Pensions Dashboards Programme

Architecture brief for suppliers

March 2021

Contents

1. Introduction

The Pensions Dashboards Programme (PDP) developed this document to brief potential suppliers of the central digital architecture that will enable pensions dashboards to operate. The Pensions Dashboards Programme is part of the Money and Pensions Service (MaPS). This document provides an overview of the central digital architecture and the interfaces.

This document represents our current position, as at 19 February 2021 and is being provided to inform suppliers of the high-level architecture design prior to commencing any future procurement exercise. The document is proprietary to MaPS and the information contained herein is confidential and must not be reproduced, either in whole or part, in any form or disclosed to others without prior written permission from the Pensions Dashboards Programme commercial team.

The pensions dashboards ecosystem comprises dashboards, data providers' interfaces to the ecosystem, and the central digital architecture. PDP is responsible for the delivery of the central digital architecture and services, which enable data providers and dashboard operators to inter-operate. The central digital architecture includes the definition of the components deployed by data providers and the functionality, which must be provided in dashboards.

Within the scope of this document, a 'data provider' is a pension provider, scheme, trust, integrated service provider[1], or other agency which is supplying data to the pensions dashboards ecosystem (PD ecosystem), including the Department for Work and Pensions (DWP) as providers of State Pension data.

A 'dashboard provider' is an organisation that operates a pensions dashboard. A pensions dashboard is a software application (a web application, or a native mobile application), which enables a user to find and to view pension information, irrespective of the diverse data providers that may manage those pension, and irrespective of each data provider's digital portal, or lack thereof.

This document provides:

- a description of the architecture for the pensions dashboards ecosystem (including the identity service)
- a commentary on the interface components of data providers to interface technically with the pensions dashboards ecosystem
- a commentary on the functionality that dashboards require to interface technically with the pensions dashboards ecosystem

Although this document describes the identity service as part of the architecture of the entire pensions dashboards ecosystem, the identity service is out of scope of this procurement vehicle.

[1] Although this document describes the identity service as part of the architecture of the entire pensions dashboards ecosystem, the identity service is out of scope of this procurement vehicle.

2. The Pensions Dashboards Programme architecture

2.1 Overview of the architecture

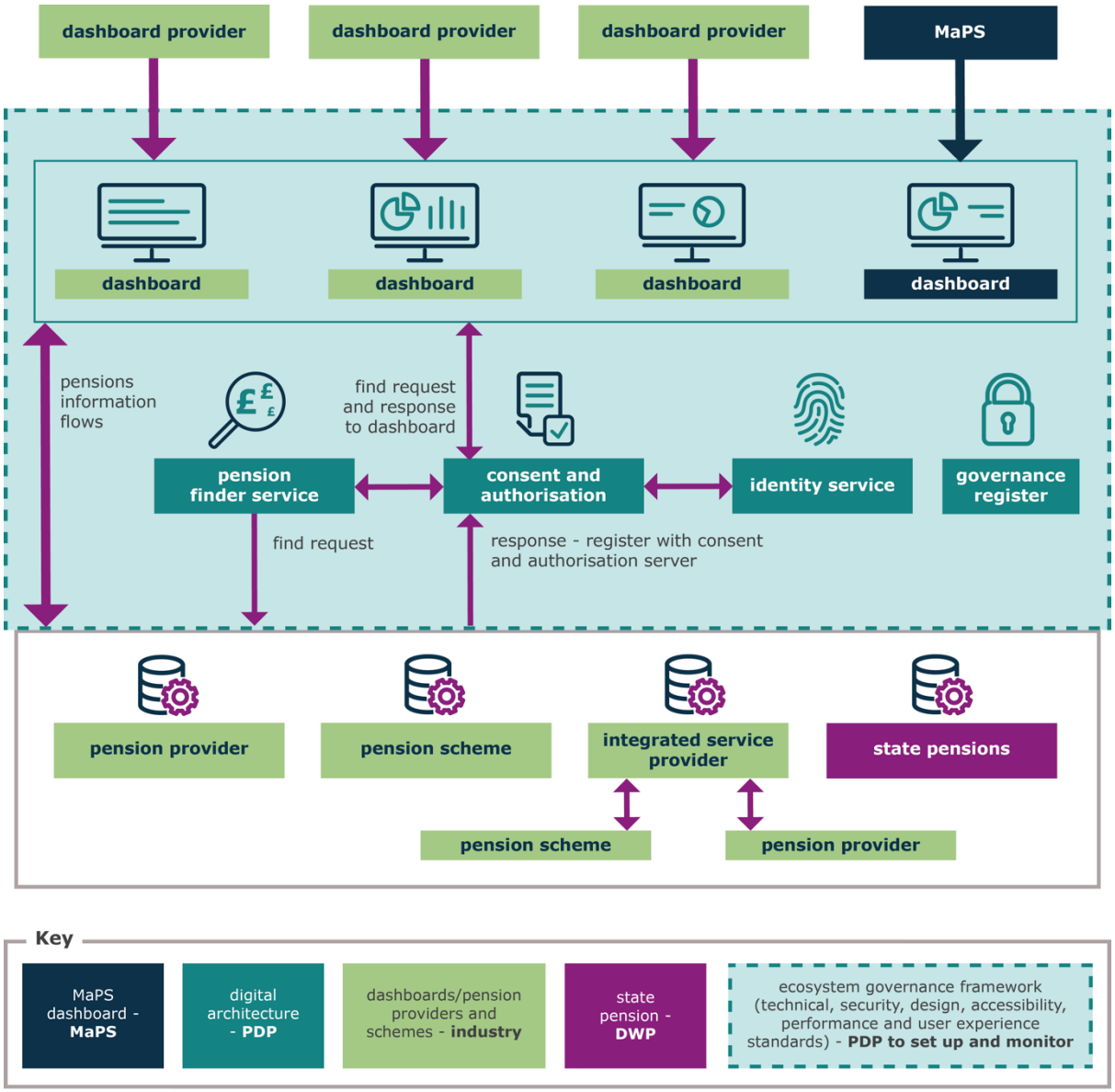

Diagram one: overview of the architecture

The **data providers**[2] comply with a standard interface with the ecosystem. This interface *provides personal data*, which the data provider uses to find records[3]. The interface also, separately, and subsequently, authorises access(es) to those located records.

---

[2] From the architectural viewpoint it makes no difference what nature of entity provides the pension data: whether a single scheme, an administrator, a third-party integrated service provider or DWP State Pension, all are equivalent in the candidate architecture.

[3] Note that the data provider interface itself does not find the records. It standardises the interface to the pension provider's internal systems, which actually search the pension provider's data for matches to the customer's personal data.

The **pension finder service** orchestrates find activity. The **consent and authorisation service** manages consent and authorisation of the pension owner, to enable the pension owner to search for and retrieve their data to a dashboard. It also enables the *pension owner* to give *delegated access* to certain human[4] third parties (financial advisers or individuals in official positions at MaPS).

Authorisation (by a pension owner to search for and to access their records) is only viable if parties to the activity of disclosure can be assured of the identity of the subject. Moreover, to save reverifying identity on each occasion, the user needs a mechanism to authenticate whenever necessary. The **identity service** will manage 'proofing of identity[5] and authentication credentials[6]. PDP cannot assume that the data providers have a business to consumer (B2C) digital capability, nor an existing digital identity mechanism, so the pensions dashboards ecosystem has to provide such capabilities on behalf of all data providers.

We expect the identity service to be a mechanism of obtaining independently verified identities, probably from a federation of independent identity providers. It will cover individuals, the pension owners and professionals, financial advisers, potentially with a check to validate membership of a register.

Dashboards present the results to the user and enable the user to view their located pensions. MaPS has a separate programme of work to deliver a dashboard. We anticipate other organisations will also provide dashboards.

The user **also** interacts with the services that provide identity, consent, and authorisation (ie these services have user interfaces to which the user is redirected from the dashboard).

A **governance register**, consisting of both *organisational processes* and *online IT components*, controls which organisations and which software instances can participate in the ecosystem. It also provides central reporting, technical monitoring and similar services.

We expect a single technical component to provide some of the technical elements of the governance register and the consent and authorisation service. This will mean that the *static* relationships in the ecosystem (eg proving an organisation is a legitimate pension provider) are managed together with the *dynamic* authorisation performed by the consent and authorisation service (eg authorising a specific user to access specific pension details).

The Department for Work and Pensions (DWP) consultation report committed to basing the consent and authorisation service on the open standard User managed access 2 (UMA2) protocol, to enable:

- a single, federated, authorisation service for the all the data providers
- delegation (both for the pension owner to their delegate/financial adviser, and for the pension owner identity at the central service to her persona at any dashboard)
- fine grained authorisation at the level of a per-pension per delegation control as determined by the pension owner's policy at the central service

[4] We specify human to clarify that the delegates must be human, to differentiate from systems, such as robo-advisers or otherwise which are out of scope.

[5] Identity proofing is the subject of the NCSC Good Practice Guide 45.

[6] Credential management is the subject of the NCSC Good Practice Guide 44.

Diagram one, above, indicates some of the information flows:

- dashboards hand off to the consent and authorisation service to initiate find requests and receive responses to such requests via that service
- the consent and authorisation service interacts with the identity service to authenticate the user if the user does not have a currently proven identity, the identity service proves the identity of the user. If the user is a delegate, it also attests to their status as an adviser or guidance official
- the consent and authorisation service solicits consent from the user and may gather self-asserted information to supplement their proven identity attributes for the find process
- the consent and authorisation service initiates the pension finder service (PFS), which orchestrates the search
- data providers respond to the consent and authorisation service, creating a unique pension identifier (PeI) for the found pension, and registering it with the consent and authorisation service, so that it can control subsequent authorisation of accesses and manage user consents
- dashboards use responses to finds (via the consent and authorisation service) to initiate requests for details of a pension by a direct call to the relevant data provider. (As a result of such a request from a dashboard to a data provider, the consent and authorisation service is involved to authorise the access based on the user's current consents)

Although not explicitly represented in diagram one, above, data providers will have a standard interface that is governed by the ecosystem governance framework (purple in the diagram). This defines the connectivity rules, internal behaviour, exposed resources and authorisation mechanism for access. Similarly, dashboard providers must interoperate with the pensions dashboards ecosystem consent and authorisation service and data providers in accord with the governance framework requirements. This document describes the processes and outlines the functionality of such interfaces.

2.2 Components and user interaction - more detail

This section presents more detail on the above components.

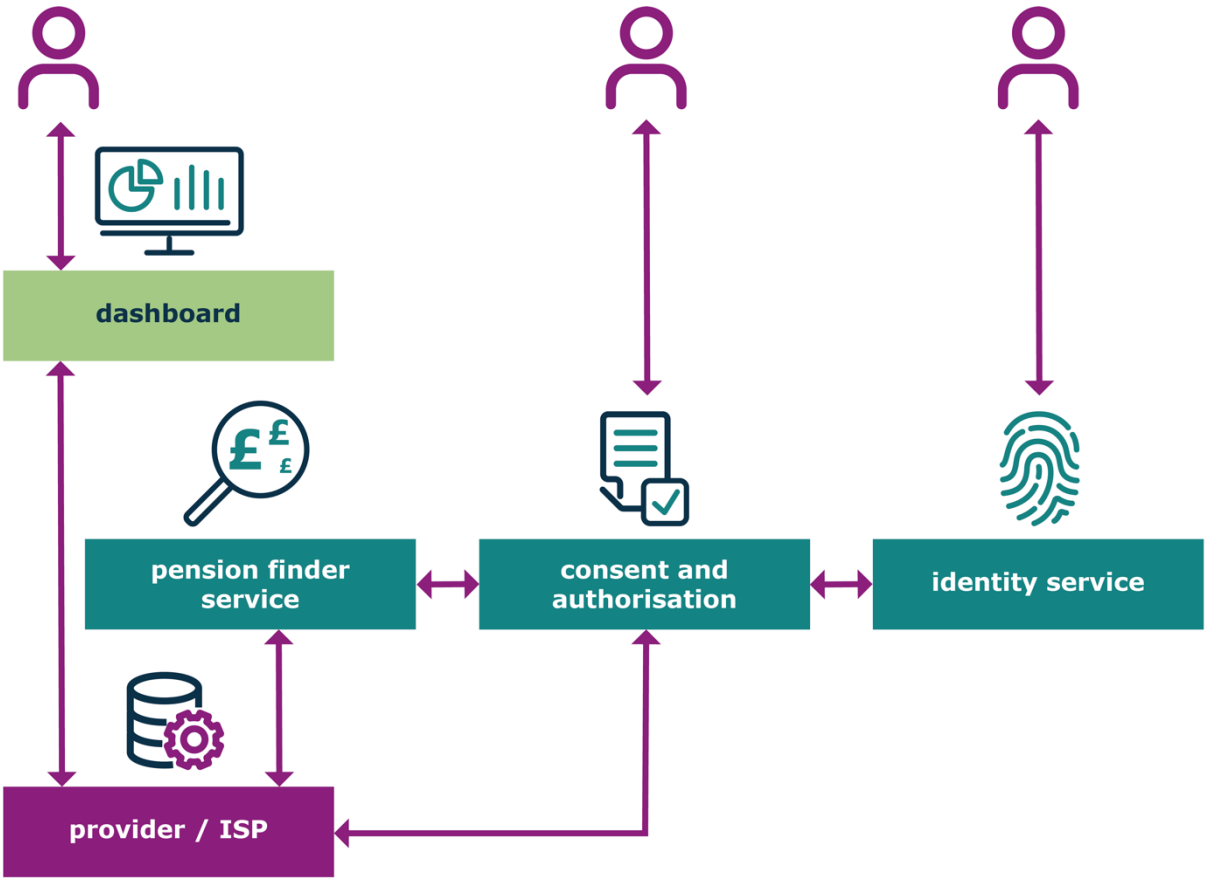

Diagram two: component detail

Note that the consent and authorisation service (C&A) is separate from the pension finder service (PFS). It is desirable that *the user* perceives that the pension finder service and the consent and authorisation services are the same thing, but this is a user experience design issue. Here we present the *distinct technical components and their roles and responsibilities*.

In this architecture, the pension finder service has the function of orchestrating instructions across providers, so that the data providers *find pensions* and *arrange for the dashboard to be informed* of the results (via the consent and authorisation service, as per the description in section **2.1**).

Prior to sending a find request to the attached data providers, the consent and authorisation service must be assured:

- *of the identity* of the user, by means of a separate identity service
- that it has the necessary *consent of that user* to perform the *find* activities and subsequently *access the details* of the pensions which were found

The consent and authorisation service creates a matching data set comprised of assured identity attributes from the identity service and potentially other attributes[7], such as a user-asserted NINO (National Insurance number), and representations of user consents for the data providers. The consent and authorisation service invokes the pension finder service, which orchestrates the calling of the find interface across the data providers.

The pension finder service sends this matching data set to each provider's standard find interface. The find interface receives the inbound data matching data set, and the provider will obtain data to match to their internal records to find a pension. A provider, having *located a pension* belonging to the user, informs the consent and authorisation service of the *existence of the pension* (in a process known as registration) and, in keeping with consents gathered from the user, enables the pension details to be accessed by the user.

Subject to user's consents previously given to the consent and authorisation service, it registers a pension identifier (PeI) representing the pension, (using UMA2 federated authorisation), then the consent and authorisation service returns the PeI to the dashboard, where it is stored on behalf of the user.

Subsequently (usually immediately in the user's first visit to the dashboard) the dashboard may *request the details associated with that pension identifier (PeI)* by directly contacting the provider, using the pension identifier.

The provider's view interface defines the retrieval and authorisation interface, which enables a dashboard to retrieve pension details. The provider *checks that the appropriate consent and authorisations* are current at the consent and authorisation service (the process of authorisation complies with UMA2 grant) and subject to these, returns the *details of the pension* to the dashboard. Please see section 6.4 of this document for more detail on the UMA2 authorisation protocol.

Thus, each provider implements a standard interface to the pensions dashboards ecosystem:

- a find service, which receives personal information from the pension finder service and requires the provider to locate records matching that personal information
- for each such match, the interface creates a pension identifier and registers it with the consent and authorisation service, which arranges for the identifier to be returned to the user's dashboard
- a retrieval service, which receives a pension identifier and access control information (tokens) from the dashboard. It checks those tokens against the consent and authorisation service and if authorised, it accesses pension details in the provider's internal records and returns them to the calling dashboard

Thus, the *user* interacts with the dashboard, with the consent and authorisation service and with the identity service. The dashboard obeys a protocol that arranges for the user at the dashboard to (temporarily) interact with the consent and authorisation service and the identity service.

[7] https://www.pensionsdashboardsprogramme.org.uk/2020/12/15/data-standards-guide/

The *data provider* interacts with the ecosystem components only via a standard interface. A data provider does not have any user interaction, ie it has no user-interface capability. The interface defines functional behaviours and responsibilities for the provider and defines mechanisms (protocols) for interoperation with the pension finder service and with the consent and authorisation service.

For the avoidance of doubt, data providers may also be dashboard providers, but, if so, that is a completely separate capability and has no bearing on their activity as a provider.

2.3 User processes

There are two distinct *user processes* when using the service.

1. User journey – authenticate, consent and find

The user of a pensions dashboard, or pension owner, will seek to find their pensions. The find process may be repeated if the user wishes but is usually a one-off activity. As the putative owner of pensions, the user will need to prove their identity to a suitable level of assurance that is acceptable to the ecosystem as a whole.

As a pension owner, the user will consent to the find and to enabling their pension details to be eventually retrieved by their dashboard. As the pension owner, the user will set policy for their own access and optionally set policy for their adviser(s). The dashboard will store unique dereferenceable pension identifiers (PeIs) for their pensions. The consent and authorisation service will keep their consent information so that it can enforce authorisation policy against their pensions.

2. User journey – authorise and view

The pension owner will seek to view their pensions at their pensions dashboard, either immediately after find or on subsequent occasions. The user's identity will have been verified for the dashboard (with whatever mechanism the dashboard operator requires) but the user will also need to periodically[8] re-verify that they are the same individual that consented to find and created the retrieval policy. Having met the conditions (of their own policy) for authorisation, the provider will take its access control decision and serve the details of the pension associated with the pension identifier to the dashboard.

The user's dashboard may hold on to tokens,[9] which may decrease the friction of their subsequent accesses. These tokens are issued as part of a User managed access 2 grant (UMA2) open standard[10] that applies *both* to the user when using their dashboard persona, *and* to their appointed advisers when they attempt to access the owner's pension details.

When **advisers** (either financial advisers or guidance bodies) use the service, they may perform only the authorise and view process above. That is, they utilise pension identifiers provided by the pension owner to attempt to retrieve the pension details. The identity service must prove who they are and their appropriate professional status will also be proven[11]. The consent and authorisation service also proves that the pension

[8] Taking a steer from Open Banking this period might be 90 days.

[9] Tokens are issued by the consent and authorisation service and represent aspects of the access control decisions it is enforcing.

[10] specifically profiled for the pensions dashboards ecosystem

[11] Professional status as adviser or guidance official will be proven either by a specialised identity service or via the governance register.

owner delegated access for the delegate to one or more pension identifiers and has not subsequently revoked that policy.

An adviser uses their client's pension identifier(s) to access the client's related pension details. These identifiers will have been provided either directly by the pension owner, or at the pension owner's instruction, when they established their policy of delegation to that adviser. The adviser's dashboard also persists tokens, which decrease the friction of the adviser's subsequent accesses, in accordance with the same protocol as for pension owners.

2.4 Nature of consent

The consent and authorisation service asks for the user's consent to process data. The pensions dashboards ecosystem is based on the principle that when it seeks consent, that consent should be clear, specific, time-bound, revocable, and ask for no more processing of data than is absolutely necessary to deliver a service in keeping with the pension owner's wishes.

The following table states the types of consent the user may grant that are applicable to the ecosystem.

Consent to	Meaning	Relates to components	Comment
Manage a digital identity	The pension owner will have a relationship with an identity and credential provider. *Duration of consent:* typically, years (managed by the identity service).	identity service	This is *outside* the pensions dashboards ecosystem but included here for clarity. An identity provider will provide verified data to the consent and authorisation service when a user authenticates their name, date of birth, address, identity provider's identifier.
Search for pensions	The pension owner will give consent for their data to be released to data providers for the purposes of finding their pensions. *Duration of consent:* each find.	consent and authorisation service, pension finder service, data providers	Verified data is obtained from the identity service. Self-asserted data, obtained from the user via consent and authorisation, eg NINO. Providers matching data set is name, date of birth, address, self-asserted data.

Consent to	Meaning	Relates to components	Comment
Place pension identifiers under authorisation control. This consent *must* be granted with search for pensions or the find operation will not be visible to the user.	On finding a pension, its identifier will be registered with the consent and authorisation service, so that future retrieval activities can be controlled by the user's policy. *Duration of consent:* typically, max 18 months, renewed when user visits the consent and authorisation service.	data providers, consent and authorisation	This consent permits the consent and authorisation service to act on the user's behalf to authorise access; it also provides the user with a list of pensions to view and control policy at the consent and authorisation service. This consent to register is separate from access at the dashboard - see the consent to retrieve pension details below.
Retrieve pension details by the dashboard user[12]	This policy explicitly consents to access by the user's dashboard(s). As part of their authorisation policy, the user will give explicit delegation consent to their persona at one or more dashboards. *Duration of consent:* typically 90 days, renewed when user re-visits the consent and authorisation service.	consent and authorisation, dashboards	Even if a user does *not* consent to enabling a specific dashboard to access pension details, there is still potential value following a find: • the user could look up the provider associated with the pension and contact the organisation in other ways • the user could delegate to a financial adviser • they could delay deciding which dashboard they will use, or add additional dashboards • they could change / add dashboards without re-finding or contacting her existing dashboards

[12] In general, identity assurance at the dashboard will be lower than at the central identity service, so the user's persona at the dashboard is delegated access based on the policy of the higher assurance policy owner at the consent and authorisation service.

Consent to	Meaning	Relates to components	Comment
Retrieve pension details by financial adviser/guidance delegate	As part of the user's authorisation policy at the consent and authorisation service, they will give explicit delegation consent to their selection of (human) delegate(s) - adviser/guider - to access one or more of their pensions. *Duration of consent:* typically three months, renewed when user visits the consent and authorisation service.	consent and authorisation service, dashboards	The user may have more than one delegate (financial advisers or guidance officials). Financial advisers will have to register separately with the governance register, so that users can select their chosen financial adviser for delegation. A financial adviser, who is a servicing agent, may also be a delegate, although their status as a financial adviser is quite separate and unknown to the service or the user's pension provider(s).

2.5 Component interaction diagrams

This section presents sequence diagrams, which illustrate the flow of the focus of the user as they undertake the two processes in the previous section. These diagrams present the logical flows (detailed requirements for profiled UMA flows are presented separately).

To illustrate these flows, we have created a persona for the user, Alice and one for a financial adviser, Bob.

Note that we denote the user persona according to the level of assurance of their identity: **a**lice (lower case a) is the lower assured persona at a dashboard, while **A**lice (capital A) is her more highly assured identity at the consent and authorisation service. We use **b**ob and **B**ob similarly, as relating to the persona for a financial adviser or guidance agent, at his client dashboard and at the consent and authorisation service respectively.

The solid red horizontal lines show what the user will see. The dotted horizontal lines are system to system interactions (APIs etc). The vertical lines are entities of the ecosystem, which interact according to the horizontal lines.

2.5.1 User journey – find

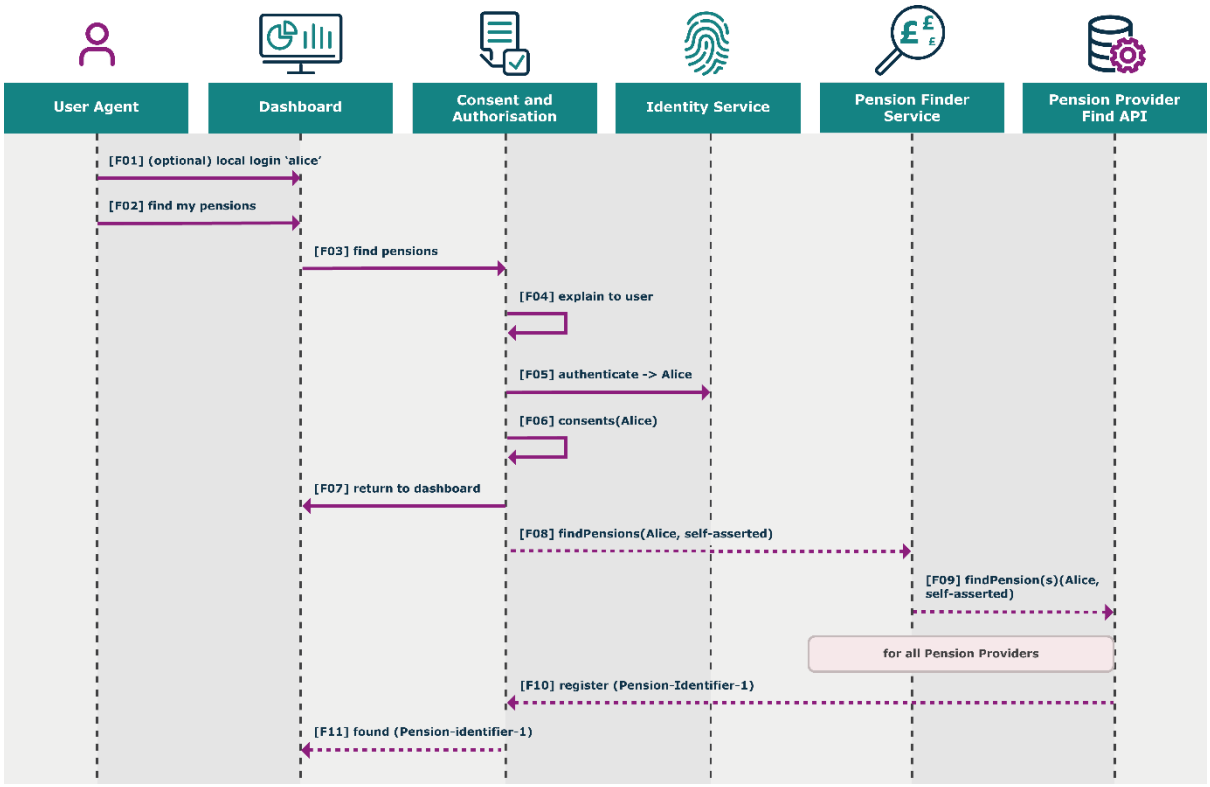

Diagram three: user journey – find pensions

Step	Explanation	Notes
F01	If the dashboard wants an account of its own, it will arrange for its user, alice to login locally to her dashboard. Little 'a' alice has the lower assurance persona 'alice@dashboard.co'.	If the dashboard is to keep the pension identifiers (PeIs) and tokens[13], there will need to be an account at the dashboard.
F02	User requests find.	
F03	User is *redirected* to the consent and authorisation service user interface.	The user interface will comply with the design standards as defined by the Pensions Dashboards Programme.
F04	Central explanations given to the user.	The user will probably also be requested to provide a self-asserted NINO in step F04.

[13] A token is a digital entity that represents some (usually secret) information about a process or activity. Here it represents the information 'authorisation to retrieve pension details' when quoted along with the correct corresponding identifier.

Step	Explanation	Notes
F05	'alice' is required to prove she is 'Alice' – to the ecosystem standards. Once proven, Alice will have an ecosystem standard identity Alice@identityservice.uk	*If* the dashboard has already used the ecosystem identity service to authenticate its customer, Alice, in this session then she will not see this step again, providing the identity service provider keeps her session open. See also **2.6.2**. Protocol is OpenID Connect (OIDC) from consent and authorisation service to the (federated) identity service.
F06	Alice is asked to consent to a range of activities (see section **2.4** above).	Various options exist to limit consents if Alice so wishes.
F07	Alice is returned to the dashboard (ie reversing the initial redirection at F03).	Where she waits for the other asynchronous processes to complete, finding and returning pension identifiers to her dashboard[14].
F08, F09	The pension finder service works its way through data provider find interface instances.	Response times from each are likely to vary.
F10, F11	The pension identifier is made available to the dashboard, via the consent and authorisation service, after registration by the pension provider, depending on user consent. Pension identifiers are usually kept at the dashboard (if the dashboard has a suitable account for alice) to retrieve pension details, whenever is convenient.	As each pension is found at a data provider, it is registered[15] with the consent and authorisation service. The technical details of this process are covered in **6.3** and elsewhere. The technical details of the consent and authorisation service giving PeI to the dashboard (step F11) are covered in **4.1.2** and elsewhere. The user at the dashboard can start the view process whenever they want, as soon as the dashboard has at least one pension identifier (see **2.6.1**).

[14] Think an insurance quotation web site or a utility switch web site. If data providers take some time to identify pension assets belonging to the user, the results may be available only in a subsequent user session.

[15] In the find flow, step F10, the pension provider's find process results in the pension identifier being registered with the consent and authorisation service and subsequently returned to the dashboard.
The newly found pension identifier has to be registered, alongside Alice's consent policy, at the consent and authorisation service, so that she, or her delegate, can subsequently be authorised to access to it. The details of the registration step are not shown here, but are based on UMA 2 federated authorisation and discussed later.

2.5.2 User journey - view

This flow presents the interactions to enable a dashboard user to view their pension details.

Note that:

- the flow is entirely independent of the data payload, ie it is the same no matter what decisions are made concerning pension details data standards
- the pension finder service plays no part in this flow: its function is solely to orchestrate the find requests to the pension provider interfaces and that function is not relevant, given the dashboard has pension identifiers in this flow
- this flow uses the consent and authorisation service to authorise *each* separate view request, ie each pension identifier is *separately* authorised and has *separate* tokens representing its authorisation issued by the consent and authorisation
- service, even if there is more than one pension identifier hosted at the same data provider
- this flow is a profile of User managed access 2, which enables a central authorisation service to handle user specified access control policies for a range of resources, irrespective of the distributed nature of those resources. (Here the resources are represented by pension identifiers, and are located across one or more data providers)

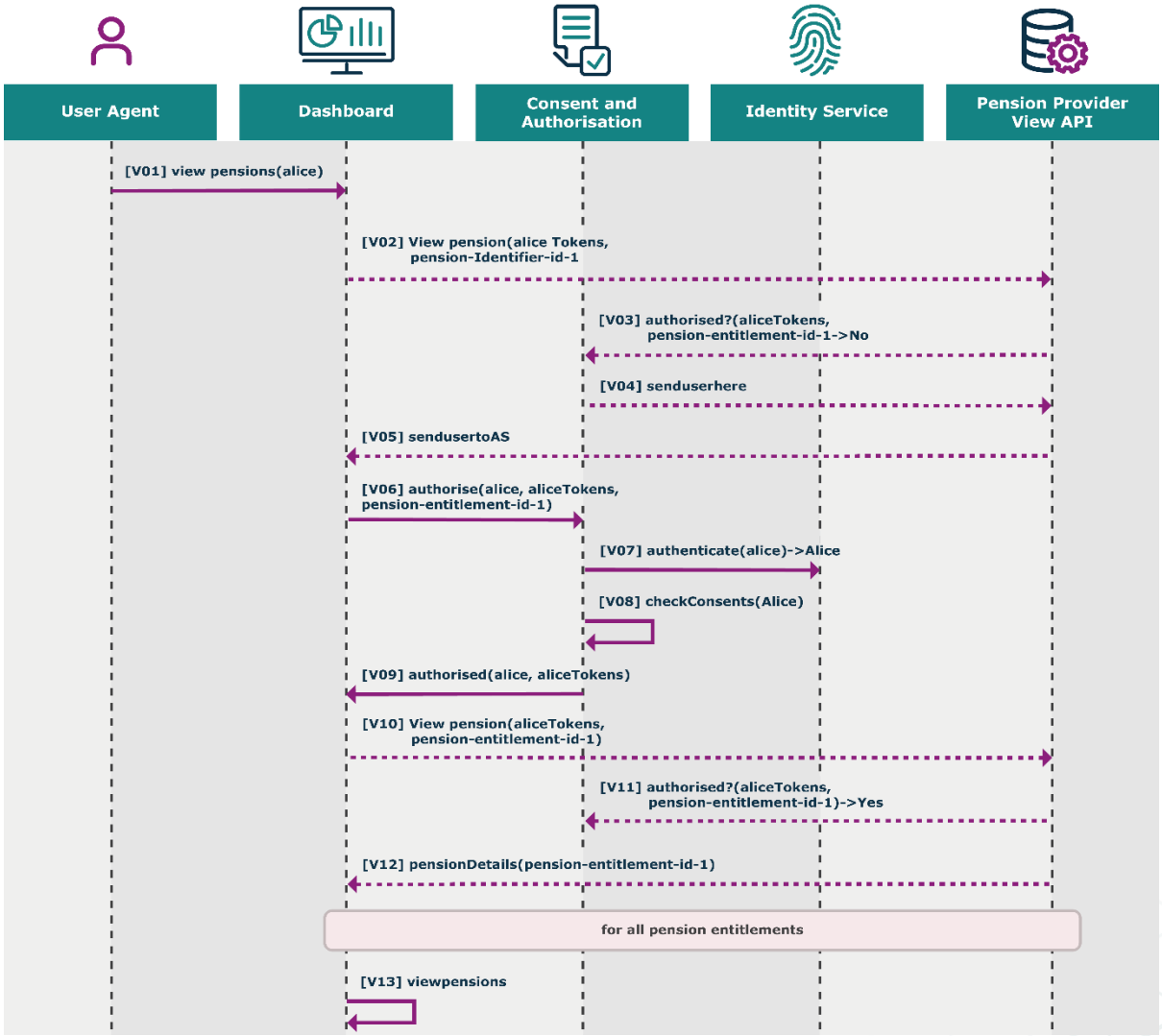

Diagram four: user journey – view pensions

Step	Explanation	Notes
V01	alice wants to view her pension(s). The dashboard has at least one pension identifier (from the find process or otherwise).	It doesn't matter whether this flow occurs immediately after find or sometime later, the flow is the same.
V02	The dashboard asks the relevant data provider view interface for the details, based on the pension identifier (PeI) and any tokens it has stored alongside the identifier.	In the first instance it will have no tokens.
V03	The pension provider view interface asks the authorisation service if the access can be granted. Initially, here, it cannot...	Initially there will be no stored tokens which is why this step initially returns 'No'.[16]
V04	... because the authorisation service needs to check the user and the relevant authorisation.	
V05	So, the dashboard is told to send (redirect) the user to the consent and authorisation service	
V06, V07	Which the user sees as a request to authenticate as Alice.	This step will only happen if the user has not already authenticated as Alice in the recent past. (If so, eg the retrieval is happening immediately after find, the user will not see this step, although the redirection to the identity service may occur silently.)
V08, V09	The consent and authorisation service authorises the access and (re)issues 'Alice tokens' to the dashboard, V09	
V10, V11	Which the dashboard can use to *retry* the failed call at V02, for which authorisation checking succeeds at V11.	This call succeeds because the information to perform the authorisation is now available and represented in the Alice tokens.
V12	So, now that authorisation has occurred, the pension details of pension-identifier-1 are returned to the dashboard,	And, unless Alice has set a different consent policy for her other pensions, these will also be authorised *without her further user interaction*, each being issued with associated tokens (ie each instance of V11 will succeed).

[16] It would also return 'No' if any other aspect of the authorisation was not proved, for instance the tokens had expired, or the owner had withdrawn her consent.

Step	Explanation	Notes
V13	Which the dashboard can show the user (subject to possible content and format rules outside the scope of this document).	

2.5.3 Consent interactions

As was noted above, in section **2.5.1**, User journey - find, at step F06, the user, having proved that she is indeed Alice, is asked to give consent to possible uses of her personal data by the pensions dashboards ecosystem. Various types of consent are discussed in section **2.4** above.

Minimally, she consented to her personal information being disclosed to some, or all, data providers for the purposes of finding whether she has pension(s) held with each provider and to the resulting pension identifiers being registered at the consent and authorisation service. Usually, she will also consent to her pension details being returned to the same dashboard that she used to initiate the find operation, and consented to her persona (alice@dashboard.co, on behalf of 'Alice' Alice@identityservice.uk) subsequently retrieving pension details from the relevant provider(s).

She can revisit the consent and authorisation service at any time, potentially independently of the dashboard, to modify her consents and to issue new consents. An example could be that Bob, her financial adviser, is permitted to access pension details associated with one or more of her pension identifiers. Separately, she may revoke consents for any pension identifier, for any dashboard she has used, or any adviser she has previously granted consent.

Alice can visit the consent and authorisation service directly, through a direct URL, or via any dashboard to manage her consents.

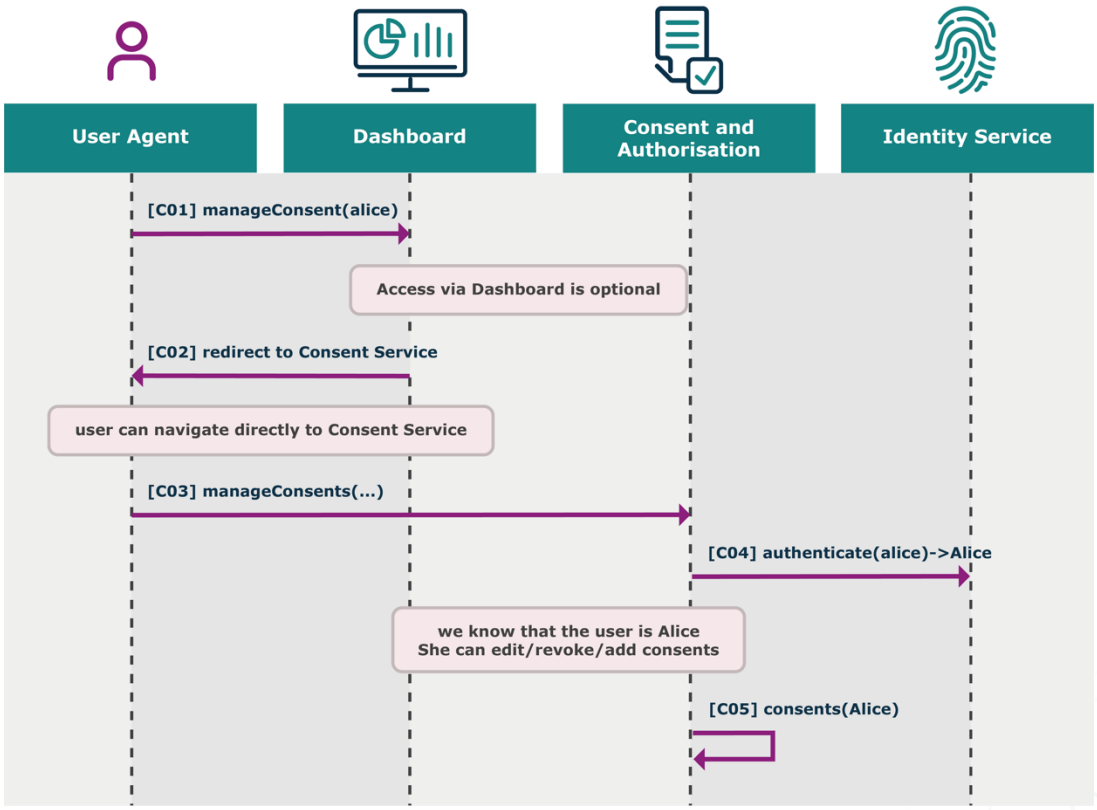

Diagram five: user journey – manage consent

Step	Explanation	Notes
C01, C02	alice wants to change or add her consents. She may go to her dashboard which, C02, is redirected to the standard user interface of the consent and authorisation service.	The user interface will comply with the design standards as defined by the Pensions Dashboards Programme.
C03	Or, she may navigate directly to the consent and authorisation service.	
C04	The consent and authorisation service must prove her identity before permitting changes to consents.	Changing consents is a sensitive operation, so the user needs to reauthenticate to ecosystem standards.
C05	She can edit, revoke, or add any consents now.	This activity also happened at F06, but in an abbreviated form.

2.5.4 Adviser interaction diagram

This section presents flow sequences for a financial adviser, as they undertake the process of viewing a client's pension details. (Advisers are not permitted to use find processes.)

The adviser view sequence is almost the same as that for the pension owner viewing their assets.

Adviser bob will probably have a different type of dashboard, but this is not architecturally necessary. bob will have to prove he is Bob, by the identity service, to the level of assurance decided by the dashboard ecosystem, and he will have to prove his professional credentials either by the specialised identity service or otherwise[17].

His dashboard uses the pension identifiers that relate to his client, Alice's pension details. She has previously arranged for him to receive her pension identifiers and store them in his dashboard. There are any number of ways to achieve this transfer. Alice could send them by email (having copied them from her dashboard, or used dashboard features to export them); she could ask the consent and authorisation service to send them, when she gave consent for Bob, the financial adviser, to be her delegate; or otherwise[18]. In any case, the pension identifiers are not secret and encode no information about Alice, nor about the pensions themselves[19].

---

[17] Perhaps via an API to the FCA or to the pensions dashboards ecosystem governance register of financial advisers. (To be decided.)

[18] The consent and authorisation service might expose a protected resource (itself under delegated access control) by which an authorised requesting dashboard can obtain the PEIs directly, see 4.1.2.

[19] The design and universal use of pension identifiers (PEI) is critical: the standard will be used by all ecosystem participants. The ecosystem needs opaque, de-referenceable, non-PII-containing identifiers, which uniquely identify a pension. These are URIs - they encode which provider manages the pension and, when dereferenced, are a resource (a URL) so that dashboards can directly access that resource. As discussed in this document and in more detailed design, PEIs are generated by pension providers according to standards and are registered at the consent and authorisation service. See section 4.3.

Note that *outside* of the technical architecture and *outside* the control of the ecosystem, there is existing regulation that limits what financial adviser Bob is permitted to do with his client's information,[20] once he has exported it from his dashboard, and that this arrangement will have to be in place to establish that Bob is Alice's financial adviser.

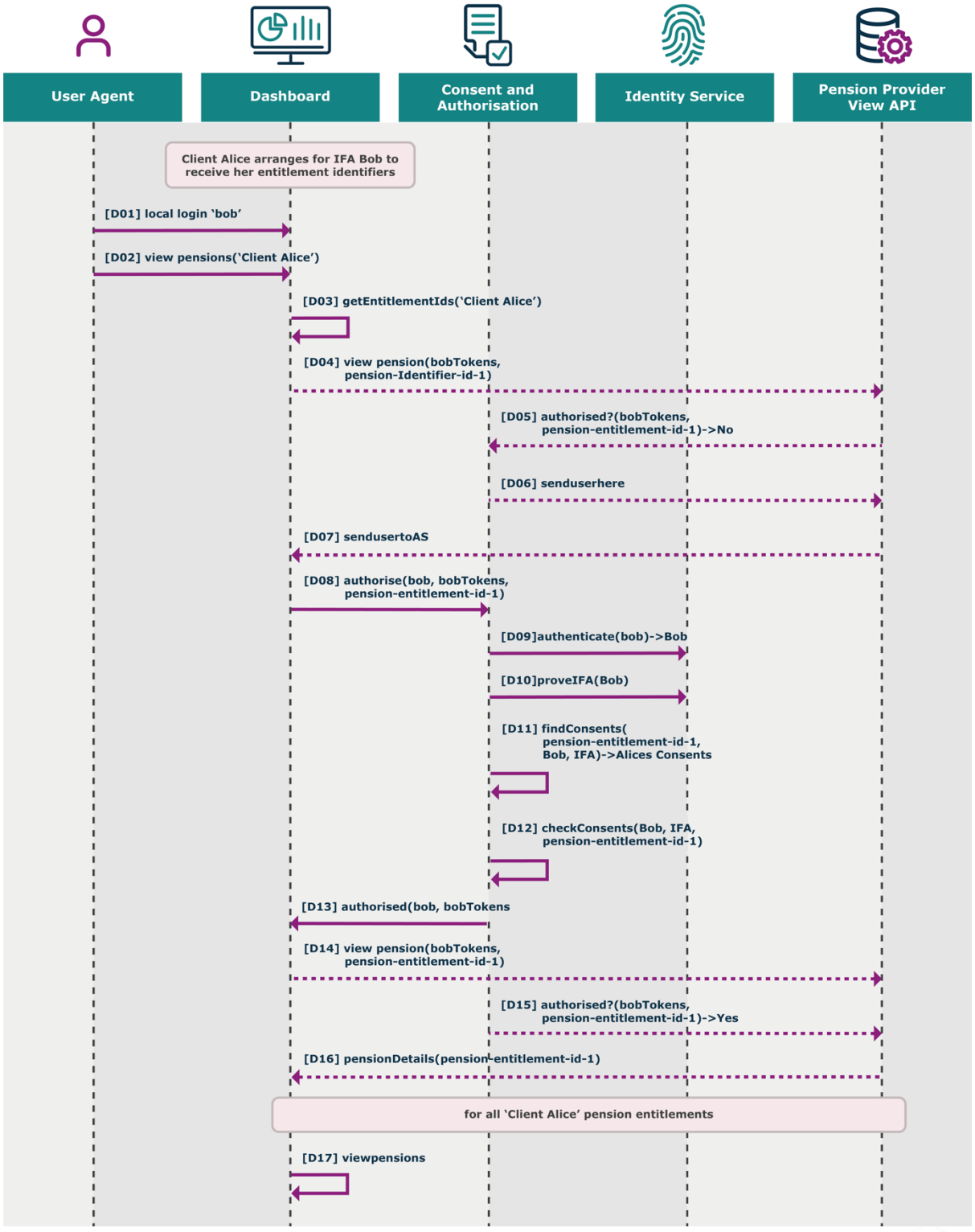

Diagram six: user journey – delegate view pensions

[20] Dashboard software for advisers will have to comply with whatever rules are imposed by the ecosystem and its regulators, but these rules will presumably permit Bob to copy Alice's data out of the dashboard, in accordance with his contract with Alice and regulatory requirements.

Step	Explanation	Notes
D01	bob wants to use his local account on commercial off the shelf (COTS)/software as a service (SaaS) dashboard software for financial advisers. His client, Alice, has signed an appointment letter – outside the pensions dashboard eco-system.	As a client of her financial adviser, Alice has previously established her policy at the consent and authorisation service, enabling Bob the financial adviser to have delegated access to her pension details for (some of) her pension identifiers. See delegate consent in section **2.4**.
D02	Part of bob's tasks for his client is to view her pension(s).	
D03	Typically, adviser dashboards will store the pension identifiers of several clients.	This step is just to emphasise that financial advisers will have PeIs (and tokens) for potentially many clients.
D04	The dashboard asks the relevant data provider view interface for the details, based on the pension identifier and any tokens it has stored alongside each PeI (which will be bob's token for each identifier).	Initially there will be no stored tokens which is why this step initially returns 'No'.
D05, D06	The data provider interface asks the consent and authorisation service if the access can be granted.	
D07	Initially, here, it cannot. So bob's dashboard is redirected to the consent and authorisation service.	Rules governing dashboard operators require compliance with ecosystem level behaviour, such as redirection as required, and compliance with authorisation protocols and so on.
D08	The consent and authorisation service needs to authorise (or otherwise) that bob may access the quoted pension identifier.	
D09, D10	First it requires that bob is really Bob, the financial adviser, by the identity service D09, D10.	Identity service or otherwise as mentioned above.
D11	bob's dashboard provided a pension identifier (PeI), which the consent and authorisation service can use to determine whose authorisation policy to use, here it finds Alice's policy for that identifier.	Note Alice's PeI keys, so *her* policy.

Step	Explanation	Notes
D12	The consent and authorisation service checks Bob the financial adviser is permitted by Alice to access that pension.	Alice must have granted Bob the financial adviser permission to access her pension(s) at some point before Bob attempts to access them (see note on D01 in this table, or C05 in **2.5.3**.)
D13	As a result, bob is given authorisation, and tokens to keep with the pension identifier.	The consent and authorisation service authorises access on the basis of Alice's policy - consent to financial adviser Bob to view her pension(s). It can also coordinate with the governance register to monitor delegated accesses.
D14, D15, D16	Which his dashboard can then use to retry access (as per D05), this time successfully (D15) to obtain pension details from the provider.	
D17	The adviser's dashboard shows bob the pension details.	Note that, given Bob the financial adviser has a *contract* with Alice (outside the pensions dashboards ecosystem), he may have been granted consent to perform other operations on her data than simply viewing it, presumably after exporting that data from his dashboard. Control and regulation of this is outside the pensions dashboards ecosystem, and we assume that financial adviser software complies with those regulations in respect of other potential uses of Alice's pension data.

2.6 Discussion of design options

This section raises design options relating to the find and view journeys above.

2.6.1 Dashboards – may use the ecosystem identity service

User experience if dashboard uses the ecosystem identity service. It is also worth emphasising the note at step F05 of the find flow (section **2.5.1**). The above design *does also* support dashboards that choose to use the *same* identity service as the consent and authorisation service. That is, steps F01 and F05, while remaining distinct steps, could use the same ecosystem identity service.

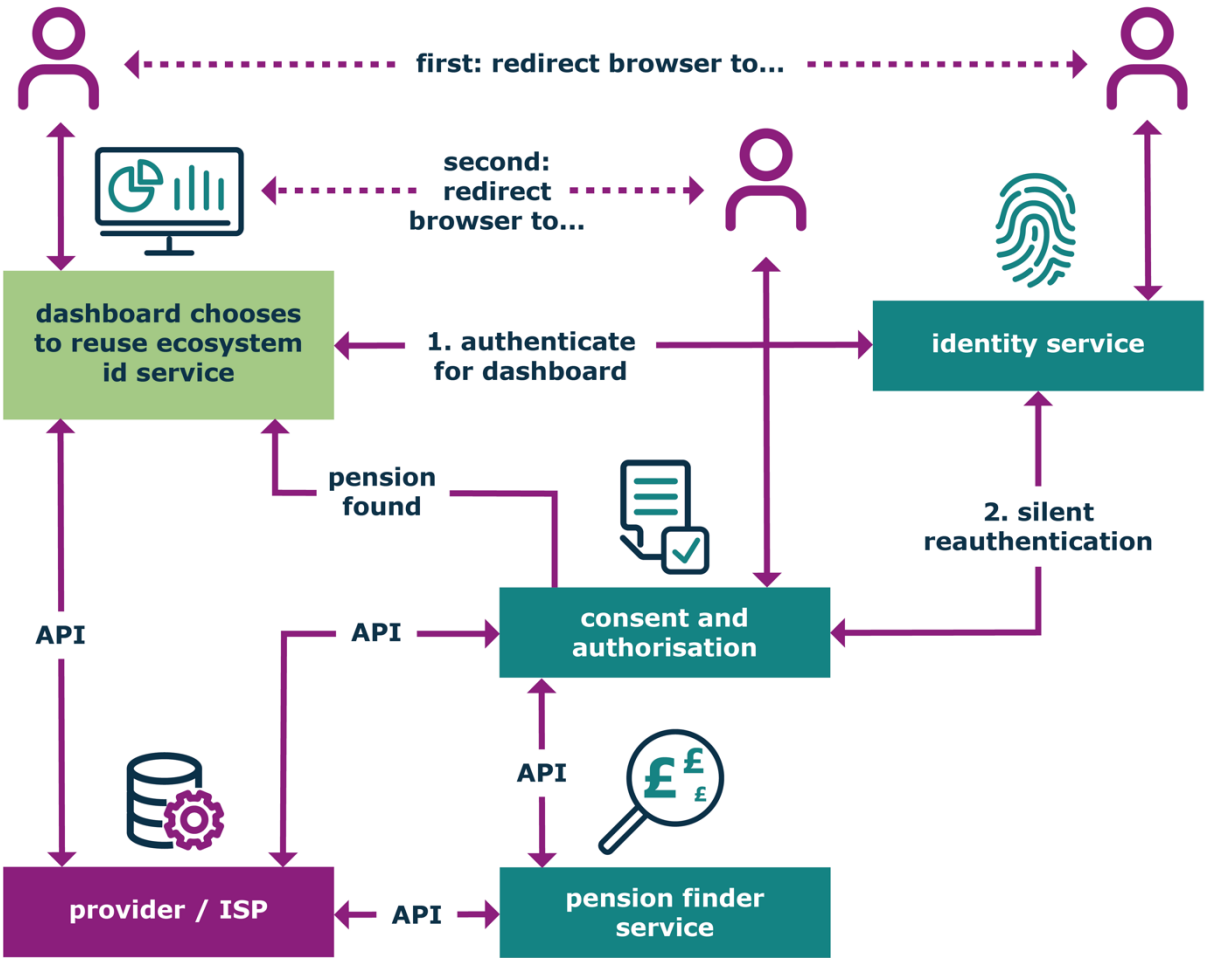

Diagram seven: user journey – dashboard uses the ecosystem identity service

In such a case, the dashboard user authenticates as **A**lice at the dashboard and the consent and authorisation service, at step F05 can re-prove this is **A**lice, with the identity service. This can be done silently, without the user being aware, if the identity service supports appropriate open sessions, as is usually the case[21].

2.6.2 Dashboards – user accounts or not

User experience if dashboard does *not* have a user account. It is also worth emphasising re step F01 of the find flow (section **2.5.1**). If the dashboard is going to keep the pension identifiers (PeIs) and tokens, there will need to be an account at the dashboard. The account can be tied to a portal or a phone or anything else that the dashboard operator wishes.

It is left open for a dashboard operator to implement a dashboard where the contents last only for a single user session, ie a dashboard with no local account. In such a case, there can be no persistence (of PeIs, tokens, or anything) beyond the session.

---

[21] Note GOV.UK Verify chose to explicitly force identity providers to close sessions immediately after initial authentication. This will have to be resolved for the Programme's selection of identity service. An OIX project used one of the Verify identity providers and slightly modified their configuration to support open sessions for this purpose, enhancing user experience greatly see https://openidentityexchange.org/networks/87/item.html?id=176 (formerly https://oixuk.org/wp-content/uploads/2017/06/OIX-White-Paper-Digital-ID-for-Pensions-Dashboard-Final.pdf).

As a consequence, the user will have to reauthenticate at the consent and authorisation and identity service for every visit to the dashboard service. The dashboard will have to retrieve the user's PeIs from the consent and authorisation service on each visit too, repeating the step at F11 each time – see **4.1.2** (or, far less desirably for both user and the ecosystem, performing a find operation each time).

2.6.3 Data providers and dashboards – maybe found

Each pension provider is responsible for determining how they utilise the matching data set to find a user's pension information.

Where the data matches in accordance with the pension provider's rules, it can be considered an exact match. However, there will be circumstances where the pension provider may believe they have a match but is not certain: we class this as a maybe find.

We recognise that there is value to the customer in taking steps to resolve the maybe match, however, this architecture does not provide direct facilities to resolve a maybe match.

To support resolution of maybe matches, the architecture will enable the data provider to register a PeI at the consent and authorisation service for the pension in question, and when subsequent access (standard view via pension provider view interface) is performed, the consent and authorisation service will authorise the access, as normal.

This will enable the pension provider to provide a maybe payload, in accordance with the Pensions Dashboards Programme data standards.

This may ask the user to, 'contact us at ... using reference REFN'.

The pensions dashboards ecosystem architecture is payload agnostic, and the data standards can potentially support variations in payload.

The consent and authorisation service can interact with the data provider registration interface to deregister the PeI, should the pension provider determine the potential match was not correct.

3. Pension provider interfaces

This section describes the interfaces which data providers must expose and presents the high-level design of those interfaces.

3.1 Diagram showing data provider interfaces

The two interfaces for data providers are shown at the lower part of diagram eight, as find and view. These are discussed in detail in the following sections. Here we note the high-level principles and how diagram eight relates to the flows earlier in this document.

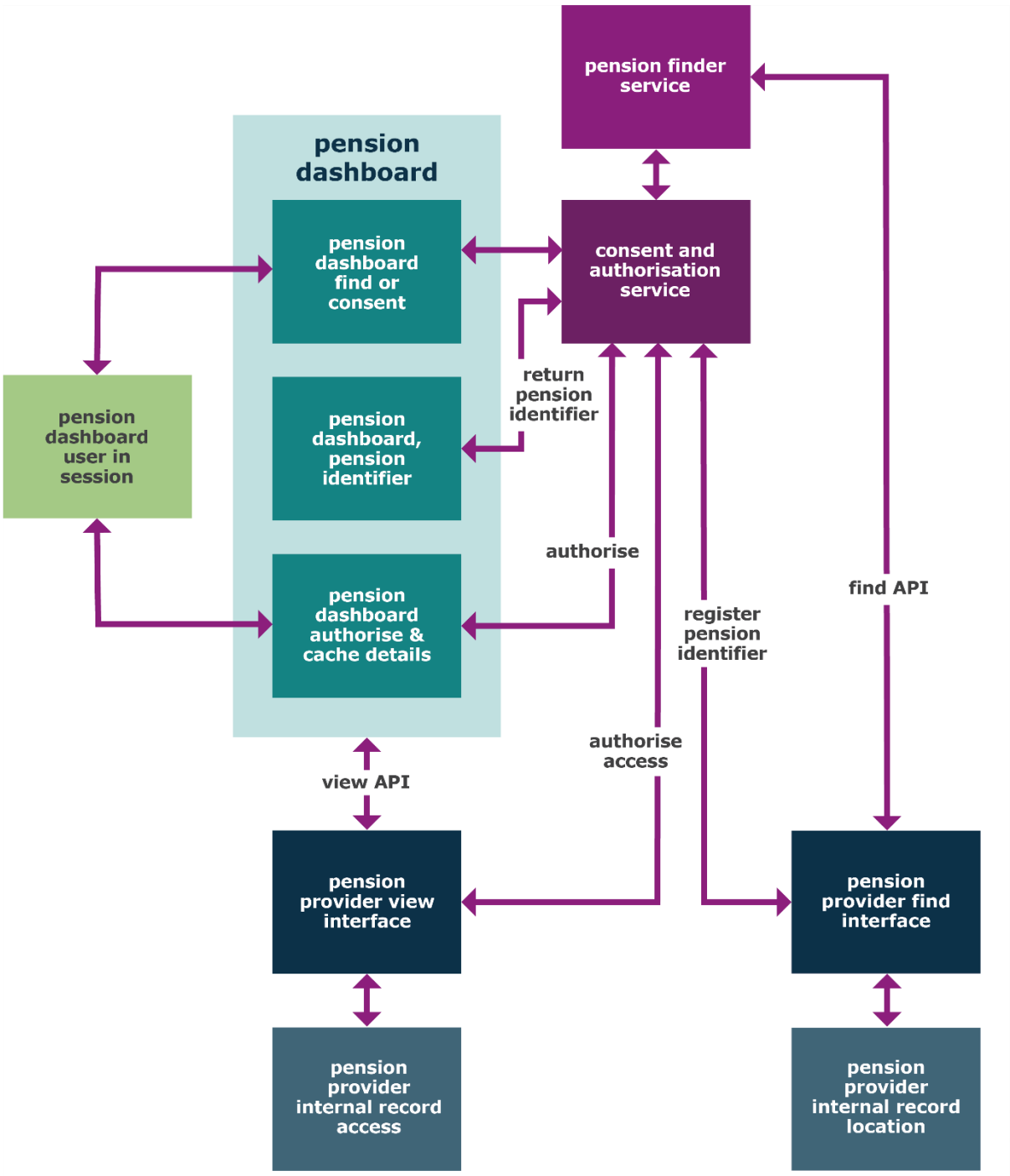

Diagram eight: data provider interfaces

Data providers are only permitted to interoperate with ecosystem components once they have fulfilled their onboarding requirements, as specified by the ecosystem governance framework.

3.1.1 Pension provider find interface

- pension provider find is invoked by the pension finder service, carrying the matching data set (ie personal data attributes), which consents (plural) have been granted by the pension owner and control information
- pension provider find returns success as an ACK, a digital acknowledgement, to tell the pension finder service that it has received the request. (Note this does not mean found, rather it simply acknowledges that the pension provider's find interface has received the message. The pension finder service operates a back off and retry mechanism to handle an absence of responses, for instance if the pension provider's find interface is down or busy)
- pension provider find starts internal matching operations (a provider responsibility) to determine if the provider has records, ie pensions, which are owned by the person identified by the matching data set, and then awaits results from the internal find processing
- if a pension is found, the pension provider find creates the pension identifier(s) and registers it with the consent and authorisation service

3.1.2 Pension provider view interface

- pension provider view is invoked by a dashboard. Specifically, a 'GET' instruction to the URL that was created by dereferencing the PeI, *is* the access request. This should also carry an authorisation token from the consent and authorisation service
- pension provider view coordinates with the consent and authorisation service to validate the tokens, or, if the tokens are absent or invalid, to initiate the authorisation process to obtain a new valid token
- if the request is accompanied by a valid token, the pension provider view interface retrieves the relevant pension details from its internal system and returns the token to the dashboard

The Pensions Dashboards Programme will create a reference build for each interface that will be available for review, which will support data providers efforts to create appropriate interfaces within their own environments.

3.2 Find and register, and view interface requirements

The table below presents simplified high-level requirements for the pension provider (PP) interfaces.

Function	Requirement – what	Rationale – why
Find	Receive a find request for pensions based on supplied verified attributes of identity and self-asserted claims (eg NINO) and consent and control information (ie account details at the consent and authorisation service) received from the pension finder service. Forward the find request for internal processing.	Find processing by the pension provider requires integration with inbound identity and other attributes. If successful, the pension provider find interface needs control information to enable registration at the consent and authorisation service.
Find	Match records internally confidently, based on the matching data set.	Fidelity of matching, eradicate false positives (and avoid false negatives if possible).
Find	Following internal processing, generate and register pension identifiers with the consent and authorisation service (using the pension owner's account details at the consent and authorisation service). Keep the registration information at the provider.	To enable the individual pension owner, or their delegate(s), to maintain policy and retrieval. Pension provider find interface (as an UMA resource server) registers PeIs, as protected UMA resources, according to UMA federated authorisation with the consent and authorisation service (as the UMA authorisation server). For use by the view authorisation protocol.
Find	If no pensions are found, received personal identity attributes will need to be deleted and replaced by a hashed value that represents the request. No personal data will be recoverable and the pension provider will have a mechanism for not repeating negative search requests.	Protect privacy yet minimise repeat processing of find requests. Optional – the pension provider can delete attributes completely, but it will then have to process all finds.

Function	Requirement – what	Rationale – why
View	Receive view requests as a dereferenced PeI – ie a URL to the asset, which was registered with the consent and authorisation service during find, and perform authorisation decisions according to protocol with the consent and authorisation service. Forward authorised request for internal processing.	PeI URL by itself is not proof of authorised access; the pension provider view interface (as an UMA resource server) authorises access, according to UMA grant with the consent and authorisation server (as the UMA authorisation server). Internal records at provider contain pension details.
View	Return pension details to the pensions dashboard that initiated the view request.	Interface is payload agnostic, but returns details to the pensions dashboard.
Governance	a) delete the pension owner's data related to the pensions dashboards ecosystem at the pension provider on request (from the consent and authorisation service or otherwise). b) delete the pension owner's data related to the pensions dashboards ecosystem at the pension provider after a significant period of non-use.	DPA 18/GDPR Right of erasure. Limit of holding data for a defined purpose.
Governance	Meet regulatory and monitoring requirements, by interface logging to central services.	Meet ecosystem wide requirements by integrating with ecosystem services.
Governance	Register pension provider software as required by the pensions dashboards ecosystem (governance register).	Integrate registered software instances by approved mechanism.
Governance	Implement the responsibilities of the pensions dashboards ecosystem trust framework, which apply to pensions providers.	Compliance with the business, legal, technical and ethical rules is assured.

4. Dashboard interfaces

This section describes the interfaces that dashboards must implement and presents the high-level design of those interfaces.

The dashboard has interactions with data providers (the data providers - view interface) and with the consent and authorisation service.

In addition to the consent and authorisation service's authorisation APIs (not shown in diagram nine, below) the dashboard will:

- redirect users to the consent and authorisation service for a variety of reasons (authorisation, find, consent management)
- obtain PeI and associated data, which result from find processes
- retrieve pension details from data providers

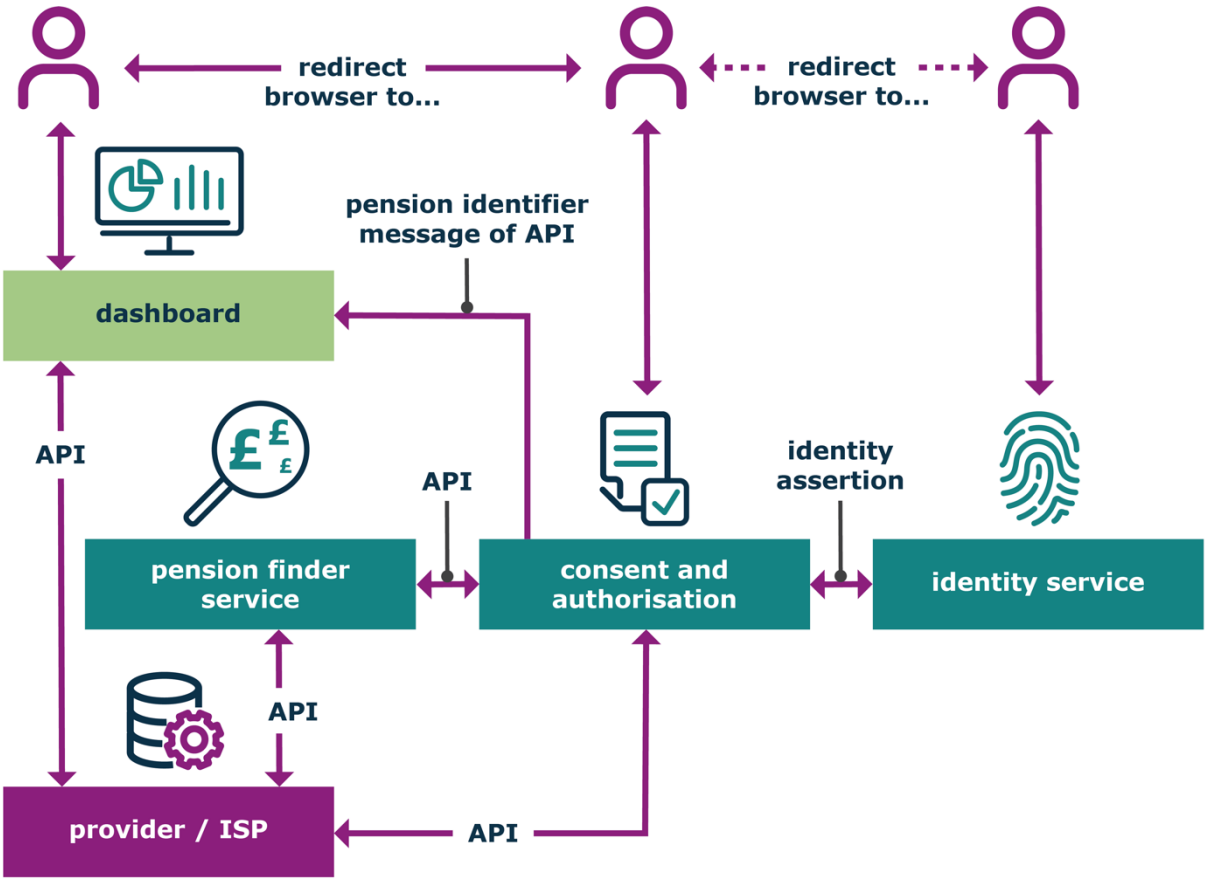

Diagram nine: dashboard user interfaces

4.1 Diagram showing dashboard interfaces

This diagram is mostly the same as that above for data providers. The identity service is added for clarity and here we focus on relationships with respect to dashboards.

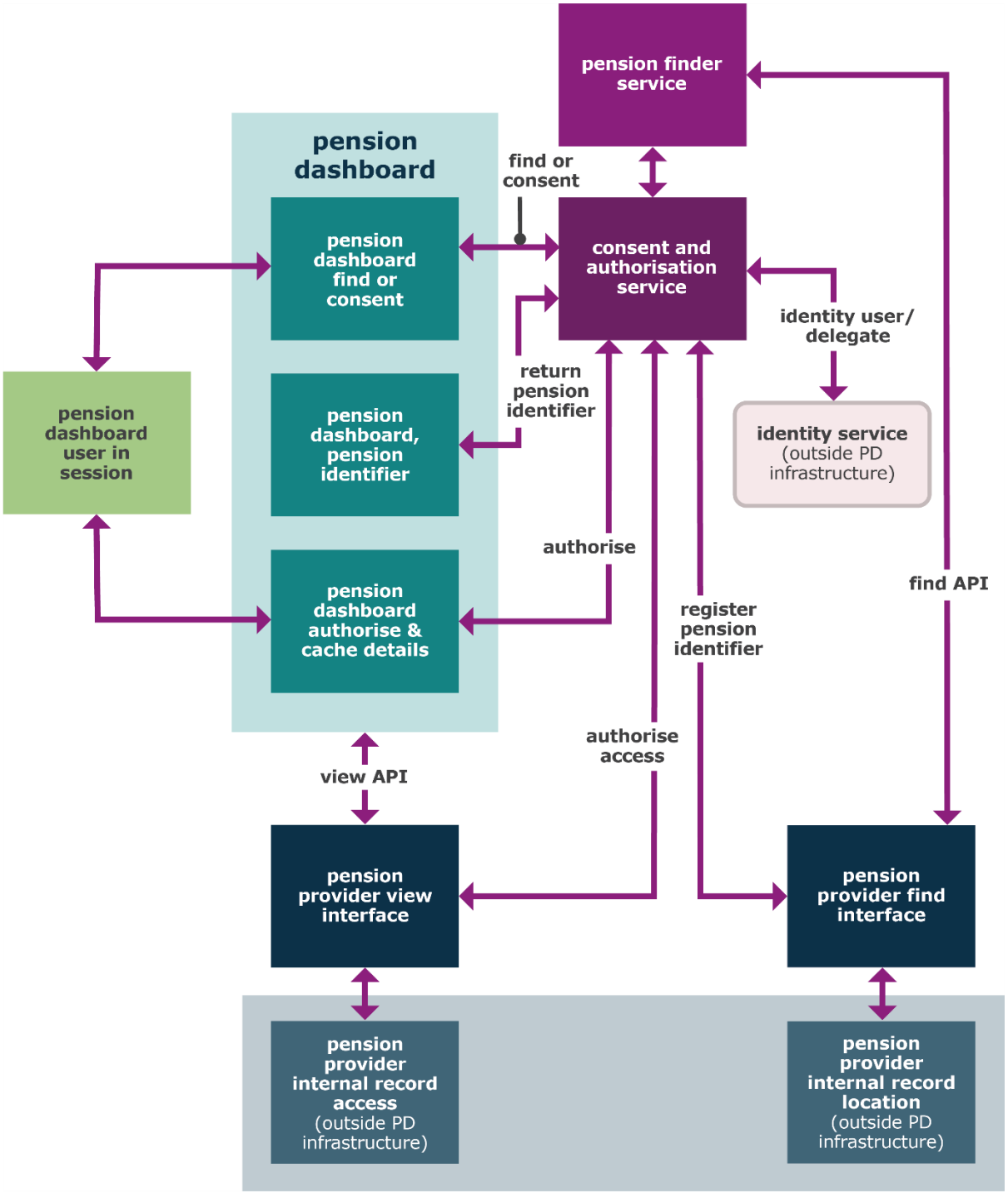

Diagram 10: dashboard interfaces

Dashboard operators are only permitted to interoperate with ecosystem components once they have fulfilled their onboarding requirements, as specified by the ecosystem governance framework.

All pensions dashboards ecosystem components, including pensions dashboards, will have requirements imposed by the governance framework (to meet regulatory, reporting, security and compliance needs). This includes monitoring information, which will be provided by pensions dashboards to the governance register.

Some of the functions associated with integration with the ecosystem components, will be provided to dashboard operators in the form of a reference implementation, which may reduce implementation and testing time.

4.1.1 Pensions dashboards: find or consent

A pensions dashboard will redirect the user to the consent and authorisation service to its find user interface or consent user interface, when appropriate.

See **2.5.1** for discussion of find and **2.5.3** for a discussion of consent, noting that the consent user interaction will be included by the consent and authorisation service in the user's find journey as is appropriate.

After user interaction with the consent and authorisation service, the user will be returned to the dashboard, via a redirection call back (eg find step F07). For the avoidance of doubt, these consent and authorisation functions return neither pension identifiers nor consent information, only status and access control information, as is determined to be necessary to manage the user interactions at the pensions dashboard and the pensions dashboard's access to the PeIs.

4.1.2 Pensions dashboards : manage pension identifiers (PeIs)

When the pension finder service has issued find requests to relevant providers, and the relevant pension provider find interfaces (PPFI) have located and registered pension identifier(s) with the consent and authorisation service, the consent and authorisation service will make these available to the dashboard, via a pensions dashboard initiated API (pull from pensions dashboards) and perhaps by a messaging service (push to pensions dashboards).

A representation of the pension/scheme name and its customer identifier in user friendly terms, as determined by the pension provider, is associated with each PeI. We assume that pensions dashboards will use this pension identifier as the item for display to its user. The pensions dashboard will usually persist (hold on to) PeIs and related information, such as user descriptions and later access tokens, so that the user can recover them in their next session.

Obtaining pension identifiers (PeIs) from the consent and authorisation service

A pensions dashboard may access a user's PeIs, if authorised to do so by the user. Authorisation will usually occur when the find operation has returned to the dashboard for the first time, in which case, the dashboard may immediately check for resulting registered PeIs.

However, the pensions dashboard does *not need* to have initiated a pension finder service find operation to request PeIs; the user may have performed a search via a different dashboard, or the user may be using a dashboard that has no local account (so cannot store anything between sessions).

The user may already be recognised by the consent and authorisation service (because consent and find has occurred in the current session). If not, they will be redirected to the identity service for verification, and to grant consent for the current dashboard to access the PeIs. (The pensions dashboard may have a token from previous accesses to

the current dashboard, which serves as a claim for this authentication process, reducing user friction.) The pensions dashboard user will then be authorised to access the protected resource[22] for their PeIs[23].

4.1.3 Pensions dashboards: authorise and store tokens

The pensions dashboard will comply with the protocol and security token arrangements imposed by the consent and authorisation service – cooperating with the authorisation process. The authorisation protocol, a profile of UMA, will issue security tokens, which pensions dashboards will manage and use to obtain authorised access to protected resources (PeIs) at the data providers.

Dashboards will assert a token associating their user and dashboard identifiers to the consent and authorisation service in each interaction and will receive tokens from the consent and authorisation service, which the pensions dashboard will store for use in accord with the protocols.

4.1.4 Pensions dashboards: cache pension details

When a pensions dashboard has retrieved pension details from a protected PeI endpoint at a provider, it may cache those details for purposes of enhancing the user experience during a session. It may not persist those details, nor use them for any other purpose than *display to the user*. (Precise rules will be published in the ecosystem governance framework.)

Pensions dashboards may implement features for the user's convenience, to export PeIs and/or pension details to *user controlled*[24] media. Pensions dashboards will comply with other requirements related to the deletion of data and of accounts, or of dormant accounts, and various regulatory requirements.

---

[22] That is, the consent and authorisation service exposes a protected resource, which is that user's resource for their PEIs. This should be implemented as an UMA RS endpoint (at the consent and authorisation service) because such an implementation decision would fit with the main UMA flows, including failure on access forcing a new authorisation dance, and the reuse of the UMA persistent claims token (PCT) would be natural. Clearly a plain OAuth2 (the industry-standard protocol for authorisation) protected API (application programming interface) is also a possible implementation (since the resource server (RS) and the authorisation server are local to the consent and authorisation service), but this may not integrate so well with the persistent claims tokens.

[23] This same mechanism might be used to support delegates in obtaining their client's PeIs. The owner might grant a delegate consent to the same protected resource, which can list all of their PeIs to the delegate dashboard.

[24] For example, user's local machine or user-controlled storage, pdf, text file, email client, etc, not dashboard operator storage, nor export to dashboard operator.
Specific rules or controls may exist in cases where the dashboard is specifically engineered for use by financial advisers, perhaps as part of customer off the shelf or SaaS for financial advisers and the operating organisation and the financial adviser have a contract which fully recognises the financial adviser's duties. The governance framework will address such matters in due course.

The table below presents simplified high-level requirements for the pensions dashboard interfaces.

Dashboard function	Requirement – what	Rationale – why
Identifier of dashboard user	Issue a unique assertion of an in-session user consistently across sessions, when communicating with the consent and authorisation service (for both pension owner and delegate users).	The consent and authorisation service correlates trust across sessions with a trust anchor established by the external identity service. Pensions dashboard user identity is only to provide consistency across sessions. (Requesting party token (RQP) and persistent claims tokens (PCT) – see section **6.4**).
Find and consent	Redirect to consent and authorisation to enable consent and find processing.	Support pensions dashboards ecosystem trust anchor and consents enabling find service.
Find	Cooperate with consent and authorisation to obtain PeIs. Note, it is not necessary for a dashboard to execute a find operation to obtain the PeIs from the consent and authorisation service.	PeIs enable subsequent view operations.
Authorise and view	Retrieve from provider and show pension details through dashboard.	Purpose of pensions dashboards; the act of attempting retrieval (to view a PeI) triggers authorisation.
Consent and authorise	Redirect to manage policy for user and delegate(s) to show pension details through their respective dashboards.	Dashboards cooperate with the consent and authorisation service, which manages all consent policy for the user and for their delegates.

Dashboard function	Requirement – what	Rationale – why
Governance	a) delete user's data from the pensions dashboard on request via a dashboard and redirect the user to the consent and authorisation service, so that it can confirm identity and support further deletion if the user requires. b) delete user's data from the pensions dashboard after a significant period of non-use. Professional dashboards must have appropriate controls to protect and manage customer data, including timely deletion.	Right of erasure. Limit of holding data for a purpose.
Governance	Import/export user's PeI and pension details from the dashboard.	Data portability right. Avoid search for either personal privacy / usability reasons or for delegates, who cannot use find.
Governance	Meet regulatory and monitoring requirements providing data via central services.	Meet ecosystem wide requirements by integrating with ecosystem services.
Governance	Implement the responsibilities of the pensions dashboards ecosystem trust framework that apply to pensions dashboard operators.	Compliance with the business, legal, technical and ethical rules is assured.
Governance	Register dashboard software as required by the pensions dashboards ecosystem in accord with the type of dashboard implementation.	Deliver and integrate registered software instances by approved mechanism for type of implementation.

5. Pension finder service

The pension finder service is orchestration middleware. It has *no* user interface. It distributes find requests across the data provider endpoints and manages the low-level interactions to achieve message delivery to providers.

It receives all of its *inputs* from the consent and authorisation service find process:

- verified identity details[25], such as name, date of birth, post code
- user asserted data related to finding eg NINO
- relevant consent information (see section **2.4**)
- control information for the find interface (a token issued by the consent and authorisation service, for the user account at the consent and authorisation service to be used by the pension provider find interface to *register* a found pension, see **6.3**)
- range of pension provider endpoints to be searched. Range might be all or a list of provider endpoints

It asynchronously (and in parallel) calls all data provider end points (in the range), providing the appropriate data from its input to each of those endpoints.

For each endpoint, the pension finder service expects a positive result from the pension provider find interface for each find request, which indicates receipt of the request. (Actual results of find requests are only manifest as registrations of PeIs at the consent and authorisation service.)

The pension finder service manages traffic volumes and handles data provider endpoint failures, operating a cache of find requests, a time out process for each endpoint for requests and a back-off retry process to throttle traffic.

The pension finder service logs and monitors (non-personally identifiable information[26]) elements of traffic to the governance register monitoring service, notably on its own status and the performance and operational status of the pension provider find interface endpoints.

[25] Note the pension finder service does not receive a unique identifier for the user's identity from the identity service. There is no need for such a disclosure to the pension provider.

[26] PII – personally identifiable information is data which can be related to a living person. For example, name, date of birth, National Insurance number and the like. Non-PII is used here to emphasise that audit feeds should not contain, and hence not aggregate PII.

6. Consent and authorisation service

6.1 Consent and authorisation service overview

The consent and authorisation service is the trust anchor for the whole of the ecosystem. It relies on the governance register public key infrastructure (see section **7**) for organisational trust and as the static trust relationships.

The consent and authorisation service manages registration of software entities (ie dashboards and pension provider view software instances).

It also manages all aspects of dynamic trust (ie trust needed in processing data access transactions) in the following ways:

- it seeks identity proofs when needed from the external identity service
- it seeks professional status proofs, when needed for delegates from the governance register
- it operates the authorisation protocol for the whole ecosystem
    - it consumes assertions from dashboards, which identify the dashboard instance and the current user of that dashboard - requesting party token (RQP)
    - it orchestrates the protocol across the parties (pension provider view interface, pensions dashboard and itself)
    - it issues long-lived tokens to the dashboard associating the assured identity to the dashboard identity - persisted claims token (PCT)
    - it issues short-lived authorisation tokens (RPT) to dashboards, which authorise a specific user to access a specific pension identifier
- it supports the central registration of all pension identifiers by data providers and management of initiation and periodic refresh of those registrations (data providers federate authorisation control to the consent and authorisation service)

The consent and authorisation service provides management functions so that pension owners (users) can:

- create, edit or revoke their policy for access to their pension details (PeIs) by themselves or their delegates, including such access by the pension owner in their persona(e) at one or more dashboard(s)
- unambiguously and easily select delegates. (The consent and authorisation service consent manager cooperates with the governance register in managing the identifiers of delegates)
- select pension providers within which the user wishes to search for pensions. (The consent and authorisation service consent manager cooperates with the governance register to determine which endpoints correspond to which schemes. It is also likely that supporting information will be provided via the user interface, so that a user can select based on information they may hold about the schemes)
- initiate find operations, including the provision of self-asserted data (eg NINO) and the selection of a range of data providers if the user wishes to limit the scope of a find operation. The consent and authorisation service provides functionality, so that the pension owner (user)'s dashboard(s) can obtain their PeIs after these are registered by data providers

Diagram 11, below, summarises the above functions, plus grouping the functions into the UI elements, the core authorisation server and supporting services.

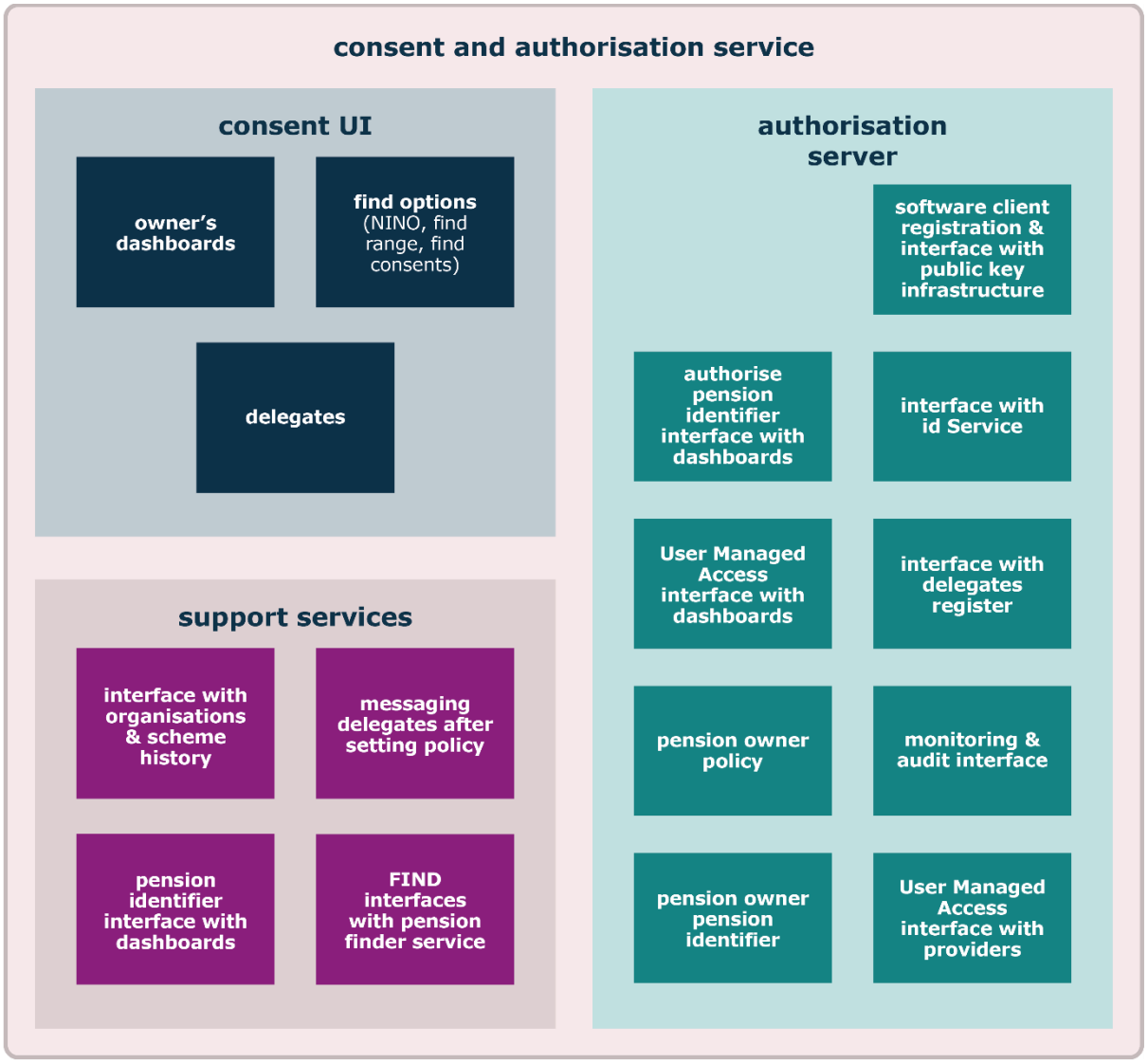

Diagram 11: consent and authorisation service functions

Note there are interfaces:

- to the governance register functions: public key interface, registers of delegates and organisations/schemes, monitoring & audit
- to the pension finder service: find interface
- to data providers: software client registration, UMA interface, pension owner PeIs
- to dashboards: software client registration, PeI interface (authorise and the API/messaging service itself)
- to delegates: messaging the pension owner's PeIs after they establish a delegation policy
- to users: consent UI (which gives access to pension owner policy, pension owner PeIs, own dashboard policy, delegation policy, and enables user entry of find options)

6.2 Trust anchor and level of confidence in identity

Data providers must trust the consent and authorisation service's assessment of the identity and the authorisation decisions it makes. The ecosystem governance framework will define the business, technical and legal arrangements for providers and participants in general.

The identity service is not connected with either data providers or with dashboards in the core architecture. Data providers do *not* have any interactions[27] with the identity service.

It is the consent and authorisation service that determines when and how to prove the identity and professional status of the dashboard users. It enforces (via the identity service) the level of confidence in an identity, from which the provider receives the verified attributes in the matching data set. *It is this level of confidence (as proven to the consent and authorisation service) that gives data providers assurance that they can disclose pension details to the identified user*. Data providers do not receive the assertion of identity itself, only the matching data set, which includes verified attributes from the identity service.

Dashboard operators *may* use the ecosystem identity service as was presented in section **2.6.2**.

6.3 PeI registration and maintenance

When a data provider finds a pension, it associates that pension with a pension identifier (PeI). The provider must register the PeI with the consent and authorisation service, so that the user managed access (UMA2) authorisation server (AS), which is a component of the consent and authorisation service, can subsequently authorise access to that pension in accordance with the policy established by the pension owner.

To register on behalf of the pension owner, the data provider exchanges a temporary credential, representing the owner at the authorisation service, provided to it during the find operation, for a long-lived access token (an UMA PAT) for that user. Registration places the PeI under UMA protection, so that subsequent attempts to access it are passed to the authorisation server for federated authorisation on behalf of the pension owner (the 'user' in user managed access), performed centrally at the consent and authorisation service.

Once a data provider has registered a PeI, the provider can also (for whatever reason it decides – see **8** below, or because the owner instructs it to do so) modify the details of the registration, or delete the registration. The provider cooperates with the consent and authorisation service to maintain the PAT for the user. Usually the PAT will be reissued periodically during user interactions with the consent and authorisation service.

The architecture uses an UMA protocol to achieve registration, UMA2 federated authorisation specifically profiled for the pensions dashboards ecosystem. Details of the UMA design and the pensions dashboards ecosystem UMA profile are given in associated documents.

---

[27] For data providers that are also dashboard providers, their dashboard does have the choice of using the identity service as its mechanism of authentication, however, from this architecture's point of view, these are entirely separate functions.

Over a protracted period of time of no use by the pension owner, for example 18 months, the consent to register and manage delegations will expire. It can be renewed at the consent and authorisation service (as discussed in **2.5.3**).

The interaction to refresh consent may be initiated by a failed attempt to access pension details, when the data provider will be unable to start the authorisation process, so will return to the dashboard, which must refresh consents before retrying. Such refreshed consents will also result in the UMA authorisation server reissuing the UMA PAT to the pension provider's resource server (RS).

6.4 Authorisation protocol

The authorisation process is dependent on the use of the User managed access version 2 (UMA2). Details of UMA are available both online and in the 'Why UMA' summary in this document (section 6.5).

When a dashboard, and its associated user, attempts to access a pension identifier (PeI) the pension provider must check that the access is authorised. It does this by cooperating with the consent and authorisation service and the dashboard, implementing an UMA protocol, (UMA2 grant for OAuth2) specifically profiled for the pensions dashboards ecosystem.

The UMA protocol requires a valid access token, which must align with the access request to each PeI. The provider's UMA resource server (RS) will ensure the access token is valid by introspecting the token at the consent and authorisation service (within an authorisation server). If the token is valid and matches the PeI, the data provider can serve the pension details to the dashboard.

If the token is missing or invalid, the data provider seeks an UMA permission token (PMT) from the authorisation server and returns this to the dashboard.

The dashboard will seek authorisation from the authorisation server, quoting the permission token. It will be instructed according to the UMA profile, and taken through an authorisation 'dance', depending on the various states of the user and the transaction. This will include pensions dashboards ecosystem specific tokens: requesting party token (RQP), which the dashboard issues associating its user with a dashboard instance, and a persisted claims token (PCT - an instance of the UMA persisted claims token) by which the authorisation server associates the requesting party token with the strongly assured identity of that same user from the identity service. In following the protocol, redirecting the user if necessary, and in accord with the pension owner's policy at the authorisation server, it may issue the required access token (RPT).

6.5 Why UMA?

The User managed access 2 protocol is the open standard, specifically for use cases that require federated authorisation, delegation to third parties and less trusted requesting clients, and user managed centralised fine-grained authorisation control over distributed resource endpoints. UMA standardises authorisation failure behaviour, which is handled by the UMA authorisation server, irrespective of the client software; UMA supports pushed claims (from the client) and client redirection on failure, so the user experience can be harmonised across diverse endpoints and state reused to reduce user friction.

Briefly the benefits of UMA are:

Federated authorisation (the consent and authorisation service)

- support for data providers that do not have online accounts for customers
- support for data providers that do not have existing API authorisation capability
- users can manage their policy centrally, independently of any specific pension provider or pensions dashboard operator, including revocation of consents (a Data Protection Act 2018 and pensions dashboard requirement)
- the federated authorisation service can control the ecosystem-wide common authorisation policy, including serving the role of ecosystem-wide identity trust anchor
- as the authorisation service protects pensions independently of dashboards, users can move between pensions dashboard operators easily, without requiring a new find operation, use multiple dashboards contemporaneously, and data providers will have reduced load on their find services
- federated authorisation enables the separation of find from subsequent view operations, segregating the pension details data flow from the find flow and avoiding intermediary components handling pension details
- when other Open Finance initiatives have developed further, the architecture is open so that users may select their own (UMA) authorisation server and still use the same pension provider architecture

Custom policy and delegation

- users can establish policy enabling controlled access by other parties (delegates – financial advisers etc)
- UMA is the only open standard that expressly gives authorisation to a user – a person, rather than a software element, including the case where the user of a client is *also* the owner of the data resource. Moreover, it is the only open standard that supports delegation to a user (eg a financial adviser) other than the pension owner. This is required by pensions dashboard policy
- users can control access to specific pensions (not bulk access) and can have a separate policy for each pension, for each delegate (whether financial adviser, guidance body officer, user's own different dashboards)
- users can manage their consent policy asynchronously from actual access attempts.
  So:
  - a user can set their policy for different dashboards, which they use with a particular dashboard provider-specific login
  - they can also set policy for individuals/organisations, which can access her pensions when they are not present
- dashboard operators can benefit from such user (as pension owner) to user persona (user as dashboard login/session) delegation, which enables dashboards to be delivered in existing portal/legacy/commercial off the shelf software environments, as well as fintech products with their own authentication regimes, including the special case of a dashboard with no local user account
- policy can be fine grained and separate for each pension. Different data elements could be controlled for different requesting dashboards/individuals. The types of access could be extended within the protocol in the future eg financial adviser-specific detailed data items

7. Governance register

This section outlines the functions of the governance register from an IT architectural viewpoint. (It does not include the business process, organisational and human elements.)

7.1 Overview of governance register

Diagram 12, below, shows information flows between major parties, including (at the bottom of the diagram) information flows that support decoupled processes with bodies outside the ecosystem.

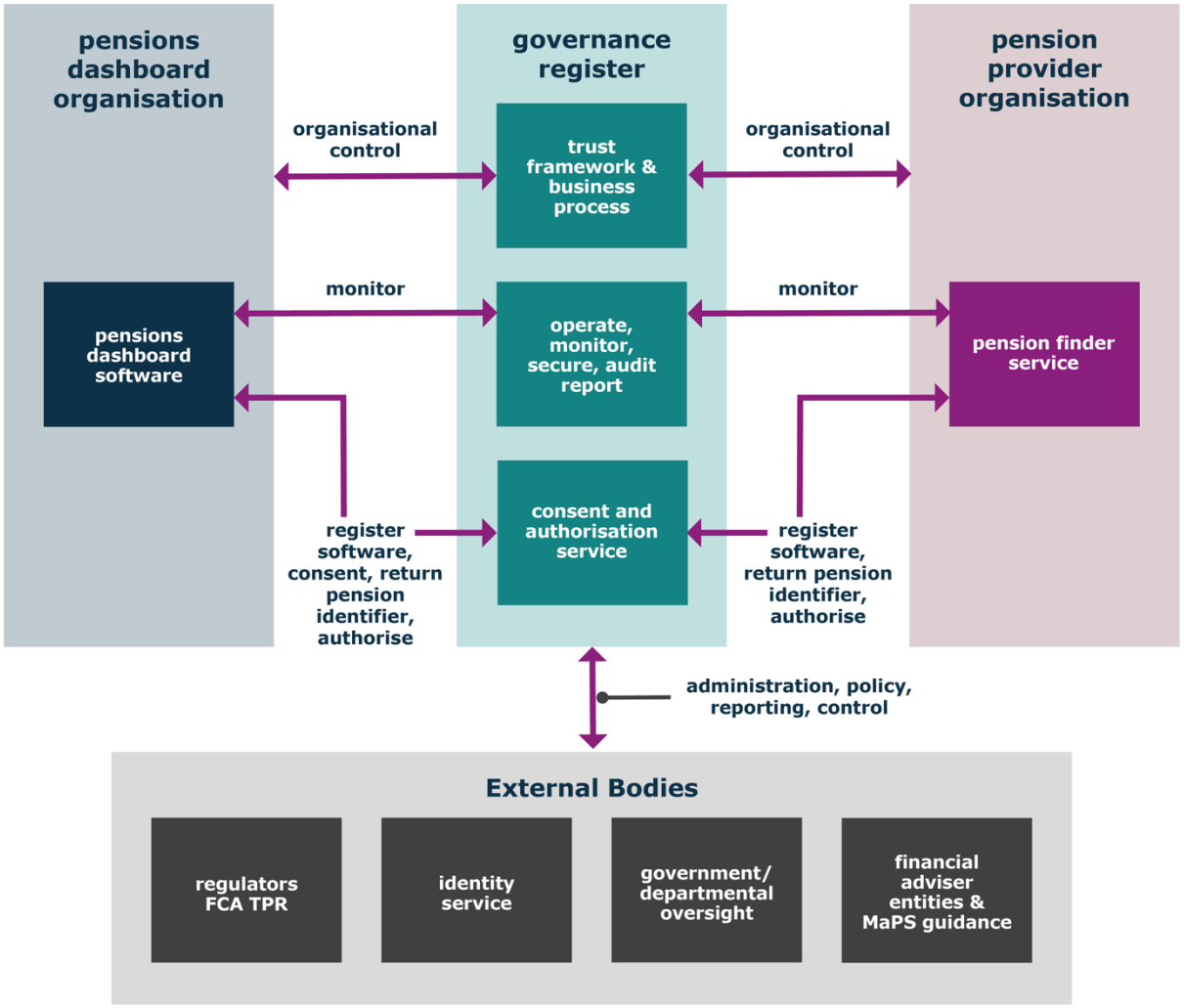

Diagram 12: information flows between pensions dashboards ecosystem and external parties

The relationships with dashboards and data providers are:

- organisational participation in the ecosystem (static trust relationships between participants)
- monitoring feeds with the governance register functions for operational and security management, and audit
- dynamic properties of the service in operation: software component registration, user interactions with consent and authorisation interfaces, and registration of and authorisation of access to PeIs

The external bodies require reporting from and feedback, to enable modification of the governance register functions.

Dashboard and provider monitoring feeds will include information which supports:

- end to end traceability of transactions (note this does not require monitoring of the identity performing the transaction, simply that the transaction is traceable to *local* audit records at each component)
- volumetric reporting by ecosystem participant
- account turnover (at dashboards)
- success rates for find by endpoint and by registered legal entity
- cyber and security monitoring of components and patterns of use

7.2 Functions of the governance register

Diagram 13, below, represents the key components within the governance register.

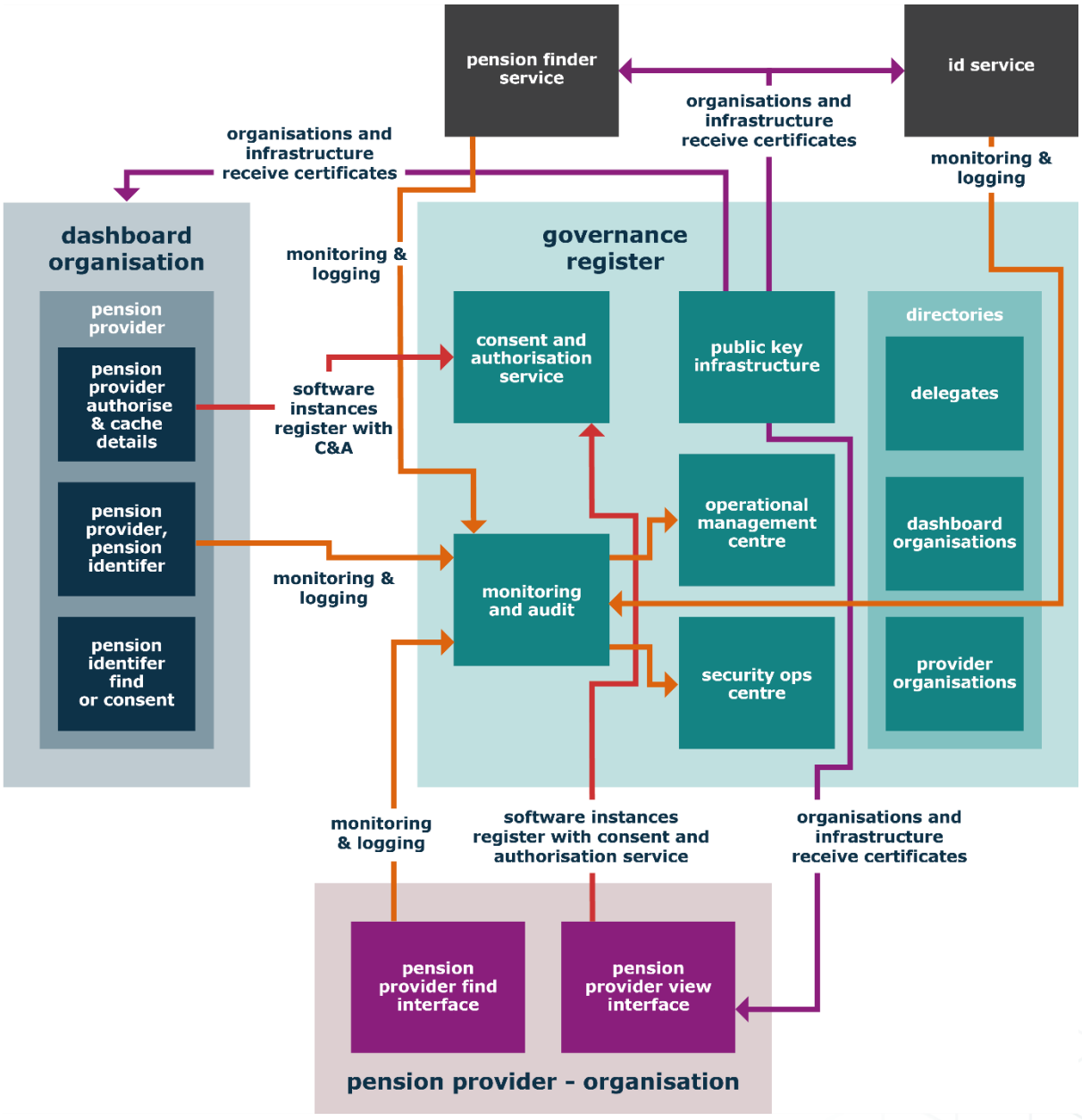

Diagram 13: key components within the governance register

Organisational membership of the ecosystem is controlled by entries in a register of organisations.

Organisational participation is managed using certificates issued by a private, public key infrastructure (PKI).

Software components deployed in organisations will register (statically or dynamically) with the consent and authorisation service (and will need appropriate key material to do so).

All software components will comply with monitoring and logging requirements to the governance register service. Such feeds support an ecosystem wide operational management and security operations centre.

Delegates (financial advisers, financial adviser organisations, and MaPS guidance officers) will be registered in a directory, so that the consent and authorisation service can support the pension owner's consent manager and can check the delegate's professional status during authorisation decisions. (It is possible that the directory for financial advisers may be a logical directory, handing off to FCA managed IT assets.)

7.3 Governance register relationships with providers and dashboards

The governance register functions enable:

- trustworthy participation and connections
    - the governance register will impose requirements on all organisations to prove their identity and regulatory purpose before being entered in a register
    - the governance register will provide a public key infrastructure (PKI) to enable organisations to participate in the ecosystem
    - the governance register will impose requirements on all participants to register their software entities
    - thus, it is not necessary for a data provider to manage its own list of permitted connecting entities; it is sufficient to know that the dashboard calling the provider's view interface has a certificate[28]
    - moreover, the provider is assured that the dashboard's call to a specific pension identifier for the current dashboard user is authorised[29] because it carries a token, which the consent and authorisation service has provided for that narrow, specific purpose. In addition, the channel from dashboard to pension provider view interface will be encrypted

[28] The governance register issues certificates to participants. This is static trust relationship.

[29] The consent and authorisation service issues a token to each separate authorised view call to each PEI. This is dynamic trust. Each separate call is proved according to the pension owner's then current policy at the consent and authorisation service. The data provider does not need to blindly trust dashboard software (static trust), rather it trusts the consent and authorisation service to authorise each call (dynamic trust).

  - calls to a provider's find interface can only be made from the pension finder service. Both provider and pension finder service will have certificates[30]. The pension finder service should be able to buffer traffic, to smooth peaks in the find load, based on the absence of ACKs, or digital acknowledgements, from a pension provider find interface. Similarly, the pension provider find interface should be able to manage flow with traffic throttling
  - similarly, dashboards will, having (dynamically if necessary) registered their software instances, interact with the content and authorisation service to obtain authorisation tokens, to provide secure access to data providers
  - dashboards will be able to receive PeIs for a specific user by cooperating with the content and authorisation service, at which the user's PeIs are registered

- monitoring of authorisation requests: the consent and authorisation service will 'see' calls to each provider's view interface, since the provider uses the consent and authorisation service for authorisation, it is involved in issuing and validating tokens
- monitoring of traffic in general: data providers interfaces will monitor (ie log relevant information) to the governance register's technical monitoring function. The latter will also receive monitoring from the consent and authorisation service and from dashboards and thus implement ecosystem monitoring, to support operational requirements, security monitoring and response, participant compliance and reporting
- traffic limiting: providers will provide denial of service protections for their estate. As ALL legitimate traffic will be identified by appropriate certificates, illegitimate traffic from outside the ecosystem can simply be dropped. The pension finder service should be able to buffer traffic to smooth peaks in the find load. As above, the central ecosystem services will provide appropriate monitoring of legitimate participants

[30] Thus, a possible lower level design for the pension provider find interface with the pension finder service and with the governance register is to use simple mutual transport layer security (TLS) based on the ecosystem PKI certificates.

8. Pension identifiers (PeIs)

A pension identifier (PeI) is the term used to cover all separately identifiable pensions, in which some individual(s) may have an interest. It is the identifier of a pension, not in itself a statement of ownership. It is entirely possible for a data provider to issue PeIs for each of its separate pension investments, without any reference to the owner of the assets.

The result of a user's pension find request is a list of pension identifiers, which the pension provider(s) have determined belong to the same person. Each separate pension is given a PeI in the pensions dashboards ecosystem. A PeI is a reference to a specific pension; it does *not* identify the owner; it is *independent* of any dashboard; it can be reused by the pension owner *across* dashboards; it can be used by *delegates* and the pension owner to request access to pension details.

PeIs have these properties:

- PeIs are unique, high-entropy, opaque, dereferenceable identifiers of specific pensions
    - unique – every separate pension identifier is given a never reused identifier, which is specific to that pension irrespective of which pension provider is managing it (and irrespective of the owner of that pension)
    - high entropy – there are vastly more pension identifiers possible than will ever be used (so it is very unlikely a random identifier will be a valid identifier)
    - opaque – contain no personal information (although they will contain information related to the pension provider/ISP which manages the asset)
    - dereferenceable – is capable of being *resolved to a URL,* which can serve the pension details associated with the PeI
- a PeI will be capable of data entry by a person using a standard keyboard (ie the character set will be selected in part for ease of reading and use)
- a PeI will have an associated (but separate) human readable description, which may include other identifiers of meaning to the pension provider, which will usually only be present in unprotected form at the consent and authorisation service, pensions dashboard or pension provider. (But it will be part of data exported from a pensions dashboard if the user requests it.)
- PeIs are *not* secret – the intention is that they have no intrinsic value, possession of a PeI does *not* enable access to the associated details, nor to the pension provider, which may manage the pension, nor to any associated personal or financial information. PeIs are thus safe to email or keep in unsecured locations

PeI's will be a new concept for the pensions industry and therefore the requirements for data providers will need to articulate the structure and proposal, so expectations can be met:

- a standard form of *associated* human readable description (so that the consent and authorisation service and pensions dashboards can show standard information)
- how dereferencing happens (ie which pension provider serves the request for value), for example - dashboards have configuration tables of PeI prefixes
- how dereferencing changes if the pension is moved

- how dereferencing changes if the pension is consolidated with others

A *simple example* of a PeI is the URL, pei:aviva00001-WS23JQ48KH789KS349, which might resolve to the view URL,[31] www.dashboard.aviva.com/pei/WS23JQ48KH789KS349, in which the asset number component WS...349 is unique, high-entropy and opaque. In the event that this asset is transferred to aegon, the above URL might *automatically resolve* to, pei:aegon00007-WS23JQ48KH789KS349[32], which a dashboard would substitute for the old PeI and retry. Note that this solution for transfers would require aviva to keep a record of its transferred pension identifiers and the target company.

The *general issue* here is that the industry needs to standardise on meaning and business use cases, which PeIs support. Dashboards resolve prefixes into standard domains, and all PeIs directly resolve to literal URLs, at which dashboards can retrieve pension details (subject to the authorisation protocol).

Although PeIs could enable the user to have stable references irrespective of industry mergers, or transfers, or consolidation events, these will have to be managed during the business processes. Notably the origin PeI (eg pei:aviva... above) will have to have its internal record updated to return the resulting PeI (eg pei:aegon... above).

Note that the architecture described in this document works, even if the industry does not arrange for such a unified scheme. Even if each provider is a silo and the PeI is deleted when the pension moves (or other event), the architecture delivers the intent. The downside, for both the user and for the industry, is that the find operation will have to be re-run to re-find the moved pension. This is an avoidable overhead for user and for the data providers.

---

[31] The URL www.dashboard.aviva.com/pei/WS23JQ48KH789KS349 is what the dashboard uses (ie performs https GET) to obtain the pension details. The access token is provided in the headers and is checked according to the UMA grant protocols.

[32] That is the above GET call to www.dashboard.aviva.com/pei/WS23JQ48KH789KS349 returns a response of 'pei:aegon00007-WS23JQ48KH789KS349' rather than the pension details. The new PEI prefix is resolved to aegon's domain and a GET is performed on that domain. (Note also that the asset number component WS... is the same – it is a fixed property of a pension.)