



*<* *PRODUCT/SERVICE NAME>*

*<mmm> <YYYY>*

Solution Architecture

ISBN

xxx-x-xxx-xxxxx-x [PRINT]  
xxx-x-xxx-xxxxx-x [PDF]  
xxx-x-xxx-xxxxx-x [DOCX]

Creative Commons

With the exception of the Commonwealth Coat of Arms, the Department’s logo, any material protected by a trade mark and where otherwise noted all material presented in this document is provided under a [Creative Commons Attribution 4.0 Australia](https://creativecommons.org/licenses/by/4.0/) licence.  
The details of the relevant licence conditions are available on the Creative Commons website (accessible using the links provided) as is the full legal code for the [CC BY 4.0 AU licence](https://creativecommons.org/licenses/by/4.0/legalcode).

The document must be attributed as the (insert document title).

Contents

[1 Introduction 6](#_Toc82761238)

[1.1 Document Scope 6](#_Toc82761239)

[1.1.1 In Scope 6](#_Toc82761240)

[1.1.2 Out of Scope 6](#_Toc82761241)

[1.2 References 6](#_Toc82761242)

[1.3 Definitions 6](#_Toc82761243)

[1.4 Stakeholders 6](#_Toc82761244)

[1.5 Document Revisions 7](#_Toc82761245)

[1.6 Assumptions 7](#_Toc82761246)

[2 Overview 8](#_Toc82761247)

[2.1 Data Holdings and Classification 8](#_Toc82761248)

[2.2 Objectives 8](#_Toc82761249)

[2.3 Functional Requirements 8](#_Toc82761250)

[2.4 Non-functional Requirements 8](#_Toc82761251)

[3 Architecture 9](#_Toc82761252)

[3.1 Application 9](#_Toc82761253)

[3.1.1 Target State Architecture 9](#_Toc82761254)

[3.1.2 System Integration 9](#_Toc82761255)

[3.1.3 User Experience 9](#_Toc82761256)

[3.1.4 Component Overview 9](#_Toc82761257)

[3.2 Security 9](#_Toc82761258)

[3.2.1 Identity and Access Management 9](#_Toc82761259)

[3.2.2 Network Security 9](#_Toc82761260)

[3.2.3 Data Protection & Governance 9](#_Toc82761261)

[3.2.4 Audit & Logging 9](#_Toc82761262)

[3.3 Operations 10](#_Toc82761263)

[3.3.1 Availability 10](#_Toc82761264)

[3.3.2 Scalability 10](#_Toc82761265)

[3.3.3 Backup Strategy 10](#_Toc82761266)

[3.3.4 Disaster Recovery 10](#_Toc82761267)

[3.3.5 Consumption & Performance Monitoring 10](#_Toc82761268)

[3.3.6 License Management 10](#_Toc82761269)

[3.3.7 Support 10](#_Toc82761270)

[4 Development Approach 11](#_Toc82761271)

[4.1 Environments 11](#_Toc82761272)

[4.2 Tools & Libraries 11](#_Toc82761273)

[4.3 Practices 11](#_Toc82761274)

[4.4 Release Management 11](#_Toc82761275)

[5 Risk Management 12](#_Toc82761276)

[6 Bill of Materials 13](#_Toc82761277)

[7 References 14](#_Toc82761278)

[8 Appendices 15](#_Toc82761279)

Revision History

| Version | Date | Created by | Comments |
| --- | --- | --- | --- |
| **0.01** |  | Geoffrey Roberts |  |
|  |  |  |  |

Attachments

| Reference | Version/Date | Comments/Notes |
| --- | --- | --- |
|  |  |  |
|  |  |  |

# Introduction

## Project Executive Summary

The objective of the ‘Modern Collection of International Education Data (PRISMS)’ project is to streamline the collection of data used by Providers when using the PRISMS system to comply with their legislative requirements.

At present, Providers that use a Student Admission System or Student Management System, must duplicate (literally copy and paste) information from their AMS/SMS into PRISMS to register a student.

The objective of this project is to create APIs that will allow the transfer of information from an AMS/SMS into PRISMS without the user needing to manually copy or duplicate that data. These APIs will increase the efficiency of Providers using PRISMS and reduce human error benefiting the sector as a whole.

This design is an output of the ‘Modern Collection of International Education Data (PRISMS)’ project. The purpose of the design is to document decisions with associated reasoning, and in so doing help steer the project to deliver its outcomes in a timely, structured and risk managed fashion.

## Project Background

The international education market is a key sector for the Australian economy. It is Australia’s fourth largest export sector and worth some $37.3 billion in the 2019-2020 financial year (ABS cat 5368.0).

The Education Services for Overseas Students Act 2000 (ESOS Act) establishes the legal framework for the quality assurance of education and training institutions (Providers) offering courses to international students (students studying in Australia on student visas) (REF01).

A key part of the ESOS Act in section 109 is the use of a computer system established by the Secretary that can be used for the ‘purpose of receiving and storing information about accepted students and former accepted students’ (REF**03**, page 117).

The Provider Registration and International Student Management System (PRISMS) was established some 20 years ago for this purpose (REF**04**). Some if its key functions include:

* Use by ESOS Agencies to register Providers and their certified courses.
* Use by Providers and their Agents for the registration of Students planning to undertake a registered course. Provides also make any changes to a Students’ study program that may affect their visa, study periods or courses undertaken.
* Submission to the Department of Home Affairs (DoHA) of a Student registration. The purpose being for DoHA to validate a student visa request against. This uses an identifier called a Confirmation of Enrolment (CoE).

The

## Document Scope

### In Scope

* Requirements to deliver the project outcomes
* Assessment of the current PRISMS architecture. See separate document.
* Software Development life cycle and release cadences.
* Environments required to deliver the project outcomes.
* Gateway and other changes required to delivery project outcomes.
* Methodology for determining API definitions.
* Identification of patterns that may be useful for follow-on projects.
* APIs to meet the initial August 2023 release.
* Note of any discovered architectural debt or debt introduced with the project.
* Impacts of the project on existing PRISMS identity – authentication and authorisation.
* Key decisions required to achieve the project outcomes and time frame.
* APIs out side of those required for Providers.
* *Item 2*

### Out of Scope

* Ongoing support and maintenance.
  + Other support services such as help desks, detailed onboarding specifications and documentation.
* Definitions of APIs post the August 2023 release.
* Any design or strategy for re-architecting PRISMS and its components.
  + Suggestions may be made for an end state, but a follow-on project should address the re-architecture and platforming of PRISMS.
* *Item 2*

## References

|  |  |  |
| --- | --- | --- |
| Ref ID | Document Name | Document location/description |
| REF01 | The ESOS legislative framework (web site at the Department of Education) | [The ESOS legislative framework](https://internationaleducation.gov.au/regulatory-information/Education-Services-for-Overseas-Students-ESOS-Legislative-Framework/ESOS-Regulations/Pages/default.aspx) |
| REF02 | Education Services for Overseas Students Regulations 2019 | [Education Services for Overseas Students Regulations 2019](https://www.legislation.gov.au/Details/F2021C01320) |
| REF03 | Education Services for Overseas Students Act 2000 | [Education Services for Overseas Students Act 2000](https://www.legislation.gov.au/Details/C2022C00066) |
| REF04 | DEWR internal web site October 2021 | [News - The Provider Registration and International Student Management System — is 20 years old this October!](https://dese.hosts.application.enet/News/Pages/The-Provider-Registration-and-International-Student-Management-System-(PRISMS)-is-turning-20-years-old-this-October!.aspx) |
| REF05 | DoE web site – International Education Data and Research | [International Education Data and Research - Department of Education, Australian Government](https://www.education.gov.au/international-education-data-and-research) |
| REF06 | PRISMS current state architecture review | [PRISMS\_CurrentState\_Architecture\_v02.docx](https://sharedservicescentre.sharepoint.com/:w:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Enterprise%20Architect/PRISMS_CurrentState_Architecture_v02.docx?d=w84c49c1676ae4c98ad5ec0b98b5f4931&csf=1&web=1&e=TBc2n9) |
| REF07 | NPP Funding Proposal  NPP-2122-IEDPH1 - DSD-Modern Collect. Internat. Edu Data | NPP document for the project: Modern Collection of International Education Data (PRISMS) |
| REF08 | DB discovery folder: PRISMS database discovery work as part of the modernisation project | [PRISMS database discovery](https://sharedservicescentre.sharepoint.com/:w:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Technical%20Documents/Technical%20Debt%20Discovery/PRISMS%20database%20discovery/Database%20Discovery%20Summary.docx?d=wa011cc10dbae4ccbbd802cd18776fd28&csf=1&web=1&e=Ip2VO7)  [Database Discovery Summary.docx](https://sharedservicescentre.sharepoint.com/:w:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Technical%20Documents/Technical%20Debt%20Discovery/PRISMS%20database%20discovery/Database%20Discovery%20Summary.docx?d=wa011cc10dbae4ccbbd802cd18776fd28&csf=1&web=1&e=fWHTbM) |
|  |  |  |
| REF09 | PRISMS and LTPS dependency discovery | [Discovery of LTPS & PRISMS dependencies.docx](https://sharedservicescentre.sharepoint.com/:w:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Technical%20Documents/Technical%20Debt%20Discovery/Discovery%20of%20PRISMS%20%26%20LTPS%20dependencies/Discovery%20of%20LTPS%20%26%20PRISMS%20dependencies.docx?d=w3d009686a289495fa92344eec979c8a4&csf=1&web=1&e=aLFAQm) |
| REF10 | PRISMSMain database discovery – users and dependencies | [PRISMSMain\_database\_usage\_dependencies.xlsx](https://sharedservicescentre.sharepoint.com/:x:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Technical%20Documents/Technical%20Debt%20Discovery/PRISMS%20database%20discovery/PRISMSMain_database_usage_dependencies.xlsx?d=w3632744509bf47c6a9917485ef273202&csf=1&web=1&e=9QkJEZ) |
| REF11 | ESOS Online Training discovery | [Discovery of PRISMS dependencies for ESOSOnline Training.docx](https://sharedservicescentre.sharepoint.com/:w:/r/sites/DESE-ModernisingPRISMS-TEAM/Shared%20Documents/General/Technical%20Documents/Technical%20Debt%20Discovery/Discovery%20for%20ESOSOnlineTraining/Discovery%20of%20PRISMS%20dependencies%20for%20ESOSOnline%20Training.docx?d=w8004a2398b234a5a8677edf76dc5c67b&csf=1&web=1&e=Vq1mRb) |
| REF12 | Govlink (previously Fedlink) | [Govlink](https://www.finance.gov.au/government/whole-government-information-and-communications-technology-services/govlink) |

## Definitions

| Term/Acronym | Expanded Form | Definition |
| --- | --- | --- |
| DHA | Department of Home Affairs | DHA manages the Visa grant and update process for international students. PRISMS sends Visa grant or update requests to the DHA. The DHA also return Visa grant update statuses to PRISMS. |
| ESOS Act | Education Services for Overseas Students Act 2000 | Act that establishes the legal framework for the quality assurance of education and training institutions (Providers) offering courses to international students (students studying in Australia on student visas). |
| OLTP | Online Transactional Processing [database] | Databases used for running applications. Designed for a larger proportion of updates compared to analytical databases. Operations are design to support concurrent and atomic application transactions. |
| PRISMS | Provider Registration and International Student Management System | System software established as part of the ESOS Act to register Providers, courses and Students. |
| Provider | Education Provider | An educational institution that is providing education courses for Students governed by the ESOS Act. |
| Student |  | From the ESOS Act view, an overseas student that is studying an approved course in Australia on a student visa. |
|  |  |  |
|  |  |  |

## Stakeholders

|  |  |
| --- | --- |
| Team / Person | Interest |
|  |  |
|  |  |
|  |  |

## Document Revisions

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Created By | Comments |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Assumptions

|  |  |
| --- | --- |
| ID | Assumption |
|  |  |
|  |  |

# Overview

## Unique System and Project considerations

The PRISMS system has evolved over some 20 years. It is therefore a mixture of:

* Software architectures
  + Tiers – PRISMS is not strictly a 3-tier system. There is a mix of business logic in the presentation (web), backend (.Net) and database layers. For example, some stored procedures will assemble data ready for presentation in the web browser.
  + It is therefore very difficult to separate out key components and logic for this
* Capabilities
  + There is the ability at present through the administrator interface to adjust reference values. In a self-contained system this can be done atomically, however with external partners consuming APIs this could introduce breaking changes.

## Key decisions

* Develop a parallel API layer. Consequence is a duplication of business logic.
* Could not separate out integrated components
* Where possible develop code libraries that can either be copied/paster directly into code or compiled as ‘static’ libraries.
* Use .Net 6 core ready for web.
* Generations of database access layers
* Existing generation of API code (ESOS Agency APIs).
* Development done via on-premise DevOps instance
* Mix of deployment methodologies for current components.
* Maintain current web interfaces whilst developing new APIs – risk is the modification of data fields out of band of the current data flows and introducing an error.
* Forms (username/password based authentication mechanism).

## Data Holdings and Classification

*Nature of the data the system/solution will hold, its physical location and classification.*

Note – we’ll need to add reference and key comments from PIA.

Have:

* Usernames and passwords.
* Personal Identifiable Information (PII)
* Commercial in confidence information on Providers.
* However, the APIs are not changing the data recorded or stored by PRISMS.
* APIs should not return information that is not already owned or shared by the Provider.

## Objectives

*List the main objectives that the system/solution will deliver*

## Functional Requirements

*Only document unusual functional requirements that may impact development/testing of this solution*

## Non-functional Requirements

*Provide a list of non-functional requirements that will be implemented by this solution. Reference other documents where required (i.e. do not duplicate content)*

## Key system changes

Here we’ll place diagram and notes on:

* APIS
* APIM
* Addition of Automated testing
  + New automated testing framework adopted.
* 2019 server upgrade
  + Move to new DNS names in lower environments – ie, from construction.enet (deprecated in 2017) to devnetwork, testnetwork, testfixnetwork, pprodnetwork
  + Move of current PRISMS apps to 2019 servers – not just APIs
* New XDC staging environment
* New Test Fix (efix environment)
* Identity model – Authentication and Authorisation
* Deployment
  + Previously - TSD did deployments from Pre-Prod up.
  + New – Auto deployment using DSD DevOps up the stack.
* Move from TSD DevOps (on-premise DevOps instance) to DSD DevOps (cloud version hosted in Azure V2) and consolidation of testing and code repositories there.
* Move to GSMA service accounts – would fail though in production as we cross domain boundaries from IDC to EDC.
  + Architectural debt: All lower environments done meaning ironically production is different to all other environments.
* Database access clean-up
  + Separation analysis of TPS, LTPS and PRISMS. Scope was too large to include in the transformation project, so there is technical dept to remediate in upcoming years.

# Architecture – Current State

The following section outlines the key system components in PRISMS. This will include ESOS Training. TPS will only be included to note any coupling with PRISMS databases and business logic or technical debt it introduces into PRISMS.

All production servers and databases are run on premise in the Departments datacentres (DCs):

* Primary Data Centre – Fyshwick
* Secondary Data Centre – Hume

Lower-level environments use the following infrastructure:

* Databases – on-premise.
* Application Servers – AWS.

Please refer to Figure 1 for system and interface references below.

Figure : PRISMS Interfaces (refer REF06)

A diagram of a computer

Description automatically generated

## Overview

The production system comprises the following components.

* Application Web interface and API based services. API numbered references are found in REF06.
  + ESOS Agency APIs (API01) – API layer for ESOS agencies. Developed for the ASQA for managing provider records. At the time of writing, ASQA is not utilising these.
  + SIA Service (API02) – API layer used by the Endeavour Online scholarships application and the Study in Australia web site.
  + PRISMS Receive (API03) – The XML based interface that receives Visa update information from DHA.
  + CRICOS Web APIs (API04) – Both a web site for displaying public CRICOS Provider information and an API interface for access this information.
  + ESOS Training Web service (API05) – API used by ESOS Training to update the PRISMS database when users have completed their training.
* Application Web interfaces. UI numbered references are found in REF06.
  + PRISMS Web (UI01) - The web interface and associated application layer for accessing the PRISMS itself.
  + CRICOS Web (UI02) – The web interface and associated application layer for the public CRICOS web site.
  + ESOS Training (UI04) – Bespoke CRM system developed by the department. It updates PRISMS when users have completed mandatory training before being granted access to PRISMS.
* Ancillary services
  + Autosys – An engine for running batch and scheduled processes.
    - Note that Autosys has a role to interact with the PRISMS database (REF**09**) - ENETAPP\ServiceAutoSys
  + Email – Via the Exchange infrastructure.
* Databases for transactional data stores and online reporting functions.
* Coupled applications that leverage PRISMS or associated databases.
  + LTPS (refer REF**09**)
    - LTPS is not a PRISMS application. It allows the Government to collect levies from education providers to support students being affected by any inability of a provider to offer a course a student was enrolled in.
    - All LTPS ‘real human’ users are created and managed through PRISMS. Authorisation uses the following roles (REF**09**):
      * LTPS-OFFICER, LTPS-ADMINISTRATOR, LTPS-IT-ADMINISTRATOR
    - It users the following PRISMS tables with associated privileges (REF**10**):
      * dbo.refRole (SELECT)
      * dbo.UserActionLog (SELECT | INSERT | UPDATE)
      * dbo.PRISMSUser (SELECT | INSERT | UPDATE)
      * dbo.UserRole (SELECT)
  + ESOS Online Training
    - This is a separate .Net application that houses a Content Management System (CMS) for training new PRISMS users.
    - Upon the completion of training, the PRISMS database is updated via a web service (API05).

## File Shares

* Used or storing CoE certificates
* ????

## Applications

### Autosys

Autosys runs a number of jobs for PRISMS, from the batch sending and receiving of messages from DHA to cleaning up unused user accounts.

The autosys service runs as a user with a number of custom roles in the PRISMSMain databasea:

* ServiceBatchRole - note this role is unused and so could be removed.
* ServicePRISMSRole (INSERT | UPDDATE |EXECUTE |DELETE)
  + See REF**10** for table permissions.
* ServiceTPSRole (INSERT |UPDATE | EXECUTE |DELETE)
  + See REF**10** for table permissions.

### DHA and PRISMS interactions - PRISMSReceive (API03)

Note this is an old interface and at some point should be modernised. See ADBT**03**.

Refer API03 in REF06.

There are two-way communications between PRISMS and the DHA for student enrolment information and subsequent updates to Visa requests. Both communication channels use the PRISMSMessages database to cache messages into batches for processing.

Both communication channels run over Govlink (REF**12**). The source IP addresses for both the communication at both agencies (DEWR and DHA) is registered with the GovLink provider. The DNS for PRISMSRecive is registered with the internal DNS only.

In terms of workflow, International students must apply for student Visas in order to enter Australia for the purpose of study. DHA uses information from PRISMS to make determinations on whether to grant or reject a Visa request from a student. Note that DHA make up a significant cohort of Government users that access the PRIMS application.

PRISMS sends enrolment messages to DHA via an autosys batch job. This batch job runs every 10 minutes. It collects the last 10 minutes worth of updates from the PRISMSMessage database, assembles them in an XML structure and sends them to DHA which is running an API interface running as a Java servlet. This interface is defined and maintained by DHA. The interface is not SOAP, but purely XML structured. There are generally two types of PRISMS actions that will send enrolment updates to DHA – Confirmation of Enrolments (CoEs) and Student Course Variations (SCVs). Note that not all SCV actions will cause updates to DHA.

Once a student has requested and been granted a Visa or Visa modification, DHA sends student Visa grant update messages to PRISMS via the PRISMSReceive API.

The PRISMS Receive API, accepts connections from DHA to receive messages related Student Visa grants. From this service, messages are loaded into the PRISMSMessages database where they are processed daily based. This daily processing is triggered by an autosys schedule.

Note that outgoing messages from PRISMS are sent every 10 minutes based on an autosys batch job. These are not sent via the PRISMS Receive interface. This batch job reads outgoing messages from the PRISMSMessages databases, assembles them and then forwards them to DHA.

This incoming connection uses: ***prismsreceive.education.gov.au***

The web service end point runs as: ***PRISMSTrainingService.asmx***

### PRISMS Web, CRICOS, TPS and LTPS

PRISMS web is the core application interface to PRISMS. It utilises .Net Web Forms and C# for its code base. There is very tight coupling between the interface, and database. There is no separate business logic layer. Business logic is distributed through the user interface, .Net server code and database stored procedures.

The code base is now some 20 years old and has evolved with changes to policy and associated legislation. Additional ‘modules’ have been added – in particular:

* TPS
* LTPS
* CRICOS

CRICOS, TPS and LTPS are separate applications with their own code base. However, they share core business logic and therefore tables within of the PRISMS database. Each needs to interpret CRICOS Provider codes, CRICOS course codes and associated PRISMS data objects.

When developed with changes in legislation, funding was not available to build an API layer between these applications. Instead, an expedient decision was made to duplicated code between applications in the PRSISM ecosystem – in particular PRISMS, TPS and LTPS. This duplication of code also requires CRICOS, TPS and LTPS to directly access the PRISMS database for some functionality. This makes the task of code maintenance more complex, costly and error prone. In the past, this has made it difficult to ensure code consistency when key components in one application are updated and therefore need to be duplicated across the ecosystem. For example, when the hashing algorithm for passwords was upgraded in PRISMS, there was a disruption when this change was migrated to product as the other applications had not been udpated.

## Databases

### Overview

There are 5 main database that are used by the production PRISMS, TPS and ESOS Training applications – refer Table 1.

In addition to the Online Transactional Processing (OLTP) databases used by the PRISMS ecosystem, there are also reporting databases used exclusively by the ‘International Research and Analysis’ section for their report generation – refer Table 2.

Databases in production directly associated with the PRISMS applications are shown in Table 1 below with current sizes as of June 2023. NOTE: PRISMSDocuments does not hold documents for PRISMS. It holds documents for TPS which is built on top of PRISMS.

Table : Key PRISMS ecosystem production OLTP databases

|  |  |  |  |
| --- | --- | --- | --- |
| DB Name | Size (MB) (07/2023) | Key purpose | HA |
| PRISMSMain | 312,103 | main database storing PRISMS data – including user tables shared by TPS. | Y |
| PRISMSHistory | 154,541 | Keeps track of transactional records in specific tables in PRISMSMain such as CoE creation, SCVs, user changes etc. Changes are written to PRISMSHistory from other tables using triggers. | Y |
| PRISMSDocuments | 59,949 | Despite having the name PRISMS, it is used by TPS, not PRISMS | Y |
| PRISMSMessage | 96,142 | Used for batch process scheduling which is triggered by autosys. It is also used for messaging with home affairs.. This batch processing is triggered by Autosys. | Y |
| ESOSOnlineTraining | 597 | Content management database for the ESOS training website. Note this database is not used by the core PRISMS applications. It is used by the ESOS Training web application. | Y |
| ESOSAnalytics | 39,230 | Operational datastore database with transformed / modelled and semi-processed data from PRISMS. The database is usually fed from an autosys job. Note that at the time of writing this autosys job was not functioning and so is manually run from scripts in SQL Management Studio.  Business areas use the semi-processed data via SQL Management Studio or PowerBI for a range of auditing, compliance and reporting purposes. For example, student transfer data is produced by reviewing the student Visa messages received from DHA. These Visa message are cross-checked with the PRISMS enrolment records to identity students that have enrolment records but no corresponding message with DHA.  The results of analysis are exported into XL files for supporting compliance activities.  The International Team (A11 in REF06), ESOS Agencies (ASQA and TESQA) and DHA receive reports generated from this database. | Y |

Table : Analytical databases used by the International Research and Analysis Team

| MIS Database Name | Size  (07/2023) | Description | HA |
| --- | --- | --- | --- |
| PrismsMIS | 411,192 | Copy of PRISMSMain – PowerBI, scripts and SQL scripts are used by the ‘International Research and Analysis’ section for a range of reporting requirements. | N |
| PrismsMIS\_Reference | 17,922 | PowerBI, scripts and SQL scripts used by the ‘International Team’ | N |

The relationships between the OLTP and analytical databases is shown in Figure 1 below.

Figure : PRISMS Database relationship

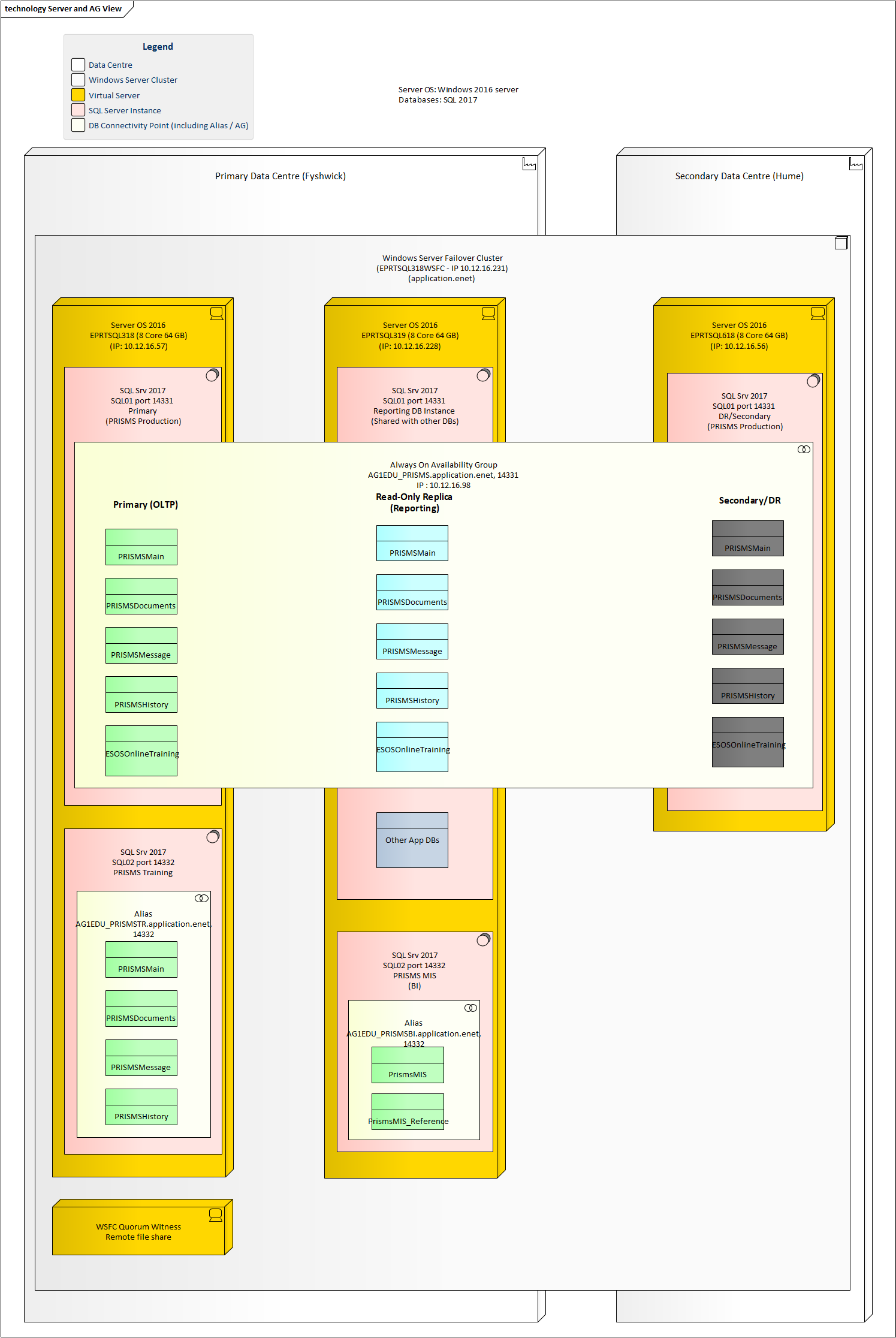
A diagram of a computer

Description automatically generated

### Hosting Arrangement

Figure 1 illustrates the current hosting arrangements for the databases in section 3.2.1 (Overview) above.

Figure : Database instance and availability architecture



## Application

# Architecture – API Interim State

*This section describes the ‘end state’ of PRISMS – that is, after the Mondernisation project has added the APIs.*

## API model

The APIs for Providers have the following characteristics. Note, some of the criteria are expected criteria. In the initial implementation, the architecture and requirements have been kept to a minimum for the initial release:

* Will need to have a central and consistent authentication and authorisation model.
  + Authorisation claims should align to existing roles and associated access restrictions in the current PRISMS application.
  + The API authentication and authorisation model should not cause breaking changes with the current PRISMS UI – this is to ensure the current UI can continue to be used.
* Will require some form of gateway or reverse proxy to:
  + Load balance.
  + Provide redundancy.
  + Allow for traffic restrictions – for example rate limits and throttling.
  + Layer of security
  + Server re-direction based on URL path and DNS names.
* Will require auditing and reporting on usage.
* Enable certificate validation and termination for TLS connections.
* Can be configurable by code – for example when a new API is released or a new API definition is created.

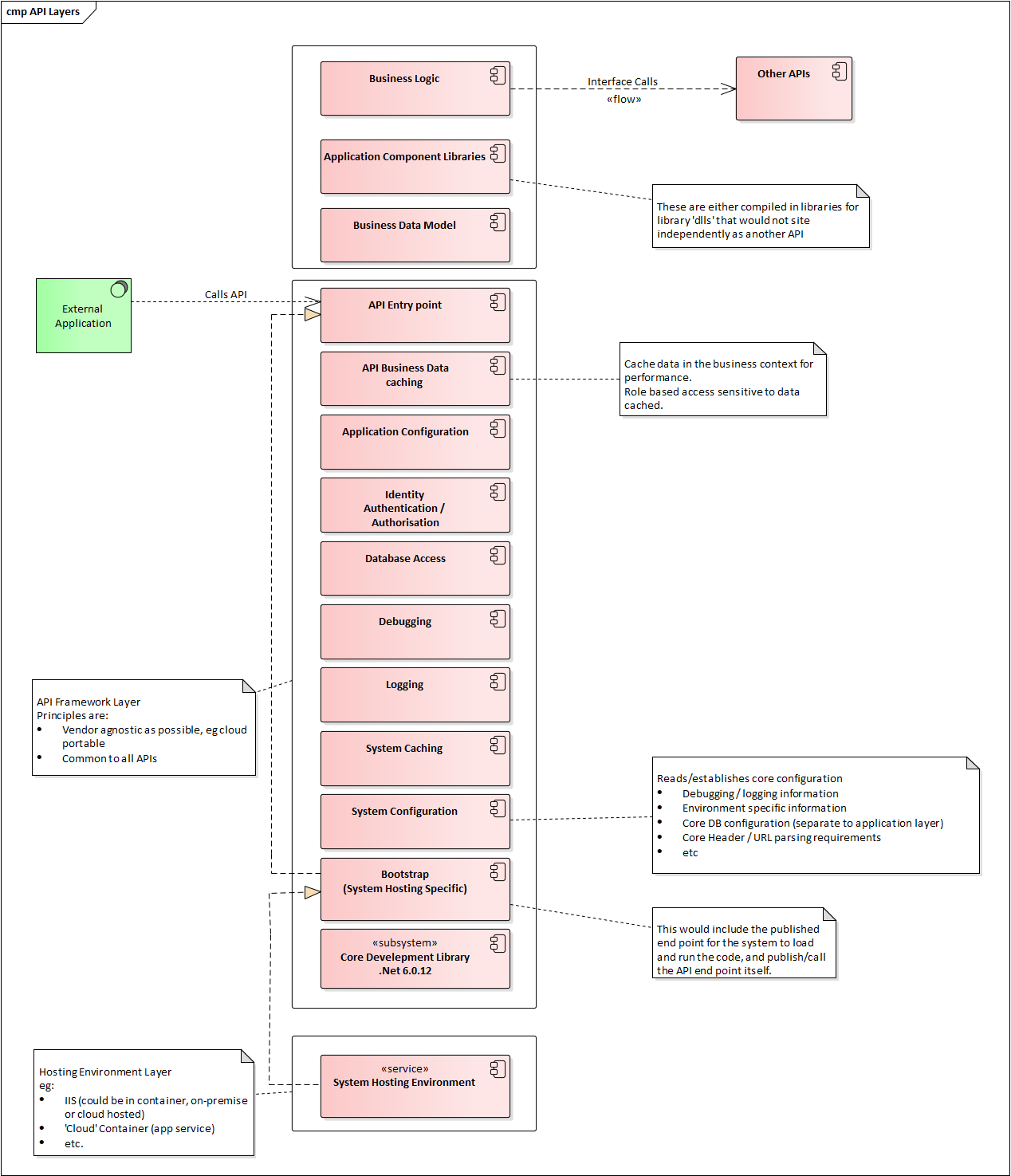
At the time of project initiation, a tender process was underway to procure an API Gateway solution. However, this tender process and successful candidate selection would not have been completed in time to meet the delivery dates of this project.

A decision was made and therefore agreed that this project would use Microsoft Azure’s APIM as the API gateway solution.

Two decision were therefore required on an API Gateway:

* Whether the project needed an API Gateway or whether a simpler reverse proxy or other existing capability could be used.
* If an API Gateway was required, which solution should be used.

|  |  |  |  |
| --- | --- | --- | --- |
| Decision ID | Decision Required | Decision Outcome | Comments |
| AD01 | Is an API Gateway required for the type of APIs to be implemented as part of the PRISMS Modernisation project? | Yes | The scope, level and nature of the APIs, their governance, development life cycle and consumer cohort will require an API Gateway |
| AD02 | With **AD**01, an appropriate API Gateway will need to be selected. | Azure APIM was selected as the appropriate API Gateway solution. | Azure APIM was selected as:   * The project can leverage existing skills and experience the Department has from other projects such as ESS and ADMS. * The Department’s procurement process for an enterprise solution will not be finalised and the solution implemented within the time frame of this project. * A decision will need to be agreed that allows for the use of APIM as an interim strategy. |
|  |  |  |  |



## Environments

### Environment Changes

Changes:

* Adding 2 new environments:
  + Staging in XDC (what is the purpose of this) – for Vendors to test and validate their software against and integration environment.
  + Test Fix in Construction (in the lower environments – mention the purpose of thi)
* Updating servers to 2019 from lower environments up to production (completed as at 25/9/2023)
* Adding 2 APIMs in v2
  + Non Production
  + Production

Current state environment view

Figure : Current state environment and server view

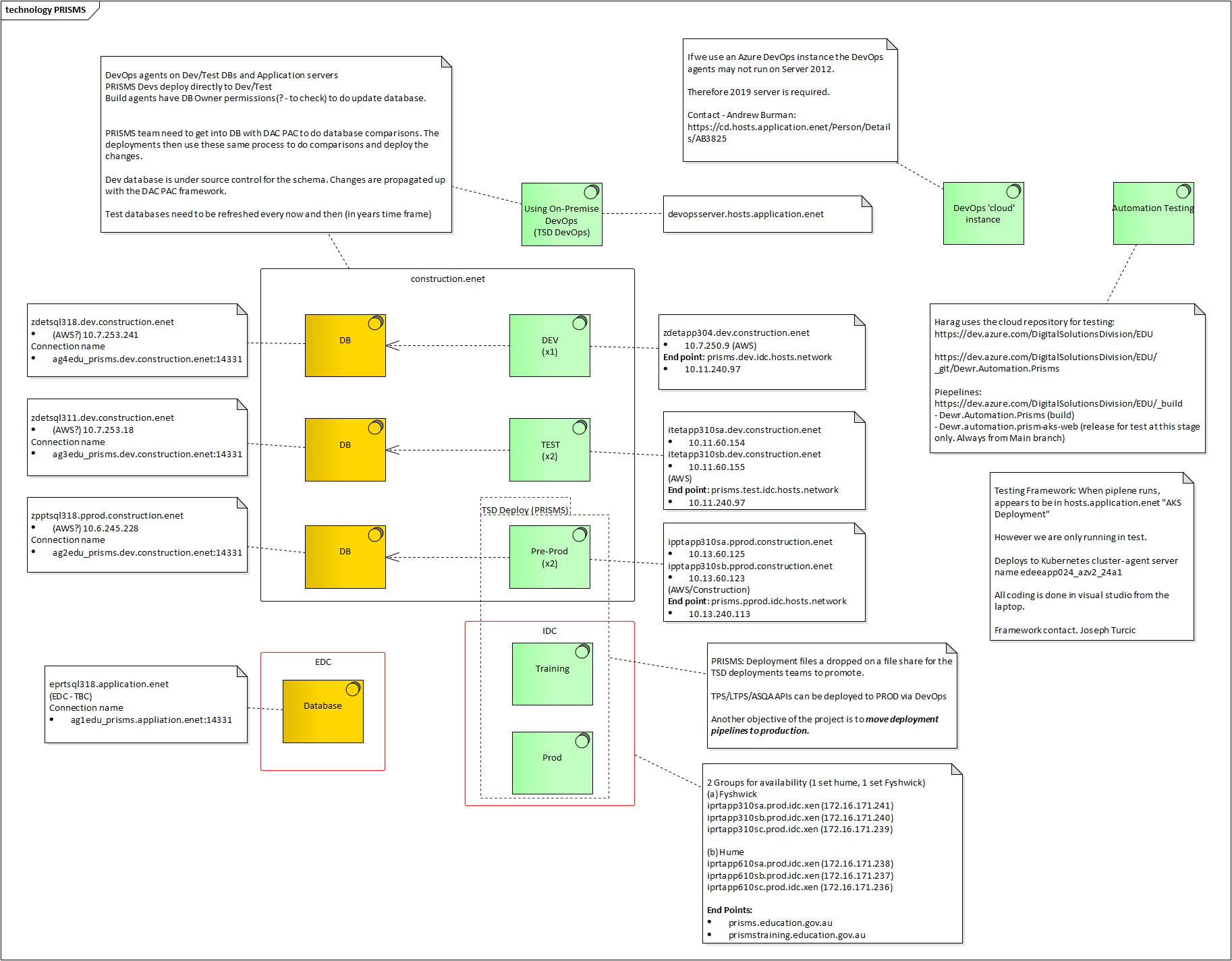


Figure : Current state TSD DevOps deployment view (obsolete with modernisation project)

A diagram of a computer

Description automatically generated

### API Modernisation State View

View of PRISMS after modernisation.

A screenshot of a computer

Description automatically generated

### Staging Environment

### Test Fix

## Application

### Target State Architecture

* *Define the expected target state of the system/solution by provide adequate information and diagrams depicting hosting environments.*
* *Proposed Logical Data Model*
* *Expected Data Volumes*

### System Integration

*List integrations with other systems and the nature of these integrations*

### User Experience

*Document the expected user experience through sample screenshots/wireframes*

### Component Overview

*List the various components that collectively form the system/solution and provide a short description of the role of each component.*

*Example:*

*Network Security Group (NSG)*

*The NSG will provide additional network filtering capability to the solution to control traffic between subnets. It will contain multiple inbound and outbound security rules to and from resources by source and destination IP address, port and protocol.*

## Security

### Identity and Access Management

PRISMS and associated applications use a ‘forms-based’ mechanism for authentication. In this pattern the identity of a user which consists of their username (in the form of an email address) and associated hashed passwords are stored in an on-premise database. The roles of an authenticated user determine which application and types of authorisations that user has within the application being accessed.

The PRISMS database contains the authentication (username and password) for PRISMS, TPS and LTPS. It also holds the roles for PRISMS, TPS and LTPS.

The exception is student identities in TPS. TPS stores student form-based identity information (username and password) in the TPS database.

Note, this means that Departmental users have an identity that is not linked to their Domain user credentials in NATION.

A summary of each system and the storage of a user’s identity is given in Table 1 below.

Table : User Identity in the PRISMS Ecosystem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Application | Users | Credential store | Role definition store | Comments |
| PRISMS | Departmental users  OGA users  Provider users  Agents | PRISMS | PRISMS | [prisms.education.gov.au](https://prisms.education.gov.au/Logon/Logon.aspx) |
| TPS | Departmental users  Provider Users | PRISMS | PRISMS | Departmental users and Providers login using the Provider form to authenticate:  [tps.gov.au](https://tps.gov.au/Home/Login) |
| Students | TPS | TPS | Students use the student form to authenticate.  [tps.gov.au](https://tps.gov.au/Home/Login) |
| LTPS | Departmental users | PRISMS | PRISMS User tables | <https://levies.tps.gov.au/> |

Need to document new Identity architecture with Amit and copy to SOD.

### Network Security

*Provide how network security has been incorporated into the overall solution design (Example: Using VNet integration or private end points etc)*

### Data Protection & Governance

*Describe how data is protected in transit, at rest and how integrity is managed*

### Audit & Logging

*Describe how the system/solution captures application and audit logs.*

## Operations

### Availability

*What are the business expectations of the system/solution in terms of availability (i.e. Expected uptime/Expected downtime for maintenance etc)?*

### Scalability

*Describe how scalability will be achieved*

### Backup Strategy

*Describe the backup strategy for the system/solution. Also include any retention and archival strategies.*

### Disaster Recovery

*Provide technical & procedural detail related to recovery of data in a disaster scenario*

### Consumption & Performance Monitoring

*Provide details about the expected peak consumption (if applicable) and how performance of the system will be monitored*

### License Management

*Provide licensing details related to the system/solution (if any)*

### Support

*Provide details of teams responsible for supporting the system/solution*

# Development Approach

Security notes

* dyantrace in production
* tennable - Dev to Prod - Kate Lee
* Brady Slaughter - use nessus in production - can be done in lower environments.
* DevOps - integrate OWASP checks as part of CI
* - OWASP Zap.
* Secure development process workshop - from DTA.

## Current State vs Future State

### Current State

At present the core tool used for software versioning and release cycle management is the so-called TSD DevOps Server. This is an on-premise Microsoft solution.

Azure DevOps Services

## Environments

*Describe environment segregation strategy and where they are hosted*

## Tools & Libraries

*List all tools & libraries used by the solution*

## Practices

*List practices and design patterns utilised in the development of the solution*

## Release Management

*Provide end-to-end steps for managing the release of the system/solution (i.e. source code management, CI/CD etc)*

# Risk Management

*List real and perceived risks and steps taken to mitigate them*

# References

# Appendices

## Architectural Debt

NOTE: The PRISMS modernisation project is increasing the architectural debt of the system until the end state for the APIs is reached.

|  |  |  |
| --- | --- | --- |
| Ref ID | Debt Summary | Addressing |
| ADBT01 | Current APIs (ESOS Agency, CRICOS Web and others) essentially use forms based authentication. In this pattern, the username and password for access is either maintained within the PRISMS database or are configured as parameters in the code directly. | These APIs should be migrated to the new Identity solution design using OAuth and OIDC patterns for attended or unattended workflows as appropriate.  Each API should be assessed so the appropriate pattern and associated identity artefacts can be created. Once assessed, a migration plan should be developed.  Public facing APIs should be published and re-routed through APIM. |
| ADBT02 | PRISMS User authentication is currently in the PRISMS database (forms-based authentication). This interim state where users continue to authentication will be in place after the release of the APIs. | Planning should be undertaken to migrate users to the end state identity solution. This will move the registration of the user and password out of the PRISMS DB and to an industry based identity management solution.  These solutions are MyGovID for Australian based users and B2C for other users that are not able to use MyGovID (eg overseas agents).  NOTE: This will require an re-write of login code and logic for the current applications leveraging the PRISMS database for authentication - PRISMS, TPS etc. |
| ADBT03 | The PRISMS messaging interface with DHA is old and uses outdated security controls. It relies on restricted access to Govlink and the restricted publishing of DNS records. | The two-way communications between PRISMS and DHA have not been modernised for a number of years. Message are packaged into a simple XML structure and parsed at each end. Security relies on the protection of GovLink, restricted publishing of DNS records and the registration of source IP addresses. DHA uses an old Java servlet and DEWR uses old .Net code. Original developers of the solution have since left the department and there is no incentive to maintain the code base.  Upgrading this communication channel would need to be a joint program between DEWR and DHA. It should be modernised to use modern authentication and REST based patterns. |
| ADBT04 | The PRISMS ecosystem of applications – PRISMS, TPS and LTPS share code libraries. Code is duplicated between applications making the maintenance more complex, costly and error prone. | The ecosystem of applications should be moved to an API layer for common code elements. TPS and LTPS should be become independent applications that consume APIs for common functionality.  The foundation of the this API layer is being built as part of the PRISMS modernisation project. Different applications in the ecosystem should be modified to consume common APIs rather than use duplicate code. |
| ADBT05 | TPS, LTPS, CRICOS and PRISMS share tables within the PRISMS database. | Database tables unique to each application should be separated out from the PRISMS database. APIs should be used to consume common services rather than directly accessing the corresponding database tables. |
| ADBT06 | TPS, LTPS directly access the user, password and role tables in PRISMS. | TPS and LTPS should be updated to use the new identity solution developed for the DoE as part of the PRISMS modernisation project. |