UK Pensions Dashboards Architecture

Design Document

to accompany draft UMA profiles

Version 1.0

November 2019

Table of Contents

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

1. Document History

This design document was created by Origo Services Ltd (Origo). Contact Kenneth May (kenneth.may@origo.com)

Version 1.0, dated November 2019 was authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo from March – Sept 2019.

This design document and the corresponding draft profile are outputs from Origo's extensive work on the UK Pensions Dashboard initiative since 2014. Origo widely demonstrated a solution based on UMA version 1 in 2016 and 2017 before undertaking further work to develop an UMA2 based profile.

In December 2020 this design document and the corresponding draft profile was contributed to the Kantara Initiative's UMA Working Group by Origo.

2. Introduction

This document presents the design of a UK Pension Dashboard digital architecture in so far as it is needed for review of the Pension Dashboard Profile of UMA. The expected audience is those who read or review the UMA profiles for technical reasons: correction, suggested improvement, technical enhancement or implementation. Accordingly, the document assumes knowledge of UMA 2 and concentrates on the domain of the use case and design decisions which shape the profile.

The work to produce this design document and accompanying draft profiles has been undertaken, and outputs produced, with the intention of the Pensions Dashboard Programme (PDP) using the outputs to accelerate the delivery of its programme to benefit millions of citizens.

It will be proposed that the draft UMA profiles (technical standards) are an excellent fit with the requirements outlined by the DWP consultation response in April 2019. They will also help accelerate progress for industry and government to deliver the digital architecture for the UK pensions dashboard ecosystem.

3. Background and Overview

In the UK individuals can acquire several pension assets in addition to state pension as they change employment or make differing investment decisions throughout their working life. Many people lose touch with their providers and their assets. Overall population engagement with pensions is considered to be inadequate.

Pensions Dashboard is an initiative of the UK Government (specifically Department for Work and Pensions, DWP) to enable persons to find their pensions and to increase engagement with them and with financial retirement planning.

The objective of the Pensions Dashboard is to enable persons who own pension assets to find them, and for the pension owner or their advisers (i.e. guidance bodies or independent financial advisers) to value their assets and present them digitally in a 'Pensions Dashboard'. It is envisaged that there will be an eco-system of potentially many providers of Pensions Dashboards, all of which use a common infrastructure to authenticate the owner, manage consents and to find the pensions.

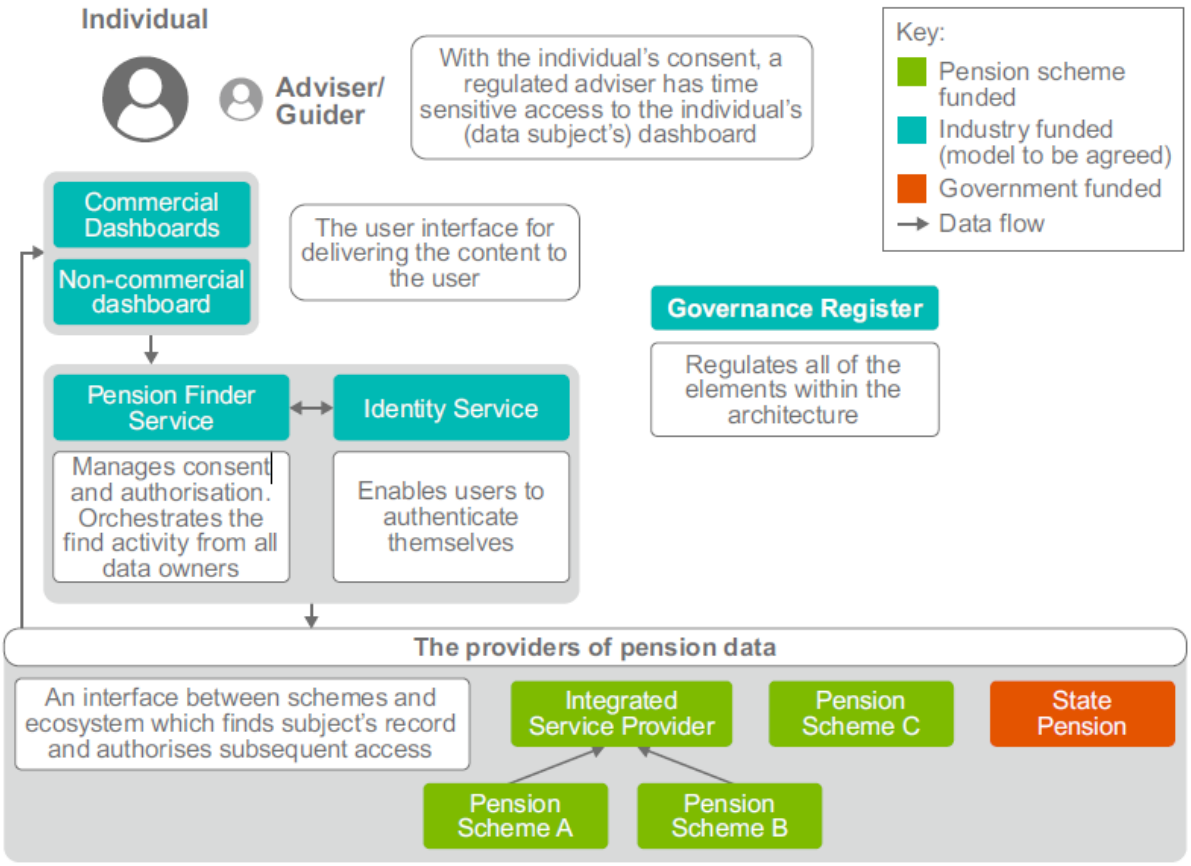
The 'common infrastructure' consists of a 'Pension Finder Service' (PFS), and associated Authorisation Server (or Servers) and appropriate Federated Identity Providers for both pension owners and for their

advisers. The PFS will orchestrate a search over the entire pensions industry – expected to be several hundred connection points – at which pension providers interface with the Pension Dashboard eco-system.

Once the user's pensions are found, unique dereferenceable identifiers for the pension resources are provided to the user's Pensions Dashboard. Under user control, the dashboard can then dereference these, the user can appropriately authorise access to the pension resources, and the related valuation and other information presented in the dashboard. The owner can also send (or have sent on her behalf) these identifiers to her adviser(s).

The Department for Work and Pensions published a consultation and response[1] covering many policy, governance and technical issues. This included a 'candidate architecture' based on User Managed Access (UMA 2) as the proposed open standard for protection and authorisation of access to pension resources for the eco-system.

The DWP report included the following diagram. This design document enhances this conceptual design, emphasising those aspects which are relevant to the Pensions Dashboard UMA profile.



[1] https://www.gov.uk/government/consultations/pensions-dashboards-feasibility-report-and-consultation

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

4. Characteristics of the Problem Domain

This section highlights those aspects of the problem domain which are relevant to the design shown in this document.

Domain Characteristic	Solution Space	Comment
Pension Owner is the user of the Find service, and the agent of the decision to put found resources under authorisation control.	The Pension Owner needs to give consent to the PFS for search. Accordingly, PFS needs to have standard identity authentication and access to assured attributes for search. **The pension owner is the UMA Resource Owne**r when choosing to put the found resources under UMA protection. The Resource Owner needs to set access policy both for her pension dashboard and for her advisers.	Advisers cannot perform 'Find' operations. Find is not an UMA operation, but putting 'found' resources under UMA protection is. Resource Owner 'Policy' is simply the delegation of read access (i.e. valuation of pension asset) to the user (or delegate) as Requesting Party.
Pension Dashboards are free to choose whatever user authentication mechanism it requires (since dashboard software may be added to existing financial service portals)	The pension owner, as UMA Requesting Party, is authenticated by the dashboard to an uncertain level of assurance, so the authorisation process (applying the Resource Owner's policy) needs to assure itself of the Requesting Party's identity to a standard level.	This profile uses need_info to request redirection so the UMA AS can further redirect to a federated Identity Service. This profile uses a PCT to persist the relationship between the assured RO owner identity and the (same) lower assurance identity of the dashboard user as Requesting Party. The PCT makes the user journey easier for subsequent accesses.
Governance of eco-system requires a standard identity for all participants so that the consent, find, and identity assurance risk is standard	A federated Identity Service is envisaged which asserts standard identities with standard attributes, so the information assurance risks associated with identity are known and common to all.	The is substantial debate over the solution space for the ID federation. 'Verify' – a UK gov scheme – is a possible contender when transferred to the private sector. (The DWP's consultation report reiterated the need for standards based identities and currently these are only issued by 'Verify' Identity Providers. The UK government is in the process of opening the market for private sector applications of Verify identities. Pensions Dashboard could well be a major application of these.
Pension Owner has ability to delegate access to advisers for each specific pension assets	The Pension Owner, as assured Resource Owner, can set policy enabling delegation for each separate asset.	This is required functionality of the UMA AS. It is expected that advisers will have read access to pension resources.
Pension Owner can revoke access without recourse to a dashboard operator	The Resource Owner needs to modify/revoke access policy both for her pension dashboard and for her advisers.	This is required functionality of the UMA AS.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

Domain Characteristic	Solution Space	Comment
Advisers have professional accreditation or are part of a government body of guidance providers	The identity and professional accreditation status of advisers and guidance personnel will be provided by a specialised ID Service.	The solution space for advisers will need to be heavily governed. There are candidate solutions already in operation in the UK IFA marketplace.
Advisers use software which may have non-standard methods of user (adviser) authentication.	The UMA Requesting Party will need to be 'stepped up' in a directly analogous manner to that of the Pension Owner.	PCT also used here.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

5. High Level Design - Components

The key components and their responsibilities are in the following table.

Component	Responsibility	Notes
Pensions Dashboard	Initiate Find, persist pension asset identifiers, cooperate with Requesting Party authorisation protocol, access (value) pension assets, persist access tokens	Currently governance rules prevent dashboards from persisting or processing pension asset data for any other purpose than viewing by the specific (human) requesting party.
Pension Finder Service (PFS)	Authenticate user to a standard level (by means of the UMA AS), gather consent for Find, enable the user to set/modify policy (by means of the UMA AS), orchestrate Find and resource protection, return resource identifiers to the dashboard.	Tangential to the UMA Profile, mentioned in this document only in so far as is relevant to the UMA Profile. Note that the PFS orchestrates find activity, the actual location of the pension assets is managed by the Pension Provider.
UMA Authorisation Server (UMA AS)	Step-up authentication for users of Find, for proof of Resource Ownership, for proof of Adviser identity and status. Manage and apply RO policy for authorisation, coordinate authorisation protocol with dashboards.	Step-up and Persistence in UMA authorisation is covered in this design for the UMA profile. UMA profile covers specifics of UMA authorisation.
Federated Identity Service	Provide definitive identity proofing and authentication services for pension owners and for advisers.	Tangential to the UMA Profile, mentioned in this document only in so far as is relevant to the UMA Profile.
Pension Provider (Find)	Present standard API to PFS to identify pension assets owned by the Pension Owner.	Tangential to the UMA Profile, mentioned in this document only in so far as is relevant to the UMA Profile.
Pension Provider (UMA Resource Server)	Place found assets under UMA protection; renew protection tokens as required. Coordinate authorisation protocol with UMA AS and Dashboard.	UMA Protection and UMA Authorisation covered in this design for the UMA Profile.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

6. High Level Design –Service Processes

Pension Owners have two distinct processes when using the service.

1. **Consent and Find**
   The user of a Pension Dashboard will seek to 'Find' her pension assets. As Pension Owner, she will need to prove her identity to a suitable level of assurance (acceptable to the eco-system as a whole). As Resource Owner, she will consent to the find and to placing resources under protection. She will set policy for her access and optionally set policy for her adviser(s). Her dashboard will persist unique dereferenceable identifiers for her pensions;

2. **Authorise and Value**
   The user of the Pension Dashboard will seek to 'Value' her pension assets. As Requesting Party, she will be authenticated to the dashboard, but will also need to (re)prove that she is the Resource Owner. Having met the conditions (of her own policy) for authorisation, the Pension Provider, acting as Resource Server, will take its access control decision and serve the value of the pension associated with the resource to the requesting party's dashboard. Her dashboard will persist tokens which may decrease the friction of her subsequent accesses.

Advisers can only use the second process above, acting as a requesting party, subject to delegation by the pension owner.

An adviser uses pension asset identifiers to access the related pension assets. The resource identifiers will have been provided either directly by the pension owner, or at the pension owner's instruction when she established her policy of delegation to that adviser. The adviser will be authenticated to his dashboard but will need to (re)prove that he is the adviser to whom the resource owner delegated access. Thus, having met the conditions of the Resource Owner's policy, the Resource Server will take its access control decision and serve the value of the pension owner's pensions to the adviser's (i.e. the requesting party's) dashboard. The dashboard will persist tokens which decrease the friction of the adviser's subsequent accesses.

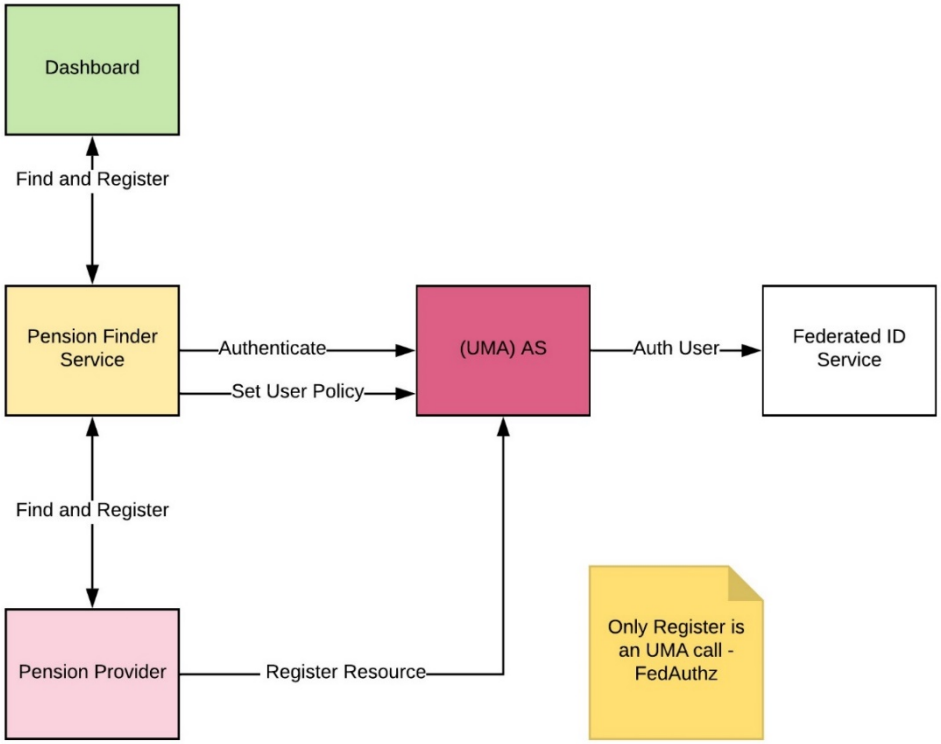
Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

7.Component Collaboration

The components which cooperate to deliver the above functionality are presented in the next two sections. The diagrams show intercomponent interactions: some will be via API, some by browser redirection. All components (other than the Pension Provider) will have a User Interface.

6.1 Consent and Find

The '**Consent and Find**' components are show in the following diagram.



A sequence diagram is presented later covering both non-UMA and UMA flow.

6.1.1 PAT issuance and Resource Registration

If a pension asset is found, and the pension owner has consented[2], the pension asset is registered with the UMA AS (i.e. is placed under UMA protection) by the Pension Provider Resource Server.

This raises the pre-requisite UMA-related issue of how does the Resource Owner 'chose the UMA AS' for federated authorisation at that AS by her Pension Provider Resource Server (RS) and thus arrange for the PAT which enables this registration.

Since the RS and the AS are in a secure eco-system and the RO is authenticated to the AS at the time of the resource discovery[3], the RS can use temporary credentials specific to the RO at the AS to obtain the PAT. It

[2] Consent to find and consent to protect are logically separate consents. Clearly the latter is a pre-requisite of the use of UMA to subsequently access the asset. Here we assume that such consent is granted.

[3] The secure eco-system includes the RS being issued with governed certs from the private PKI operated by the governance register, the RS client will be authenticated (probably using a cert) to the AS, the transport will be MTLS protected and the PATs when issued will be OAuth mTLS Bound.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

is proposed that the AS will issue an authorisation token for use by the RS for grant type `"urn:ietf:params:oauth:granttype:jwt-bearer"` as in [RFC7523] Section 2.1[4]

6.1.2 Policy

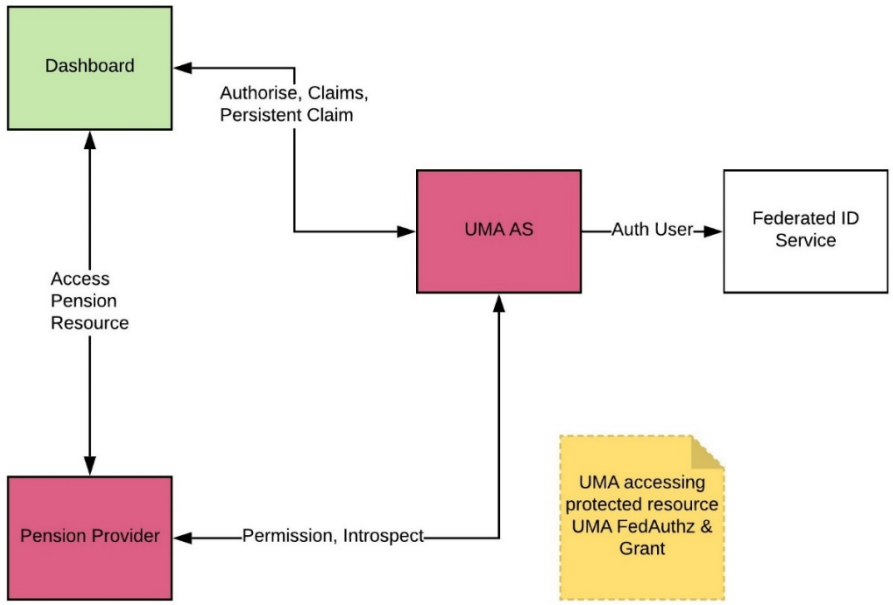
The Pension Owner, acting as Resource Owner can establish her policy which controls her own access and that of her delegates.

The Policy is at least one instance of this policy template:

- *<Pension Owner> grants <scope="value"> to <Requesting Party> of <role="owner" xor "delegate"> at <specific identified Pension Dashboard> to <list of Pension resources> until <time limit>*
- *Any such policy statement may be revoked by the Pension Owner before the time limit*

6.2 Authorise and Access

The '**Authorise and Access**' components are show in the following diagram.



6.2.1 Arranging Access Parameters

The initiating attempt (from dashboard to pension provider) to access a pension resource will carry sufficient information:

- *An identifier for the 'Pension Owner at the RS'*
- *An identifier for the 'pension resource' owned by the Pension Owner at the RS which is being accessed*
- *The type of requesting party[5] (owner or delegate)*

[4] Implementation consideration. See RFC7521 4.1 re lifetime of a resulting token should not exceed that of the grant assertion (i.e. of the AS issued JWT). In the AS will issue the JWT representing a grant, i.e. 'temporary credentials', and this is exchanged for the PAT. The lifetime of the PAT needs to be very long since it is required for permission tickets etc. In this application the stipulation of RFC7521 4.1 should be ignored as other mitigations will be in place to protect and to refresh the PAT.

[5] This is necessary so that the RS can request the appropriate access scopes in its permission ticket request.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

The assumed design[6] of the 'unique dereferenceable identifier' for each pension asset[7] is of the following form; and the access can carry a query parameter asserting the nature of the requesting party user:

- *<pension-provider-resource-server>/Customer/<ALICEUUID>/Benefit/<PENSIONASSETUUID>*
- *?user=owner (the default if absent) or ?user=delegate (for advisers or guidance staff)*

6.2.2 Data needed for Authorisation

The initial access request determines the specifics of which authorisation is requested:

- *The resource server identifier (at which the access was attempted)*
- *The registered resource identifier (issued at the AS, stored at the RS)*
- *The requested permissions (determined by the RS, derived from the access attempt, in this design this will be 'value' plus either 'owner' or 'delegate')*
- *Resource Owner identifier at the RS*

The claims used by the AS to make authorisation decisions against this policy for a specific pension resource are:

- *Identity of requesting party (at specific identified pension dashboard)*
- *Identified pension dashboard (making access request)*
- *Identity of requesting party at AS (using federated identity service)*
- *Trusted professional status of the requesting party at AS*
- *Trusted status of requesting party (owner or delegate)*
- *PCT representing the association of Identity of requesting party at PD and at AS, specific identified dashboard, status of requesting party*

These claims are gathered during the UMA protocol a sequence diagram is presented later.

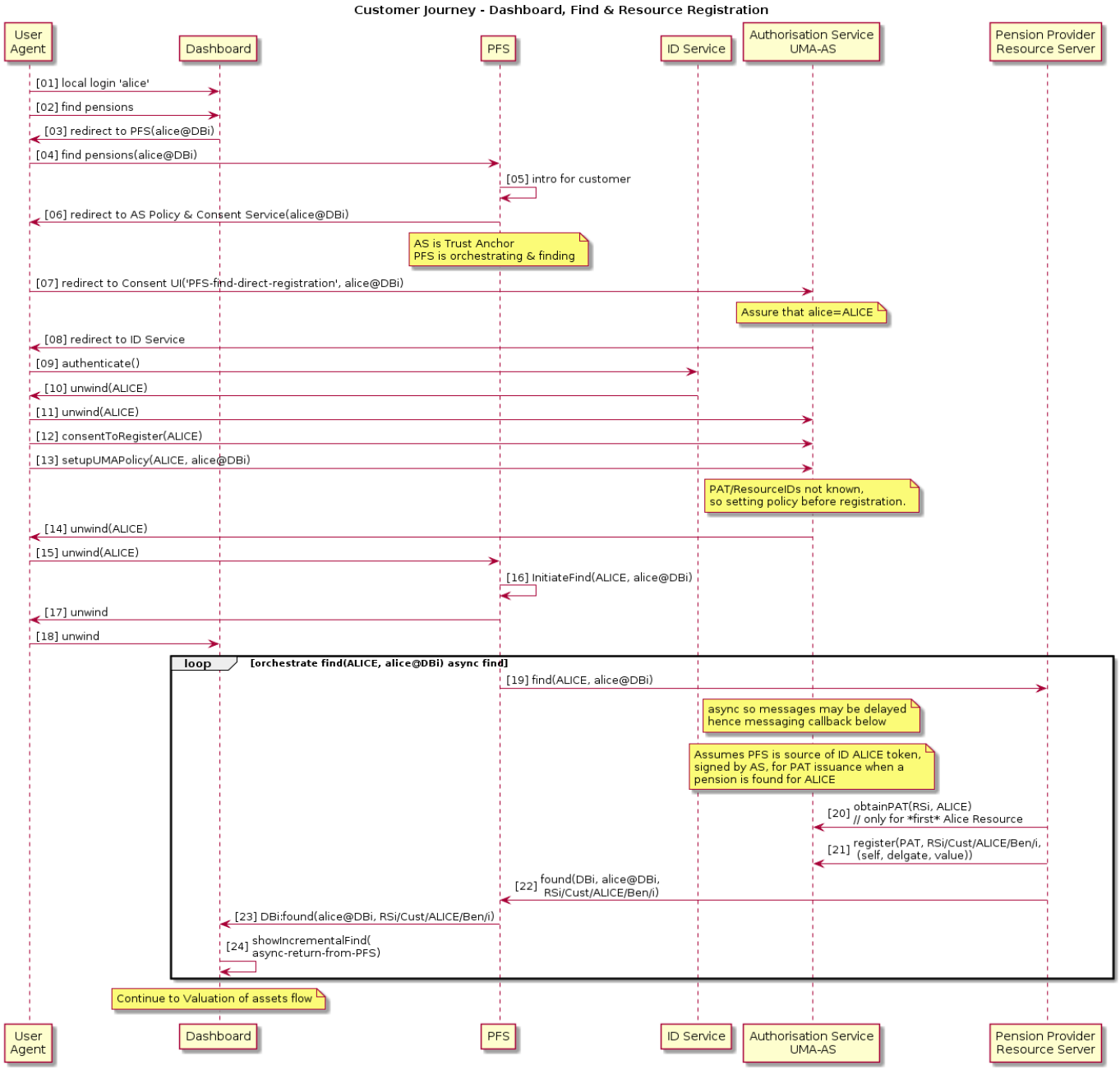
[6] This is the form used in the current sequence diagrams for the UMA profile. There is an ongoing discussion about the nature of this URL and whether the customer segment is desirable. Clearly a resource server could derive the customer identifier from the asset identifier if the latter were unique, so this discussion need not concern us for the purposes of the UMA profile.

[7] It is possible that a pension owner has more than one asset at a pension provider. A design decision has been taken that each asset will have a unique protected resource which will be managed individually. Thus, for each pension access a separate protocol trip is required (even though UMA itself does not impose this).

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

8. High Level Flow

7.1 Consent, Find and Register



Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

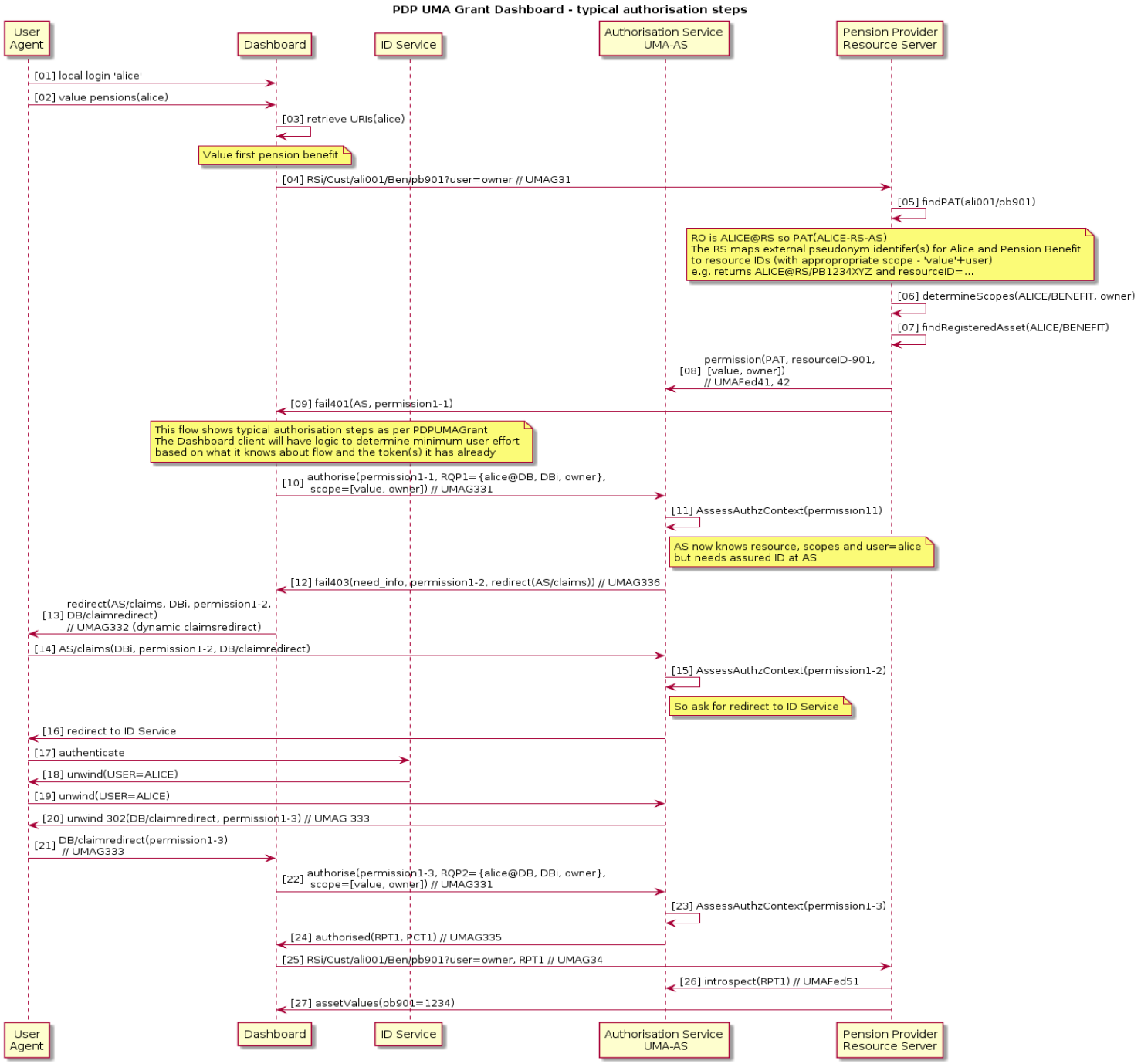
Step	Description	Notes
01	(Potential) Pension Owner, 'alice' logs into a Pensions Dashboard	
02	She requests that the dashboard finds her pensions for her	
03	The dashboard redirects her browser to the PFS, 04, where she reads and introductory section and choses 'continue', 05.	
06	Her browser is redirected to the Consent Manager & Authorisation Server, at which she is required to assert a standard identity, by redirection 07, 08.	It is possible that in the future pension owners may prescribe which AS they wish to use, but this is not envisaged in the short term.
09	The Federated ID service may proof her, or just authenticate her, if she already has an identity, and unwinds the redirect to the AS, 10, 11, carrying an ID assertion that she is ALICE for the Authentication Service as a relying party (i.e. audience).	
12	The Consent Manager at the AS gathers her consent to find and to register the results of her find with the AS.	
13	The UMA Policy manager records her policy	Policy: "*ALICE* delegates to *alice* at a specific Dashboard for a time".
14	AS unwinds redirect to return browser to PFS, 15	
16	PFS initiates the (asynchronous) find process across all pension providers.	
17	And unwinds the redirect from the dashboard returning control of the browser, 18. Where the dashboard waits for async returns 24.	
19	The find is orchestrated across all pension providers, each receiving a ALICE token (derived by the AS from ALICE's IDP issued identity token) which contains her attributes for the search and information enabling found resource identifiers to be returned to the PFS for onward messaging to the dashboard.	Here ALICE is a token for all of the data needed related to ALICE's assured identity and attributes, and for use as a temporary credential for PAT issuance.
20	The first time a pension is found for the assured pension owner, the RS (i.e. pension provider resource server) needs to obtain a PAT (as described above section 0)	PAT issuance is required only for the combination AS-RSi-ALICE.
21	The newly found resource (logically represented as Rsi/Cust/ALICE/Ben/i) is registered at the Authorisation Server, returning the resource_id issued by the AS for that resource.	In this design the full list of scopes is registered for all resources (this is so that the resource owner can subsequently delegate access if required without further RS activity).
22	The newly registered resource identifier is messaged back to PFS and hence to the relevant dashboard, 23	
24	The dashboard may present the found pension identifiers as they arrive.	It is possible that some pension providers will respond slowly. The dashboard may move on to access the value of the pension asset as it chooses[8].

[8]The async timings of registration and subsequent access must not be a problem for the implementation of the UMA AS, RS (or other components). ...

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

8.2 Authorise and Access

Value the first pension resource



This design only guarantees that a specific resource is registered before its unique identifier is returned to the dashboard; thus, concurrent FINDs and Values are possible, but value of a specific resource must follow its registration.

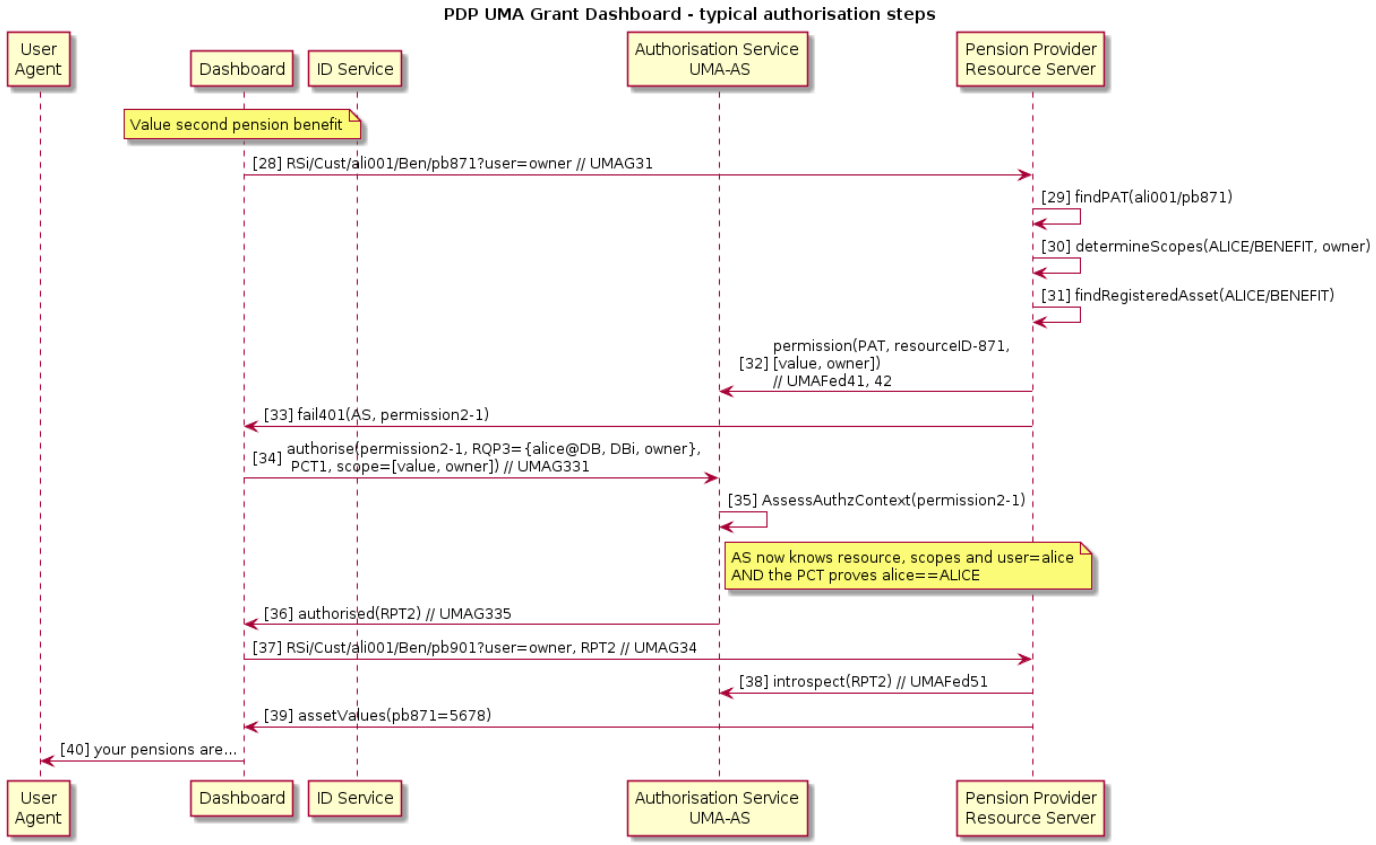
Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

Step	Description	Notes
01	The requesting party 'alice' logs into her dashboard	
02	'alice' requests a valuation of her pensions	
03	The dashboard is permitted to persist the resource identifiers, so retrieves them, in this instance, two pension assets, both at the pension provider RSi	Dashboards may also persist RPTs (UMA access tokens) and PCTs (UMA persisted claims tokens)
04	The dashboard tries a blind access to the first resource, stating that the user is acting as the pension owner.	
05	The resource server uses information in the request to determine the PAT, and requested scopes, 06, and its record of the resource_id, 07	See section 0.
08	The RS seeks a permission ticket for this information which the AS returns	
09	The access call fails, returning the permission ticket and as required by the protocol, the relevant Authorisation Server.	For compliance and for extensibility it is possible that in the future resource owners may prescribe which AS they wish to use, but this is not envisaged in the short term.
10	The dashboard attempts to obtain a token (RPT). For each authorisation attempt it is required to mint a new assertion of its identifier (DBi), its user (alice@DBi) and its assertion of the role of that user (owner). (The type of this token in the profile is RQP.)	See section 0 for a list of claims needed for authorisation.
11	AS assesses the authorisation context. In this case there is not sufficient information as the AS has no current proof of the assured authentication of the relying party user and therefore cannot correlate alice@DBi with her identity.	The profile contains detailed definition of the state required to do so and suggests that the implementation of the AS is as stateless as possible, by carrying the authorisation context in the incremental versions of the permission token.
12	So, a failure results, issuing a new permission ticket, and the dashboard complies with the need_info redirect request, 13, redirecting the browser to the AS claims redirect endpoint 14.	
15	The authorisation context is re-evaluated and the AS forces redirection to the ID service, 16.	
17	The requesting party user is authenticated as ALICE and the redirect unwinds 18, to the AS, 19	
20	The AS unwinds its claims redirect via the browser 20, providing a new permission ticket, to the dashboard redirection endpoint 21.	
22	The dashboard tries again to authorise the access citing a new RQP and the permission ticket.	
23	This time the authorisation context is sufficient to authorise the access, so the successful return, 24, contains an access token RPT, and a PCT.	The PCT contains a persistent claim, binding ALICE with alice@DBi as an owner.
25	The dashboard retries the initial access (step 04) this time citing the RPT	

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

Step	Description	Notes
26	The RS introspects the RPT successfully	This call requires the PAT which the RS retrieves based on the call, potentially accelerated if it can read the structured RPT token which contains the RO identifier at the RS. See the UMA profile for details of token design.
27	The RS performs any access control logic it may require (in addition to the authorisation by the RPT), and serves the value associated with the resource.	
	... continued...	

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd



Step	Description	Notes
28	The dashboard seeks the value of the second pension asset (see step 03 above)	
29	The resource server uses information in the request to determine the PAT, and requested scopes, 30, and its record of the resource_id, 31	See section 0. and step 05, 06, 07 above
32	The RS obtains a permission ticket for this attempted access from the AS, and returns it with the failed access code to the dashboard 33	
34	The dashboard seeks to authorise, citing its permission ticket, a newly minted RQP (stating the user of the dashboard DBi is alice@DBi in role ‘owner’ of the pension), and citing the user’s PCT as provided above in step 24.	
35	The AS assesses the authorisation context and finds that all of the required claims are available and thus issues an access token, 36.	
37	The dashboard retries the initial access (step 28) this time citing the RPT	
38	The RS introspects the RPT successfully	
39	The RS performs any access control logic it may require (in addition to the authorisation by the RPT), and serves the value associated with the resource.	
40	‘alice’ can see the value of her two pensions!	

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

9. Design Decisions

This section states design decisions taken during the development of the profile. In addition, it highlights those features of the profile which either reflect these design decisions, or which may need specific attention by reviewers.

8.1 UMA

Pensions Dashboard design proposes to use UMA because it meets the 'Characteristics of the domain', as above. Notably:

- Managed delegation is intrinsic to the Pensions Dashboard domain
- Delegation to 'self' supports existing industry portals and independent choices of authentication and assurance by dashboard providers
- Longer term the user can choose her own Authorisation Server (as the notion of dashboards extends to cover more financial services, and in line with evolving consumer protection)
- **UMA defines federated authorisation which is a requirement of the domain as most of the pension providers do not have any, nor standard based, authorisation services**.
- UMA standardises the above and provides standard mechanisms of handling asynchronous policy and access control.
- UMA standardises authorisation fail flows, including their management by pushing claims and AS mediated activity (independent of the client) and therefore standard 'authorisation error handling' for users and for authors of client software and pension provider software.
- UMA separates authorisation from access, thus supporting the design objective that the pension value data does not pass through a central service (minimising the attack surface)

8.2 Federated Standardised Identity

- Dashboards do not have standard identity regimes
- Pension Providers do not have digital accounts, nor standard identity
- All data controllers need a known baseline identity for consent and access to their data
- All data controllers need a known baseline of PII attributes to support search (and confidence that these attributes are those of the assured identity which has also consented for the search and data release)
- If the person being authenticated is a financial adviser or member of staff of a guidance body, the ID service will also attest to the person's professional status to provide such advice/guidance

8.3 Resource Owner

The profile assumes that the Pension Owner (a natural person) is also the Resource Owner and that this person authorises the PAT for federated authorisation to the UMA AS.

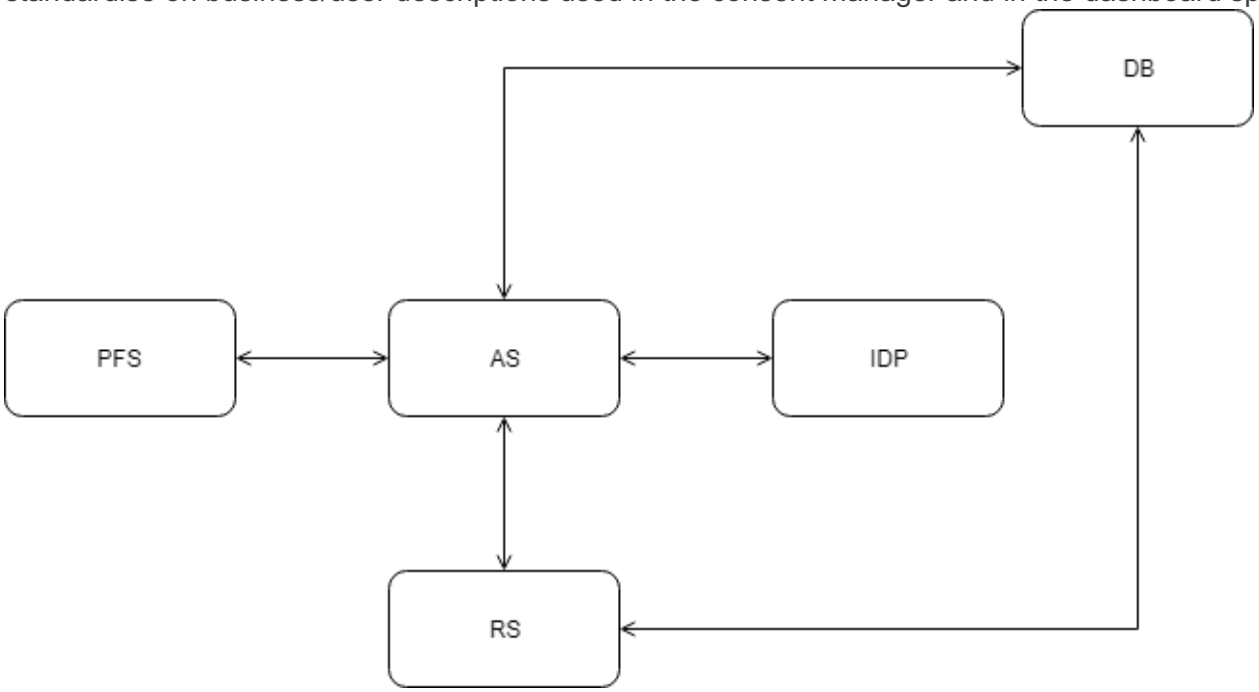
- The model in which the Pension Provider acts as a corporate resource owner was considered: corporate RO would have only one PAT for all its pension asset customers. This was rejected
- The Pension Owner acting as RO makes a personal consent decision to enable her pension resources under protection. This is correctly represented by a PAT per RO
- The Pension Owner may (in the fullness of time) choose to use another UMA AS, and this is facilitated by fully implementing the PAT per natural RO model
- The AS has an assured identity for the Pension Owner at the time a find is initiated, and thus is in the position to mint temporary credentials for the RS to use to obtain the PAT

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

8.4 One Resource per Access

The profile assumes that each separate pension asset valuation will have a separate access (GET) request even if the owner has multiple assets with the same pension provider, and accordingly an RPT and PMT will represent a permission request/access token for only a single pension resource.

- Simplifies the logic of dashboards which can maintain a 1:1 association of pension asset URI with associated RPT, and delete the RPT without ambiguity
- Supports user requirement to manage policy to delegate each asset individually
- Supports the user in enabling a per asset textual representation of the asset so that the industry can standardise on business/user descriptions used in the consent manager and in the dashboard space



- In the longer term this supports the user in allocating a different AS for each of her financial assets
- Enables the unambiguous semantics of the proposed URI ie 'the value of the pension owner- pppp's, asset number nnnn'. (There is no need to enumerate assets or other list/collection activities)
- Supports the industry performing consolidation or moving of assets across pension providers for whatever reason. Each asset could be remapped separately

8.5 Relationship between PFS and Authorisation Server

This document (sequence flows etc) states that the AS is the point of integration with the Identity Service and that the AS manages the whole of the consent policy for the user (consent to find, place under UMA protection and UMA authorisation policy). Accordingly, the sequence diagrams show this relationship. Moreover the 'find' orchestration is performed by the PFS which is relaying the token represented as 'ALICE' to the pension providers, which includes the temporary credential for the RO to obtain a PAT on behalf of ALICE if she has a resource at that Resource Server.

- This is adequate for an understanding of the UMA profiles and to conceptualise the non-UMA architecture of the system as-a-whole
- It is possible in the implementation that a more complex/nuanced relationship may exist and the above (consent/policy) functionality be distributed more between the PFS and AS. Indeed, it may be that UMA policy could be managed by a delegated UI[9], perhaps to dashboards themselves

[9] Noting that this feature would involve the user in authenticating to the AS to use the policy management capability, although conceivably a current PCT might be used in this context. (Details out of scope of this document.)

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

- These implementation design details are not considered germane to the review of the UMA profiles

8.6 Representation of Resources at the AS Policy

When found a pension asset is registered for UMA protection, the AS allocates an UMA resource_id for the asset. The AS is required to enable a Pension Owner to configure policy for her resources. This means she needs a human readable representation (not just an UMA resource identifier).

- The UMA Fed Authz resource description is profiled to include extra information to carry the URI and description information into the AS to provide the data against which the policy UI will function
- This is another place in the profile where there is an impact of the decision for 1:1 relationship of registered resource and user-facing and client-facing URI

8.7 PFS Communication with Dashboards

The result of finding a pension asset is that (with owner's consent) it is registered and the URI is returned to the dashboard instance for the user who initiated the find there. The sequence diagram shows this happening asynchronously, via the PFS, to the dashboard, as each positive find returns.

- These decisions are likely, but not certain. They make no difference to the UMA profile
- The UMA profile does define, and UMA requires, that a resource is registered with the AS before an access attempt can result in an UMA permission ticket

8.8 Nature of Relationship with Adviser or Guidance staff

- The delegation will be performed by the Pension Owner herself, ie as an assured identity in a session with the AS. (Other mechanisms of delegation are not discussed here)
- The delegation involves (for Independent Financial Advisers) an existing, out of band, relationship between the customer and the adviser. The system delegation of authorisation assumes, but does not prove, that this relationship exists
- The adviser/guider may attempt to access pension resources. The UMA authorisation flows as defined here, and in the profile, will apply
- As defined in the profile, the UMA capability to poll the authorisation server waiting for RO approval will *not* be used for reasons of attack surface and scalability, if no policy exists at the time of attempted access, the access attempt will *not* return a permission ticket
- The PFS and/or consent management features may arrange for the relevant resource URIs to be sent to the adviser at the time the owner manages her authorisation policy. This feature is not UMA
- Equally it is possible for the owner to send her URIs by any other means to whomsoever she wishes
- When a resource server registers resources with the AS, it will apply all three scopes even though at that time the pension owner may not have any delegates. This is to simplify the management of scopes at the RS: it simply enables the 'delegate' scope to be used in the future if the owner enables a delegation in policy at the AS

8.9 Authorisation Server State

The profile makes explicit the claims and other dynamic entities which are needed for authorisation decisions in the form of profiled tokens (RQP, PMT, PCT). These are managed in accord with the UMA profile to provide all the functional authorisation state. In particular it is proposed that Authorisation Server state is externalised in permission tickets during multiphase authorisation. This is clearly an implementation choice, but it seems highly desirable that the attack surface of the AS is reduced, its working memory reduced, and authorisation transactions are made stateless across protocol stages. These characteristics will assist scalability.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

8.10 Tokens

Summarising decisions on tokens:

- The design uses structured tokens (see section '0')
- RPT is required by UMA and is profiled
- PMT is required by UMA and is profiled
- PAT is required by UMA Federated Authorisation and is profiled
- There is a custom claim token (RQP) which represents the Pensions Dashboard's assertion at the time of authorisation of the identity of the user, the dashboard instance and the user's role
- PCT is permitted by UMA and is profiled. This claim binds the asserted user at dashboard and role to that user's assured identity and professional status
- OAuth MTLS Token Binding is mandated
- Properties of tokens incl time to live are summarised below

Token	Iss	Sub	Aud	Other Attributes	Bound to (DB/RS)	Crypto	TTL
PAT	AS	RO@AS	RSi	RO@RSi	Y (RS)	S	18 month[10]
PMT	AS	RO@AS	AS	Requested permissions + extensive – used for AS state	N	S,E	< 60 sec One time
RQP	DBi	RP@DBi	AS	Role of RP at dashboard	N	S	< 60 sec One time
PCT	AS	RO@AS or Delegate@AS	AS	RP@DBi, role	Y (DB)	S,E	1-3 month[11]
RPT	AS	RO@AS or Delegate@AS	RSi	Granted permissions, RP@DBi, role	Y (DB)	S,E	< 5 day[12]

8.11 Structured Tokens

The profile uses structured tokens (proposed in the form of JWTs) rather than bearer tokens.

- Information leakage is mitigated by appropriate use of encryption
- Tokens carry state which makes the AS lighter and assists scalability, and minimises data attack target at the AS
- Dynamic registration of non-confidential clients (SPA, native app, as assumed required by dashboard writers) needs clients to manage crypto material so the conceptual load (on developers) is a given

[10] PAT is held by secure RS and bound to it. (Refresh tokens may be considered but if so the persistence of state at the AS needs to be carefully considered. In any case revocation by the RO at the AS of the PAT needs to be considered, and this may create further state at the AS.)
[11] PO up to 3 months. Delegates much less, perhaps up to 1 month
[12] Dashboard clients may very (public/confidential), to be decided based on AS load, caching policy at DB, and exact implementation details including those of reflection. If the dashboard needs to refresh the RPT it is expected that the whole UMA dance will be necessary, but the user journey will be transparent if a current PCT is played.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

- Token binding (using the JWT Certificate Thumbprint Confirmation Method) uses structured tokens (RPT, PCT) so development load is a given (but the client developer does not have to be aware)
- The clients are not required to encrypt their token (RQP – a specialised claim of user and dashboard identity e.g. alice@DBi), but the AS does need this information as a claim, so the client developer will have to create a structured token
- The clients do not need to decrypt other tokens (PMT, RPT, PCT) since these have the AS as the audience, so the client does not have to be aware of the structure
- The profile states that the RPT will be introspected at the AS. Consideration of the structure of the RPT token, i.e. that it contains all the information apparently obtained from introspection may lead the reader to doubt the need for introspection. In this profile introspection is for two reasons:
    - a) to validate the OAuth MTLS Token Binding fingerprint (of the client, by the AS);
    - b) to manage the functional requirement for user revocation of permissions to access at any time.

  The latter of these is of course part of the debate about TTL of the RPT vs AS performance. If the RPT had a short life-time

8.12 Client Types

It is assumed that Dashboard providers will wish to deliver non-confidential clients. Moreover, it is assumed that the eco-system will wish to perform dynamic registration of clients in general.

- The AS will play a role in dynamic registration of clients. This is mentioned in the profile, and is dependent on the technical and business design of a 'governance register' (not in the scope of this design document). There will be an eco-system PKI provided within the technical implementation of the governance register
- Client authentication to the AS will be defined in accord with dynamic registration requirements. (Outside the scope of the UMA profile)
- It is noted that the use of the entity 'DBi' in this document and the profile as an identifier for the instance of the dashboard used by the requesting party. This is used, for example, in the PCT which will persist in the dashboard for a significant period. Of course, if the dashboard instance does not have persistence capability (e.g. a SPA web app served to a public endpoint, or in general if the user terminates a SPA session or a mobile app is uninstalled) then the PCT should be deleted. An implementation concern is DBi must not be reallocated. DBi may be implemented for public clients as a per instance clientId (as is stated in the profile). As also noted in the profile, we need to be open to hybrid designs for public clients where there may be other arrangements to support persistence of PCT, RPT and resource URIs and hence 'DBi' across public client instances
- The profile mandates token binding, and thus the AS must support token binding. For non-confidential clients, the client and the AS will support OAuth MTLS Token binding using self-signed certs.
- Long lived tokens will be managed by the relevant entity:
    - RS will manage PATs.
    - PCT and RPTs will be managed by requesting party clients (i.e. dashboard client software). TTL for public clients may have to be reduced.
- ***Opinions of reviewers are sought, considering the other protections in this profile (Sections 8.10, 8.11,8.12)***

8.13 Privacy

Remarks on privacy have been made throughout the profile document. This design document has not commented further on these matters. However, please note the points below which are also shown following the diagram below:

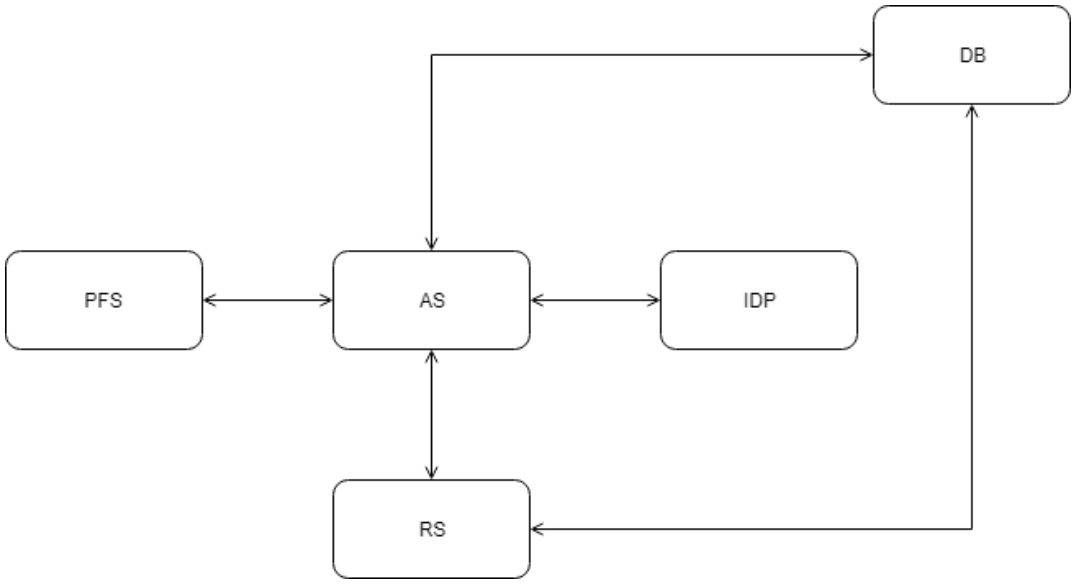
- The token used here 'ALICE' is shorthand for a number of attributes, both PII and identifiers for her account at the AS and at the RS. In general, it is expected that there will be independent UUIDs, perhaps derived by hashing, to prevent correlation across the pension providers

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

(The AS cannot avoid correlating the accounts for Alice at each resource server, since it is an AS function to manage Alice's authorisation policy)

- Similarly, the notation 'alice@DBi' will be specific to a dashboard and is expected to prevent correlation across dashboards.
  (Although the AS cannot avoid the ability to perform such correlation as it is the trust anchor for ALICE@'Identity Provider' from which the AS derives ALICE@AS, the AS will not actually perform such correlation other than to associate Alice's chosen dashboard account with her AS assured identity. Even in this context the association is persisted in the PCT which the AS does not locally keep)
- The notation used for the URIs of the pension owner's assets is written as if it encodes PII. It is expected that obfuscation will be applied such that the identifier for the customer will be unique to a resource server and not be what is used within that server nor within the AS to identify the customer accounts
- The tokens which contain higher assurance information (PCT, PMT, RPT) are encrypted and thus do not leak information to the dashboard

High-Level Dataflows to key UMA-related privacy issues



Link (directional)	Data Item	Use of item
DB->AS	alice@Dbi, Dbi (in a RQP)	DB identifies itself and its user AS uses in authorisation decision
IDP->AS	ALICE@idp = (MDS, IDP-UID[13])	Matching Data Set is not kept at AS, UID is hashed H(IDP-UID) H(IDP-UID) is the AS' unique account reference for ALICE i.e. ALICE@AS
AS->PFS	H(ALICE@AS), MDS	PFS profile id for ALICE – hashed again, Alice's MDS (for matching)
AS->RS	H(ALICE@AS)	Temporary credential for PAT issuance

[13] UID is persistent identifier: some unique identifier – issued by a source and may be kept by a recipient, often hashed H(UID), potentially repeatedly H(H(UID))

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

Link (directional)	Data Item	Use of item
RS->AS	ALICE@RS, RSi	Unique reference for Alice at the RS, so that the AS can correlate PAT and resource_ids with the correct RS, RSi the unique identifier of the RS
AS->RS	PAT	Encrypted for AS
DB->RS	URLforAliceAndAsset *<pension-provider>/ Customer/<ALICEUUID> /Benefit/<PENSIONASSETUUID*	This URL *ALICEUUID* should have been obfuscated by the RS when it was issued after the find. Inside the RS only it is mapped to ALICE@RS
AS->RS->DB->AS AS->DB->AS	Permission Tickets	Encrypted for AS
AS->DB->AS	PCT	Encrypted for AS
AS-DB->RS->AS	RPT	Encrypted for RS, RS can introspect at AS, At RS, permissions, ALICE@RS, alice@DBi RS needs to know the user as identified at the DB and at itself so that it can perform its access control decision.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

10. Notes for UMA Implementors

This section comments briefly on what may be needed from suppliers of UMA products to support this profiled UMA design.

9.1 UMA Features

The profile relies on standard UMA 2 capabilities as can be seen from the sequence diagrams above. Specifically, we note that client redirection endpoints must be supported, including pushed claims, dynamic redirection (to federated ID service) and persistent claims tokens.

The AS must support structured claims: RQP token from the DB, PCT to the DB and RPT.

The profile makes use of structured tokens to avoid state at the AS (to minimise state complexity and decrease the attack surface).

The AS must support user managed policy against a standard templated policy type as stated in this document. Policy based consent for delegations must be revocable by the user.In general, the design profiles UMA 2 and other security standards, based on the assumption that an open standard design enables a potential market place of vendors, avoiding lock-in due to proprietary designs, and facilitating services in both software and related eco-system supporting services.

9.2 Security Features

In addition to UMA2 features the profile requires the following (derived from security BCP)

- Signing and encryption of tokens
- Structured tokens supporting OAuth MTLS token binding
- Token binding using self signed certs (for non-public clients)
- Issue/accept temporary credentials for PAT issuance

In addition, dashboards will be OAuth clients of the PFS and the AS should support this (including dynamic registration of clients) so that the authentication journey is simplified and integration testing is reduced.

9.3 Dynamic Registration and Related

The PD eco-system is expected to have an overall governance register which comprises both offline process-based assurance and online PKI and dynamic registration of client software.

To support clients (especially non-confidential) clients the design requires the AS to support dynamic registration and to provision clients appropriately. The AS needs to support software statements to bootstrap the registration process securely.

9.4 SDK for Dashboard Clients

The UMA vendor(s) should provide an SDK to enable dashboard clients to be written more easily against the UMA and security profiles, and to decrease integration testing time.

9.5 Adapters for Pension Providers

The Pension Provider must implement its UMA Resource Server which must

- comply with the profile
- manage its credentials
- Manage temporary RO credentials during PAT issuance
- store a PAT per user and associated UMA resource_ids
- Issue and Map external URIs for assets

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd

If every pension provider were to develop and test its own solution to the UMA RS requirements, the overall cost in development and testing over 400+ RS endpoints would be large. The UMA supplier(s) must provide an 'adapter' as an executable, a service, or a configurable SDK which offers a cost-effective deployment route, minimising cost and particularly integration testing.

9.6 Inter-op Testing

Even given an adapter approach to RS integration, a common SDK for client software, and a common shared AS, there will still be the need for

- Interop testing of components to prove compliance of vendor(s) products with the profile[14]
- sandbox and test environments for incremental test for the industry

[14] This is the type of service which Kantara has offered for UMA and other protocols, but other vendor(s) may offer such services.

Authored by Mike Pegman (info@vinesolutions.co.uk), Vine Solutions Ltd for Origo Services Ltd Mar-Sept 2019
Contributed to Kantara Initiative by Kenneth May (kenneth.may@origo.com), Origo Services Ltd