License Manager Design Document

EOEPCA.SDD.xxx

TVUK System Team

Version 0.1, dd/mm/yyyy:

License Manager Design Document

EO Exploitation Platform Common Architecture
License Manager Design Document
EOEPCA.SDD.xxx

COMMENTS and ISSUES	PDF
If you would like to raise comments or issues on this document, please do so by raising an Issue at the following URL https://github.com/EOEPCA/um-license-manager/issues.	This document is available in PDF format here.
EUROPEAN SPACE AGENCY CONTRACT REPORT The work described in this report was done under ESA contract. Responsibility for the contents resides in the author or organisation that prepared it.	**TELESPAZIO VEGA UK Ltd** 350 Capability Green, Luton, Bedfordshire, LU1 3LU, United Kingdom. Tel: +44 (0)1582 399000 www.telespazio-vega.com

AMENDMENT HISTORY

This document shall be amended by releasing a new edition of the document in its entirety. The Amendment Record Sheet below records the history and issue status of this document.

Table 1. Amendment Record Sheet

ISSUE	DATE	REASON
0.1	dd/mm/yyyy	Initial in-progress draft

Chapter 1. Introduction

1.1. Purpose and Scope

This document presents the License Manager Design for the Common Architecture.

1.2. Structure of the Document

Section 2 - Overview

Provides an overview of the License Manager component, within the context of the wider Common Architecture design.

Section 3 - Design

Provides the design of the License Manager component.

1.3. Reference Documents

The following is a list of Reference Documents with a direct bearing on the content of this document.

Reference	Document Details	Version
[EOEPCA-UC]	EOEPCA - Use Case Analysis EOEPCA.TN.005 https://eoepca.github.io/use-case-analysis	Issue 1.0, 02/08/2019
[EP-FM]	Exploitation Platform - Functional Model, ESA-EOPSDP-TN-17-050	Issue 1.0, 30/11/2017
[TEP-OA]	Thematic Exploitation Platform Open Architecture, EMSS-EOPS-TN-17-002	Issue 1, 12/12/2017
[WPS-T]	OGC Testbed-14: WPS-T Engineering Report, OGC 18-036r1, http://docs.opengeospatial.org/per/18-036r1.html	18-036r1, 07/02/2019
[WPS-REST-JSON]	OGC WPS 2.0 REST/JSON Binding Extension, Draft, OGC 18-062, https://raw.githubusercontent.com/opengeospatial/wps-rest-binding/develop/docs/18-062.pdf	1.0-draft
[CWL]	Common Workflow Language Specifications, https://www.commonwl.org/v1.0/	v1.0.2

Reference	Document Details	Version
[TB13-AP]	OGC Testbed-13, EP Application Package Engineering Report, OGC 17-023, http://docs.opengeospatial.org/per/17-023.html	17-023, 30/01/2018
[TB13-ADES]	OGC Testbed-13, Application Deployment and Execution Service Engineering Report, OGC 17-024, http://docs.opengeospatial.org/per/17-024.html	17-024, 11/01/2018
[TB14-AP]	OGC Testbed-14, Application Package Engineering Report, OGC 18-049r1, http://docs.opengeospatial.org/per/18-049r1.html	18-049r1, 07/02/2019
[TB14-ADES]	OGC Testbed-14, ADES & EMS Results and Best Practices Engineering Report, OGC 18-050r1, http://docs.opengeospatial.org/per/18-050r1.html	18-050r1, 08/02/2019
[OS-GEO-TIME]	OpenSearch GEO: OpenSearch Geo and Time Extensions, OGC 10-032r8, http://www.opengeospatial.org/standards/opensearchgeo	10-032r8, 14/04/2014
[OS-EO]	OpenSearch EO: OGC OpenSearch Extension for Earth Observation, OGC 13-026r9, http://docs.opengeospatial.org/is/13-026r8/13-026r8.html	13-026r9, 16/12/2016
[GEOJSON-LD]	OGC EO Dataset Metadata GeoJSON(-LD) Encoding Standard, OGC 17-003r1/17-084	17-003r1/17-084
[GEOJSON-LD-RESP]	OGC OpenSearch-EO GeoJSON(-LD) Response Encoding Standard, OGC 17-047	17-047
[PCI-DSS]	The Payment Card Industry Data Security Standard, https://www.pcisecuritystandards.org/document_library?category=pcidss&document=pci_dss	v3.2.1
[CEOS-OS-BP]	CEOS OpenSearch Best Practise, http://ceos.org/ourwork/workinggroups/wgiss/access/opensearch/	v1.2, 13/06/2017
[OIDC]	OpenID Connect Core 1.0, https://openid.net/specs/openid-connect-core-1_0.html	v1.0, 08/11/2014

Reference	Document Details	Version
[OGC-CSW]	OGC Catalogue Services 3.0 Specification - HTTP Protocol Binding (Catalogue Services for the Web), OGC 12-176r7, http://docs.opengeospatial.org/is/12-176r7/12-176r7.html	v3.0, 10/06/2016
[OGC-WMS]	OGC Web Map Server Implementation Specification, OGC 06-042, http://portal.opengeospatial.org/files/?artifact_id=14416	v1.3.0, 05/03/2006
[OGC-WMTS]	OGC Web Map Tile Service Implementation Standard, OGC 07-057r7, http://portal.opengeospatial.org/files/?artifact_id=35326	v1.0.0, 06/04/2010
[OGC-WFS]	OGC Web Feature Service 2.0 Interface Standard – With Corrigendum, OGC 09-025r2, http://docs.opengeospatial.org/is/09-025r2/09-025r2.html	v2.0.2, 10/07/2014
[OGC-WCS]	OGC Web Coverage Service (WCS) 2.1 Interface Standard - Core, OGC 17-089r1, http://docs.opengeospatial.org/is/17-089r1/17-089r1.html	v2.1, 16/08/2018
[OGC-WCPS]	Web Coverage Processing Service (WCPS) Language Interface Standard, OGC 08-068r2, http://portal.opengeospatial.org/files/?artifact_id=32319	v1.0.0, 25/03/2009
[AWS-S3]	Amazon Simple Storage Service REST API, https://docs.aws.amazon.com/AmazonS3/latest/API	API Version 2006-03-01

1.4. Terminology

The following terms are used in the Master System Design.

Term	Meaning
Admin	User with administrative capability on the EP
Algorithm	A self-contained set of operations to be performed, typically to achieve a desired data manipulation. The algorithm must be implemented (codified) for deployment and execution on the platform.
Analysis Result	The *Products* produced as output of an *Interactive Application* analysis session.

Term	Meaning
Analytics	A set of activities aimed to discover, interpret and communicate meaningful patters within the data. Analytics considered here are performed manually (or in a semi-automatic way) on-line with the aid of *Interactive Applications*.
Application Artefact	The 'software' component that provides the execution unit of the *Application Package*.
Application Deployment and Execution Service (ADES)	WPS-T (REST/JSON) service that incorporates the Docker execution engine, and is responsible for the execution of the processing service (as a WPS request) within the 'target' Exploitation Platform.
Application Descriptor	A file that provides the metadata part of the *Application Package*. Provides all the metadata required to accommodate the processor within the WPS service and make it available for execution.
Application Package	A platform independent and self-contained representation of a software item, providing executable, metadata and dependencies such that it can be deployed to and executed within an Exploitation Platform. Comprises the *Application Descriptor* and the *Application Artefact*.
Bulk Processing	Execution of a *Processing Service* on large amounts of data specified by AOI and TOI.
Code	The codification of an algorithm performed with a given programming language - compiled to Software or directly executed (interpreted) within the platform.
Compute Platform	The Platform on which execution occurs (this may differ from the Host or Home platform where federated processing is happening)
Consumer	User accessing existing services/products within the EP. Consumers may be scientific/research or commercial, and may or may not be experts of the domain
Data Access Library	An abstraction of the interface to the data layer of the resource tier. The library provides bindings for common languages (including python, Javascript) and presents a common object model to the code.
Development	The act of building new products/services/applications to be exposed within the platform and made available for users to conduct exploitation activities. Development may be performed inside or outside of the platform. If performed outside, an integration activity will be required to accommodate the developed service so that it is exposed within the platform.
Discovery	User finds products/services of interest to them based upon search criteria.
Execution	The act to start a *Processing Service* or an *Interactive Application*.

Term	Meaning
Execution Management Service (EMS)	The EMS is responsible for the orchestration of workflows, including the possibility of steps running on other (remote) platforms, and the on-demand deployment of processors to local/remote ADES as required.
Expert	User developing and integrating added-value to the EP (Scientific Researcher or Service Developer)
Exploitation Tier	The Exploitation Tier represents the end-users who exploit the services of the platform to perform analysis, or using high-level applications built-in on top of the platform's services
External Application	An application or script that is developed and executed outside of the Exploitation Platform, but is able to use the data/services of the EP via a programmatic interface (API).
Guest	An unregistered User or an unauthenticated Consumer with limited access to the EP's services
Home Platform	The Platform on which a User is based or from which an action was initiated by a User
Host Platform	The Platform through which a Resource has been published
Identity Provider (IdP)	The source for validating user identity in a federated identity system, (user authentication as a service).
Interactive Application	A stand-alone application provided within the exploitation platform for on-line hosted processing. Provides an interactive interface through which the user is able to conduct their analysis of the data, producing *Analysis Results* as output. Interactive Applications include at least the following types: console application, web application (rich browser interface), remote desktop to a hosted VM.
Interactive Console Application	A simple *Interactive Application* for analysis in which a console interface to a platform-hosted terminal is provided to the user. The console interface can be provided through the user's browser session or through a remote SSH connection.
Interactive Remote Desktop	An Interactive Application for analysis provided as a remote desktop session to an OS-session (or directly to a 'native' application) on the exploitation platform. The user will have access to a number of applications within the hosted OS. The remote desktop session is provided through the user's web browser.
Interactive Web Application	An Interactive Application for analysis provided as a rich user interface through the user's web browser.
Key-Value Pair	A key-value pair (KVP) is an abstract data type that includes a group of key identifiers and a set of associated values. Key-value pairs are frequently used in lookup tables, hash tables and configuration files.
Kubernetes (K8s)	Container orchestration system for automating application deployment, scaling and management.

Term	Meaning
Login Service	An encapsulation of Authenticated Login provision within the Exploitation Platform context. The Login Service is an OpenID Connect Provider that is used purely for authentication. It acts as a Relying Party in flows with external IdPs to obtain access to the user's identity.
EO Network of Resources	The coordinated collection of European EO resources (platforms, data sources, etc.).
Object Store	A computer data storage architecture that manages data as objects. Each object typically includes the data itself, a variable amount of metadata, and a globally unique identifier.
On-demand Processing Service	A *Processing Service* whose execution is initiated directly by the user on an ad-hoc basis.
Platform (EP)	An on-line collection of products, services and tools for exploitation of EO data
Platform Tier	The Platform Tier represents the Exploitation Platform and the services it offers to end-users
Processing	A set of pre-defined activities that interact to achieve a result. For the exploitation platform, comprises on-line processing to derive data products from input data, conducted by a hosted processing service execution.
Processing Result	The *Products* produced as output of a *Processing Service* execution.
Processing Service	A non-interactive data processing that has a well-defined set of input data types, input parameterisation, producing *Processing Results* with a well-defined output data type.
Products	EO data (commercial and non-commercial) and Value-added products and made available through the EP. *It is assumed that the Hosting Environment for the EP makes available an existing supply of EO Data*
Resource	A entity, such as a Product, Processing Service or Interactive Application, which is of interest to a user, is indexed in a catalogue and can be returned as a single meaningful search result
Resource Tier	The Resource Tier represents the hosting infrastructure and provides the EO data, storage and compute upon which the exploitation platform is deployed
Reusable Research Object	An encapsulation of some research/analysis that describes all aspects required to reproduce the analysis, including data used, processing performed etc.
Scientific Researcher	Expert user with the objective to perform scientific research. Having minimal IT knowledge with no desire to acquire it, they want the effort for the translation of their algorithm into a service/product to be minimised by the platform.

Term	Meaning
Service Developer	Expert user with the objective to provide a performing, stable and reliable service/product. Having deeper IT knowledge or a willingness to acquire it, they require deeper access to the platform IT functionalities for optimisation of their algorithm.
Software	The compilation of code into a binary program to be executed within the platform on-line computing environment.
Systematic Processing Service	A *Processing Service* whose execution is initiated automatically (on behalf of a user), either according to a schedule (routine) or triggered by an event (e.g. arrival of new data).
Terms & Conditions (T&Cs)	The obligations that the user agrees to abide by in regard of usage of products/services of the platform. T&Cs are set by the provider of each product/service.
Transactional Web Processing Service (WPS-T)	Transactional extension to WPS that allows adhoc deployment / undeployment of user-provided processors.
User	An individual using the EP, of any type (Admin/Consumer/Expert/Guest)
Value-added products	Products generated from processing services of the EP (or external processing) and made available through the EP. This includes products uploaded to the EP by users and published for collaborative consumption
Visualisation	To obtain a visual representation of any data/products held within the platform - presented to the user within their web browser session.
Web Coverage Service (WCS)	OGC standard that provides an open specification for sharing raster datasets on the web.
Web Coverage Processing Service (WCPS)	OGC standard that defines a protocol-independent language for the extraction, processing, and analysis of multi-dimentional coverages representing sensor, image, or statistics data.
Web Feature Service (WFS)	OGC standard that makes geographic feature data (vector geospatial datasets) available on the web.
Web Map Service (WMS)	OGC standard that provides a simple HTTP interface for requesting geo-registered map images from one or more distributed geospatial databases.
Web Map Tile Service (WMTS)	OGC standard that provides a simple HTTP interface for requesting map tiles of spatially referenced data using the images with predefined content, extent, and resolution.
Web Processing Services (WPS)	OGC standard that defines how a client can request the execution of a process, and how the output from the process is handled.
Workspace	A user-scoped 'container' in the EP, in which each user maintains their own links to resources (products and services) that have been collected by a user during their usage of the EP. The workspace acts as the hub for a user's exploitation activities within the EP

1.5. Glossary

The following acronyms and abbreviations have been used in this report.

Term	Definition
AAI	Authentication & Authorization Infrastructure
ABAC	Attribute Based Access Control
ADES	Application Deployment and Execution Service
ALFA	Abbreviated Language For Authorization
AOI	Area of Interest
API	Application Programming Interface
CMS	Content Management System
CWL	Common Workflow Language
DAL	Data Access Library
EMS	Execution Management Service
EO	Earth Observation
EP	Exploitation Platform
FUSE	Filesystem in Userspace
GeoXACML	Geo-specific extension to the XACML Policy Language
IAM	Identity and Access Management
IdP	Identity Provider
JSON	JavaScript Object Notation
K8s	Kubernetes
KVP	Key-value Pair
M2M	Machine-to-machine
OGC	Open Geospatial Consortium
PDE	Processor Development Environment
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PIP	Policy Information Point
RBAC	Role Based Access Control
REST	Representational State Transfer
SSH	Secure Shell
TOI	Time of Interest
UMA	User-Managed Access

Term	Definition
VNC	Virtual Network Computing
WCS	Web Coverage Service
WCPS	Web Coverage Processing Service
WFS	Web Feature Service
WMS	Web Map Service
WMTS	Web Map Tile Service
WPS	Web Processing Service
WPS-T	Transactional Web Processing Service
XACML	eXtensible Access Control Markup Language

Chapter 2. Overview

2.1. Building Block Overview

*Content Description*

This section contains:

- High-Level Description of the Building Block

- Context within EOEPCA

The functionality of the License Manager will mainly be as support for the PDP Policy Engine when assessing policy checks that directly
relate to Licenses owned by the Platform or by the End-User requesting access to a specific resource.

- Licence Managing Core, with the capability of recording which Licenses an End-User has available and
whether or not the limits applied to that license have been reached (i.e. if more than 5 users are concurrently using a service with the same License key, a PDP request for Authorization could be
turned down).

- SCIM Client, allowing to retrieve user information that is local to the platform, whenever possible or to update User Profiles based on the
assignment of a specific license

- OIDC Client, allowing to authenticate the component as trusted within the architecture and used by the License Manager to identify itself as trusted party within the EP domain in
order to able to perform queries against the Login Service SCIM endpoints.

- Billing Connector allowing the generation of charging requests to the Billing Service, either timebased or volume-based.

- Management API for License Owners to register Licenses into the system and to assign licenses to the Platform to specific End-Users

2.1.1. Initialization flow

The figure below, identifies the main workflows on which the License Manager participates, along with it's components:
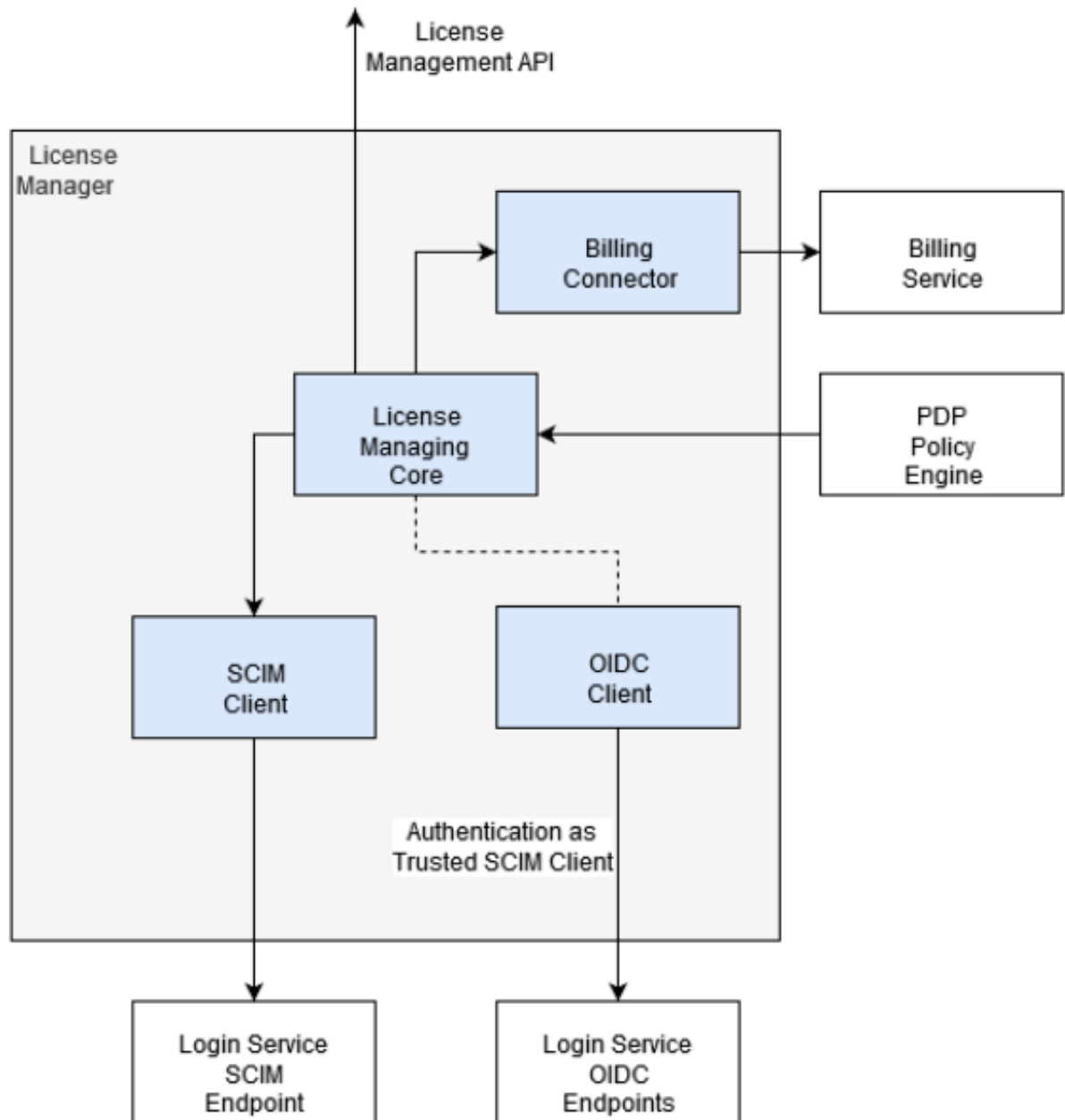
2.1.2. Exposed Interfaces

2.1.2.1. Licenses API

The API will be the platform where the owners will access in order to actually manage the licenses in use. For that an special credentials are needed since not every user has access, same as the login-service.

2.1.2.2. User Profile

In the User Profile web interface each user has access to their own license list stored into the login service user information endpoint. In order to retrieve that values it reuses the SCIM library of EOEPCA.

2.1.3. Consumed Interfaces

2.1.3.1. OIDC (to Login Service)

The PDP uses the OIDC protocol in order to authenticate itself as a valid UMA client, and uses this OIDC client in all UMA-related queries.It allows Clients to verify the identity of the End-User. (https://gluu.org/docs/gluu-server/4.0/admin-guide/openid-connect/)

These queries are done against the Login Service, and the endpoints used are:

- Discovery Endpoint: /.well-known/openid-configuration

And the keys used from Well Known Handler:

- Token Endpoint: KEY_OIDC_TOKEN_ENDPOINT
- UserInfo Endpoint: KEY_OIDC_USERINFO_ENDPOINT

2.1.3.2. SCIM (to Login Service)

The PDP has the capability to auto-register itself as a client if there is no client pre-configured from previous starts or previous configuration. In order to do this, it utilizes the SCIM protocol which is designed to reduce the complexity of user management operations. (https://gluu.org/docs/gluu-server/3.1.1/user-management/scim2/)

The keys used from Well Known Handler:

- User Atributes: KEY_SCIM_USER_ENDPOINT
- Private Key JWT Key: ENDPOINT_AUTH_CLIENT_PRIVATE_KEY_JWT

2.2. Required resources

Content Description

This section contains:

- List of HW and SW required resources for the correct functioning of the building Block
- References to open repositories (when applicable)

2.2.1. Interdependencies

The following list organizes main identified features based on inter-dependencies and impact on the
overall reference implementation functionality:

- UM-LIM-010: SCIM Client connection to License Manager
- UM-LIM-020: OIDC Client connection to License Manager
- UM-LIM-030: License Manage Core (Use Cases)

- UM-LIM-040: Billing Connector

- UM-LIM-050: License Management API

2.2.2. Software

The following Open-Source Software is required to support the deployment and integration of the Policy Enforcement Point:

- EOEPCA's SCIM Client - https://github.com/EOEPCA/um-common-scim-client

- EOEPCA's OpenID - https://github.com/EOEPCA/um-common-oidc-client

- EOEPCA's Well Known Handler - https://github.com/EOEPCA/well-known-handler

2.3. Use cases

2.3.1. Owner Run Operation

In this case the owner interacts with the API in order to operate against the licenses. This secuential diagram shows the different components the Building Block manages.
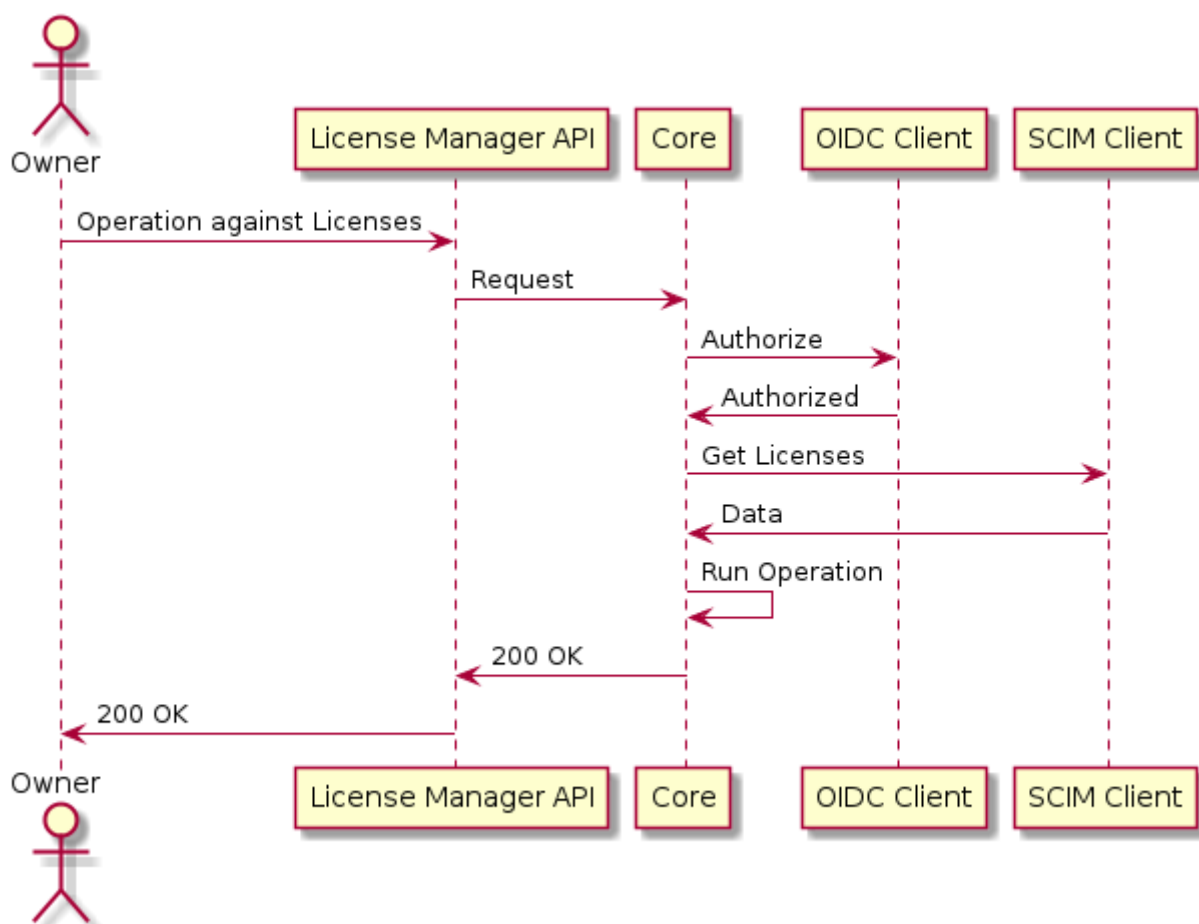

Figure 1. Owner Operation Case

2.3.2. End User access to a Resource protected with Licenses

The interaction within the system can be shown in the following diagram that illustrates how the

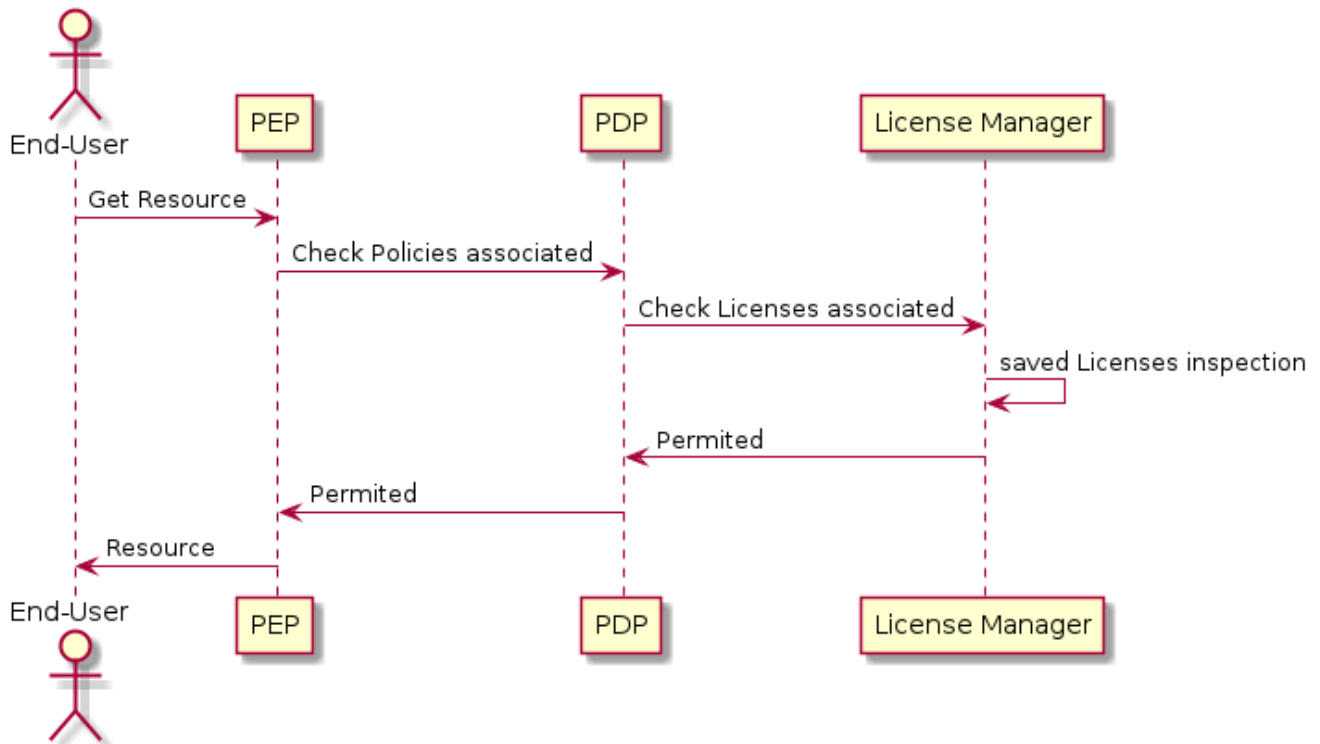end user activates a license check when accessing a resource through the PEP and PDP

Figure 2. Owner Operation Case

2.3.3. End User Licenese management through the User Profile

The User Profile offers an interface to manually edit the user attributes, in specific the Licenses associated.
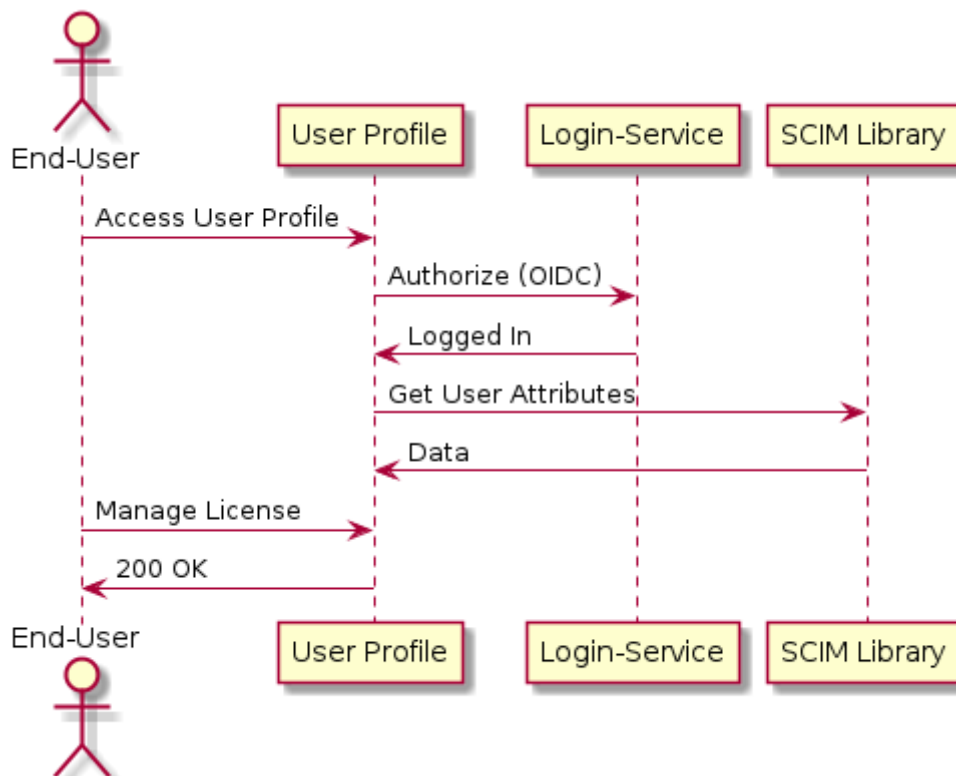The next diagram will show that interaction:

Figure 3. Owner Operation Case

Chapter 3. Design

3.1. Building Block Design

Content Description

This section contains:

- A concise breakdown of the Building Block in several independent services (when applicable). For each component, the following subsections are added:

  ◦ Overview and purpose: indicating the functionality covered by the component

  ◦ SW Reuse and Dependencies: indicating reuse of third party open source solutions (if any) and any pre-required Dependencies

  ◦ Interfaces: both internal to the building block and those exposed externally

  ◦ Data: Data usage of the building block, data flow and any GDPR concerns should be addressed here

  ◦ Applicable Resources: links and references to (Reference Docs), and repositories.

When a breakdown is necessary, a general overview of the building block can be given. On the contrary, no breakdown indicates a single component development with the same expected sections.

3.2. License Manager Core

3.2.1. Overview and Purpose

It is the main component of the building block, in which the scope of each license and its owner are determined. The objective is to recursively allocate licenses according to the privileges of the End User or the owner of the platform.
Within the core of the building block the information of the relations between licenses, resources and Users is stored and managed.

3.2.2. Software Reuse and Dependencies

All requirements for the executing of the reverse proxy are found under src/requirements.txt, and expect Python 3.6.9 or greater to work.

The most important are:

- **EOEPCA-SCIM**: Used to auto-register itself as a client to the Auth. Server upon startup

- **EOEPCA-OIDC**: Used to generate PAT tokens, validate OAuth tokens and JWTs.

- **WellKnownHandler**: Used to dynamically check the configuration of the Authorization Server on each execution. For example, it can get the needed endpoints for any API the PEP needs, such

as the token request for OIDC.

3.2.3. Interfaces

This component doesn't have any internal interfaces. For a reference of external interfaces see [External Interfaces] on Section 2 Overview

3.2.4. Data

3.2.4.1. Configuration

The License Manager gets all its configuration from the file located under `config/config.json`.

3.2.4.2. Data flow

The information that the License Manager handles is mainly user related since the authentication is properly handled by the OIDC library and the user information from the SCIM Library.

As it is a component that serves the PDP, a policy check will run the data flow of the License Manager to allow or deny access to a requested resource.

The core also deals with the billing service when it comes to communicating the use of volume and time of the licenses and its consequent impact on the invoicing of the associated resource.

3.2.5. Applicable Resources

- EOEPCA's SCIM Client - https://github.com/EOEPCA/um-common-scim-client
- EOEPCA's UMA Client - https://github.com/EOEPCA/um-common-uma-client
- EOEPCA's Well Known Handler - https://github.com/EOEPCA/well-known-handler

Chapter 4. User Story Traceability

Table 2. User Stories

Code	Description
EOEPCA-12	Reusable UMA Client Implementation
EOEPCA-221	Administrative tooling for integration and operation
EOEPCA-214	Add Usage of Persistence Volumes
EOEPCA-35	Ownership management for Resources
EOEPCA-210	Implementation of Helm Charts
EOEPCA-25	Registration of Resource References
EOEPCA-121	Propagation of End-User claims to the Resource Server
EOEPCA-205	Create Swagger Endpoint for Resource Protection API
EOEPCA-203	Separation of Proxy and Resource Management concerns
EOEPCA-194	Usage of relative URLs without proxy prefix
EOEPCA-189	Security: Verification of RPT Signatures
EOEPCA-187	Allow both RPT and ID Token Forwarding
EOEPCA-178	Default protection of resources
EOEPCA-126	Policy to Resource Data Model Extension
EOEPCA-173	Implementation of strict RPT Validation Measures
EOEPCA-120	Path-based resolution of Resource IDs
EOEPCA-114	Local Registration of Resources
EOEPCA-99	Command Line Interface UMA Client
EOEPCA-98	Baseline Enforcement Functionality
EOEPCA-94	Reusable UMA Client Implementation - End-User Functionality
EOEPCA-144	Resource Ownership Enforcement

<< End of Document >>