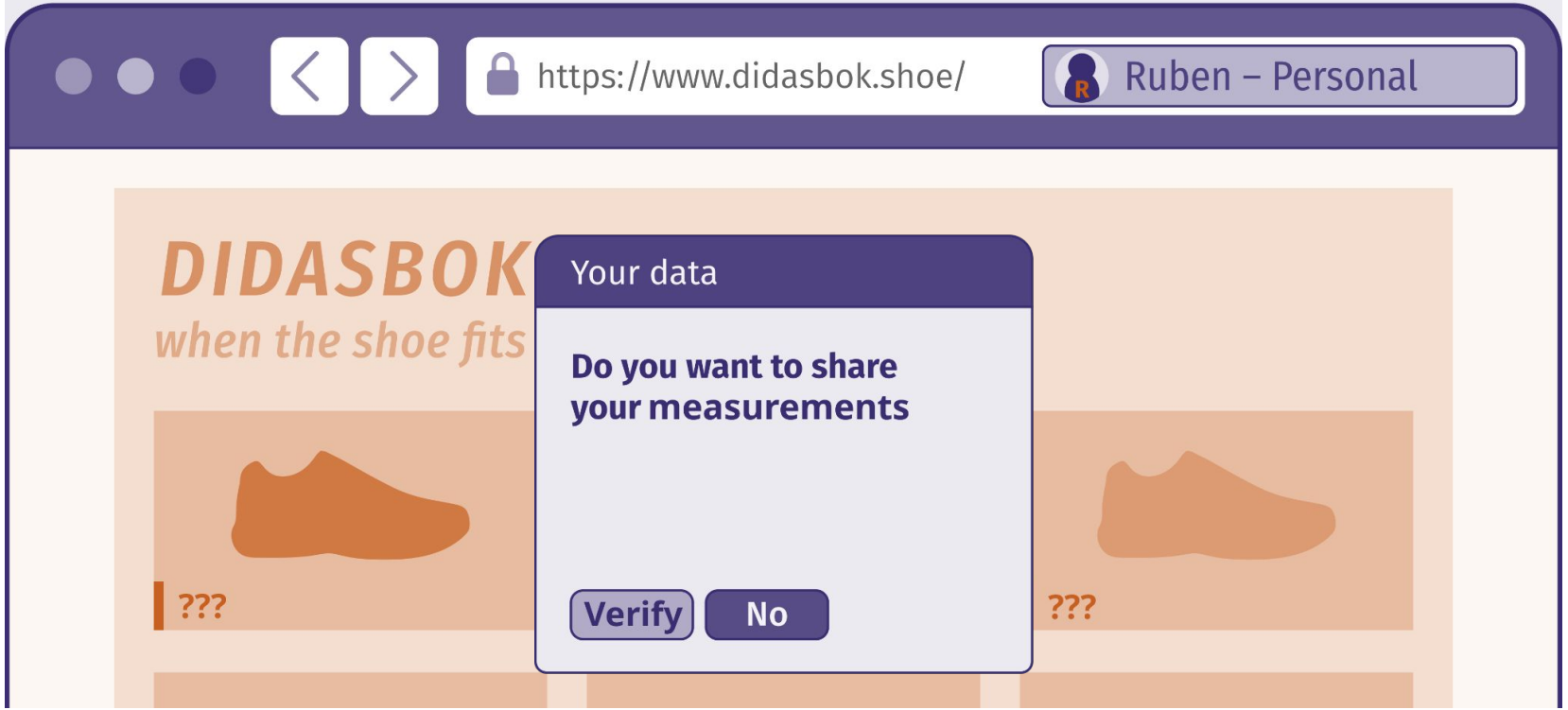
From Access to Usage Control with User-Managed Access

Wout Slabbinck

Ruben Dedecker


https://www.didasbok.shoe/
Ruben – Personal
DIDASBOK
when the shoe fits
???
???
???

Requesting Data Access



Requiring Verifiable Claims

Imposing Usage restrictions

Providing information

Access provision requires a set of functionality …

Actors **Request Access** to my information

I define **Access Constraints** based on dynamic (provable) claims

I define **Usage Restrictions** over shared information

My data space **Provides Access** based on these criteria

… that is not covered by simple access control

Actors **Request Access** to my information

I define **Access Constraints** based on dynamic (provable) claims

I define **Usage Restrictions** over shared information

My data space **Provides Access** based on these criteria

UMA provides access requests and claim negotiation …

Actors **Request Access** to my information

I define **Access Constraints** based on dynamic (provable) claims

I define **Usage Restrictions** over shared information

My pod **Provides Access** based on these criteria

… we build on towards Usage Control with ODRL

Actors **Request Access** to my information

I define **Access Constraints** based on dynamic (provable) claims

I define **Usage Restrictions** over shared information

My pod **Provides Access** based on these criteria

From Access to Usage Control with User-Managed Access

- What is “User-Managed Access”?
- Bridging the gap with User-Managed Access
- A Solid UMA architecture

From Access to Usage Control with User-Managed Access

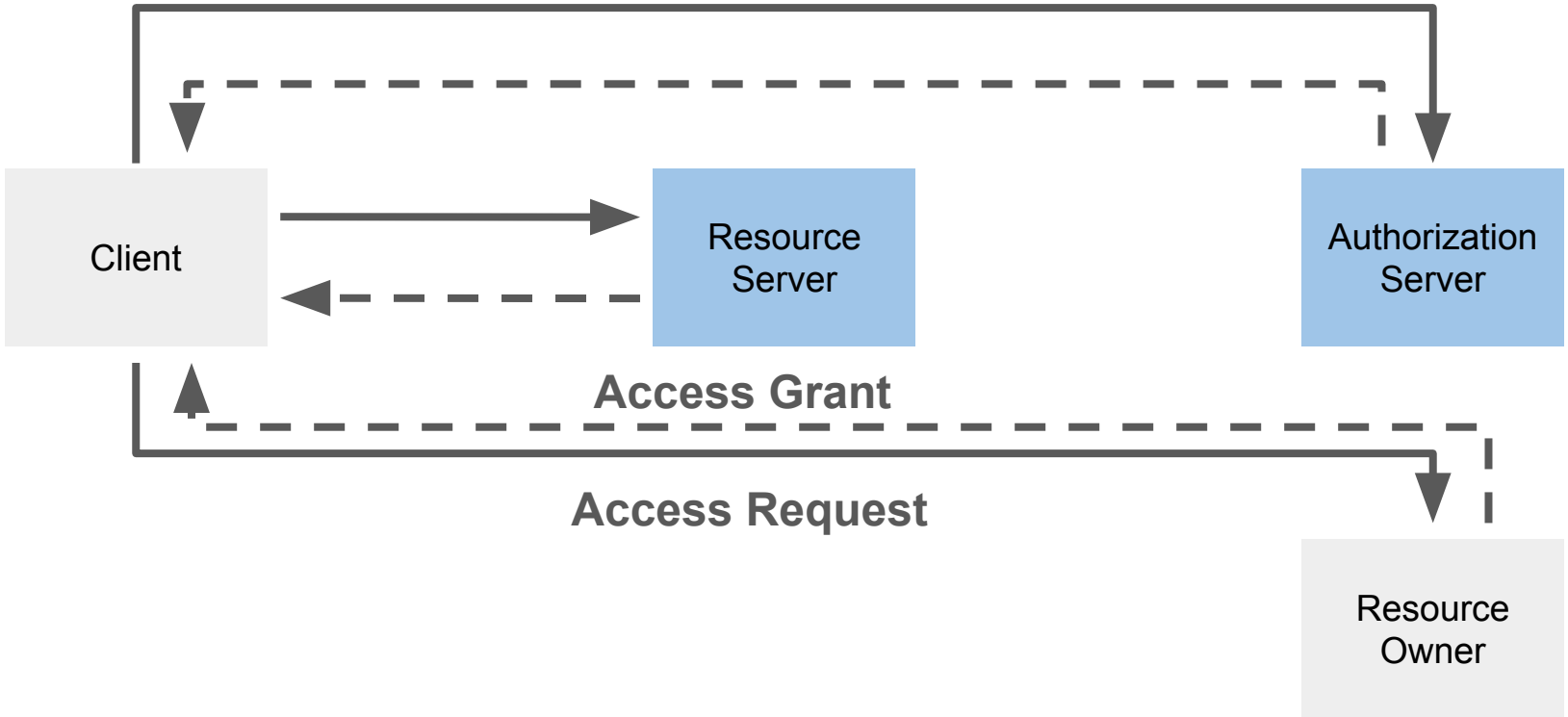
- **What is “User-Managed Access”?**
- Bridging the gap with User-Managed Access
- A Solid UMA architecture

User-Managed Access (UMA)

Extension of **Oauth 2.0**

- Delegated Access Control
  - Allow third-party apps access to protected resources (without having to share credentials)
- Separation of concerns
  - Authorization Server and Resource Server
  - Standardizes obtaining an Access Token

Oauth 2.0: Delegated Access and Separation of concerns



Oauth 2.0: https://datatracker.ietf.org/doc/html/rfc6749

User-Managed Access (UMA)

Extension of **Oauth 2.0**

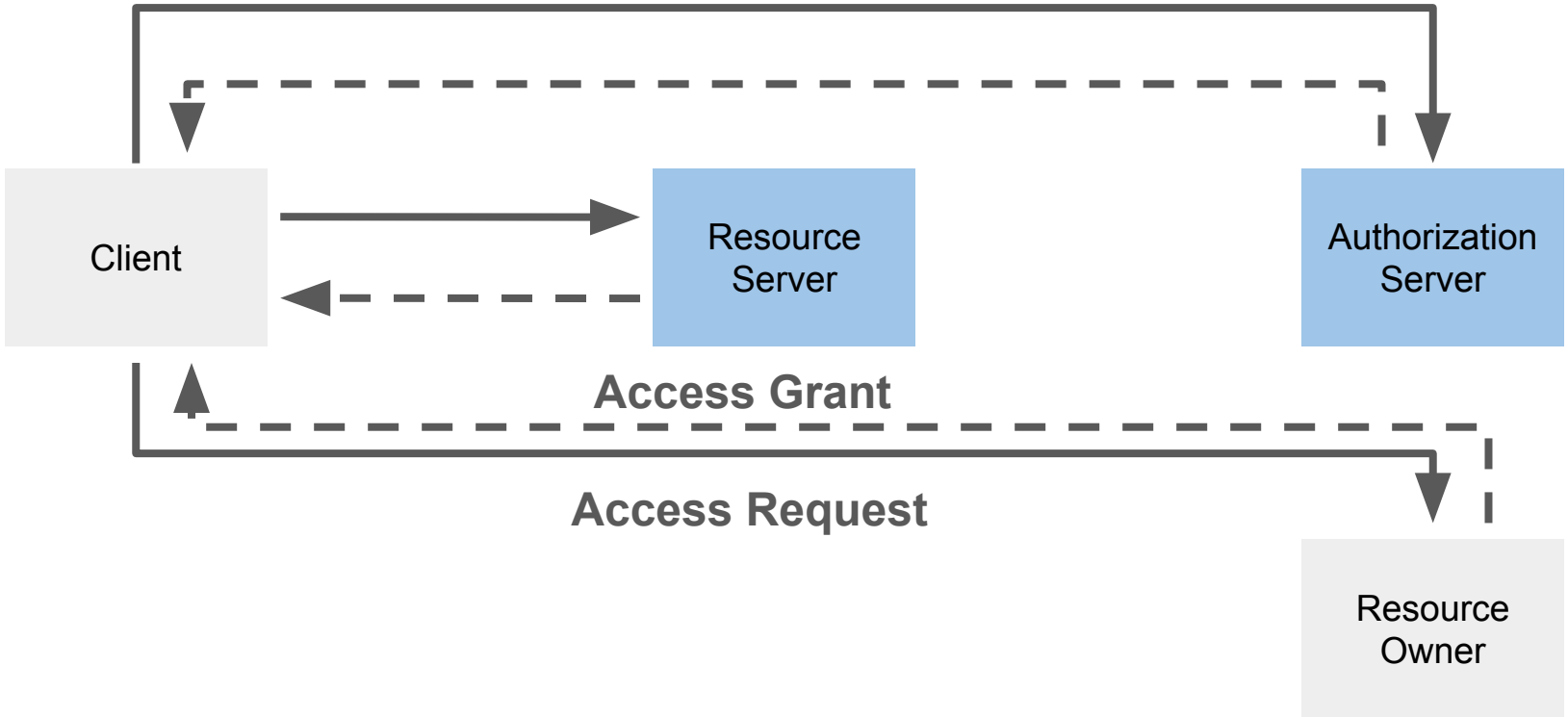
- Delegated Access Control
  - Allow third-party apps access to protected resources (without having to share credentials)
- Separation of concerns
  - Authorization Server and Resource Server
  - Standardizes obtaining an Access Token

UMA introduces

- Asynchronous access delegation through the distinction of **Requesting Party** and **Resource Owner**

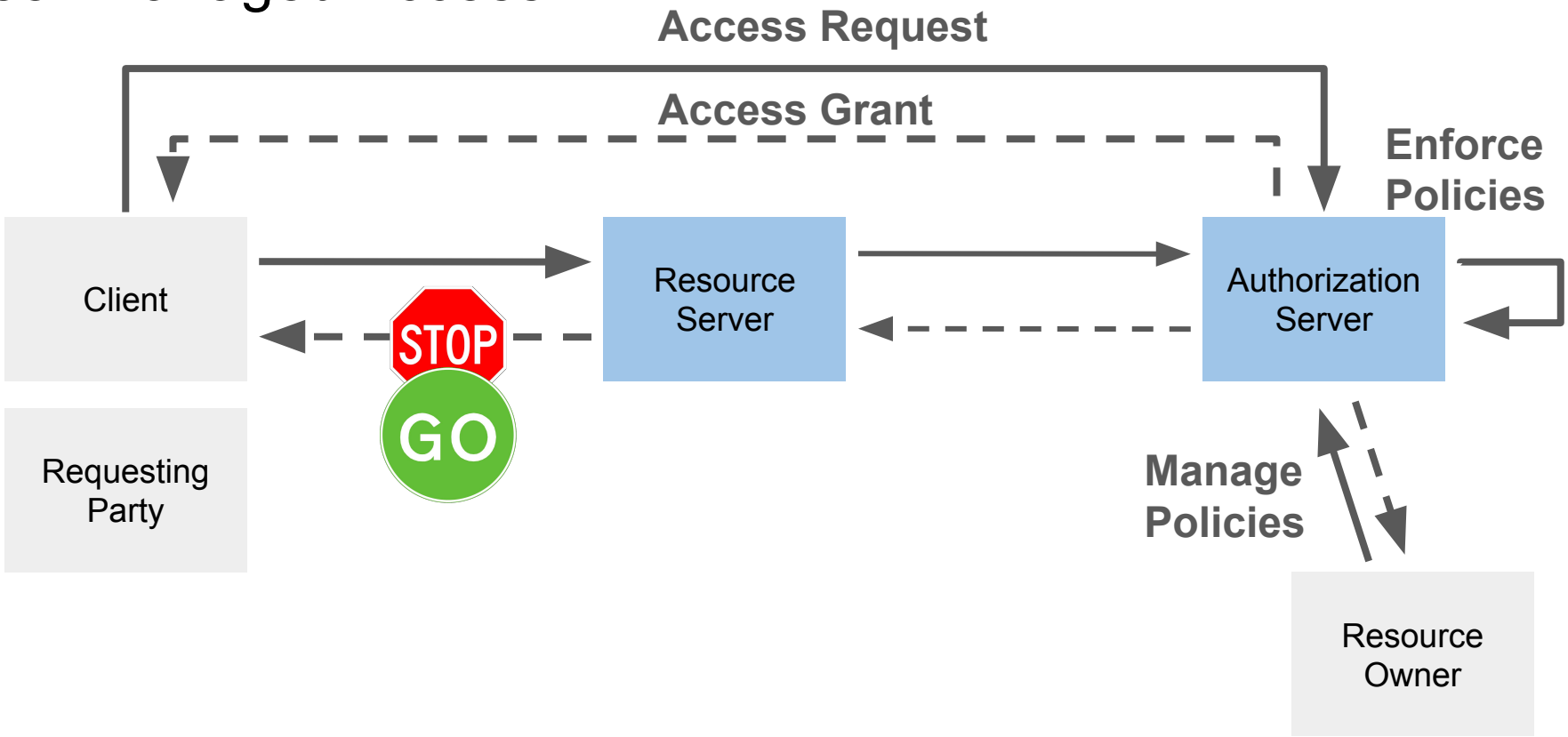
UMA: https://docs.kantarainitiative.org/uma/wg/rec-oauth-uma-grant-2.0.html

Oauth 2.0: Delegated access and Separation of concerns



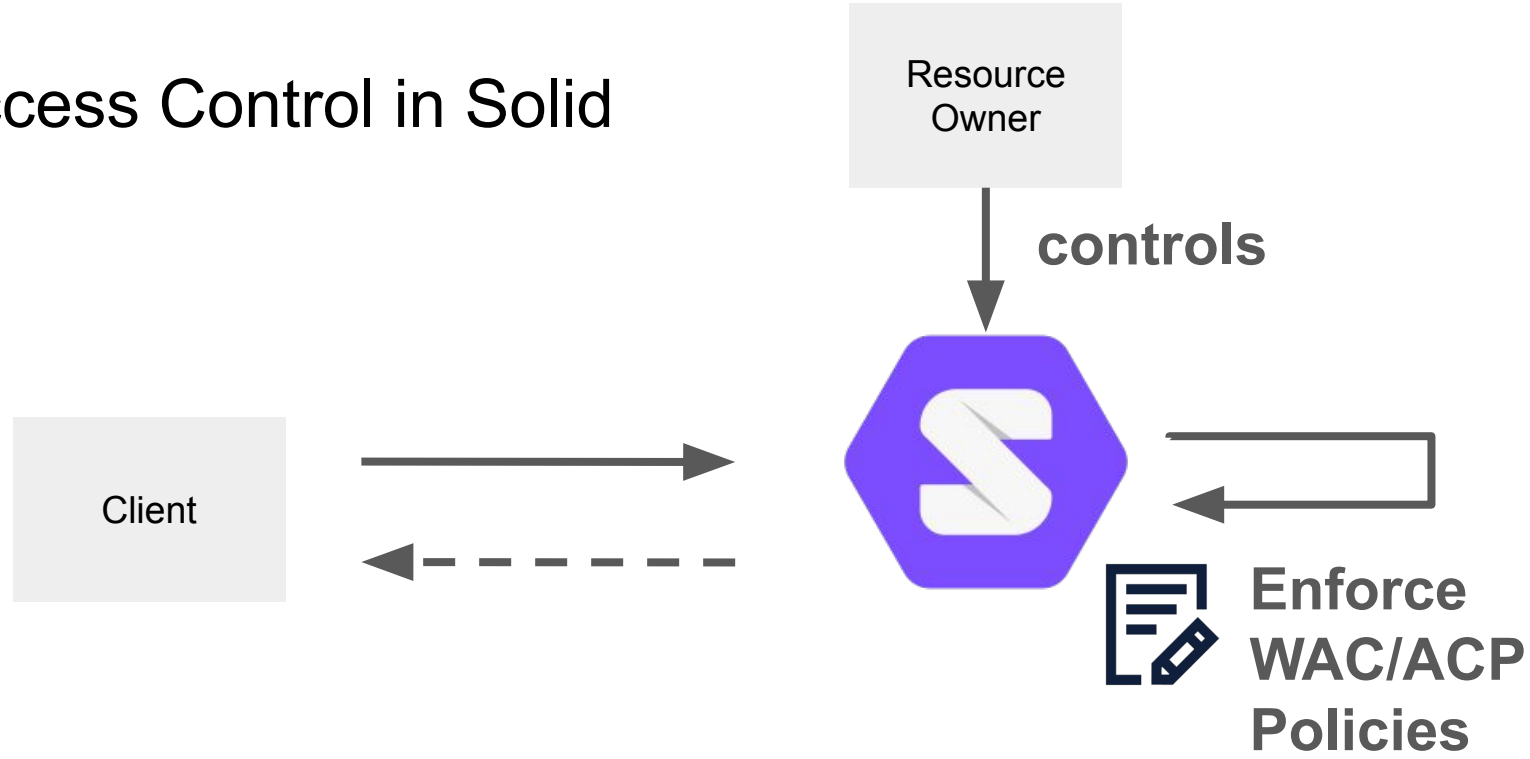
Oauth 2.0: https://datatracker.ietf.org/doc/html/rfc6749

User-Managed Access



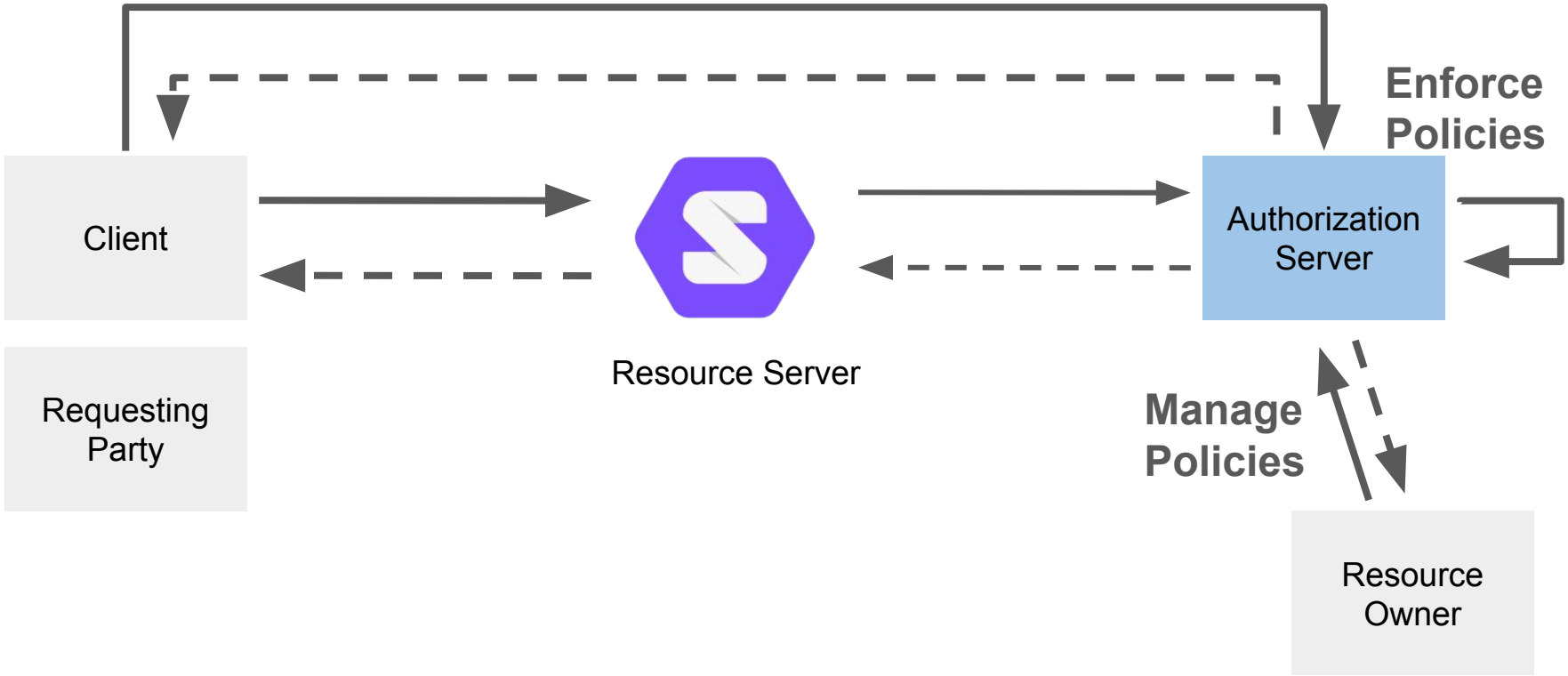
UMA: https://docs.kantarainitiative.org/uma/wg/rec-oauth-uma-grant-2.0.html

Access Control in Solid



Solid Protocol (§11): https://solidproject.org/TR/protocol#authorization

User-Managed Access and Solid



Solid-OIDC (§9): https://solidproject.org/TR/oidc#resource

From Access to Usage Control with User-Managed Access

- **What is “User-Managed Access”?**
- Bridging the gap with User-Managed Access
- A Solid UMA architecture

From Access to Usage Control with User-Managed Access

- What is “User-Managed Access”?
- **Bridging the gap with User-Managed Access**
- A Solid UMA architecture

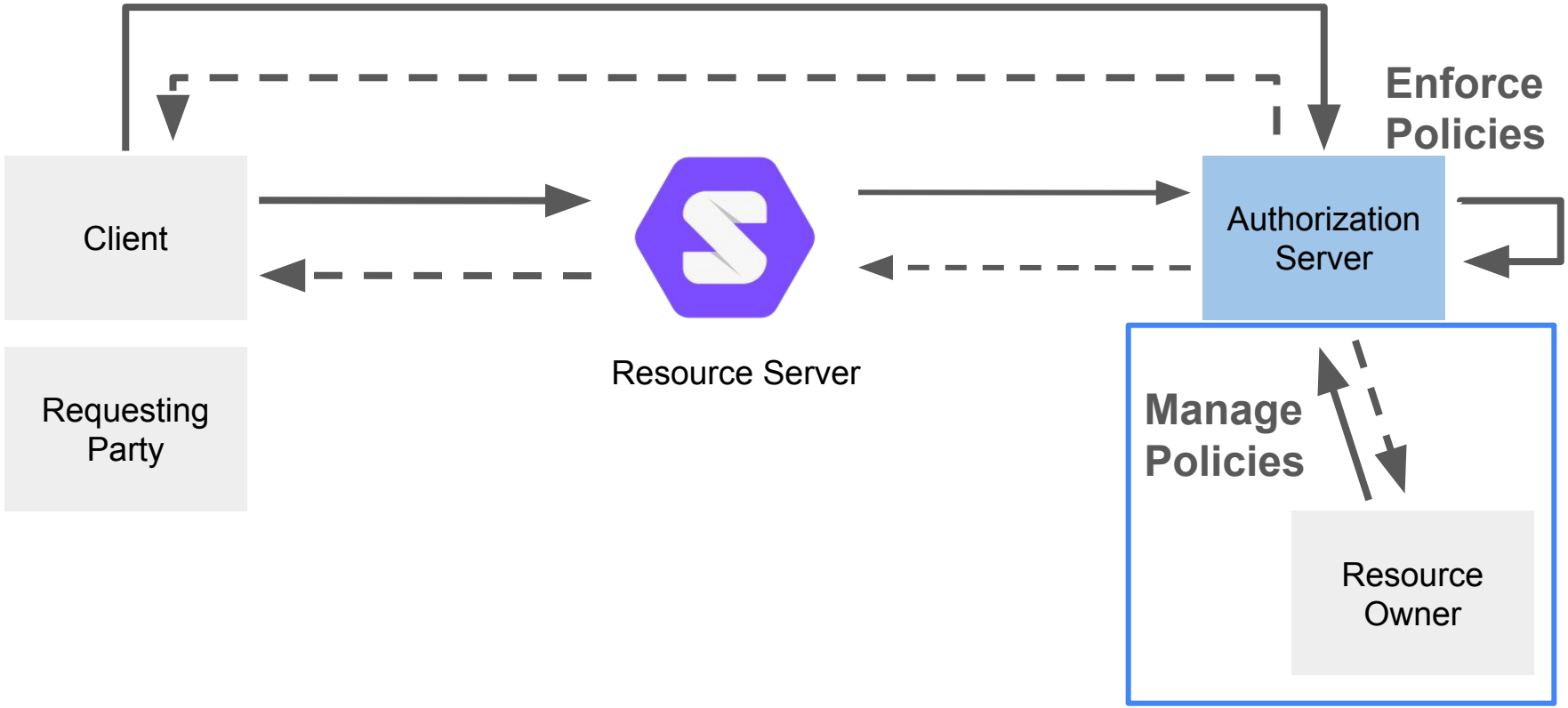
Bridging the gap with User-Managed Access

- Requesting and Granting Access
- Is WebID enough? Handling more Generic Claims
- Defining terms of usage over data

Bridging the gap with User-Managed Access

- **Requesting and Granting Access**
- Is WebID enough? Handling more Generic Claims
- Defining terms of usage over data

Requesting and Granting Access
What if the Requesting Party does not have access yet?



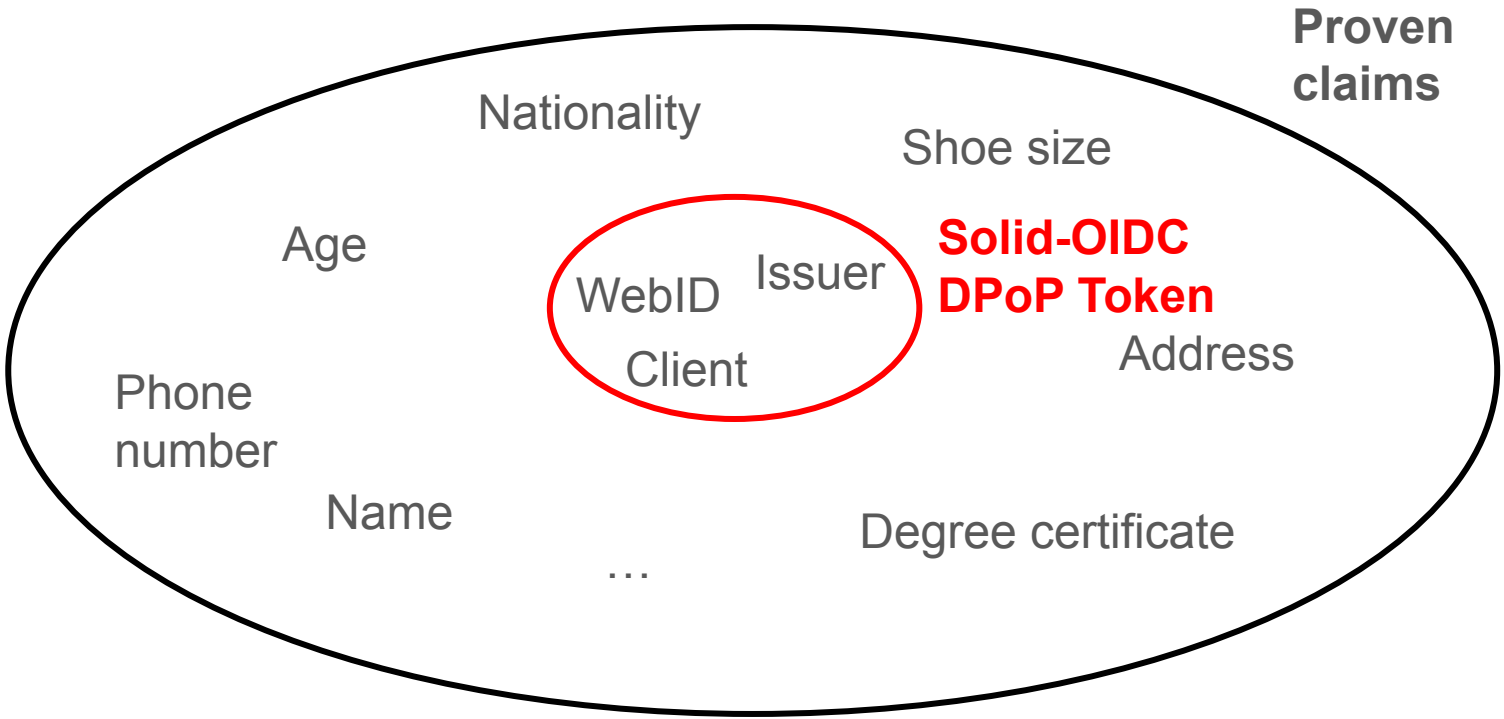
Bridging the gap with User-Managed Access

- **Requesting and Granting Access**
- Is WebID enough? Handling more Generic Claims
- Defining terms of usage over data

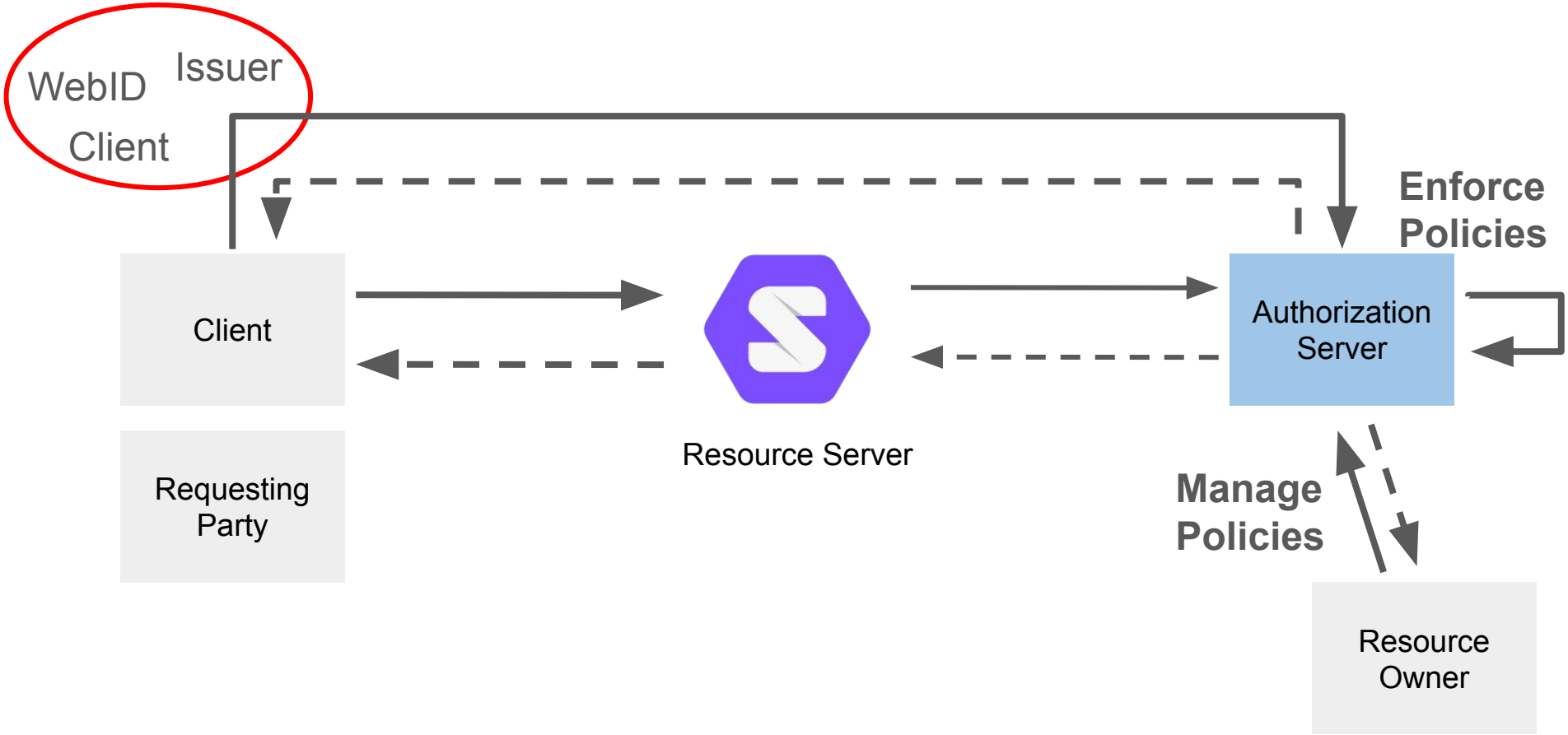
Bridging the gap with User-Managed Access

- Requesting and Granting Access
- **Is WebID enough? Handling more Generic Claims**
- Defining terms of usage over data

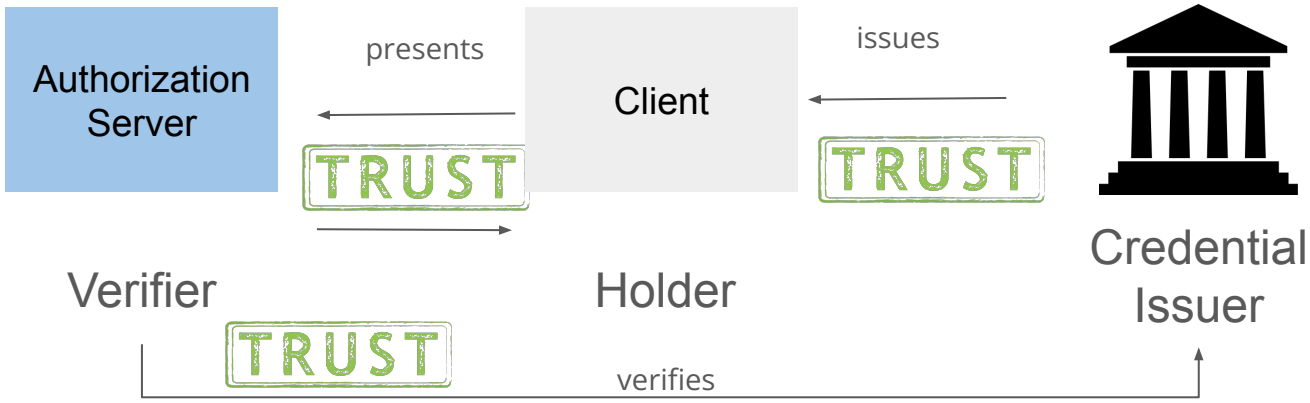
Is WebID enough? Handling more Generic Claims
Which set of claims can we proof?



Is WebID enough? Handling more Generic Claims



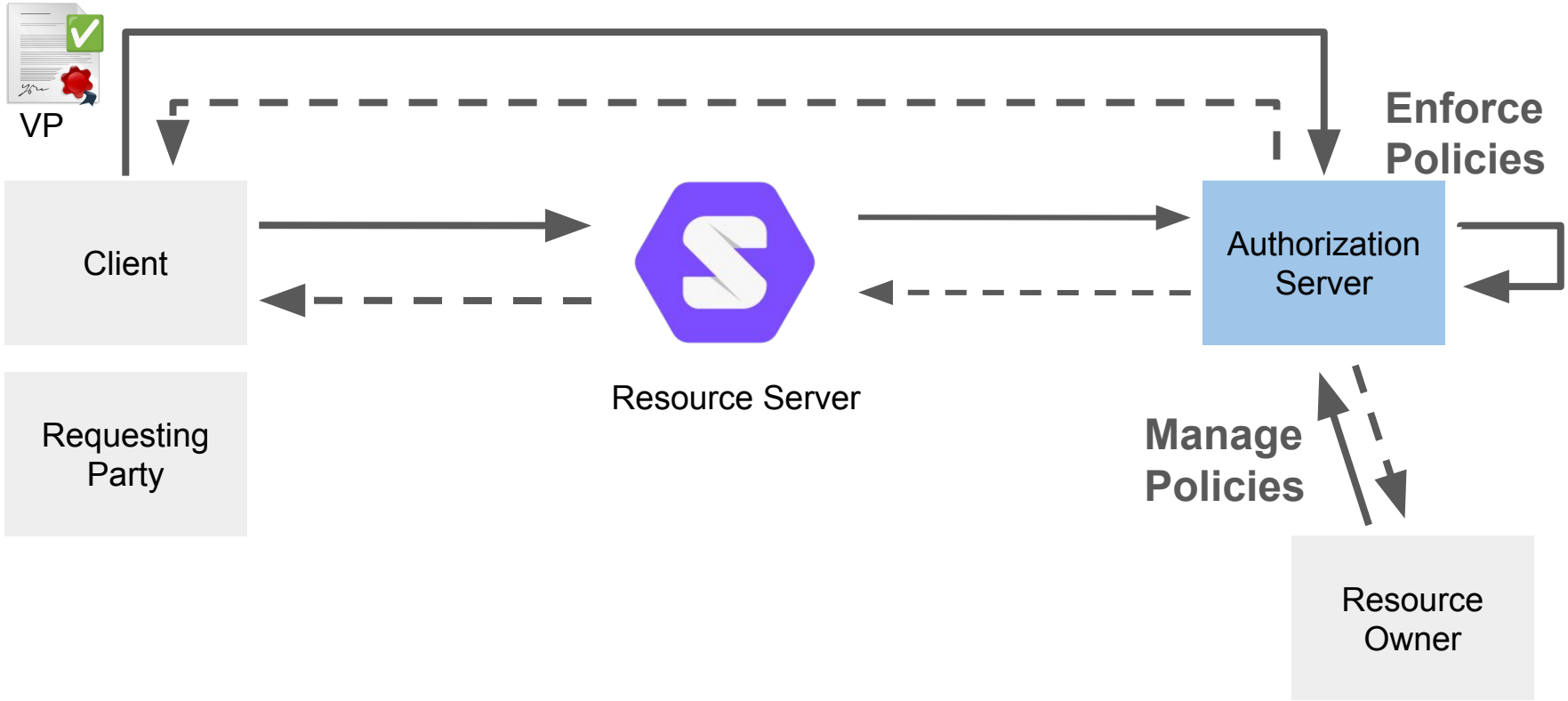
Is WebID enough? Handling more Generic Claims Verifiable Credentials



Verifier trust Issuer -> Issuer trusts Holder -> Verifier trusts Holder

https://www.w3.org/TR/vc-data-model-2.0/

Is WebID enough? Handling more Generic Claims Verifiable Credentials



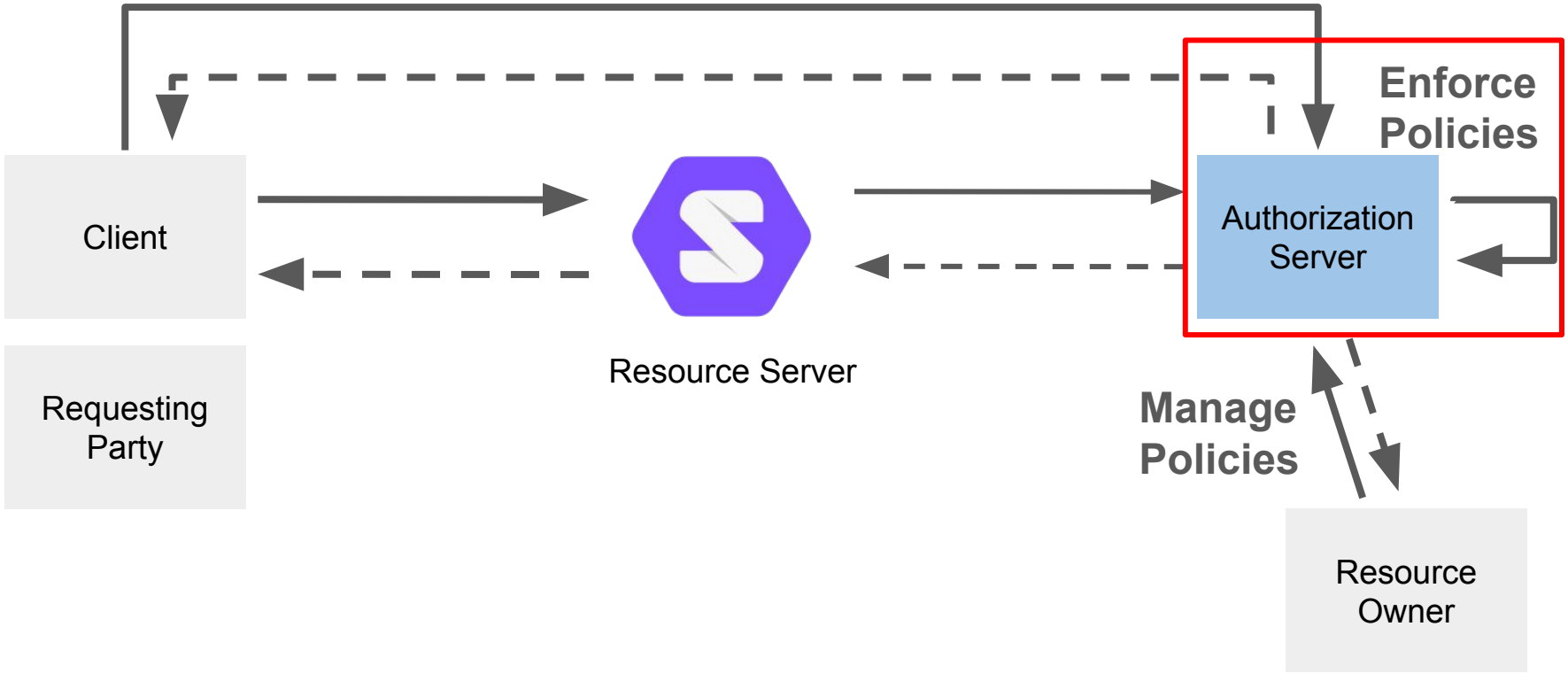
Bridging the gap with User-Managed Access

- Requesting and Granting Access
- **Is WebID enough? Handling more Generic Claims**
- Defining terms of usage over data

Bridging the gap with User-Managed Access

- Requesting and Granting Access
- Is WebID enough? Handling more Generic Claims
- **Defining terms of usage over data**

Defining terms of usage over data: Usage Control



Defining terms of usage over data: Usage Control

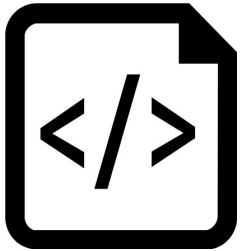
WHO can perform what **ACTION** on which **RESOURCE**

Create

Read

Update

Delete



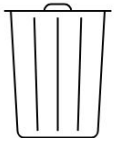
Defining terms of usage over data: ~~Usage Control~~

WHO can perform what **ACTION** on which **RESOURCE**

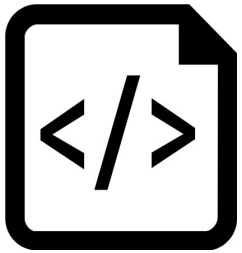
Create

Read

Update



Delete



ACCESS CONTROL RULE

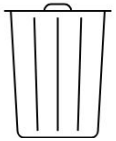
Defining terms of usage over data: Usage Control

WHO can perform what **ACTION** on which **RESOURCE** under which **CONDITIONS**

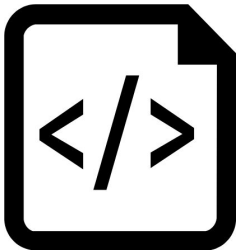
Create

Read

Update



Delete

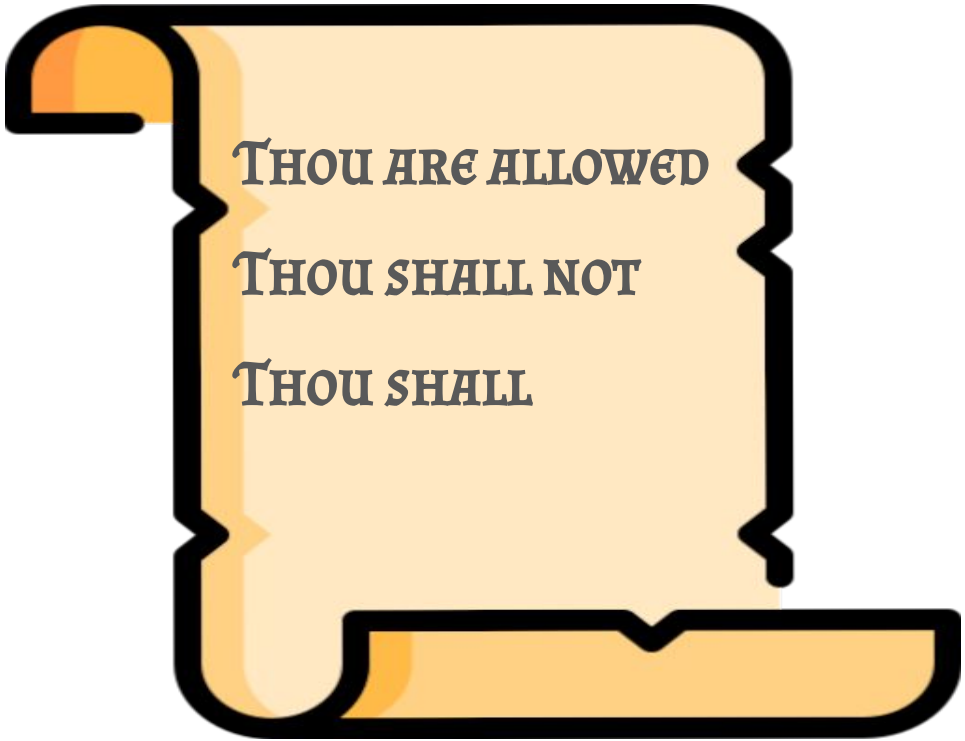


Temporal
Purpose
…

Defining terms of usage over data: Usage Control

Usage Control Rule

- Access Control + conditions
- **Deontic concepts**
  - Permission rules
  - Prohibition rules
  - Obligation rules



Defining terms of usage over data with ODRL

A W3C standard,
enabling interoperability

Policy

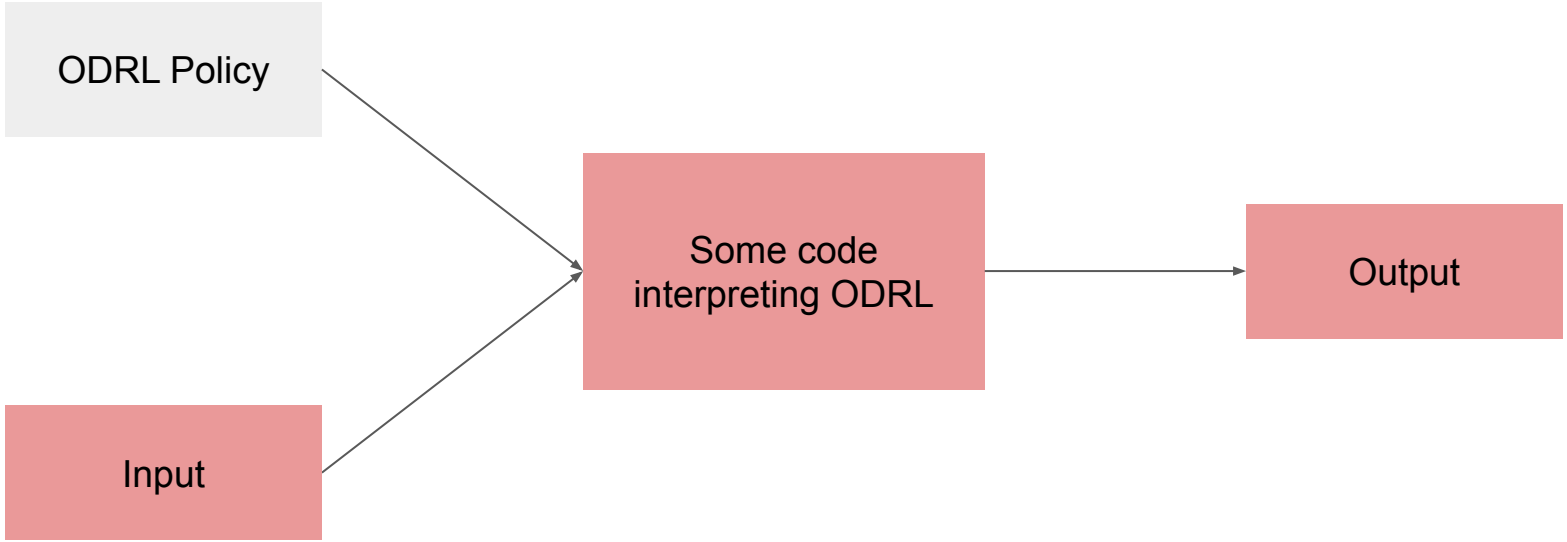
```
ex:policy a odrl:Set;
  odrl:permission ex:permission.

ex:permission a odrl:Permission;
  odrl:target <urn:uuid:resource>;
  odrl:assignee <urn:uuid:bob>;
  odrl:action odrl:read;
  odrl:constraint <urn:uuid:purpose_constraint>.
```

How to systematically interpret ODRL policies?

Standardized

Custom/
Not Standardized



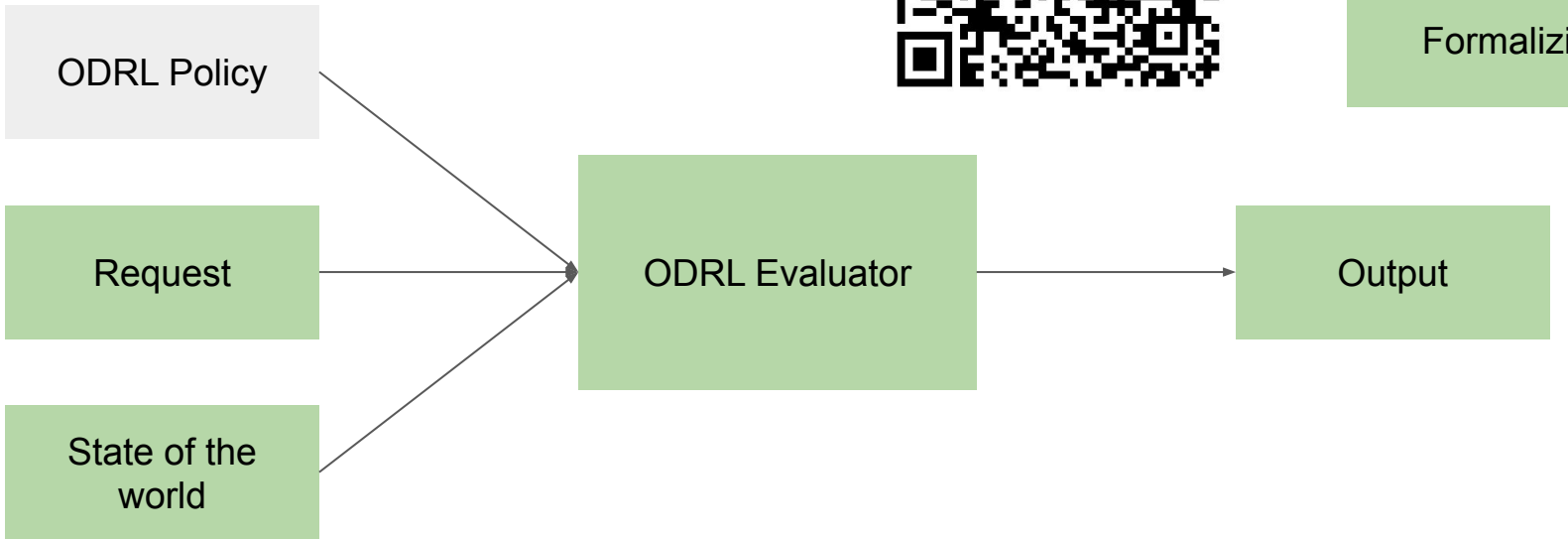
Formal Semantics for ODRL

ESWC25 paper

Standardized

Custom/
Not Standardized

Formalizing



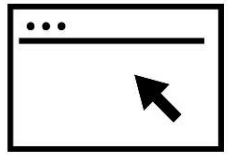
From Access to Usage Control with User-Managed Access

- What is “User-Managed Access”?
- **Bridging the gap with User-Managed Access**
  - **Requesting and Granting Access**
  - **Is WebID enough? Handling more Generic Claims**
  - **Defining terms of usage over data**
- A Solid UMA architecture

From Access to Usage Control with User-Managed Access

- What is “User-Managed Access”?
- Bridging the gap with User-Managed Access
- **A Solid UMA architecture**

Solid chitecture



Requesting Party

Legend: components data

SolidLab's UMA AS

Authorization Server

- Interface
- Credential Verifier
- Policy Engine
- Policy Storage

Community Solid Server

Resource Server

- Interface
- Data Storage

FACULTY OF ENGINEERING AND ARCHITECTURE

DEPARTMENT ELIS
RESEARCH GROUP IDLab Technology

From Access to Usage Control with User-Managed Access

Wout Slabbinck, Ruben Dedecker, Wouter Termont, Beatriz Esteves, Pieter Colpaert, Ruben Verborgh

GHENT UNIVERSITY

imec

IDLab
INTERNET & DATA LAB