Publications Office of the European Union

infeurope

Initiative for Modelling the Legal Analysis Methodology

Target enterprise architecture

Disclaimer

Document metadata

Reference	WP 1.2: Target enterprise architecture
Corporate Author	Publications Office of the European Union
Author	Eugeniu Costetchi
Reviewers	Juraj Kuba, Miltiadis Charitos, Marcin Baryn
Contractor	Infeurope S.A.
Framework contract	10688/35368
Work package	WP 1.2
Delivery date	26 Feb 2021
Suggested readers	system architects, business analysts, business staff, technical staff

Abstract

Legal Analysis Methodology (LAM) for OP legal data aims to define the semantic aspects of the Publications Office of the European Union (OP) legal data on very specific level. It provides description of the metadata elements meaning, links them to various document types published in Official Journal or on EUR-Lex and describes the rules for attribution of values. It serves as an overarching framework for describing usage of a suite of legal data standards as applied in in their working context.

LAM plays an important role in the discovery of the EU legal resources (CELLAR, EUR-Lex, OP Portal), which is a central objective for the OP at large. The proper use of LAM leads to a significant decrease in the missing, confusing, incorrect or insufficient data for the stakeholders. In addition, it can increase the data interoperability and can facilitate automation for legal data at various levels.

This document provides a working definition of the architectural stance and design decisions that are to be adopted for the LAM data maintenance and dissemination life-cycle and the supporting services. This process is aligned with the semantic asset publication workflow currently employed by the Metadata and Standardisation Unit at the Publications Office of the European Union (OP).

Contents

1 Introduction **6**
- 1.1 Context . . . 6
- 1.2 Background considerations . . . 8
- 1.3 EU trajectory towards public sector linked open data . . . 9
- 1.4 Target audience . . . 9
- 1.5 Document scope . . . 10

2 Architecture building blocks **12**
- 2.1 Methodology . . . 12
- 2.2 Architecture views . . . 13
- 2.3 ArchiMate elements . . . 13
- 2.4 Service Oriented Architecture (SOA) . . . 18

3 Motivation architecture **20**
- 3.1 Prototypical motivation structure . . . 20
- 3.2 OP legal analysis team . . . 22
- 3.3 Different OP services and EU institutions . . . 23
- 3.4 LAM contractors . . . 24
- 3.5 Publications Office of the European Union . . . 25

4 Business architecture **28**
- 4.1 Prototypical business structure . . . 28
- 4.2 Actors and roles . . . 29
- 4.3 Maintenance of semi-structured LAM data . . . 32
- 4.4 Maintenance and publication of structured LAM data . . . 34
- 4.5 Asset lifecycle process . . . 35
- 4.6 Role allocation in the lifecycle process . . . 37

5 Application architecture **39**
5.1 Prototypical application structure 39
5.2 LAM specific application architecture 41
5.2.1 LAM transformation tool 41
5.2.2 LAM online tool 42
5.2.3 LAM validation tool 44
5.3 LAM lifecycle application architecture 45
5.3.1 Evolution management 45
5.3.2 Implementation 46
5.3.3 Validation 47
5.3.4 Release 47
5.3.5 Publication 48

6 Conclusions **49**
6.1 Summary 50
6.2 Final word 50

Chapter 1

Introduction

This document provides a working definition of the architectural stance and design decisions that are to be adopted for the Legal Analysis Methodology maintenance and dissemination lifecycle and the supporting services. This process is aligned with the semantic asset publication workflow currently employed by the Standardisation Unit (SU) at the Publications Office of the European Union (OP).

In this document is proposed a target architecture supported by a motivation structure derived from the project requirements specifications.

1.1 Context

OP manages EU legal data coming from different sources. Based on common standards (IMMC[1], CDM - Common Data Model [18, 19], Controlled Authority Tables[2], ELI – European Legislation Identifier [11, 12], ECLA – European Case-law Identifier[36, 37]), the legal data can be received from institutions, legal analysis contractor or created directly by OP.

Legal Analysis Methodology (LAM) for OP legal data aims to define the semantic aspects of the OP legal data on very specific level. It provides description of the metadata elements meaning, links them to various document types published in Official Journal or on EUR-Lex and describes the rules for attribution of values. It

[1]see `https://op.europa.eu/en/web/eu-vocabularies/immc`
[2]see `https://op.europa.eu/en/web/eu-vocabularies/authority-tables`

serves as an overarching framework for describing usage of other standards in their working context.

LAM plays an important role in the discovery of the EU legal resources (CELLAR, EUR-Lex, OP Portal), which, at large, is a central objective for the OP. The proper use of LAM leads to a significant decrease in the missing, confusing, incorrect or insufficient data by the stakeholders. In addition, it can increase the data interoperability (by better understanding) and can facilitate automation for legal data at various levels.

Originally LAM started as a set of rules and definitions organised in a Word document. This representation is arguably operational for humans, but is non-readable for machines, and will be referred further on as *unstructured LAM data*.

In 2019 a series of discussions started between the LAM maintenance team in OP.C2 unit and metadata and standardisation team from OP.A1 unit leading to a set of modelling experiments aiming to model and organise the descriptions comprised in the LAM Word document[27]. These further evolved into the unofficially called *Initiative for Modelling the Legal Analysis Methodology* (IM-LAM) [7] aiming (a) to provide a level of formalisation to LAM used at the Publications Office for the legal document metadata and (b) to create an online tool offering an easy access to LAM for different stakeholders (consult, search, download) and enabling exchange of information between them (changes in LAM, proposed changes, consultations, feedback).

The first part of the initiative, scoped to modelling and formalisation of LAM, referred here as LAM#1 project [7], was concluded at the end of 2019. The project deliverables, which comprised documents, transformation scripts and formal data and models, are available in the *lam4vb3* GitHub repository[3]. As a prerequisite the LAM was transformed into an Excel file [6], which we will call in this document a *semi-structured LAM data*. This Excel file was used as the source for creation of the LAM data in LAM-SKOS-AP representation [8], further referred to as *structured LAM data*.

The second part of the initiative, to establish a single point of access for LAM data through a so called *online tool*, began in 2020 and is referred here as LAM#2 project. Besides the establishment of the dissemination mechanism for the LAM data in the OP Portal (requested in the project requirements specifications [22]) the project

[3] see `https://github.com/eu-vocabularies/lam4vb3`

also aims at establishing architectural and design decisions both at the application and business levels (presented in this document). This implies placements of the LAM dataset, as a semantic asset, into the OP ecosystem where a data governance methodology and a lifecycle model is followed. The metadata and standardisation team from OP.A1 unit plays a central role here and will be further detailed in Section 4.2.

1.2 Background considerations

Given the increasing importance of data standards for the EU institutions, a number of initiatives driven by the public sector, industry and academia have been kick-started in recent years. Some have grown organically, while others are the result of standardisation work.

Each of these initiatives introduce specific vocabularies, semantics and technologies, resulting in a heterogeneous state of affairs. These differences hamper data interoperability and thus its reuse by the other institutions or by the wider public. This creates the need for a common approach for publishing public reference data and models. Moreover, the data which instantiates these public models, available from different sources, shall be easily accessed, linked, and consequently reused.

In order to improve transparency and to boost innovation via the reuse of public data, the PSI directive [17] across the EU calls for open, unobstructed access to public data. The reference data maintained and published by the OP has been identified as data with a high-reuse potential [1]. Therefore, making this data available in machine-readable formats, as well as following the data as a service paradigm, are required in order to maximise its reuse.

In this context, the Publications Office of the European Union maintains and publishes an ever-increasing number of *reference data* which are vital in the context of inter-institutional information exchange. With regards to reference data, the OP provides an ever-increasing number of services to the main institutional stakeholders and with the aim to extend them to a broader public, enabling active or passive participation in the reference data life cycle, standardisation and harmonisation.

1.3 EU trajectory towards public sector linked open data

European institutions started out to adopt Semantic Web and Linked Data technologies as part of their visions to become data-centred e-government bodies [13, 15].

The EU institutions also aim for implementation of a single digital gateway to "facilitate interactions between citizens and businesses, on the one hand, and competent authorities, on the other hand, by providing access to online solutions, facilitating the day-to-day activities of citizens and businesses and minimising the obstacles encountered in the internal market.

The existence of a single digital gateway providing online access to accurate and up-to-date information, to procedures and to assistance and problem-solving services could help raise the users' awareness of the different existing online services and could save them time and expense" [16]. This is well in line with earlier established goals for encouraging the open data and the re-use of public sector information [14, 17].

Many of the legacy systems used in the EU institutions use XML data format governed by the XSD schemes [26]. These formats are used for both: document structure and document exchange. The aim is to evolve technologically so that both existing and new systems are capable to operate with semantic data representations using RDF [39], OWL [25, 24], SHACL [21] and other representations, and serialised in at least the RDF/XML [2, 28], Turtle [4] and JSON-LD [31, 30] formats.

For this reason, the OP has already been publishing data in RDF format for over a decade using the Cellar repository [18]. Also, the LAM team is committed to publishing and disseminating reference data in semantic formats and also making them available in a human readable representation in OP Portal.

1.4 Target audience

The present document is intended to be read and understood by the following audience:

- Enterprise architects and data governance specialists
- Business team involved in the data lifecycle

- Technical staff in charge of operating workflow components
- Developers in charge of workflow and component implementation
- Third parties using the services and LAM data

1.5 Document scope

This document aims to describe the baseline architecture for establishment of the business and application services involved in the initiative for modelling (and dissemination) LAM data.

This architecture covers the maintenance and management of the semi-structured LAM data (see Section 4.3) and the transition to authoring and publishing of LAM structured data (see Section 4.4) following the lifecycle process adopted in the A2 unit for semantic assets (see Section 4.5).

This includes managing the incoming requests, editing the reference assets in VocBench3 system [32, 33], then exporting the RDF data and passing them as input to a set of processes that validate, assess, transform, package and, finally, publish the LAM data in OP Portal, as human-readable content and Cellar [18], as machine-readable data.

This document provides a motivation, business and application account. Each of these accounts is limited strictly to the success scenario of the above-mentioned use case and does not include possible extensions and variations.

There is a series of aspects that were intentionally left out out scope. For example the recommendations related to the data governance both internally within the LAM team and also externally in relation with partners, stakeholders and clients are not covered. Also, no implementation details are specified for the new components. Little or not account is provided about the data structures and static objects used in the business process or exchanged between the application services. No monitoring or performance measurement systems are foreseen by this architecture, which, in future work shall be considered across all architectural levels: starting from motivation level key performance indicators (KPI), continuing with business level process monitoring, down to performance measurement of the applications and the infrastructure indicators.

A high level treatment is provided on how the workflow orchestration shall be organised, what process automation service to use, or what technologies could be chosen for that. Such decisions shall be carried out in subsequent steps in close cooperation with A2 unit at the level of the technical team, business team and the sector management.

With this scope in mind, the next section presents a short introduction into the enterprise architecture language and methodology adopted in this document. The next section can be skipped by the readers familiar with ArchiMate Language [35], who can proceed to Section 3 detailing the structure of motivations behind this project and overall initiative for modelling LAM data.

Chapter 2

Architecture building blocks

This chapter provides some foundations about the notations, definitions and the general approach adopted here to model the enterprise architecture.

2.1 Methodology

In this document we take an enterprise architecture perspective and aim to provide several architecture views (see Section 2.2) which are necessary and sufficient to describe the asset lifecycle process.

In developing this architecture, we are in part using the TOGAF [34] methodology, which is, in fact, a framework for enterprise architecture that provides an approach for designing, planning, implementing and governing an enterprise information technology architecture. Although we do not follow this framework extensively, relevant parts of it were applied to the goals of this architecture.

For representing the architecture models, ArchiMate language [35] is adopted. It is an open and independent enterprise architecture modelling language to support the description, analysis and visualisation of architecture within and across business domains in a clear and unambiguous way.

Based on the initial requirements specifications document [22] and a series of interviews conducted with the LAM team, the structure of the project stakeholders motivations was established. It is presented in Section 3.

The business and application layers of this architecture, are a gradual fleshing out of the use cases presented in the project preliminary requirements [7], the functional requirements in the requirements specifications document [22] and the architecture for managing and publishing semantic assets [9] adopted by the A2 unit.

The ArchiMate diagrams corresponding to each architecture layer were modelled and designed using Enterprise Architect Tool [29]. Finally this report was written covering the overall architecture.

2.2 Architecture views

Architecture views are an ideal mechanism to purposefully convey information about architecture areas. In general, a view is defined as a part of an Architecture Description that addresses a set of related concerns and is tailored for specific stakeholders. A view is specified by means of an architecture viewpoint, which prescribes the concepts, models, analysis techniques, and visualisations that are provided by the view. Simply put, a view is what you see, and a viewpoint is where you are looking from [35].

An architecture view expresses the architecture of the system of interest in accordance with an architecture viewpoint (or simply "viewpoint"). There are two aspects to a viewpoint: the concerns it frames for the stakeholders and the conventions it establishes on views [35].

Viewpoints are designed for the purpose of communicating certain aspects and layers of an architecture. In this document we address *the motivation view* (Section 3), *the business view* (Section 4), and *the application view* (Section 5).

Instead of describing what each of these views represents in this section, we decided to provide such a description in the beginning of each of the subsequent sections. This way, we aim to ease reading the section by providing the reader a fresh introduction into the structure of a prototypical layer architecture before the actual architecture is described.

2.3 ArchiMate elements

This section presents the ArchiMate elements, in terms of their definition and the graphical notation, which we employ in each of the architecture views.

Table 2.1: Overview of the relevant motivation elements [35]

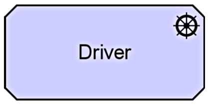
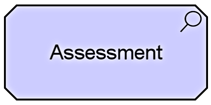
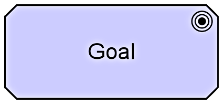
Element	Definition	Notation
Stakeholder	Represents the role of an individual, team, or organisation (or classes thereof) that represents their interests in the effects of the architecture.	
Driver	Represents an external or internal condition that motivates an organisation to define its goals and implement the changes necessary to achieve them.	
Assessment	Represents the result of an analysis of the state of affairs of the enterprise with respect to some driver.	
Goal	Represents a high-level statement of intent, direction, or desired end state for an organisation and its stakeholders.	

Table 2.2: Overview of the relevant business layer elements [35]

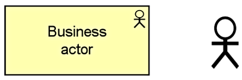
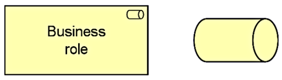
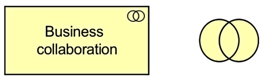
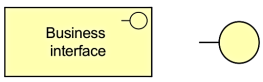
Element	Definition	Notation
Business actor	Represents a business entity that is capable of performing behaviour.	
Business role	Represents the responsibility for performing specific behaviour, to which an actor can be assigned, or the part an actor plays in a particular action or event.	
Business collaboration	Represents an aggregate of two or more business internal active structure elements that work together to perform collective behaviour.	
Business interface	Represents a point of access where a business service is made available to the environment.	

Table 2.2 continued from previous page

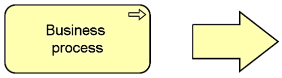
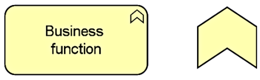
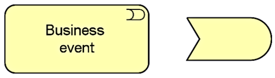
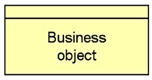
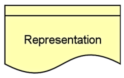
Business process	Represents a sequence of business behaviours that achieves a specific result such as a defined set of products or business services.	
Business function	Represents a collection of business behaviour based on a chosen set of criteria (typically required business resources and/or competencies), closely aligned to an organisation, but not necessarily explicitly governed by the organisation.	
Business event	Represents an organisational state change.	
Business service	Represents explicitly defined behaviour that a business role, business actor, or business collaboration exposes to its environment.	
Business object	Represents a concept used within a particular business domain.	
Representation	Represents a perceptible form of the information carried by a business object.	

Table 2.3: Overview of the relevant application layer elements [35]

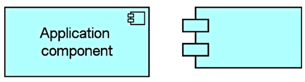
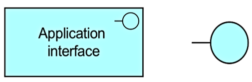
Element	**Definition**	**Notation**
Application component	Represents an encapsulation of application functionality aligned to implementation structure, which is modular and replaceable.	
Application interface	Represents a point of access where application services are made available to a user, another application component, or a node.	

Table 2.3 continued from previous page

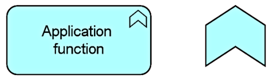
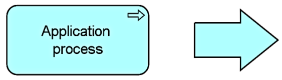
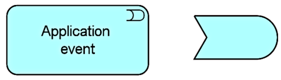
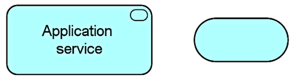
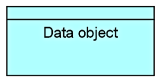
Application function	Represents automated behaviour that can be performed by an application component.	
Application process	Represents a sequence of application behaviours that achieves a specific result.	
Application event	Represents an application state change.	
Application service	Represents an explicitly defined exposed application behaviour.	
Data object	Represents data structured for automated processing.	

Table 2.4: Overview of the relevant technology layer elements [35]

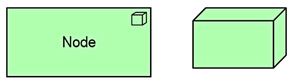
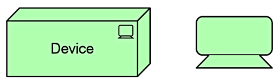
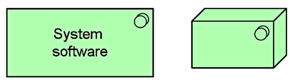
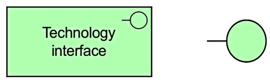
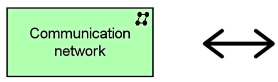
Element	**Definition**	**Notation**
Node	Represents a computational or physical resource that hosts, manipulates, or interacts with other computational or physical resources.	
Device	Represents a physical IT resource upon which system software and artefacts may be stored or deployed for execution.	
System software	Represents software that provides or contributes to an environment for storing, executing, and using software or data deployed within it.	
Technology interface	Represents a point of access where technology services offered by a node can be accessed.	
Communication network	Represents a set of structures that connects nodes for transmission, routing, and reception of data.	

Table 2.4 continued from previous page

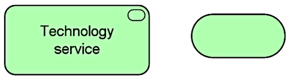
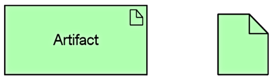
Technology service	Represents an explicitly defined exposed technology behaviour.	
Artefact	Represents a piece of data that is used or produced in a software development process, or by deployment and operation of an IT system.	

Table 2.5: Overview of the ArchiMate relationships [35]

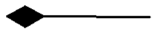
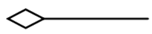
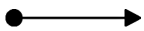
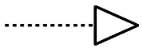
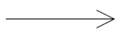
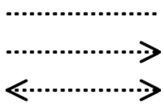
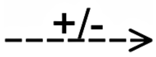
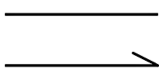
Element	**Definition**	**Notation**
	Structural Relationships	
Composition	Represents that an element consists of one or more other concepts.	
Aggregation	Represents that an element combines one or more other concepts.	
Assignment	Represents the allocation of responsibility, performance of behaviour, storage, or execution.	
Realisation	Represents that an entity plays a critical role in the creation, achievement, sustenance, or operation of a more abstract entity.	
	Dependency Relationships	
Serving	Represents that an element provides its functionality to another element.	
Access	Represents the ability of behaviour and active structure elements to observe or act upon passive structure elements.	
Influence	Represents that an element affects the implementation or achievement of some motivation element.	
Association	Represents an unspecified relationship, or one that is not represented by another ArchiMate relationship.	
	Dynamic Relationships	

Table 2.5 continued from previous page

Triggering	Represents a temporal or causal relationship between elements.	
Flow	Represents transfer from one element to another.	
	Other Relationships	
Specialisation	Represents that an element is a particular kind of another element.	
Junction	Used to connect relationships of the same type.	(And) Junction Or Junction

2.4 Service Oriented Architecture (SOA)

This section provides a motivation for adopting the design guidelines and best practices offered by the *Service Oriented Architecture* (SOA) [20] paradigm. This is especially important for guiding the development and engineering teams away from a monolith tightly coupled software capabilities that are executed on a dedicated server, towards loosely coupled services exposed through REST APIs capable of running as scalable processes in a distributed environment.

Service-oriented architecture is a style of software design where services are provided to the other components by application components, through a communication protocol over a network. A SOA service is a discrete unit of functionality that can be accessed remotely and acted upon and updated independently. SOA is also intended to be independent of vendors, products and technologies.

SOA is believed to help businesses respond more quickly and more cost-effectively to changing market conditions. This style of architecture promotes high component reuse and facilitates interconnection between components. The SOA systems are known to be resilient in the light of changing technologies because the components are loosely coupled and could easily be replaced by alternative implementations exposing with the same interfaces. SOA could be regarded as an architectural evolution rather than as a revolution. It captures many of the best practices of previous software architectures and it is a practical approach for cloud computing [38].

Service-oriented architecture can be implemented with web services or Micro-services [3]. This is done to make the functional building-blocks accessible over standard Internet protocols that are independent of platforms and programming languages. These services can represent either new applications or just wrappers around existing legacy systems to make them network-enabled [5]. By following this approach, the application implementation materialises as a suite of micro-services. Where necessary, the micro-services can be implemented as wrappers around existent components and at a later stage the components can be replaced by new implementations.

Chapter 3

Motivation architecture

This chapter presents the motivation and goal structure of the LAM team for this project. This motivation structure is also situated in the context of the Publications Office, this way providing a rationale for the initiative for modelling LAM data in the first place.

This motivation view helps address questions on why a stakeholder demand for certain capabilities is meaningful, model crucial drivers and root causes behind the demand, actual goals and related outcomes, as well as concrete requirements for further development. In short, it answers the crucial questions to WHOM, WHY and WHAT.

We do not aim for an in depth coverage of the motivation architecture here: in the sense that it cannot be considered as a fully fledged decision-making tool for the management. The focus is to account for the context, stakeholders and their drivers and interests.

3.1 Prototypical motivation structure

The structure of motivations, in ArchiMate, is organised hierarchically in several layers. For simplicity, we have chosen to use the top four layers: *stakeholders, drivers, assessments and goals*; leaving out the *outcomes*, *principles* and *requirements*. Figure 3.1 depicts the organisation of the motivation architecture. The structure starts at the top with enumerating the stakeholders, who can be individuals, teams or organisations that represent their interests in the effects of the architecture [35].

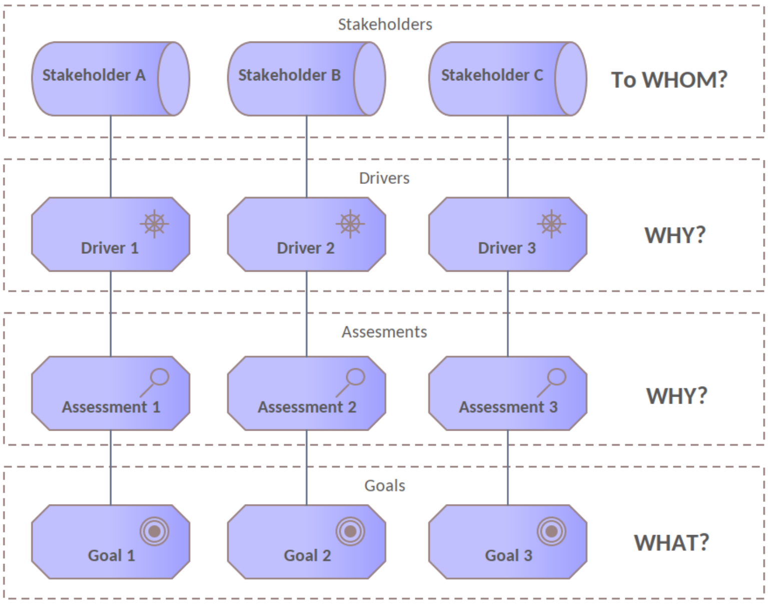


Figure 3.1: The layered motivation structure

Stakeholders have associated interests, concerns or *drivers*, which represent internal or external conditions that motivate an organisation to define goals.

Assessments represent results of analysis of the state of affairs with respect to a driver. They reveal strengths and weaknesses, opportunities and threats to an area of interest. Assessments are associated with *goals* which represent a high-level statement of intent, plus direction to desired end state for an organisation and its stakeholders [35].

In the context of the current project the following stakeholders have been identified:

- OP legal analysis team (OP.C.2.003)
- Different OP services
- EU institutions
- LAM contractors
- Publications Office of the European Union (OP)

Next we present the motivation structure of spread over several sections addressing each stakeholder in part.

3.2 OP legal analysis team

The legal analysis team at the OP is the main stakeholder in this project. The main driver of this team is to establish a single point of access for the LAM data that can serve also as the single point of truth for this dataset.

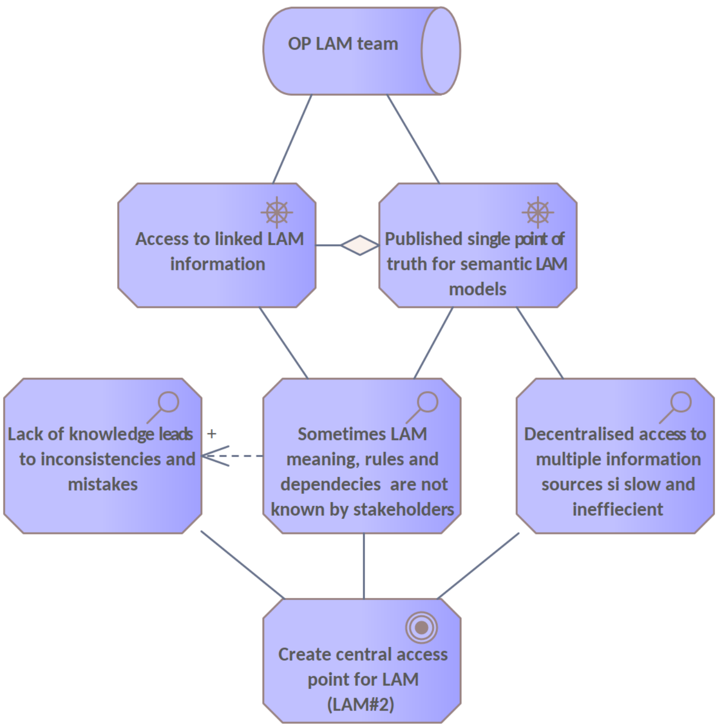


Figure 3.2: Motivation structure of the OP legal analysis team

One particular feature that is of special importance is to to also link other various datasets on which LAM relies, such as Common Data Model (CDM) [18, 19], authority tables published at the EU Vocabularies[1], European Legislation Identifier (ELI) and others. The linked LAM information driver is a sub-goal to establishing a single point of access driver, and this is modelled via part-of relationship in Figure 3.2.

[1] https://op.europa.eu/en/web/eu-vocabularies

In the context of the project these two drivers are hindered by three issues. First, multiple sources of information published in an uncoordinated manner on disparate sources are difficult to access and consume. This is especially the case when the information available at decentralised data sources needs to be used coherently in combination with other data sources.

Another issue is that the meaning, rules and dependencies of the LAM model are sometimes not known by the stakeholders due to various reasons. One reason is the failure to find this information. Another reason is the informal explanation which may be incomplete, ambiguous or vague leading to multiple interpretations. And this leads to the third issue that the lack of precise formally defined knowledge is further propagated into the domain where LAM is applied and materialises as inconsistencies and mistakes in the data, system implementations, infrastructure configurations, exchange protocols and other aspects of the information systems.

To overcome these issues the goal of creating a central access point for the LAM data is adopted. This being the main goal of this project (LAM#2).

3.3 Different OP services and EU institutions

At the Publications Office various internal units and the services they expose operate with legal data and metadata. Having access to the semantic description of the OP legal data is of primary concern for these services collectively. This is schematically depicted in Figure 3.3.

Implementation of the single point of access for semantic LAM model can be conceptualised as a sub-driver for the need to access semantic descriptions of OP legal data and metadata, which is represented through an aggregation relation in Figure 3.3. Both motivations are hindered by the fact that decentralised access to multiple information sources is slow and inefficient. Moreover, LAM meaning, rules and dependencies are not always known to the interested stakeholders. To overcome these limitations, the current architecture aims at describing how a central dissemination point for LAM can be established.

The EU institutions, at large, as a collective consumer of legal metadata definitions has the same needs as the OP services. In addition, a notification mechanism is desired to inform the interested players of changes and updates in the LAM data. This need materialises directly as a feature of the system to disseminate LAM data.

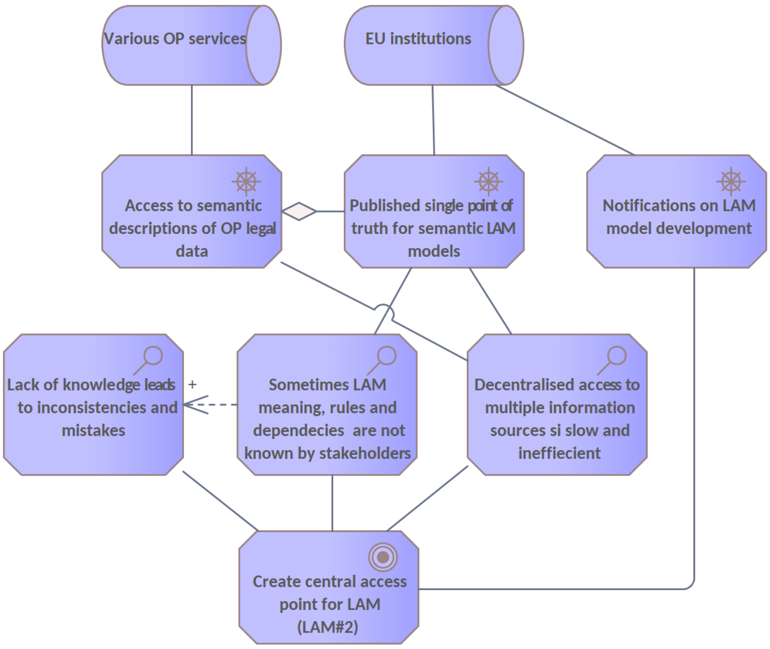


Figure 3.3: Motivation structure of different OP services

3.4 LAM contractors

The LAM contractors are a set of special stakeholders as they not only need to consult LAM data for information, but they are the agents that are actively involved in applying the specifications in practice. Often times, they will be those who inform the LAM team about possible issues in the LAM model or request extensions to it in order to accommodate new situations. The main driver for the LAM consultants is the consultancy on LAM and follow-up, depicted in Figure 3.4.

Traditionally, the LAM model was maintained as a MS Word document that is an unstructured (at least not for the machines) data representation. A direct consequence of this approach is that no automation, validation or consistency checking is possible with such representations. To overcome this limitation an initiative to structure LAM data into a machine readable model was performed at the end of 2019 (referred as LAM#1 project). LAM#1 deliverables are available in the *lam4vb3* GitHub repository[2].

[2]see `https://github.com/eu-vocabularies/lam4vb3`

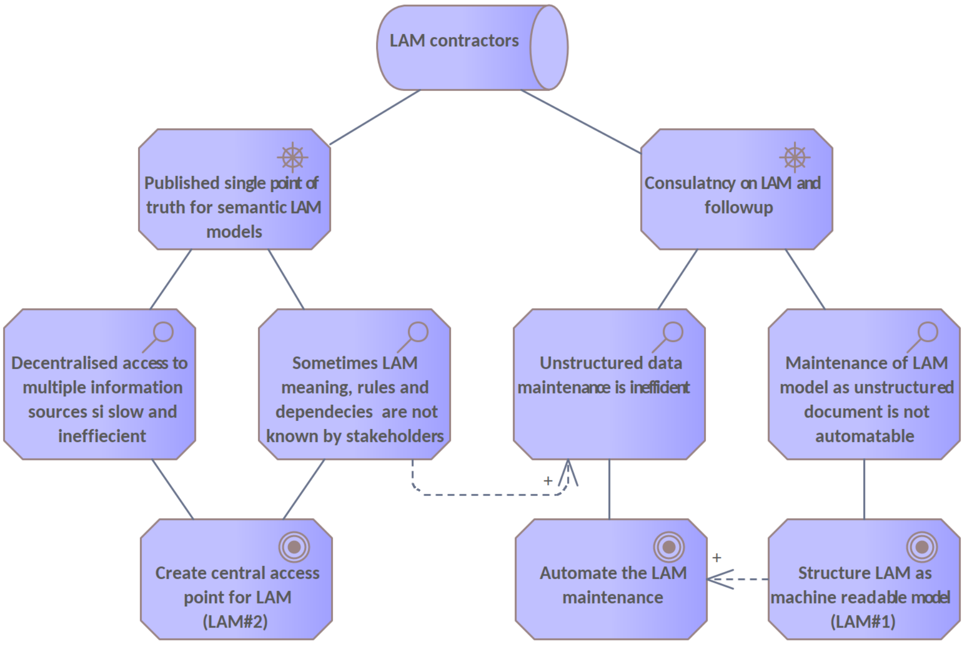


Figure 3.4: Motivation structure of LAM contractors

Another issue is that, as no machine assistance is possible to implement, the maintenance of these data becomes increasingly more difficult due to highly interlinked nature of the LAM model. This approach does not scale and is inefficient. Moreover the effect is amplified as sometimes the LAM meaning, rules and dependencies are not known by the LAM contractors or even the LAM maintenance team. In order to overcome this limitation, a set of automation functionalities and processes shall be established. This automation is out of current project scope and shall be addressed elsewhere.

On the left side of Figure 3.4, the driver, assessments and goal are repeated from the sections above as they are central to the current project and are, therefore, shared by all of the stakeholders.

3.5 Publications Office of the European Union

The Publications Office of the European Union defines drivers at a higher level of abstraction; yet they are very relevant to mention because the current project contributes directly to those interests. Figure 3.5 depicts the motivation structure of the OP relevant to the context of the current project.

OP is interested in the semantic operability both across EU institutions and the intra-institutional information systems. To increase the shared common conceptualisation captured by the data models, they need to carry a certain level of formality, semantics that shall be verifiable for completeness and especially for soundness. Unfortunately not all data is represented in machine readable format and even less is based on semantic models.

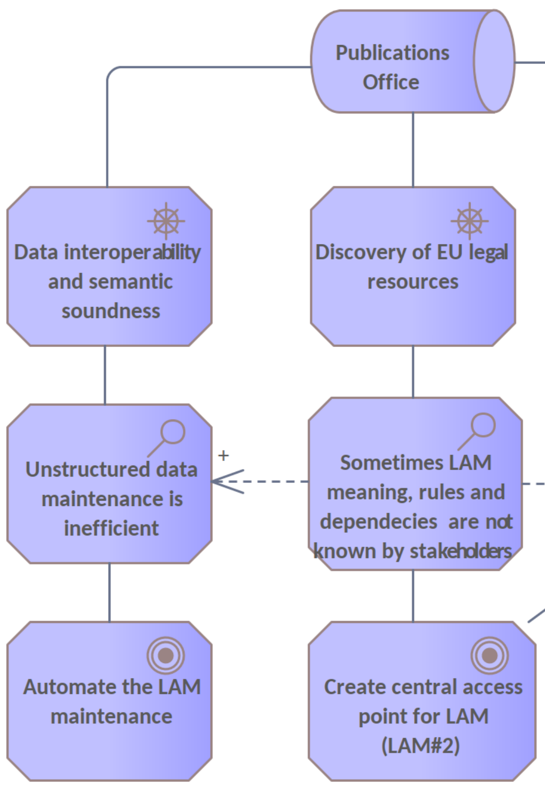


Figure 3.5: Motivation structure of the Publications Office of the European Union

Another broad OP interest is maintaining and increasing by possible means the data quality. Causes such as lack of or impaired access to knowledge, unfortunately directly leads to inconsistencies and mistakes, which decrease the data quality. In order to overcome these limitations, the creation of a central access point for LAM addressed to a large extent the problem of knowledge shortage is required.

Finally, a driver, which is at the heart of the OP as an institution, is to facilitate discovery of EU legal resources. In the context of the current project, this driver is hindered by the inability to easily find and access LAM meanings, rules and dependencies. Therefore, the dissemination of LAM data shall be done in such a way that the relations to external data sources are presented in an intuitive manner and the links are easy to navigate. Moreover, an inventory of links to the most used resources shall be disseminated with the LAM data.

Chapter 4

Business architecture

This chapter addresses the business architecture of the initiative to model LAM data. The aim is to describe the internal processes, events and roles answering questions concerning WHO shall do WHAT and WHEN.

This chapter first presents the decontextualised architecture specific to LAM modelling initiative and then proceeds to place the management of the LAM asset in the context of a lifecycle model.

However, the description starts by explaining how a prototypical business architecture is structured and that will serve as a framework to better understand the diagrams in this chapter.

4.1 Prototypical business structure

Following the metaphor of layers presented in the motivation view (see Section 3.1), the organisation of business structure is also explained in terms of layers. Figure 4.1 depicts three layers with the most important elements of the business structure.

The topmost layer accounts for the external players or *actors*, which represent a business entity that is capable of performing behaviour and *roles*, which represent skills and responsibilities for performing specific behaviours, and to which an actor can be assigned [35].

The middle layer represents the *services* that are offered by the organisation to

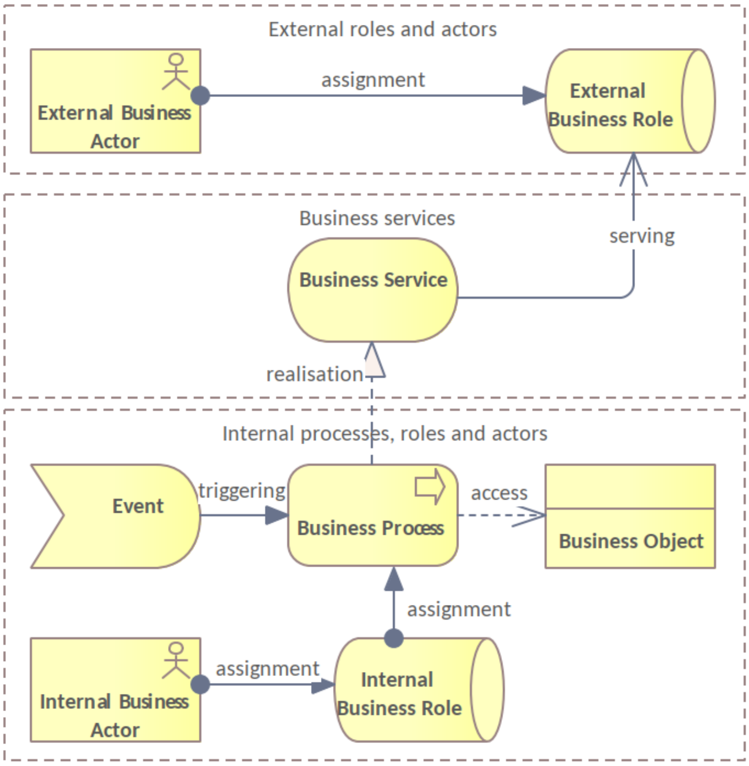


Figure 4.1: The prototypical business structure view

external players. A business service represents explicitly-defined behaviours that a business role, business actor or business collaboration exposes to its environment [35].

The lower layers accounts for the internal organisation in terms of *events*, *roles*, *processes* and *objects*. The business process represents a sequence of business behaviours that achieves a specific result such as a defined set of products or business services. The business event represents an organisational state change; while a business object represents a (passive) concept used within a particular business domain.

4.2 Actors and roles

This section describes identified actors and roles relevant to the context of LAM data dissemination. Figure 4.2 depicts their relations.

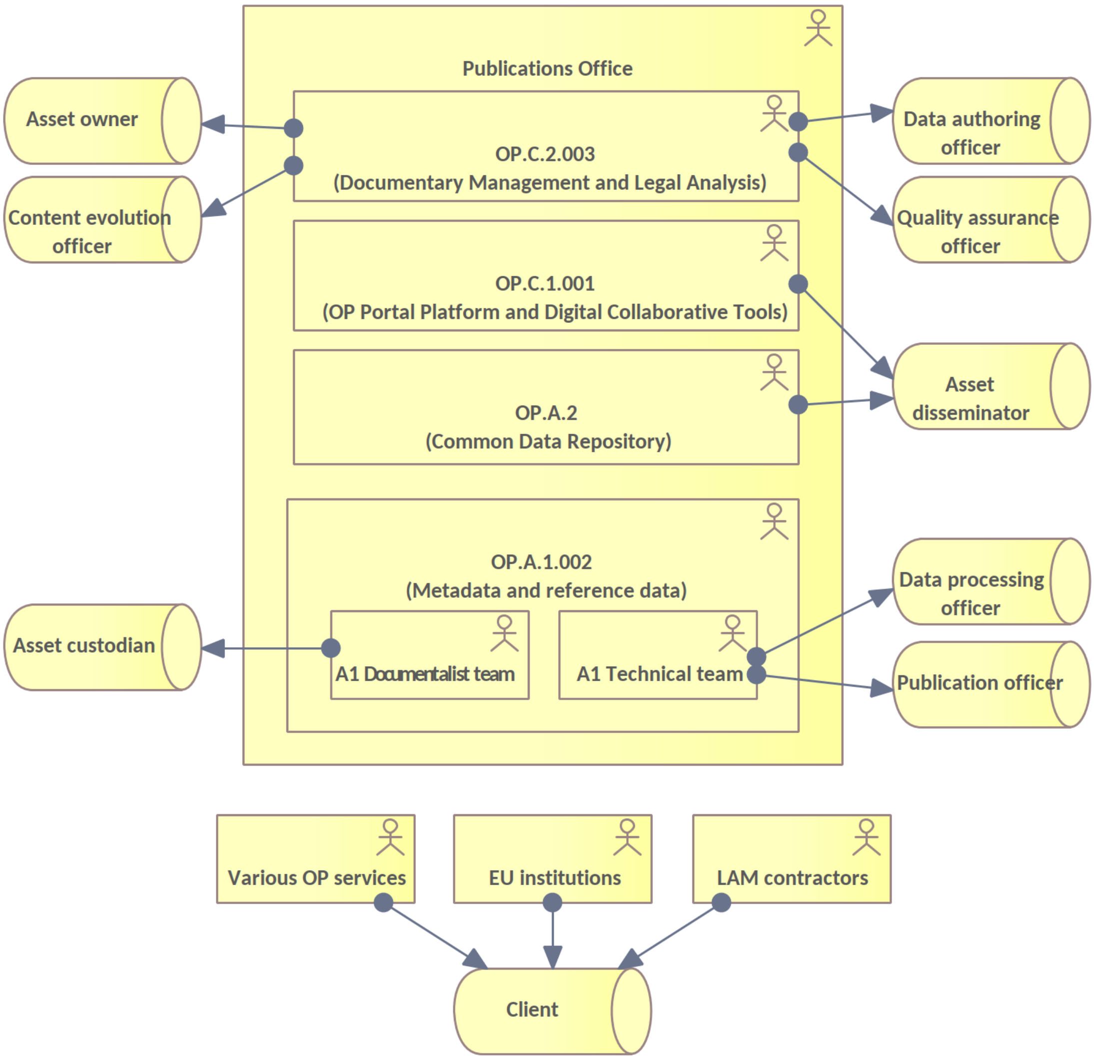


Figure 4.2: The involved actors and their roles in the LAM data lifecycle

The OP is a stakeholder in the project but does not have a direct role in the business processes, rather it serves as a frame of reference for the other actors.

The *Documentary Management and Legal Analysis sector (OP.C.2.003)* at the OP is the project initiator and has a central role to play in the business process. It plays the roles of asset owner, content evolution, data authoring and quality assurance officer. It is not yet decided whether it will play the role of the asset manager or if this role will be transferred to the Metadata and Reference Data sector, who is also the data custodian.

The *asset owner* has accountability for the asset content throughout its life cycle, including decision making authority for creating, classifying, restricting, regulating and administering its use or disclosure. The implementation of these decisions can be delegated.

The *content evolution officer* is the interface with the client collecting change requests, assessing business needs and translating them into data management requirements, all being summarised and documented case-by-case.

The *data authoring officer* (informally referred to as the *editor*) is responsible for editing data in a content management system implementing the cases prepared by the request manager. This is a business role that is responsible for implementing the request case specifications by modifying the data asset accordingly with the provided tools.

The *quality assurance officer* (informally referred to as the *validator*) is a business role that is responsible for ensuring the request case implementation is complete and correct. This role has a special importance and contributes to applying the four eyes principle in the asset lifecycle. Quality assurance officers validate that the content implementation is correct from both technical and business points of view.

The *Metadata and Reference Data sector (OP.A.1.002)* at the OP offers the technical support for LAM data lifecycle management, including editing, validation capabilities, and publication on the dissemination platforms. Because it offers business services and technical services, the actor is split into two sub-components: *the documentalist team* to act as asset custodian and correspondingly *the technical team* taking the role of data processing and publication officer.

The *asset custodian* (informally referred to as the *asset manager*) operates as a trustee on behalf of the asset owner and is responsible for data content, context, and associated business rules. This role ensures the development and enforcement of standards for data within their care.

The *data processing officer* is a technical role that is responsible for preparing the assets for publication and distribution on various channels. The responsibilities include, but are not limited to, data storage, manipulation, automatic transformation and generation of validation and assessment reports.

The *publication officer* is a technical role responsible for packaging and disseminating assets to specialised platforms. This role may also include preparation of release notes and impact assessment preparation.

The OP Portal website is the main dissemination channel for the human readable representation of LAM data. The sector in charge of *OP Portal Platform and Digital Collaborative Tools (OP.C.1.001)* play the role of the main asset disseminator.

The *Common Data Repository unit (OP.A.2)*, in charge of Cellar system [18] also plays the role of asset disseminator because the machine readable representation of the LAM data is published in the Cellar system.

The *asset disseminator* role provides with reliable data the dissemination capabilities which are meant to make assets available for the clients. The dissemination of assets is done either in human readable or in machine readable representations.

The LAM contractors, various EU institutions and OP services have been identified already in the motivation structure section (see Section 3) as stakeholders. From the business point of view these stakeholders are agents playing the role of a client.

The *client* (either the change requester and or the data user) is a generic external role who, on the one hand, consumes data and services provided by the asset owner and, on the other hand, suggests creation and publication of new assets or modification of existing ones.

4.3 Maintenance of semi-structured LAM data

In a previous project dealing with creating a model for the LAM data (internally referred to as LAM#1 [7]) a set of artefacts and a data management methodology was created in order to aid the LAM management team to organise and edit the data.

After the LAM#1 project [7] was completed three new capabilities were added to the LAM team: (a) management of LAM data in a semi-structured representation (in an Excel workbook [23]) thanks to a specification on how to structure the workbook [6], (b) the possibility to transform the LAM data into RDF representation following the LAM-SKOS-AP [8] model specifications and (c) uploading and editing the LAM data in the VocBench3 system [32, 33]. Figure 4.3 depicts this situation.

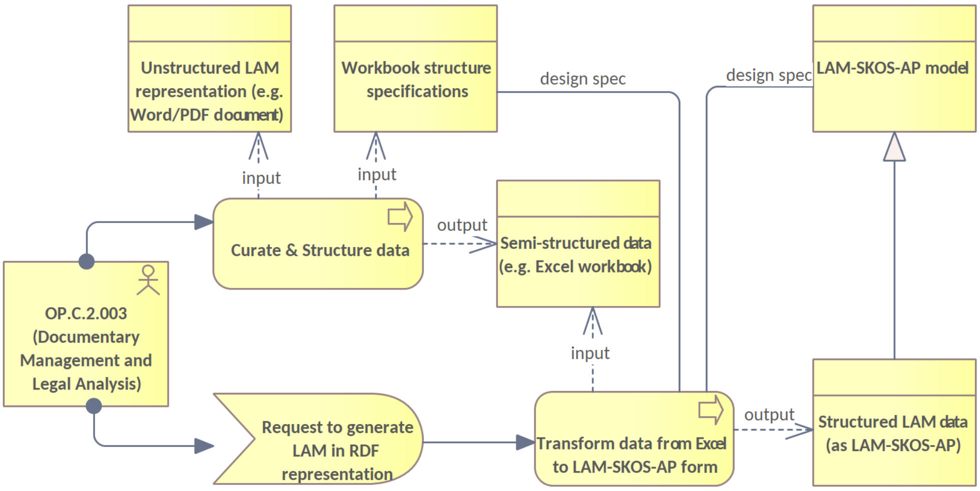


Figure 4.3: The maintenance of LAM data in the semi-structured form

Over time, the LAM team (the actor on the left side of the diagram) performs the curation and structuring of the LAM data producing the Excel workbook (semi-structured representation). As the first input to this process, the team relies on the existent Word documents describing Eur-Lex LAM structure in human readable form [27] and the second input is the specifications for structuring the Excel workbook [6].

When a satisfiable version of the semi-structured data is available, the LAM team requests generation of the RDF representation for it. The reason behind this is to ultimately migrate toward editing LAM data in VocBench3 [32] and away from Excel [23].

In the past, this transformation was performed by the contractor. Now this may be transferred to the Metadata and Standardisation Unit or continue using the contractor for technical support. This is necessary because the transformation is not a one time operation but continues over a period of time as both the data and the generation script need to pass through a series of evolutions before arriving at a stable RDF representation.

The transformation output is the LAM data instantiating LAM-SKOS-AP model [8]. The instantiation relationship is depicted in Figure 4.3 through a realisation connector from Structured LAM data business object to the LAM-SKOS-AP model business object. Next section explains how the LAM data is maintained and published in RDF representation.

4.4 Maintenance and publication of structured LAM data

Ultimately, the LAM data shall be maintained in LAM-SKOS-AP representation using VocBench3 system. This is depicted in upper part of Figure 4.4.

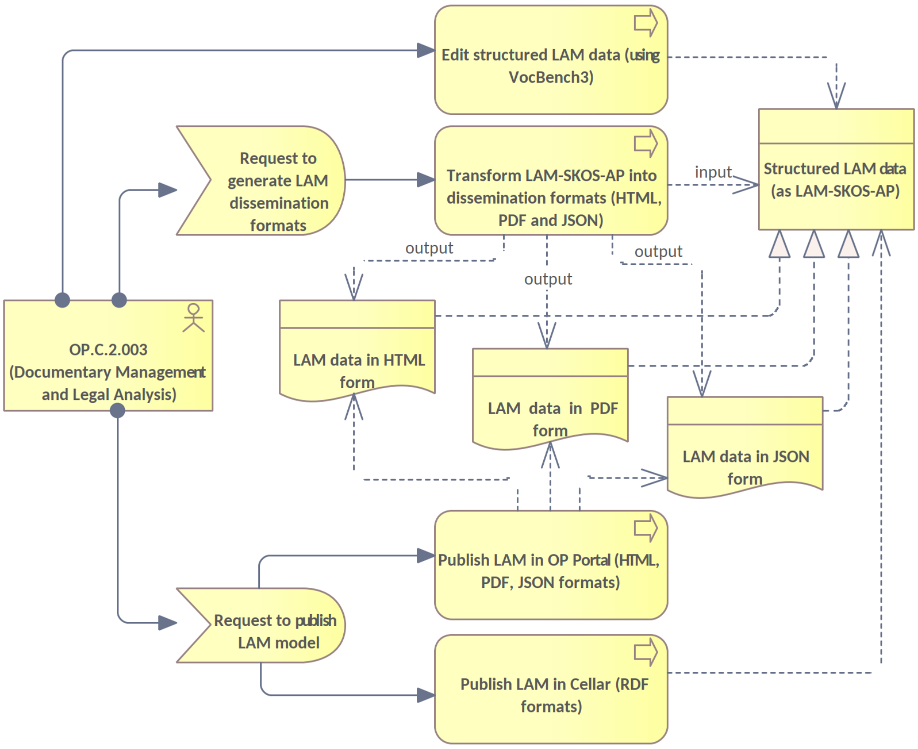


Figure 4.4: The publication architecture for LAM data

When a stable version of the data is achieved, it can be published for dissemination to the clients. To do so, a request to generate dissemination formats is issued by the LAM team, which will trigger the transformation process from LAM-SKOS-AP into HTML, PDF and JSON representation forms, which are human readable. This is depicted in the central part of Figure 4.4.

The human readable dissemination forms need to be validated and assessed for publication. After which the LAM team issues a new request to publish the data on the dissemination platforms. This triggers two processed: the first one is to publish the human readable representations into OP Portal and the other to publish the LAM data in LAM-SKOS-AP, the machine readable representation in Cellar.

The responsible agent for running these processes is the OP Metadata and reference data sector (OP.A.1.002). This shall be organised in an internal agreement at the OP. Moreover, the management of LAM data should be aligned with the practices and methods implemented by OP.A.1.002. These practices are internally known as asset lifecycle process and are briefly presented in the next section.

4.5 Asset lifecycle process

The previous sections presented the processes and business object specific to the context of LAM#2 project. These processes, however, need to be viewed from a data management perspective. This leads to the need to adopt a data management methodology.

This section presents the overview of the asset lifecycle process applicable to LAM data. This lifecycle process is adopted from the A1 unit, which is in the business of semantic asset management and publication. The diagram summarising the lifecycle process is provided in Figure 4.5.

The asset lifecycle process is organised in six stages: *evolution management*, *implementation*, *validation*, *release*, *publication* and *consumption*. Each of the stages represents a business sub-process accessing the LAM data object positioned centrally in Figure 4.5.

The *evolution management* stage deals with management change request cases. This stage also includes recording, analysis, negotiating back with the client and then finally deciding and planning the implementation of change request cases.

The *implementation* stage deals with performing the actual changes in the LAM data implementing one case at a time and verifying that the modifications reflect the original client request.

The *validation* stage follows the implementation and is performed by a different actor than the one performing the implementation process. Having the validation done by a second pair of eyes enforces the "four eyes principle" adopted by the OP in the proofreading and other authoring tasks.

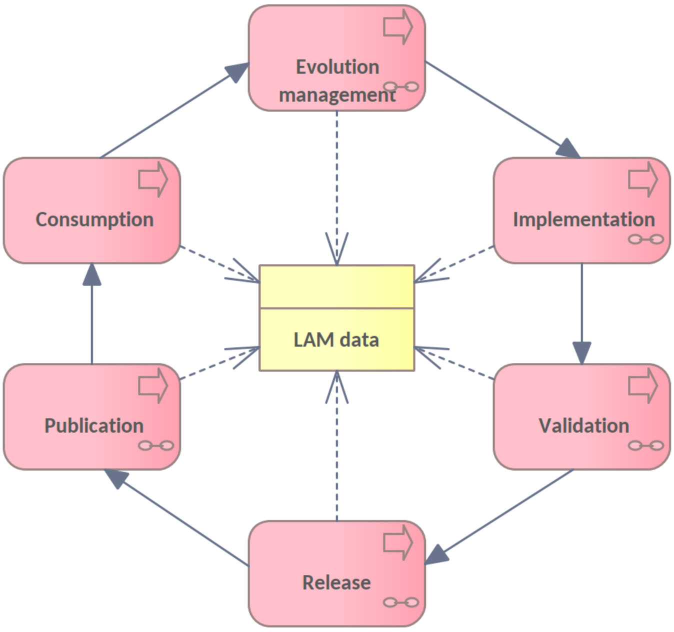


Figure 4.5: The baseline business architecture of LAM data

The *release* stage deals with all data transformations and preparation of artefacts to be disseminated and consumed by the final clients.

The *publication* stage deals with packaging the content and disseminating it to the selected data disseminators, OP Portal and Cellar being the main ones. During this stage a set of announcements and communications ensure that the main stakeholders and the broad public are aware of the published new version of the asset.

In the *consumption* stage only external actors are involved acting as clients. During this phase the data is accessed and used as necessary by each of the clients. While using the data assets, clients come up with additional requests for either changing content of the existent assets or adding and publishing new ones.

A more detailed description of the stages in the lifecycle process is provided in the document describing the asset publication workflow architecture [9] owned by A1. This architecture document was not yet released to the public so the A1 team shall be contacted for consultation.

4.6 Role allocation in the lifecycle process

This section brings together the actors and roles presented in Section 4.2 and the lifecycle process. In Figure 4.6 the actors (C2, A1, C1, A2 and others) are represented as swim-lines (see Figure 4.2 for details).

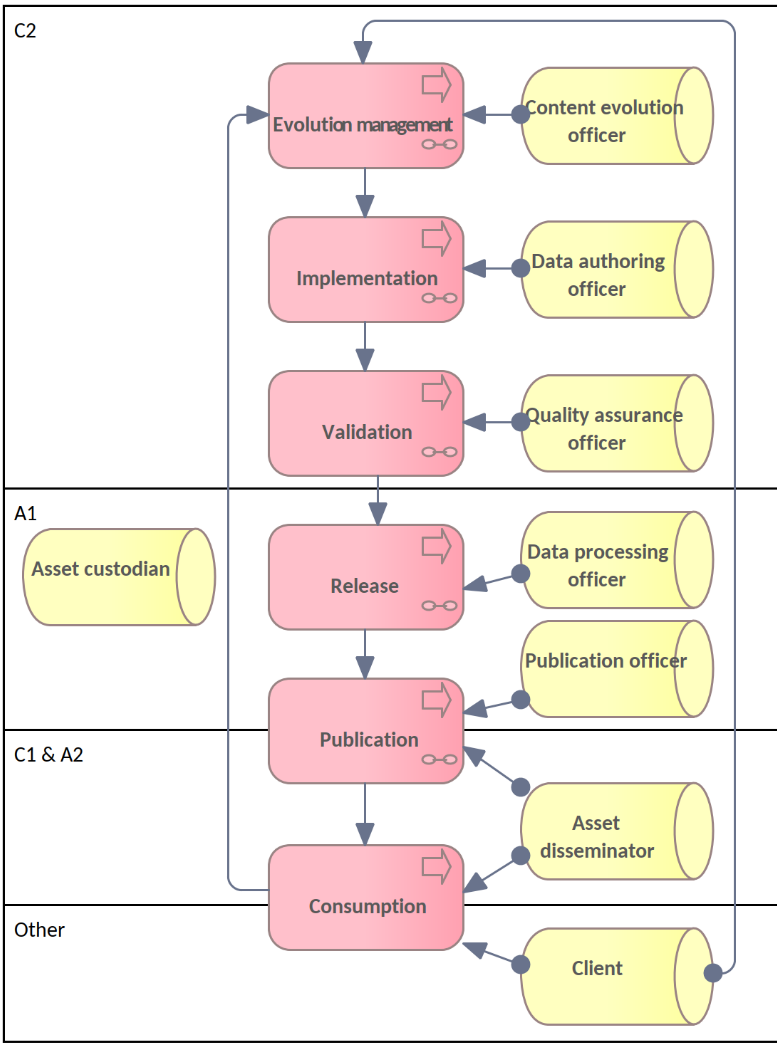


Figure 4.6: The lifecycle actors and roles

The process steps are organised top-down, and the roles are associated to each process step. The first three steps: evolution management, implementation and validation are executed by the LAM team in C2 taking consecutively the role of the content evaluation officer, data authoring officer and the quality assurance officer.

After validation, the release and publication steps are taken on by the A1 technical team who deals with technicalities of data transformation, packaging and transmission to the dissemination platforms. The A1 team takes the roles of data processing officer and that of data publication officer.

The A1 team also plays the role of asset custodian and is responsible for data content, context, and associated business rules, acting as a trustee on behalf of the asset owner (the C2 unit). This role is involved in overseeing the whole lifecycle process and ensuring its proper execution. The reason why this role is taken by A1 is because this unit provides the technical infrastructure and capabilities for editing, validating, transforming and publishing semantic assets.

The C1 and A2 units, where the OP Portal team and Cellar teams are situated, play their part partially in the publication and partially in the consumption stages of the lifecycle. They represent the dissemination platforms and, therefore, participate in the asset upload on the one hand and asset access by the clients on the other hand.

The last swim-line, at the bottom of the diagram, titled “Other”, includes all the stakeholders (see details in Section 4.2) that play the role of the client and participate in the consumption phase (and, of course, in the evolution management phase, if they send new requests for evolution).

Chapter 5

Application architecture

This chapter covers the application architecture. The essential services and application components that enact the business processes are presented here.

This architecture layer addressed the questions of how each application functions, through which services that functionality can be operationalised and which data objects are affected. In addition, cross layer view is offered in Section 5.3 presenting through which application capabilities the asset lifecycle is realised. But first, a prototypical application structure is presented in order to facilitate better understanding of the diagrams in this chapter.

5.1 Prototypical application structure

This section presents the application architecture from the solution architecture point of view. A generic solution architecture is depicted in Figure 5.1.

The application architecture presented covers the application as a "white box", its internal component structure, services and interfaces with adjacent applications. Typically the solutions architecture takes the technology aspects into account, accounting for parts of the infrastructure.

The central element of the application architecture is the *application service*, which represents application behaviour or functionality. The application services, from an inter-layer perspective, serve the processes in the business layer and provide support for their realisation.

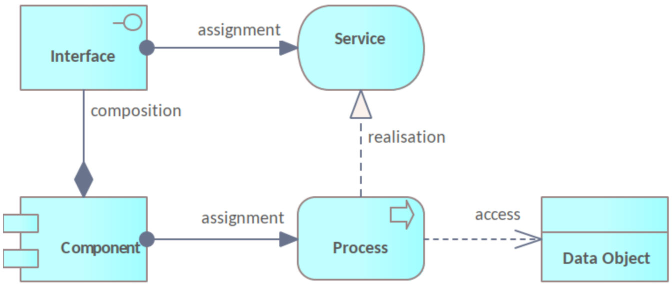


Figure 5.1: The prototypical application structure view

The application services are realised through application processes. The processes have application components assigned to them signifying their place of encapsulation. Application components are modular and replaceable blocks encapsulating implementation of application services and functionalities. In practice, for clarity, we take a shortcut, and say that the application services are realised through *application components* directly.

Components are said to expose interaction *interfaces* which are modelled, in ArchiMate, as proper parts of the components. The interfaces are assigned to services signifying how the latter are to be accessed and consumed.

Also, the components as well as the processes they encapsulate, access *data objects*, which are passive components of the application architecture.

The solution architecture presented in this section is an adaptation of the generic architecture. Here we focus on presenting what application services are used to support each business process. Moreover, we are interested in grasping the difference in the application layer between the current and new versions of the business processes.

To do so we split the application view diagrams into three vertical lanes. The left lane hosts the current version of the business process as well as the application services and components that are used to support it. In the right lane, we place the new business process and the new application services and components that will have to be adopted for the digital transformation. The middle lane hosts the services and components that are are currently employed and will be carried over into the new application architecture: they are common to both the current and new architectures.

Below we present an overview of the application architecture, in terms of services alone, depicting how the asset life-cycle stages are served.

5.2 LAM specific application architecture

This section presents the application architecture for services developed in the context of LAM#2 project. In Section 5.3 these applications will be placed in the context of the asset lifecycle process described in Section 4.5.

The project specific tools are *the transformation tool*, *the validation tool* and *the online (dissemination) tool*.

5.2.1 LAM transformation tool

The first of the three tools is the transformation tool from structured RDF data into human readable representations. This tool contributes directly to the goal of producing and dissemination the LAM data for end-user consumption on the OP Portal. This tool is used during the release phase of the lifecycle as will be explained in Section 5.3.4.

Figure 5.2 depicts the application architecture, with LAM transformer component in the centre of the diagram.

The main service this application exposes is the content transformation, positioned on the top of the diagram. This service is exposed through two interfaces: a web graphical user interface and an application programming interface. This service is realised by three application functionalities: generation of the HTML representation, generation of the PDF representation and generation of JSON indexes. The first two representations are meant to be distributed as such for the end-user consumption, while the indexes are meant to enable the search functionality provided by the OP Portal. The input taken by these functionalities shall be structured according to LAM-SKOS-AP[8].

In order to facilitate the transmission of the artefacts generated by the LAM transformer an additional service is foreseen, which aggregates the results of the transformation service into a ZIP archive. In the next section is described the LAM online tool, which ingests the ZIP archive and disseminates its content on a web interface.

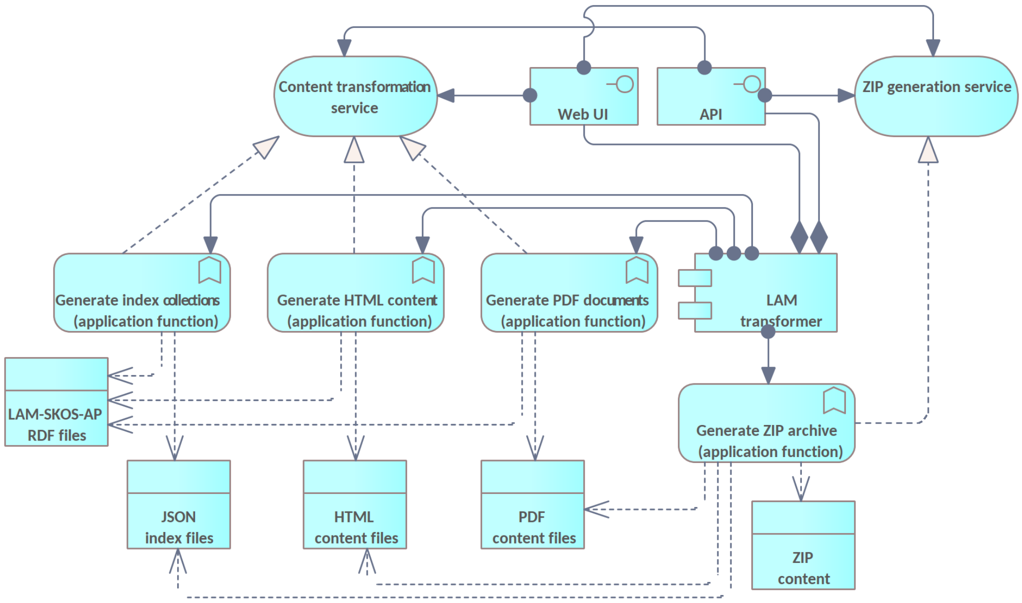


Figure 5.2: LAM transformation tool application architecture

5.2.2 LAM online tool

The LAM online tool is a mini-website hosted within the OP Portal ecosystem. Its main services are the content import, content consultation and content search. Figure 5.3 depicts the architecture supporting the content ingestion. This tool is used during the publication and consumption phase of the lifecycle as will be explained in Section 5.3.5.

The import service is exposed through a dedicated Axway folder where a scheduled job regularly checks for new content. As soon as new content is placed there the ingestion functionality starts. The expected input is the ZIP archive containing the PDF, HTML and JSON content. When this archive is unpacked each of these representations is treated accordingly for different purposes.

The content import service is realised through four application functions assigned to the LAM online tool component (see central area in Figure 5.3). The PDF files are ingested into a WebPub repository using Liferay framework. The HTML files are ingested into a Liferay CMS repository. The JSON index files are loaded into the Elasticsearch index. The import operation is triggered by running a scheduled task.

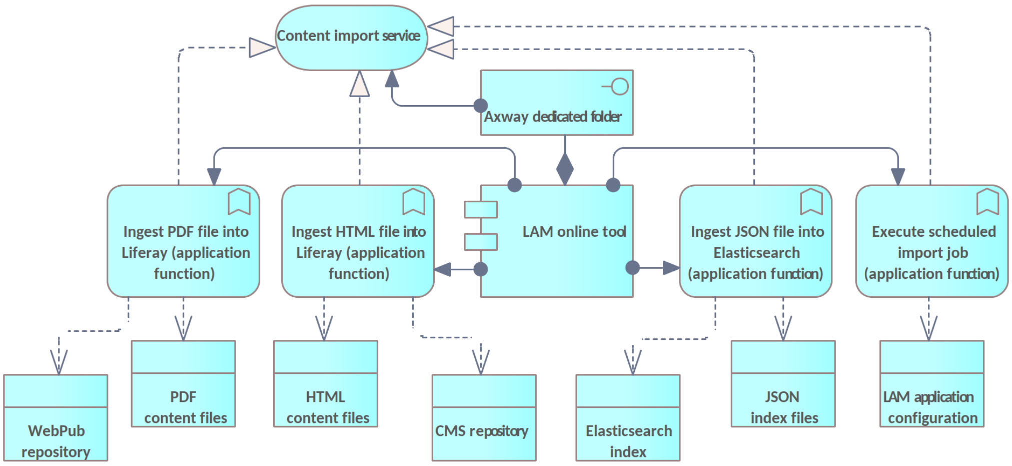


Figure 5.3: LAM online tool application architecture: content import service

The LAM online tool is conceived as a mini-website in the OP Portal. The main dissemination method is a Web user interface, exposing two services: the content consultations service and the content search service.

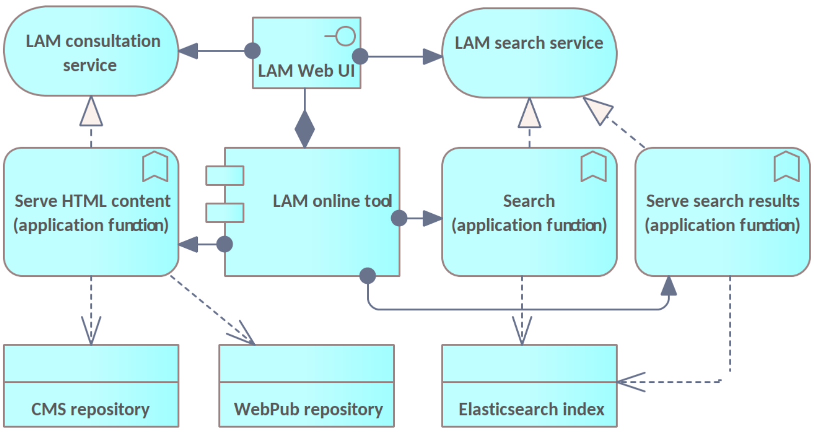


Figure 5.4: LAM online tool application architecture: content dissemination services

The search service is realised by two functionalities. The first is the Elasticsearch search functionality that takes a query and provides back the search results. The second is serving of the search results from the index and rendering them in a web interface.

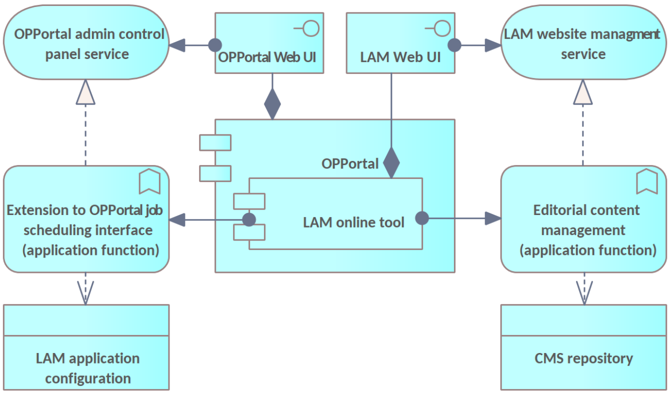


Figure 5.5: LAM online tool application architecture: administration services

The consultation service is realised by serving web pages from the Liferay CMS and the PDF documents from the WebPub repository. In addition, there is a set of Liferay editorial pages exposing custom content such as news, publication change notes, links to related resources etc. These pages are edited through LAM website management console offered by the Liferay framework. This service is depicted in Figure 5.5.

Another configuration that is available in the administration console is that of the scheduling task. This functionality constitutes, in fact, an extension to the scheduling functionality available in the OP Portal. For this reason the configuration is exposed through the OP Portal web interface.

5.2.3 LAM validation tool

In Section 4.4 was presented that structured LAM data is maintained using VocBench3 system. This system is versatile and well suited for the task, yet structural deviations from the designed LAM-SKOS-AP model are likely to happen. The motivation for using the LAM validation tool is that after the data is exported from VocBench3 and fed as input to the LAM transformation tool, it is imperative that the data is valid. Otherwise, the LAM transformation tool may render unexpected results or behave in an unpredictable manner.

In order to ensure that the transformation tool generates satisfiable results, the input data needs to be validated. This is a procedure preformed in the implementation phase, as will be shown in Section 5.3.2.

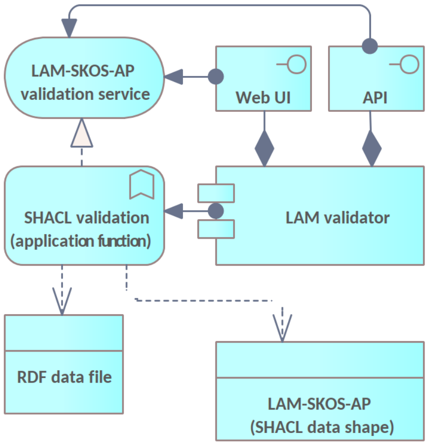


Figure 5.6: LAM validation tool application architecture

Figure 5.6 depicts the LAM validator application architecture. It exposes a single service through a Web user interface and an API. The service is realised through a SHACL [21] validation functionality with a predetermined set of SHACL shapes - the LAM-SKOS-AP[8] SHACL representation. Any RDF file that is provided to the service is tested against this preset data shape.

5.3 LAM lifecycle application architecture

In this section the lifecycle process described din Section 4.5 is connected to the application services and components realising them. Many of these services are available for usage, and describing them in detail is out of scope here. What is relevant to show here is how the lifecycle processes are served and where the LAM specific applications are involved.

5.3.1 Evolution management

In the first stage of the asset lifecycle, the application requirements are limited to client communication and the request documentation services as depicted in Figure 5.7. The email service is realised by the Outlook software. Issue management is realised by the Jira system.

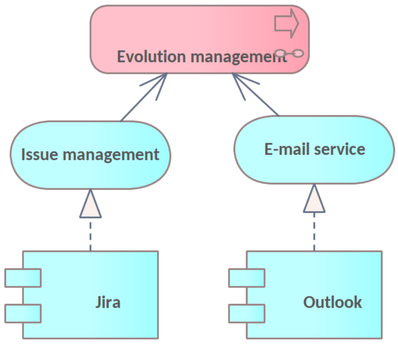


Figure 5.7: Application services and component that serve evolution management lifecycle stage

5.3.2 Implementation

The implementation stage is the first place where considerable number of application services are involved. This is depicted in Figure 5.8.

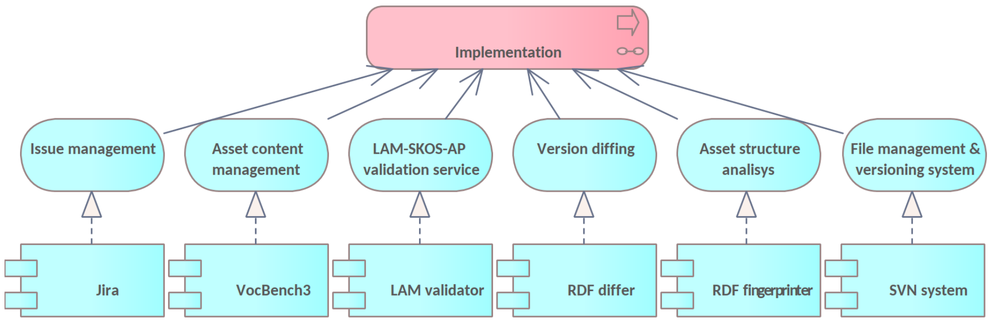


Figure 5.8: Application services and component that serve implementation lifecycle stage

The issue management, just like in the case of evolution management stage, is realised by Jira system. The content editing is realised through VocBench3 system, and once the editing operation is complete, the content is exported for further processing. First it is validated using the LAM validation tool. Then it is compared to a previous version to check whether the set of changed data corresponds to what is requested in the Jira ticket. The asset is also fingerprinted, to asses for possible structural deviations, not covered by the validation service. It is important to mention that the LAM asset is stored using a file management and versioning service realised by the SVN system. This file based repository is used as a medium to transit the assets through all the lifecycle phases, except consumption.

5.3.3 Validation

Validation is the next stage following the implementation. The services involved in this phase are issue management and file management services depicted in Figure 5.9.

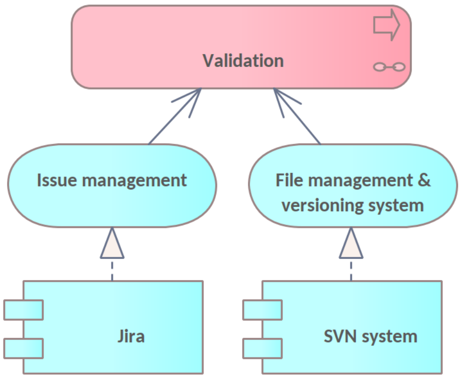


Figure 5.9: Application services and component that serve validation lifecycle stage

Generation for the validation artefacts is done during the implementation stage and stored in SVN. This stage is designed to allow a second person to check whether the implementation is done correctly following the "four eye principle". The validation office simply needs to access the validation artefacts from SVN and compare them to the original change request.

5.3.4 Release

Once the asset has been validated it is considered fit for publication. To do so it has to be transformed into all formats and representations necessary for the dissemination platforms and for final consumption by the end-users.

In this stage, depicted in Figure 5.10, a general data transformation service is involved, but also the LAM transformation tool that was described in Section 5.2.1. The input and output of the transformations are read from and written back into the SVN system. Optionally, a data validation service is used to perform a technical assessment before the data is sent to the dissemination systems.

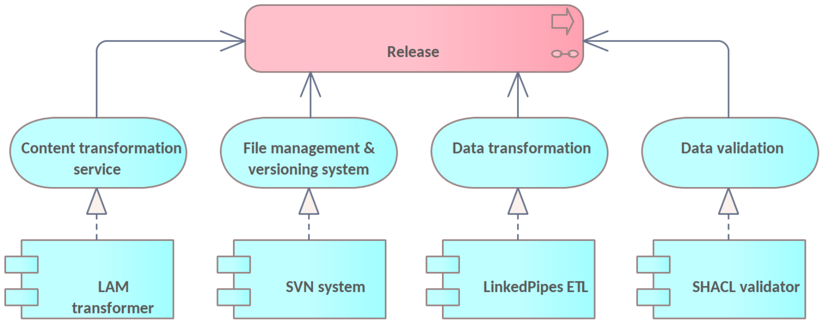


Figure 5.10: Application services and component that serve release lifecycle stage

5.3.5 Publication

The publication is the last stage in the asset life-cycle process. Its application architecture is depicted in Figure 5.2. In this phase, the assets are packaged and transmitted to the dissemination system.

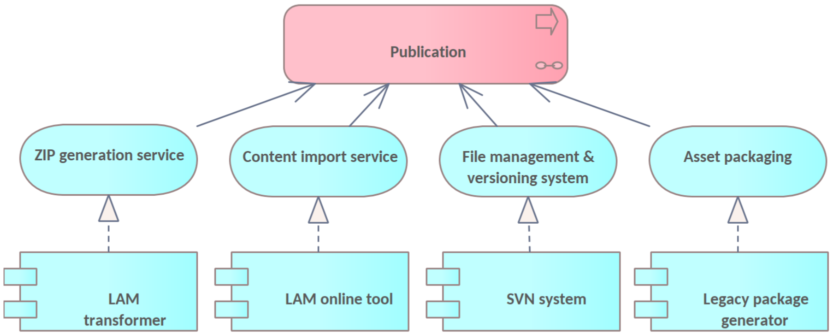


Figure 5.11: Application services and component that serve publication lifecycle stage

The packaging is realised through two services. First is the ZIP generation service, realised by the LAM transformation tool, which prepares the asset to be transmitted to the OP Portal system, where the LAM online tool is deployed.

Second is the METS[10] asset packaging service is realised by the legacy package generator which assembles in various ways the asset artefacts as necessary for partner dissemination systems and final consumers. This packaging is specific to the Cellar system where the LAM data is published in machine readable format. This brings us to the end of the application architecture description.

Chapter 6

Conclusions

This document presented the architectural stance for the LAM modelling initiative. Through the development of this architecture, we bring clarity of the project to the main stakeholders, what their interests and drivers are and what issues and solutions are associated to those motivations. We explicitly describe the internal processes, events and roles, answering questions concerning who shall do what and when. At a more specific level, valuable especially to the technical staff, is the application architecture, which answers the questions about what application service and capability supports which process in the LAM lifecycle.

This document constitutes a way to move forward with the digital transformation of the LAM team given its current interinstitutional context, management goals and demands from third parties. We aim to guide transitions in the asset source representation from the current unstructured and semi-structured sources towards structured representation of the source data.

The data quality is addressed in the current architecture through the introduction of manual and automatic verification and validation steps operating at both the form and meaning levels.

This architecture sets the reference points for establishing the necessary capabilities and technologies for LAM data maintenance, processing and dissemination. However, it is not addressing the entire digital transformation in order to prevent disruption in the current production system. Rather, it's a part of the application that is foreseen to evolve, specifically the part responsible for editing the asset content (VocBench3 is already operational). The rest can be addressed in a subsequent step as a natural follow-up.

This architecture organises the business processes aiming to optimise the workflow process reducing the bottlenecks and increasing the speed of the overall lifecycle process.

6.1 Summary

We presented in Section 1.1 the context of the current work given by the EU decisions and directives towards the semantic web technologies, open data and digital re-use of public sector information, along with implementation of a single digital gateway. The description of the state of play sets the baseline technical assessment which is extended by a recommendation of a joint trend towards the semantic web technologies and service oriented architecture.

The architecture proposed here consists of four layers: motivation in Section 3, business in Section 4, application in Section 5 and this section.

6.2 Final word

In this document, we propose the first step towards a modern enterprise-level application, that streamlines the process of asset publication lifecycle from both the LAM team and for external partners involved in the process.

The offered vision is a service oriented and semantically enriched system that operates in a cloud infrastructure and provides seamless experience to all involved parties in performing their duties and responsibilities in a coordinated asset lifecycle process. Such a system can constitute a cornerstone for management, publication and dissemination of public sector reference data bringing the single digital gateway one step closer to reality.

Bibliography

[1] P. Archer, L. Bargiotti, M. D. Keyzer, S. Goedertier, N. Loutas, and F. V. Geel. Report on high-value datasets from eu institutions. Deliverable SC17DI06692, European Commission, 2014.

[2] D. Beckett. RDF/xml syntax specification (revised). W3C recommendation, W3C, Feb. 2004. http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/.

[3] M. Brandner, M. Craes, F. Oellermann, and O. Zimmermann. Web services-oriented architecture in production in the finance industry. *Informatik-Spektrum*, 27(2):136–145, 2004.

[4] G. Carothers and E. Prud'hommeaux. RDF 1.1 turtle. W3C recommendation, W3C, Feb. 2014. http://www.w3.org/TR/2014/REC-turtle-20140225/.

[5] K. Channabasavaiah, K. Holley, and E. Tuggle. Migrating to a service-oriented architecture. *IBM DeveloperWorks*, 16:727–728, 2003.

[6] E. Costetchi. The structure of Excel workbook for bootrstraping the legal analysis methodology descriptions. Recommendation, Publications Office of the European Union, October 2019.

[7] E. Costetchi. Preliminary requirements specification for the legal analysis methodology models. Requirements, Publications Office of the European Union, October 2019.

[8] E. Costetchi. LAM-SKOS-AP: an application profile for the legal analysis methodology description. Recommendation, Publications Office of the European Union, October 2019.

[9] E. Costetchi. Asset publication lifecycle architecture. Recommendation, Publications Office of the European Union, September 2020.

[10] E. Costetchi and J. Delahousse. METADATA ENCODING AND TRANSMISSION STANDARD. Primer and reference manual, Digital Library Federation, September 2010.

[11] Council of the European Union. Council conclusions inviting the introduction of the European Legislation Identifier (ELI). *OJ*, C 325:3–11, 2012. CELEX:52012XG1026(01).

[12] Council of the European Union. Council conclusions inviting the introduction of the European Legislation Identifier (ELI). *OJ*, C 441:8–12, 2017. CELEX:52017XG1222(02).

[13] European Parliament and the Council. Decision No 456/2005/EC of the European Parliament and of the Council of 9 March 2005 establishing a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable (Text with EEA relevance). *OJ*, L 79:1–8, 2005. CELEX:32005D0456.

[14] European Parliament and the Council. Directive 2013/37/EU of the European Parliament and of the Council of 26 June 2013 amending Directive 2003/98/EC on the re-use of public sector information Text with EEA relevance. *OJ*, L 175: 1–8, 2013. CELEX:32013L0037.

[15] European Parliament and the Council. Decision (EU) 2015/2240 of the European Parliament and of the Council of 25 November 2015 establishing a programme on interoperability solutions and common frameworks for European public administrations, businesses and citizens (ISA2 programme) as a means for modernising the public sector (Text with EEA relevance). *OJ*, L 318:1–16, 2015. CELEX:32015D2240.

[16] European Parliament and the Council. Regulation (EU) 2018/1724 of the European Parliament and of the Council of 2 October 2018 establishing a single digital gateway to provide access to information, to procedures and to assistance and problem-solving services and amending Regulation (EU) No 1024/2012 (Text with EEA relevance.) . *OJ*, L 295:1–38, 2018. CELEX:32018R1724.

[17] European Parliament and the Council. Directive (EU) 2019/1024 of the European Parliament and of the Council of 20 June 2019 on open data and the re-use of public sector information. *OJ*, L 172:56–83, 2019. CELEX:32019L1024.

[18] E. Francesconi, M. W. Küster, P. Gratz, and S. Thelen. The ontology-based approach of the publications office of the eu for document accessibility and open data services. In *International Conference on Electronic Government and the Information Systems Perspective*, pages 29–39. Springer, 2015.

[19] E. Francesconi, M. W. Küster, P. Gratz, and S. Thelen. Semantic modeling of the eu multilingual resources. In *ICAIL Multilingual Workshop on AI & Law Research*, page 13, 2015.

[20] O. Group et al. Soa reference architecture. *Technical Standard, available at http://www. opengroup. org/soa/source-book/soa_refarch/, accessed in*, pages 12–12, 2016.

[21] H. Knublauch and D. Kontokostas. Shapes constraint language (SHACL). W3C recommendation, W3C, July 2017. https://www.w3.org/TR/2017/REC-shacl-20170720/.

[22] J. KUBA and N. GEORGOSOPOULOS. Online tool for legal analysis methodology: User requirements specification. User requirements, Publications Office of the European Union, September 2020.

[23] Microsoft. Excel 2013, 2013. URL `https://www.office.com/`.

[24] B. Parsia, P. Patel-Schneider, and B. Motik. OWL 2 web ontology language structural specification and functional-style syntax (second edition). W3C recommendation, W3C, Dec. 2012. http://www.w3.org/TR/2012/REC-owl2-syntax-20121211/.

[25] P. Patel-Schneider, B. Parsia, and B. Motik. OWL 2 web ontology language structural specification and functional-style syntax. W3C recommendation, W3C, Oct. 2009. http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/.

[26] D. Peterson, A. Malhotra, S. Gao, M. Sperberg-McQueen, P. V. Biron, and H. Thompson. W3C xml schema definition language (XSD) 1.1 part 2: Datatypes. W3C recommendation, W3C, Apr. 2012. http://www.w3.org/TR/2012/REC-xmlschema11-2-20120405/.

[27] Publications Office of the European Union. EUR-Lex legal analysis methodology. Recommendation, Publications Office of the European Union, June 2017.

[28] G. Schreiber and F. Gandon. RDF 1.1 XML syntax. W3C recommendation, W3C, Feb. 2014. http://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/.

[29] Sparx Systems. Enterprise Architect, 2000. URL `https://sparxsystems.com/products/ea/`.

[30] M. Sporny, D. Longley, G. Kellogg, M. Lanthaler, and N. Lindström. Json-ld 1.0. *W3C recommendation*, 16:41, 2014.

[31] M. Sporny, D. Longley, G. Kellogg, M. Lanthaler, P.-A. Champin, and N. Lindström. JSON-LD 1.1–a JSON-based serialization for Linked Data. W3C recommendation, W3C, 2020. https://www.w3.org/TR/json-ld11/.

[32] A. Stellato, A. Turbati, M. Fiorelli, T. Lorenzetti, E. Costetchi, C. Laaboudi, W. Van Gemert, and J. Keizer. Towards vocbench 3: pushing collaborative development of thesauri and ontologies further beyond. In *17th European Networked Knowledge Organization Systems Workshop, NKOS 2017*, volume 1937, pages 39–52. CEUR-WS, 2017.

[33] A. Stellato, M. Fiorelli, A. Turbati, T. Lorenzetti, W. van Gemert, D. Dechandon, C. Laaboudi-Spoiden, A. Gerencsér, A. Waniart, E. Costetchi, et al. Vocbench 3: A collaborative semantic web editor for ontologies, thesauri and lexicons. *Semantic Web*, (Preprint):1–27, 2020.

[34] The Open Group. *The TOGAF Standard, Version 9.2*. The Open Group, 2018.

[35] The Open Group. *ArchiMate 3.1 Specification*. The Open Group, 2019.

[36] M. Van Opijnen. European case law identifier: indispensable asset for legal information retrieval. 2011.

[37] M. van Opijnen, G. Peruginelli, E. Kefali, and M. Palmirani. On-line publication of court decisions in the eu: Report of the policy group of the project ‘building on the european case law identifier’. *Available at SSRN 3088495*, 2017.

[38] A. T. Velte, T. J. Velte, and R. Elsenpeter. Cloud computing: A practical approach, 2019.

[39] D. Wood, R. Cyganiak, and M. Lanthaler. RDF 1.1 concepts and abstract syntax. W3C recommendation, W3C, Feb. 2014. http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/.