

Initiative for Modeling the Legal Analysis Methodology

Service Architecture

Disclaimer

The views expressed in this report are purely those of the Author(s) and may not, in any circumstances, be interpreted as stating an official position of the European Union. The European Union does not guarantee the accuracy of the information included in this study, nor does it accept any responsibility for any use thereof. Reference herein to any specific products, specifications, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favouring by the European Union.

Document metadata

Reference	WP 1.2: Service 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	19 Feb 2021
Suggested readers	system architects, business analysts, system administrators

Abstract

This document provides a working definition of the architectural stance and design decisions that are to be adopted for the Legal Analysis Model maintenance and dissemination life-cycle 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 the European Union, the public sector is one of the most open-data intensive sectors. The re-use of open data can contribute, for example, to the growth of the European economy, the development of artificial intelligence and overcoming societal challenges.

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. The vocabularies and semantics that they are introducing, together with the technologies that they are using, all differ. These differences hamper data interoperability and thus its reuse by them or by the wider public. This creates the need for a common data standard for publishing public reference data and models, hence allowing data from different sources to be easily accessed, linked, and consequently reused.

The Public Sector Information (PSI) directive across the EU calls for open, unobstructed access to public data in order to improve transparency and to boost innovation via the reuse of public data. The reference data maintained and published by the Publications Office of the European Union (OP) has been identified as data with a high-reuse potential. 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 assets which are vital in the context of inter-institutional information exchange, access, retrievability and re-use of information. With regards to reference data, the PO 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.

This document provides a working definition of the architectural stance and design decisions that are to be adopted for the asset publication life-cycle process. This process is materialised as the publication workflow and is currently employed by the Standardisation Unit (SU) at the Publications Office of the European Union (OP).

Contents

1	Architecture building blocks	7
1.1	Methodology	7
1.2	Architecture views	8
1.3	ArchiMate elements	9
2	Motivation architecture	14
2.1	Prototypical motivation structure	14
2.2	OP legal analysis team	16
2.3	Different OP services and EU institutions	17
2.4	LAM contractors	18
2.5	Publications Office of the European Union	20
3	Business architecture	22
3.1	Prototypical business structure	22
3.2	Actors and roles	24
3.3	Maintenance of semi-structured LAM data	26
3.4	Maintenance and publication of structured LAM data	28
3.5	Asset lifecycle process	29
3.6	Role allocation in the lifecycle process	31
4	Application architecture	33
4.1	Prototypical application structure	33
4.2	LAM specific application architecture	35
4.2.1	LAM transformation tool	35
4.2.2	LAM online tool	36
4.2.3	LAM validation tool	38
4.3	LAM lifecycle application architecture	39

4.3.1	Evolution management	39
4.3.2	Implementation	39
4.3.3	Validation	41
4.3.4	Release	41
4.3.5	Publication	42

Chapter 1

Architecture building blocks

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

1.1 Methodology

In this document we take an enterprise architecture perspective and aim to provide several architecture views (see Section 1.2) which are necessary and sufficient to describe the asset lifecycle process.

In developing this architecture, we are in part using the TOGAF [13] 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 completely, we took inspiration from parts of it which were applicable to the goals of this architecture.

For architecture representation, we adopt ArchiMate language [14], which 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.

We have conducted a series of interview with the SU management, the technical team and the business teams. In developing the motivation architecture, we entirely rely on the input from the SU management, presented in Section 2.

The business use cases represent the knowledge elicited from the technical team and the business teams and are presented in Section ??.

The other layers of this architecture, are a gradual fleshing out of the use cases, and rely on the author experience of working a few years side by side with SU business (documentalists) and technical teams. This experience results in the knowledge of the applications and technical peculiarities of the SU.

The corresponding ArchiMate diagrams were modelled and designed using Enterprise Architect Tool [10]. Finally this report was written covering the overall architecture.

1.2 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 [14].

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 [14].

Viewpoints are designed for the purpose of communicating certain aspects and layers of an architecture. In this document we address the motivation view (Section 2), the business view (Section 3), the application view (Section 4) and the technology view (Section ??).

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

1.3 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 1.1: Overview of the relevant motivation elements [14]



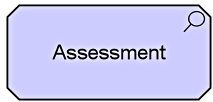







Element	Definition	Notation
Stakeholder	Represents the role of an individual, team, or organisation (or classes thereof) that represents their interests in the effects of the architecture.	
Driver	Represents an external or internal condition that motivates an organisation to define its goals and implement 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 organization and its stakeholders.	

Table 1.2: Overview of the relevant business layer elements [14]

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.	 

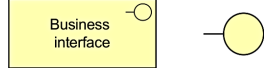




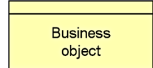

<i>Table 1.2 continued from previous page</i>		
Business interface	Represents a point of access where a business service is made available to the environment.	
Business process	Represents a sequence of business behaviours that achieves a specific result such as a defined set of products or business services.	
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 1.3: Overview of the relevant application layer elements [14]

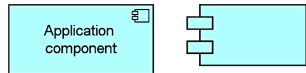
Element	Definition	Notation
Application component	Represents an encapsulation of application functionality aligned to implementation structure, which is modular and replaceable.	

Table 1.3 continued from previous page











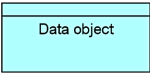
Application interface	Represents a point of access where application services are made available to a user, another application component, or a node.		
Application function	Represents automated behaviour that can be performed by an application component.		
Application process	Represents a sequence of 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 1.4: Overview of the relevant technology layer elements [14]

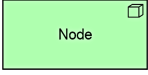

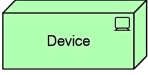

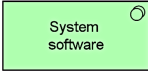



Element	Definition	Notation	
Node	Represents a computational or physical resource that hosts, manipulates, or interacts with other computational or physical resources.		
Device	Represents a physical IT resource upon which system software and artefacts may be stored or deployed for execution.		
System 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.		

Table 1.4 continued from previous page







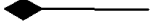
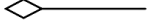
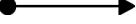




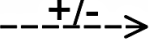
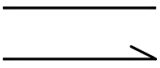
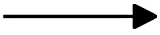
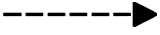
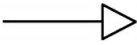


Communication network	Represents a set of structures that connects nodes for transmission, routing, and reception of data.		
Technology service	Represents an explicitly defined exposed technology behaviour.		
Artefact	Represents a piece of data that is used or produced in a software development process, or by deployment and operation of an IT system.		

Table 1.5: Overview of the ArchiMate relationships [14]

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.	

<i>Table 1.5 continued from previous page</i>		
Association	Represents an unspecified relationship, or one that is not represented by another ArchiMate relationship.	
Dynamic Relationships		
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

Chapter 2

Motivation architecture

This chapter presents the motivation and goal structure of the Documentary Management and Legal Analysis sector (OP.C.2.003) scoped to this project. This motivation structure is also presented in the context of the Publications Office, which scopes and documents the rationale of the initiative for modelling legal analysis methodology.

We do not aim for an in depth coverage of the motivation architecture here. So it cannot be considered as a fully fledged decision-making tool for the management. What we rather aim at is accounting for the context, stakeholders and their drivers and interests.

This motivation view helps address questions on why a demand 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.

2.1 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 2.1 depicts the organisation of the motivation architecture. The structure starts at the top with enumerating the stakeholders, who can be in-

dividuals, teams or organisations that represent their interests in the effects of the architecture [14].

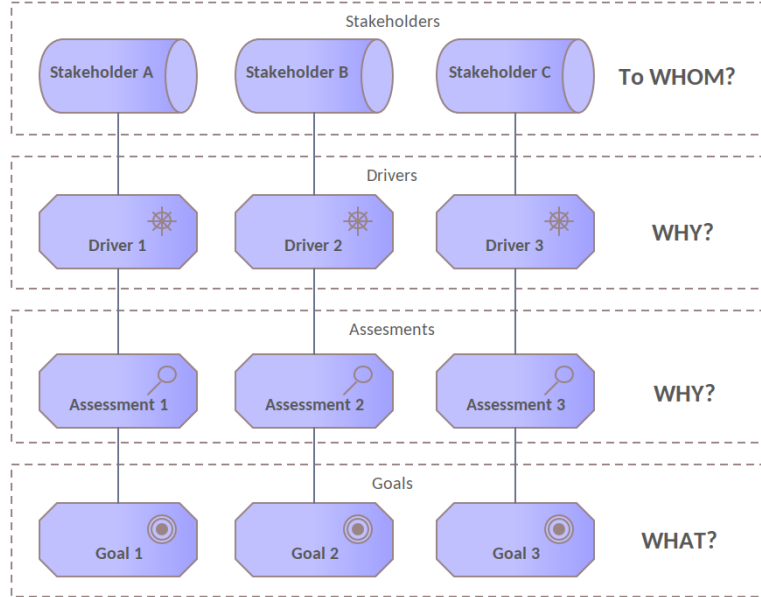


Figure 2.1: The layered motivation structure

Stakeholders have associated interests, concerns or drivers, which represent internal or external conditions that motivate an organisation to define goals [14].

Assessments represent results of analysis of the state of affairs with respect to a driver. They reveal strengths and weaknesses, opportunities or threats to an area of interest [14].

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 [14].

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.

2.2 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. One particular feature that is of special importance is to also link other various datasets on which LAM relies, such as Common Data Model (CDM) [5, 6], authority tables published at the EU Vocabularies¹, 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 2.2.

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 is 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).

¹<https://op.europa.eu/en/web/eu-vocabularies>

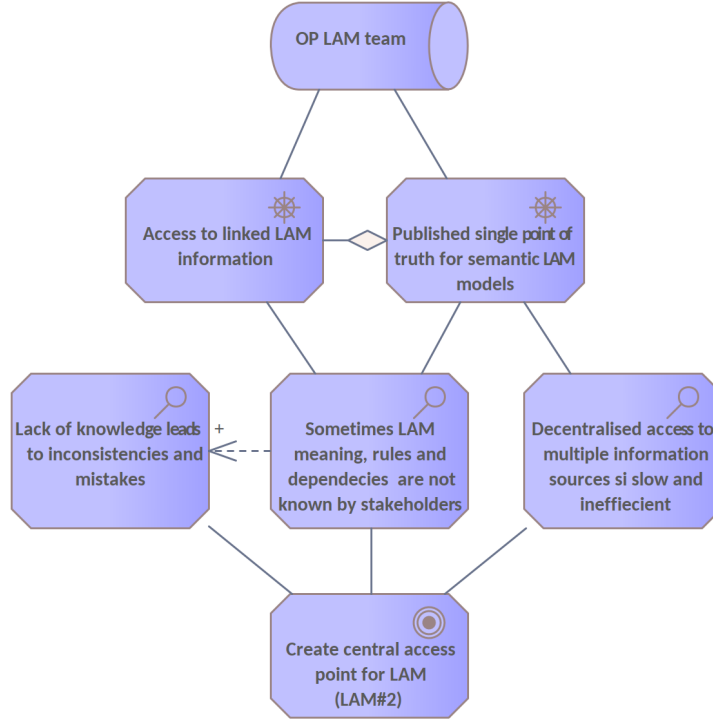


Figure 2.2: Motivation structure of the OP legal analysis team

2.3 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 2.3.

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 2.3. 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.

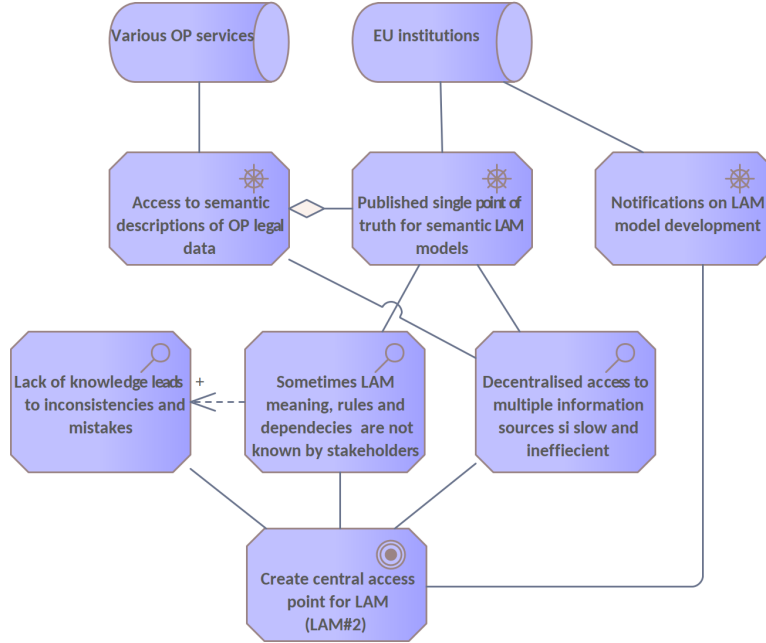


Figure 2.3: Motivation structure of different OP services

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.

2.4 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 2.4.

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, no validation or consistency checking

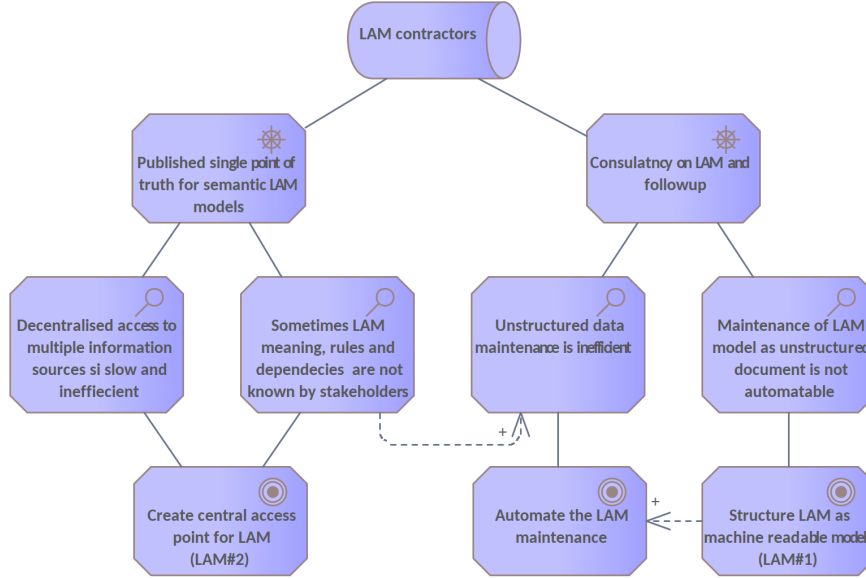


Figure 2.4: Motivation structure of LAM contractors

is possible with such representation. 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)².

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.

In the left side of Figure 2.4, the driver, assessments and goal are repeated from the sections above as they are central to the current project and therefore shared by all of the stakeholders.

²LAM#1 deliverables are available in the following GitHub repository <https://github.com/eu-vocabularies/lam4vb3>

2.5 Publications Office of the European Union

At 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 2.5 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 are represented in machine readable format and even less is based on semantic models.

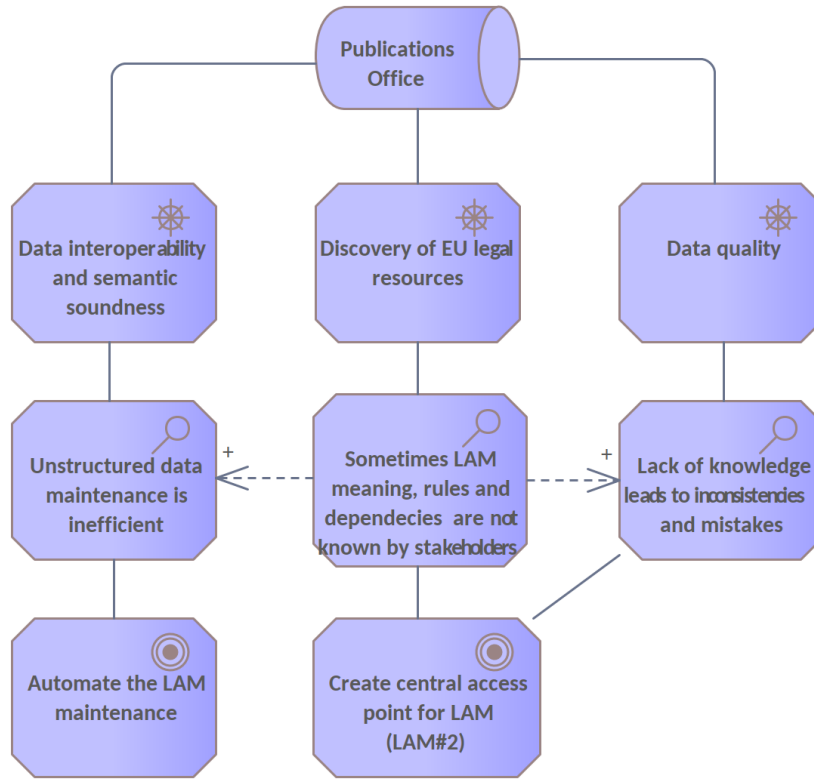


Figure 2.5: Motivation structure of the Publications Office of the European Union

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

Finally, a driver which is a 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 inability to easily find and access LAM meaning, rules and dependencies. Therefore the the dissemination of LAM data shall be don in such a way that the relations to external data sources are presented in an intuitive manner and making such links easy to navigate. Moreover, an inventory of links to the most used resources shall be disseminate with the LAM data.

Chapter 3

Business architecture

This chapter addresses the business architecture aiming to describe the internal processes, events and roles answering questions concerning who shall do what and when.

This chapter first presents the baseline business architecture, as it is currently established and, then, presents a new (target) business architecture, as it is envisaged in the light of the digital transformation moving towards goals identified in the motivation structure (Section 2). The new business processes (see Section ??) are aligned with use case descriptions from Section ?. The latter are derived from materials describing the current workflow and interviews with the SU technical and business team members.

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

3.1 Prototypical business structure

Following the metaphor of layers presented in the motivation view (see Section 2.1), the organisation of business structure is also explained in terms of layers. Figure 3.1 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

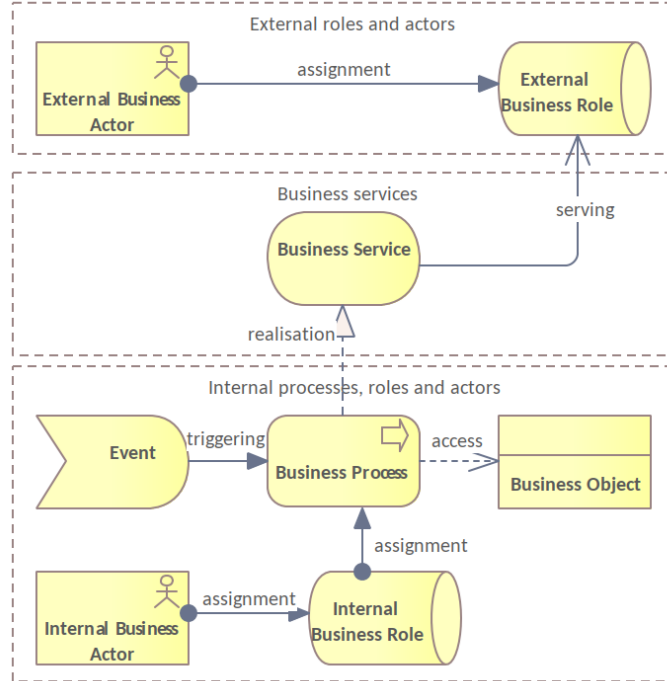


Figure 3.1: The prototypical business structure view

skills and responsibilities for performing specific behaviour, and to which an actor can be assigned [14].

The middle layer represents the *services* that are offered by the organisation to external players. A business service represents explicitly-defined behaviour that a business role, business actor or business collaboration exposes to its environment [14].

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.

3.2 Actors and roles

This section describes identified actors and roles relevant to the context of LAM data dissemination. Figure 3.2 depicts their relations.

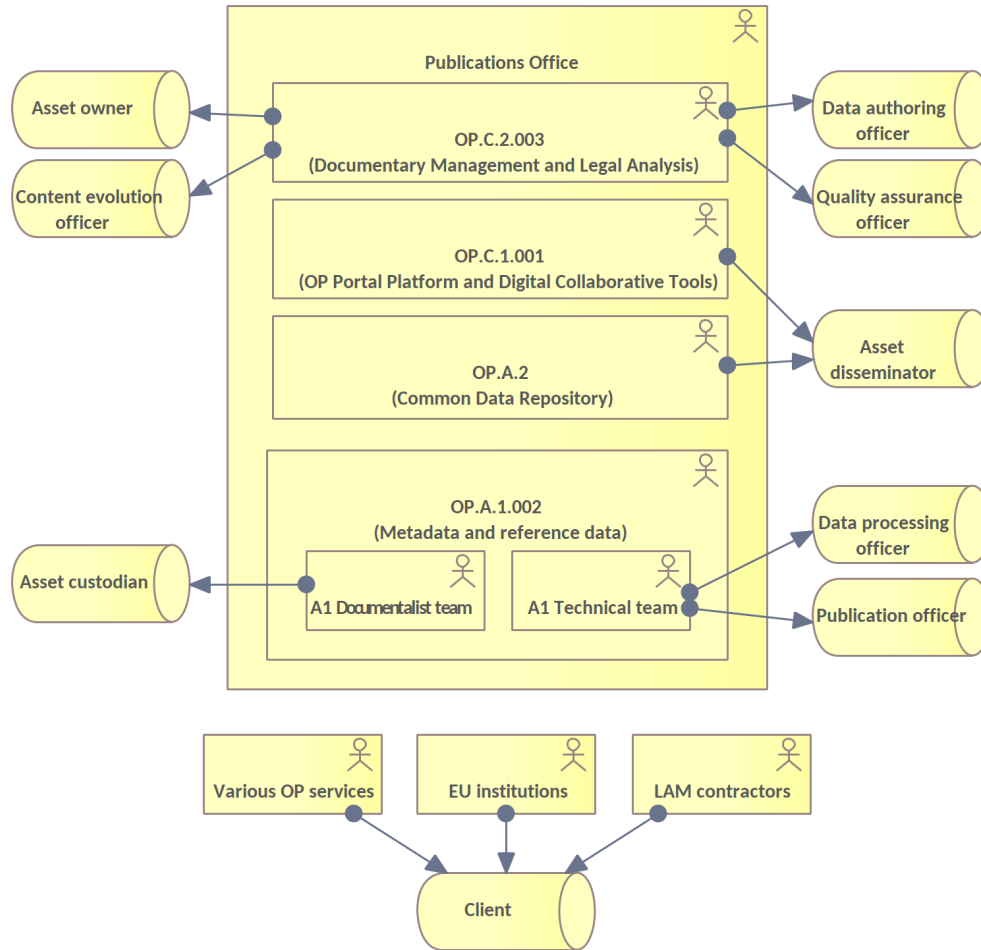


Figure 3.2: The prototypical business structure view

The OP is a stakeholder in the project but does not have a direct role in the business processes, rather it serves as 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 asset manager or this role will be transferred to the Metadata and Reference Data sector, who is also the data custodian.

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.

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.

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.

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 documentalists team* to act as asset custodian and correspondingly **the technical team** taking the role of data processing and publication officer.

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.

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.

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 [5] also plays the role of asset disseminator because the machine readable representation of the LAM data is published in the Cellar system.

Asset disseminator the role providing with reliable data 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 2) as stakeholders. From business point of view these stakeholders are agents playing the role of a client.

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

3.3 Maintenance of semi-structured LAM data

In a previous project dealing with creating a model for the LAM data (internally referred to as LAM# [2]) 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 [2] was completed three new capabilities were added to the LAM team: (a) management of LAM data in semi-structured representation (in an Excel workbook [8]) thanks to a specification on how to structure the workbook [1], (b) the possibility to transform the LAM data into RDF representation following

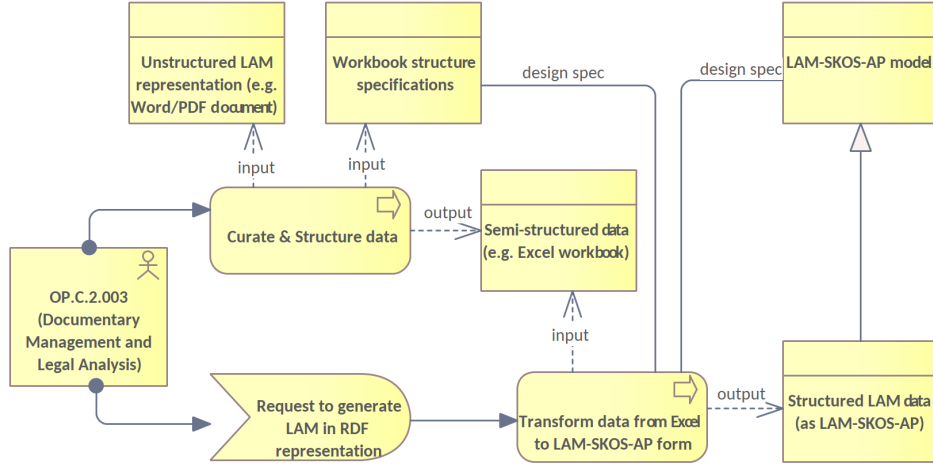


Figure 3.3: The maintenance of LAM data in the semi-structured form

the LAM-SKOS-AP model specifications [3] and (c) uploading and editing the LAM data in VocBench3 system [11, 12]. Figure 3.3 depicts this situation.

LAM team (the actor on the left side of the diagram) over time performs the curation and structuring of the LAM data producing the Excel workbook (semi-structured representation). As input to this process, the team relies on the existent Word documents describing EurLex LAM structure in human readable form [9] and second input is the specifications for structuring the Excel workbook [1].

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[11] and away from Excel[8]. 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 output is the LAM data instantiating LAM-SKOS-AP model [3]. The instantiation relationship is depicted in Figure 3.3 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.

3.4 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 3.4.

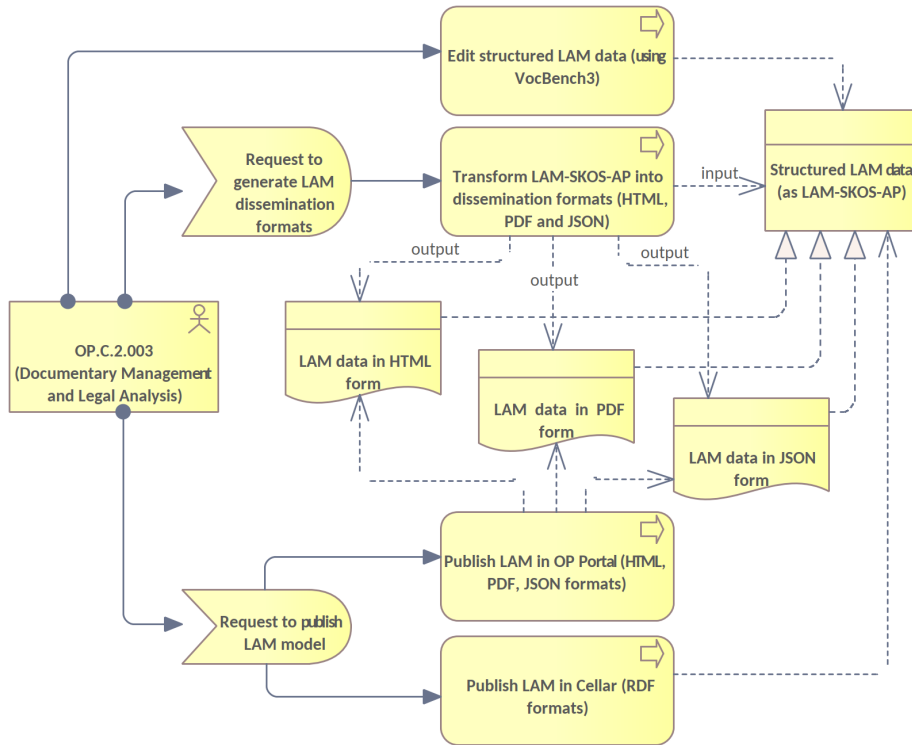


Figure 3.4: The publication architecture for LAM data

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 3.4.

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 processes: one, 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.

3.5 Asset lifecycle process

In the previous sections were 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 A1 unit, which is in the business of semantic asset management and publication. The diagram summarising the lifecycle process is provided in Figure 3.5.

The asset life-cycle 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 3.5.

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 the and scheduling for implementation of the 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.

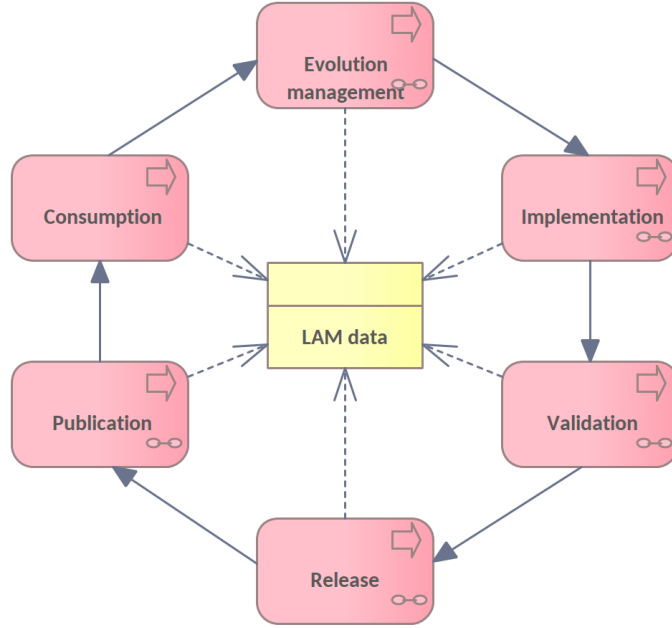


Figure 3.5: The baseline business architecture of LAM data

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 life-cycle process is provided in the document describing the asset publication workflow architecture [4] owned by A1. This architecture document was not yet released to public so the A1 team shall be contacted for consultation.

3.6 Role allocation in the lifecycle process

This section brings together the actors and roles presented in Section 3.2 and the lifecycle process. In Figure 3.6 the actors (C2, A1, C1, A2 and others) are represented as swim-lanes (see Figure 3.2 for details).

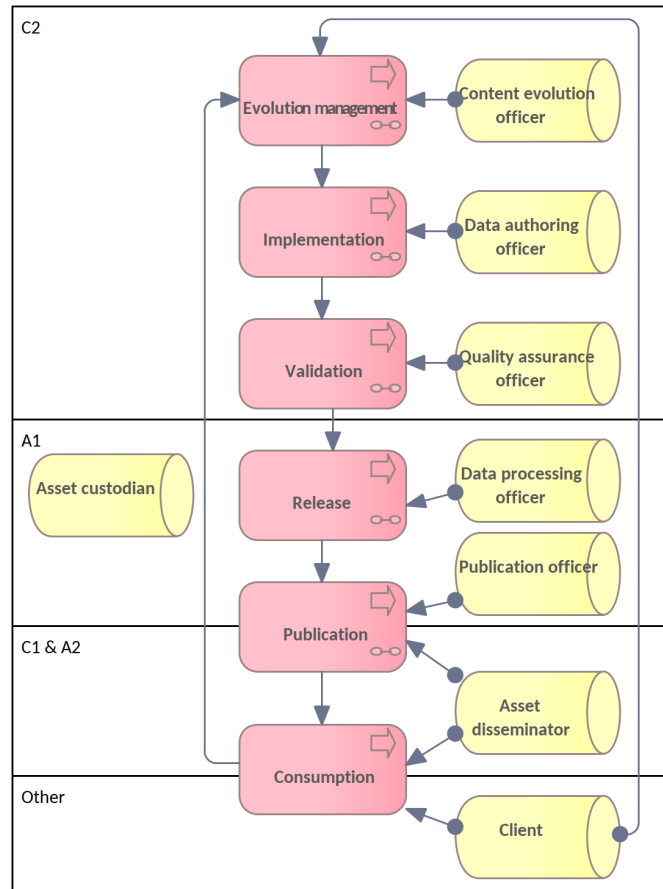


Figure 3.6: The lifecycle actors and roles

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 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 3.2) 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 4

Application architecture

This chapter covers the application architecture. We present here the essential services and application components that enact the current and the new life-cycle processes.

4.1 Prototypical application structure

This section presents the application architecture from the solution architecture point of view. A generic solution architecture is depicted in Figure 4.1.

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.

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,

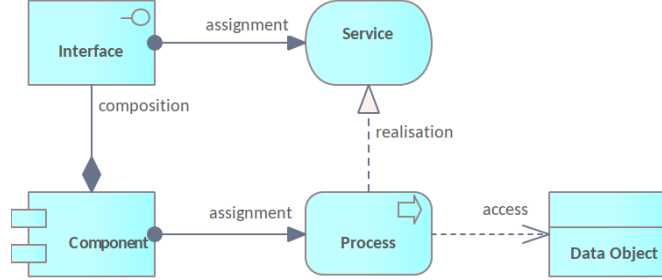


Figure 4.1: The prototypical application structure view

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, components, as well as 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 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.

4.2 LAM specific application architecture

This section presents the application architecture for services developed in the context of LAM#2 project. In Section 4.3 these applications will be placed into the context of asset lifecycle process described in Section 3.5.

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

4.2.1 LAM transformation tool

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 4.3.4.

Figure 4.2 depicts the application architecture, with LAM transformer component in the centre of the diagram.

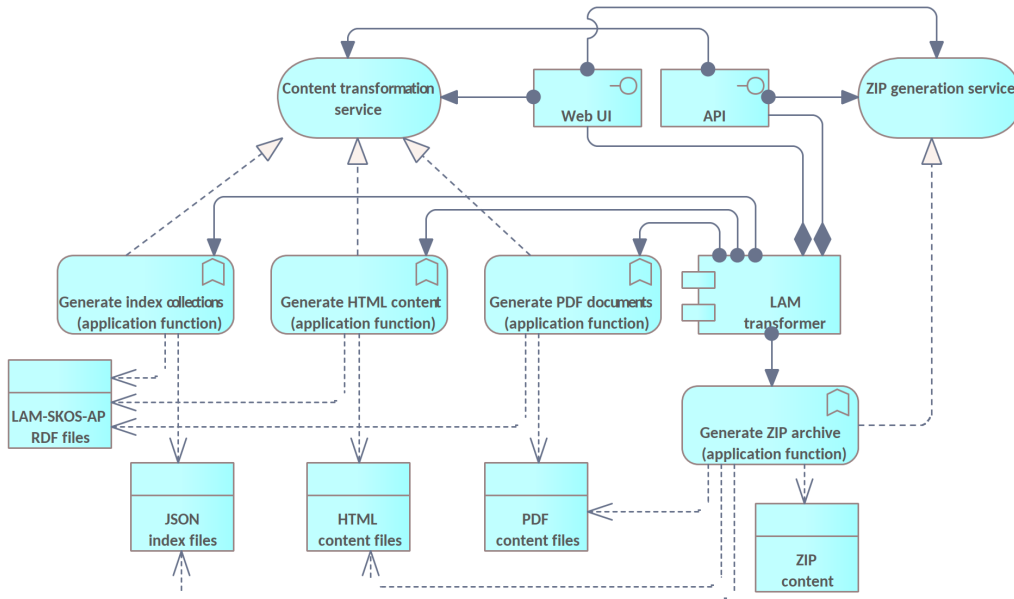


Figure 4.2: LAM transformation tool application architecture

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[3].

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.

4.2.2 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 4.3 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 4.3.5.

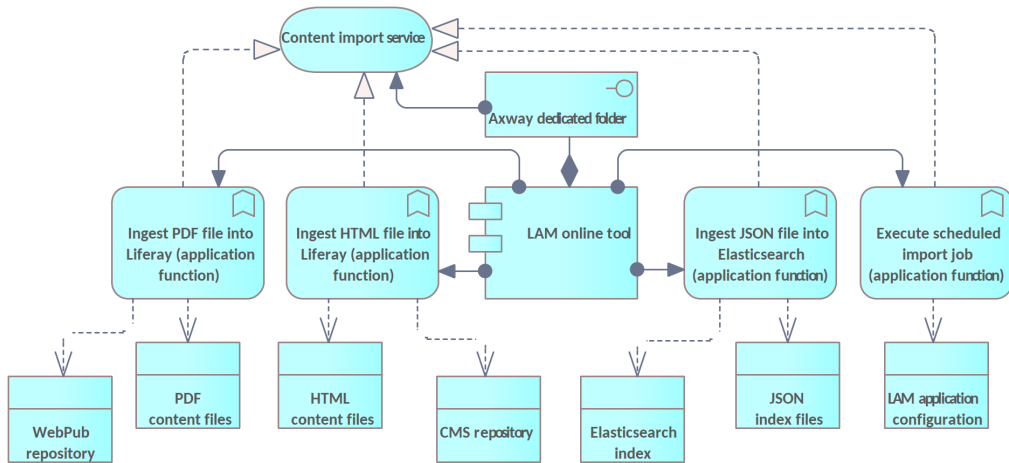


Figure 4.3: LAM online tool application architecture: content import service

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 4.3). 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.

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.

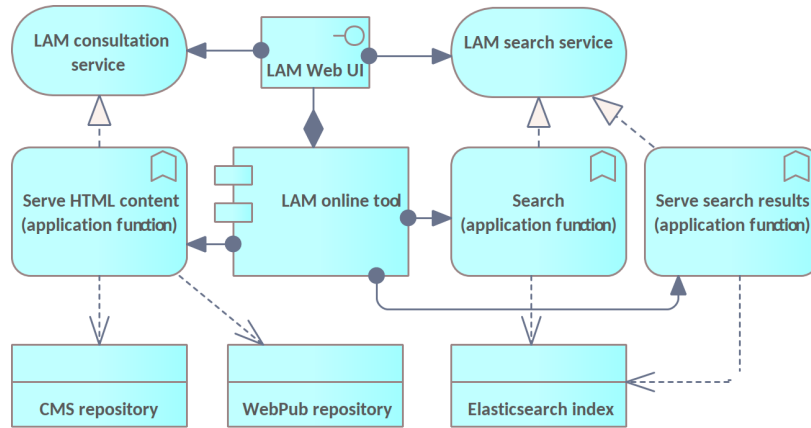


Figure 4.4: LAM online tool application architecture: content dissemination services

The search service is realised by Elasticsearch search functionality and serving the search results from the index and rendering them in a web interface.

The consultation service is realised by serving web pages from Liferay CMS and the PDF documents from 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 Liferay framework. This service is depicted in Figure 4.5.

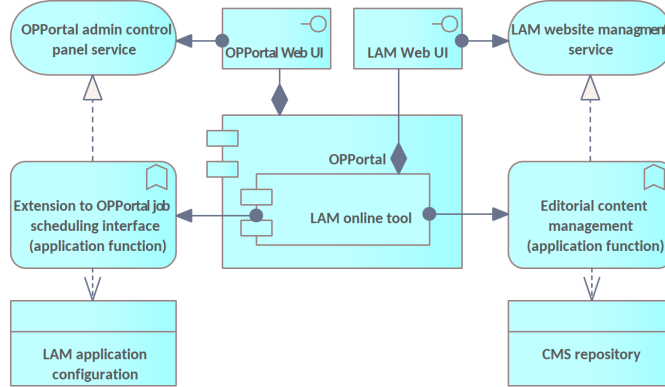


Figure 4.5: LAM online tool application architecture: administration services

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

4.2.3 LAM validation tool

In Section 3.4 was presented that structured LAM data are maintained using VocBench3 system. This system is versatile and well suited for the task, yet structural deviations from the designed model, LAM-SKOS-AP, are likely to happen. The motivation for using the LAM validation tool is that after the data are exported from VocBench3 and fed as input to the LAM transformation tool, it is imperative that the data are 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 4.3.2.

Figure 4.6 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 [7] validation functionality with a predetermined set of SHACL shapes - the LAM-SKOS-AP[3] SHACL representation. Any RDF file that is provided to the service is tested against this preset data shape.

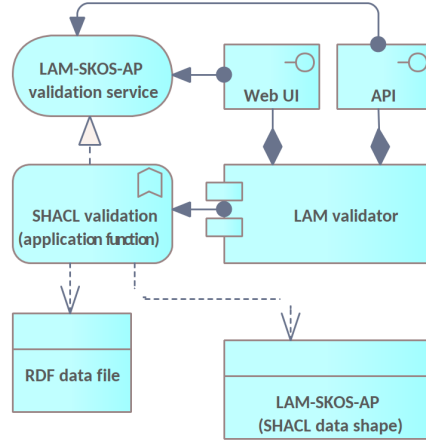


Figure 4.6: LAM validation tool application architecture

4.3 LAM lifecycle application architecture

In this section the lifecycle process describe din Section 3.5 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 how the lifecycle processes are served and where the LAM specific applications are involved.

4.3.1 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 4.7. The email service is realised by the Outlook software. Issue management is realised by the Jira system.

4.3.2 Implementation

The implementation stage is the first place where considerable number of application services are involved. This is depicted in Figure 4.8.

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,

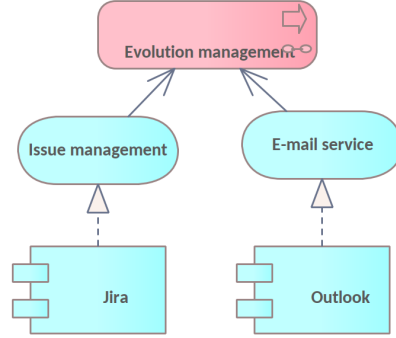


Figure 4.7: Application services and component that serve evolution management lifecycle stage

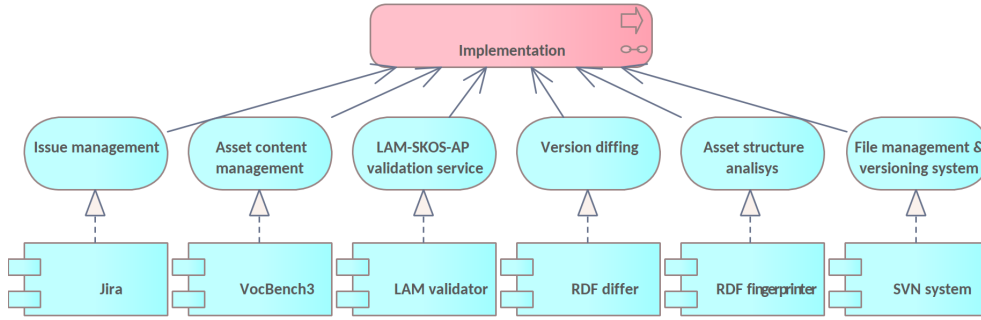


Figure 4.8: Application services and component that serve implementation lifecycle stage

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 changes done corresponds to what is requested in the Jira ticket. The asset is also fingerprinted, to assess 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.

4.3.3 Validation

The stages following the implementation is validation. The services involved in this phase are issue management and file management service depicted in Figure 4.9.

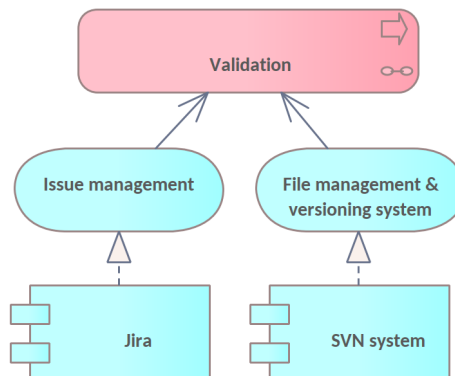


Figure 4.9: Application services and component that serve validation lifecycle stage

Generation of 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 to the original change request.

4.3.4 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 4.10, general data transformation service is involved, but also LAM transformation tool that was described in Section 4.2.1. 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 being sent to the dissemination systems.

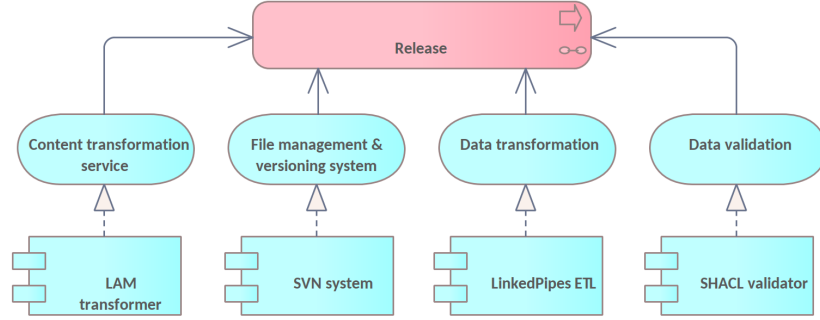


Figure 4.10: Application services and component that serve release lifecycle stage

4.3.5 Publication

The publication stage is the last one in the asset life-cycle process. Its application architecture is depicted in Figure 4.2. In this phase, the assets are packaged and transmitted to the dissemination system.

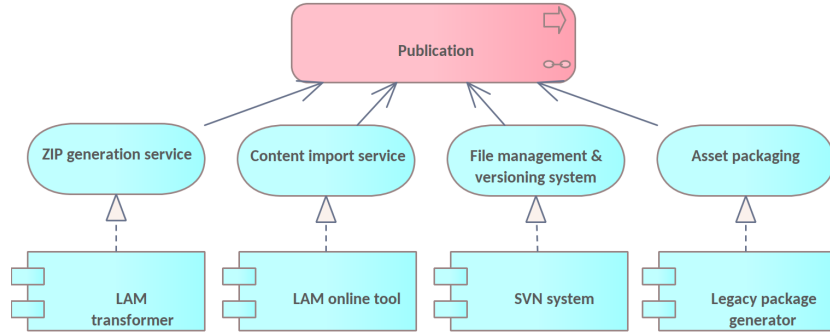


Figure 4.11: Application services and component that serve publication lifecycle stage

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

The METS asset packaging service is realised by a legacy package generator hosted by the A2 unit. This packaging is accepted by the Cellar system where the LAM data are published in machine readable format.

Bibliography

- [1] E. Costetchi. The structure of Excel workbook for bootstrapping the legal analysis methodology descriptions. Recommendation, Publications Office of the European Union, October 2019.
- [2] E. Costetchi. Preliminary requirements specification for the legal analysis methodology models. Requirements, Publications Office of the European Union, October 2019.
- [3] E. Costetchi. LAM-SKOS-AP: an application profile for the legal analysis methodology description. Recommendation, Publications Office of the European Union, October 2019.
- [4] E. Costetchi. Asset publication lifecycle architecture. Recommendation, Publications Office of the European Union, September 2020.
- [5] 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.
- [6] 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.
- [7] H. Knublauch and D. Kontokostas. Shapes constraint language (SHACL). W3C recommendation, W3C, July 2017. <https://www.w3.org/TR/2017/REC-shacl-20170720/>.
- [8] Microsoft. Excel 2013, 2013. URL <https://www.office.com/>.

- [9] Publications Office of the European Union. EUR-Lex legal analysis methodology. Recommendation, Publications Office of the European Union, June 2017.
- [10] Sparx Systems. Enterprise Architect, 2000. URL <https://sparxsystems.com/products/ea/>.
- [11] 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.
- [12] 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.
- [13] The Open Group. *The TOGAF Standard, Version 9.2*. The Open Group, 2018.
- [14] The Open Group. *ArchiMate 3.1 Specification*. The Open Group, 2019.