Quem foi citado por A1 e é relacionado?



# Anotações

“A3 aborda o problema a partir...”

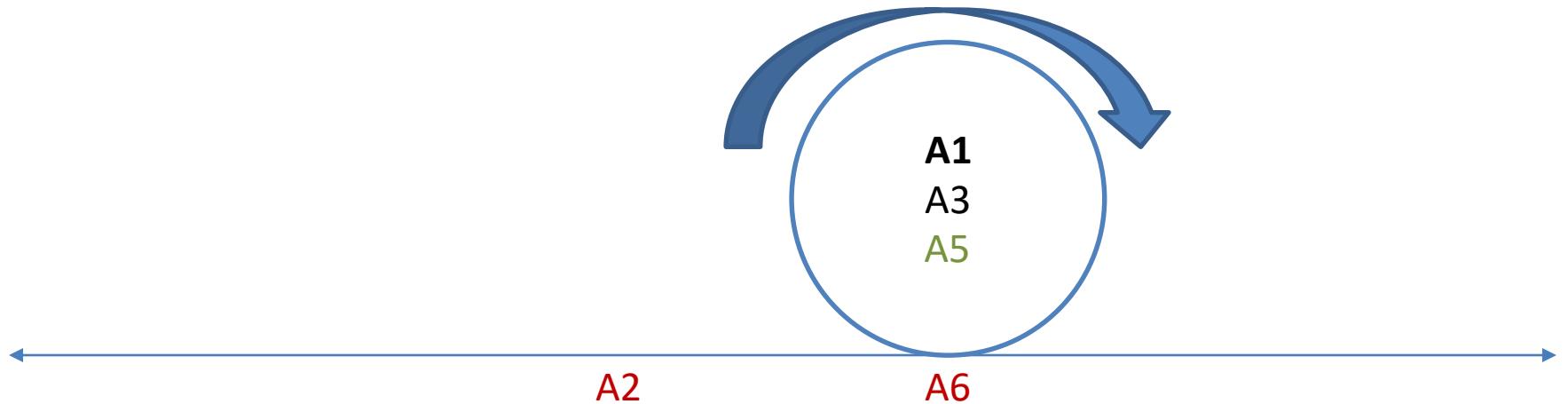


“A1 cita A3 ao falar de ...”

“A2 não trata do tema pois ...”

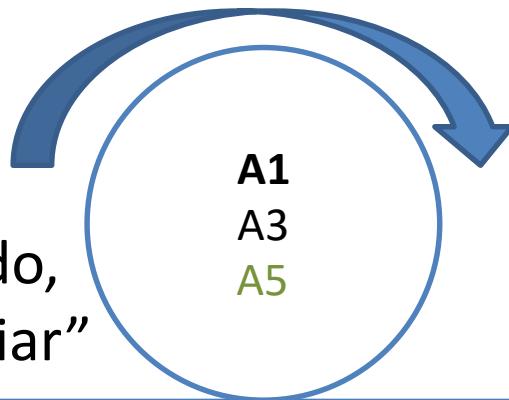
# Forward Snowballing

- Quem cita A1 e é relacionado?



# Outros tipos de Anotações

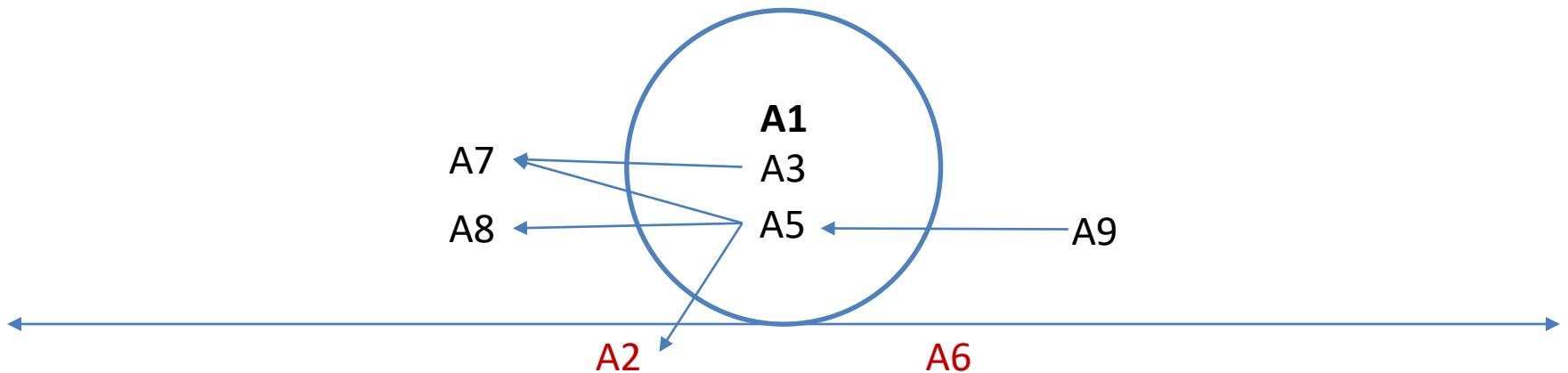
“Terminei A1 no dia 02/04/2019”



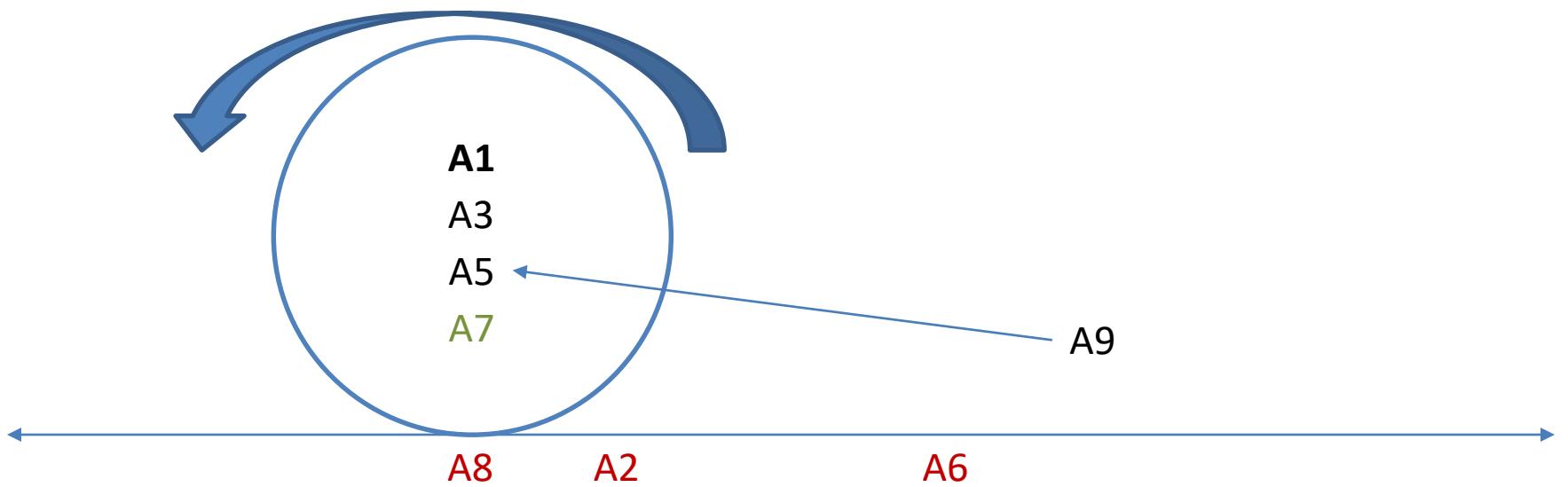
“A5 pode não ser tão relacionado,  
vou deixar marcado para reavaliar”

# Continuando o Processo

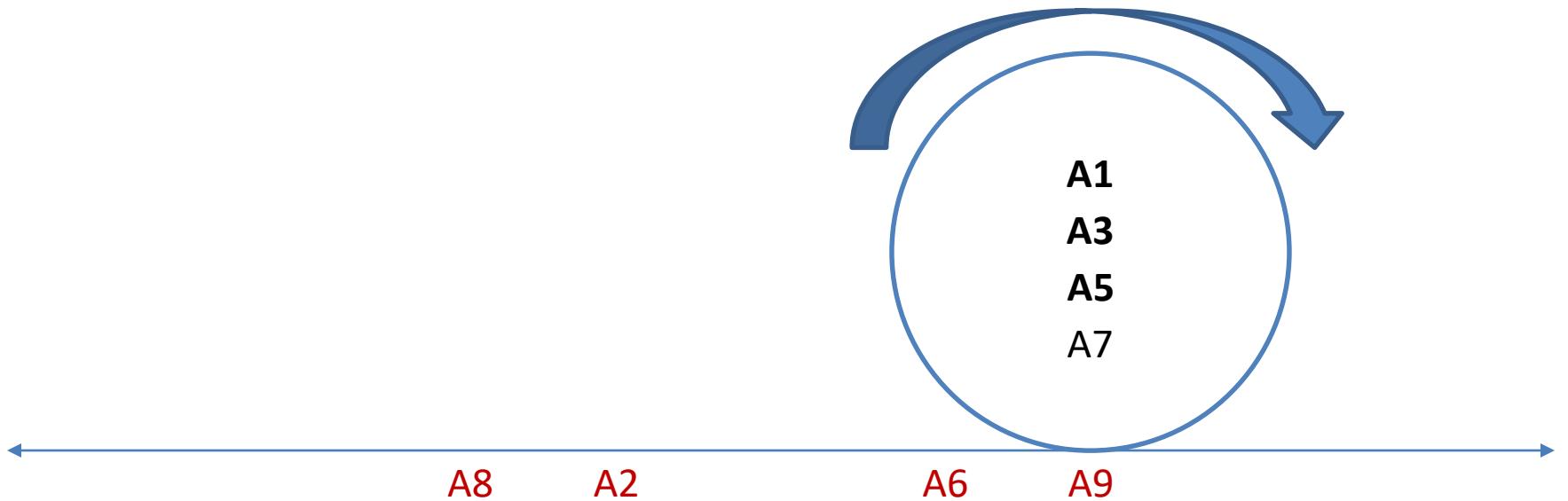
- A1 está completo, mas A3 e A5 possuem referências e citações



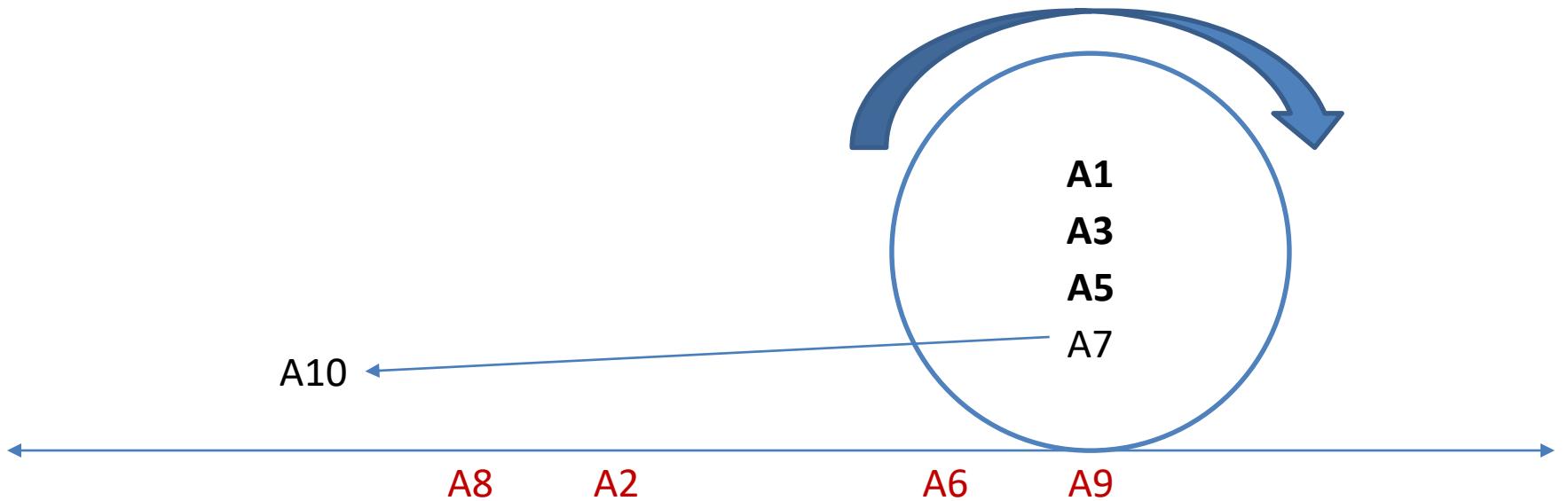
# Backward



# Forward

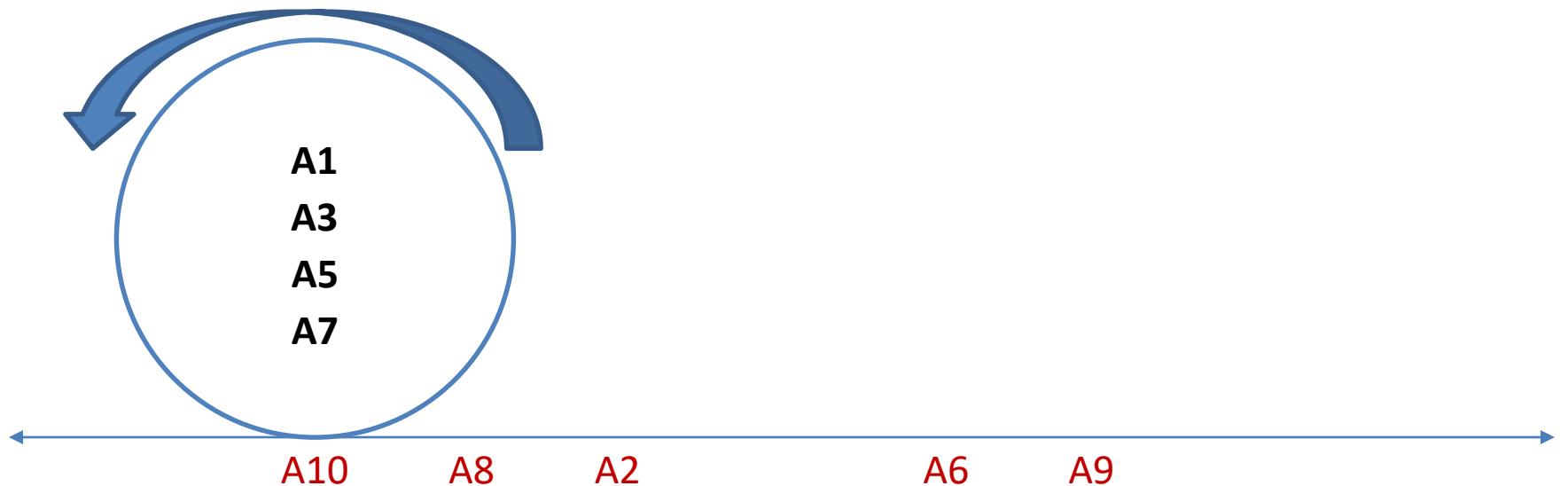


# Referências e Citações



# Parada

- Nenhuma referência nova nem citação nova



# Desafios

- Processo é bem trabalhoso e repetitivo
- Depois de centenas de artigos, fica difícil lembrar todos que já foram descartados
  - É bom ter um cadastro de todos os artigos visitados e não apenas dos relacionados
- Cadastro precisa ser fácil, consistente e extensível
- Busca e análise flexível para consultar o que foi cadastrado

# Ferramenta

- Conjunto de scripts e Jupyter Notebooks
  - Criados ao longo de um snowballing para automatizar e facilitar o processo
  - Alguns scripts são relacionados ao processo de **busca/snowballing** em si
  - Outros são relacionados à etapa de **análise**
- Ferramenta dividida em duas partes:
  - **Biblioteca** genérica
  - Área de trabalho do **projeto**

# Instalação e Uso

- Instale a biblioteca genérica
  - \$ pip install snowballing
- Instale geckodriver  
(<https://github.com/mozilla/geckodriver/releases>)
  - Não sei se é obrigatório, mas eu descompacto no diretório do Firefox: C:\Program Files\Mozilla Firefox
  - Adicione o diretório de instalação no PATH
- Instale graphviz e adicione no PATH
- Comece um novo projeto:
  - \$ snowballing start <nome>

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Estrutura do Projeto



**database** - Armazenamento dos dados de snowballing



**files** - Armazenamento de pdfs dos artigos



**notebooks** - Notebooks de análise



**.gitignore** - Ignora o diretório files e cache de Python



**Backward.ipynb** - Auxilia Backward Snowballing



**Forward.ipynb** - Auxilia Forward Snowballing



**Index.ipynb** - Descreve a estrutura do projeto



**Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex



**Progress.ipynb** - Monitora o progresso do Snowballing



**SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar



**Validate.ipynb** - Valida e atualiza o que foi inserido

# Estrutura de database

-  `__init__.py` - Configuração básica
-  `places.py` - Armazena locais de publicação
-  `groups` - Armazena abordagens. Ignorar para a etapa de busca
-  `work` - Armazena artigos
  -  `y2008.py` - Artigos de 2008. Remover
  -  `y2014.py` - Artigos de 2014. Remover
  -  `y2015.py` - Artigos de 2015. Remover
  -  `y9999.py` - Artigos sem ano. **Não Remover**
-  `citations` - Armazena citações. Um arquivo por artigo relacionado
-  `murta2014a.py` - Arquivo de exemplo. Remover

Arquivos  
de  
Exemplo

# Código em places.py

```
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"))
```

# Código em y2014.py

```
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,
))

))
```

# Código em y2014.py

```

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

variável      estado no snowball
murta2014a = DB(WorkSnowball)
2014, "noWorkflow: capturing and analyzing provenance of scripts",
display="noWorkflow", Nome na análise
authors="Murta, Leonardo and Braganholo, Vanessa and Chirigati, Fernando
and Koop, David and Freire, Juliana",
place=IPAW, Local de publicação. Referência à variável de places.py
))

    
```

config.CLASSES no `__init__.py`:

- Work: cadastrado, mas sem decisão
- WorkNoFile: arquivo não encontrado
- WorkLang: outra língua
- WorkUnrelated: não relacionado
- WorkOk: considerado, antes backward
- WorkSnowball: relacionado depois backward

# Código em y2014.py

```
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""",
))

```

# Código em y2014.py

```
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",
))

```

# Código em y2014.py

```
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,
))
```

# Código em y2014.py

```
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",
))

```

# Código em citations/\*.py

```
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,
))

))
```

# Código em citations/\*.py

```
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"  

    ],
))

```

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Insert.ipynb 1/4

In [1]:

```
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
```

In [2]:

```
2 Converter("bibtex")
```

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

4

5

```
[  

{  

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

scripts",  

  "authors": "Pimentel, João 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"  

}  

]
```

In [3]:

```
len(article_list)
```

out[3]: 1

# Insert.ipynb 2/4

In [4]:

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

1
◀ Previous Article
2 Reload Article
4
▶ Next Article

Unrelated: Scripts	Unrelated: Provenance	Both	Ok
Type	Work	File	
Due		Place	
Year		Prefix Var	
PDFPage		Related	
Display		Summary	
Star		Link	

0
1/1

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

pimentel2017a.pdf

Proceedings of the VLDB Endowment

- 3 Se este campo aparecer, o local de publicação não foi identificado
- Abra o places.py e adicione-o ou escreva uma variável existente em Place

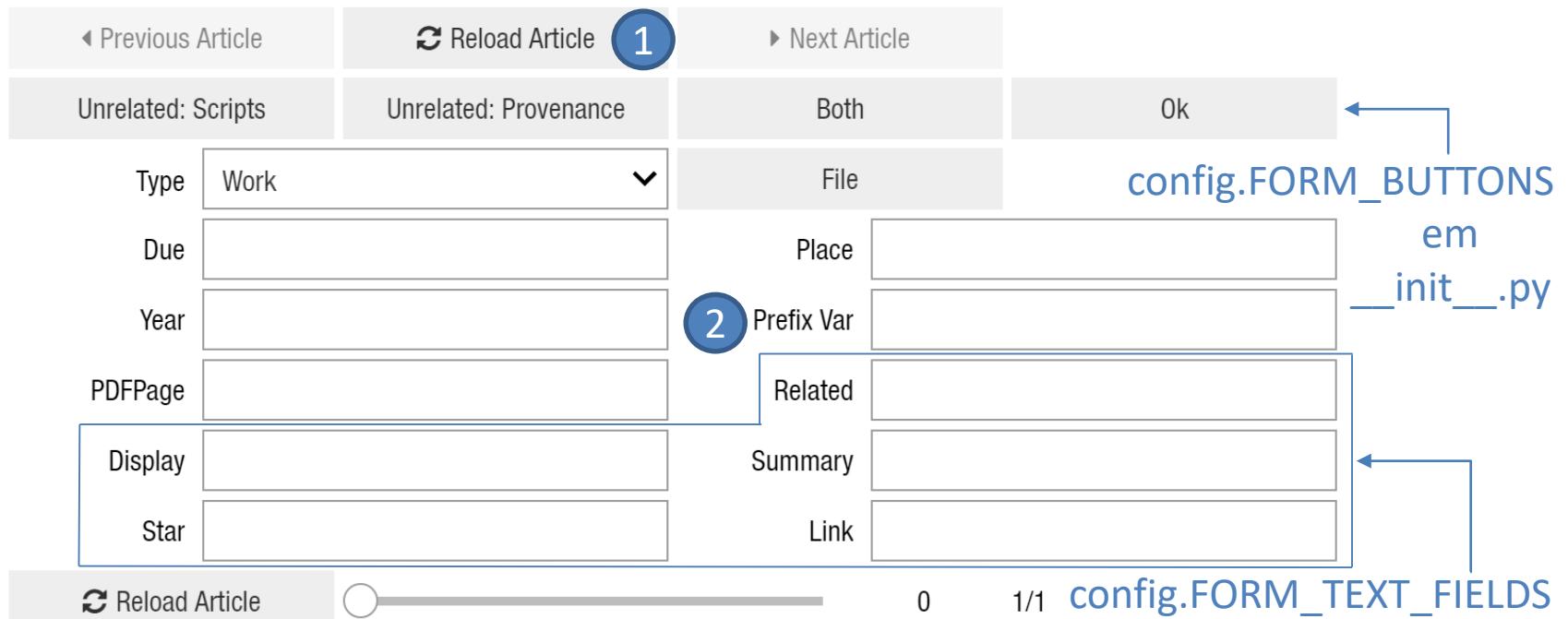
# Código em places.py

```
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)



◀ Previous Article      ⚡ Reload Article (1)      ▶ Next Article

Unrelated: Scripts      Unrelated: Provenance      Both      Ok

Type	Work	File	config.FORM_BUTTONS
Due		Place	em _init__.py
Year		Prefix Var	
PDFPage		Related	
Display		Summary	
Star		Link	

config.FORM\_TEXT\_FIELDS

⌚ Reload Article      0      1/1

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)
```

2

◀ Previous Article

⟳ Reload Article

▶ Next Article

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# SearchScholar.ipynb 1/3

In [1]:

```
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
```

In [2]:

```
2 querier = SeleniumScholarQuerier()
querier.apply_settings(10, 4)
```

[ INFO] settings applied

Out[2]: <snowballing.selenium\_scholar.SeleniumScholarQuerier at 0x219d9ef2be0>

In [3]:

```
3 manager = SearchScholar(querier)
```

In [4]:

```
4 manager.browser()
```

5 Tracking and analyzing the evolution of provenance from scripts

6

Page

Article

← Previous Page

⟳ Reload

→ Next Page

Debug

Search and press 'Reload Article'

# SearchScholar.ipynb 2/3

◀ Previous Article
⟳ Reload Article
2
▶ Next Article

Unrelated: Scripts
Unrelated: Provenance
Both
Ok

Type	WorkOk	▼	File
Due		Place	
Year		Prefix Var	
PDFPage		Related	
Display		Summary	
Star		Link	

⟳ Reload Article
0
1/1

[PDF] [semanticscholar.org](https://semanticscholar.org) 3

## *Tracking and analyzing the evolution of provenance from scripts*

[JF Pimentel](#), [J Freire](#), [V Braganholo](#), [L Murta](#) - ... Provenance and Annotation ..., 2016 - Springer

Script languages are powerful tools for scientists. Scientists use them to process data, invoke programs, and link program outputs/inputs. During the life cycle of scientific experiments, scientists compose scripts, execute them, and perform analysis on the results. Depending on the results, they modify their script to get more data to confirm the original hypothesis or to test a new hypothesis, evolving the experiment. While some tools capture provenance from the execution of scripts, most approaches focus on a single execution ...

☆ ↗ Cited by 13 Related articles All 10 versions Import into BibTeX ➤ ➥

pimentel2016a.pdf

# 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

# Estrutura do Projeto



**database** - Armazenamento dos dados de snowballing



**files** - Armazenamento de pdfs dos artigos



**notebooks** - Notebooks de análise



**.gitignore** - Ignora o diretório files e cache de Python



**Backward.ipynb** - Auxilia Backward Snowballing



**Forward.ipynb** - Auxilia Forward Snowballing



**Index.ipynb** - Descreve a estrutura do projeto



**Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex



**Progress.ipynb** - Monitora o progresso do Snowballing



**SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar



**Validate.ipynb** - Valida e atualiza o que foi inserido

# Backward.ipynb 1/4

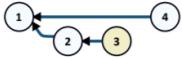


Figure 5: Experiment history with trials as nodes.

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 YourLab Notebook in Computational Infrastructure*. In *TaPP*, volume 12, pages 1–7, 2012.
- [9] T. McPhillips, T. Song, T. Kolisnik, S. Aulenbach, K. Belhajjame, T. 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. Braum, 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. Stamatogiannis, 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]:

```

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]:

```

2 Converter().browser()
  
```

3

4

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. 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 YourLab Notebook in Computational Infrastructure. In TaPP, volume 12, pages 1–7, 2012.
[9] T. McPhillips, T. Song, T. Kolisnik, S. Aulenbach, K. Belhajjame, T. Bocinsky, Y. Cao, F. Chirigati, S. Dey, J. Freire, et al. YesWorkflow: a user-oriented, language-independent tool for recovering workflow information from scripts. In TaPP, pages 1–7, 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. Braum, 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. Stamatogiannis, 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.
  
```

# 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

[1]

C. Bochner, R. Gude, and A. Schreiber.

A python library for provenance recording and querying.

IPAW

pp=229--240

2008

1

[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

Set article\_list variable

19

[  
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

In [4]: BackwardSnowballing("pimentel2017a", articles=article\_list)

◀ Previous Article
**3** Reload Article
**5** ▶ Next Article

Unrelated: Scripts
Unrelated: Provenance
Both
Ok

Type	Work	File
Due		Place
Year		Prefix Var
PDFPage		Related
Display		Summary
Star		Link

**2**
**4**

**1**
0
1/2

◀ Previous Article
Reload Article
▶ Next Article

A python library for provenance recording and querying.

bochner2008a.pdf

# 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

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Validate.ipynb 1/3

- Verifica place1
- Verifica tipo Work
- Verifica WorkUnrelated sem “due=”
- Verifica artigo com nome terminado em “.” ou sem nome
- Verifica formato dos autores
- Verifica inexistência de arquivos
- Verifica se WorkNoFile tem atributo “request”
- Atualiza dados de artigos com informações do scholar
- Verifica data de forward snowballing
- Verifica se WorkSnowball possui alerta
- Verifica dash nos atributos “pp”, “volume”, “number”

# Validate.ipynb 1/3

- Verifica place1
- Verifica tipo Work
- Verifica WorkUnrelated sem “due=”
- Verifica artigo com nome terminado em “.” ou sem nome
- Verifica formato dos autores
- Verifica inexistência de arquivos
- Verifica se WorkNoFile tem atributo “request”
- **Atualiza dados de artigos com informações do scholar**
- Verifica data de forward snowballing
- Verifica se WorkSnowball possui alerta
- Verifica dash nos atributos “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

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		<a href="http://scholar.google.com/scholar?cites=5386952245854088621&amp;as_sdt=2005&amp;sciodt=0,5&amp;hl=en">http://scholar.google.com/scholar?cites=5386952245854088621&amp;as_sdt=2005&amp;sciodt=0,5&amp;hl=en</a>

In [19]:

```
# 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&as_sdt=2005&sciodt=0,5&hl=en')
set_attribute('bochner2008a', 'scholar_ok', True)
None
```

3

5

# 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		<a href="http://scholar.google.com/scholar?cites=5386952245854088621&amp;as_sdt=2005&amp;sciodt=0,5&amp;hl=en">http://scholar.google.com/scholar?cites=5386952245854088621&amp;as_sdt=2005&amp;sciodt=0,5&amp;hl=en</a>

In [19]:

```
# 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
```

3

5

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Forward.ipynb 1/3

In [1]:

```
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
```

In [2]:

```
2 querier = SeleniumScholarQuerier()
```

In [3]:

```
3 querier.apply_settings(20, 4)
```

[ INFO] settings applied

Out[3]: <snowballing.selenium\_scholar.SeleniumScholarQuerier at 0x2166dae19b0>

In [5]:

```
4 manager = ForwardSnowballing(querier, "pimentel2017a", start=0)
```

In [6]:

```
5 manager
```

6 Page

Article

7

← Previous Page

⟳ Reload

→ Next Page

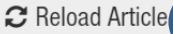
Debug

0

Click on 'Article' and press 'Reload Article'

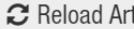
# Forward.ipynb 2/3

Page Article

◀ Previous Article  Reload Article **1** ▶ Next Article

Unrelated: Scripts Unrelated: Provenance Both Ok

Type	WorkOk	File
Due		Place
Year		Prefix Var
PDFPage		Related
Display		Summary
Star		Link

 Reload Article  0 1/11

[\[PDF\] cnrs.fr](#) **3**

**DfAnalyzer: runtime dataflow analysis of scientific applications using provenance**  
[V Silva](#), [D De Oliveira](#), [P Valdoriez](#)... - Proceedings of the VLDB ..., 2018 - dl.acm.org

We present DfAnalyzer, a tool that enables monitoring, debugging, steering, and analysis of dataflows while being generated by scientific applications. It works by capturing strategic domain data, registering provenance and execution data to enable queries at runtime ...

  Cited by 7 Related articles All 4 versions Import into BibTeX 

silva2018a.pdf

# 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
```

# Cuidado com os termos do Google Scholar!

- O scholar não permite o uso de bots
- A cada página de resultados, a ferramenta realiza 21 requisições ao scholar para obter o BibTex dos artigos
- Essas requisições podem fazer com que ele detecte como bot
- Preste atenção na janela do Selenium para preencher captchas
  - Se um campo de texto aparecer pedindo captcha na ferramenta, **preencha no Selenium** e digite “<ok>” no campo que aparecer
- Se possível, faça login na janela do Selenium com uma conta Google que não seja a sua principal

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
-  **files** - Armazenamento de pdfs dos artigos
-  **notebooks** - Notebooks de análise
-  **.gitignore** - Ignora o diretório files e cache de Python
-  **Backward.ipynb** - Auxilia Backward Snowballing
-  **Forward.ipynb** - Auxilia Forward Snowballing
-  **Index.ipynb** - Descreve a estrutura do projeto
-  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
-  **Progress.ipynb** - Monitora o progresso do Snowballing
-  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
-  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Estrutura do Projeto

-  **database** - Armazenamento dos dados de snowballing
  -  **files** - Armazenamento de pdfs dos artigos
  -  **notebooks** - Notebooks de análise
- 
-  **.gitignore** - Ignora o diretório files e cache de Python
  -  **Backward.ipynb** - Auxilia Backward Snowballing
  -  **Forward.ipynb** - Auxilia Forward Snowballing
  -  **Index.ipynb** - Descreve a estrutura do projeto
  -  **Insert.ipynb** - Auxilia inserção de artigos a partir de BibTex
  -  **Progress.ipynb** - Monitora o progresso do Snowballing
  -  **SearchScholar.ipynb** - Auxilia inserção de artigos a partir do Scholar
  -  **Validate.ipynb** - Valida e atualiza o que foi inserido

# Estrutura de notebooks



output - Resultados dos notebooks



[ApproachesHTML.ipynb](#) - Exporta HTML falando das abordagens



[Bibtex.SearchWork.ipynb](#) - Consulta artigos e exporta BibTex



[CitationGraph.ipynb](#) - Gera grafo de citações



[Place.ipynb](#) - Gera histograma do local de publicação



[SnowballingProvenance.ipynb](#) - Descreve processo de snowballing

# Estrutura de notebooks



output - Resultados dos notebooks



`ApproachesHTML.ipynb` - Exporta HTML falando das abordagens



`Bibtex.SearchWork.ipynb` - Consulta artigos e exporta BibTex



`CitationGraph.ipynb` - Gera grafo de citações



`Place.ipynb` - Gera histograma do local de publicação



`SnowballingProvenance.ipynb` - Descreve processo de snowballing

# 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

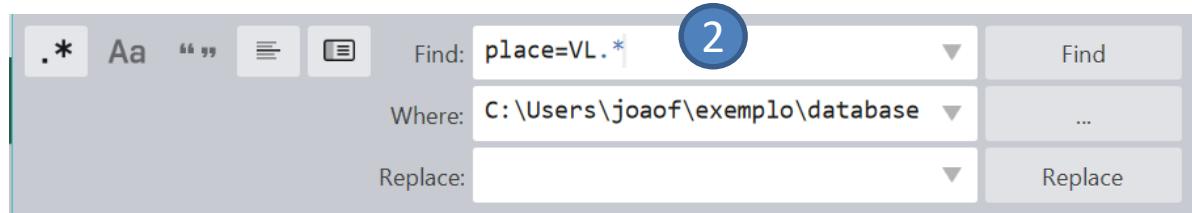
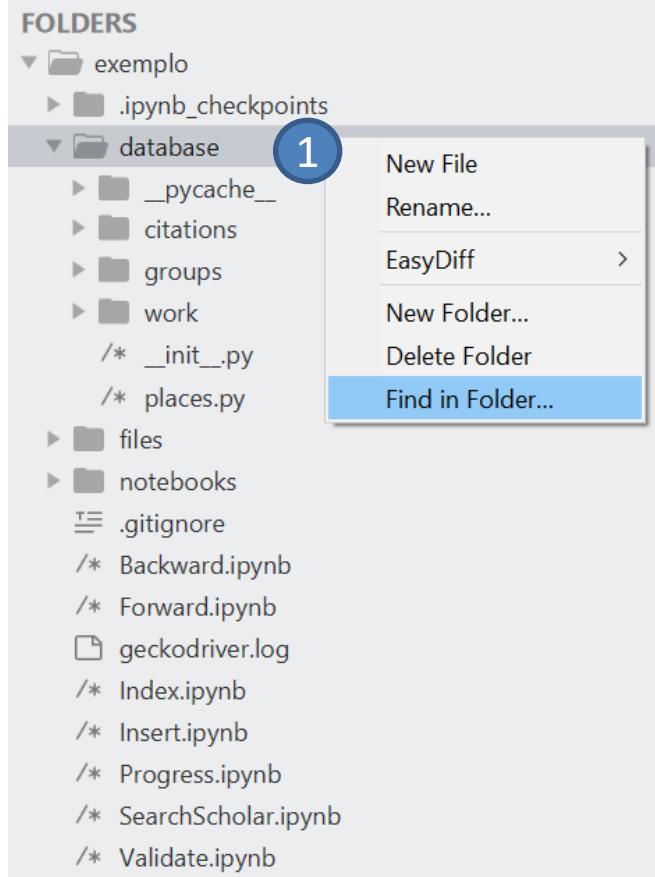
# Bibtex.SearchWork.ipynb 2/2

text vldb

```
@article{pimentel2017a,
    author = {Pimentel, João Felipe and Murta, Leonardo and Braganholo, Vanessa and Freire, Juliana},
    journal = {VLDB Endowment},
    number = {12},
    title = {noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts},
    volume = {10},
    year = {2017}
}
```

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

# Busca Alternativa



3

```
C:\Users\joaoef\exemplo\database\work\y2017.py:
 8     display="pimentel",
 9     authors="Pimentel, João Felipe and Murta, Leonardo and B
10: place=VLDB,
11     entrytype="article",
12     volume="10",

C:\Users\joaoef\exemplo\database\work\y2018.py:
 8     display="silva",
 9     authors="Silva, Vítor and De Oliveira, Daniel and Valduri
10: place=VLDB,
11     pp="2082--2085",
12     entrytype="article",
```

# Estrutura de notebooks



output - Resultados dos notebooks



[ApproachesHTML.ipynb](#) - Exporta HTML falando das abordagens



[Bibtex.SearchWork.ipynb](#) - Consulta artigos e exporta BibTex



[CitationGraph.ipynb](#) - Gera grafo de citações

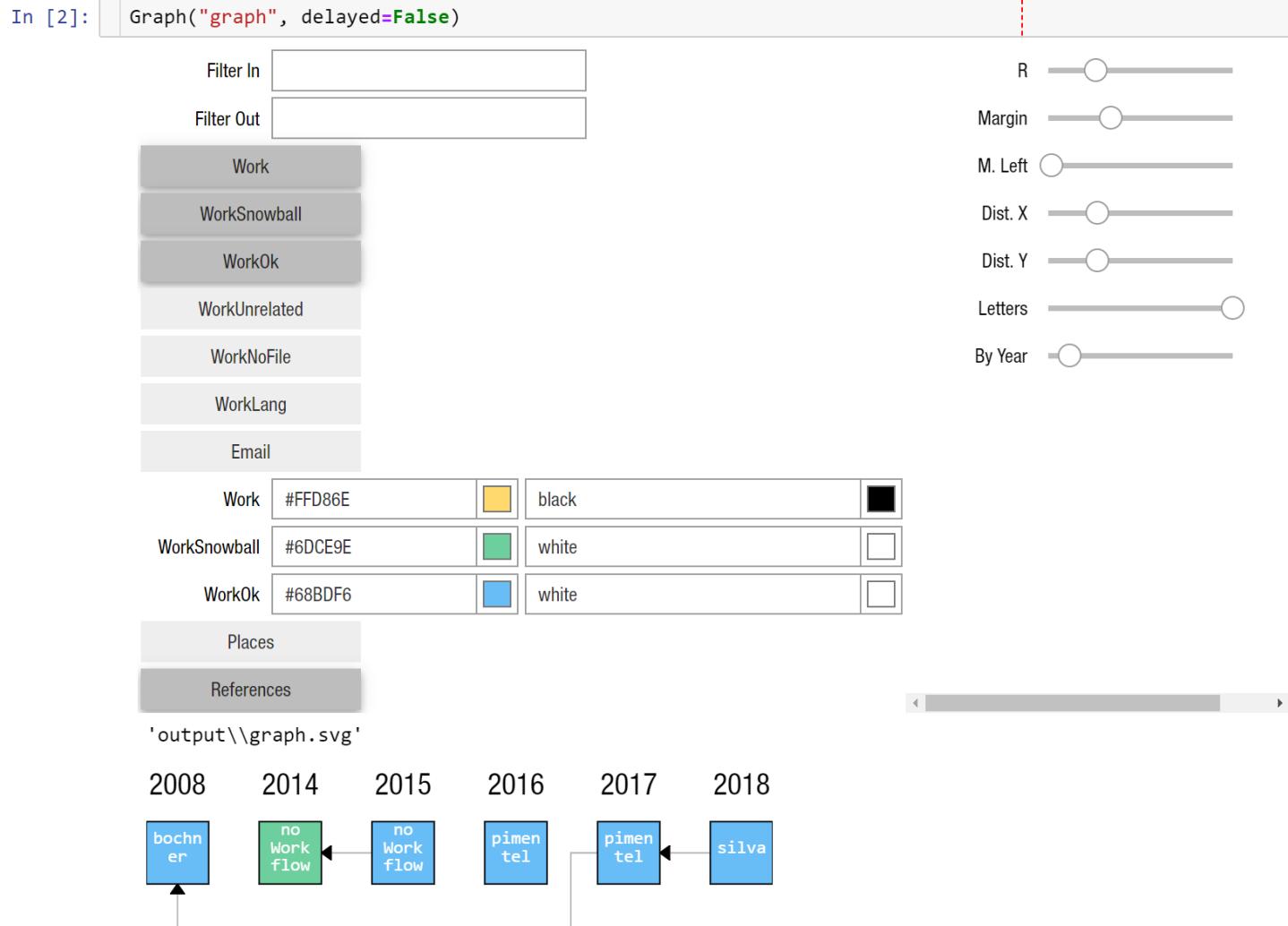


[Place.ipynb](#) - Gera histograma do local de publicação



[SnowballingProvenance.ipynb](#) - Descreve processo de snowballing

# CitationGraph.ipynb



# Estrutura de notebooks



output - Resultados dos notebooks



ApproachesHTML.ipynb - Exporta HTML falando das abordagens



Bibtex.SearchWork.ipynb - Consulta artigos e exporta BibTex



CitationGraph.ipynb - Gera grafo de citações



Place.ipynb - Gera histograma do local de publicação



SnowballingProvenance.ipynb - Descreve processo de snowballing

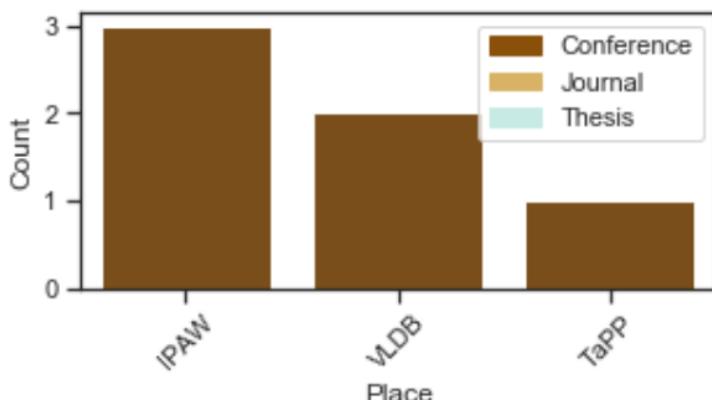
# Place.ipynb

```

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")

```



# Estrutura de notebooks



output - Resultados dos notebooks



ApproachesHTML.ipynb - Exporta HTML falando das abordagens



Bibtex.SearchWork.ipynb - Consulta artigos e exporta BibTex



CitationGraph.ipynb - Gera grafo de citações



Place.ipynb - Gera histograma do local de publicação



SnowballingProvenance.ipynb - Descreve processo de snowballing

# SnowballingProvenance.ipynb

```
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()
```

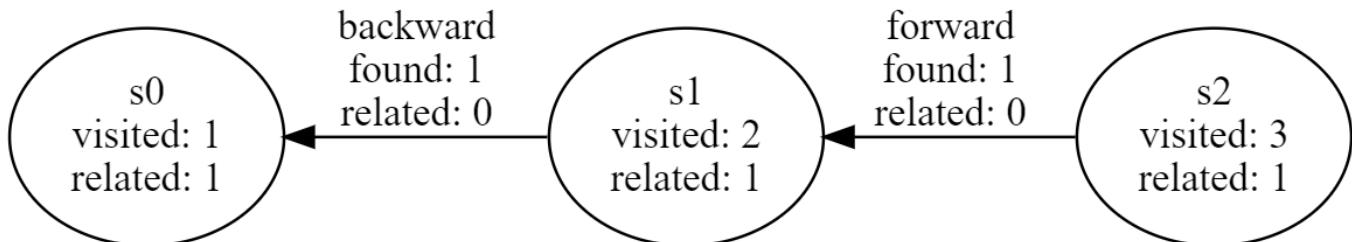
```
In [2]: reload()
frontier = {work_by_varname(x) for x in (
    "murta2014a", conjunto inicial
)}
filter_function = lambda x: x.category == "snowball"
strategy = Strategy(frontier, filter_function).bfbf()

len(strategy.visited)
```

Out[2]: 3

In [3]: strategy

- .bb(): apenas BS
- .ff(): apenas FS
- .bbff(): todo BS, seguido de todo FS
- .ffbb(): todo FS, seguido de todo BS
- .s2bbff2u(): todo BS e FS, em paralelo
- .bfbf(): alterna BS e FS**
- .fbfb(): alterna FS e BS**
- .sbfu(): BS e FS em paralelo, com união**



# Abordagens

- Agrupar artigos para descrever em conjuntos



database/groups - Armazenamento



related – Abordagens relacionadas



noworkflow.py - Abordagem de Exemplo. Remover



constants.py - Definição de constantes para evitar typos



unrelated.py - Local para colocar abordagens não relacionadas



notebooks - Análises



ApproachesHTML.ipynb - Exporta HTML baseado em \_about

# Código em noworkflow.py

```
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>  
    """,  
)
```

# Código em noworkflow.py

```
approach = Group(  
    murta2014a, pimentel2015a, pimentel2016a, pimentel2017a, Lista de artigos  
    display="no Work flow", Substitui display dos artigos  
    approach_name="noWorkflow", Nome que aparece na análise abordagens  
    _cite=False, Adicionar \cite{varnames} quando usar approach_name  
    _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>  
    """,  
)
```

# Código em noworkflow.py

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

HTML a ser exportado por notebook

# Biblioteca 1/2

from snowballing import

- approaches
  - **Análise de abordagens**
  - Group, GroupUnrelated, Item
  - get\_approaches, name, wcite, wlatex\_name, wcitea
- dbindex
  - **Funções que dizem onde está cada informação no “BD”**
  - citation\_file, year\_file, places\_file, this\_file, parse\_varname
- dbmanager
  - **Inserir, renomear e remover artigos e citações**
  - rename\_citation, insert\_citation, remove\_source\_citation, remove\_target\_citation, insert\_work, rename\_work, set\_attribute, insert
- graph
  - **Configuração e criação de grafo de citações**
  - Graph, create\_graph
- jupyter\_utils
  - **Botões e células para jupyter**
  - display\_cell, idisplay, new\_button, work\_button
- models
  - **Classes de dados**
  - Place, Work, Site, Email, Citation, Database
- operations
  - **Operações de busca e acesso**
  - **reload, load\_work, load\_citations, load\_work\_map, work\_by\_varname, work\_to\_bibtex, find\_citation, find**
  - **metakey, metakey\_title**

# Biblioteca 2/2

```
from snowballing import
```

- scholar
  - **Biblioteca scholar.py original**
- selenium\_scholar
  - **Reimplementação de parte da scholar.py para usar com selenium**
  - SeleniumScholarQuerier, ScholarSettingsTask
  - get\_scholar\_url
- utils
  - **Funções extras**
  - text\_y, multiline\_wrap, Point
  - import\_or\_reload, import\_submodules
  - consume, setitem, match\_any
- snowballing
  - **Widgets usados na etapa de snowballing**
  - Converter: Insert.ipynb, Backward.ipynb
  - ArticleNavigator: Todos
  - BackwardSnowballing: Backward.ipynb
  - ForwardSnowballing: Forward.ipynb
  - ScholarUpdate: Validate.ipynb
  - SearchScholar: SearchScholar.ipynb
- strategies
  - **Análise do processo**
  - State, Strategy

# Ferramenta de Snowballing

<https://github.com/Joaofelipe/snowballing>