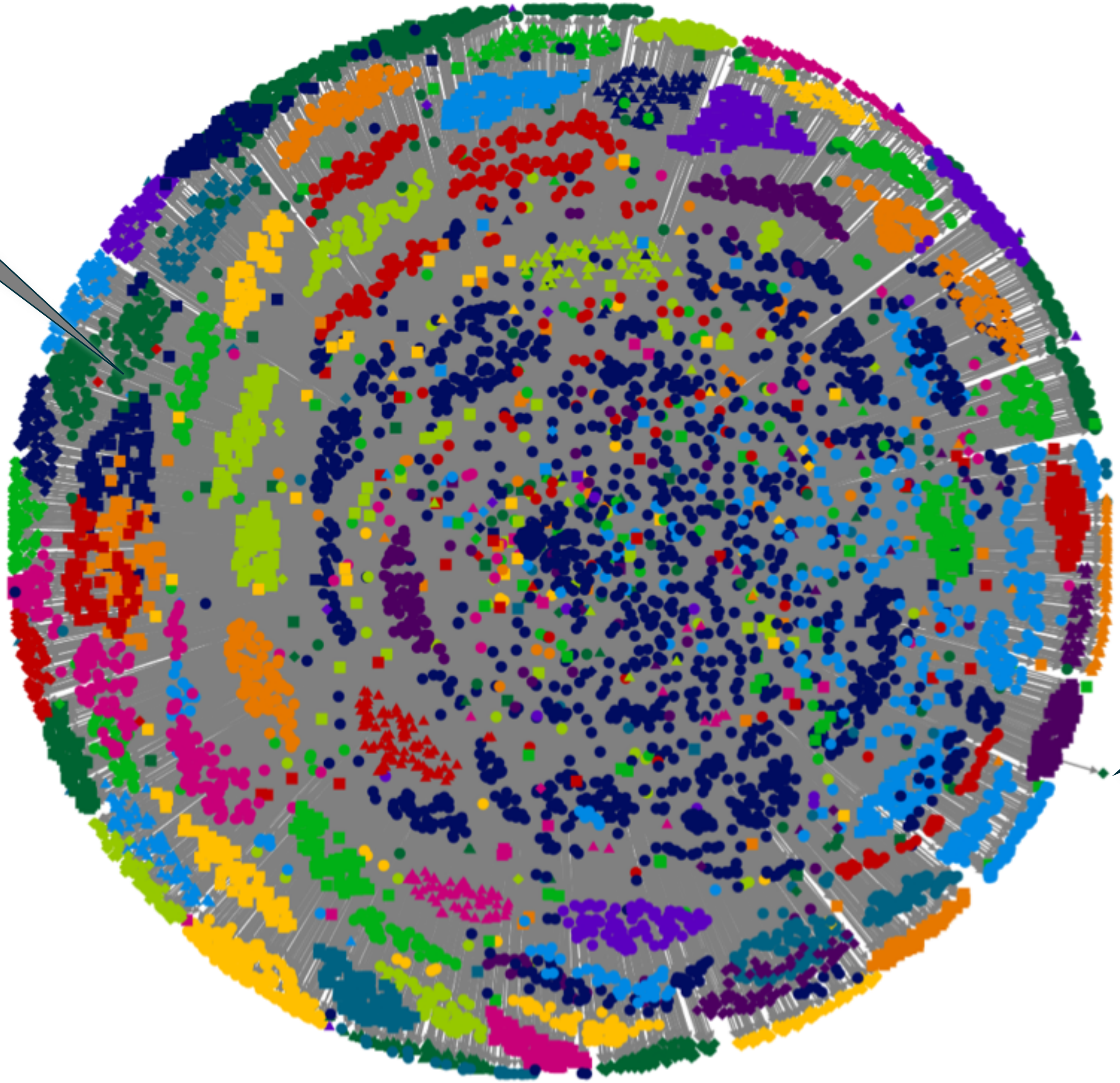
# GDPR, PSD2, CIAM, and the Role of User-Managed Access 2.0

Eve Maler
VP Innovation & Emerging Technology, ForgeRock
@xmlgrrl | eve.maler@forgerock.com
Chair and founder, Kantara UMA Work Group
@UMAWG | tinyurl.com/umawg

24 Jan 2018

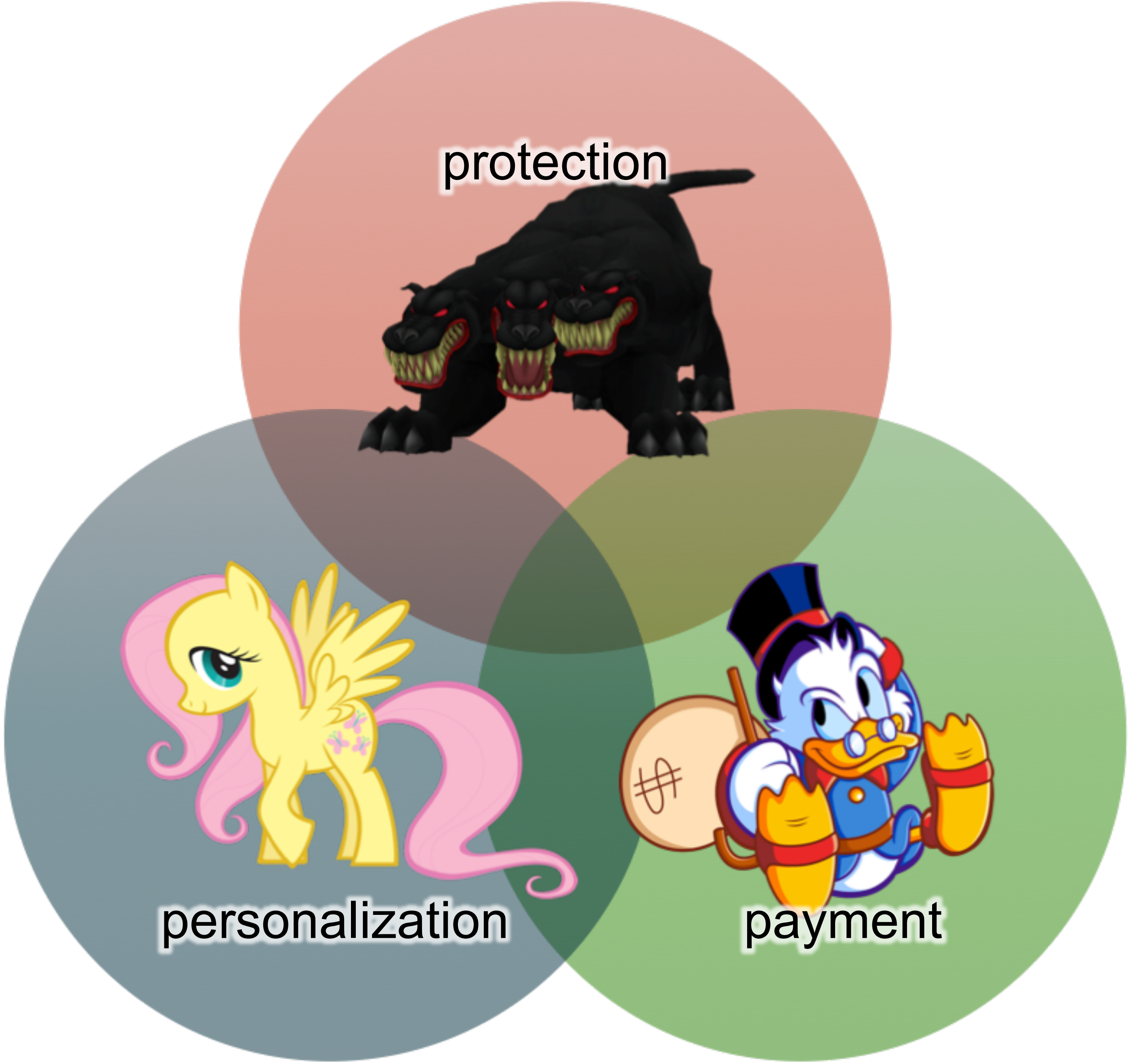
# The new business imperatives

LIKE A BOSS



We're all individuals!

I'm not



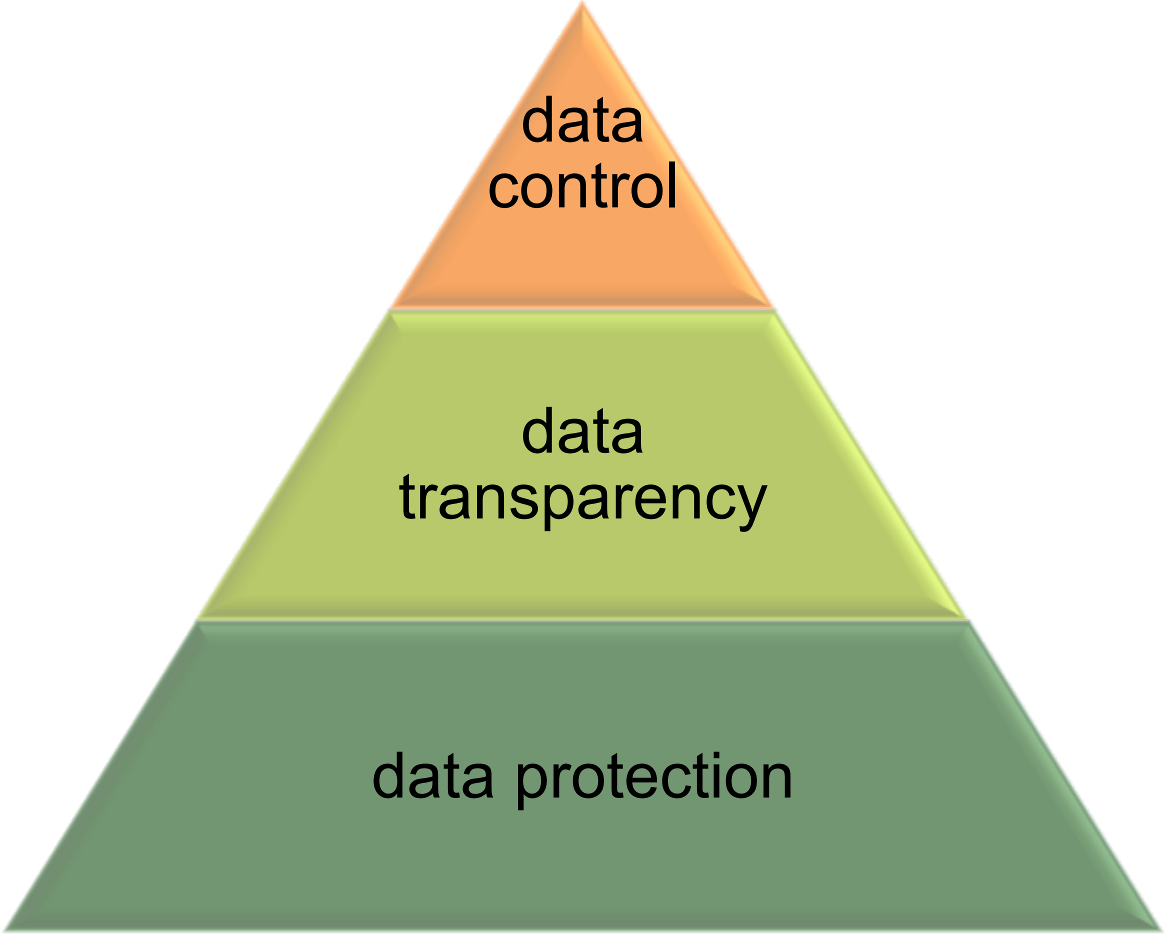
protection

personalization

payment

# What makes data privacy regulations different this time around?

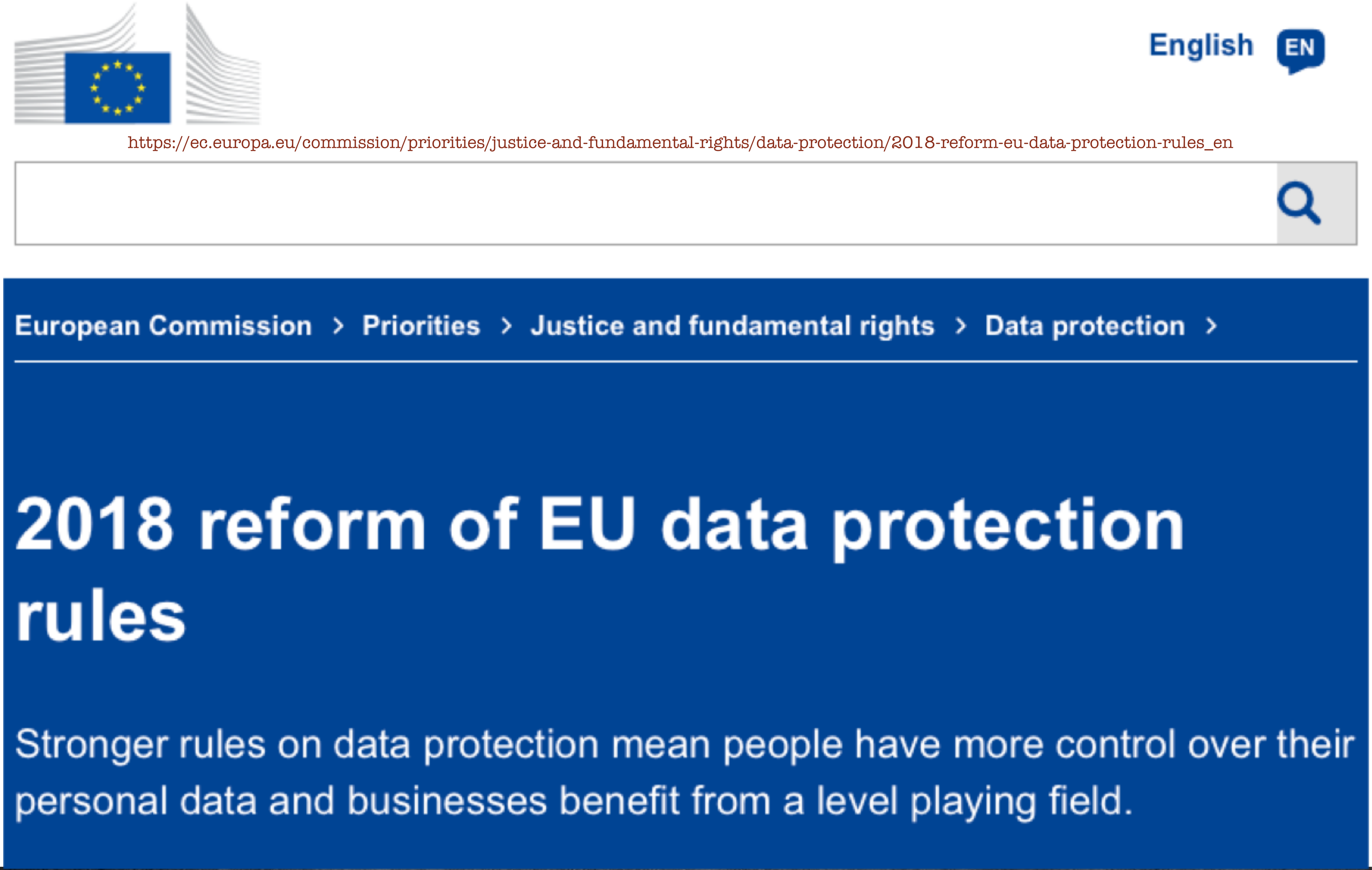
- Virality
- Digital transformation
- Aspirations



data control

data transparency

data protection



https://ec.europa.eu/commission/priorities/justice-and-fundamental-rights/data-protection/2018-reform-eu-data-protection-rules_en

European Commission › Priorities › Justice and fundamental rights › Data protection ›

# 2018 reform of EU data protection rules

Stronger rules on data protection mean people have more control over their personal data and businesses benefit from a level playing field.

## Take steps...

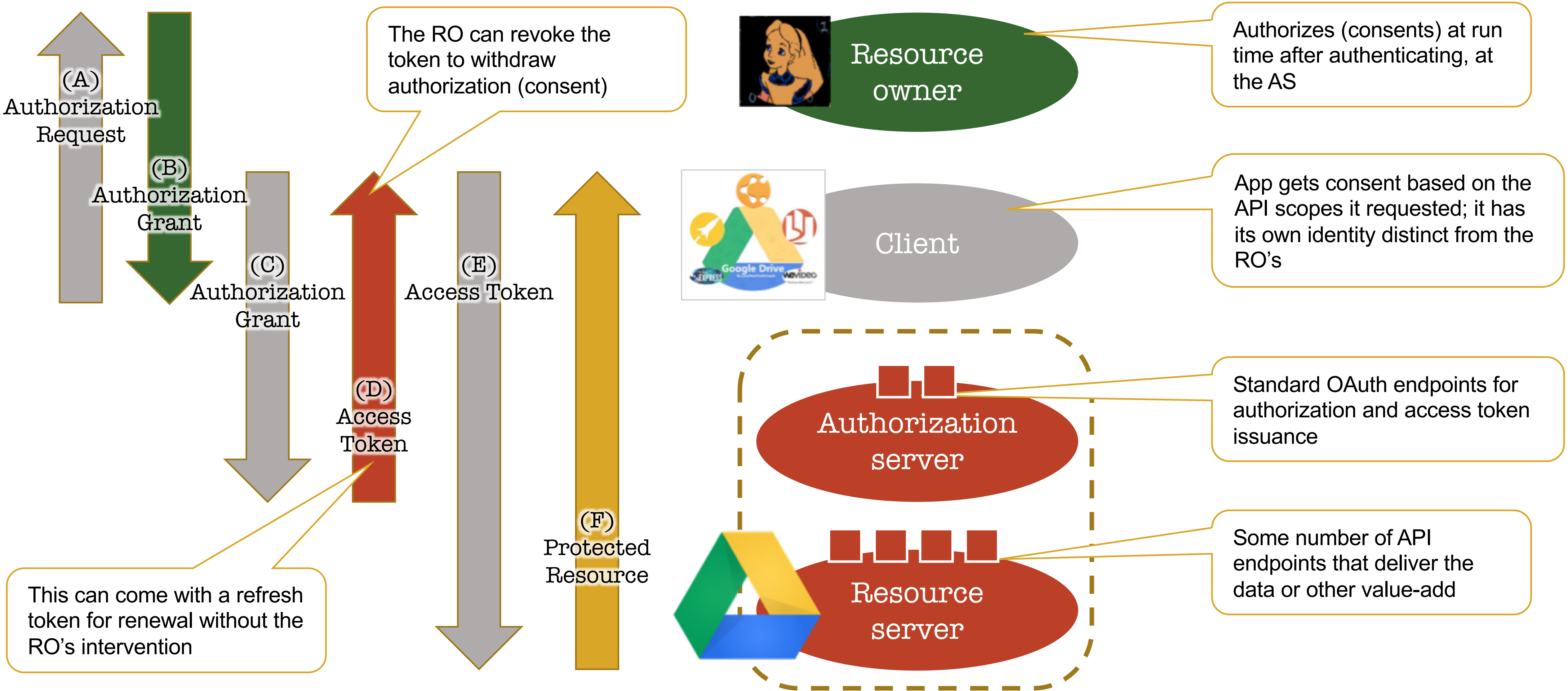
1. Find where digital transformation opportunities and user trust risks intersect
2. Conceive of personal data as a joint asset
3. Lean in to consent
4. Take advantage of identity and access management for building trust

https://www.flickr.com/photos/adpowers/16808090/ | CC BY 2.0

# UMA 101

# OAuth is for constrained delegation to apps

## It has helped to kill the “password anti-pattern”



The RO can revoke the token to withdraw authorization (consent)

This can come with a refresh token for renewal without the RO’s intervention

Authorizes (consents) at run time after authenticating, at the AS

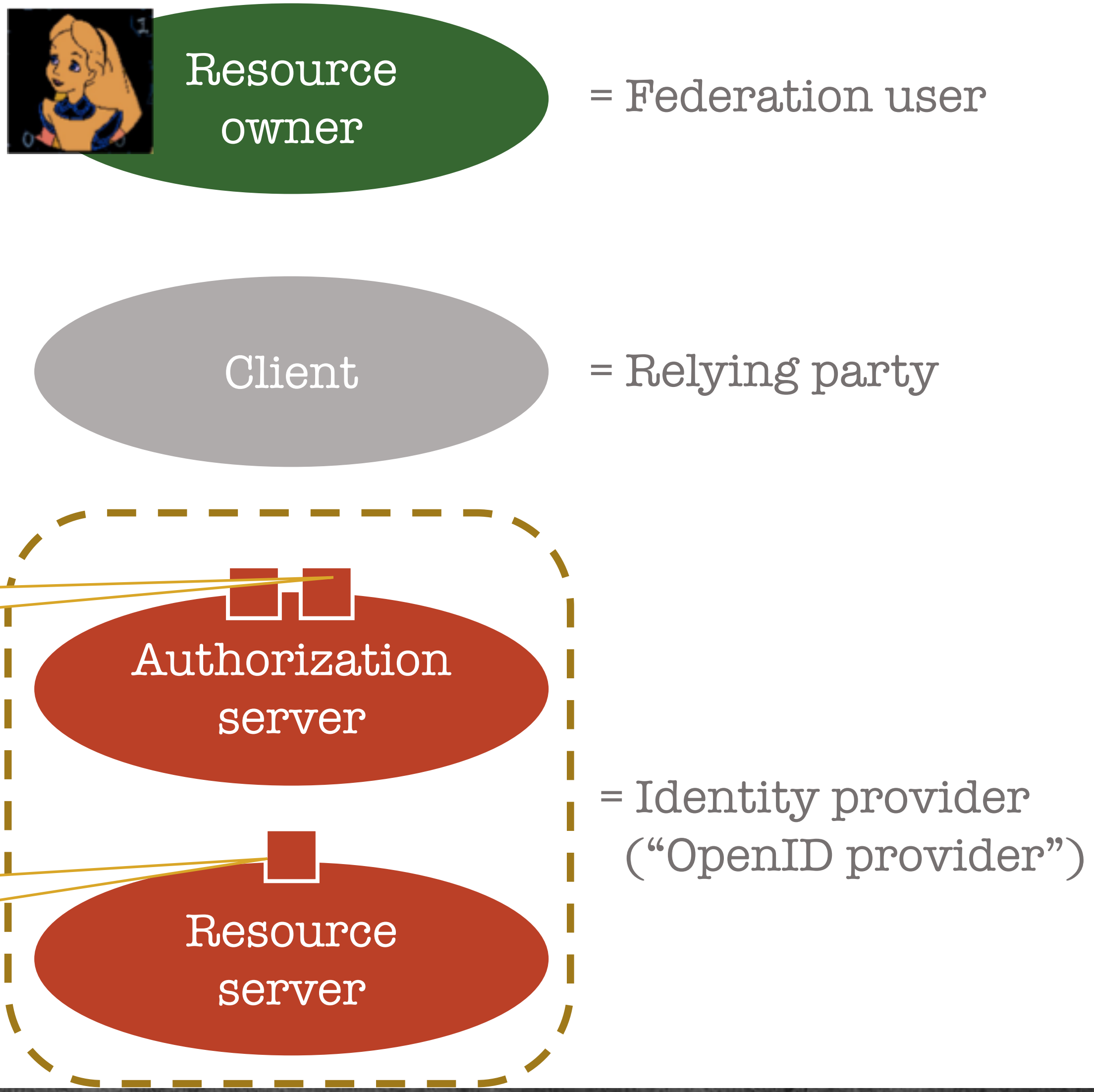
App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

Standard OAuth endpoints for authorization and access token issuance

Some number of API endpoints that deliver the data or other value-add

# OpenID Connect does modern-day federation

## It is an OAuth-protected identity API, plus a bit more

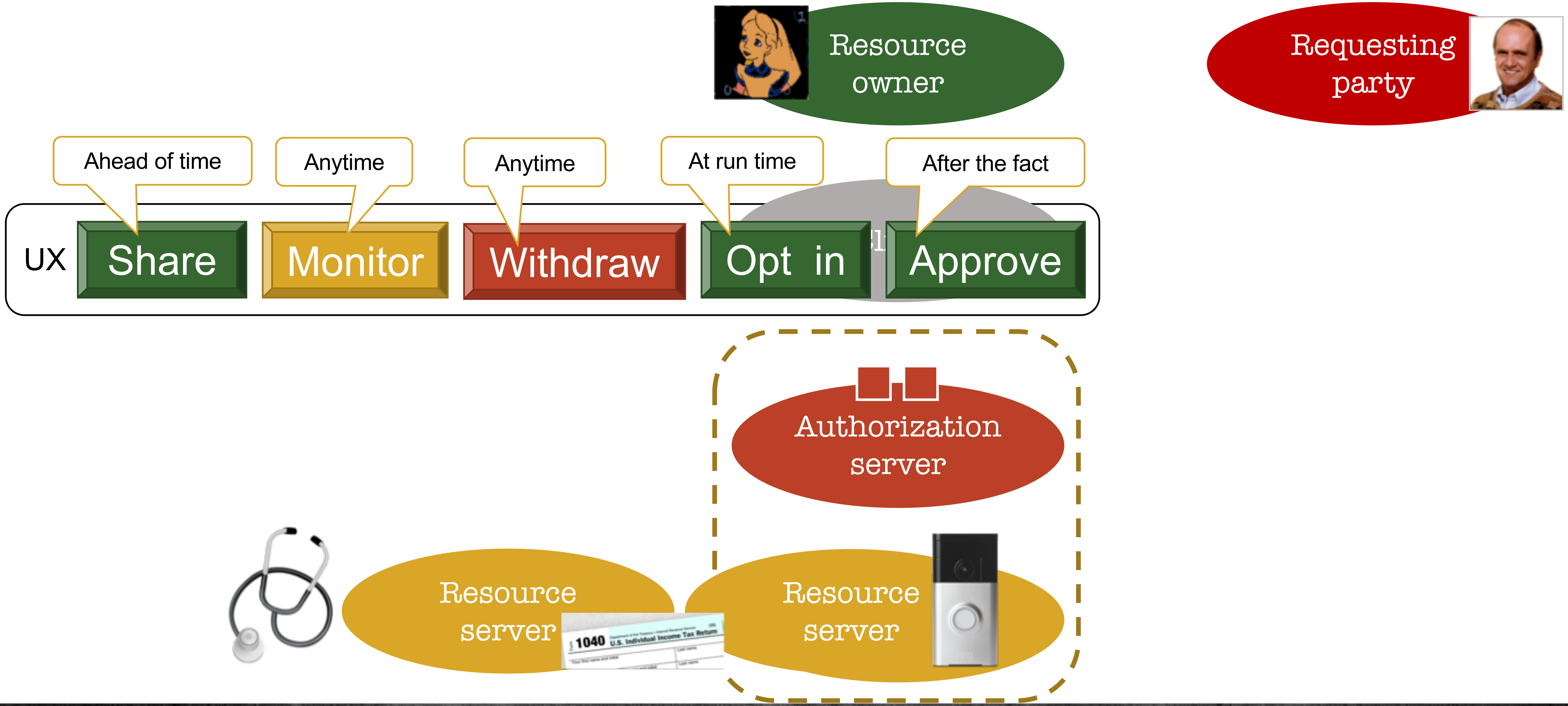


Token endpoint typically delivers an "ID token" similar to a SAML assertion

Standard UserInfo endpoint can be called with an access token to look up identity claims

# User-Managed Access is for cross-party sharing

## UMA brings next-gen delegation and consent to OAuth



# A FinTech use case: Origo solution for the UK Pensions Dashboard project

ORIGO

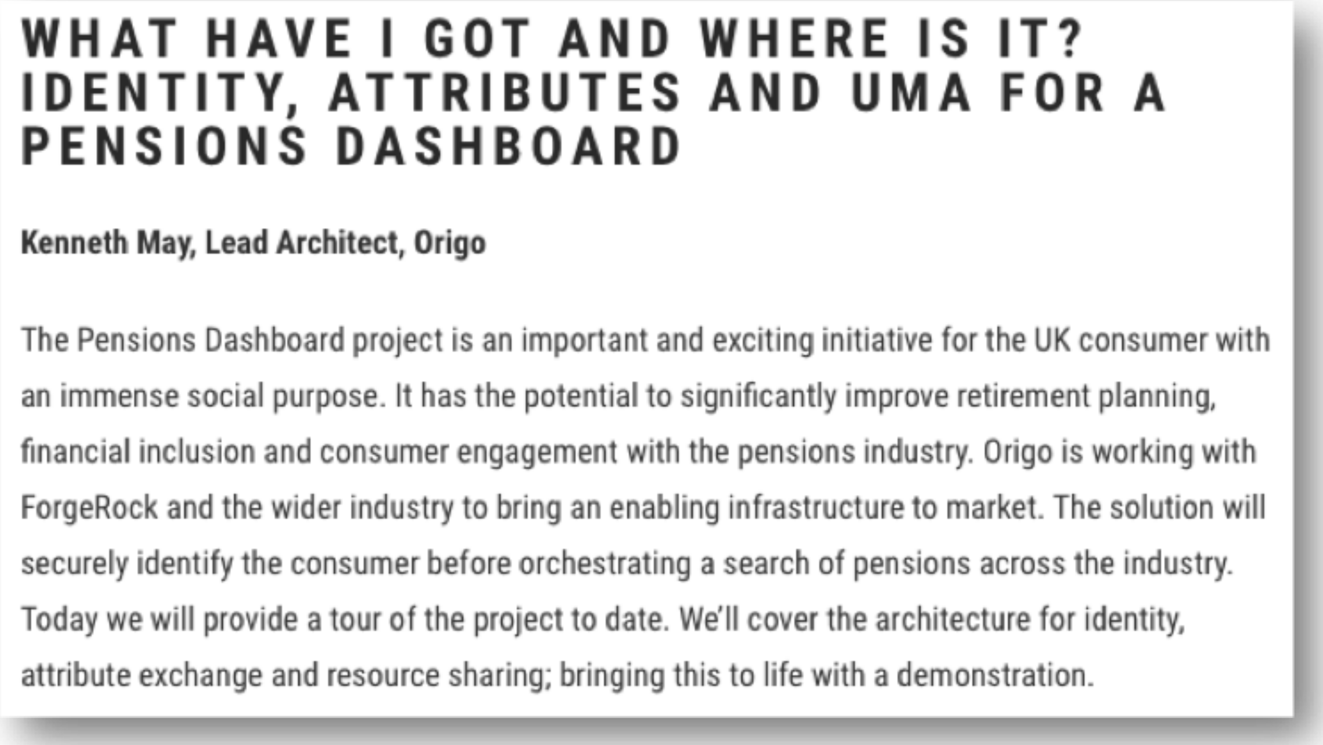
OIX UK Europe

PensionsDashboard Prototype Project

FORGEROCK

https://youtu.be/LjWPyy94NgA

oixuk.org/digital-id-for-pensions-dashboards/



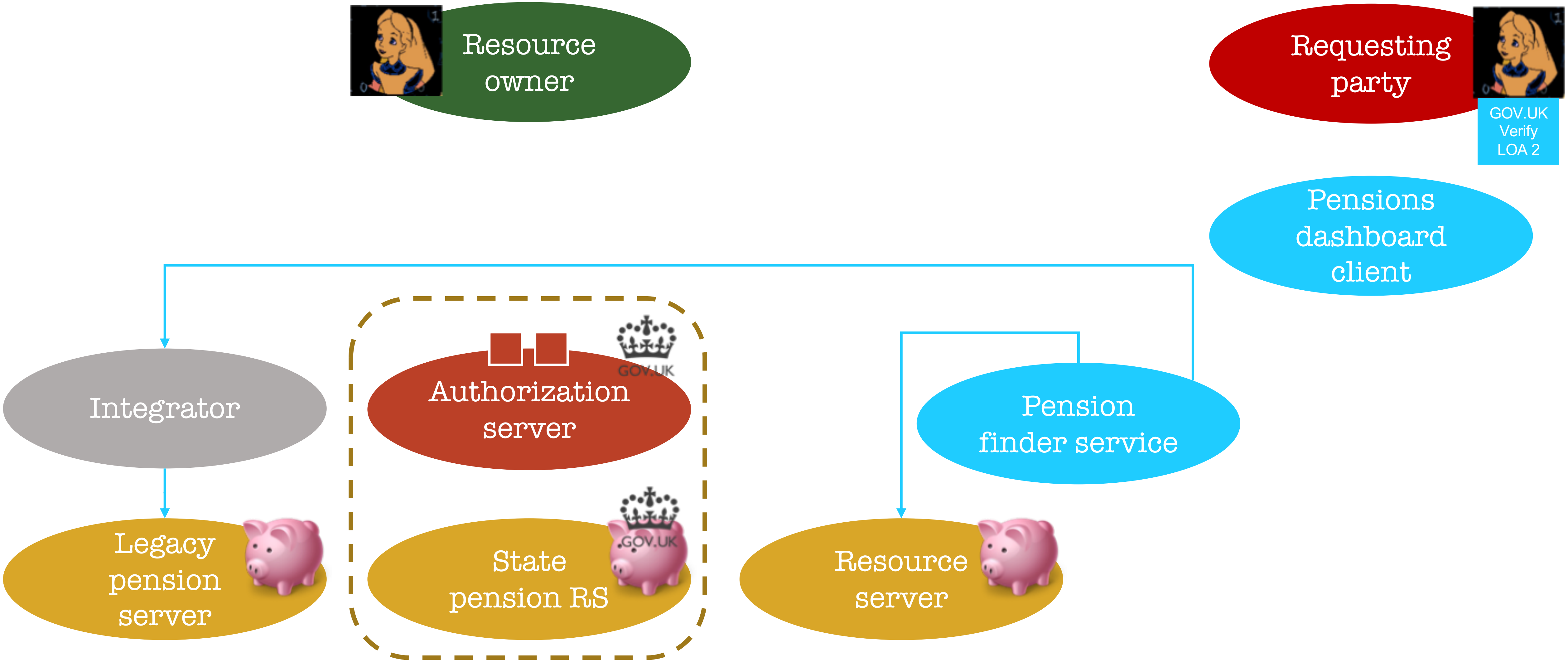
## WHAT HAVE I GOT AND WHERE IS IT? IDENTITY, ATTRIBUTES AND UMA FOR A PENSIONS DASHBOARD

**Kenneth May, Lead Architect, Origo**

The Pensions Dashboard project is an important and exciting initiative for the UK consumer with an immense social purpose. It has the potential to significantly improve retirement planning, financial inclusion and consumer engagement with the pensions industry. Origo is working with ForgeRock and the wider industry to bring an enabling infrastructure to market. The solution will securely identify the consumer before orchestrating a search of pensions across the industry. Today we will provide a tour of the project to date. We'll cover the architecture for identity, attribute exchange and resource sharing; bringing this to life with a demonstration.

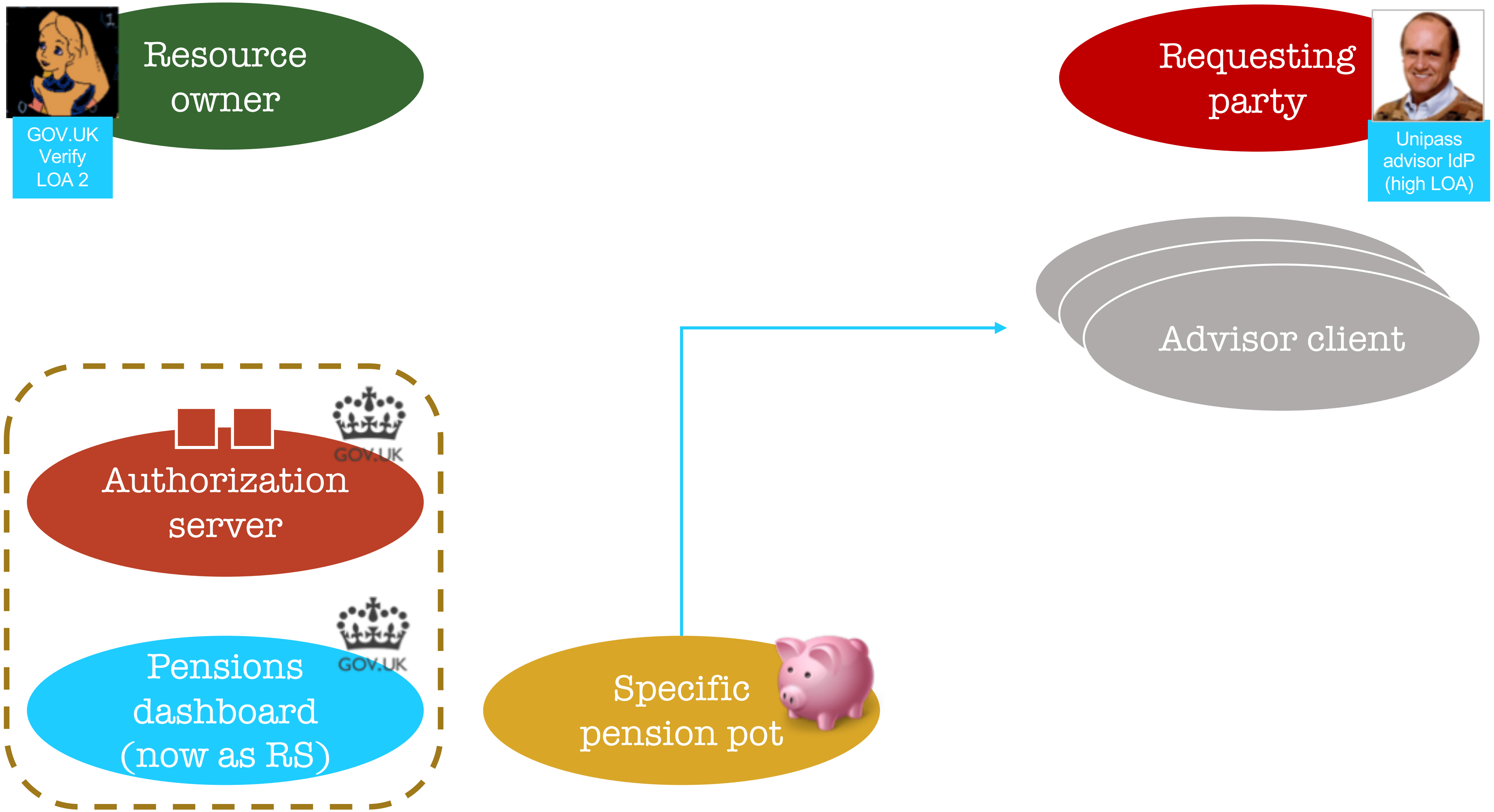
# First, discovery and aggregation of pension pots

## Alice consents to pension pot discovery results with a client *she* uses



# Now she can share with financial advisors

## Permissions are granular and revocable



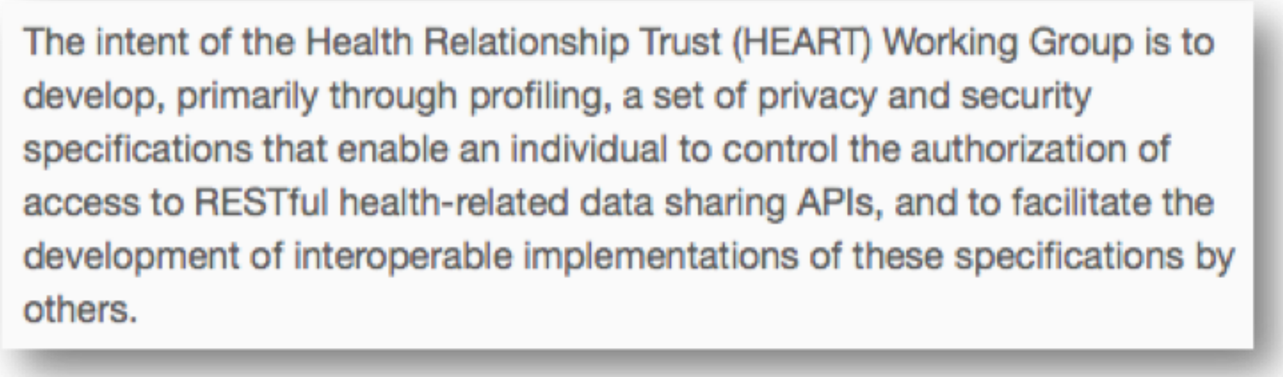
# Health use cases: The HEART Working Group at OpenID Foundation

OpenID®

MHDFC
MOVE HEALTH
DATA FORWARD
CHALLENGE

FHIR®

openid.net/wg/heart/

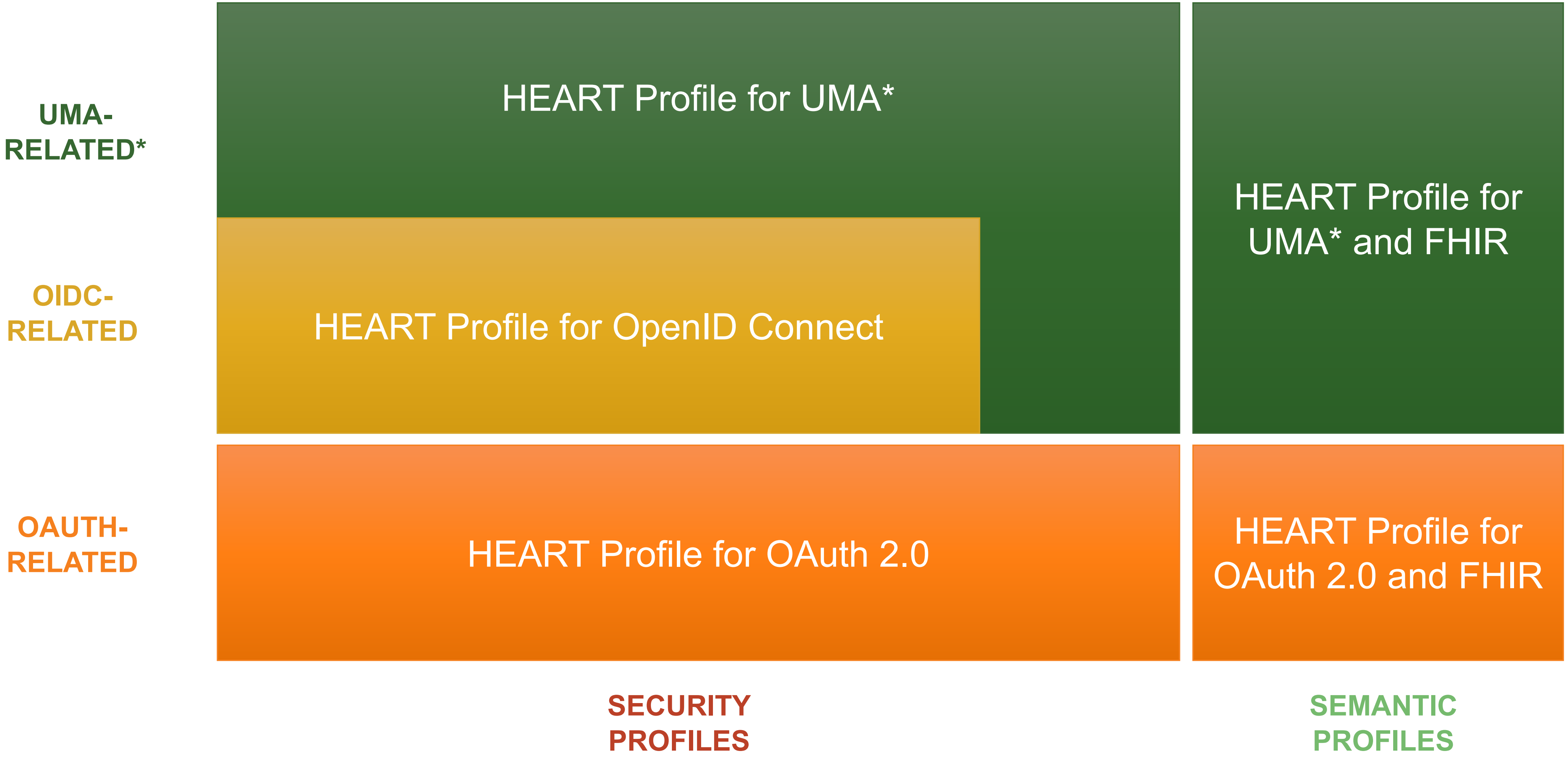


The intent of the Health Relationship Trust (HEART) Working Group is to develop, primarily through profiling, a set of privacy and security specifications that enable an individual to control the authorization of access to RESTful health-related data sharing APIs, and to facilitate the development of interoperable implementations of these specifications by others.

# HEART use cases collected

- Multiple portals
- Virtual patient registration
- Post-myocardial infarction implant and rehab
- VA secure RESTful use case
- Patient data for clinical and research purposes
- Primary care physician first appointment
- Alice selectively shares health-related data with physicians and others

# Deliverables: All are in Implementer’s Draft status


UMA-RELATED*
OIDC-RELATED
OAUTH-RELATED
HEART Profile for UMA*
HEART Profile for OpenID Connect
HEART Profile for OAuth 2.0
HEART Profile for UMA* and FHIR
HEART Profile for OAuth 2.0 and FHIR
SECURITY PROFILES
SEMANTIC PROFILES

# Demonstration

Consumer/clinical health IoT scenario making use of device attestation and Identity Relationship Management

# From the UMA2 Recommendations…

## The UMA extension grant spec enhances OAuth in the following ways:

- The resource owner authorizes protected resource access to clients used by entities that are in a requesting party role. This enables **party-to-party authorization**, rather than authorization of application access alone.
- The authorization server and resource server interact with the client and requesting party in a way that is **asynchronous** with respect to resource owner interactions. This lets a resource owner configure an authorization server with authorization grant rules (policy conditions) at will, rather than authorizing access token issuance synchronously just after authenticating.

## The (optional) federated authorization spec enhances the UMA grant as follows:

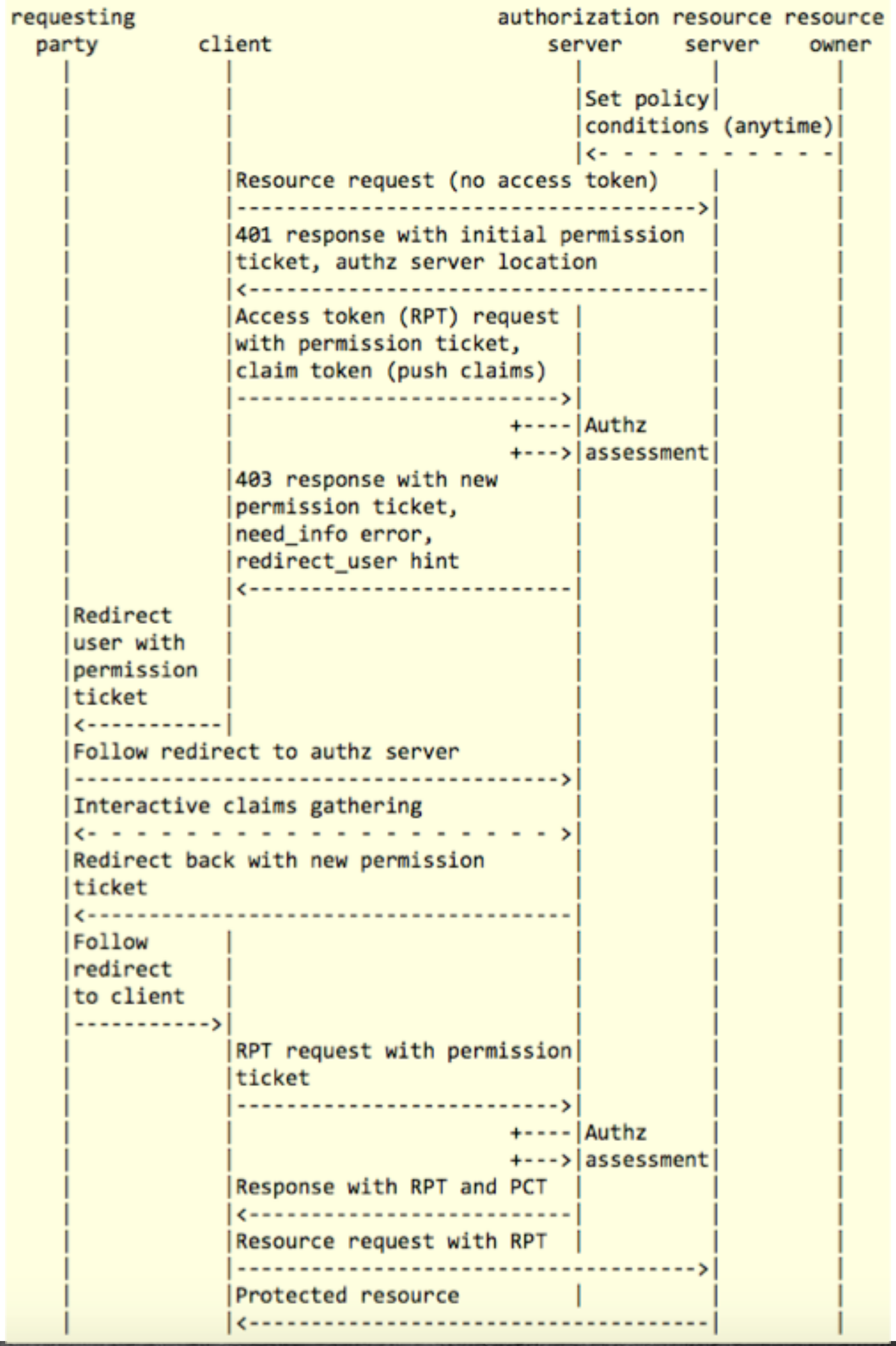
- This specification extends and complements [UMAGrant] to **loosely couple, or federate,** its authorization process.
- This enables **multiple resource servers operating in different domains** to communicate with a single authorization server operating in yet another domain that acts on behalf of a resource owner.
- A service ecosystem can thus **automate resource protection**, and the resource owner can **monitor and control authorization grant rules** through the authorization server over time. Further, authorization grants can **increase and decrease at the level of individual resources and scopes**.

# Sample UMA Grant flow as it appears in the spec

See also:

tinyurl.com/uma2grantwsd

tinyurl.com/uma2fawsd

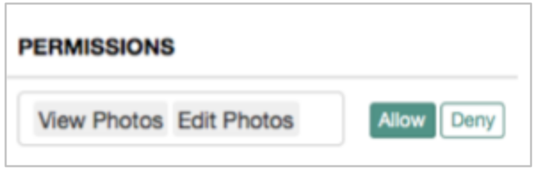


```
requesting                             authorization resource resource
  party        client                      server     server   owner
    |            |                           |          |         |
    |            |                           |Set policy|         |
    |            |                           |conditions (anytime)|
    |            |                           |<- - - - - - - - - -|
    |            |Resource request (no access token)    |         |
    |            |------------------------------------->|         |
    |            |401 response with initial permission  |         |
    |            |ticket, authz server location         |         |
    |            |<-------------------------------------|         |
    |            |Access token (RPT) request |          |         |
    |            |with permission ticket,    |          |         |
    |            |claim token (push claims)  |          |         |
    |            |-------------------------->|          |         |
    |            |                      +----|Authz     |         |
    |            |                      +--->|assessment|         |
    |            |403 response with new      |          |         |
    |            |permission ticket,         |          |         |
    |            |need_info error,           |          |         |
    |            |redirect_user hint         |          |         |
    |            |<--------------------------|          |         |
    |Redirect    |                           |          |         |
    |user with   |                           |          |         |
    |permission  |                           |          |         |
    |ticket      |                           |          |         |
    |<-----------|                           |          |         |
    |Follow redirect to authz server         |          |         |
    |--------------------------------------->|          |         |
    |Interactive claims gathering            |          |         |
    |<- - - - - - - - - - - - - - - - - - - >|          |         |
    |Redirect back with new permission       |          |         |
    |ticket                                  |          |         |
    |<---------------------------------------|          |         |
    |Follow      |                           |          |         |
    |redirect    |                           |          |         |
    |to client   |                           |          |         |
    |----------->|                           |          |         |
    |            |RPT request with permission|          |         |
    |            |ticket                     |          |         |
    |            |-------------------------->|          |         |
    |            |                      +----|Authz     |         |
    |            |                      +--->|assessment|         |
    |            |Response with RPT and PCT  |          |         |
    |            |<--------------------------|          |         |
    |            |Resource request with RPT  |          |         |
    |            |------------------------------------->|         |
    |            |Protected resource         |          |         |
    |            |<-------------------------------------|         |
```

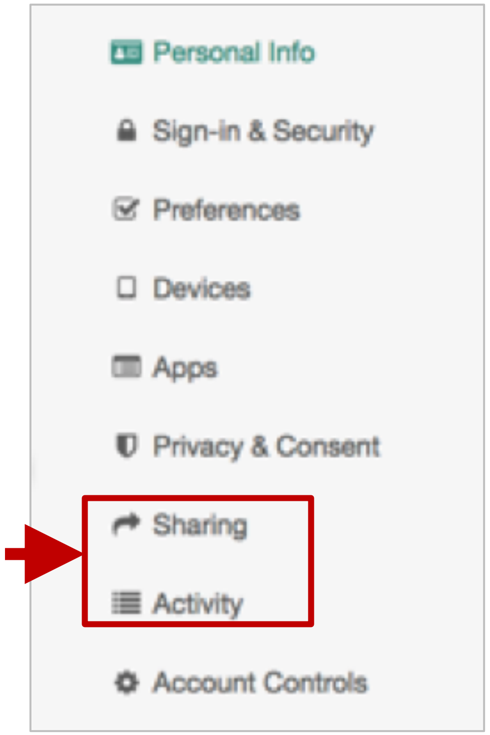
# Key benefits of UMA to consumers

## Constrained party-to-party delegation

## Granting consent without external influence



## Centralized monitoring and management

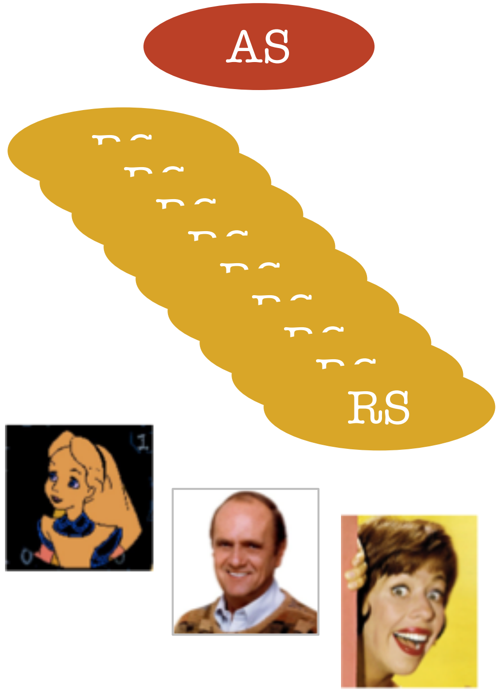


## Control of consents at a fine grain

# Key benefits of UMA to service providers

True security of delegated access

Scalability of resource permissioning



API-first protection strategy

Fosters control for compliance and trust

# The “BLT sandwich”

Putting together a business, legal, and technical framework for UMA usage and deployment

# The UMA Legal effort

tinyurl.com/umalegal

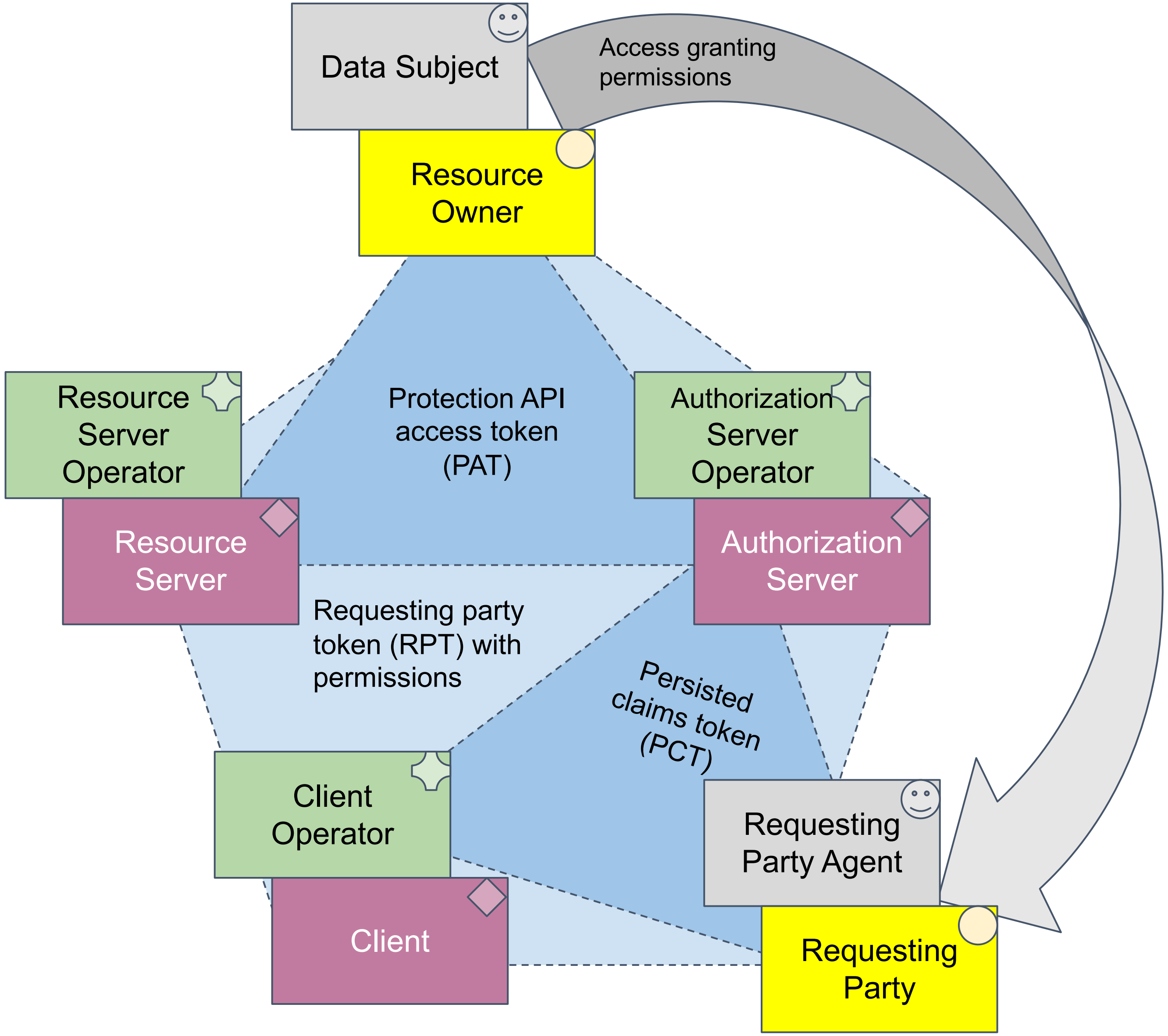
- A subgroup of the UMA Work Group, with expert legal help from Tim Reiniger, has produced a **draft** framework defining:

> How the UMA protocol enables a license-based model for controlling access rights to personal digital assets

- The business tools we (and others) can build from this framework can be single- and cross-jurisdictional, and single- and cross-sector
- Our contention:

> UMA can provide the autonomy, reciprocity, and objectivity to grow market trust in widely sharing access to personal digital assets with devices, apps, and Internet databases.

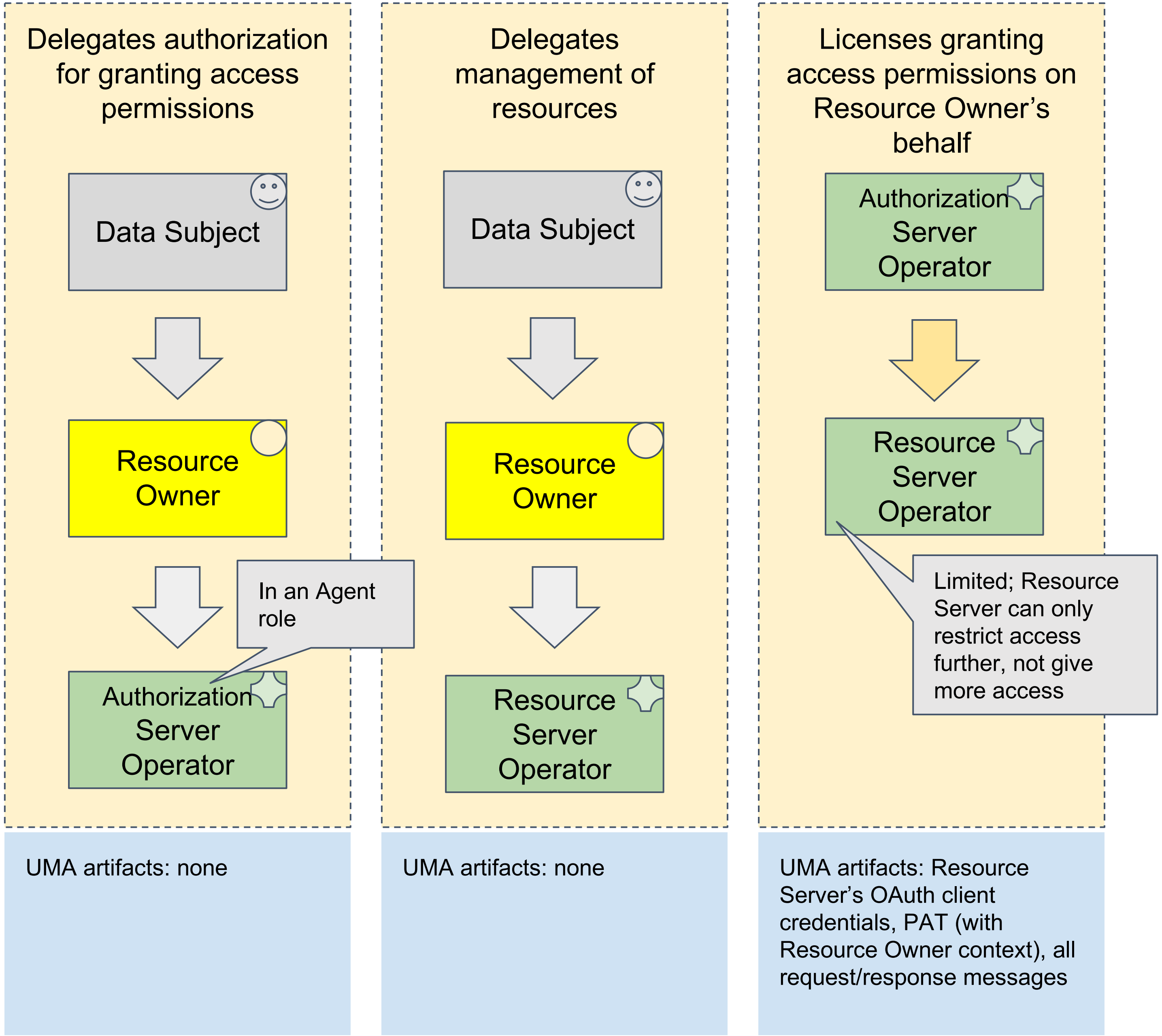
# Legal roles and artifact interactions


Data Subject
Access granting permissions
Resource Owner
Resource Server Operator
Protection API access token (PAT)
Authorization Server Operator
Resource Server
Authorization Server
Requesting party token (RPT) with permissions
Persisted claims token (PCT)
Client Operator
Requesting Party Agent
Client
Requesting Party

**Key:**

- Legal party name exclusively ☺
- Legal party name ✧
- UMA technical entity name ◇
- UMA party/technical entity name ○

# Delegation and licensing: RO-centered



Delegates authorization for granting access permissions

Data Subject

Resource Owner

In an Agent role

Authorization Server Operator

UMA artifacts: none

Delegates management of resources

Data Subject

Resource Owner

Resource Server Operator

UMA artifacts: none

Licenses granting access permissions on Resource Owner's behalf

Authorization Server Operator

Resource Server Operator

Limited; Resource Server can only restrict access further, not give more access

UMA artifacts: Resource Server's OAuth client credentials, PAT (with Resource Owner context), all request/response messages

**Key:**

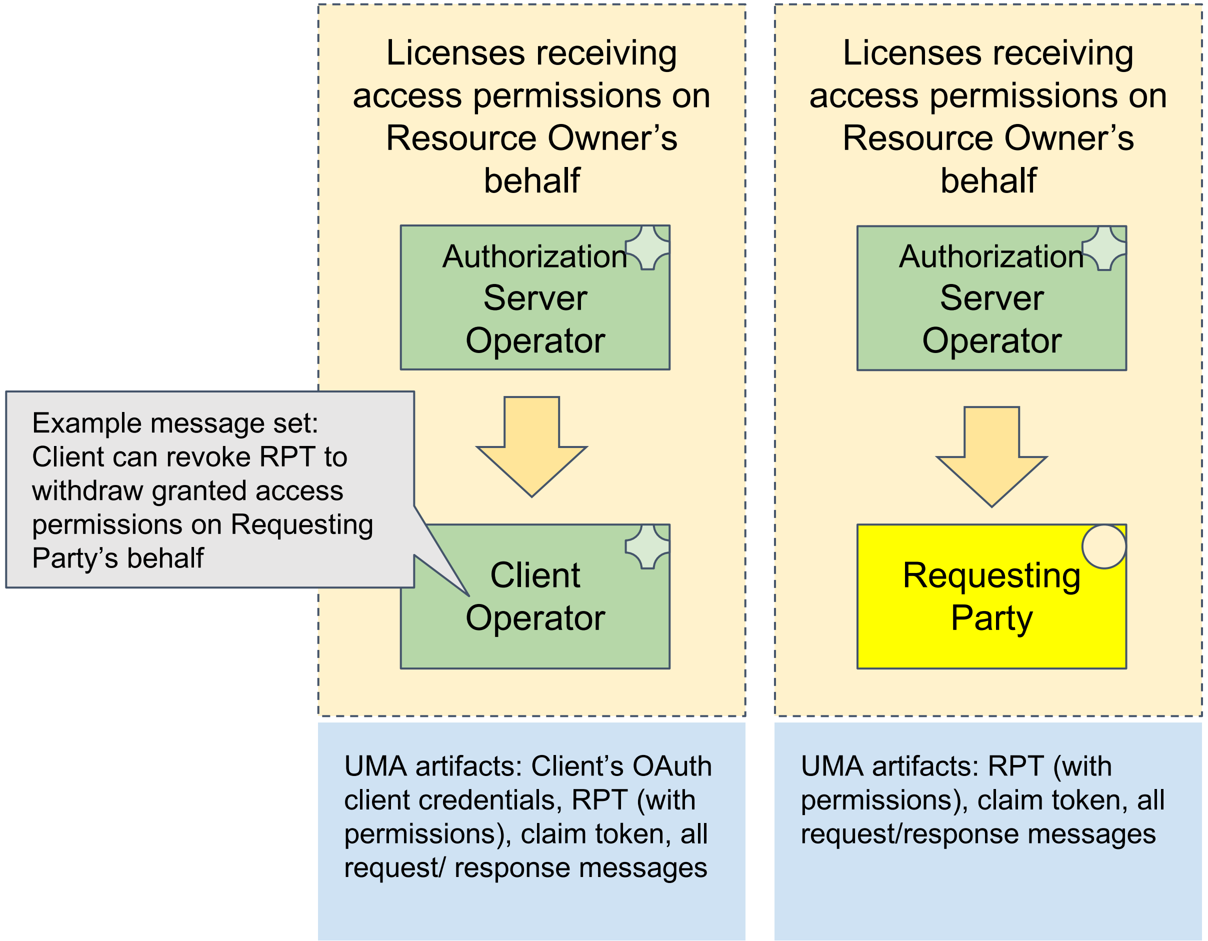
⇩ Legal devices only

⇩ Bound to UMA artifacts

v.2018-01-22a

FORGEROCK

# Delegation and licensing: receiving permissions


Licenses receiving access permissions on Resource Owner's behalf
Authorization Server Operator
Client Operator
Licenses receiving access permissions on Resource Owner's behalf
Authorization Server Operator
Requesting Party

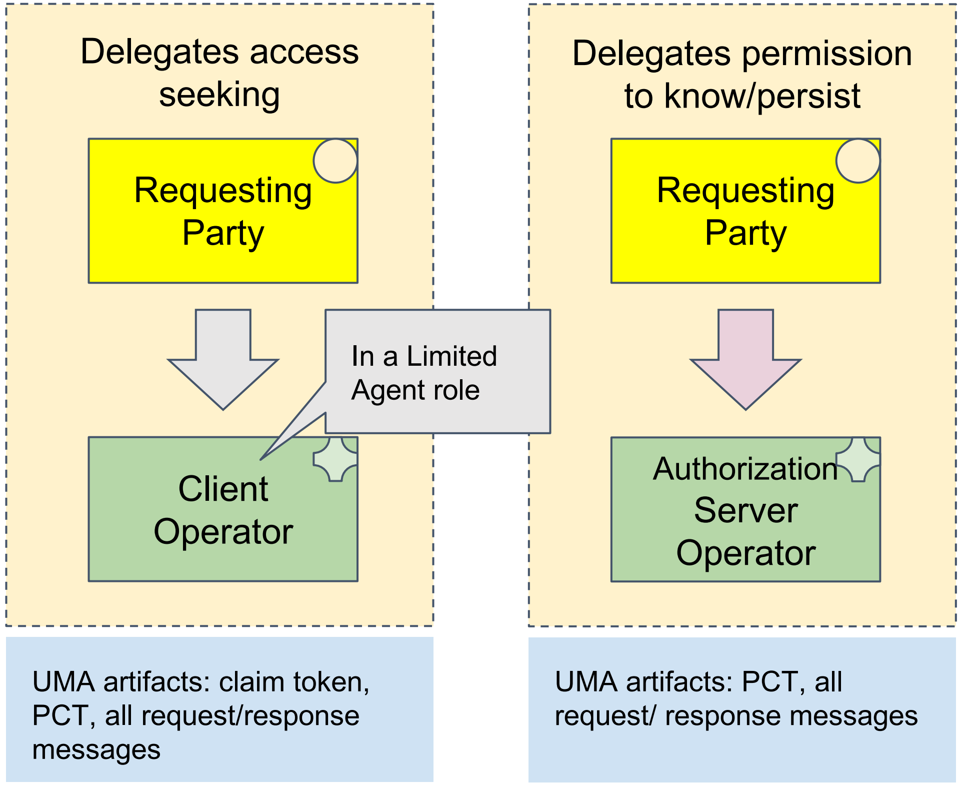
Example message set: Client can revoke RPT to withdraw granted access permissions on Requesting Party's behalf

UMA artifacts: Client's OAuth client credentials, RPT (with permissions), claim token, all request/ response messages

UMA artifacts: RPT (with permissions), claim token, all request/response messages

v.2018-01-22a

# Delegation and licensing: RqP-centered



Delegates access seeking

Requesting Party

In a Limited Agent role

Client Operator

Delegates permission to know/persist

Requesting Party

Authorization Server Operator

UMA artifacts: claim token, PCT, all request/response messages

UMA artifacts: PCT, all request/ response messages

v.2018-01-22a

**Key:**

⇩ Partially bound to UMA artifacts

# To which GDPR articles is UMA most relevant?

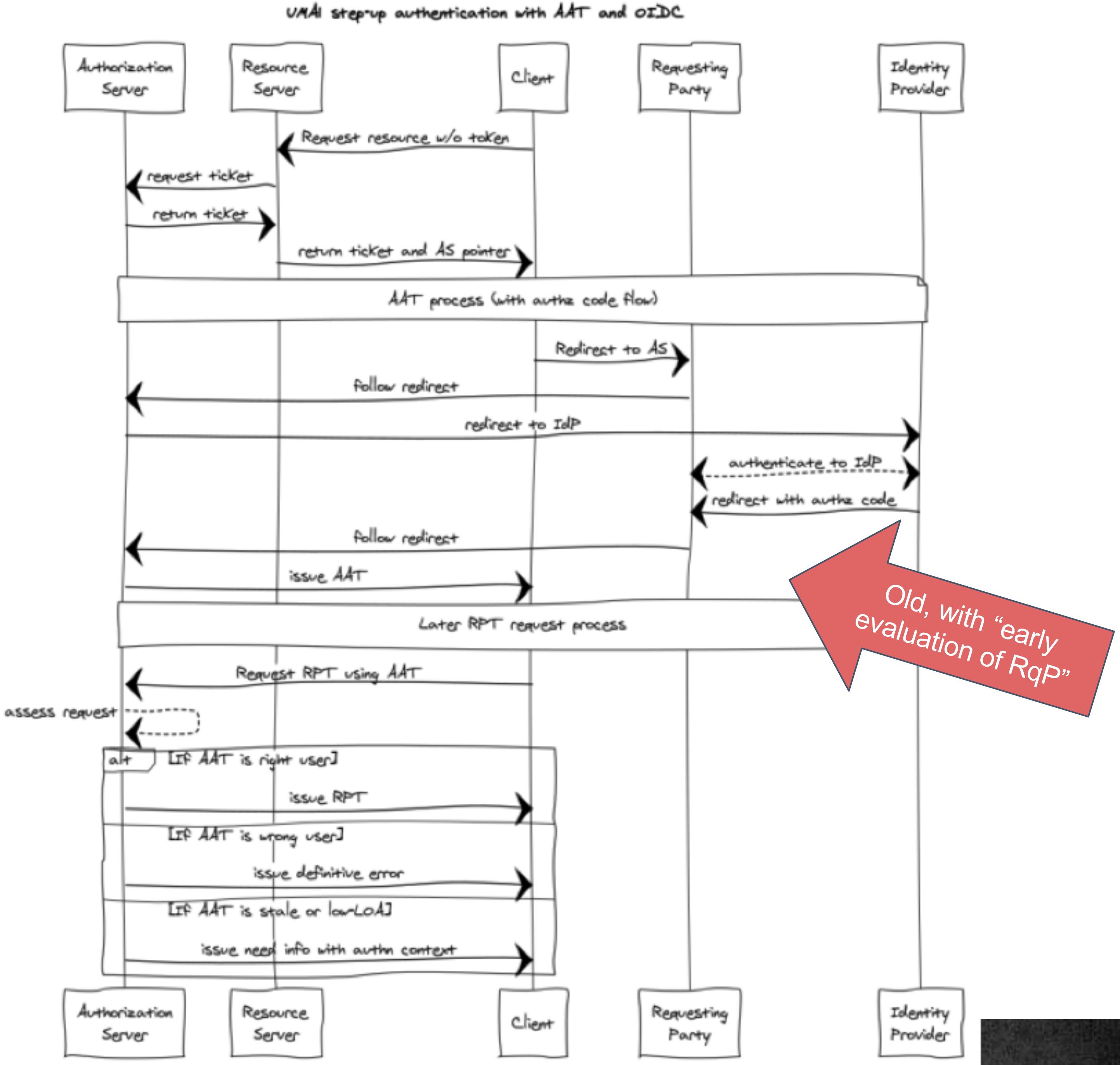
- **Article 5:** Principles relating to personal data processing
- **Article 7:** Conditions for consent
- **Article 8:** Conditions applicable to child's consent in relation to information society services
- **Articles 12-23:** Rights of the data subject
- **Article 25:** Data protection by design and by default
- **Right of access, to restrict processing, to withdraw consent at any time**
- **User-controlled delegation of access**
- **Centralizable management of consents**
- **Transparency of and control over consented access**
- **Consented sharing directly from sources for greater data accuracy**
- **Parent/guardian-to-child constrained delegation**

# What changed from UMA1 to UMA2 and what's coming next

# Key delta: The AAT is gone

- It got in the way of just-in-time wide-ecosystem trust elevation of requesting parties the AS had never met before
  - Though we don’t say “trust elevation” anymore…

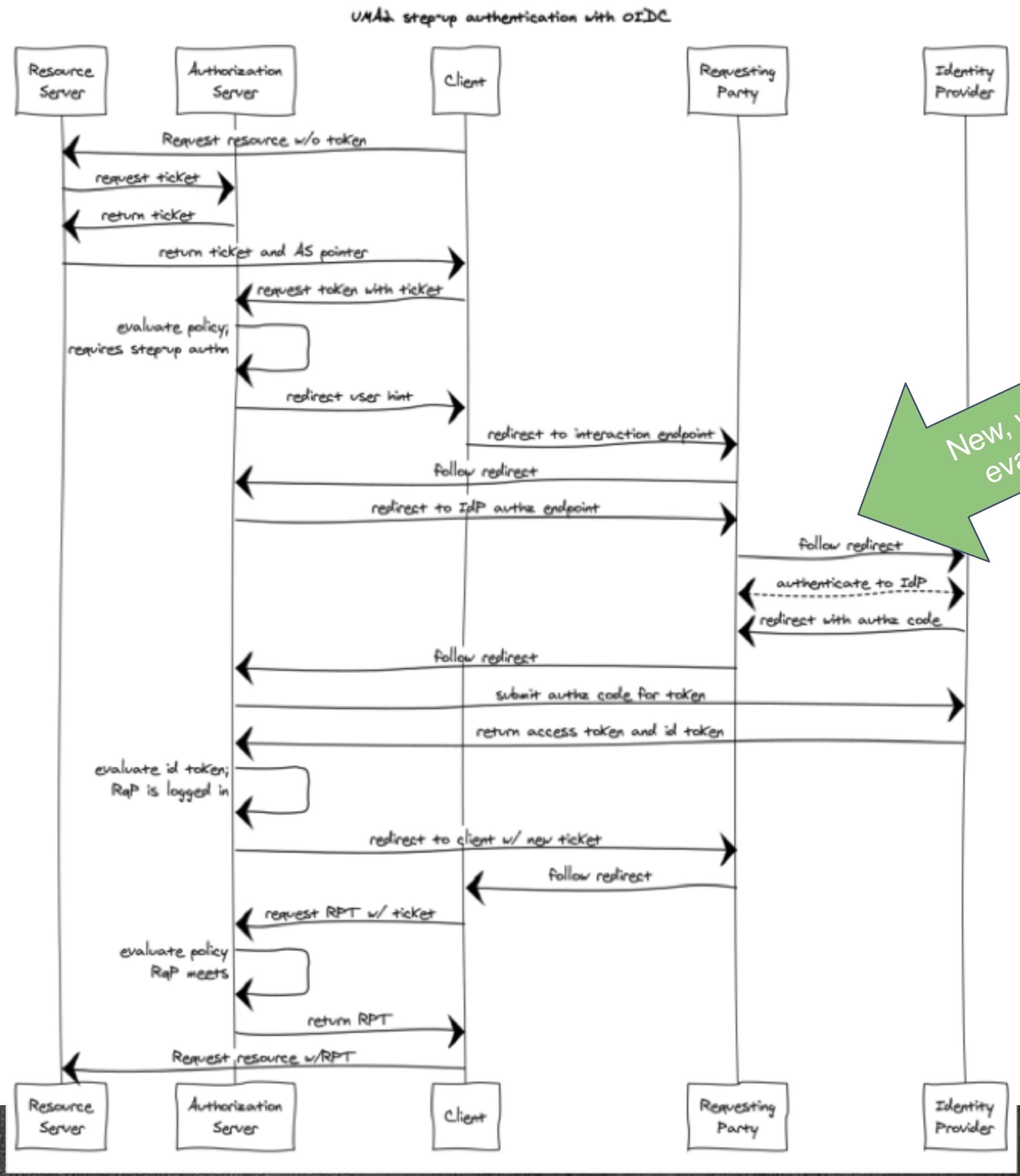
- We removed the requirement to use OAuth for the AAT’s previous job, and now offer an optional PCT (persisted claims token) to capture results of interactive claims gathering, noting that the AS could give the RqP a chance to authorize its issuance



Old, with “early evaluation of RqP”

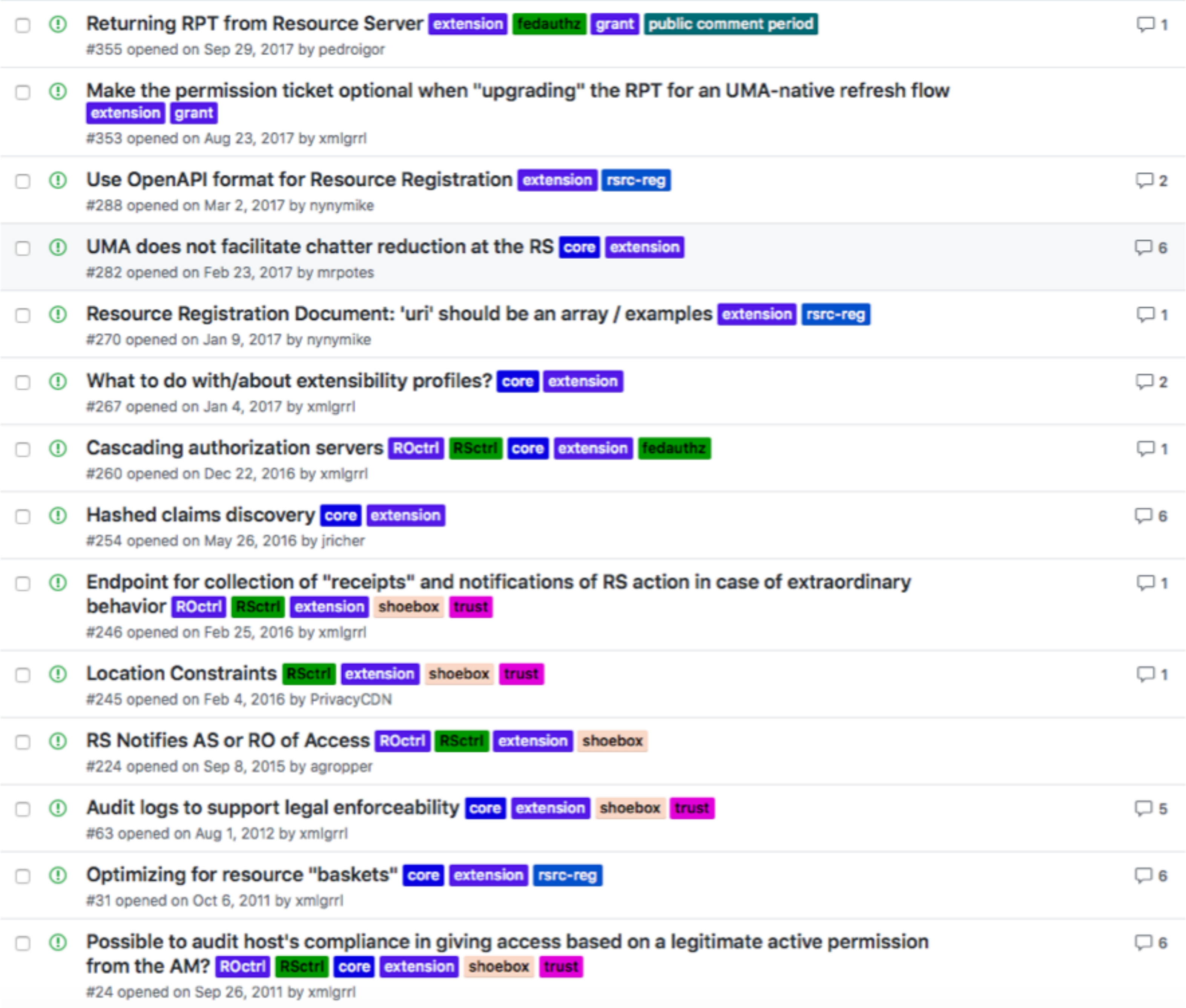
# Key delta: The AAT is gone

- Now, if you’re using a federated login flow for the RqP, the first time they have to do anything, it’s squarely in the realm of interactive claims gathering



New, with “late evaluation”

# GitHub issues with extension ideas



□ Returning RPT from Resource Server extension fedauthz grant public comment period 1
#355 opened on Sep 29, 2017 by pedroigor

□ Make the permission ticket optional when "upgrading" the RPT for an UMA-native refresh flow extension grant
#353 opened on Aug 23, 2017 by xmlgrrl

□ Use OpenAPI format for Resource Registration extension rsrc-reg 2
#288 opened on Mar 2, 2017 by nynymike

□ UMA does not facilitate chatter reduction at the RS core extension 6
#282 opened on Feb 23, 2017 by mrpotes

□ Resource Registration Document: 'uri' should be an array / examples extension rsrc-reg 1
#270 opened on Jan 9, 2017 by nynymike

□ What to do with/about extensibility profiles? core extension 2
#267 opened on Jan 4, 2017 by xmlgrrl

□ Cascading authorization servers ROctrl RSctrl core extension fedauthz 1
#260 opened on Dec 22, 2016 by xmlgrrl

□ Hashed claims discovery core extension 6
#254 opened on May 26, 2016 by jricher

□ Endpoint for collection of "receipts" and notifications of RS action in case of extraordinary behavior ROctrl RSctrl extension shoebox trust 1
#246 opened on Feb 25, 2016 by xmlgrrl

□ Location Constraints RSctrl extension shoebox trust 1
#245 opened on Feb 4, 2016 by PrivacyCDN

□ RS Notifies AS or RO of Access ROctrl RSctrl extension shoebox
#224 opened on Sep 8, 2015 by agropper

□ Audit logs to support legal enforceability core extension shoebox trust 5
#63 opened on Aug 1, 2012 by xmlgrrl

□ Optimizing for resource "baskets" core extension rsrc-reg 6
#31 opened on Oct 6, 2011 by xmlgrrl

□ Possible to audit host's compliance in giving access based on a legitimate active permission from the AM? ROctrl RSctrl core extension shoebox trust 6
#24 opened on Sep 26, 2011 by xmlgrrl

FORGEROCK

# Thank you!
# Questions?
# Join us!

Eve Maler
VP Innovation & Emerging Technology, ForgeRock
@xmlgrrl | eve.maler@forgerock.com
Chair and founder, Kantara UMA Work Group
@UMAWG | tinyurl.com/umawg

24 Jan 2018