Linking Design-Time and Run-Time:

A Graph-based Uniform Workflow Provenance Model

[1]**Xiaoyi Duan**, [1]Jia Zhang, [1]Qihao Bao, [2]Rahul Ramachandran,
[3]Tsengdar J. Lee, [4]Seungwon Lee, 4Lei Pan

1 Department of Electrical and Computer Engineering, Carnegie Mellon University, USA
2 NASA/MSFC, USA
3 Science Mission Directorate, NASA Headquarters, USA
4 Jet Propulsion Laboratory, California Institute of Technology, USA

Outline

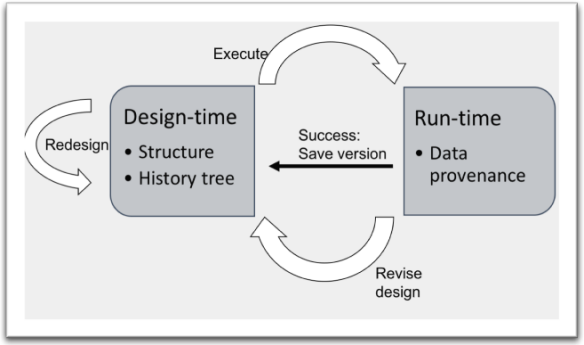
- Background and Motivation
- Related Work
- Our method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Outline

- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Background

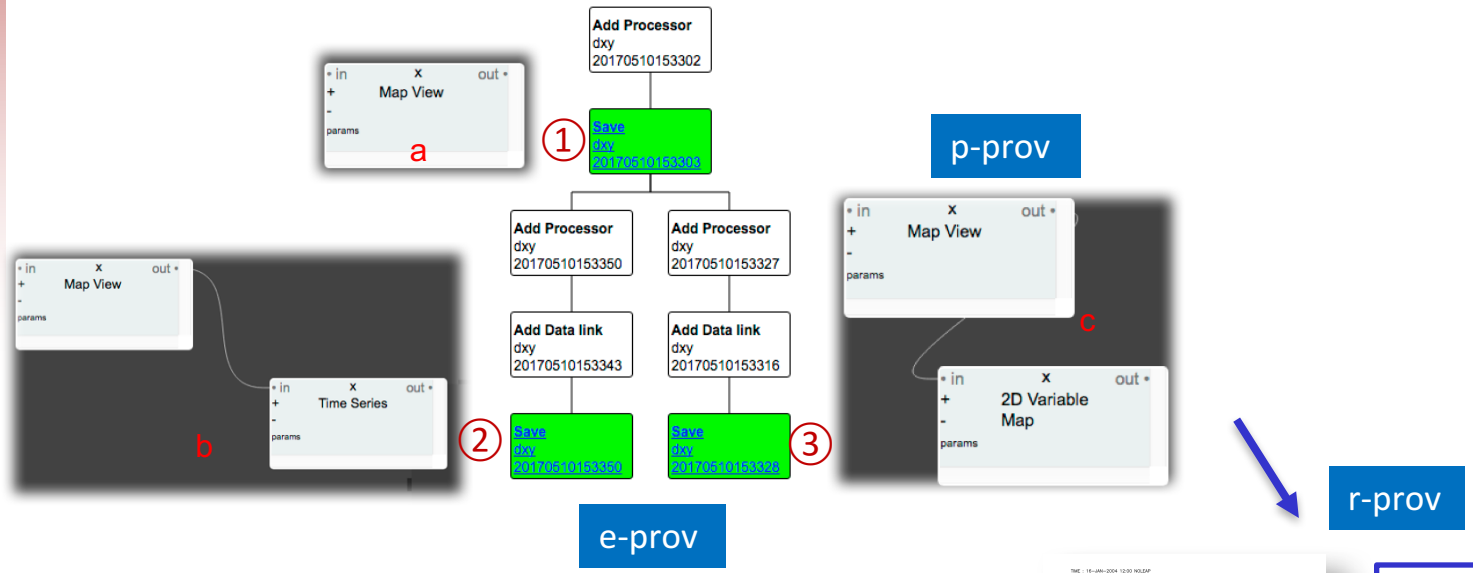
- **Scientific experiment design requires many trials and errors**
  - **Back and forth between design-time and run-time**
  - **Which information should be kept?**

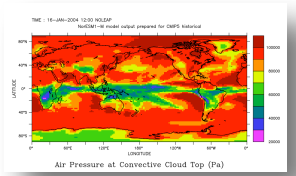


	r-prov (retrospective provenance)	p-prov (prospective provenance)	e-prov (evolution provenance)
Design-time	—	Workflow structure	Design history
Run-time	• Workflow execution • Data derivation	—	—

Motivation

- One integrated model handling all three types of provenance





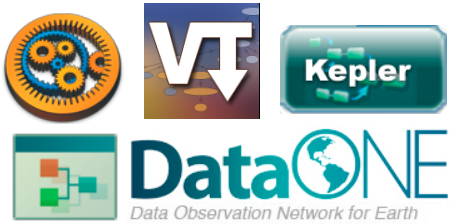
Sservice Execution Log:
http://hawking.sv.cmu.edu:8080/assets/html/twoDimMap.html?model=GFDL_ESM2G&var1=cct&pres1=-999999&purpose=GoodTest&lonS=0&lonE=360&latS=-90&latE=90&timeS=200401&timeE=200412&months=1,2,3,4,5,6,7,8,9,10,11,12&scale1=0&Image=http://hawking.sv.cmu.edu:8090/static/universalPlotting/mock.png&data_url=http%3A//hawking.sv.cmu.edu%3A8090/static/universalPlotting/mock.nc&userid=92

Outline

- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Related Work

- **Related Scientific Workflow Management Systems do not link different types of provenance**
- **Provenance models do not carry both run-time and design-time provenance**
  - Open Provenance Model
  - Provenance Data Model (PROV-DM)



	r-prov	p-prov	evolution	Linked
Kepler	✔			
Vistrails	✔	✔	✔	
Taverna	✔			
Trident	✔		✔	
DataOne	✔			
OPM	✔			
PROV-DM	✔			
Our Model	✔	✔	✔	✔

Outline

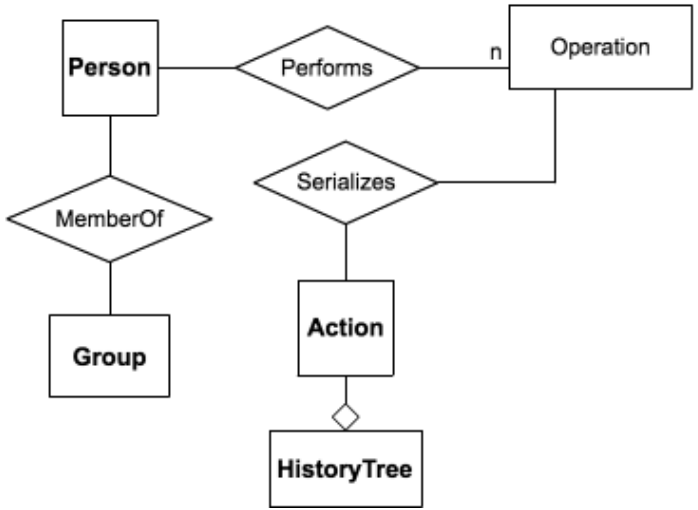
- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
    - Workflow-level Colored Petri Net
    - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Provenance Model

- **Design-time Provenance**
  - Action-Oriented Provenance (AOP)
  - Structure-Oriented Provenance (SOP)
- **Run-time Provenance**
  - Retrospective Provenance
- **Portals Linking Design-time and Run-time Worlds**

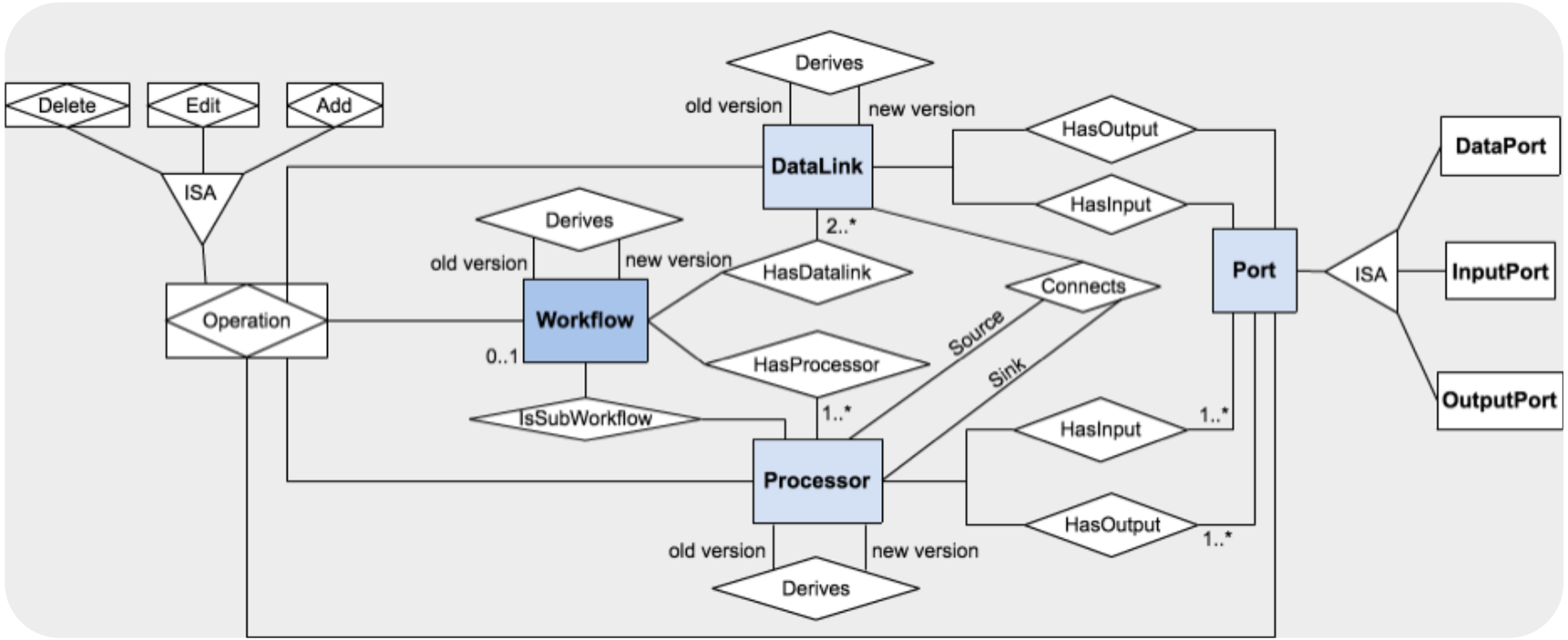
Action-Oriented Provenance (AOP)

- **When a researcher implements an operation on a workflow, the operation is serialized to an action node in the AOP.**



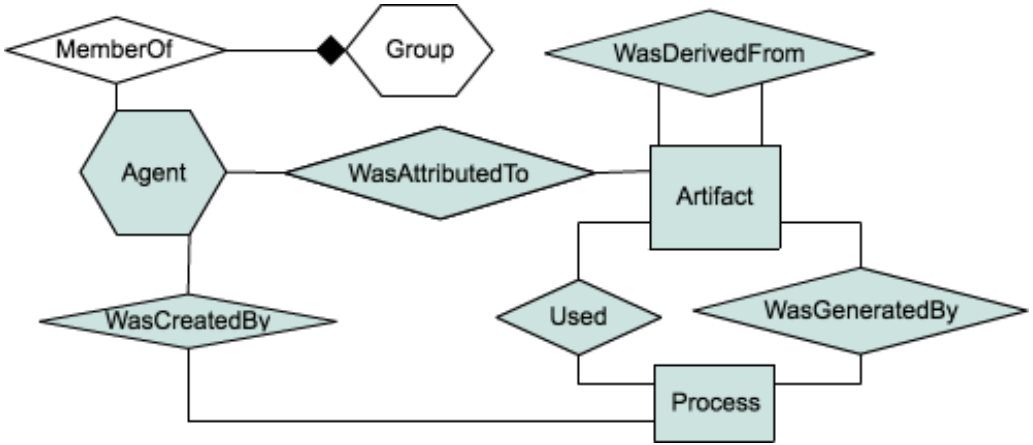
Structure-Oriented Provenance (SOP)

- **Capture the representation of workflow structure**

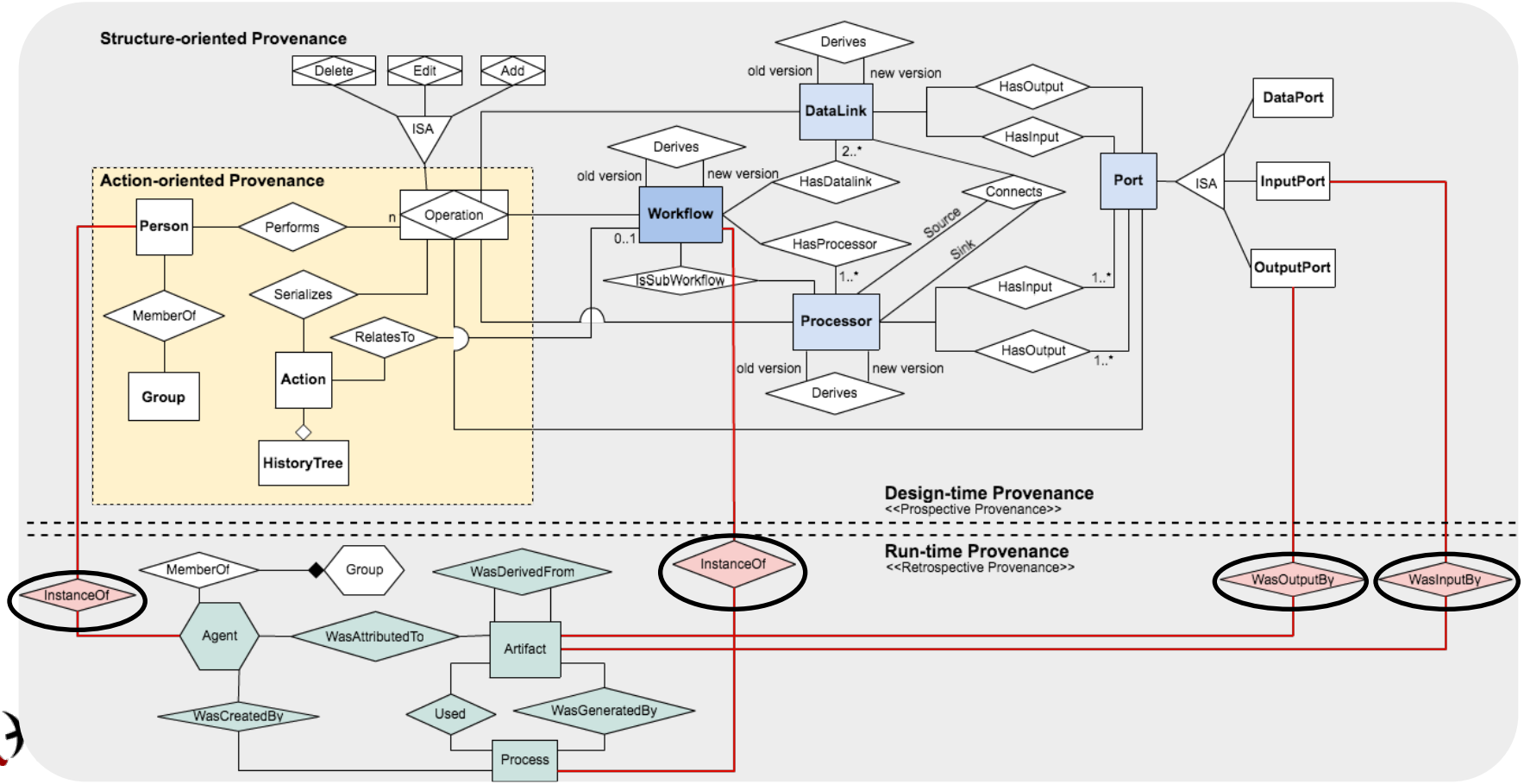


Retrospective Provenance

- During workflow execution time, provenance captures past workflow execution and data derivation information.



Portals Linking Design-time & Run-time Worlds



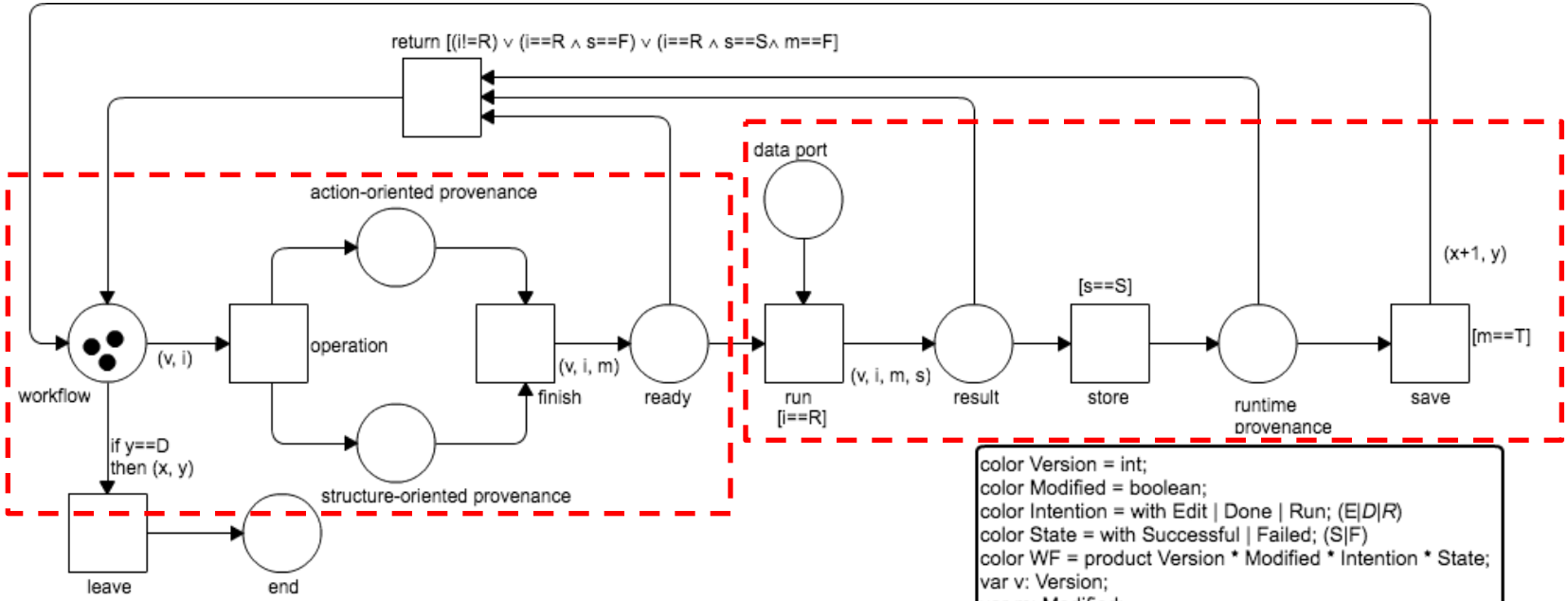
Outline

- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Colored Petri nets

- CPN is the language developed by Kurt Jensen et al.
- CPN is one type of graphical modelling language for concurrent systems.
- CPN supports the extensions with time, color and hierarchy.

Workflow-level Colored Petri Net

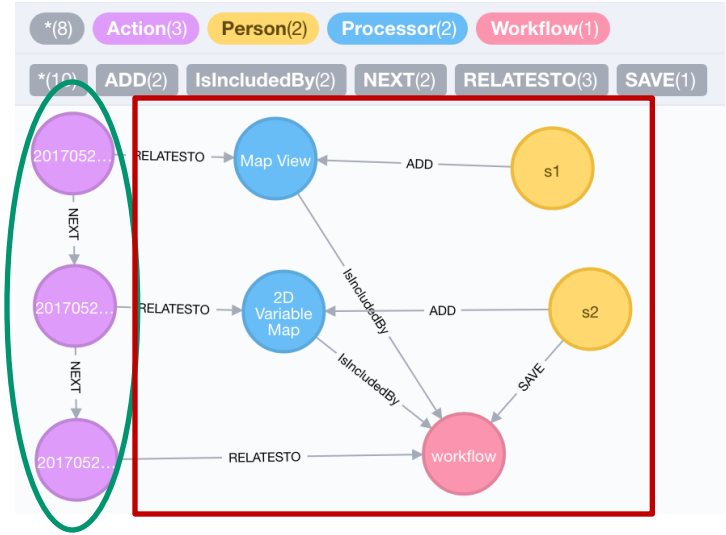


Outline

- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Provenance Storage

- Three types of provenance are stored separately
- Design-time provenance is stored in graph
- Communicate through APIs



Provenance Management

- **SQL-like Query**
  - Query types: structure, collaborator, contributor
  - Easy to handle recursive queries

Q1a: *Select structure*
Where workflow.name='W', workflow.version='3'

Q1b: *Select action*
From AOP, SOP
Where workflow.name='W', workflow.version='3'
Limit 2

Q1c: *Select derivation*
Where workflow.name='W', workflow.version='3'

Q2a: *Select collaborator*
Where workflow.name ='W', workflow.version ='3'

Q3a: *Select contributor*
Where workflow.name ='W', workflow.version ='3'

Q3b: *Select entity*
Where workflow.version ='W', workflow.version ='3', person='s1'

- **Queries written in SQL**

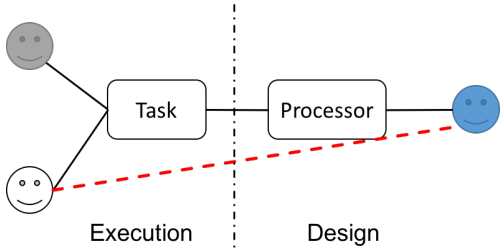
```
Q1a:
use wf;
select processor.pr_name
from include,processor , workflow
where workflow.wf_name='workflow1' and
workflow.version='1.1' and workflow.wf_id = include.en_id
        and processor.pr_id = include.included_id
union
select datalink.dl_name
from include,datalink , workflow
where workflow.wf_name='workflow1' and
workflow.version='1.1' and workflow.wf_id = include.en_id
        and datalink.dl_id = include.included_id ;
```

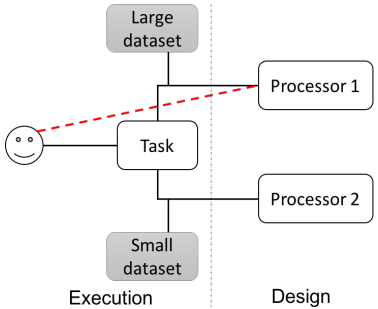
```
Q2a:
Select p1. p_name, p2. p_name
From person as p1, person as p2, workflow,action
Where  workflow.wf_name='workflow1'
        and workflow.version='1.1'
        and p1.p_name!=p2.p_name
        and workflow.wf_id = action.en_id;
```

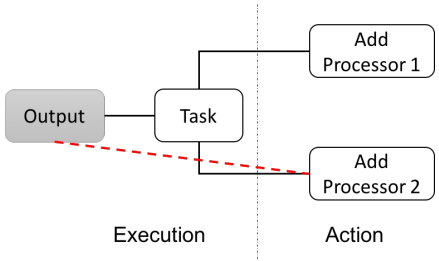
```
Q3a:
Select person.p_name
From person, action, include, workflow
Where  workflow.wf_name='workflow1'
        and workflow.version='1.1'
        and action.p_name = person.p_name
        and include.en_id = workflow.wf_id
        and action.en_id = include.included_id ;
```

Recommendation

- Collaborators are not necessarily in the same domain.
- Different methods are good at different aspects so that they fit in different scenarios.
- A scientific experiment is typically a trial and error process.





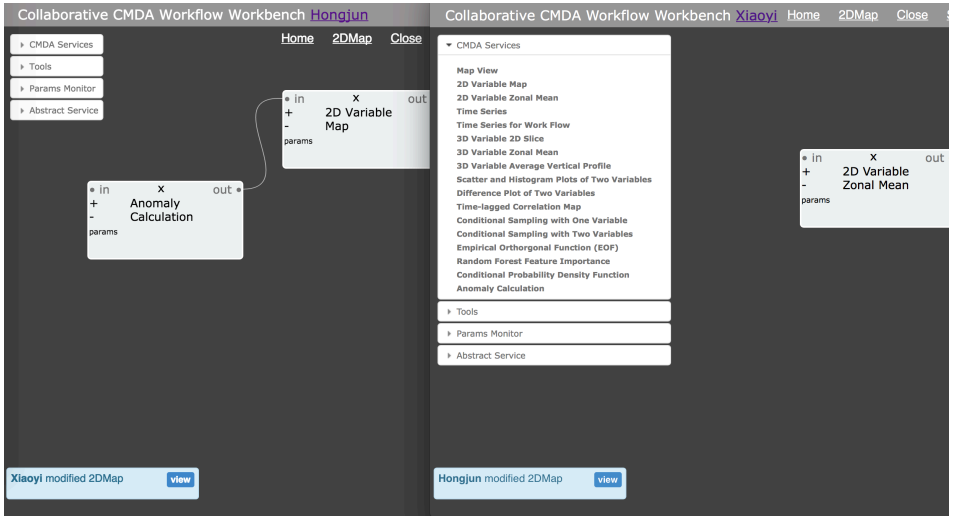


Outline

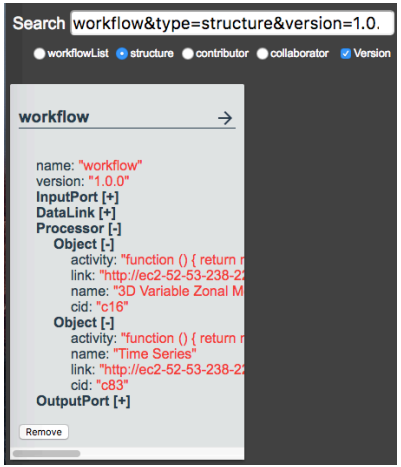
- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- **Implementations and Experiments**
- Conclusions and Future Work

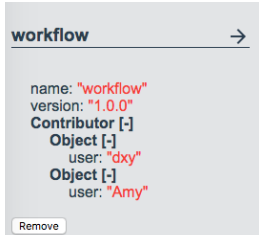
Prototype System

- **Collaborative workflow design system**



- **SQL-like query support**





Loading Time and Data Size Experiments

- **Database: Neo4j**
- **Verify the effectiveness of our model**

#people	1	2	3
#Actions	4	8	12
#Forks	0	1	2
Loading time	30 ms	43 ms	71 ms
Structure query	3 ms	9 ms	12 ms
Contributor query	13 ms	17 ms	18 ms
Collaborator query	5 ms	9 ms	12 ms
History query	14 ms	20 ms	26 ms
Size	233 KB	243 KB	277 KB

Outline

- Background and Motivation
- Related Work
- Our Method
  - Provenance Model
  - Workflow-level Colored Petri Net
  - Provenance Management
- Implementations and Experiments
- Conclusions and Future Work

Conclusions and Future Work

Conclusions

- Developed an integrated data model for workflow provenance management.
- Implemented new applications based on the model, such as advanced query and cross-provenance recommendation.
  - Feasibility and effectiveness

Future work

- Applicability of our uniform provenance model in large-scale projects