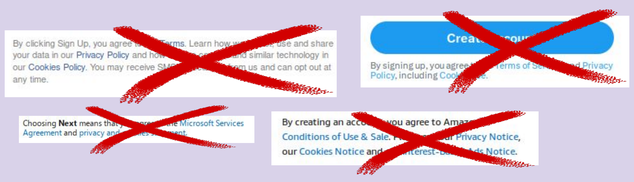
THE WEB CONFERENCE ACM

Research Track | Semantics and Knowledge

Perennial Semantic Data Terms of Use for Decentralized Web

Rui Zhao & Jun Zhao University of Oxford doi:10.1145/3589334.3645631

Motivation



"Biggest Lie on the Internet"

- (Although) We **want** *to* control our data
- (But) We **do not** have time
- (Then) How can we **automate** it
- (While) **Without** huge effort
- (And) In a **future-proof** manner
- (Thus) For decentralized Web, IoT, Smart homes, Metaverse, …

Challenges and Goals

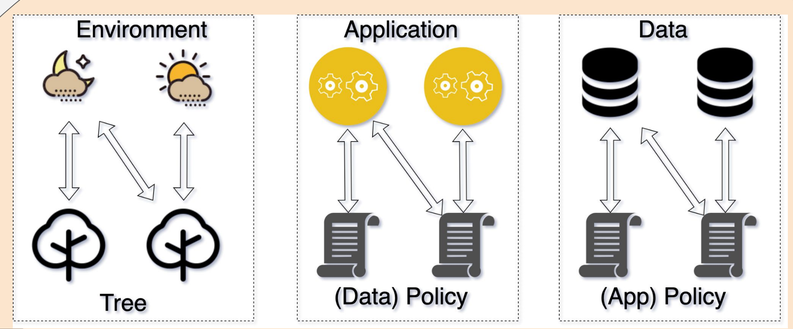
C1: Expressing **data provider's** DToU
C2: Imparting **application's** DToU
C3: Performing **compliance checking** over data usage requests;
C4: Supporting DToU policy reusing **across** *applications* and *data providers*;
C5: Facilitating **apt** DToU-compliant cross-application data sharing.

Do I need to redefine policy for every App?

"Should I permit this App?"

What if App2 uses output of App1?

Perennial Policy

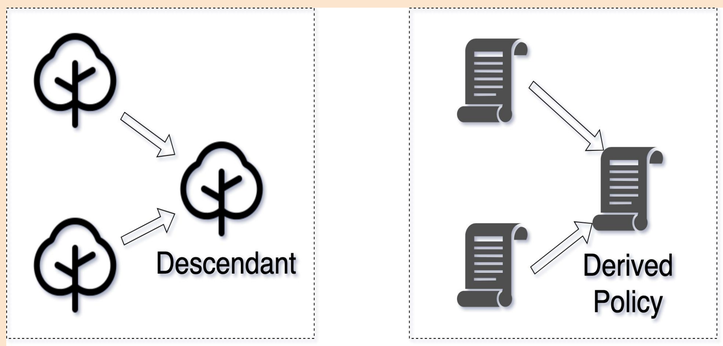


Define once and apply anywhere, for both **Data Policy** and **App Policy**

Cross-Context

Deriving-Policies

Corresponding **output policies** are derived based on input **policies** and **processing**



Example *Data Policy*

My payment info cannot be used by *<http://duckpay.com/>* , either directly or indirectly

```
:pr1 a :Prohibition;
     :mode :Use;
     :activation_condition
          [ :app_name <http://duckpay.com/> ];
     :validity_binding :attr2.
```

Application must email Alice when using the data for *research* purposes

```
:ob1 a :Obligation;
     :obligation_class :send-email;
     :args (:attr1);
     :activation_condition [ :purpose :research ].
```

Application must respect *banking* security level to use this data (payment information), and only for *make-payment* and/or *verify-ownership* purposes

```
:attr-tag2 a :Attribute;
     :name :tag-2;
     :class :banking;
     :value :nil.
:attr-tag3 a :Attribute;
     :name :tag-3;
     :class :make-payment;
     :value :nil.
:attr-tag4 a :Attribute;
     :name :tag-4;
     :class
:verify-ownership;
     :value :nil.
```

```
:tag2 a :SecurityTag;
     :attribute_ref :attr-tag2;
     :validity_binding :attr2.
:tag3 a :PurposeTag;
     :attribute_ref :attr-tag3;
     :validity_binding :attr2.
:tag4 a :PurposeTag;
     :attribute_ref :attr-tag4;
     :validity_binding :attr2.

:attr2 a :Attribute;
     :name :det;
     :class :data-content;
     :value :payment-details.
```

Example *App Policy*

HappyShop reads data *<http://a.b/payment-info>* (through input port *"payment-info-in"*), promises to comply with security level *:banking*, and uses the data only for *:making-payment* purpose; it will send the data to a downstream, named *<http://goodpay.com/>*, for purpose of *:making-payment*

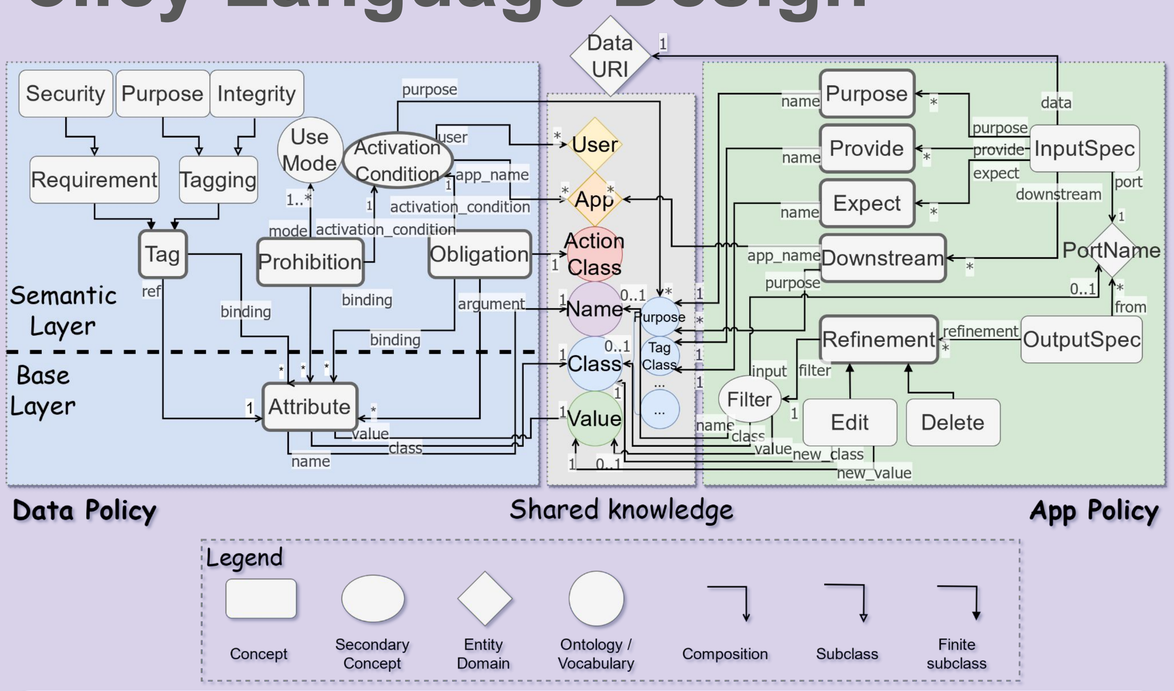
```
:input1 a :InputSpec;
     :data <http://a.b/payment-info>;
     :port [:name "payment-info-in"];
     :security :banking;
     :purpose :making-payment;
     :downstream [  :app_name <http://goodpay.com/>;
                    :purpose :making-payment ].
```

HappyShop may record purchase histories in user's Pod, which contains derived data (the copy) of the delivery address, and a declassified version of payment details

```
:out1 a :OutputSpec;
     :from [ :name "address-in" ],
     [ :name "payment-info-in" ];
     :refinement :refine-no-payment-details.

:refine-no-payment-details a :Delete;
     :filter [ :class :data-content;
               :value :payment-details ].
```

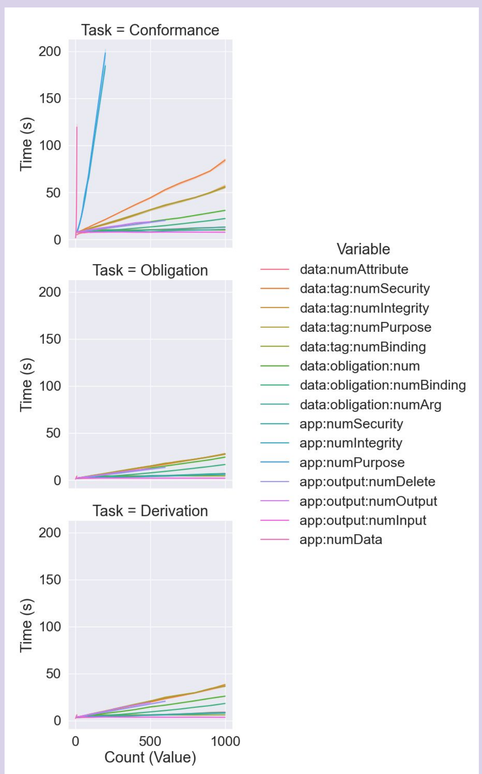
Policy Language Design



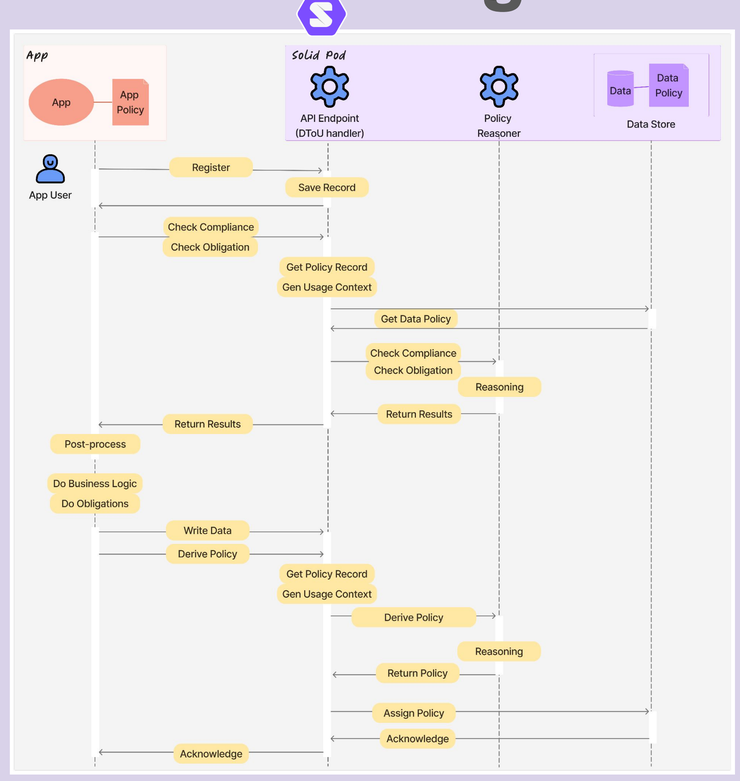
Reasoning tasks

- Compliance Check: Does the App Policy *comply with* all Data Policies?
- Obligation Check: What *obligations* should the App/Operator perform?
- Policy Derivation: What Data Policies should be associated with *output data*?

SoLiD Integration



Scalability



Policy Sequence Diagram

OXFORD MARTIN SCHOOL

UNIVERSITY OF OXFORD

FIND OUT MORE!

Site: https://renyuneyun.github.io/solid-dtou/
Video: https://youtu.be/ERFZdnOq09Y
Email: rui.zhao@cs.ox.ac.uk
Project: https://ewada.ox.ac.uk/

Artifacts Available acm