

# Prudential Life Assurance

# Migration Discovery Report

## Migration Discovery (DRAFT)

Date:

Authors:

Prepared for: PT. Prudential Life Assurance

Document type:

# Contents

<b>1. Introduction</b>	<b>3</b>
1.1 Success criteria	3
1.2 Engagement summary	3
<b>2. Summary of outcomes</b>	<b>4</b>
<b>3. Foundational architecture</b>	<b>4</b>
<b>4. Prerequisites and requirements</b>	<b>4</b>
4.1 Business	4
4.2 Technical	4
4.3 Security	5
<b>5. Discovery results</b>	<b>5</b>
5.1 Summary	5
5.2 PMN Portal	5
5.2.1 Business case	6
5.2.2 Technical overview	6
5.2.3 Servers	6
5.2.4 Services	7
5.2.6 Dependencies	8
5.3 PMN PRUHub	8
5.3.1 Business case	8
5.3.2 Technical overview	8
5.3.3 Servers	9
5.3.4 Services	9
5.3.5 Dependencies	10
5.4 Customer Universe	10
5.4.1 Business case	10
5.4.2 Technical overview	10
5.4.3 Servers	10
5.4.4 Services	11
5.4.5 Dependencies	11
<b>6. Proposed Architecture</b>	<b>12</b>
6.1 High Level Architecture	12
6.2 PMN Portal Architecture	12
6.2.1 PMN Portal Load Balancers	12
6.2.2 PMN Portal Frontend	13
6.2.3 PMN Portal Backend	13

6.2.4 PMN Portal Database	13
6.3 PMN PRUHub Architecture	14
6.3.1 PMN PRUHub Backend	14
6.3.2 PMN PRUHub PostgreSQL Database	14
6.4 Customer Universe Data Lake	14
6.4.1 Customer Universe Consumer API Backend	14
6.4.2 Customer Universe Database	14
6.5 Monitoring, Logging and Backup	14
<b>7. Move groups and migration strategy</b>	<b>14</b>
6.1 Migration timeline	14
6.2 PMN Portal Migration	15
6.3 PMN PRUhub Migration	15
6.4 Customer Universe Data Lake Migration	15
6.5 Risks and issues	16
<b>10. Engagement data</b>	<b>16</b>
10.1 Server List	16
10.2 Application List	16

# 1. Introduction

PT Prudential Life Assurance is embarking on a project to migrate all applications in their on-premises data centers to Google Cloud Platform (GCP).

## 1.1 Success criteria

PT. Prudential Life Assur expressed the following as expectations, success criteria, and objectives for the engagement:

- Use managed solutions as much as possible.
- Data locality to comply with regulations.
- Adhere to Prudential's security and compliance directives.
- Planned system reliability and performance should match current capabilities; ideally better.
- Cutover related downtime should be minimal
- Ensure a secure and scalable environment following best practices
- Ensure successful migration of Customer Universe and PMN application groups.

Based on discussions between Aliz and PT Prudential Life Assurance, the following goals were determined to be out of scope for this engagement:

- Performing the application migration from data center(s) to GCP.
- 

## 1.2 Engagement summary

# 2. Summary of outcomes

## 3. Foundational architecture

This project does not cover foundational architecture since Google Cloud Platform foundational architecture is managed by **regional team**. Aliz will follow standards and best practices defined by **regional team**.

## 4. Prerequisites and requirements

To ensure a successful engagement, the following prerequisites and requirements to the migration were gathered. These should be kept in mind when moving into planning and active migration.

### 4.1 Business

- **Maximum downtime window is 6 hours (23:00 - 05:00 Jakarta time)**

### 4.2 Technical

- Application deployment in the cloud should reduce the downtime period. Assuming the application support the deployment method.
- Application and database deployment should have high availability (HA) setup.

### 4.3 Security

- Data at-rest should be encrypted.
- Data in-transit from users to load balancers should be encrypted.
- Data in-transit to database server should be encrypted.

# 5. Discovery results

This section describes the findings that were aggregated by Aliz from the data collected from questionnaire surveys, and interviews with application owners.

## 5.1 Summary

The following table presents the list of applications which were analyzed for this migration:

Application Name	Description	Business Criticality
PMN Portal	PMN Portal is partner facing web application used by hospitals and clinics to manage insurance claim to Prudential	P1
PMN PRUhub	PMN PRUhub is back-office system which acts as bridge between PMN and Life Asia (Prudential core systems)	P1
Customer Universe Data Lake	Customer Universe Data Lake ingest data from core system (Life Asia) and being used by system like PMN Portal.	P1

The following applications were excluded from the scope of this discovery:

- [List any applications that were excluded from discovery, usually for technically prohibitive or security reasons. These will not be listed in detail or in move groups.]

The following applications have been identified as decommissioning candidates:

//TODO

## 5.2 PMN Portal

PMN Portal is a partner-facing application that is being used by hospital and clinic staff to create insurance claims for patients admitted to their facilities.

### 5.2.1 Business case

- Critical part of insurance claim in Prudential systems. Hospitals and clinics rely on PMN portal to process insurance claim.

### 5.2.2 Technical overview

- This application consists of a frontend, XX microservices on the backend and one Couchbase database.
- The frontend application built using react native and deployed in an Nginx server.

- The backend applications are written in Java. Currently deployed on multiple platforms, some applications run as docker container, some of them run on top of Apache Flink. Application is currently being refactored to use Quarkus framework with the end goal all applications can be run as container on top of GKE.
- The database server is a Couchbase 6.5.0 cluster running on 7 servers (3 query nodes, 2 index nodes, 2 data nodes). This cluster host one bucket to be migrated. The cluster is planned to be migrated to run on GKE using Couchbase operator.
- PMN Portal also access data from Customer Universe Data Lake. At the time of discovery session there are two methods to access this data, directly to the database (deprecated method) and through customer universe API.

### 5.2.3 Servers

The following table list servers currently used by PMN Portal application.

VM Name	Description	vCPU	RAM (GB)	Operating Systems	Total Storage (GB)
VIDDCLXPPULAP01	PMN App - Docker Compose	8	32	RHEL 7 (64-bit)	142
VIDDCLXPPMND01	PMN Couchbase DB1	8	16	RHEL 7 (64-bit)	75
VIDDCLXPPMND02	PMN Couchbase DB2	8	16	RHEL 7 (64-bit)	75
VIDDCLXPPMND03	PMN Couchbase DB3	8	16	RHEL 7 (64-bit)	70
VIDDCLXPPMND04	PMN Couchbase DB4	8	16	RHEL 7 (64-bit)	70
VIDDCLXPPMND05	PMN Couchbase DB5	12	16	RHEL 7 (64-bit)	65
VIDDCLXPPMND06	PMN Couchbase DB6	12	16	RHEL 7 (64-bit)	65
VIDDCLXPPMND07	PMN Couchbase DB7	12	16	RHEL 7 (64-bit)	65
VIDDCLXPPMNAP03	PMN nginx proxy 1	4	4	RHEL 7 (64-bit)	60
VIDDCLXPPMNAP04	PMN nginx proxy 2	4	4	RHEL 7 (64-bit)	60
VIDDCLXPPULAP07	Pulse & PMN - EWS Nginx 1	4	8	RHEL 7 (64-bit)	118
VIDDCLXPPULAP08	Pulse & PMN - EWS Nginx 2	4	8	RHEL 7 (64-bit)	236

### 5.2.4 Services

// todo: add priority

service	type	techstack	min-memor y	min-cpu	remarks
couchbase (prutopia)	database	couchbase	N/A	N/A	should be equal to production
qrks-couchbase-wrapper	api service	java	1,5Gi	100Mi	
data-prutopia-interface	api service	java	1,5Gi	100Mi	
data-policy-inquiry-interfa	api service	java	1,5Gi	100Mi	

ce					
data-interface	api service	java	1,5Gi	100Mi	
zookeeper	message queue	kafka	1Gi	250Mi	will be adjusted later
kafka	message queue	kafka	1Gi	250Mi	will be adjusted later
kafka-interface	api service	java	2Gi	250Mi	
qrks-document-broadcast	api service	java	1,5Gi	100Mi	
qrks-batch-cm	scheduler service	java	1,5Gi	100Mi	
qrks-batch-hpx	scheduler service	java	1,5Gi	100Mi	
qrks-document-download	api service	java	1,5Gi	100Mi	
qrks-document-upload	api service	java	1,5Gi	100Mi	
qrks-edoc-wrapper	api service	java	1,5Gi	100Mi	
qrks-document-generate	api service	java	1,5Gi	150Mi	
portal-notification	api service	java	1,5Gi	100Mi	
pulse-data	api service	java	1Gi	150Mi	
pmn-scheduler	scheduler service	java	1,5Gi	150Mi	
qrks-service-document-exit	api service	java	1,5Gi	100Mi	
qrks-service-user-mgmt	api service	java	1,5Gi	100Mi	
qrks-service-prebooking	api service	java	1,5Gi	100Mi	
qrks-service-flash-message	api service	java	1,5Gi	100Mi	
qrks-service-mailjob	api service	java	1,5Gi	100Mi	
qrks-service-policy	api service	java	1,5Gi	100Mi	
qrks-service-case-mgmt	api service	java	1,5Gi	100Mi	
qrks-service-case-discharge	api service	java	1,5Gi	100Mi	
qrks-service-case-monitoring	api service	java	1,5Gi	100Mi	
qrks-service-daily-monitoring	api service	java	1,5Gi	100Mi	
qrks-service-claim-settlement	api service	java	1,5Gi	100Mi	
qrks-service-integration	api service	java	1,5Gi	100Mi	
qrks-service-admission	api service	java	1,5Gi	100Mi	
pmn-hospital-portal	frontend	react native	4Gi	250Mi	

## 5.2.6 Dependencies

- Imperva Incapsula is running in front of PMN portal as WAF
- Several services in PMN PRUHub
- Customer Universe Consumer API
- Customer Universe Database (Prucare)
- IBM Content Manager
- HPX
- Email and SMS services (maintained by another tribe)

## 5.3 PMN PRUHub

### 5.3.1 Business case

PRUHub is Prudential back-office application used by multiple tribes. PMN PRUHub is part of the PRUhub systems which used by PMN to bridge between PMN and Prudential core systems (Life Asia)

**PMN PRUhub** is a web-based application which is purposely to support the business users to process the incoming cashless claim from the Hospital submission stage until the claim approve with excess or without excess then for the final stage for Claim Settlement from the Hospital. PMN PRUhub is developed as the bridge between the business users and the Life Asia, as the core system, which deliver more values for business users and overcome the limitation of Life Asia.

### 5.3.2 Technical overview

PMN PRUHub consist of XX microservices backend applications written in Java running on top of JBoss Application Server. Some of these microservices connected to a shared PostgreSQL server (used by multiple tribes, not only PMN).

PMN PRUhub frontend is using shared PRUhub frontend which currently being used by multiple tribes. The frontend will stay on the on-premise data center for the time being and not migrated to Google Cloud.

### 5.3.3 Servers

The following table contain list of server currently being used by PMN PRUHub in production.

VM Name	Description	vCPU	RAM (GB)	Operating Systems	Total Storage (GB)
VIDDCLXPABDB 02	DB Workflow	8	44	RHEL 7 (64-bit)	1064
VIDDRLXUMSCAP 04	Microservice app 4 PAA2	4	64	RHEL 7 (64-bit)	334
VIDDRLXUMSCAP 03	Microservice app 3 PAA2	4	64	RHEL 7 (64-bit)	334

Note:

- VIDDCLXPAOBDB02 will be partially migrated since this DB is shared by multiple tribes. Only data belong to PMN PRUHub will be migrated to Google Cloud SQL for PostgreSQL.

### 5.3.4 Services

service	type	techstack	min-memory	min-cpu	remarks
pmn-bpm-service	api service	java	750Mi	250m	
pmn-transaction	api service	java	750Mi	250m	
pmn-ui	api service	java	750Mi	250m	
pmn-brms	api service	java	750Mi	250m	
pmn-masterbrms	api service	java	750Mi	250m	
pmn-scheduller	api service	java	750Mi	500m	Sharing Folder/ Mounting
pmn-case-la-interface	api service	java	750Mi	500m	Sharing Folder/ Mounting
pmn-claimanalyst	api service	java	750Mi	250m	
pmn-claimanalyst-pss	api service	java	750Mi	250m	
pmn-hacalculation	api service	java	750Mi	250m	
pmn-hapluscalculation	api service	java	750Mi	250m	
pmn-hscalcalculation	api service	java	750Mi	250m	
pmn-claimhistory	api service	java	750Mi	250m	Database Onprem
pmn-benefitcode	api service	java	750Mi	250m	Database Onprem
pmn-master-benefit	api service	java	750Mi	250m	Database Onprem (CDSWF)
pruhub-pmn-datamaster	api service	java	750Mi	250m	Database Onprem
pruhub-pmn-data-engine	api service	java	750Mi	250m	Database Onprem (TBC)
pmn-mastertransaction	api service	java	750Mi	250m	
pmn-baw-services	api service	java	750Mi	250m	TBC
pmn-baw-log-services	api service	java	750Mi	250m	TBC
pruhub-pmn-cm-interface	api service	java	750Mi	250m	DB2 (TBC)
PostgreSQL Database	database	PostgreSQL			

### 5.3.5 Dependencies

PMN PRUhub depends on several services currently available on-premises:

- IBM BPM
- IBM BRMS
- Windows file sharing (used by scheduler to store reports)
- IBM Content Manager

## 5.4 Customer Universe

### 5.4.1 Business case

Customer universe Data Lake contain data aggregated from multiple core system data source. This data will be used by systems like PMN portal.

### 5.4.2 Technical overview

- Customer Universe data lake consumer API consist of three services (keycloak, krakend, and consumer API service)
- There are CDC systems using debezium and attunity which will write the data change to a Kafka cluster. The data in Kafka will be read by a data transformation service that will write the data to a Couchbase cluster (Prucare).

### 5.4.3 Servers

The following table contain list of servers currently being used by Customer Universe Data lake in production environment.

VM Name	Description	vCPU	RAM (GB)	Operating Systems	Total Storage (GB)
VIDDCLXPCUVAP01	Customer Universe	4	8 (64-bit)	RHEL 7	160
VIDDCLXPCUVAP02	Customer Universe	4	8 (64-bit)	RHEL 7	2684
VIDDCLXPCUVAP03	Customer Universe	48	50 (64-bit)	RHEL 7	850
VIDDCLXPCUVAP27	Customer Universe	4	8 (64-bit)	RHEL 7	70
VIDDCLXPCUVAP05	Customer Universe - Atlassian Bamboo Remote Agent	8	16 (64-bit)	RHEL 7	9273
VIDDCLXPCUVAP10	Customer Universe - Couchbase Data Node #1	20	96 (64-bit)	RHEL 7	1596
VIDDCLXPCUVAP22	Customer Universe - Couchbase Data Node #2	20	96 (64-bit)	RHEL 7	1596
VIDDCLXPCUVAP23	Customer Universe - Couchbase Data Node #3	20	96 (64-bit)	RHEL 7	1596
VIDDCLXPCUVAP28	Customer Universe - Couchbase Data Node #4	20	100 (64-bit)	RHEL 7	1596
VIDDCLXPCUVAP13	Customer Universe - Couchbase Index Node #1	24	64 (64-bit)	RHEL 7	572
VIDDCLXPCUVAP24	Customer Universe - Couchbase Index Node #2	24	64 (64-bit)	RHEL 7	572

VIDDCLXPCUVAP09	Customer Universe - Couchbase Index Node #3	24	32	RHEL 7 (64-bit)	572
VIDDCLXPCUVAP21	Customer Universe - Couchbase Index Node #4	24	32	RHEL 7 (64-bit)	572
VIDDCLXPCUVAP14	Customer Universe - Couchbase Query Node #1	10	32	RHEL 7 (64-bit)	160
VIDDCLXPCUVAP25	Customer Universe - Couchbase Query Node #2	10	32	RHEL 7 (64-bit)	160
VIDDCLXPCUVAP26	Customer Universe - Couchbase Query Node #3	10	32	RHEL 7 (64-bit)	160
VIDDCLXPCUVAP20	Customer Universe - Flink JM Node #2	6	12	RHEL 7 (64-bit)	110
VIDDCLXPCUVAP15	Customer Universe - Kafka2	8	20	RHEL 7 (64-bit)	120
VIDDCLXPCUVAP29	Customer Universe - Quarkus + API gateway 2	8	20	RHEL 7 (64-bit)	120
VIDDCLXPCUVAP06	Customer Universe - Zookeeper1	6	8	RHEL 7 (64-bit)	160
VIDDCLXPCUVAP11	Customer Universe - Zookeeper2	6	8	RHEL 7 (64-bit)	1207

#### 5.4.4 Services

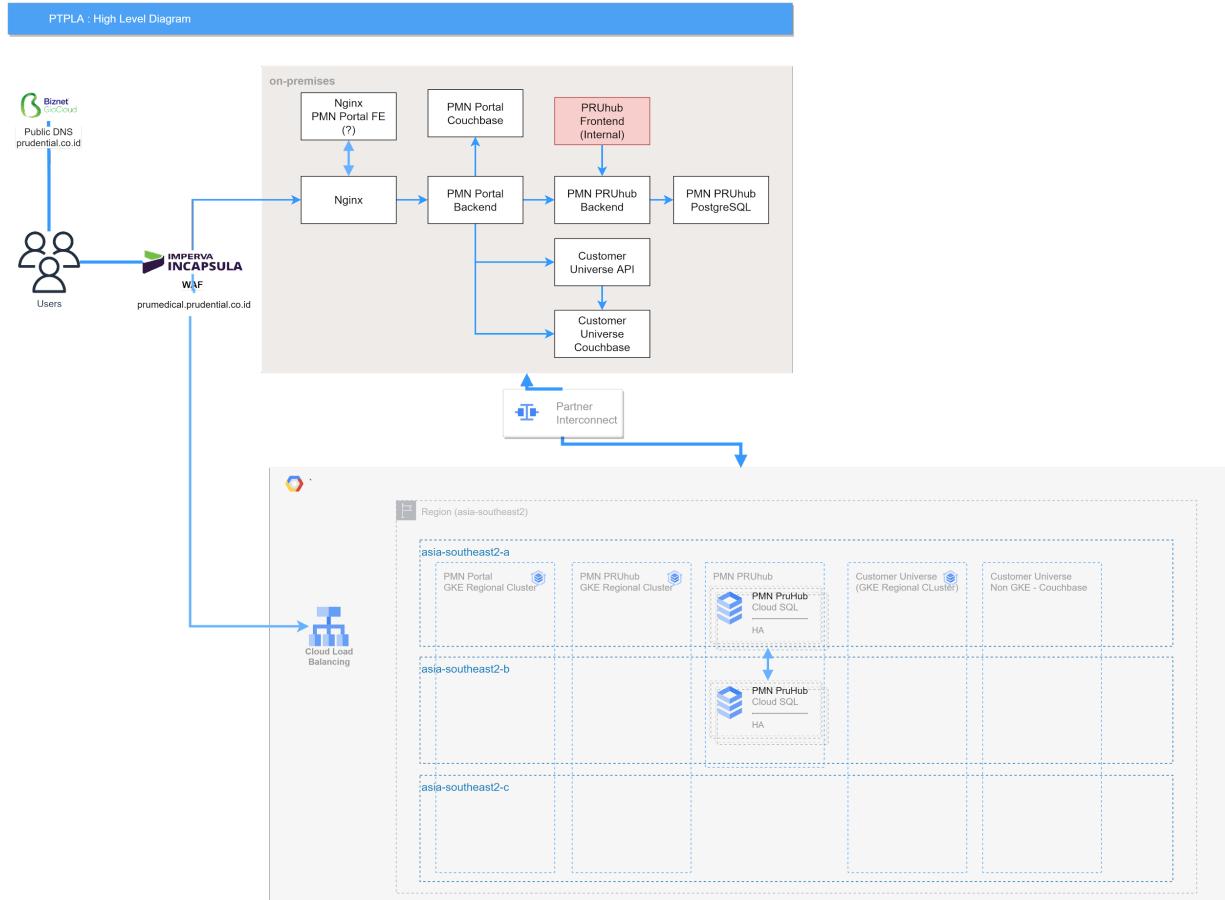
#### 5.4.5 Dependencies

- Life Asia (core systems)
- Data processing system to Couchbase

## 6. Proposed Architecture

### 6.1 High Level Architecture

The following diagram shows the relationship between PMN Portal, PMN PRUhub and Customer Universe resources in Google Cloud Platform and on-premise data center



## 6.2 PMN Portal Architecture

### 6.2.1 PMN Portal Load Balancers

Currently PMN Portal use Nginx as load balancers to API and frontend. We propose replacing the load balancer with Google Cloud Regional External HTTPS Load balancers

From the Nginx configuration provided by PMN team, below are the corresponding features in Cloud Load Balancing for feature / config directive being used in current load balancer.

No	Nginx Features	Cloud Load Balancing
	Keepalive Settings	Not supported
	SSL Ciphers Settings	Might be able to support currently used SSL ciphers or we can use Google Managed SSL Policies ( <a href="https://cloud.google.com/load-balancing/docs/s">https://cloud.google.com/load-balancing/docs/s</a>

		sl-policies-concepts)
	Custom headers	Supported. Since there will be Cloud Armor in this setup. The use of custom header will not add additional costs.
	proxy_hide_server	Not supported. (need to confirm with team whether this is specific for Nginx use case or not)
	Deny access to specific path (path patterns). For example:  location ~ /\.ht { deny all; }	Can use Cloud Armor policies or dummy backend.
	Custom error message	Not supported

### 6.2.2 PMN Portal Frontend

PMN Portal frontend currently deployed on Nginx server. In GKE the frontend will be deployed as an Nginx container running in GKE

### 6.2.3 PMN Portal Backend

Each backend microservice will run as container in GKE. The GKE cluster will be configured so each service will have two or more instances running on different availability zones.

For singleton service (service that cannot have multiple instances running) we will only deploy one instance of the service in GKE.

### 6.2.4 PMN Portal Database

The Couchbase cluster will be deployed on separate GKE cluster. The reason we choose this approach are:

- Maintenance of the control plane is independent of the application GKE cluster. The cluster which contain Couchbase GKE might need to have slower upgrade plan.
- Independent resource sizing policy.
- The cluster configuration might be more strict (or less strict) than the application cluster.

## 6.3 PMN PRUHub Architecture

Two GCP project will be created. The first GCP project will be use to host PMN PRUhub applications running on top of GKE. The second project will be use to host Cloud SQL for PostgreSQL project

### 6.3.1 PMN PRUHub Backend

The Java-based application will be deployed as container in GKE. Multiple pods will be deployed per service to ensure high availability of the service (assuming the service supports multiple instances running simultaneously)

### 6.3.2 PMN PRUHub PostgreSQL Database

- Cloud SQL for PostgreSQL will be deployed in HA setup

[PostgreSQL migration method alternatives](#)

## 6.4 Customer Universe Data Lake

Two GCP projects will be used for Customer Universe data lake. The first project will be used to host GKE cluster to run customer universe consumer API backend. The second project will be used to deploy Couchbase cluster on Google Compute Engine.

### 6.4.1 Customer Universe Consumer API Backend

Customer universe consumer backend will be deployed on GKE. Each service will have multiple pods running on different availability zones to ensure high availability.

### 6.4.2 Customer Universe Database

Customer Universe Couchbase cluster (PRUCARE) will be migrated to Google Compute Engine. [We will follow best practices for deployment of Couchbase in Google Cloud Platform.](#)

## 6.5 Monitoring, Logging and Backup

We will follow the standards on monitoring, logging and backup provided by regional (SRE??) team.

# 7. Move groups and migration strategy

## 6.1 Migration timeline

See migration timeline in this document : [+ Migration Timeline](#)

## 6.2 PMN Portal Migration

The following table contains wave of migration for PMN Portal and list of applications to be migrated per wave.

<b>Wave 1</b>	<ul style="list-style-type: none"><li>Services which have no direct access to database and not public facing API</li></ul>
<b>Wave 2</b>	<ul style="list-style-type: none"><li>Public Facing API (nginx config changes needed)</li></ul>
<b>Wave 3</b>	<ul style="list-style-type: none"><li>Database and Database wrapper (applications which have direct access to database) migration</li></ul>
<b>Wave 4</b>	<ul style="list-style-type: none"><li>Frontend and Load Balancer Migration</li></ul>

### ► **Wave 1**

The following applications will be migrated in Wave 1:

- [List application names here. These are the same as in the chart above.]

## 6.3 PMN PRUhub Migration

The following table contains wave of migration for PMN PRUhub and list of applications to be migrated per wave.

<b>Wave 1</b>	<ul style="list-style-type: none"><li>Application not used by PMN Portal, not database wrapper</li></ul>
<b>Wave 2</b>	<ul style="list-style-type: none"><li>Application used by PMN Portal</li></ul>
<b>Wave 3</b>	<ul style="list-style-type: none"><li>Applications which have dependencies to services on-premises</li></ul>
<b>Wave 4</b>	<ul style="list-style-type: none"><li>Database Migration and database wrapper migration</li></ul>

## 6.4 Customer Universe Data Lake Migration

The following table contains wave of migration for Customer Universe data lake and list of applications to be migrated per wave.

<b>Wave 1</b>	<ul style="list-style-type: none"><li>Deploy all services in GKE</li></ul>
---------------	--

## 6.5 Risks and issues

Various risks or potential blockers have been identified which may slow or block the advancement of the migration phases. These risks should be monitored and mitigation steps identified.

# 10. Engagement data

This section contains engagement data provided by PTPLA teams

## 10.1. Server List

## 10.2 Application List