



© Crown copyright 2024

This publication is licensed under the terms of the Open Government Licence v3.0 except where otherwise stated. To view this licence, visit [nationalarchives.gov.uk/doc/open-government-licence/version/3](https://nationalarchives.gov.uk/doc/open-government-licence/version/3).

Where we have identified any third party copyright information you will need to obtain permission from the copyright holders concerned.

Any enquiries regarding this publication should be sent to us at: MoJ Data Access Team Mailbox [data.access@justice.gov.uk](mailto:data.access@justice.gov.uk)



HM Courts &  
Tribunals Service

Publishing & Information Service (P&I)

Low Level Design

Version: 1.0

## Table of Contents

<b>1</b>	<b>DOCUMENT CONTROL.....</b>	<b>3</b>
<b>1.1</b>	<b>Document History.....</b>	<b>3</b>
<b>1.2</b>	<b>Document Authors .....</b>	<b>3</b>
<b>1.3</b>	<b>Document Assurance .....</b>	<b>3</b>
<b>1.4</b>	<b>Document References.....</b>	<b>4</b>
<b>2</b>	<b>INTRODUCTION.....</b>	<b>5</b>
<b>2.1</b>	<b>Scope.....</b>	<b>5</b>
<b>2.2</b>	<b>Intended Audience .....</b>	<b>5</b>
<b>2.3</b>	<b>Definitions and Terminology.....</b>	<b>6</b>
<b>3</b>	<b>TARGET DESIGN .....</b>	<b>8</b>
<b>3.1</b>	<b>API Specification .....</b>	<b>8</b>
<b>3.2</b>	<b>Consumers and Connectivity.....</b>	<b>8</b>
<b>3.3</b>	<b>User Authentication and Verification with IDAMs .....</b>	<b>18</b>
<b>3.4</b>	<b>Data Storage.....</b>	<b>33</b>
<b>3.5</b>	<b>Reference Data .....</b>	<b>35</b>
<b>3.6</b>	<b>Network .....</b>	<b>39</b>

## 1 Document Control

This section provides a history of how the document has been changed and how the changes have been governed.

### 1.1 Document History

The following provides a log of changes to this document:

Version	Date	Author	Notes
0.1	01/December/21		Initial Draft
0.2	07/February/22		
1.0	14/February/22		Document published

### 1.2 Document Authors

Role	Name	Responsibility
Author		P&I Technical Lead

### 1.3 Document Assurance

This document has been reviewed and approved by the following, the list has been grouped into different sections to aid readability as this document will be required at SDS PDG and possibly TDA.

Role	Description
Information	Shared for informational purposes including use as reference document for their domain's documentation and processes
Reviewer	Actively reviews the content to ensure alignment to guidelines and policies and technical accuracy, will contribute as required
Approver	Provides assurance that their domain is correct in context of the LLD

#### 1.3.1 Publishing & Information Service (P&I Service) Project

Version	Date	Role	Reviewer/Approver	Responsibility
1.0	DD/MM/YY	Approver		Future Hearings Solutions Architect
1.0	DD/MM/YY	Reviewer		P&I Delivery Manager
1.0	DD/MM/YY	Reviewer		P&I Technical Lead
1.0	DD/MM/YY	Reviewer		P&I Business Analyst
1.0	DD/MM/YY	Reviewer		P&I Business Analyst
1.0	DD/MM/YY	Reviewer		P&I DTS Technical Lead
1.0	DD/MM/YY	Information		Security Architect
1.0	DD/MM/YY	Information		HMI Delivery Manager
1.0	DD/MM/YY	Information		Crime Solutions Architect

## 1.4 Document References

The following documents should be read in conjunction with this document:

Reference	Document	Version	Author	Source
<b>DOCREF.01</b>	Target Operating Model	2.0		Confluence
<b>DOCREF.02</b>	DACS Technical Guidance Library – Item 1			Confluence
<b>DOCREF.03</b>	Host Security Pattern	NA		Confluence
<b>DOCREF.04</b>	Availability Standards	N/A	Microsoft	<a href="#">Azure</a>
<b>DOCREF.05</b>	DR Standards	N/A	Microsoft	<a href="#">Azure</a>
<b>DOCREF.06</b>	Logging and Monitoring Policy	N/A	PlatOps	Confluence
<b>DOCREF.07</b>	Azure Database Storage	N/A	Microsoft	<a href="#">Azure</a>
<b>DOCREF.08</b>	Azure Job Scheduler	N/A	Microsoft	<a href="#">Azure</a>
<b>DOCREF.09</b>	IDAM Operations Manual	3.0	IDAM	Confluence
<b>DOCREF.10</b>	NCSC Cloud Security Principles	N/A	NCSC	<a href="#">Gov.UK</a>
<b>DOCREF.11</b>	DTS DTU (BIAS) HLD	0.11	BIAS	Teams
<b>DOCREF.12</b>	File Based Patterns	1.0	PDG	Teams
<b>DOCREF.13</b>	Cloud Infrastructure	N/A	DACS	Confluence
<b>DOCREF.14</b>	MyHMCTS Register Your Organisation	March 2020	MyHMCTS	Confluence
<b>DOCREF.15</b>	P&I Service Application Component Reuse	1.0	Project	Teams
<b>DOCREF.16</b>	HMCTS Media Guidance	March 2020	HMCTS	<a href="#">Gov.UK</a>
<b>DOCREF.17</b>	SDS PDG P&I Hub HLD	1.2	Project	Teams
<b>DOCREF.18</b>	API Schema Specification	0.16	Project	Teams

## 2 Introduction

The Business Vision of the P&I Service is to support the delivery of HMCTS's commitment to open justice and to modernise and improve public access to information provided by HMCTS by publishing or displaying court/tribunal information (such as court and tribunal lists), according to the relevant policy requirements and business rules.

The service will provide a publishing platform which will enable the sharing or display of information provided by HMCTS and allow for updates of this information, as and when appropriate.

The P&I Service will simplify and streamline elements of the work currently required to publish lists, outcomes, judgments and enable information to be displayed via relevant presentation hardware, consistent with jurisdictional procedures and business rules.

This will improve HMCTS' commitment to open justice and enable the provision of transparent and consistent court and tribunal information across all jurisdictions.

When considering the publication of information there are a number of issues that need addressing, which may be summarised below:

1. Differing Formats
2. Differing routes for updates
3. Manual processes and interventions
4. Multiple places for updates
5. No single source or centralisation for publications online, in print or via communication channels

The Publication and Information (P&I) Service will ease the collection, representation and routing/distribution of updates to the channels ensuring that the most recent data is available and is both presented and formatted consistently.

### 2.1 Scope

This is the Low Level Design for (P&I) Service and spans all the design domains (business, data, application, and technology) and examines some of the relevant states of the design.

### 2.2 Intended Audience

The document is intended for technical people looking to understand the purpose and operation of the Publishing & Information Service (P&I Service).

## 2.3 Definitions and Terminology

The following terms and abbreviations are used within this document.

Term	Meaning
<b>API</b>	An application programming interface is a computing interface which defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow.
<b>APIM</b>	API Management
<b>DTS</b>	HMCTS Digital Technology Services
<b>DTU</b>	The DTS Data Transfer Utility
<b>IDAM</b>	Identity and Access Management
<b>Judgment</b>	A judgment is a decision of a court regarding the rights and liabilities of parties in a legal action or proceeding. Judgments also generally provide the court's explanation of why it has chosen to make a particular decision or court order. is a writeup of how the judge has reached their decision.
<b>List</b>	A list is defined as a set of ordered information that contains hearing details that can be published to different user groups based on permissions and information provided on the list templates. An example of a list would be a "public List" this would be published displayed by the court and highlight the hearings on the given parameters
<b>Resource</b>	Unique resource URI for the RESTful api endpoint
<b>Notification (If required)</b>	Would allow verified users to receive a notification (likely email) to inform them that a list they are interested in has been published and they can log in to view that publication.
<b>Outcome</b>	An Outcome is the conclusion or result of a hearing (not necessarily the final outcome 'judgement for a case' as there can be many hearings for a case to conclude to a verdict).
<b>RESTful</b>	Representational state transfer - REST has been employed throughout the software industry and is a widely accepted set of guidelines for creating stateless, reliable web services.
<b>S&amp;L</b>	Scheduling and Listing
<b>SDP</b>	Strategic Data Platform
<b>Subscription</b>	P&I subscription service will allow verified users to subscribe to lists of interest through (e.g. media who will use this to report on appropriate cases). Lists are attached to an email, which is automatically sent out when a list is available for publication.
<b>JSON</b>	JavaScript Object Notation
<b>META</b>	Set of data describes and gives the information about the data

<b>Verified User</b>	Those users having been either authenticated by an IDAM OR subscribers having had their e-mail addresses verified by Court Staff
<b>Unverified User</b>	Users that access the P&I service via the UI without having been authenticated
<b>Method</b>	The proposed function call type for the RESTful API call
<b>URL</b>	Uniform Resource Locator
<b>URI</b>	Uniform Resource Identifier
<b>AAD</b>	Azure Active Directory
<b>B2C</b>	Business to Consumer (Azure B2C)
<b>HMI</b>	Hearing Management Interface
<b>CFT</b>	Courts & Family Tribunals
<b>REST API Logic</b>	Market consumed references from REST api design principles
<b>Publication lists</b>	Crime mags cft et, sjp cft data lists,
<b>LCSU</b>	Screen references for data and updates to the list of events to be updated
<b>OPENAPI – specification</b>	<a href="https://swagger.io/specification/">https://swagger.io/specification/</a>
<b>HMCTS API Reference/RESTful API</b>	<a href="https://hmcts.github.io/restful-api-standards/#http-requests">https://hmcts.github.io/restful-api-standards/#http-requests</a>



### 3 Target Design

This section describes the Target Architecture needed to meet the business outcomes, goals and objectives listed in **Error! Reference source not found.** and **Error! Reference source not found.**

#### 3.1 API Specification

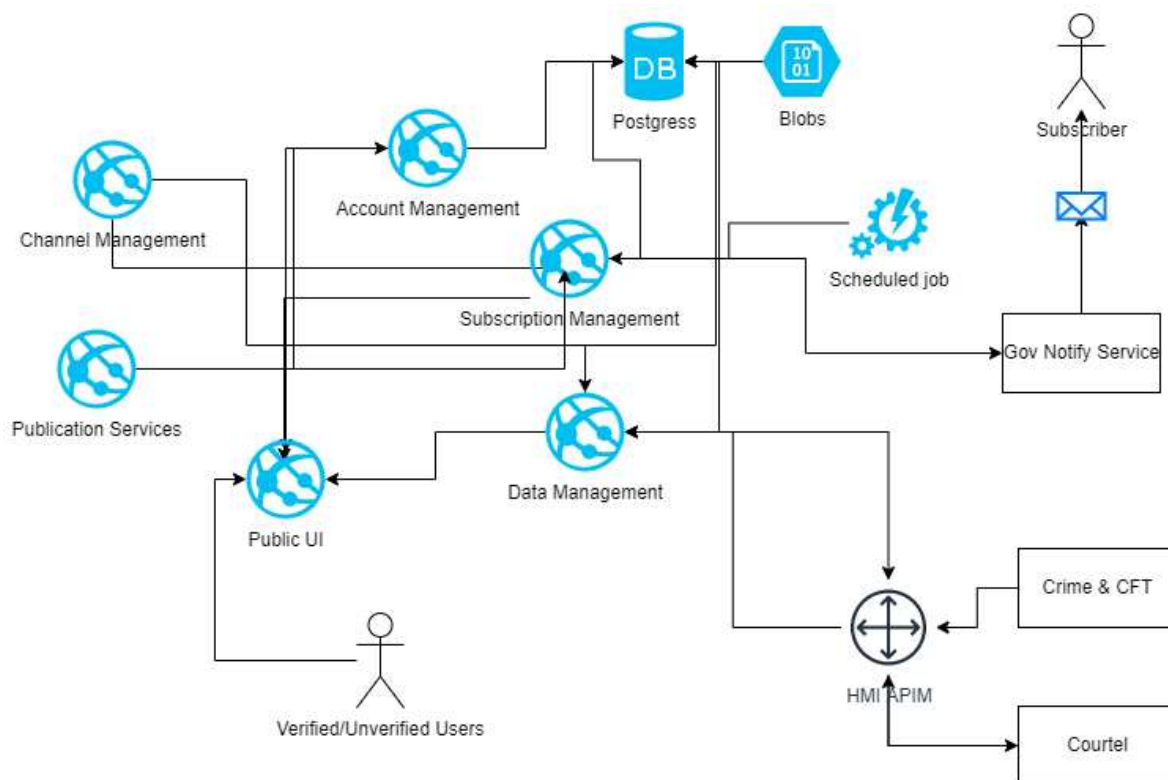
This section is detailed in a separate document:

[API Schema Specification 0.16](#)

#### 3.2 Consumers and Connectivity

There are only two ways to consume information into the P&I Service, one is via **Public UIs** available to the public, the other is via **API** (for further details please see section 3.1)

As a representation of the consumers and connectivity below there is a diagram representing the services involved to consume information and also to produce and feed P&I Service with data.



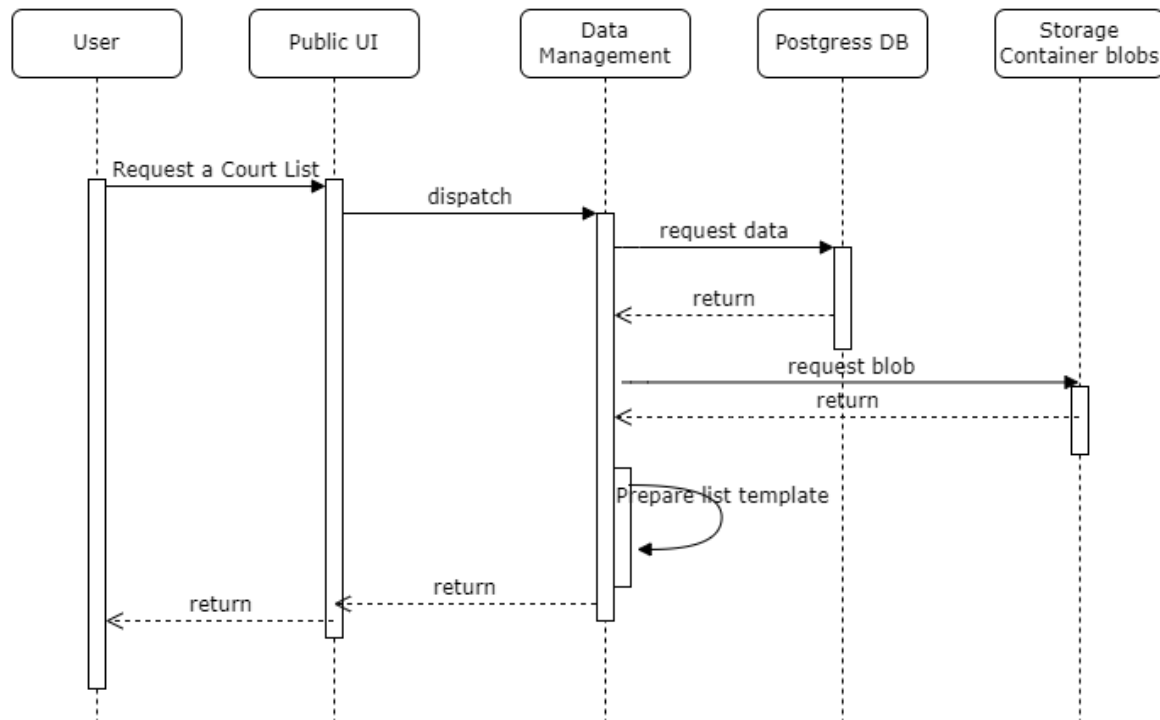
Information consumers:

- Verified and Unverified Users on the **Public UI**.
- Subscribers on the **Gov Notify Service**
- External parties via **API: Courtel**

### 3.2.1 Public UI – Verified and Unverified Users

P&I Service users will have access to a set of screens accessing a public domain via Portal UI and Live Case Updates

Below there is a user flow representation of the flow to have data available in Public UIs for Verified and Unverified users.



#### 3.2.1.1 P&I Portal

A **user interface** that will be made available through web/mobile which would be typically accessed via a web redirect.

Below are example screens representing some of the UI views in P&I Service.

[Home](#)[Sign in](#)[< Back](#)

## What do you want to do?

**Find a court or tribunal**

View time and type of hearings and more

**Find a Single Justice Procedure case**

TV licensing, minor traffic offences such as speeding and more

[Continue](#)[Help](#) [Privacy policy](#) [Cymraeg](#) [Cookie policy](#) [Accessibility statement](#) [Contact](#)

[< Back](#)

Filter

Selected filter

[Clear filters](#)

Apply filters

Jurisdiction

☐ Tribunal

☐ County Court

☐ Crown Court

☐ Magistrates' Court

Region

☐ Scotland

☐ Wales

☐ South West

## Find a court or tribunal

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

National lists

[Single Justice Procedure cases](#)

Court or tribunal

A

[Aberdeen Tribunal Hearing Centre](#)

[Aberystwyth Justice Centre](#)

[Aldershot Justice Centre](#)

[Amersham Law Courts](#)

[Ashford Tribunal Hearing Centre](#)

[Ayr Social Security and Child Support Tribunal](#)

B

[Barkingside Magistrates' Court](#)

[Barnet Civil and Family Courts Centre](#)

Low Level Design

Page 11 of 49

v1.0

### 3.2.1.2 Live Case Updates

As part of the same UI P&I has another way to display services for courts consumers requiring another set screens as a UI (**In courts Screen**). Crown Courts will be able to display information on screens regarding the court, live hearing updates, rooms information, etc.

The intention with this URL is just merely to consume information to be displayed on screen and there will not be any other interaction with the P&I service.

There is no authentication required for the courts and tribunals within P&I Service to access

#### How To:

Each court will be given a URL using a base64 encoded with hidden parameters.

The LCSU and the consumption of the data will be static. The screens will refresh periodically to reflect any possible updates on their screens. The time which the screen will refreshed will be configurable.

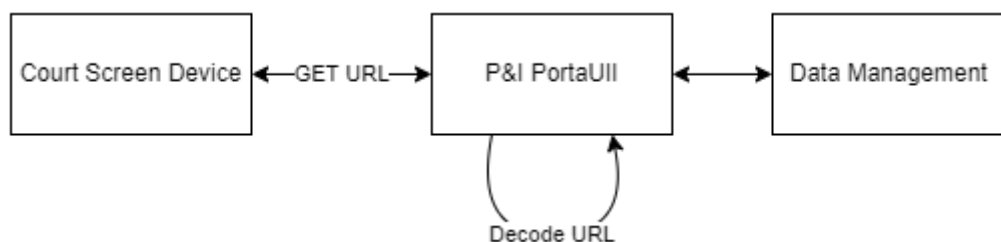
#### Example:

Court Screen Device makes a GET request to the URL provided

<https://.../44aW4tY291cnQvbGl2ZS1jYXNlXN0YXR1cz9jb3VydlkPTQ0>

#### P&I decodes base64 URL and serves:

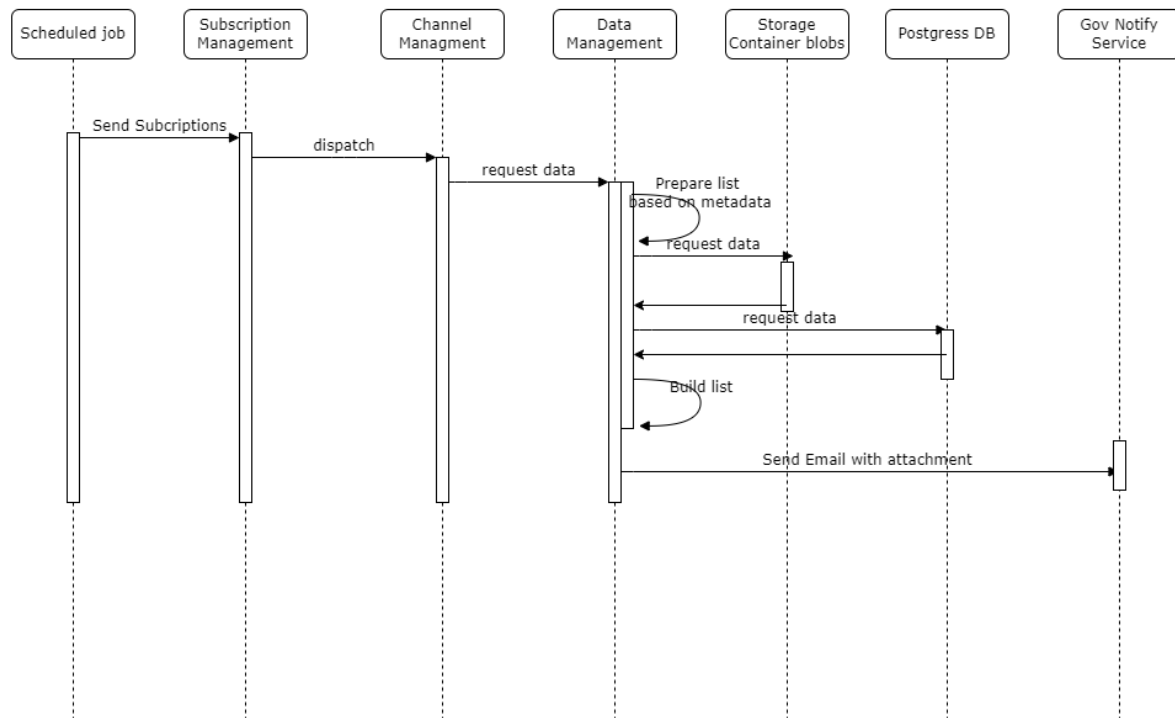
<https://.../in-court/live-case-status?courtId=44>



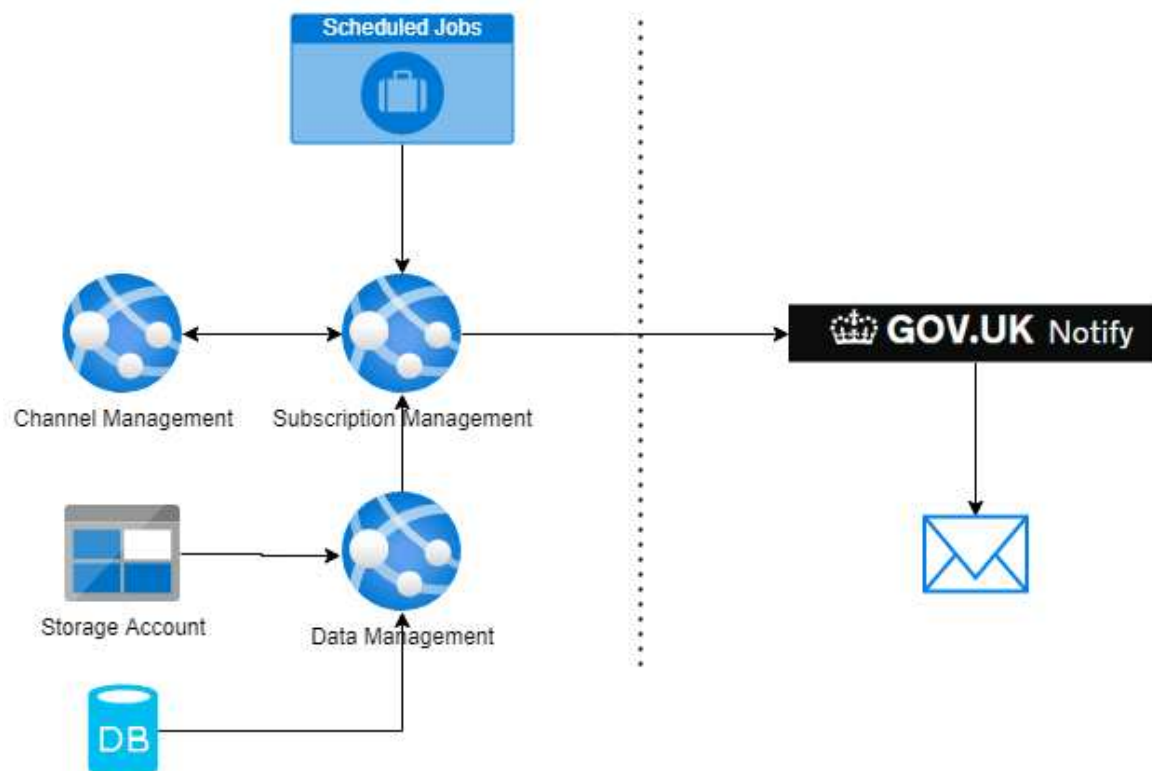
### 3.2.2 Subscribers - Gov Notify Service

P&I subscription service will allow verified users to subscribe to lists of interest through (e.g. media who will use this to report on appropriate cases). Lists are attached to an email, which is automatically sent out when a list is available for publication.

Below is a flow with the sequence of events that will run on a daily schedule using a job scheduler.



As per the component diagram the below, resources are involved in delivering lists to subscribers via Gov Notify Service.



### 3.2.3 Courtel – External consumers

Courtel is an external party having available two ways to consume publications. One is via the API Schema (see section 3.1), another is a subscriber set up as describe in the section for Subscribers above.

At this moment is uncertain the integration criteria to setup an external partner like Courtel as the conversation are still in progress.

### 3.2.4 Information producers

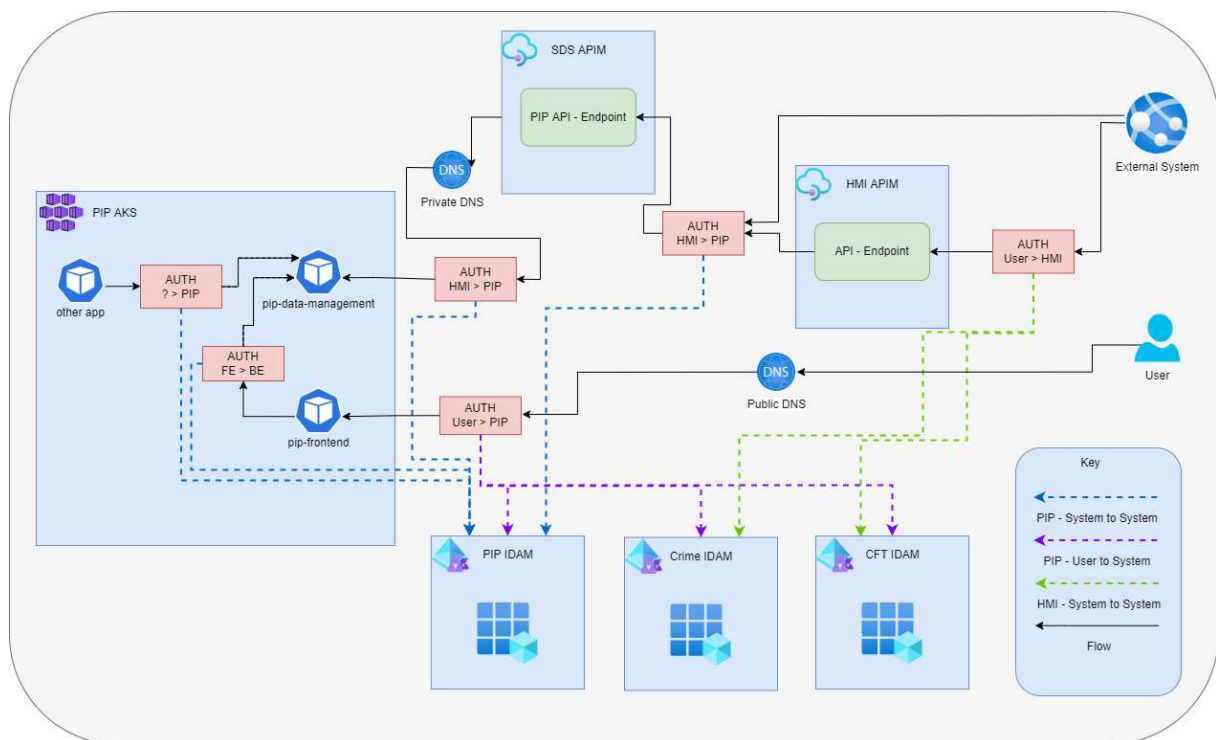
As described in section 3.1, the P&I Service provides API endpoints to allow external data providers like Common Platform and List Assist to ingest with publications to the hub.

However, an alternative solution is required to allow users without API connectivity the ability to send files to P&I. This file uploader tool available in the **Public UI** works as a temporary business feature replacement till their data ingestion capabilities are sorted from their data providers.

### 3.2.4.1 External Parties – Upload Via HMI

For external HMCTS parties (Common Platform and List Assist), the upload of files will be done via HMI through an API

The diagram below shows the interactions between the User, other external systems, HMI and P&I.



The file is sent via an API endpoint (POST -> /publication), which will then be validated against the schema and persisted in P&I.


See section 3.1 for details on the schema and further details on how authentication will occur for users.

For HMI to authenticate with P&I, an application will be registered in Azure, and a Client Secret + ID provided to HMI. We will then validate the token that HMI sends to us using Azure.



### 3.2.4.2 Upload for CTSC - Courts Tribunals Service Centre Staff

Admin roles in the P&I Service can ingest data via P&I Service UI. The manual process of uploading a file via UI portal (screenshots below) would mimic the process of the API ingestion. Therefore, data will be published and accessible.

 **GOV.UK**

Court and tribunal hearings

## Manual upload

Manually upload a JSON, CSV, PDF, Word, HTM or HTML file, max size 2MB

No file chosen

Court name

Bolton Combined Court

List type

SJP Public List


Hearing dates

For example, 16 01 2022 to 20 01 2022

Day	Month	Year		Day	Month	Year
<input type="text"/>	<input type="text"/>	<input type="text"/>	to	<input type="text"/>	<input type="text"/>	<input type="text"/>

Available to

Public

 **GOV.UK**

Court and tribunal hearings


[Home](#) [Subscriptions](#) [Sign out](#)

[< Back](#)

Check your answers

File	FurtherAI_NeuroscienceInfluencedA I.pdf	<a href="#">Change</a>
Court name	Aberdeen Tribunal Hearing Centre	<a href="#">Change</a>
Document type	LIST	<a href="#">Change</a>
List type	SJP_PUBLIC_LIST	<a href="#">Change</a>
Hearing dates	3 Feb 2022 to 3 Feb 2022	<a href="#">Change</a>
Available to	PUBLIC	<a href="#">Change</a>
Language	ENGLISH	<a href="#">Change</a>
Display file dates	3 Feb 2022 to 3 Feb 2022	<a href="#">Change</a>

Confirm

 **GOV.UK**

Court and tribunal hearings

[Home](#) [Subscriptions](#) [Sign out](#)

# Success

Your file has been uploaded

What happens next

[Upload another file](#)  
[Remove file](#)  
[Home](#)

### 3.3 User Authentication and Verification with IDAMs

In this section the following IDAM integrations, available in P&I, will be outlined:

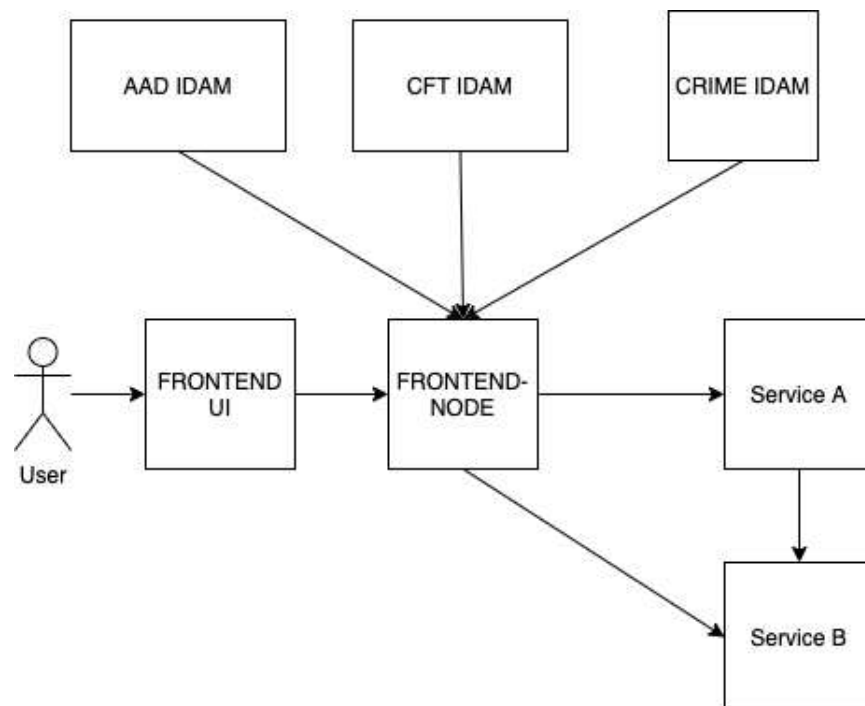
- P&I Service AAD IDAM (built within P&I Service)
- External IDAM integration for CFT
- External IDAM integration for Crime

Users authenticated against any of the three different IDAMs will be considered **Verified** users.

Public users, however, will be considered **Unverified** users.

#### 3.3.1 P&I Service AAD IDAM

There are several service-to-service level interactions, most notably from the node backend to the java backend services, however also interaction between the java services themselves.



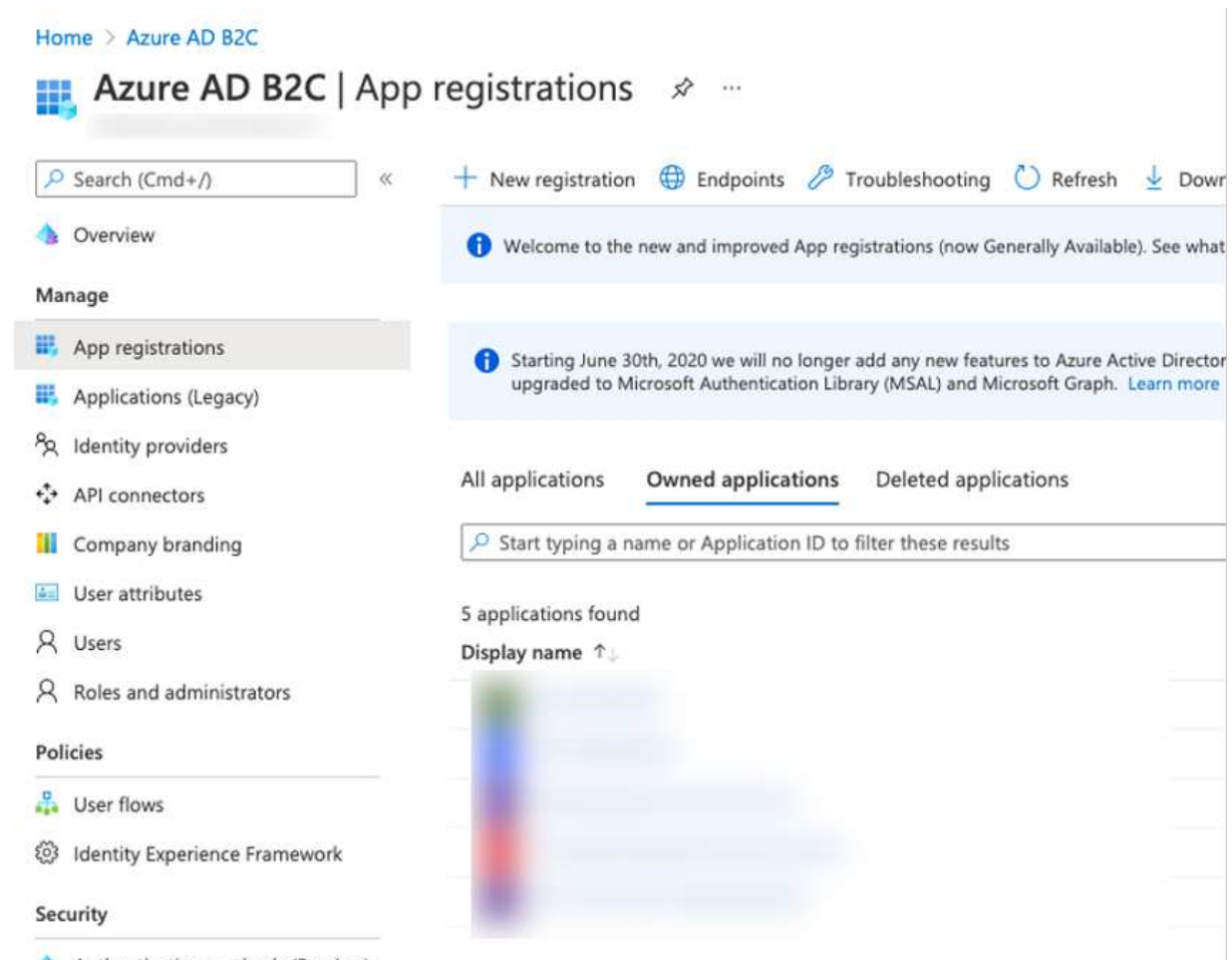
(Service A / Service B represent any service)

##### 3.3.1.1 Authentication flow

Authentication between service to service will use Azure Active Directory client credentials flow, detailed here

<https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-client-creds-grant-flow>

To enable this, it is required to create Applications with Azure Active Directory for each of our services.



### 3.3.1.2 Passing user details

On-behalf OAUTH flows (where the service acts on behalf of the user) was considered as a way of passing details of the source user between systems.

If services require details of the user that originally made the request (e.g. FRONTEND-Node to Data management), then we can pass the detail that it needs as part of the request to the backend (e.g. the role of the user making the request).

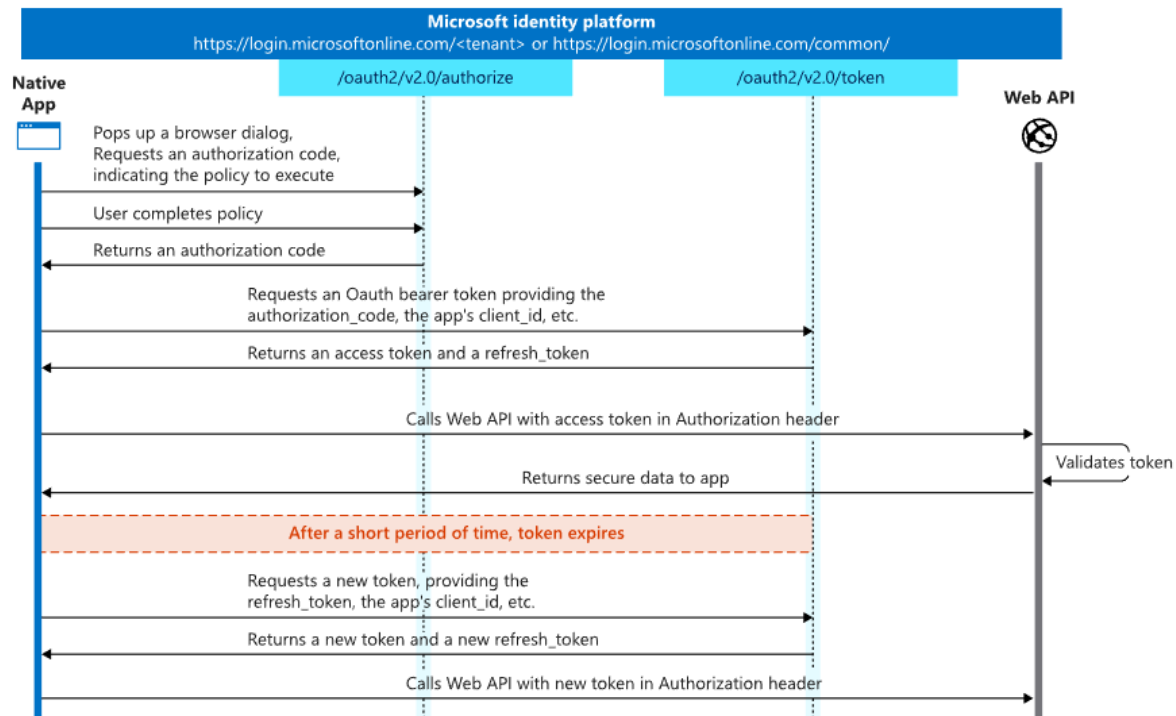
The authentication with Azure Active Directory uses Open ID connect.

We will specifically be using Azure B2C (Business to consumer) for the configuration.

OpenID Connect 1.0 is a simple identity layer on top of the OAuth 2.0 protocol. It allows Clients to verify the identity of the End-User based on the authentication performed by an Authorization Server, as well as to obtain basic profile information about the End-User in an interoperable and REST-like manner.

For Azure Active Directory, we specifically use the Authentication code flow, which is documented on the link here: <https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-auth-code-flow>

It is also detailed in the diagram below



The process involves the user navigating to a sign on screen hosted by Azure, where they can enter their credentials. Once the user has been successfully authenticated, the user is re-directed back to the return URL in the PIP Frontend application. The frontend node server (using 'Passport') then handles the authentication code and retrieves the principal details.

### 3.3.1.3 Code Configuration

For the interaction with Azure, it is used Passport <http://www.passportjs.org/>

This wraps a large amount of the interaction (retrieving tokens, user principals), and we just need to configure it to point to our Azure instance.

There are libraries for passport that integrate with Azure, see:

<https://github.com/AzureAD/passport-azure-ad>

Once configured, the user is then accessible throughout the Node JS server

```

public get(req: PipRequ
  if (req.user) {
    // currently only 2
  }

```

And you can check if a request is for a logged in user via.

```

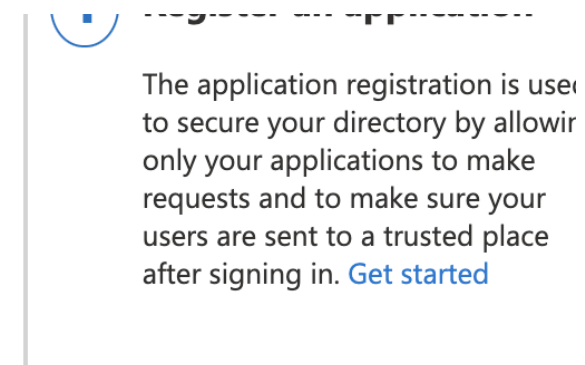
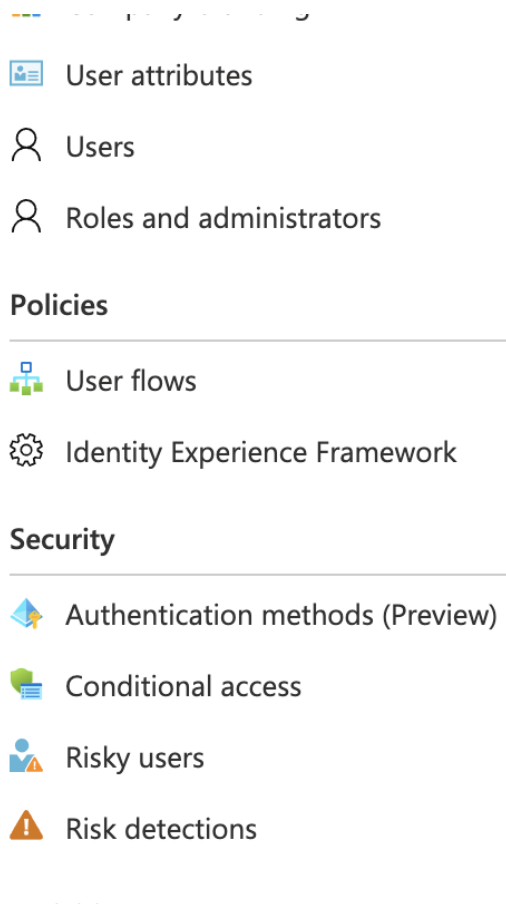
function ensureAuthenticated(re
  if (req.isAuthenticated()) {
    // ...
  }

```

### 3.3.2 Maintenance

P&I Service IDAM is set up and configured using 'User Flows' in Azure.

The User Flows provides a simpler, out of the box solution for user sign in's without the need for the need for complex configuration.



[Provide feedback](#)

## What's new

**July 31, 2021**

[API connectors for sign-up is GA and new API connectors for sign-in](#)

API connectors for sign-up built-in user flow integrate with external cloud systems like identity providers. You can also enrich tokens for your sign-in and other cloud services.


#### 3.3.2.1 Customizing the P&I Service user flow processes

The sign on and other user flow screens will be customised to provide the look and feel of a Gov UK site. Azure provides this facility which is documented here:

<https://docs.microsoft.com/en-us/azure/active-directory-b2c/customize-ui-with-html?pivots=b2c-user-flow>

This requires the hosting of a static HTML page that is publicly accessible, which Azure then uses as it's template when the user navigates to the sign on process. This page can be hosted using our frontend application.

An example of a customised Azure B2C page, with the look and feel of a gov site is below.

 **GOV.UK**

Court and tribunal hearings

[Home](#) [Sign in](#)

## Sign in with your email address

Email Address


Password

[Forgot your password?](#)

[Sign in](#)

[Help](#) [Privacy policy](#) [Cymraeg](#) [Cookie policy](#) [Accessibility statement](#) [Contact](#) [Terms and conditions](#)

**OGL** All content is available under the [Open Government Licence v3.0](#), except where otherwise stated

  
© Crown copyright

### 3.3.2.2 User Details

Details about the user are stored within the Azure Active Directory. An initial set of user attributes is provided by Azure, however this is fully extendable to include custom attributes.

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use in your directory. [Learn more about user attributes and claims.](#)

Name	Data Type	Description	Attribute type
<input checked="" type="checkbox"/> Account Status	String	The users account status dictating whether or not the account is locked or unlocked	Custom
<input checked="" type="checkbox"/> Address Line 1	String	the address line 1 of the user	Custom
<input checked="" type="checkbox"/> Address Line 2	String	the extra address line of the user	Custom
<input checked="" type="checkbox"/> City	String	The city in which the user is located.	Built-in
<input checked="" type="checkbox"/> Country/Region	String	The country/region in which the user is located.	Built-in
<input checked="" type="checkbox"/> County	String	The users county	Custom
<input type="checkbox"/> Display Name	String	Display Name of the User.	Built-in
<input checked="" type="checkbox"/> Email	String	the users email address	Custom
<input type="checkbox"/> Email Addresses	StringCollection	Email addresses of the user.	Built-in
<input checked="" type="checkbox"/> Given Name	String	The user's given name (also known as first name).	Built-in
<input checked="" type="checkbox"/> Identity Provider	String	The social identity provider used by the user to access to your application.	Built-in
<input checked="" type="checkbox"/> Identity Provider Access Token	String	The access_token returned by the OAuth identity provider.	Built-in
<input checked="" type="checkbox"/> Job Title	String	The user's job title.	Built-in
<input checked="" type="checkbox"/> Language	String	the users language	Custom
<input checked="" type="checkbox"/> Last Login	String	the date/time the user last logged in	Custom
<input type="checkbox"/> Legal Age Group Classification	String	The legal age group that a user falls into based on their country and date of birth	Built-in
<input checked="" type="checkbox"/> Postal Code	String	The postal code of the user's address.	Built-in
<input checked="" type="checkbox"/> Pub Unique ID	String	The unique identifier for the user within P&I	Custom
<input checked="" type="checkbox"/> Salutation	String	the users salutation	Custom
<input type="checkbox"/> State/Province	String	The state or province in user's address.	Built-in
<input type="checkbox"/> Street Address	String	The street address where the user is located.	Built-in
<input checked="" type="checkbox"/> Surname	String	The user's surname (also known as family name or last name).	Built-in
<input checked="" type="checkbox"/> Telephone Number	String	The users phone number	Custom
<input checked="" type="checkbox"/> User Role	String	The role of the user	Custom
<input checked="" type="checkbox"/> User's Object ID	String	Object identifier (ID) of the user object in Azure AD.	Built-in
<input checked="" type="checkbox"/> Username	String	the users username	Custom
<input checked="" type="checkbox"/> Verified	Boolean	boolean to whether or not the user is verified	Custom

The above screenshot shows the default set of fields (selected in blue) that is planned to store about the user in Azure.

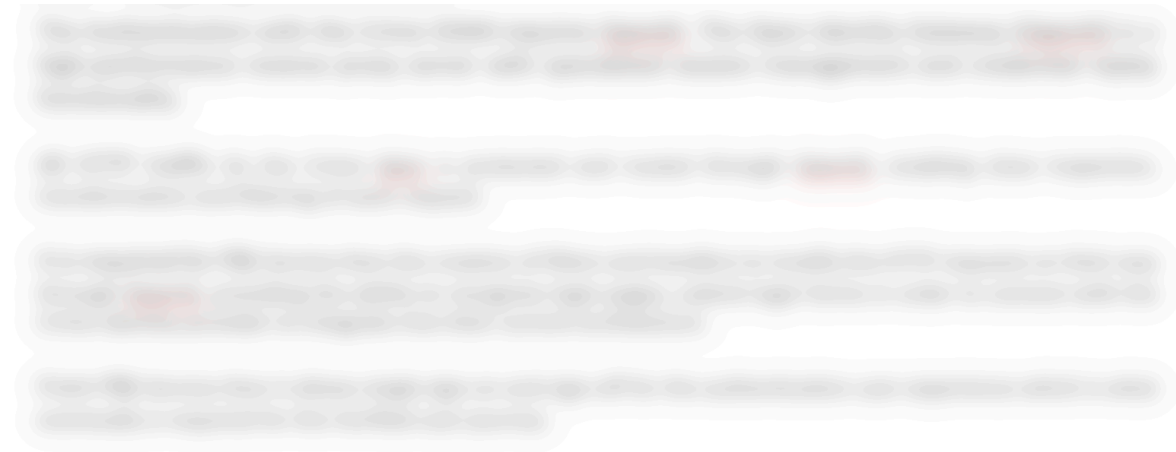
### 3.3.2.3 Environments

At time of writing, Azure B2C has been tested in a sandbox environment that has been set up by platform operations.

The setup of this environment will be migrated to use Infrastructure as Code (terraform), and the separation of each environment (Staging, Production) will be considered and discussed.



### 3.3.3 Integrating with Crime IDAM



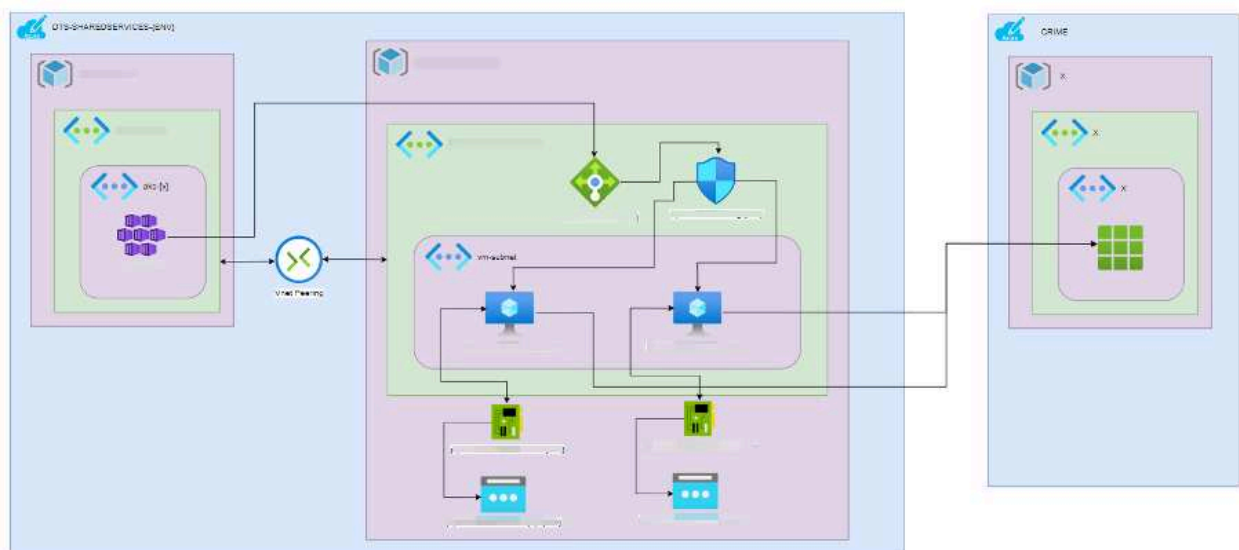
The Interface with OpenIG provided is the following one:

step	method	URL	headers	cookies		response code
				name	domain	
1	GET	<url> =		+ amtIdamCookie	.<idam_domain>	302
		https://<new-service>.<domain>/<anything>		+ FRONTEND_LD_ID	.<idam_domain>	
2	GET	https://<new-service>.<domain>/				302
		openig/SPInitiatedSSO?				
		metaAlias=/sp1&RelayState=<url>				
3	GET	automatic redirect		+ pidpSessionIdamCookie	.<idam_domain> path=/proxyidp/	302
				+ TOKEN_LB_ID	<idp_proxy>.<idam_domain>	
4	GET	login page		ISG cookies		200
5	POST	submit login form				200
6	POST	automatic redirect		+ pidpIdamCookie	.<idam_domain>	200
				+ pidplbIdamCookie	.<idam_domain>	
7	POST	automatic redirect				302
7*	GET	redirect to phone verification page if				200
		`phonenumvalidated` is `false`.				
8	GET	<url> from step 1	- CJSCPPUID	- COOKIE_LOCALE_LANG	<amt>.<idam_domain>	200
				- FRONTEND_LB_ID	<amt>.<idam_domain>	
				- amtIdamCookie	.<idam_domain>	
				- pidpIdamCookie	.<idam_domain>	
				- pidplbIdamCookie	.<idam_domain>	

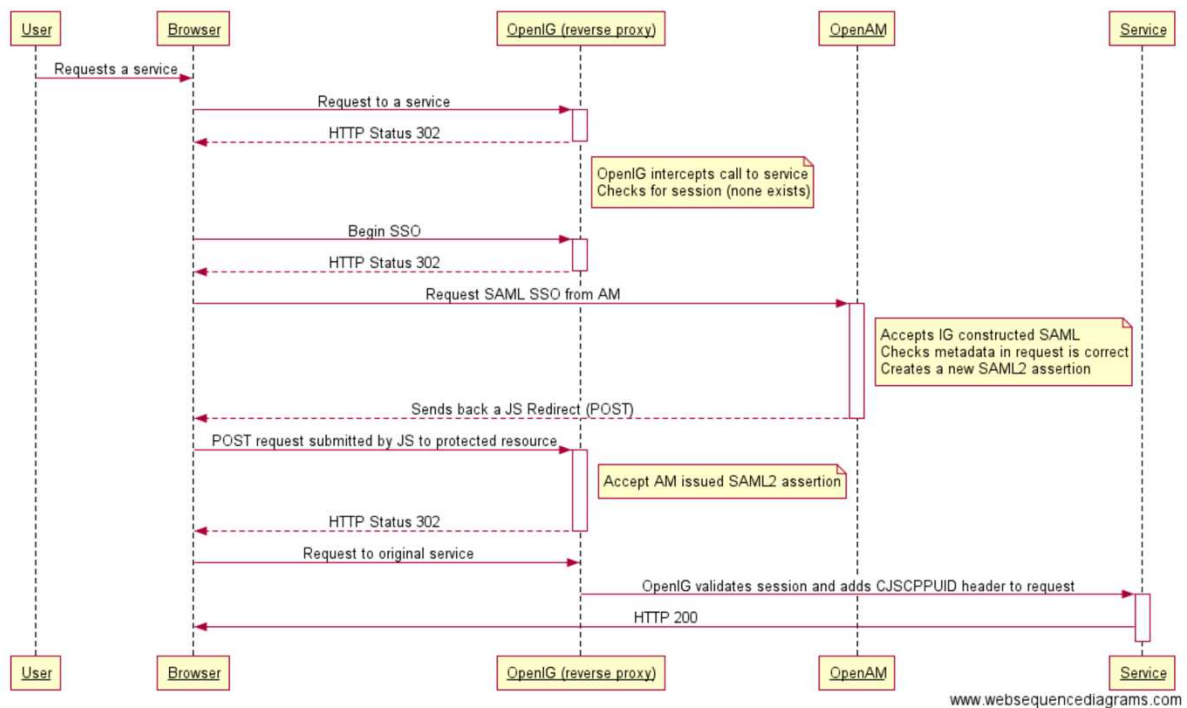
As a sample of attributes we get from Crime Idam after authenticating with a user.

```
{  
  "uid": "be7d1fff-0566-43cc-841f-34c3ea9b6aae",  
  "name": "fn_aae sn_aae",  
  "email_address": "fn@aae",  
}
```

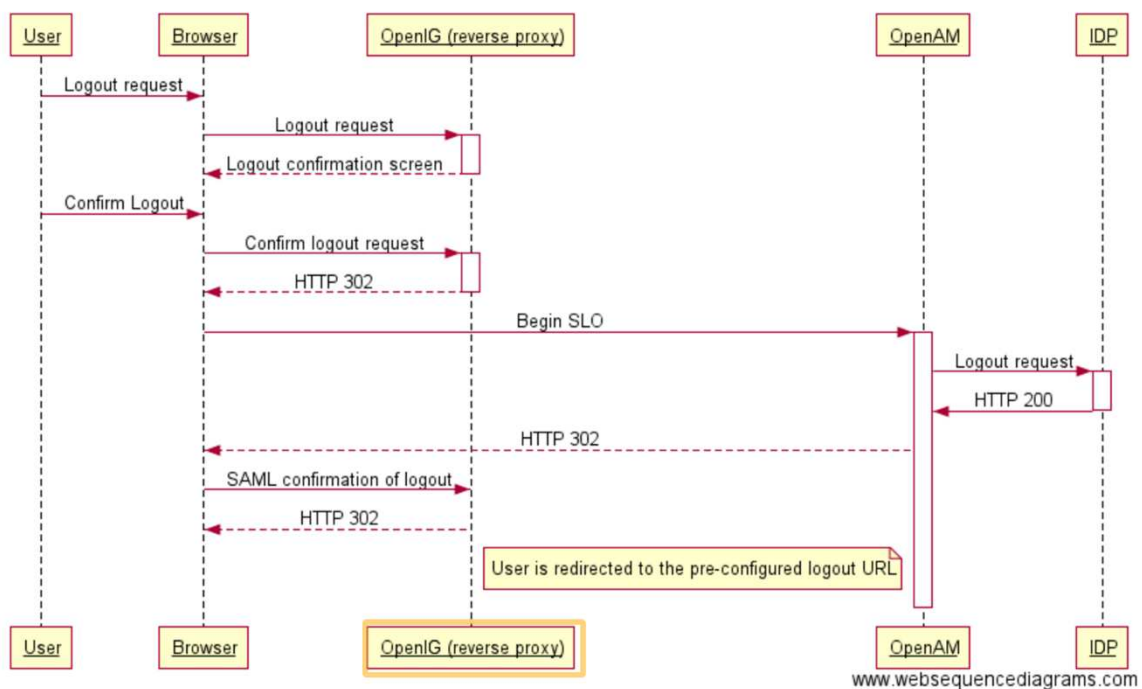
**Note :** email\_address is not part of the current Idam solution therefore there is a change request to the Idam team to include this attribute after the user gets authenticated. Crime IDAM service team requires to provide a protected service authentication to create a dedicated VM. These requires P&I Service to supply infrastructure components as described in the On-Boarding Documentation described as below:



The sign-in flow process:



The sign-out flow process:



### 3.3.3.1 Out of Scope:

The Crime IDAM journeys are provided by the Crime IDAM team and in consequence the following journeys are out of the scope of the P&I Service:

- [Self Registration,](#)
- [PIN journey - Defendant access,](#)
- [Creating Users within IDAM WebAdmin,](#)
- [User Account Activation,](#)
- [Password Reset flow,](#)
- [Account lockout - steps to unlock,](#)
- [Edit User Details,](#)
- [Suspend / Unsuspend a User,](#)
- [MFA Journey,](#)
- [Managing Roles within IDAM WebAdmin,](#)
- [Managing Services within IDAM WebAdmin,](#)
- [IDAM emails,](#)
- [Configured values,](#)
- [Re-activate Inactive User](#)

### 3.3.4 Integrating with CFT IDAM

The Authentication with the CFT IDAM requires OpenID Connect

OpenID Connect 1.0 is a simple identity layer on top of the OAuth 2.0 protocol. It allows Clients to verify the identity of the End-User based on the authentication performed by an Authorization Server, as well as to obtain basic profile information about the End-User in an interoperable and REST-like manner.

Authenticating the user involves obtaining an ID token and an access token. ID tokens are a standardized feature of OpenID Connect designed for use in sharing identity assertions.

The ID token is a token used to identify an end-user to the client application and to provide data around the context of that authentication.

An ID token is in the JSON Web Token (JWT) format and is signed according to JSON Web Signing (JWS) specifications.

#### 3.3.4.1 How the P&I Service Authentication with the CFT IDAM works:

- 1 **Get an OpenID Connect id\_token and an access\_token.**  
By leveraging an OAuth2 grant type, an application will request an OpenID Connect id\_token by including the "openid" scope in the authorization request.
- 2 **Validate the id\_token.**  
Validate the id\_token to ensure it originated from a trusted issuer and that the contents have not been tampered with during transit.
- 3 **Retrieve profile information from the UserInfo endpoint.**  
Using the OAuth2 access token, access the UserInfo endpoint to retrieve profile information about the authenticated user.
- 4 **Invalidate the user access\_token.**  
When users log out of your application, invalidate their access token to remove any risks of potentially leaking the token.
- 5 **(optionally) Refresh the user access\_token.**  
Extend the lifetime of the access\_token by refreshing it without asking the user to re-authenticate.

#### The Code

The JWT provides relevant information about the user and in consequence the users attributes to determine the role in the P&I Service.

Sample of user's attributes available in the JWT:

```
{
  "sub": "caseworker-aae@test.email",
  "uid": "be7d1fff-0566-43cc-841f-34c3ea9b6aae",
  "roles": [
    "caseworker"
  ],
  "name": "fn_aae sn_aae",
  "given_name": "fn_aae",
  "family_name": "sn_aae"
}
```

Below there is an example from a code perspective in how to access the user's data:

```
package uk.gov.hmcts.sample.oidc.client;

import com.fasterxml.jackson.annotation.JsonAlias;
import com.fasterxml.jackson.annotation.JsonProperty;
import lombok.Data;

import java.util.List;

@Data
public class UserInfo {

    @JsonAlias({"sub", "email"})
    private String email;

    private String uid;

    private List<String> roles;

    private String name;

    @JsonProperty("given_name")
    private String firstName;

    @JsonProperty("family_name")
    private String lastName;
}
```

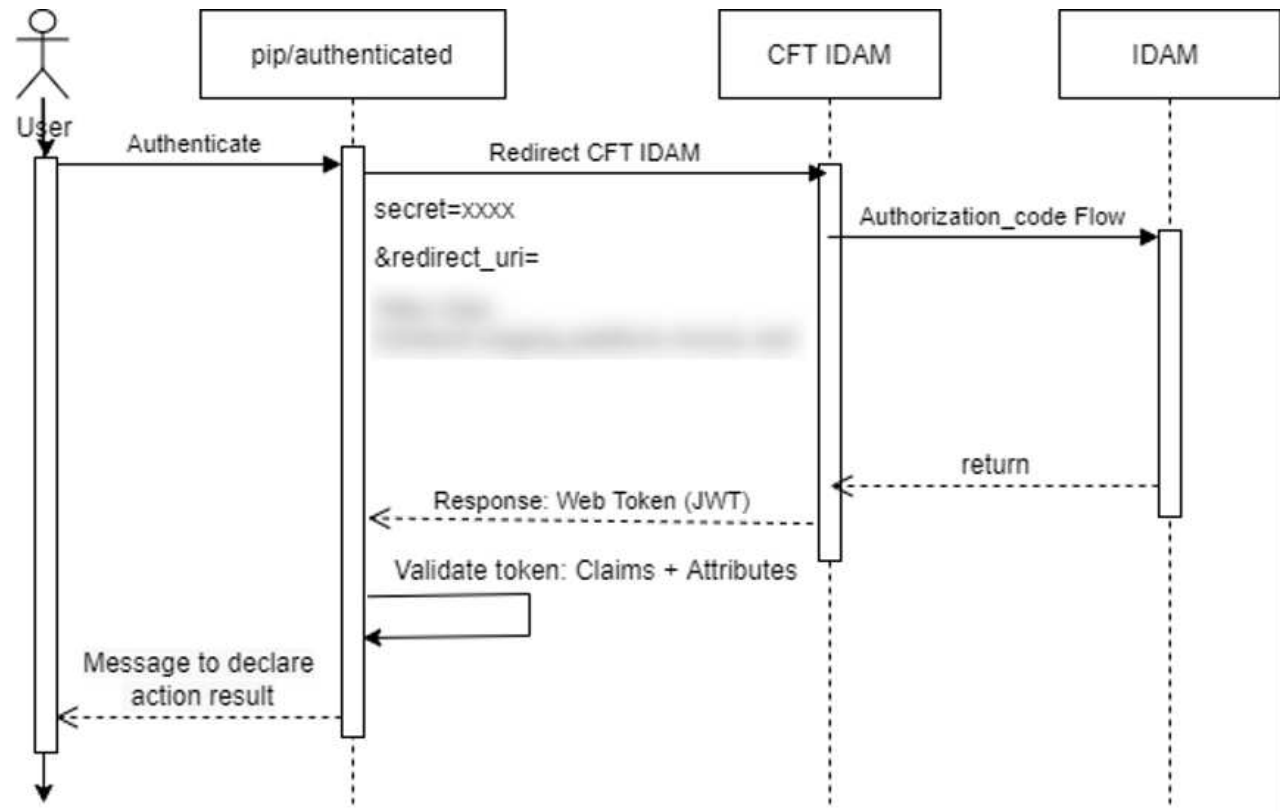
## Environments

This is the list of environments available for the CFT IDAM integration within P&I Service.

Environment	URL
PROD	
AAT	
Demo	
ITHC	
PREVIEW	
SANDBOX	

Example below:

The CFT IDAM has created an ad Hoc service for P&I Service to allow the authentication mechanism take place.



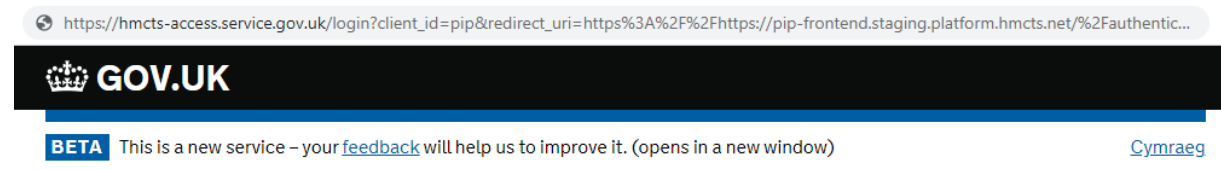
The requirement is to have a client secret to be shared with CFT IDAM. This secret has been issued by P&I Service and stored securely in the Azure Portal KeyVault.

The P&I Service will redirect the user for the CFT IDAM with the following parameters: **secret + redirect\_uri**.

Example below:

[https://.../login?client\\_id=pip&redirect\\_uri=https%3A%2F%2Fhttps://.../%2Fauthenticated&ui\\_locales=en&response\\_type=code&state=2e1631e7-d354-4287-9e01-a61fa64f2dc4](https://.../login?client_id=pip&redirect_uri=https%3A%2F%2Fhttps://.../%2Fauthenticated&ui_locales=en&response_type=code&state=2e1631e7-d354-4287-9e01-a61fa64f2dc4)





Once successfully authenticated the user will be redirected to P&I Service to the **redirect\_uri** url originally provided

#### 3.3.4.2 Out of Scope:

The CFT IDAM journeys are provided by the CFT IDAM team and in consequence the following journeys are out of the scope of the P&I Service:

- [Self Registration,](#)
- [PIN journey - Defendant access,](#)
- [Creating Users within IDAM WebAdmin,](#)
- [User Account Activation,](#)
- [Password Reset flow,](#)
- [Account lockout - steps to unlock,](#)
- [Edit User Details, Suspend / Unsuspend a User,](#)
- [MFA Journey,](#)
- [Managing Roles within IDAM WebAdmin,](#)
- [Managing Services within IDAM WebAdmin,](#)
- [IDAM emails,](#)
- [Configured values,](#)
- [Re-activate Inactive User](#)

### 3.4 Data Storage

#### 3.4.1 Access of the data

The P&I Service is a publisher and not the owner of data. As such it is not a system of record or the master of any data, aside from any it holds for specific logging/auditing requirements. The data it holds is provided by its data sources that comes from CFT, Crime and other potential HMCTS sources. It is the responsibility of those data source owners to be responsible for the content of both the data and meta-data they send to the P&I Service.

Data will be exposed via a GDS compliant web portal, via API based publication services used for third party access and also via attachments for subscribers using the gov notify service email communication.

Data will be stored within the P&I Service and availability of publications will be driven by retention periods provided by the data sources in their meta-data (e.g. from/to dates – please see references on the API Schema Specification 0.16 in section 3.1 ).

The P&I Service will not manipulate nor change the content of the data stored that was originally received from the data providers. Where publications need to be provided to different user roles it is expected that the data source will provide drafts that are flagged in the meta data for the correct recipients, such that P&I Service will display different draft views for Legal Professionals and for General Public.

#### 3.4.2 Storage Component View

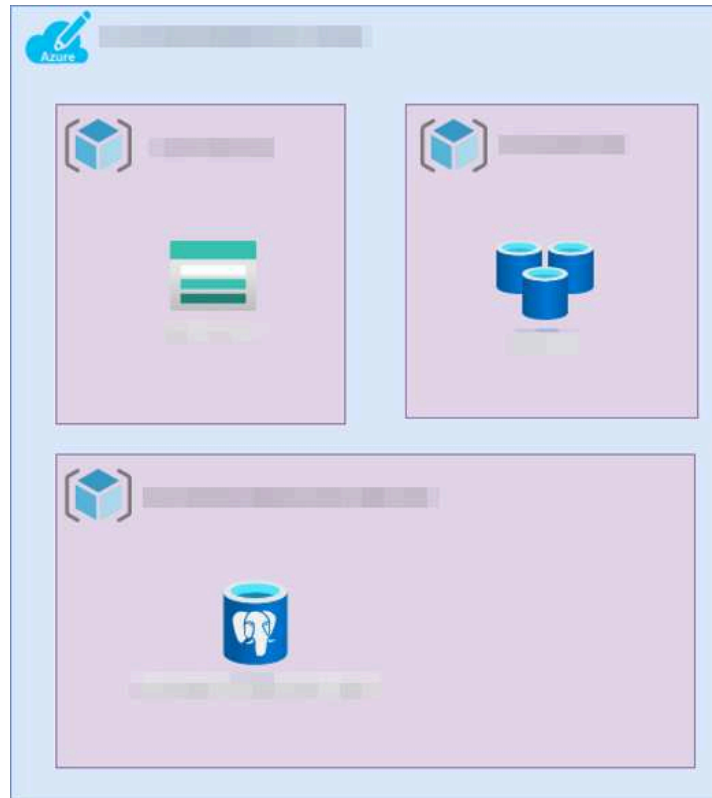
Data Access is handled differently within each environment due to the usage of the environment. However, for every environment they are all secured by a Virtual Network (VNet), meaning you require to be on the Virtual Private Network (VPN) to access the resources. If they are not protected by a VNet then they will have a Private Endpoint configured.

Lower environments like Sandbox and Development only require to be part of the DTS SDS Developer Active Directory group in Azure, which is managed by a GitHub repository and Platform Operations approval.

This means only approved and known users can get access to the Data and Access keys to see the data.

For Staging as well as the above, you will also need to request access via the Microsoft My Access self-service. If you have permissions to request the access, then you can request read access for the staging environment so you can access the keys.

For Production you will need to do the above, but also will need to have Security Clearance before the access will be granted.

**Postgres DB**

The P&I Service will be using a Postgres database to store data. The P&I API Service read/write data from this data storage. The reason for choosing this DB model is due its relational DB properties and the capabilities to work with blobs supporting a few file formats.

**Azure Storage Account**

Azure storage account v2 has been chosen to support the blob storage activities, list court publications within P&I. This will enable the storage data ingestion for handling high volume of workloads, archives.

**Redis Cache**

The Redis Cache will serve the P&I Service User interface for supporting the front-end functionalities. The only purpose of using Redis is for Caching Data. It saves data for applications, servers, and web browsers.

**Azure B2C**

The P&I Service IdAM users related logic will be stored using the User profile attribute B2C. The directory user profile comes with a built-in set of attributes.



### 3.5 Reference Data

Reference data will remain static in the application. P&I Service will be in charge to maintain and upload this data in the application to be used. For this process there could be a need to add/update or delete some data on base of requirements.

There are two type of reference data:

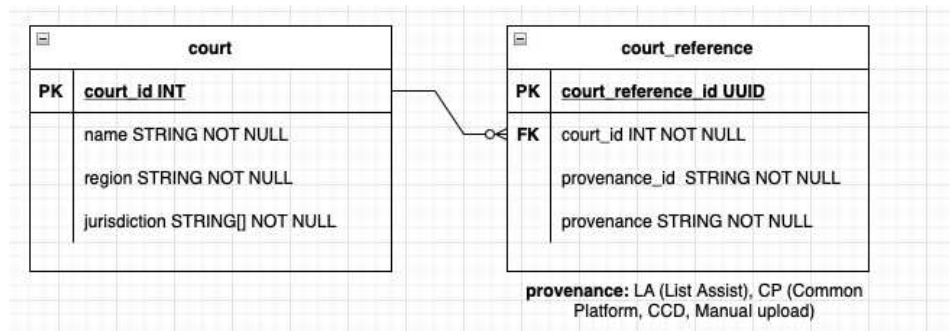
1. Court lists
2. Court Event Glossary

#### 3.5.1 Court lists

This file will contain list of all the courts with all the relevant information like name of the court, region and jurisdiction.

This file will be provided in CSV format and needs to be uploaded manually into the database. Once file will be uploaded, it will be used this data to show court information on the application.

Table Structure:



Sample of CSV file is below:

```

Court Desc,Region,Jurisdiction,Provenance,ProvenanceID
ALTON MAGISTRATES COURT,SOUTH WEST,MAGISTRATES COURT,Forced in,788490
ASHFORD TRIBUNAL HEARING CENTRE,South East,TRIBUNAL,Forced in,239985
BARKINGSIDE MAGISTRATES COURT,London,MAGISTRATES COURT,Forced in,218723
BARNSTAPLE MAGISTRATES; COUNTY AND FAMILY COURT,South West,MAGISTRATES COURT; Family Court; County Court,Forced in,774335
BARROW-IN-FURNESS COUNTY COURT AND FAMILY COURT,North West,COUNTY COURT; Family Court,Forced in,761518
BASILDON MAGISTRATES COURT AND FAMILY COURT,South East,MAGISTRATES COURT; Family Court,Forced in,538351
BASINGSTOKE COUNTY COURT AND FAMILY COURT,South West,COUNTY COURT; Family Court,Forced in,457273
BATH MAGISTRATES; COUNTY AND FAMILY COURT,South West,MAGISTRATES COURT; Family Court; County Court,Forced in,411234
BEDFORD AND MID BEDS MAGISTRATES COURT AND FAMILY COURT,South East,MAGISTRATES COURT; Family Court,Forced in,446255
BERWICK UPON TWEED MAGISTRATES COURT,North East,MAGISTRATES COURT,Forced in,500233
BEVERLEY MAGISTRATES COURT,North East,MAGISTRATES COURT,Forced in,359723
BEXLEY MAGISTRATES COURT,London,MAGISTRATES COURT,Forced in,381649
BEXLEYHEATH SOCIAL SECURITY AND CHILD SUPPORT TRIBUNAL,London,TRIBUNAL,Forced in,29955
BIRKENHEAD COUNTY COURT AND FAMILY COURT,North West,COUNTY COURT; Family Court,Forced in,444097
BIRMINGHAM MAGISTRATES COURT,Midlands,MAGISTRATES COURT,Forced in,784730
BLACKBURN COUNTY COURT AND FAMILY COURT,North West,COUNTY COURT; Family Court,Forced in,150431
BLACKBURN MAGISTRATES COURT,North West,MAGISTRATES COURT,Forced in,215156
BLACKBURN SOCIAL SECURITY AND CHILD SUPPORT TRIBUNAL,NORTH WEST,TRIBUNAL,Forced in,107017
  
```

##### 3.5.1.1 Initial Upload:

Data for the courts will be provided in CSV format. A file will be uploaded to Postgres SQL database. Process to upload the data will be as followed:

- An endpoint will be created in the data management api which will take a CSV file as a parameter.
- Once file is selected, it will save the data to tables court and court\_reference in Postgres SQL database.
- Once *courts* table will be populated, link the *provenance\_id* with *court\_id* which were generated in above step and populate *court\_reference* table.
- *court\_id* and *provenance\_id* will be of type INT.

### 3.5.1.2 Maintenance updates:

There is a possibility that some courts have been closed or some information about the courts has been changed. In this case, P&I Service needs to update the data in our database. A CSV file will be obtained which will contain all the data in it. While uploading the new data, it will be needed to consider following factors:

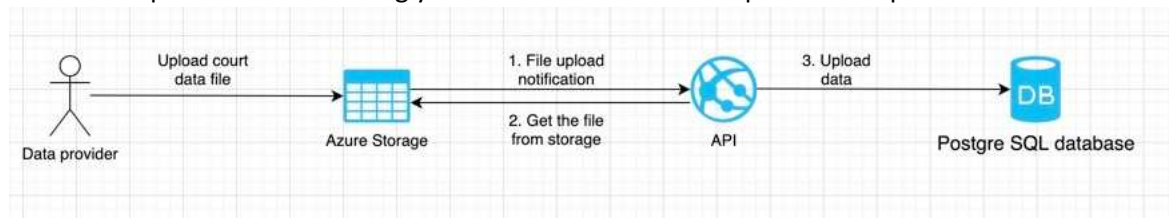
- Is there any new court to be added?
- Is there any court data which has been updated?

In case of updating, a column *date\_updated* in *courts* table will be updated with current date.

For the time being, this whole upload process will be manual, but in future, it might be needed to automate this process.

### 3.5.1.3 Automated Data Reference – Future release

For process automation, as mentioned above, a CSV file to the P&I Service will be provided. Then the file will be updated and accordingly the data in database. The process to upload the data is as follow:



1. The reference Data provider will upload the data to Azure storage using FTP.
2. Once file is uploaded, API will get notification from the Azure about the new uploaded file.
3. API will pick the file, store into the memory, process the data and store into the database.

An endpoint in API will be created which can be called from *Swagger*. Using *swagger*, admin can manually trigger the upload functionality. API endpoint will pick the latest uploaded file and update the data accordingly.

### 3.5.1.4 API processing:

Once API will get notification about the newly uploaded file. Data update process be as follows:

- It will validate the most recently uploaded file.
- Store the file into Postgres SQL database temporary table
- Once file has been saved in temporary table, it will call a database stored procedure which will upload the data to the *court* and *court\_reference* tables.

Once data will be stored in temporary table, database stored procedure will do process the data as follows:

- Find all the newly added courts and add to the *Courts* and *court\_reference* tables.
- Find all the data which has been updated for the existing courts. Update all the relevant fields in the table including region and jurisdiction.
- Once data will be updated, remove all the data from temporary table.

### 3.5.1.5 Court Reference data sources:

We are expecting to receive court reference data from different sources. These sources are:

- ListAssist
- Common Platform
- CCD
- Manual ingestion

ListAssist will only provide us data for those courts which are LIVE. We can also upload the court data using Manual ingestion. There is a possibility that same court will appear in different sources but *provenance\_id* will be different for each source. So, as described above in Initial upload and Maintenance updates section, these external *provenance\_id* will be mapped to internal *court\_id*.

When P&I Service receives a publication for a specific court, we will map the external court ID using provenance information in the API request headers and save the publication using internal court ID. P&I Service application will use internal court id for the courts. For example, P&I received publication from ListAssist. It will have ListAssist court ID (123) and *provenance* (listAssist) in the header. API will match *court* ID and *provenance* in *court\_reference* table, find the P&I court ID and save publication against P&I court ID.

There is a possibility that P&I will receive blob having a court ID which does not exist in P&I. In this case, the original ID is retained and the blob is marked as an errored/orphaned court ID (Assumption (Example Court ID "100" becomes "NoMatch100")). API will return "NoMatch100" and we will use these records later for reporting purpose.

### 3.5.2 Court Events Glossary:

Court event glossary page will contain all the court event statuses which will be mapped to daily court status page of any court. When user clicks on the status in current status column, it will be redirected to Court Event Glossary page. Court event status on Live status update page will be linked through name.

This data will be saved in JSON file, and it will be stored on Azure blob storage. Application will read the data from the JSON file and displayed on the Court Event Glossary page.

**File structure:**

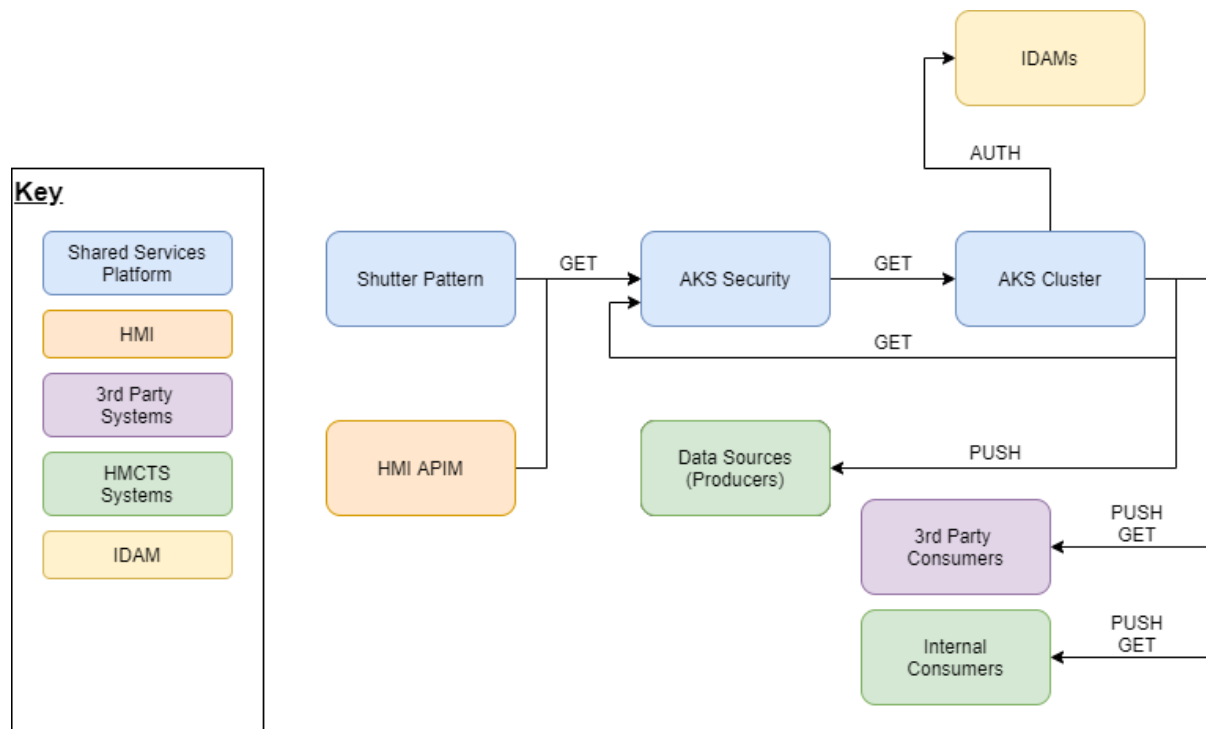
```
[
  {
    "name": "Adjourned",
    "description": "The case has been adjourned."
  },
  {
    "name": "Appeal Interpreter Sworn",
    "description": "An appeal could require one or more interpreters for the witnesses and/or the appellants. All interpreters are required to take an oath prior to performing interpretations for the court."
  },
  .....
]
```

### 3.6 Network

The Publication and Information project will use the existing Shared Service Platform for its networking and security. Requests for the User Interface will go through the Shutter Pattern as documented by Platform Operations <https://hmcts.github.io/ways-of-working/path-to-live/shutter.html>

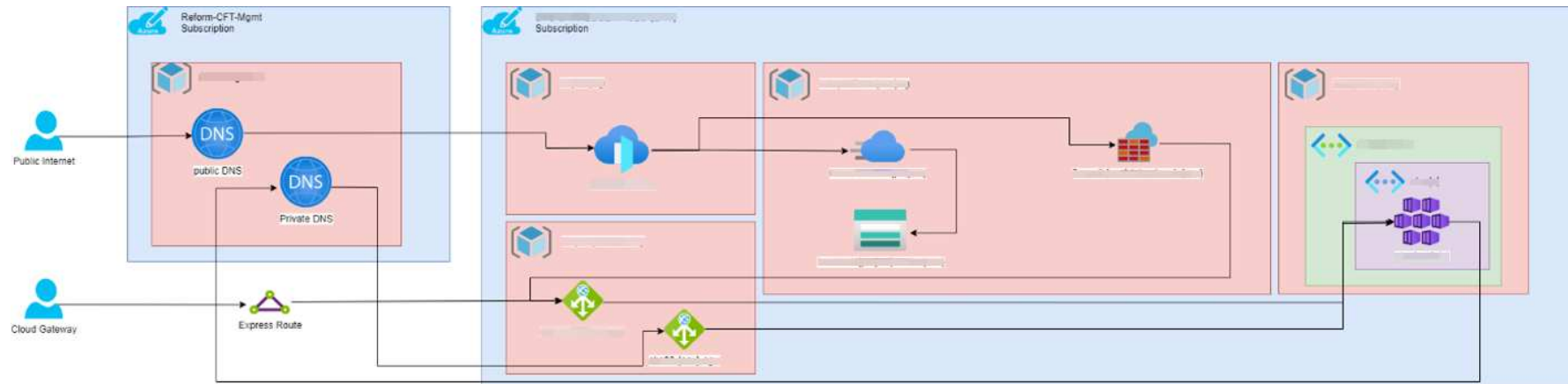
Requests for the project APIs will go straight through the AKS Security Layer, which is only internally accessible

#### 3.6.1 Logical Network Component Diagram





### 3.6.2 Physical Network Implementation



The Shutter Pattern provides an Azure Frontdoor, which will provide TLS security and a built in WAF. It will route traffic to the relevant Palo Alto Firewall for further restrictions and security set out by the HMCTS Standards.

From there traffic for both internal and external will go through the relevant Application Gateway, before going to the Azure Kubernetes Service for the application. Even requests between applications will need to go back through the Application Gateway.

On-Premises or Traffic over the Cloud Gateway consume and make calls via the Express Route without going over the internet.

### 3.6.3 Confidentiality, Integrity, Availability View

The information and assets within scope of this architecture have been assessed.

Item	Assessment
Confidentiality	The Shutter Pattern is a generic facility for data of all levels of classification. Therefore, the highest classification must be considered. Disclosure of this information could lead to reputational damage to HMCTS and potential harm to individuals (both judicial and public).
Integrity	Corruption of information within API calls could lead to incorrect judgements or outcomes being published. Business processing that fails to match the correct information with the stakeholders (e.g., case identifier, name and address) could lead to multiple complaints and claims for compensation and hence reputational damage.
Availability	The Shutter Pattern/Application Gateway is common to all API's and therefore unavailability would affect many functions. Partners would have a lower opinion of HMCTS. Only if an embarrassingly long outage were to be made public could reputational damage occur.

### 3.6.4 Security Controls View

P&I Service will be following the HMCTS approved PaaS Patterns for the resources to make sure they follow the best practices and are secure. These practices can be read at:

<https://tools.hmcts.net/confluence/display/DACS/v2+-+F.2a+Architecture+Patterns>

Control ID	Category	Description
001	Developer Authentication	Individual user IDs must be used to maintain accountability, any access to the system and its components must be separate from one another and provided to DevOps in a controlled process.
002	Developer Authentication	Passwords must never be stored, displayed or transmitted in cleartext, they must be stored in a separate location to system data.
003	Developer Authentication	Accounts must be locked after 5 incorrect login attempts
004	Developer Authentication	Complex passwords must be enforced that comply with the MOJ password policy
005	Developer Authentication	Role Based Access Controls (RBAC) and entitlements based on the principle of least privilege must be enforced
006	Developer Authentication	Multifactor Authentication must be used for access to the system
007	App Sec	The application must be designed and implemented to prevent common security attacks such as the OWASP Top 10
008	App Sec	Applications must validate all data input, and reject input that: <ul style="list-style-type: none"> <li>is not formatted as expected</li> </ul>

Control ID	Category	Description
		<ul style="list-style-type: none"> <li>falls outside the bounds</li> <li>contains code and characters other than expected</li> <li>contains embedded queries that include illegal characters</li> <li>contains any other unexpected content</li> </ul>
009	App Sec	All the request must be made over a secure channel, Insecure services, protocols, etc. must not be used
010	App Sec	Full penetration test must be performed before go-live and annually thereafter in accordance with detailed scope and test plans
011	Encryption	Data in transit must be encrypted using at least 128-bit TLS with v1.2 preferred
012	Encryption	Data at rest must be encrypted
013	Encryption	Applications must never use self-signed certificates
014	Encryption	Wildcard certificates are not permitted in the production environment. All certificates must match the Fully Qualified Domain Name for internet facing services
015	Data Management	The use of production data for testing must be explicitly authorised
016	Asset Management / Dev Access	Assets must be identified, classified and documented within an asset inventory with a defined owner
017	Asset Management	Access to sensitive/confidential data must be restricted to those with a valid business need and for a specified time period
018	Incident Management	Events must generate an alert to the centralised Security Incident and Event Monitoring (SIEM) tool
019	Incident Management / Dev Access	Privilege account usage must generate log events and these events must be reviewed periodically by the Security team
020	Logging	Access to logs must be restricted to those who have an appropriate business requirement
021	Logging	Applications must log the following security-related events with userID and date/timestamps: <ul style="list-style-type: none"> <li>authentication (success of failure)</li> <li>authorisation/permission granting</li> <li>all configuration changes performed using a privilege account - e.g. admin</li> <li>data access attempts</li> <li>data deletions</li> <li>data transfers</li> <li>user lockouts</li> </ul>
022	Hosting	The server must be configured running only the required services. All unrequired services must be disabled to lower the attack surface
023	Hosting	The environments (i.e. production, staging, test) must be logically isolated from one another.
024	Account privileges	All accounts (user or system) must be configured to provide the least privileges

### 3.6.5 Protective Monitoring View

The following protective monitoring views must be adhered to:

1. Success and failure of job execution must be reported.
2. Login failure must be recorded and reported on.
3. Privileged access must be logged and reported on.
4. Data exports must be logged and reported on.

Both CFT and Crime IDAMs monitor user logins and log each user attempt, and if it was successful or unsuccessful, which will also be captured within the P&I Service

The egress endpoints to external systems will have monitoring in place to record each external request. This will record the service token of the requesting service, along with what records it was attempting to retrieve. Platform Operations should be able to provide further details on each request if necessary.

### 3.6.6 Environments and Automation

The P&I Service project will use the standard HMCTS DevOps approach to the use of environments and the respective activities of build, test, deploy (<https://hmcts.github.io/ways-of-working/#ways-of-working>)

Azure subscriptions will be used for the environments needed with Continuous Integration and Continuous Development (CI/CD) approach adopted using the Jenkins component to manage the CI/CD pipeline tasks.

Terraform scripts will be written to manage the build and deployment of the Azure Infrastructure that supports the SDS Platform and the project.

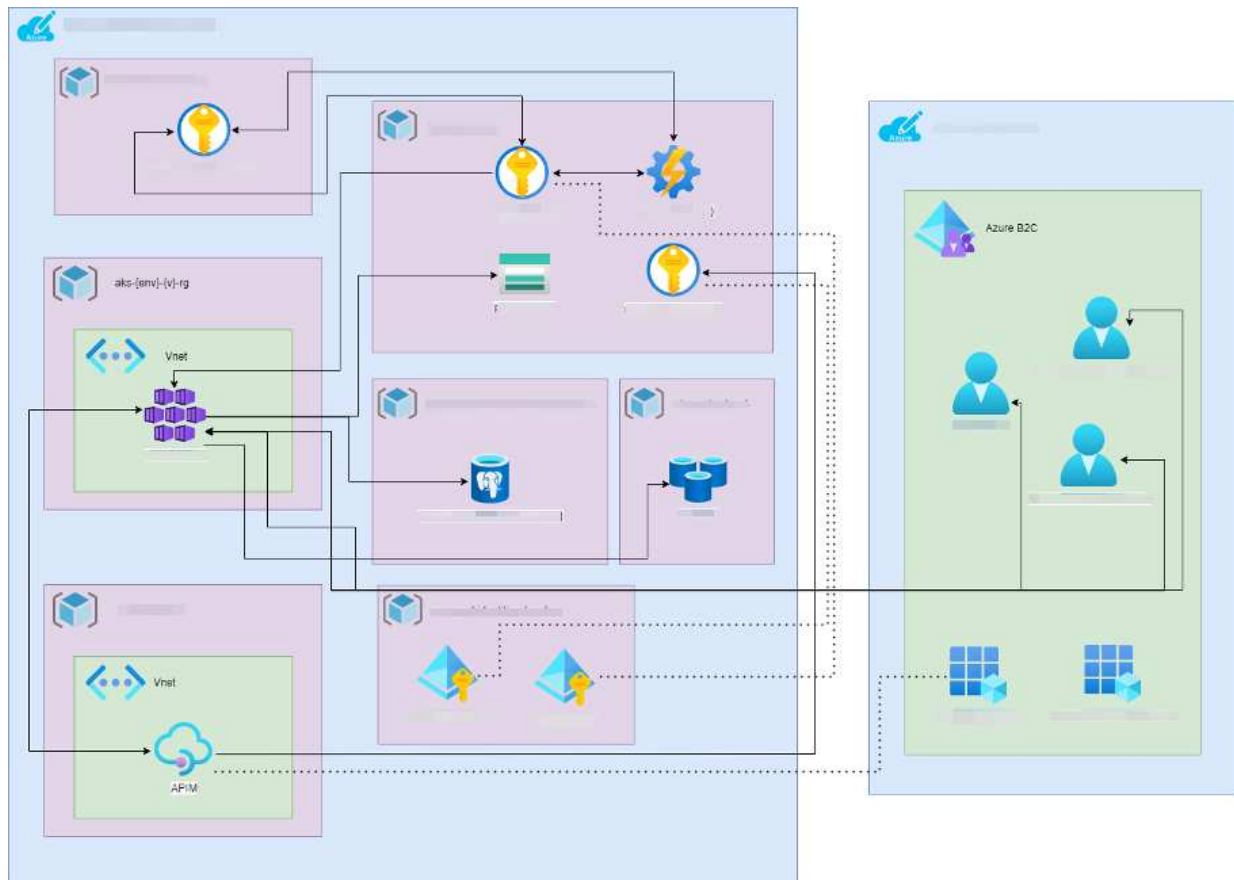
The code will be linked to a Github repository which will be used for version control.

All environment management such as restarting or making large changes will be entirely controlled by the Platform Operations team. The P&I Service team will manage the code that is on the environments and some basic configuration.

Environment	Activities	Connectivity needed	Rationale
<b>SANDBOX</b>	POCs, Development, unit testing changes, building tests (needs connectivity to S&L)	Connectivity to S&L stubs, Connectivity to AKS for running automated testing tools	Allows parallel dev activities if Dev is unavailable. Is a test bed for testers to build tests before TEST environment deployment
<b>DEV</b>	Development and unit testing, acceptance testing of tests before merge to master (ensures integrity of PR merge to master)	HUB access to Active Directory, connectivity to S&L stubs, Connectivity to AKS for running automated testing tools	No need to connect to S&L

Environment	Activities	Connectivity needed	Rationale
<b>TEST</b>	Dev integration testing, smoke, acceptance(whitebox, blackbox, boundary, value partition,static against LLD), CDCs (contract testing),Functional testing (connectivity of S&L, lifecycle behaviour -create, list, amend hearings), Security (pipeline jobs), Performance(pipeline jobs)	HUB access to Activity Directory, Connectivity to S&L UAT/Test environments, connectivity to PACT broker (where is this?), S&L need access to PACT broker	Full complement of tests
<b>STAGING</b>	Performance testing and ITHC	HUB access to Active Directory, Connectivity to S&L UAT/TEST environments, Connectivity to CFT and CRIME, Connectivity to AKS, ITHC connectivity, SPLUNK and Dynatrace connectivity	All connectivity as its replicating PROD
<b>PROD</b>	Deployment of production ready code from master	Connectivity to S&L prod end points, Connectivity to CFT and CRIME prod end points	Production connectivity to CFT, CRIME and S&L endpoints

### 3.6.7 Environment component View



#### Diagram Notes

'env' is a representation for the environment, which will either be the short or long version. For example, Staging could be 'staging' or 'stg'.

'v' represents the Azure Kubernetes Services cluster version offered by the Shared Services Platform. This will either be '00' or '01', with '00' being the primary cluster.

'NON' represents an option between the two versions. 'PIP-AD-NON-PROD' is for all non-production systems and 'PIP-AD-PROD' will be for production usage.

The goal of this project is to utilise as much of the Shared Services platform as it can. This includes the ingress for front end access via the Azure Front Door and backend access for APIs to the Azure Kubernetes Service. This will also be the networking and security, therefore in the above diagram we have only included the bespoke resources supplied for Publication and Information project.

The diagram below illustrates the core tasks undertaken during the CI/CD pipeline process for our code and infrastructure.

For deployments we are using the Common Pipeline platform provided by the Platform Operations team. Below is the diagram explaining the design and the flow of code with more information on the HMCTS Ways of Working website <https://hmcts.github.io/ways-of-working/common-pipeline/common-pipeline.html#common-pipeline>

There will be no exceptions to the programme standard approach, and our pipelines will follow the same extensive tests seen elsewhere through the programme.

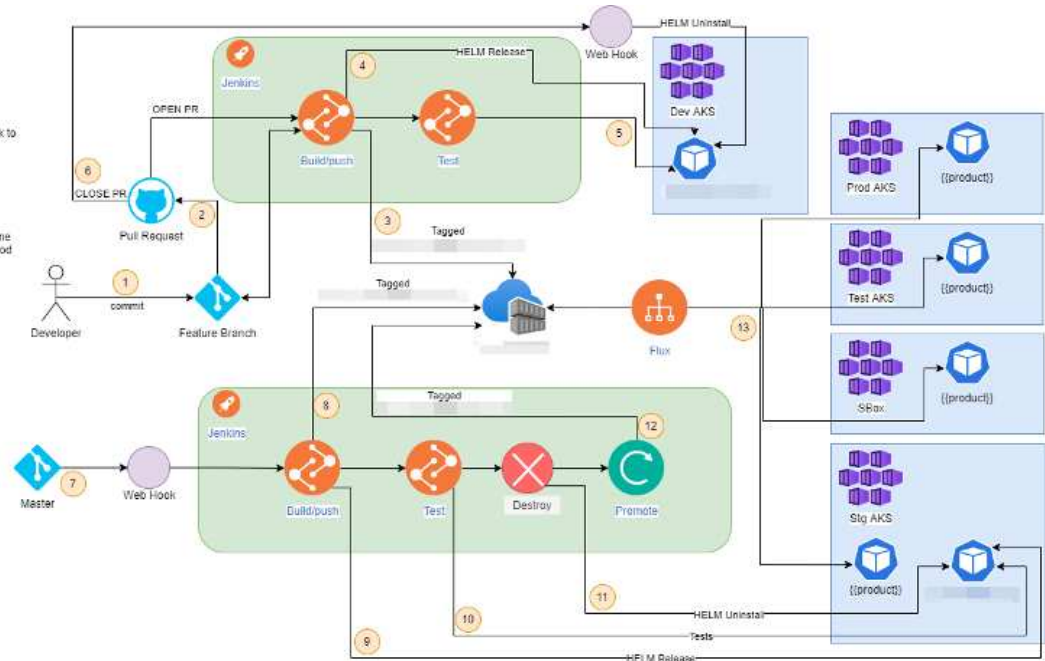
**End to End Process**

1. Commit Changes to Feature Branch
2. Create Pull Request
3. Jenkins will push image with tag
4. Jenkins will force release to AKS
5. Jenkins will run automated tests
6. Close PR will Merge Feature with Master
7. The merge with master will trigger a webhook to start Jenkins pipeline
8. Jenkins will push image with tag
9. Jenkins will force release to AKS
10. Jenkins will run automated tests
11. Jenkins will destroy POD
12. Jenkins will promote image
13. Jenkins will update flux repo with new image name
14. Flux will see new changes and deploy new pod

**Notes**

Tags are the below pattern:  
 {{env}}-{{commit}}-{{timestamp}}

Pods are named:  
 PR Pod = {{product}}-pr-{{pr id}}  
 Auto Testing Pod = {{product}}-staging  
 Pod = {{product}}

**3.6.8 Monitoring and Alerting with Dynatrace**

The solution will be supported by the HMCTS PlatOps team in the first instance, and will require monitoring, logging and alerting in accordance with Reform operational standards. Any issues that cannot be resolved directly by DevOps will be escalated to the P&I Service delivery team for support. (Logging and Monitoring Policy can be found here:

**Logging and Monitoring Policy**

The Dynatrace application will be used to monitor System events and activities. The Dynatrace agent and API will monitor and Log. Within the P&I Service stack the Azure Monitor component will be used to monitor from a system administrator perspective such as End point statuses.

**3.6.9 Failover and Disaster Recovery**

Automatic Failover is built into the Azure Frontdoor that if it does detect a unhealthy application then it will switch to use the static error page served for a Azure Storage Account. This is described in the HMCTS Ways of Working as the Shutter Pattern <https://hmcts.github.io/ways-of-working/path-to-live/shutter.html>.

If the Azure Kubernetes Cluster primary has a fault that the Platform Operations Team cannot resolve within the SLA times, then there is an alternative cluster that will be switched to.

Azure Storage Accounts by default are replicated across two regions in Azure for data resiliency and can be failed over to, unless done automatically, with a manual click of a button.

Availability is described here:

[Availability Standards](#)

DR here:

[Disaster Recovery](#)





### 3.6.10 Backup and Restore

Backups and restoration will be performed as follows:

1. The P&I Service project will back up the list of registered media subscribers whenever a new user is added onto the system.
2. No backups of the published information will be performed by the P&I Service project as we are not the source of information.
3. Backups of logs will be managed by the Platform Operations team as per programme standard.
4. Any restoration will have to be carried out with consultation from the Platform Operations team.

Azure PostgreSQL Server has built in backup and restore as documented in the [Azure Documents](#). It has been configured 35 days backup of the databases, which is Geo replicated across regions to prevent regional failure.

### 3.6.11 Archiving

No archiving requirements and no exceptions from the programme standard approach. It is up to the source systems providing data to the P&I Service, to ensure that the data is accurate and up-to-date.