Universiteit Leiden

Master ICT in Business and the Public Sector

A Perceived Architecture Description Development Process

Name:	Arjan Visser
Student ID:	s1698087
Date:	April 22, 2021
Specialisation:	ICT in Business
1st supervisor:	Drs. Bas Kruiswijk
2nd supervisor:	Dr. Werner Heijstek

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

A Perceived Architecture Description Development Process

INTRODUCTION: Organisations grow beyond solely describing the current state of the architecture to just IT stakeholders. Although many works exist on architectural description development, many works reason from a purely theoretical perspective. In this theoretical process, one elicits concerns from a stakeholder about their system of interest. Subsequently, one frames these concerns in viewpoints and models the viewpoints through a set of views. The sum of all views about a system is its architecture description. However, empirically observed and reported architectural development processes using views and viewpoints remains scarce. There is a gap in the literature regarding implementation and empirical development processes of enterprise architecture descriptions. In this research, we fill this gap with a case study in which we follow the architectural development cycle of a reference architecture description within the educational sector..

OBJECTIVES: This study has two main objectives. First and foremost is to describe an observed enterprise architecture description development process and compare this development process to the theory. Secondly, is to add a real-world case for enterprise architecture development to the literature.

METHODS: This study was an observational detailed single case study with participation by the researcher in the public sector. This study is further supported by interviews, questionnaires and project's documents analysis. Furthermore, the single case study takes place in the Dutch education sector. In order to compare to said theory, first, this theory must be stated. Consequently, the best practices and theoretical approaches to enterprise architecture description development are stated. Additional reference points are given by comparing the results to existing empirical studies into enterprise architecture description development.

RESULTS: The process described is about a knowledge-diverse project team of eighteen people who created a new enterprise reference architecture description for the Dutch vocational education. The observed development project can be divided into four phases: the conception phase, the initiation phase, the development phase, and the governance phase. In the conception phase, they create a project initiation document in which project principles, budget, goals and organisation is established. After which, a project framework is created in the initiation phase. The development phase starts with gathering information and the distillation of the information into architectural building blocks. Subsequently, two development iterations can be distinguished within the development phase: creating main viewpoints and views and creating stakeholder group-specific viewpoints and views. Main viewpoints and views are created through iterative cycles of analysis, sketch, description formalisation and acceptance testing. We find that the used development iteration for creating main viewpoints and views bears a resemblance to the process proposed by the open group in TOGAF and, to a lesser extent, the development method described by DODAF. Additionally, we find both discrepancies and conformations of general best practices and additional lessons learned using the case study.

CONCLUSION: An empirical enterprise architecture development process is added to the literature, which bears a resemblance to TOGAF. This development processes used a project-specific framework based on existing enterprise frameworks and the industry. Moreover, it shows the benefit of creating an enterprise architecture description in the public sector by using other public organisations as information sources. Consequently, their own development time is minimalised.
The process is characterised by freedom for the working groups and management by exception nature and a pragmatic approach. This pragmatic approach is contrary to many theoretical approaches, which requires intensive documentation and commitment to formalities. This observed process also suggests the usefulness of a diverse team in terms of knowledge fields and business or ICT affiliation, bearing in mind the common ground requirement. Additionally, it also shows that an architectural development description can be achieved with an agile mindset.

CCS Concepts: • **Applied computing → Enterprise architectures**; *Enterprise architecture frameworks*; *Enterprise architecture management*; *Business-IT alignment*.

Additional Key Words and Phrases: Enterprise Architecture, Architectural Viewpoints and Views, Architecture Description Development, Empirical Process

1 Introduction

Architecture is the inherent fundamental organisation of a system often described, managed, and maintained by the process of architecting. Additionally, architecture is used to refer to the discipline responsible for executing the architecting process [9] [32] [43]. Organisations increasingly invest in the maturity of their architecting [60], such that the organisation grows beyond solely describing the current state of the architecture to just IT stakeholders [57] [81]. This investment is made on the promise, that this would lead to better IT-&-business alignment [21] [30] [87], create a shared context for corporate decision making [13] [50] [6], improved performance [43], a possibility to identify weaknesses [57] [17] and a way to optimise resource allocation [57] [21]. Other benefits include improved management of fragmented enterprises, creating a reliable infrastructure [6], helping change management, and aid strategic planning [57] [6] [21]. However, a mere five per cent will see the fruits of their investment, as organisations trip over the many hurdles and challenges in the architecting process [33]. To help guide them in this process, organisations often use the best practice of selecting an architectural framework, method or standard. [4]

Many different architectural frameworks and methods exist, such as TOGAF [66], DYA [16], GEA [82], NATO–AFv4 [10] and doDAF [59], to name a few. Today, most organisations use a mix of several architectural frameworks and methods [26]. To clarify an architectural framework is a structure, conventions and best practices for architecting [51]. Moreover, a framework delivers an initial setup for architectural descriptions established within a specific business domain or stakeholder community [26]. On the other hand, one defines architectural methods as 'a structured collection of techniques & process steps for creating and maintaining architecture products and processes [49]. While each framework is different overall, most share the same core, wherein one recognises stakeholders, systems of interest, viewpoints and views [32] [58]. Stakeholders are persons who have an interest in a system, making it a system of interest. Concerns are made up of the stakeholders' interests. These concerns can be requirements, goals, assumptions, expectations or key performance indicators. Furthermore, the viewpoint frames the concerns in the architectural description. The viewpoints govern one or more views, which form the architectural description with architecture description elements, relations and patterns [32].

In the same way to frameworks, most iterative methods are not the same but share a common denominator. In this common denominator, one begins with the definition of the problem statement, the identification of the stakeholders and their concerns, the identification of possible solutions, the identification & creation of viewpoints, followed by the development of the views and models, acceptance testing and the implementation of governance & architecture change management [10].

Although there are shared aspects in the various frameworks and methods, the implementation and the connection between these aspects differ. In the scientific field, the same applies, although many works use the term view or viewpoint, the used working definition differs from one work to the other. A possible explanation is that enterprise architecture is a young field that mostly builds upon best practices, rather than a complete scientific underpinning [77]. As a result, there are several issues in science related to enterprise architecture, such as the aforementioned language confusion and the lack of empirical research topics [81] [21] [74] [85].

Literature works on frameworks, methods, and the adoption of enterprise architecture (EA) are plentiful; however, there is an empirical gap in literature where an EA development process is observed and reported [21] [81]. Moreover, literature analysis suggests that most EA research focuses on the modelling, while EA processes are underrepresented. There are five architecture processes to be distinguished, the enabling processes, the governance process, the management process, the evaluation process and the description process [10]. In the latter, the architectural description is developed amongst other activities. The practical application of the architectural development process is seldom reported in the scientific literature [7] [21] [74] [85].

To form a better view of the architecture description development process, the question is raised: "How is EA developed in practice? ". However, there is reason to believe that the public sector's enterprise architecture processes are substantially different from the private sector and should be researched separately [21]. The public sector institutions know little to no competition, strife towards the same goal (the betterment of its citizens) and share knowledge across public institutions freely and willingly for the sake of this goal [53]. This cooperation and knowledge sharing influences the EA processes. This free sharing of knowledge is seldom found in a highly competitive field, where knowledge is often a competitive advantage [20]. Furthermore, public sector institutions are differently structured than their private sector counterparts [17]. As a result, the research question becomes:

"How is EA developed in practice in the public sector? "

This paper investigates the problem statement in the Dutch educational sector by conducting a case study with participation and relates the found empirical process to the theory. The education sector is a domain within the public sector, which shares the same properties as stated earlier. First, it actively shares information on enterprise architecture through public publication. Secondly, it shares a collective goal, namely educating the people. Lastly, the organisation is differently structured, having more external relations, such as the school inspection, the government, the ministry of education & culture, and other educational institutions.

This research helps fill the gap between theory and practice and the underrepresentation of EA development processes in scientific literature by participating in a real–world case. By doing so, theoretical works can improve their practicality; others can use the lessons learned to improve their own EA development process; and a new case is introduced, which can be used in future works.

(1) How are the stakeholders identified?
(2) How does one elicit the concerns from the identified stakeholders?
(3) How are the viewpoints used?
(4) How are the views created?

The exploratory research investigates architectural description development in a real–world setting by reporting on the process undertaken in an architecture development project. First, a theoretical framework is created to define concepts, commonly used methodologies & frameworks, and what has already been found in relevant empirical research. Subsequently, a case study is introduced, described, and its process is reported. The findings are then compared to existing literature, followed up by a conclusion and discussion.

2 Theoretical framework

In this section, we explain the theoretical ground of this research. Recall that several frameworks, methods, and standards describe and aid the EA development cycle. Either by assisting in modelling the EA or the EA development process. In this research, only a few are named, TOGAF [66], NATO–AFv4 [10], doDAF [59] and ISO 42010–42030 [2]. Note that the ISO standards are not mentioned separately, since NATO–AFv4 has adopted the ISO standards. First, some concepts are defined about architecting.

2.1 ENTERPRISE ARCHITECTURE AND ARCHITECTURAL DESCRIPTIONS

Enterprise architecture is the enterprise's long–term vision, structure, methods, activities or motivations revolving around the enterprise architectural descriptions, such as its governance, maintenance and development [73] [81]. An architectural description (AD) is the description of architecture in the past, present or future employing one or more models within a set of views. Each model consists out of two or more architectural description elements, which have relations and correspondence which each other [32]. Hilliard, states that "'*M is a model of S if M can be used to answer questions about S.*" Additionally, a model kind describes the conventions for a type of modelling.

One of the benefits of enterprise architecture includes its scale since it describes the entire enterprise. As a result, the EA functions as a shared reference and perspective, improving communication and decision making between different parties within the enterprise. Furthermore, EA is a starting point for architecting specific enterprise portions, reducing individual projects' starting costs when working under architecture [13].

Note that the ontology and taxonomy within the discipline of architecture have no final consensus and many different definitions exist for the various concepts within architecture [81]. For example, Lankhorst [49] uses the following definition for EA: 'a coherent whole of principles, methods, and models that are used in the design and realisation of an enterprise's Organisational structure, business processes, information systems, and infrastructure', while Armour [43] distinguishes between EA, as the architectural description with a vision, and enterprise architecting, as the methods, processes, tools and structures needed to design and realise EA. Moreover, armour notices that EA architecting should begin by stating clear definitions used within the enterprise to prevent language confusion [13]. In this research, most definitions regarding ontology and taxonomy are lent from ISO–42010 [2].

STAKEHOLDERS, CONCERNS AND VIEWPOINTS Stakeholders are groups, individuals or an organisation, with an interest in a specific system [32]. Note that a system can be many things such as processes, organisations, information systems, data or infrastructure. A stakeholder's interest in a system is defined by a non-empty set of concerns, where one or more stakeholders may share the same concern. These stakeholders' concerns will influence the architectural descriptions throughout its life cycle, and such a concern can take different forms. It could be but is not limited to, a system requirement, a user story, a goal or an expectation. How these concerns are framed, communicated, and related to other descriptions, is defined in the viewpoint. The view, on the other hand, is the set of models of a single viewpoint. Note that stakeholders are diverse. Therefore, the same set of concerns may be answered with a diverse group of views. We refer to this as architectural interchange [29] [55].

VIEWS AND VIEWPOINTS A viewpoint is a product establishing conventions for construction, use and interpretation of views to frame specific concerns. [32] [26] Moreover, the definition states that a viewpoint defines a model kind, frames concerns and their corresponding stakeholders, the information sources used and how the view should be interpreted. Note that some extend the viewpoint by actively documenting the correspondence and correspondence rules [31]. A viewpoint thus prescribes a view of a system of interest. A view, on the other hand, is the implementation of the viewpoint. It expresses the architecture of a system through architectural models. The plural is chosen deliberately, as one view is a set of one or more models pertaining to one and only one viewpoint [26], contrary to some existing works on the discipline of enterprise architecture, which state that a view comprises a single model [41]. The language with which the enterprise description is modelled is called an architecture description language (ADL). Furthermore, an ADL can be a formal language, such as Archimate3.0 or UML, or an enterprise's informal language [49].

It did not take long before one had the idea of interchangeable viewpoints, or rather object–orientated viewpoints, such as Kruchten 4 + 1. That is modular viewpoints, viewpoints which can be used independently of its context. Imagine an information system landscape as in a typical organisation A, how many information systems has A in common with the next organisation B? There are quite a few common systems in practice, which have the same purpose and the same type of users. Therefore, one could also have a set of interchangeable, reusable viewpoints. These modular viewpoints, one could use across industry domains. On the other hand, more industry–specific viewpoints could be added to a modular viewpoint library for use within that particular domain. Hilliard comments, that one would then expect organisations to create a formal syntax and language to describe modular viewpoints and share them as a best practice. [26]

In the previous standard (IEEE 1447), the architectural description deals with known stakeholders and concerns on the inherent architecture of a system of interest. Whereas, the architectural framework would work on the assumption that stakeholders are unknown, and by extension, concerns and viewpoints. In the new standard (IEEE/ISO 42010), which replaces IEEE 1447, viewpoints are the main component. However, this is not something genuinely new since the architectural framework developer would already implicitly do this. When developing an architectural framework, the developer thinks about people, or rather stakeholders, who want to achieve and understand something about a given system and tries to codify the substantial elements into the framework [26]. Rather than keeping this knowledge intangible, one should make it explicit, such that it actively adds to the architectural knowledge captured within an organisation. Even more valuable is when such knowledge could be open source, shared as a best practice as stated in the previous paragraph, this possible because these are modular viewpoints. The modular viewpoints can transcend a single organisation to an industry domain, or even cross–domain [31].

Additionally, some viewpoint domains occur in most frameworks and literature, which work independently of the manufacturer. Some refer to these domains as classes or vantage points [13] [43]. Although the individual names often differ, due to lack of consensus within the field, the contents of these viewpoints do not. However, due to the abstract nature of known frameworks, concerns and stakeholders are not mentioned. For the sake of simplicity, we refer to these viewpoints as main or primary viewpoints. The following main viewpoints can be distinguished and are imaged in figure 1:

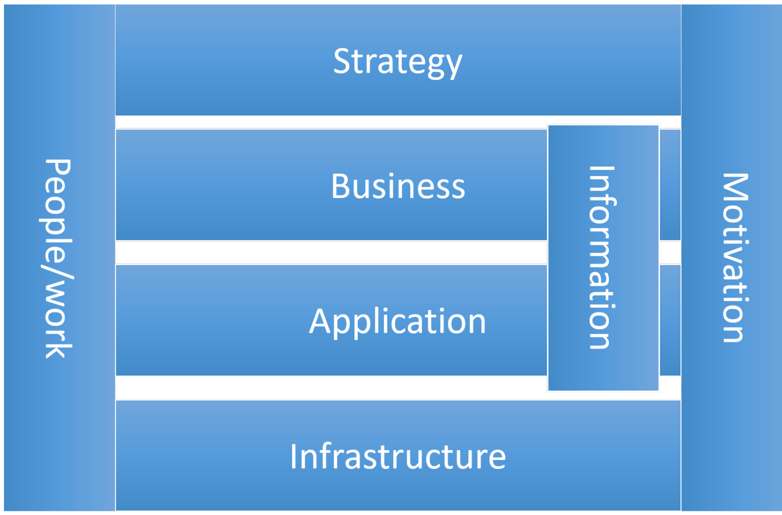

Fig. 1. Primary viewpoints visualised

- Strategy: gives insight into the why question, as it states the Organisation's strategic direction and vision, constrained to the impact on architecture. This domain influences all other domains. Moreover, think of objectives, goals, strategies, requirements, constraints and rationales. Note some further divide this strategy domain, with a separate motivation domain containing the needs, concerns, stakeholders and explanations. Some frameworks also acknowledge the existence of a separate capability viewpoint.
- Business: a functional view on the use of the process, their logical dependencies & patterns. This domain answers how the enterprises set out to fulfil the why question in the strategy domain.
- Application: a functional view of the use of information systems and automation. This domains answers how the business is supported
- Information: which information system/data is needed to operate, what is shared, and its relation to the functional views.
- Infrastructure: how are the information and application domains supported. The hardware and technical infrastructure support this.
- People/ Work: who is responsible for what. It describes which actors are responsible for which behaviour or structure, such as processes, capabilities, data objects, applications, or networks. Furthermore, it allocates organisational components to the locations & actors; how they operate & communicate and divide behaviour or structure into workable segments. The organisational structure supports this viewpoint.

The primary focus of these viewpoint domains and EA, in general, is to present an integrated picture of the business, work, functionality and IT through a set of views serving as a baseline for the as–is and to–be architecture descriptions. These viewpoints provide principles, documentation and reference models to guide future architectural developments, as well. Furthermore, using the idea from Ryoo & Saiedian [76], one can use primary viewpoints to classify and extent viewpoints, through the use of hierarchical relations, an example is given in figure 2.

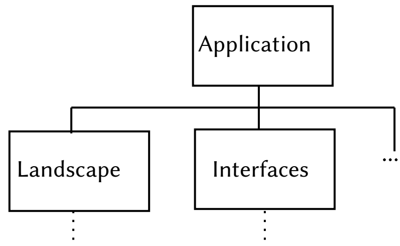

Fig. 2. Example of hierarchy relations with primary viewpoint application to classify and extent viewpoints.

CORRESPONDENCE AND CORRESPONDENCE RULES Each model's traceability and consistency within one view are ensured by correspondence on architectural description's elements. In comparison, correspondence rules ensure the traceability and consistency of views [26] [32]. Other literary works refer to this as architecture integration [9].

A set of the correspondence consists out of patterns, standards and rules, which relate two or more individual architecture description elements to each other within a view. [26] [32] An example is a meta–model or that a business function should relate only to business processes, such that no business function has any direct relation to an application service. While the correspondence is within views, correspondence rules transcend the view plane and exist in the plain of viewpoints. They describe how AD–elements should relate across all or most architectural descriptions, which has the side effect of specifying relations between viewpoints at the same time [26] [32].

Good consistency between viewpoints, views and architecture descriptions elements is necessary to make enterprise architecture useable. For example, mapping the application functionalities directly to which processes are supported by that functionality, is necessary to create a notion of impact. For example, when a particular application is migrated [13].

ARCHITECTURAL DECISIONS AND RATIONALES Architectural decisions are defined in this paper as decisions, assumptions, constraints and other elements that have significantly impacted the creation and design of an architectural description. The architectural decision can be decomposed in the actual decision, pertaining to a specific set of concerns, and a decision rationale, giving reasons why it has been made. These decisions affect how an architectural description is made, and often these decisions are not well documented [31]. As a result, this knowledge merely resides in the minds of the architects. Kruchten et al. [48] refer to these decisions as architectural knowledge since they will determine the architectural description's design and visualisation. Documenting this intangible knowledge yields significant benefits. First, there is no need to reverse engineer architectural descriptions, the so–called architecture–recovering process, to uncover why the architectural descriptions are as they are today. Secondly, the architectural descriptions' maintenance over time is more straightforward, as discussions need to be waged only once, and there is no need to reverse engineer the descriptions. The documentation of architectural knowledge also benefits the detection of wrong decisions or to teach other architects. Furthermore, explicit knowledge can be reused in new projects, can help architectural conformance and can be used to communicate more clearly. Finally, it also opens the possibility of peer reviewing and makes sure that new architects know why architectural descriptions are designed as they are, even when the old guard of architects have left. [48] [32] [31]

ARCHITECTURAL KNOWLEDGE The capture of architectural knowledge through codification, such as viewpoints, has its implications. First is the increased costs, capturing means codifying, which costs time and resources. However, this is a short–term investment, as to the long–term costs are saved [40]. By having architectural knowledge well documented and easily accessible [15], one creates opportunities such as evaluation, durability, automation and increased understanding. Regarding evaluation, by having architectural knowledge explicit, one can evaluate made decisions, use rationales, and correspond to architectural descriptions to uncover risks or identify sub–optimal choices. Consequently, one can take adherent action [56].

Moreover, durability and increased understanding reduce costs, since one uses modular viewpoints, which means that viewpoints are reusable [26] [48]. Consequently, the cost of creating a viewpoint is not repeated, at best small incremental improvements are made.

A stated earlier one could also start automating architecture. Automation examples include automatically generated documentation [56] or automated analysis tests in tandem with the architectural description. Think of analysis on syntax errors, quantitative (performance) tests, functional (static/dynamic) analysis, risk analysis, portfolio analysis or a capability analysis [49]. Do note that automation relies on the assumption that the codified technique relies on a well–defined syntax.

Furthermore, the architectural process is sped up since modules are plug and play. Secondly, the increased understanding means new employees, and by extension, new architects, know which decisions were made, why they were made and which consequences they have had. Consequently, one does not need to reinvent the wheel when an architect leaves the organisation. It would reduce maintenance costs of architectural descriptions as well; for example, one does not need to reverse engineer to uncover why a particular design decision was made. Do note that some have argued that this type of working is incompatible with the agile mindset. However, there have been a lightweight knowledge mechanism proposed, which are claimed to be compatible [15] [51].

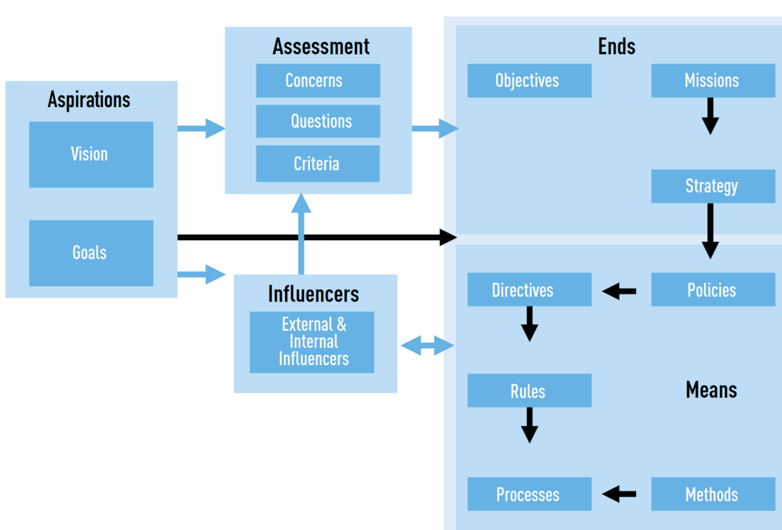

Fig. 3. NATO–AF visualisation of architectural motivation.[10]

ARCHITECTURAL PRINCIPLES AND MOTIVATIONS The principles are direct statements giving context to the organisation by defining constraints [13] [12]. Another way of saying this is that principles are structured ideas that define the organisational context and guide the enterprise to fulfil its mission. TOGAF distinguishes two kinds of principles, enterprise principles, which provide a basis for decision making, and architecture principles, which are principles related to architecting and architectural products [69]. Every principle has implications for the organisation. Hence, it is common to describe some implications for each stated principle since this gives a better idea of how principles affect the business [13].

Where principles actively guide the architectural process, motivation data are the underlying reasons for starting with architecting. These reasons shape the architecting process's initialisation, orientation, and products and are often strongly correlated to stakeholders' concerns [10]. Examples are a cooperate strategy, external drivers, policies, engineering processes or business developments. Moreover, figure 3, omitted from the NATO–AF, gives a graphical representation of motivational data.

As a summary, in figure 4, the relation between the ontology as used in this paper is represented.

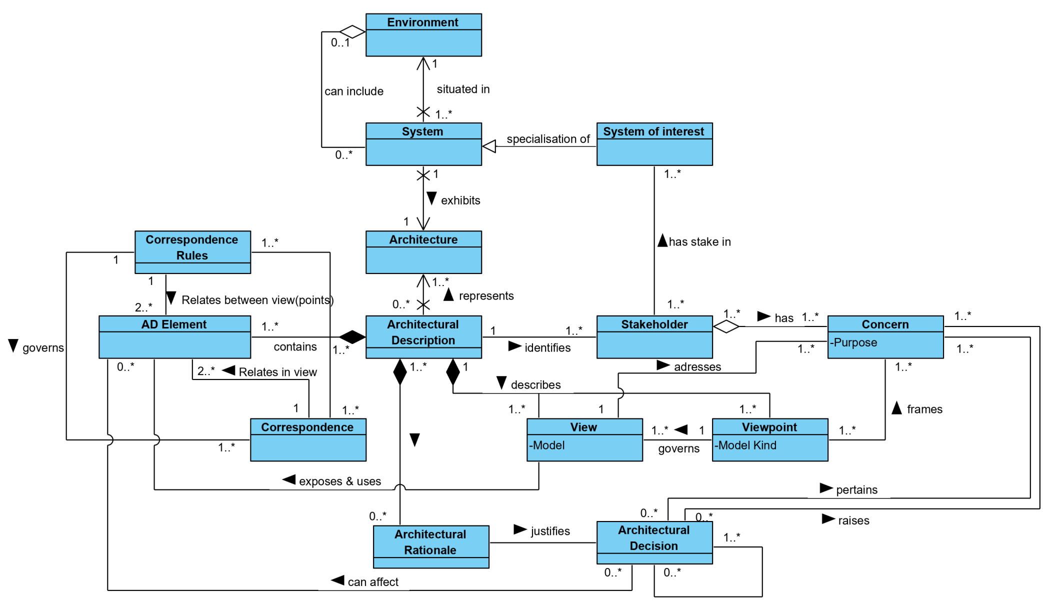

Fig. 4. Ontology

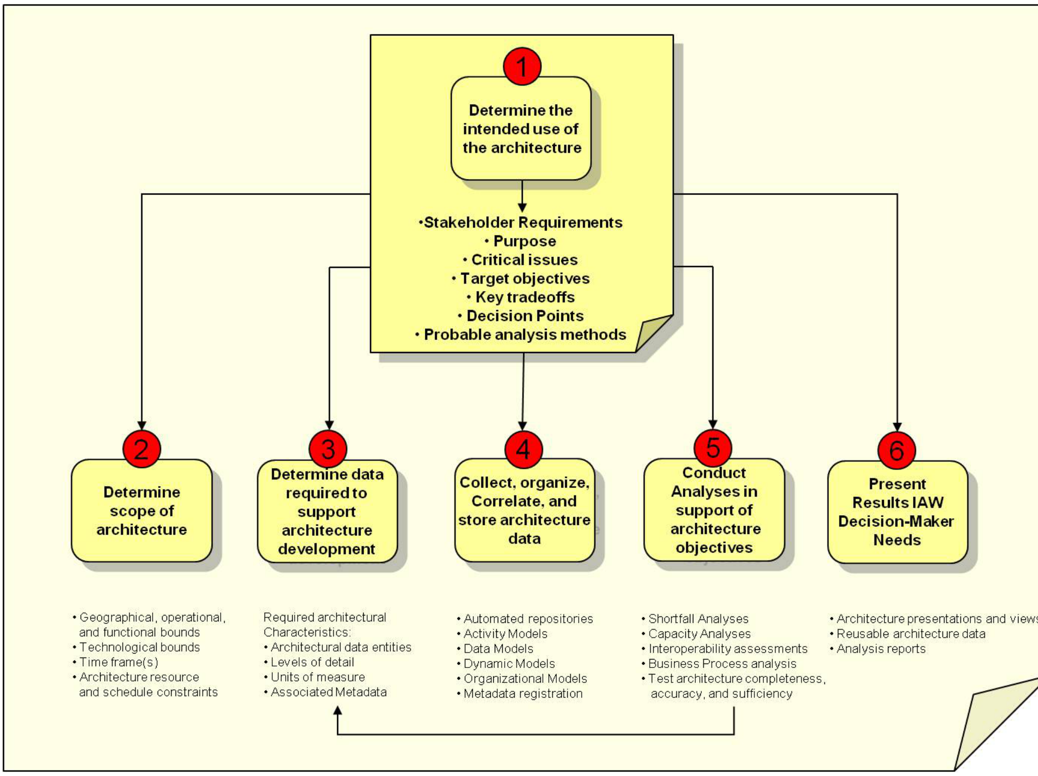

Fig. 5. DoDAF's development methodology [23].

2.2 DODAF

The latest version of doDAF has been released in 2010 and has its scope within the US government [59]. It consists of two volumes:

(1) Definitions and guidelines.
(2) Product & descriptions.

The DoDAF framework describes the architecture description such that key stakeholders can focus on specific areas whilst retaining sight of the big picture for decision making. DoDAF deviates from the earlier defined concepts, as the architecture description in DoDAF consists of models and are sometimes referred to as products instead of views. These models belong to a viewpoint. Furthermore, DoDAF defines views as a way of presenting multiple models to stakeholders. Additionally, DoDAF states 41 models divided over eight viewpoints, see table 8 in the appendix, which quotes the source text.

Besides the models, viewpoints and views DoDAF also states meta–models and a high over the development process, depicted in figure 5.

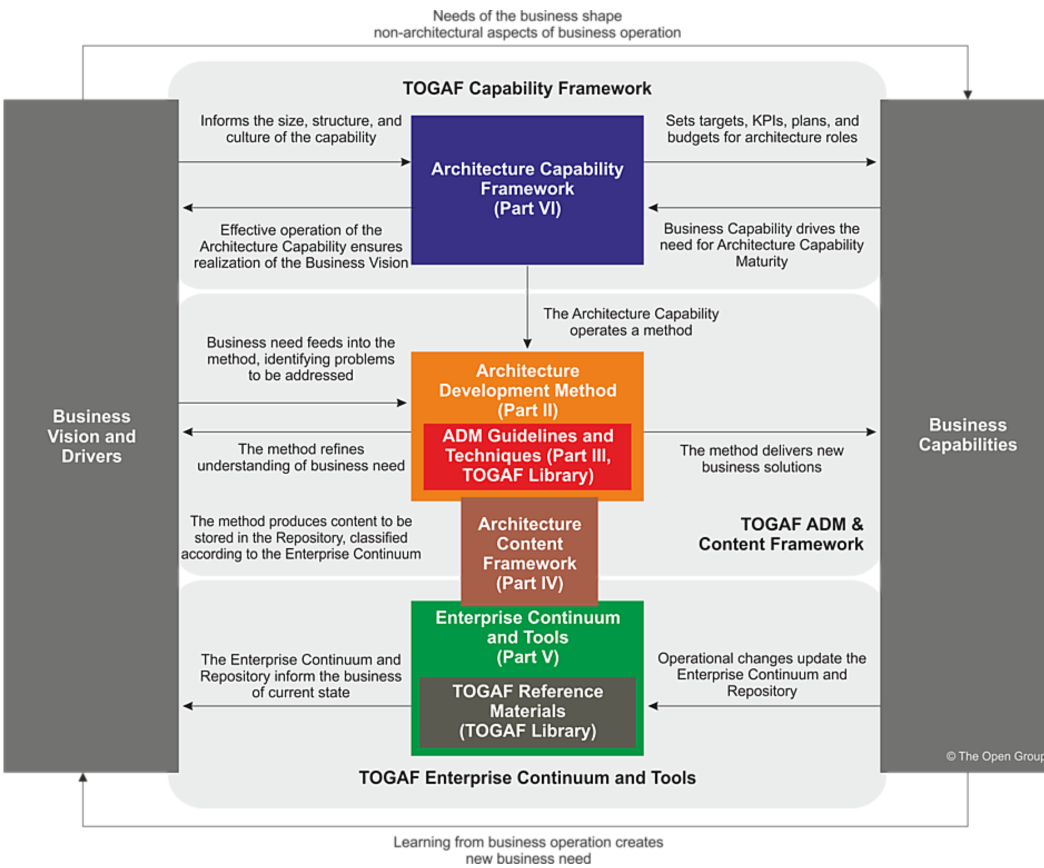

Fig. 6. TOGAF structure visualised [65].

2.3 TOGAF

TOGAF [66] is one of the most prominent frameworks [81] and consists of a framework and a methodology (ADM), depicted in figure 6. Moreover, the open group states that TOGAF is mostly ISO–42010 compliant [68]. Furthermore, TOGAF recognises four different architectural domains.

- Business Architecture, such as strategy, governance, organisation and processes.
- Data Architecture, such as logical and physical data structures and data management.
- Application Architecture, such as the application landscape, their interdependencies and the relation to processes.
- Technology Architecture, such as infrastructure supporting business, data and application services.

TOGAF'S ADM Depicted in figure 7, is the architecture development method (ADM) from TOGAF [66].

(P) The development project is prepared in the preliminary phase, such as the project method, scope, team, communication plan, governance, strategy, and a preferred tailored framework. The general architectural principles are defined as well.
(A) In the architecture vision phase:

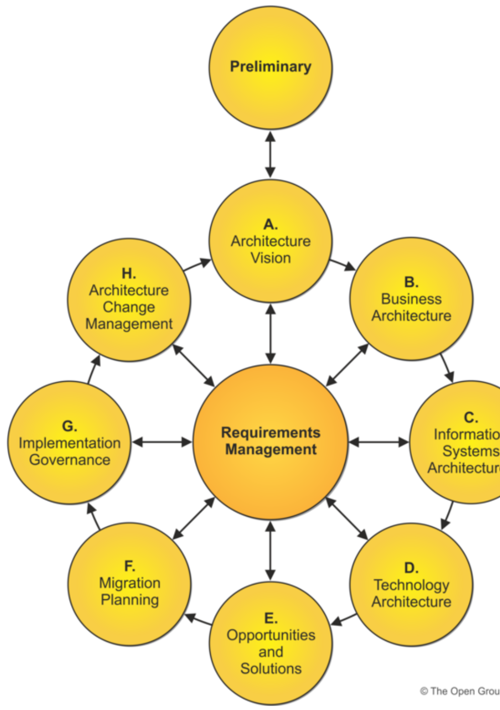

Fig. 7. TOGAF's ADM[64].

(a) One establishes the project, to ensure recognition, support from management and commitment from employees.
(b) One identifies stakeholders, concerns and business requirements.
  (I) Note that stakeholders are the ones that can raise:
    (i) Concerns. Concerns are directly correlated to the overall acceptance of a product.
    (ii) Requirements. The demands of the product which can be measured.
    (iii) Principles. A way to describe the context and state qualitative requirements.
    (iv) Patterns. Techniques, standardised solution to clearly stated problems [22].
(c) The capabilities needed are evaluated.

(B) In step B to D, one develops the architectural descriptions according to an eight–step process.
(a) Select relevant reference model, viewpoints and tools.
(b) Develop a baseline, a formal specification that is reviewed and agreed upon, and future reference (often an as–is description).
(c) Develop the architecture description.
(d) Perform gap analysis.
  (I) Resolve conflicts between the views.
  (II) Validate models support principles, objectives & constraints.
  (III) Document deviations from selected viewpoints.
  (IV) Test models against requirements.
(e) Define business roadmap components, which can be prioritised on the business roadmap.
(f) Perform an impact analysis.
(g) Conduct a formal stakeholder acceptance review & finalise.

(h) Create an architecture definition document. (Design decisions, rationales, footprints, element descriptions, . . .)

(E) In the opportunities and solution phase, one harmonises, consolidates and refines the products from step B to D. After which work packages and a roadmap is created.

(F) Step F to H are the migration, implementation and maintenance of the architecture.

TOGAF'S FRAMEWORK The TOGAF framework states a meta–model, deliverables and several artefacts [66]. Moreover, artefacts are architectural products. Aiding in the creation of said products are building blocks, building blocks are containers of architecture with various levels of detail depending on the maturity of the architecture development. These building blocks can be combined to create an architecture description. TOGAF distinguishes two building blocks; a logical building block referred to as an architecture building block (ABB) and a physical building block referred to as a solution building block (SBB).

TOGAF distinguishes three classes of model kinds in viewpoints, which can use these building blocks, namely:

- Catalogues, which are lists of building blocks.
- Matrices, which model relationships between blocks.
- Diagrams, which are visualisations of the former two classes.

These classes are related to the meta–model content for TOGAF, which creates the set of viewpoints found into TOGAF. Note that these viewpoints do not state stakeholders, concerns, motivations or patterns, as TOGAF expects the architects to add them themselves to enrich the viewpoints. This enrichment is not a requirement of TOGAF but is a requirement to be ISO–42010 compliant. The metamodel and the viewpoints are visualised in figure 9 below.

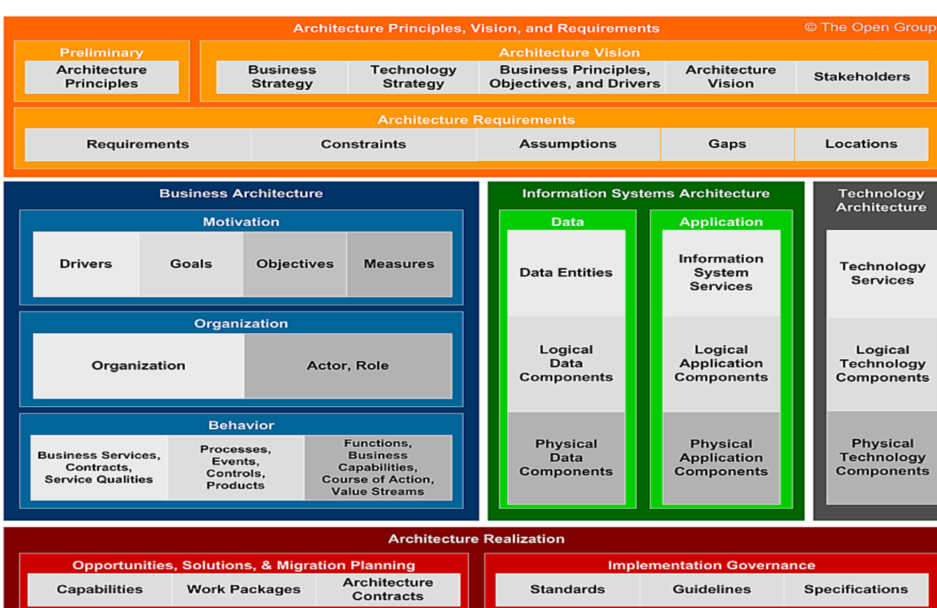

Fig. 9. TOGAF's content framework [63].

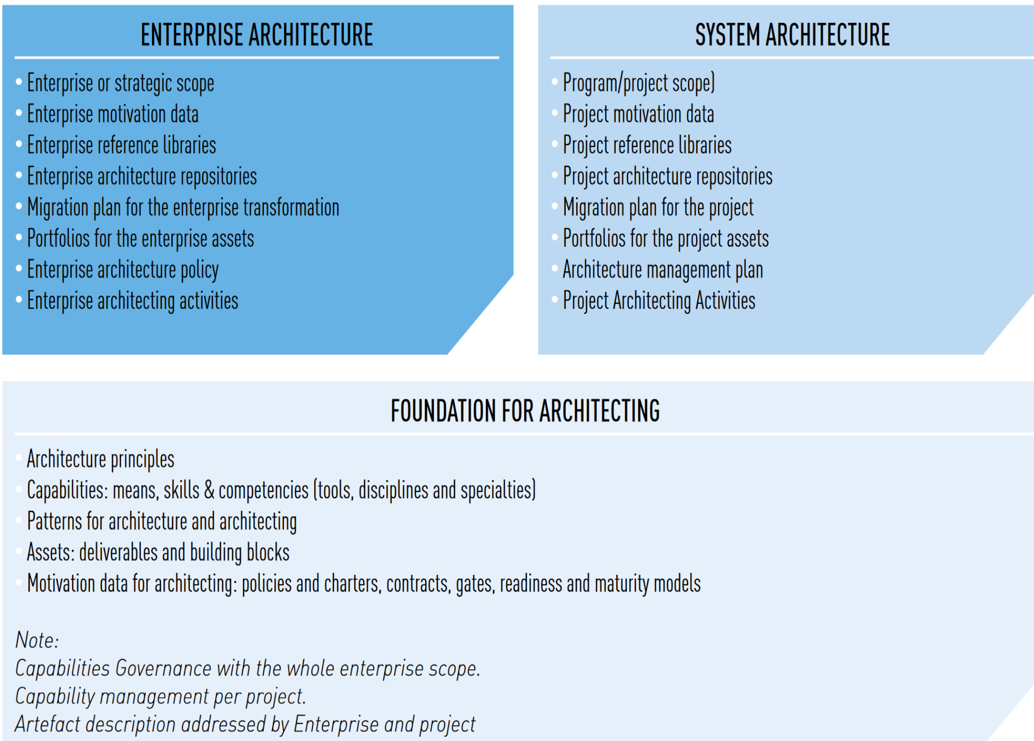

Fig. 10. NATO–AFv4's fundamentals [10].

2.4 NATO–AFV4

The NATO Architectural framework [10] consists of a methodology, a collection of viewpoints and a meta–model. This framework has fully adopted the ISO 42010–42030 standard and distinguishes three methodological areas, imaged in figure 10:

- **Enterprise architecture**: The common ground, where the enterprise vision and structure are located, as well as for instructions on how to perform activities concerning enterprise motivation and project governance.
- **System architecture**: The project level, where goal and structure are located regarding a subset of systems found in the enterprise architecture.
- **Foundation for architecting**: That what is required and used by the other areas, such as tools, principles, patterns, motivations and the architect' repository.

Contrary to TOGAF, NATO–AF does not actively state architecture kinds, such as business architecture and information system architecture. Instead, it lets the architect decide based on enterprise needs. In contrast, it does actively state architecting styles, namely:

- Authoritative.
- Directive.
- Coordinative.
- Supportive.

Another literature work proposes four different kinds of styles based on the purpose [9]:

- Design.
- Communication with a particular group.
- Realisation & integration.
- Change & transformation.

By choosing an architecting style, the architects and the problem owners are helped. A small number of fixed architecting styles can lead to standardised approaches, set expectations and clarify which type of value should be delivered.

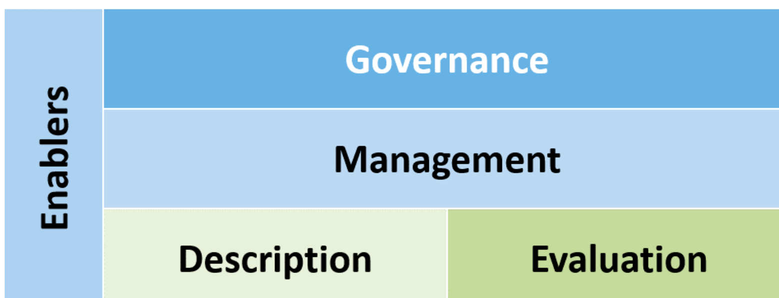

Fig. 12. architecting process domains [10].

Within each style, there are five distinguished processes, see figure 12. For this research, the focus lies in the description process. The architecture description process has nine main activities:

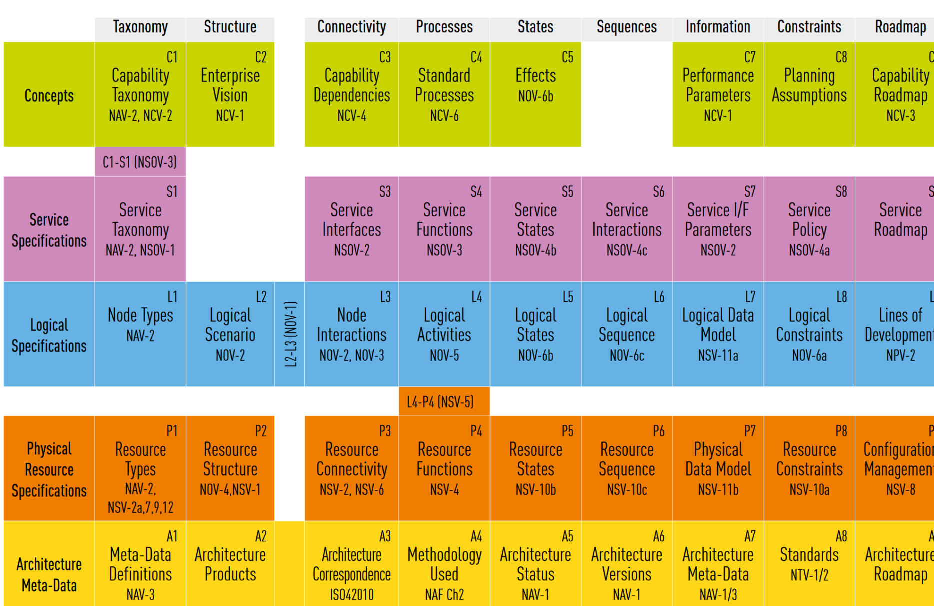

Fig. 11. NATO–AF's viewpoint framework [10].

(1) Analyse the problem situation.
    (a) The vision, strategy, goals and expected result.
(2) The identification of stakeholders and their concerns.
(3) Distil requirements from the identified concerns.
(4) Identify viewpoints pertaining to earlier identified concerns.
(5) State potential solutions.
(6) Develop views from the viewpoints.
(7) Actively state the reasoning behind views and viewpoints regarding requirements and motivation data. Additionally, one can create a dedicated architectural decision viewpoint.
(8) Review the candidate views with stakeholders.
(9) State the relations between views, viewpoints, design and other activities. This correspondence is often documented in the viewpoints.

As stated earlier, the NATO–AF also expresses several viewpoints as its framework. These viewpoints can be classified, which is represented in figure 11. The y–axis is explained in table 1; the x–axis is based on the same viewpoint classification used in DoDAF.

Aspects	Description
Taxonomy	Specialisation hierarchies of architecture elements such as capabilities, services, etc.
Structure	How elements are assembled (enterprise, nodes, resources, etc.).
Behaviour	How things work, Processes (Flows & Decomposition), States (Allowable state transitions), Sequences (How things interact and in what order).
Information	What information is used, and how it is structured.
Constraints	Rules that govern the enterprise, nodes, resources, etc.
Roadmap	Project timelines and milestones affecting the elements in the architecture.

Table 1. Aspects of NATO–AF [10].

2.5 GENERAL METHODS, FRAMEWORKS AND BEST PRACTICES

In this section, common approaches, problems, solution and cases are presented for enterprise architecting. The focus lies in enterprise architecture (EA) as conducted within the public sector.

In the broadest term possible, systems can no longer function separately but should be conceptualised, modelled and developed at an enterprise level, to improves its value [12]. Note that within enterprise architecture, there is no such thing as a greenfield [12] [14]. Even an Organisation who starts with the architecting process has an architecture. The reason is simple the Organisations exist; therefore, architecture exists. To clarify, an Organisation has processes, automation, and data amongst others, and these elements have an inherent architecture. As a result, an Organisation only beginning with the architecting process, already has architecture, but has substantial architectural debt, as the architecture is not adequately described and structured based on the unified enterprise vision [12]. Moreover, an Organisation without a long–term business and ICT vision will find enterprise architecting exceedingly complex and costly. A clear indicator of a vision's importance is that every method or framework sets the vision and the people central.

One of the challenges in the architecting process is the adoption of a framework. In this choice, there is a devil's dilemma. One needs to balance a strict enough framework that aids the architecting process and allows enough freedom to withstand all modifications conducted by different project groups within the enterprise. Furthermore, frameworks should follow Organisational vocabulary and culture. Therefore, even if one is to adopt external frameworks, one still needs to refractor the vocabulary [13].

FRAMEWORK & METHODS One may have noticed common steps amongst the different methods presented thus far. It starts with a concept phase in which the scope and vision are determined. Incidentally, architecting on an enterprise level is more concerned about defining the future state than reaching the future state. Also, note that the architectural vision should answer questions such as for whom (stakeholders), what (concerns & systems of interest), which priority (what goes first), how (methods, standards and tools), with whom (work location, team development). The phase after that is to describe the as–is situation or baseline. The baseline might be an in–between product; however, it is still useful. It already creates a shared reference, helps identify shadow–IT, gaps and redundancies. Moreover, it helps answer several critical questions about the company, such as why, how and what. Furthermore, one should not forget to acceptance test made views with the stakeholders. This phase followed up with a phase where the future situation is described. Finally, one plans for the transition, migration and implementation phases. [14]

Hence, one can distil a broadly described architecting process, such as the following incremental process [13]:

(1) Define a shared business & IT vision, which include architecture principles.
(2) Initiate the development process and create a common vocabulary [12].
(3) Create a baseline, or in other words, describe the current state.
(4) Describe the future state, think of trends, viewpoints, views and standard profiles.
(5) Form a migration plan on how to get there.
(6) start/continue the process of architecture administration and maintenance [12].

Note that there has been an agile compliant architecture development process described. However, these agile processes are more suitable for project architecture as distinguished in NATO–AF. The literature is scarcer within the scope of enterprise architecture [51].

To resume, there is no such thing, in any kind of development, as a silver bullet [1] [14]. That is to say, a generic approach with guaranteed success. It follows then that organisations mix and weave different approaches and frameworks together into a mix that works best for their organisation. This 'mixing & weaving' is also referred to as method engineering, and several theories exist on its execution, such as MEMA [1]. Method engineering is something that can also be applied to the architectural development process [1].

However, one would be hard–pressed to mix and weave due to the sheer numbers of methods and frameworks [35]. Moreover, most frameworks and methods forget an essential facet. Within architecture, arguably, the most crucial facet is communication. If architecture as a discipline does not improve or benefit the communicative aspects within an organisation, then the promises, such as business & IT alignment and better decision making, will never realise. In this aspect, one should not forget that the people make the world go round, not the systems or processes; they are dependent on the people. Moreover, the knowledge of architecture are inside people's heads. In conclusion, architecture as a discipline is entirely dependent on people and its success in communication & elicitation with these people.

One can distinguish three layers within human communication, syntax, semantics and pragmatics [35]. The syntax is the structure of communication in architecture; these are symbolic models. In other words, models which express properties of a given system through symbols [9]. Note that the symbols express no meaning. Semantic models are the interpretation of symbolic models; now, the symbols do have meaning. In architecture, the semantic models are the subjective interpretation of architecture by a stakeholder group, also known as views and viewpoints [9]. A simple example of a symbolic model is $7 + 5$; an example of a semantic model is noticing that $7 + 5$ is 12 [9]. Lastly, pragmatics can be best expressed as the social impact or impact on a group. In architecture, pragmatic models are architecture vision, principles, reference models, migration planning, and future state modelling, among other topics, that directly impact a social group [35]. When communicating one shares knowledge, this is no different from communicating architecture, through views and viewpoints. Within the sharing of knowledge, there are three stages [35]:

(1) Aware– One is mindful that architecture descriptions exist and knows where to find it.
(2) Agreed– One has incorporated the knowledge within architecture descriptions into oneself. Note that one can either agree or disagree; a consensus is not a prerequisite.
(3) Committed– One has adopted the knowledge within the architecture descriptions and has actively changed one's behaviour according to the gained insights.

Consequently, architectural artefacts should clearly state the purpose, function, design quality, and costs. Moreover, the knowledge goal and communicative purpose must become evident [35] [34]. For example, a viewpoint should state what knowledge the views should communicate towards the stakeholders. The success of said viewpoint could be measured in the three presented stages, aware, agreed and committed. Note that there is a zero–state oblivious.

However, stakeholders rarely know what they want, what they wish to be informed on, and each stakeholder speaks a different language from the other. This language confusion can exist due to natural evolution over time, different subgroups within an Organisation, the difference in social networks or unique outside influences [34]. To lift this language confusion architecture as a discipline and its viewpoints & views should be a means to create a shared understanding of the Organisation across heterogeneous stakeholders. This requires articulating concise and coherent definitions and concepts, such that one reaches a consensus of concepts amongst stakeholders [35] [34].

REFERENCE ARCHITECTURE There are several references an architect can use whilst architecting. These include categories as principles, capabilities, assets (repository), motivations and patterns. Think of standard profiles, which describe the available standards within an enterprise and a reference architecture. Reference architecture descriptions give insight into the complete structure of an enterprise as a set of services. These services are a logical representation categorised by functional areas. [13]

Another way of looking at a reference architecture is as a pattern of architectural construction. Since it extracts an often–complex diversity of sources, think of principles, expert knowledge, heterogeneous systems and environments. Consequently, reference architecture descriptions guide the architect's understanding during architecting. As it gives the architect a reference as to what components mean, their responsibility, the vertical & horizontal structure, standard building blocks, and the integration of components. [57]

COMMON ISSUES As stated in the introduction, enterprise architecture has a low success rate [4]. This low success rate does not stem from ICT issues, but from internal politics and Organisational issues [43]. One problem is that enterprise architecture should move away from an IT focus to a more business focus. Consequently, it can be used as strategic planning, business transformation and improved business-&-IT alignment [81]. One research notes that within EA, there should always be a business representation, in which there is a non–IT description of the architecture of the enterprise. In a business representation processes and business–events are central to the descriptions [43].

Even more, enterprise architecture should support most, if not all viewpoints and views. All stakeholders should get the general gist off, how the enterprise works and sets out to achieve its vision. These stakeholders include the business, but other stakeholder domains as well.[43] The main issue here is the often abysmal communication between architects and stakeholders [42]. Another problem is the maintenance of architecture descriptions. Architects tend to wish to be perfect at once and build an enterprise architecture that will stand the test of time. However, it is more viable to be pragmatic and describe the 80/20 of the architecture well. Furthermore, one should start small and build up in iterations. Armour notes that one should build to change, not to last [43]. To add insult to injury, once an architecture description is delivered and never changed, then the information from the description used in decision–making is dated or dead wrong, diminishing enterprise architecture's value. [18] [43]

An additional issue within the architecting field is that architecture is not just structured in tech and information but also social–economical, namely the business and the context [9]. This social factor makes it hard to have a limited number of models and modelling languages as the business, and the context is dynamic. An enterprise's context exists out of many heterogeneous stakeholders and concerns. As a result, enterprises often use heterogeneous models and descriptions, since architecture needs to be understood by different social groups. The resulting heterogeneous views must have proper correspondence which creates a complex control and management environment. Furthermore, while creating the views, architects often get confused between the distinction of model presentation, content, and semantics. In other words, architects forget the communicative purpose syntax, semantics, and pragmatics. Furthermore, when one creates heterogeneous views for architectural interchange, one must ensure that the differing views and viewpoints are as similar as possible. If not, EA would create confusion, rather than mutual understanding and effective communication between different business domains [9]. There have been some propositions on this front on how to achieve this, for example, using an ontology as a basis of integration [8], using action tables and algebra [9] or using extensible architectural views [76].

Lastly, architecture does not seem to add any value to the organisation on paper. If one reads the literature, then architecture as a discipline should provide improved communication, decision–making and better business and IT alignment. [43] However, it has seldom been measured in the field. To clarify, no standard metrics or key performance indicators exist within companies. As far as the accountant or the manager is concerned, architecting is a black hole for company resources. Moreover, the time–consuming nature of enterprise architecting does not improve this image many non–architects have about architecture [42]. In other words, how does an architect contribute to Organisational profitability and how well does the architect fulfil the promises of architecting, such as the better business and IT alignment [43]? The lack of clear indicators hurts the credibility, compliance and the repeatability of architecting within an Organisation [42]. Although someone could argue that architecture is a qualitative and fuzzy discipline. Consequently, it is hard to measure architectural quality. Despite this argument, some indicators can still be implemented. Armour states a few examples such as [43]

- The annual updates to EA.
- The ratio of compliant systems.
- The number of changes done on EA.

Moreover, it has been proposed to create indicators along three axes to evaluate architecting and its products [57], namely:

- Effectiveness: how well is IT aligned to the business, how well is the value contribution of systems optimised.
- Efficiency: how well are systems consolidated into standard building blocks, what is the standardisation and reusability level within the Organisation.
- Reliability: how transparent is architecture, how well do stakeholders know architecture, how has architecture minimised risk within the Organisation.

BEST PRACTICES & LESSONS LEARNED As stated earlier, there are few empirical studies into the architectural development process, despite this literature state some best practices and lessons learned:

- Use dry runs for the framework, which should result in quick wins and lessons learned [13].
- Start small let EA grow [43].
- The right tooling [43].

- Each architecture description is unique but shares common patterns and principles [43].
- Look into how the neighbours do EA [43].
- Reuse and adopt standards where possible, do not just build & reinvent the wheel[43] [51].
- Fanciness does not make profitable EA [43].
- Require EA for review boards, acquisition and investment plans [43].
  - Most IT systems come from ad–hoc department acquisitions rather than enterprise planning, which leads to undoing the EA work and can lead to shadow IT [50].
- Focus first on principles and reference models [12].
- Develop an as–is situation before the future state [12].
- Make sure all information is readily available [12].
- A highly distributed enterprise should have separate EA for each sub–organisation [12].
- Make sure experienced experts develop EA. A wrong architectural decision has a significant impact on the enterprise. It costs less to hire or train knowledge than it is to mitigate bad architectural decisions. One could also hire an external consultant to provide aid [14].
- Establish agreed–upon methods and standards, especially for communicative purposes towards the non–architect outside of the project development team. Be open and actively share drafts for review and discussion. Architecture is owned by the business, which should mean everyone in the Organisation; nothing related to architecture should be secret [14].
- Top management must be committed, and if they are not, stop the development project, it will never succeed [14].
- Architectural development is a full–time job requiring a high commitment to the development members. Putting part–timers or indifferent employees on the job is a sure way to devalue the resulting products [14].
- Stakeholders, all stakeholders, should reach consensus and accept the delivered architectural products. This unanimous consensus will be especially hard in Organisation with competing internal groups, but architecture is a means to communicate and harmonise, it requires commitment and acceptance of all groups [14].
- Be aware of architectural rigidity. Enterprise architecture is about the enterprise with heterogeneous groups [76] [14]. All groups must understand and be able to use architectural products since they will be the communication base. Although, architects often adopt the best practice of choosing a formal architectural description language, rarely do all stakeholders understand such a formal language. Therefore, it is advisable to be more lenient in enforcing the formal language or adopting special informal notations for certain stakeholder groups. Note that this architectural interchange requires architectural correspondence rules [14].
- There are many possibilities to automate documentation based on data flows. Documentation is often overlooked and rushed; it is, therefore, advisable to use these means of automated documentation [18].
- Prioritise concerns and tackle significant concerns first [51].
- Always use architecture as a means for communication [35] [51].
- Document decisions, architectural changes and rationales [31] [51].

2.6 PREVIOUS RELEVANT WORK

The public sector needs to change its Organisation to respond quickly in a volatile, uncertain, complex and ambiguous world (VUCA) [17]. The adoption of enterprise architecture within the public sector is one piece of the puzzle, as it can aid the public sector to solve this problem. In particular, the high redundancy, rigidity and the lack of modularity within the architecture of public sector Organisations. This finding is also reported by an empirical study in Portugal [72].

Furthermore, the public sector is often still organised in lines. In this situation, EA can be the common ground as the public sector transforms into a more agile structure. This common ground is necessary as the path to transformation is especially hard for the public sector due to cross Organisational dependencies. Where each Organisation has its share of legacy systems and inflexible interfaces, EA can help the public sector manage fragmented Organisations and processes [6]. Additionally, an extra hurdle exists for the enterprise architecting process in the public sector, namely, the increased number of stakeholders. This increase in stakeholders is due to the interconnectivity of Organisations within the public sector [17].

Although empirical studies in the public sector concerning architecture development are few, there are some studies. We have identified seven. Three of which research the educational industry, another three local ministries, and one in the health care sector.

Architecture adoption challenges in the Malaysian public sector. Nur et al. investigated enterprise architecture adoption challenges within the Malaysian public sector through three case studies [3]. Across all three case studies, they find the following key challenges:

- The lack of understanding one's internal processes.
- The lack of standards adoption across the enterprise and within architecting.
- The lack of evaluation criteria, such as KPIs.
- Documentation that exists is either partially completed or is not used.
- The architects who are employed lack experience.
- The usage of overcomplicated tooling in relation to architecture maturity.
- The difficulty to retain expertise within the enterprise.
- The lack of an architectural board or central expert EA team.

EA development in the Malaysian public sector. In the same Malaysian sector, Suraya and Mahdi's later study reports on the empirically observed EA development project [5]. The process they find can be summarised as follows:

(1) Establish project governance.
  (a) Gather a development team divided into workgroups.
  (b) Assign a steering committee.
(2) Hire an external consultant to provide needed architecting experience.
(3) Start a parallel process:
  (a) Establish an EA knowledge mechanism. Subsequently, create a framework.
  (b) Investigate the use of internal IT and standards.
(4) Reach consensus on a final enterprise framework, named GAF.
(5) Start architecting.
  (a) Continuous iteration of workgroup engagements.

In step one and two, they note that the steering committee should have IT affinity and actively pursue senior management's continuous commitment. Furthermore, the steering committee did not hire an external consultant. Instead, they gave the command to the development team to 'just find someone'. This external consultant was a must as the architects were new. An architect notes '… none of us had (any) formal experience'.

On step 3a, they note something interesting, the EA framework they developed, based on TOGAF, COBIT and ITIL, was to be used in all Malaysian public sector agencies. On a side note, it is quite a challenge to develop a framework useable in all public sector agencies and create views for a ministry in one project, whilst the architects had minimal experience. To resume, they also note that a custom enterprise framework was necessary since TOGAF was focused on the private sector. This statement is exciting but is not explained. How one has reached such a verdict is interesting as it shines a light on how the public sector functions differently than the public sector. Especially, since TOGAF is a set of best practices and frameworks, made to be plug and play, such that any Organisation, be it public or private sector, can use TOGAF. Moreover, the open–group actively publishes scenarios for the use of TOGAF in the public sector [62].

Within their custom-made framework, they define, principles, main deliverables and a process to identify stakeholders. More on the stakeholder identification process, the development team sends a letter to senior management in this process. This letter is more or less a job advertisement, where the minimal skills and time required is defined. The senior management sends this letter throughout the company.

Meanwhile, the development team waits for a reply. After a while of collecting replies, the stakeholder applicants were taken in for a job interview. Consequently, if they accepted, they were distributed amongst the workgroups. This direct stakeholder involvement was necessary for the architects since they did not know the Malaysian government.

The workgroups worked as follows. Every set period, there is an active workgroup engagement in which the stakeholders' review made products, give feedback, and describe the next architectural product. Between these engagements, the architects incorporate feedback and create a related architectural product. This defined process reiterates until the custom framework, named GAF, is filled. On a side note, there is no remark on what non–architect workgroup members did between the meetings or the further involvement from the steering group.

During each iteration, the architects needed to fill six different roles:

- A role that acquires architecting knowledge.
  - Self–training.
  - Consultancy.
- EA development investigator.
  - Find out how EA in the public sector works.
  - Literature research.
- Framework developer.
- Workgroup former.
  - Recruit stakeholders into the workgroups and perform stakeholder management throughout the enterprise, namely the entire ministerial Malaysian public sector.

- The business analyst. Note that the original work named this role the data analyser; however, this is probably a translation error.
- Architect.
  - The one who creates the architectural descriptions.

The study does not report whether the implementation was a success or directly relates findings to the previous research identifying the Malaysian ministerial public sector's adoption challenges. Despite this, it is one of the few studies which describes the development process.

Empirical research of the adoption of SOA and EA in the public sector. An additional study by Axel et al. conducted empirical research on the EA evolution in a public agency [7]. They find that the agency started with a clear vision and an EA object, such as the alignment of strategy, business and technology. Further investigation yields that the underlying reason to begin with EA is that EA helps decision–makers by providing a coherent vision on all information described from different viewpoints. There were four primary objectives to EA described:

- Strategic: for the sake of improved alignment and aiding decision making.
- Governance: for the sake of the execution of strategy, requirements and governance of IT & business.
- Operational: for the sake of coherent documentation and optimisation of the enterprise.
- ICT: for the sake of requirements, development and reduced complexity.

After which the public agency mixes and weaves their framework based on Zachman and TOGAF. Whilst describing the evolution of EA, they do not describe in detail the development of architectural descriptions. They do make three conclusions, which influences the architectural development, namely:

- A good EA baseline is a prerequisite for further EA projects, such as SOA introduction. Not only for implementation but also impact analysis and coordination.
- Realising and maintaining the EA objectives improves engagement with other Organisational activities, such that they may require an EA.
- Well implemented EA is dynamic; it is a process directly intertwined with strategy and enterprise evolution, rather than a set of static documentation and descriptions.

EA adoption challenges in the Norwegian higher education sector Another exploratory study in the Norwegian higher–education sector notes that universities compete for students and funding, but they struggle with the same IT and business issues [60]. Therefore, cooperation and standardisation across the sector are possible and beneficial. To achieve improved cooperation in the Norwegian higher–education sector, it started the development of a joint EA. This movement's direct cause was that the Norwegian government had set out the objective that the educational sector should achieve more with less, through smart resource allocation. This exploratory case study tries to identify significant challenges and perceived benefits to a joint EA. They provide a short description of the EA process as well, summarised the process can be described as:

(1) The IT committee is formed to achieve cooperation on the IT front between educational institutions and recommend creating a shared EA. It acts as a centralised agency responsible for the common EA.
(2) A project was carried out to formalise the cooperation between institutions, which recommended establishing an architecture council responsible for developing the EA.
(3) Meanwhile, universities themselves come up with architectural principles, used to harmonise the IT landscape before the EA was developed.
(4) EA started development.

Further steps are unknown as the research took place during stage four. On the front of adoption challenges, they note that significantly smaller institutions were waiting on a joint EA, as they did not have the capabilities to create it themselves. Overall, the sector hopes that EA brings centralisation, where systems are shared, and interfaces are standardised. Moreover, centralisation would yield scale advantages, lower costs and reuse. Some institutions notice that EA and its promised benefits are a requirement to meet increased market demands.

Furthermore, it is interesting to note, while the government committee set out to harmonise the IT landscape from which processes may be standardised, the institutions themselves speak of the business's standardisation, from which standardisation in the ICT follows. The research notes that there was too much focus on the IT, which led to 'system thinking' and kept the reasoning in existing technologies and solutions, rather than the logical, abstract, reasoning joint EA requires. This physical architecture approach to EA, lead to alienating many institutions as they had other technologies and solutions.

The first major challenge that was identified was the absent commitment of top management. As EA is a long–term process that never ends, but always reiterates, it requires senior management commitment, less it is to fail. The leading cause is that EA is not concrete. It is unclear for management what EA does for the bottom line. As a result, the EA process was stopped in some educational institutions. Another reason is that the joint EA required cooperation, which smaller institutions could not provide since the IT department was too small and too busy with daily operations.

The second major adoption challenge identified was that there was no clear mandate for any group to control the EA process, recall that the architectural council was still being formed. Hence, universities themselves came up with the principles and the first EA artefacts. Consequently, there was no well–scoped architecture governance process or agreed–upon vision. Combine this with a lack of common ground, save the TOGAF adoption, and most institutions diverted from each other. Another reason for concern was the IT department's overall influence. The IT affiliates attended TOGAF courses and were trained in architecture, whilst the business and top management had very little insight into architecting. Someone notices how the institution expected employees to be architects after such a course, whilst any good architect needs to have experience and understands how the Organisation works. Consequently, the study states that only a tiny number of institutions had success with EA. Furthermore, one should address the challenges beforehand if a joint EA's development is to have a larger percentage of success.

Additional studies into information systems structure in the education sector. Further, two studies propose a unified information system description, to be used within the higher–education sector [79] [28]. The proposed descriptions are based on literature research as well as interviews. These studies underline that the educational institutions, even internationally, operate the same. Hence, the structuring of processes and information systems opens opportunities to standardise the educational sector within a country or even internationally. However, the practicality of the proposed architectural descriptions is limited, as clear definitions to concepts, symbols and relations within the architectural description are not given. Moreover, there is no viewpoint or clearly defined requirements or any other form of stakeholder concerns.

EA adoption in the Indian health care sector. Lastly, Anjali and Aparna notice how hospitals, in theory, should operate the same but are unnecessarily complicated in practice. To counter this complexity in hospitals, all hospitals' organisational foundation should be built on a shared strategy and vision. From this starting point, EA was used to achieve centralisation with improved decision making, lower costs for procurement, increased reliability of IS records, improved traceability, connectivity, and access to data. Furthermore, EA opened to door to the automation of processes, cut costs and improvements in quality. The study also notes the importance of clear EA objectives as it guides the process. As well as proper training to all institutions to ensure full participation in the project and adopt the project products. Further, details on the description development process is not given. [44]

3 METHODOLOGY

This study investigates architectural description development in a real–world setting by reporting on the process and deliverables in an organisation's architecture development project. As stated earlier, a case study is held in which one observes the work done. Moreover, the researcher actively participates in the development project to experience the process. Throughout the process, several interviews and questionnaires will be undertaken whilst documenting the process and gathering relevant deliverables. This qualitative participatory approach, supported by interviews and questionnaires, is chosen as EA development is often fuzzy and complex. A case study is recommended for a research domain characterised by blurry definitions and context based on an empirical inquiry [86]. In the research findings, a case study in the public education sector is presented, based on said approach.

During the participation in the architecture development process, the focus lies in the process undertaken. Subsequently, the relation of the perceived process is compared to what the theory prescribes.

During the process questionnaires, interviews are held to help answer these questions, as well as how the project members have come to their conclusions. Moreover, the process can be divided into four main chunks: the process to identify stakeholders, the approach to elicit concerns, and how they create and deliver the corresponding viewpoints and views. By becoming an overt participant in the project group responsible for developing the reference architecture, including viewpoints, we hope to observe how professionals would execute their work for each chunk. Subsequently, we record events related to arrangements and agreements about the work order and activities undertaken. These events will be recorded using shorthand codes in the central repository, such that analysis should be relatively quick. Furthermore, this 'observing the work' will be carried out through the entire length allocated for the project, namely one year.

4 DATA

During the architecture description development process, two questionnaires were conducted. The first questionnaire validated the stakeholder groups as identified by the project group earlier that year. Fourteen of the eighteen project members have answered the questionnaire. In the same questionnaire, project members were also asked to rank the different stakeholder groups based on relevance and importance concerning the project's vision. In the second questionnaire, the top five stakeholder groups were asked about architecture, their vision on the project, their concerns and their preferences.

Moreover, fifty people were asked to take part in the questionnaire. Furthermore, thirty–five stakeholders have answered the questionnaire. Furthermore, there have also been interviews with each stakeholder group to further elaborate on the questionnaire and the given answers.

During the last month of the project, six interviews with senior project members reviewed the development process and created architectural products, including any follow–ups on the development cycle. Each interview lasted approximately 45 minutes and was recorded & transcribed.

Besides own notes and experience, 52 documents used within the development cycle relevant to the development process were also gathered during the case study, to support the development process analysis. A full data representation is given in Table 10 in the appendix. In this table, every document has gained an assigned ID, following the mask $D\text{–}[0\text{–}9]\{2\}$. Moreover, each record has a class, archive, internal or external, and a short description. In figure 13, the documents are described in which phase of the project; they were more relevant than in other phases.

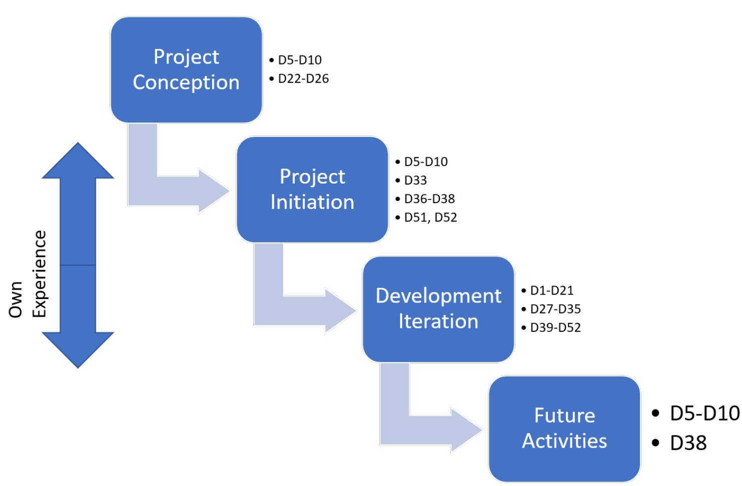

Fig. 13. Data sources related to process phases.

5 CASE

The proposed case study approach is conducted within the Dutch public sector, or more specifically within the public sector's education domain. We participate in the vocational education institutions' architecting process within the Dutch educational industry, developing a new enterprise reference architecture.

In order to further introduce the case, we first describe the environment of the Dutch educational sector and its relation to the international known educational structure. Secondly, we briefly describe the maturity of EA within the educational sector, followed by the reason the vocational education institutions are renewing their standard reference architectural descriptions. After which the project development team, its organisation and our role within the project are discussed.

5.1 DUTCH EDUCATIONAL SECTOR ENVIRONMENT

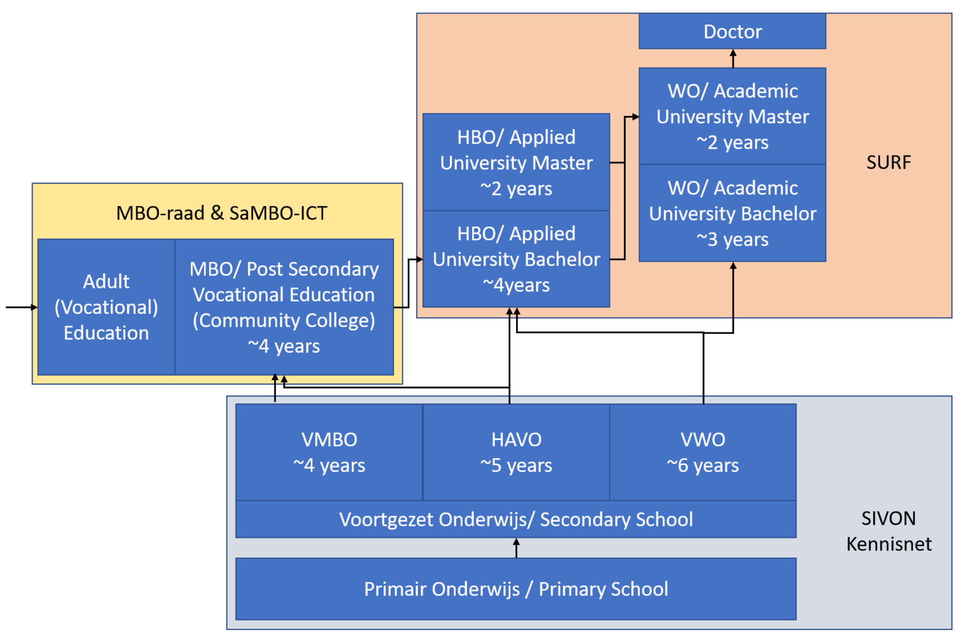

Fig. 14. A simplified sketch of the Dutch educational sector, the naming convention is Dutch term/ English equivalent. The three domains distinguish the three different advocacy and collaboration organisations, MBO–raad & saMBO–ICT, SIVON & Kennisnet and SURF, in the educational sector.

In figure 14, one can see the simplified Dutch educational sector and its flow of students [80]. The model is divided into three domains, a domain for the primary and secondary school, a vocational education domain, and a higher–education domain. Each domain has an advocacy organisation that strives to enhance collaboration within the environment, advocate the field towards suppliers, and consult individual educational institutions. The entire sector falls under the ministry of education, culture and research or abbreviated in Dutch 'OCW'. The case study is done with saMBO–ICT in the post–secondary vocational education, which we will refer to with the Dutch abbreviation 'MBO'.

The MBO provides several education programs such as welding, economics or farming. These programs are referred to as sectors, and an educational team organises each sector. An educational team can consists of lecturers, operational managers and supporting roles, such as quality managers. Furthermore, The MBO differs from the higher–education domain, as there is an increased emphasis on practice. Consequently, the education program consists out of theory, practice in class and mandatory internships with companies. Although they do practice some empirical studies, which they call 'practoraat' [70], it is not common in the education programs. Another significant difference compared with the higher education is, that students who start at the MBO are around the age of sixteen. Which means the students are still at school–going age, also referred to as compulsory education. Consequently, the MBO domain must actively make sure students attend classes, and they have 'duty of care' towards the students. 'Duty of care', or 'zorgplicht', is enforced by the OCW and means that the MBO must actively pursue the student's mental and physical wellbeing and, if necessary, provide aid such as a psychiatrist.

5.2 DUTCH EDUCATIONAL SECTOR ARCHITECTURE ENVIRONMENT

The Dutch government has pushed heavily on a complete digital transformation of the public sector [37] [71]. One of the many initiatives to achieve is through enterprise architecting [38]. Moreover, the Dutch government has a central reference architecture called the 'Nederlandse Overheid Referentie Architectuur' or NORA for short [36]. Each public domain has its specialisation of this base architecture. Consequently, the educational sector has a chain reference architecture called ROSA [11]. The architectural board is the product owner of the ROSA.

Each collaborative organisation within the educational sector has its reference architecture derived from the ROSA for their particular domain. For higher education, this is the HORA maintained by SURF [83]. For the primary and secondary school, it is the FORA supported by Kennisnet [46]. For the MBO it is the triple–A [78] maintained by saMBO-ICT. The latter, triple–A, is being replaced by a newer reference architecture. The case in this research is about the architectural description development, which will replace the triple–A architecture description.

One of the Dutch public sector principles is to share an architectural description freely for anyone to use. Additionally, Felix et al.[79] came to the same conclusion during their research. This open access policy has heavily influenced the development process.

5.3 PROJECT ORGANISATION

In figure 15, one can view the project's organisation. Note how sambo–ICT is the client and has two groups lower in the hierarchy which will influence the project. On the one hand are the information managers and on the other hand, is the internal consulting group. The latter is a set of advocacy groups representing chains of processes found within the MBO, such as education support and logistics. They provide feedback, raise concerns and provide a vital role in accepting the delivered products. Below are the project steering and publication group, which we will refer to as the core group. This group consists of those that coordinate or directly support the project, such as project managers, external consultants, and external researchers. The final size of this group is around three members; this researcher included. There is also a sounding board, referred to as the Leusden group; they provide feedback and raise concerns. The Leusden group members have developed or were directly involved with the previously create architecture descriptions, such as the triple–A. Moreover, this group exists out of managers, information managers, representatives from the educational institutions amongst other backgrounds, they are the ones who have committed themselves for a new reference architecture description at the beginning of 2021.

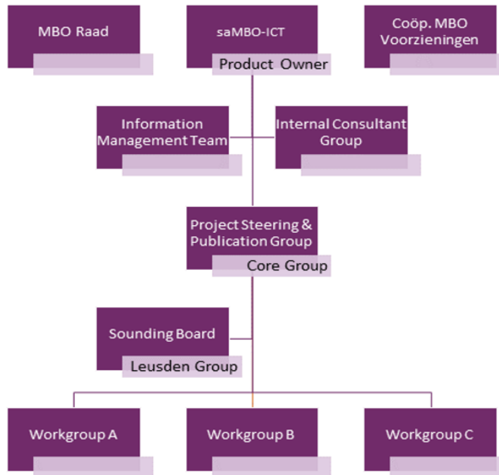

Fig. 15. Project groups' organisational chart.

Lastly, there are three workgroups A to C. Each workgroup had an average of six members. Each workgroup had a dedicated core group member who participated in the group's process and coordinated the workgroup if necessary. Furthermore, in the appendix table A4, one can find the project team composition. To summarise:

- Leusden group: The group committed to delivering a new reference architecture description and acts as a sounding board during the project.
- Core group: The group consisting of project managers, change managers and matter experts, such as researchers and consultants.
- The workgroup: MBO sector experts with process thinking affinity. They develop the architecture descriptions incrementally in short iterations.

6 RESULTS

The results of this study are given in chronological order. The project conception started in May 2019, its preparation began in September of 2019, and the development process began in February of 2020. Furthermore, this researcher joined the project at mid–February 2020 and stayed till the beginning of January 2021. Finally, the reference architecture description's first publication would be at the end of January of 2021. After which a new development cycle will start. Additionally, table 7 in the appendix provides a short overview of events.

6.1 PROJECT CONCEPTION

In this section, of the project is reported. In this phase, the project principles, budgeting, scope, planning, tooling, organisation, information and other preliminary matters are arranged.

The MBO sector most recent reference enterprise architecture description 'triple–A' dated from 2007 and was scoped around the education support processes, such as scheduling, administration and education program development. Areas as the actual teaching, the examination and supporting business functionalities, such as finance, were out of scope. Additionally, the triple–A architectural description is based on Kruchten 4 + 1 [47]. The architectural description describes a logical and process view tied to several scenarios; triple–A refers to these scenarios as use cases.

The education teams, those responsible for executing and organising the education courses, had little use for the triple–A, due to the limited scope. As their work has become more complex, they felt the need for architecture descriptions. Consequently, the advocacy group for these education teams within the MBO–raad started developing an architectural description about teaching, tutoring and examinations. This architectural description is referred to as the TPO [27]. Despite the different scope, goals and stakeholders, the architectural descriptions sometimes describe the same concepts but do so differently in syntax and semantics.

The advocacy organisations, sambo–ICT and MBO–raad, noted that this was an undesirable situation. They felt that the goal of reference architecture descriptions should be to connect different MBO education institutions on many subjects by offering one structure, one language and one straightforward communication instrument. This common ground is a requirement if individual institutions with different solutions and systems are to cooperate. During the discussions on these discrepancies, the urgency to collaborate and to centralise the sector increased rapidly. Several developments fed the sense of urgency.

The first and foremost development was the strategic agenda of the MBO sector to increase digital maturity in the industry, aptly named Digital Agenda 2018–2022. Such a plan consists of a finite set of sections or branches, and one such section is 'leven lang ontwikkelen' translated 'lifelong development'. This branch seeks to modularise and decouple the MBO sector based on educational demand from potential students. At the base of this modularisation is the student's journey. Where a student starts with an education program, which the student tailors to his liking, in this customisation, the student should be able to weave several education modules together. Note that one module can be given by a different institution than the next module. Furthermore, another five years later, after graduating the student should be able to come back and participate in another module to his liking to hone his skills further. This modularisation has several consequences amongst which is the requirement for institutions to share one central student file, which the student keeps during his career.

This modularisation is not only a section on a strategic agenda, but it is also an apparent demand from the market. Moreover, the governmental employee insurance agency (UWV) has promised to facilitate a STAP–budget [84]. This large pool of resources is used to enhance the individual's development of the employed and unemployed in the Netherlands. Where each citizen gains €1.000, – a year, such that he or she keeps learning and developing oneself, to achieve this the citizen should be able to select several educational modules to undertake from the STAP platform. Note that the MBO sector was always the go–to sector for adult vocational education. Consequently, if the industry wishes to keep its pedestal, it needs to modularise and decouple its education programs, such that it able to join the STAP–platform.

Other developments include the increased security and privacy requirements, such as the GDPR, and the demand to be more efficient, a.k.a. do more with less. Furthermore, there is a need to have a centralised acquisition and procurement of ICT and other resources. As of now, most institutions make their acquisitions individually, resulting in a lack of standardisation within the sector. Consequently, there is a difficulty when cooperating, due to the different ICT solutions. Moreover, smaller institutions do not have the resources to undertake prolonged acquisition processes. Hence, their ICT solutions are often out of the box and not a perfect fit in the business.

One of the many disciplines which should aid the MBO sector in these developments is enterprise architecture. Consequently, sambo–ICT, some MBO institutions and the MBO–raad set out in May of 2019 to align and harmonise the TPO and triple–A on the short term, at the end of 2019. Whereas in the long term, at the end of 2020, there should be one reference architecture describing the most important aspects of an average MBO institution. This resulting reference architecture is to replace the previous architectural descriptions triple–A and the TPO. In this case research, we focus on the latter, namely the reference architecture description development. Since the goal was to have a new adopted reference architecture at the beginning of 2021, the development project was named Route21. On a side note, the group which conferred these goals would be later be called the Leusden group.

Later in that year, the project plan for Route21 was created. The rationale, environment, budget, planning, quality, scope, communication, and project principles were stated in this plan. One of the essential project principles defined in this document was the requirement of adoption by institutions and the end product's uniformity. The reference architecture description was created on expert opinion, experience, and input from the institutions to achieve the set goals. The institutions were to decide whether the completed work is accepted. This acceptance by institutions requires that the reference architecture description adds value in the eyes of the institutions.

One of the problems with the other architecture descriptions was described by some as "being too academic". What they meant was that the descriptions were too formal and strict. Common terminology used in the field was not used in the architecture description, since the actual definition diverted from the field's used working definition. Another example was that the presentation of the architecture descriptions followed IT standards. Although this sounds fine on paper, it meant it looked unappealing.

Moreover, employees without IT affinity struggled to read the descriptions. As a result, the challenge would be to balance correctness & formality with readability & pragmatics. Another principle was that the reference architecture should cover all process domains within MBO institutions. These decisions were made for the advocacy organisation since it did not want to have new reference architecture descriptions appear because a vital stakeholder was left out of scope.

There was another project principle to align as much as possible to relative sectors as well. This alignment meant to be developed architecture description should consider the pre–existing relative sectors' reference architecture. As a result, the closely related sectors should connect to the MBO sector as much as possible. Examples of relative sectors' reference architectures are the ROSA (OCW), HORA (higher education), FORA (Primary & Secondary education), RIO (Information architecture of the ministry), Edustandaard (standards for the education sector in the Netherlands) and if available international or non–education architecture descriptions. Moreover, the resulting description should be useable in a few use case scenarios such as procurement, base architecture for other projects, a communication instrument when discussing operations & responsibilities, and a way to structure the individual institutions.

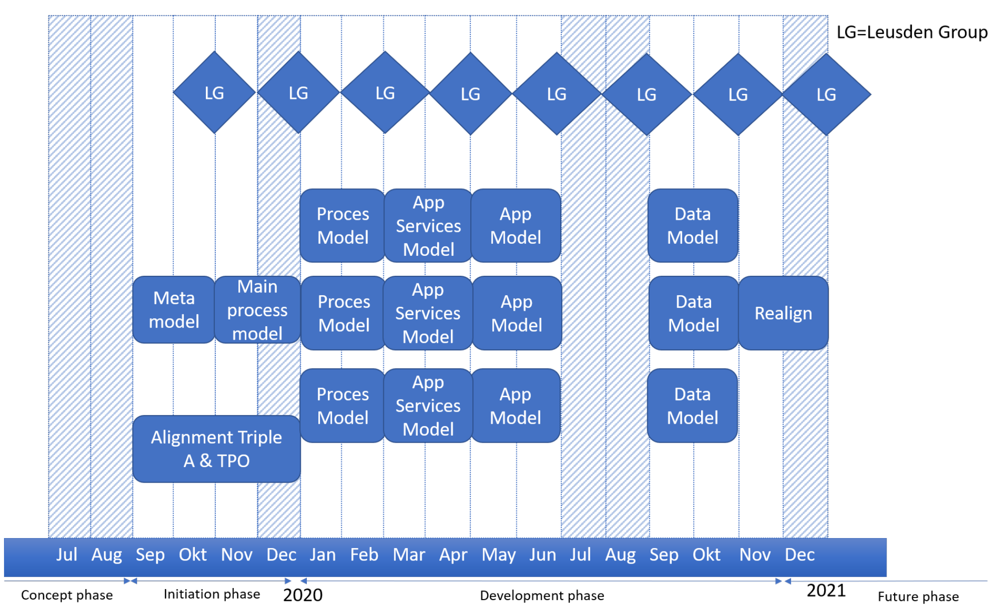

Fig. 16. Initial project planning.

The last primary principle was to involve students and researchers. Reasoning that even such a business project can function to help & develop people. Additionally, students & researchers often have a different perspective and more recent theoretical insights, which the workgroups can utilise in their creative process.

Additionally, it was decided that the metamodel should have an extra layer of abstraction. Usually, one speaks of business functions such as HRM, which consist of a set of processes. However, previous architecting learned that the use of business functions leads to confusion in MBO institutions. For example, the business function HRM may be responsible for allocating resources to education teams in one institution. In contrast, in the next institution, it is facilities or the education teams who are responsible. This confusion exists since many business functions are confused with departments with the same name. Rather than starting a campaign that seeks to educate all possible stakeholders on this distinction between a department (actor) and a business function, the project group chose the pragmatic option of altering the metamodel by adding the extra level of abstraction main processes. Consequently, rather than stating a business function, such as HRM, one would note high over processes commonly found in an HRM function, such as on– & offboarding, managing wages and developing personnel. Additionally, the concept of main processes is a common one in the MBO sector.

Beside project principles, the project plan also states the project planning. In figure 16, one can find the initial planning for the project. Note how there is a six–week session with the sounding board Leusden group and that every two months a description is delivered. Also, note that one begins with the meta–model development and that of the main processes. Recall that main processes are comparable to business functions but are renamed and further specified as the architecture description stakeholders find the distinction between functions and processes difficult. The main–processes description was to be used as the common ground from which the three workgroups would develop their architectural descriptions. Each workgroup was to be responsible for some regions of the main–process description. These domains were further specified into subprocesses, application services, application components and business objects.

During the triple–A & TPO alignment and harmonisation, sambo–ICT had a researcher investigate architecting's maturity within the MBO institutions [88]. Especially in terms of perceived benefits, existing artefacts, use of base architecture descriptions, use of reference architecture descriptions, used modelling languages, tools, as well as working principles and situations where architecture was used. The result of this research was input for the requirements, concerns and goals of route21, as well as the meta–model that was to be created. Moreover, the research report suggests that architecting is still in the beginning phase in most institutions, where some institutions are entirely new to architecting. However, all institutions have started or started a while ago with architecting.

6.2 PROJECT INITIATION

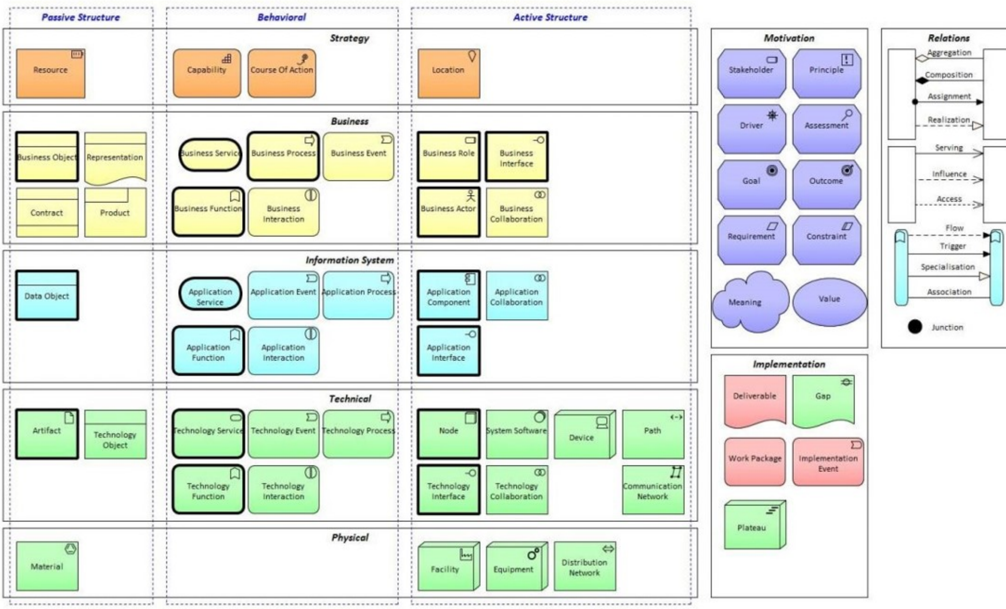

Fig. 17. Archimate3.0 ontology [67].

In September, the meta–model decision and the official architecture description language (ADL) began. It ended later than planned at the beginning of January. A recommendation of the architecture maturity research was to adopt an architecture description language formally. Rather than reinventing the wheel, it was decided to adopt the standard used by the other education sectors and when necessary, tweak it to the needs of the MBO. The architecture description language adopted was Archimate3.0 [67]; a description of the Archimate language is depicted in figure 17. Other findings from the same research suggested that institutions wanted to use architecture primarily to communicate with colleagues from a different team, department or institution.

The process in September began by reiterating why there is a need for architecture. After that, the project's goal, as stated in the project concept, was emphasised. Figure 18 was utilised to convey the message. It shows how the project was to deliver a reference architecture recognised by several stakeholder groups, with each their view. Moreover, all those views were to be from the same core architecture, also referred to in the project as the cube. This cube should result in architecture descriptions understood by everyone. Each description served in one of the stakeholders' language, but still sharing the same fundamental concepts. That is to say, a common ground. However, in the past architecture, triple–A, had shown that stakeholders with non–ICT affinity had severe troubles in comprehending models build from formal ICT languages such as UML and Archimate.

Consequently, the views created with the adopted ADL Archimate3.0 were no good to communicate to non–ICT stakeholder groups. Therefore, it was decided that Archimate3.0 described the cube, but a professional designer described the views. This designer was to create attractive visualisations for non–ICT stakeholder groups.

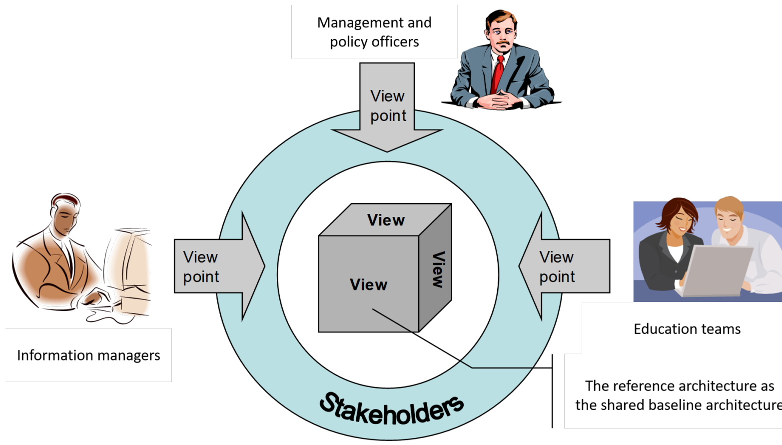

Fig. 18. Image used to convey the idea of viewpoints and views to non–IT stakeholders.

Furthermore, the information managers and architect conference had to identify the following requirements to reach the project goals:

- The stakeholder groups.
- Their concerns.
- The required viewpoints.
- To be developed views.

These requirements were elicited through several techniques, namely perspective–based reading, brainstorming, expert opinion and focus groups. For the perspective based reading the project group used several documents, which were used to create the previous architectures, triple–A and TPO. The triple–A had several scenarios which proved useful as well. Another primary source was the adjacent education sectors of higher education and the primary/secondary education. They either had an already established reference architecture or were amid developing a reference architecture themselves. Moreover, a contributor to the higher–education sector reference architecture was invited to present their process and findings, as well as to formalise the relationship between the to be developed MBO reference architecture and that established higher education's reference architecture HORA.

The findings were presented later in a conference of several information managers, architects and other functionaries on the middle x–axis of the model of Maes [52]. Several things were explained at this conference, such as the prior architecture, the chosen ADL, several meta–models, among which was TOGAF, and the existing reference architectures of the government and other education domains. During this conference of experts, the stakeholder groups were defined and their concerns. These concerns were determined according to the read documentation and expert opinion. They distinguished eight stakeholder groups:

- Students and clients
- Education teams
- Education support (Education program development, scheduling, resource allocation, internships, administration and resource allocation)
- Secondary business functions (Finance, HRM, IT, etc.)
- (Top) Management

- The structure groups. (Architects, analysts, information managers, etc.)
- Quality assurance. (Controllers, data protection officer, security officer, inspectors, etc.)
- External. (Suppliers, ministry, local government, community, etc.)

After that, the conference group was divided into three workgroups, and each workgroup was to define:

- Why the stakeholders wanted a new reference architecture?
- When were the stakeholders satisfied?
- What is the result?
- How should one set out to achieve the above?

As was in line with the project concept mentioned earlier, they found that if the MBO sector cooperated on topics such as student exchange, modular learning, centralisation, GDPR and procurement, then there is a need for shared principles and semantics. It would then behove the stakeholder groups to have a common reference architecture with a set of traits. First and foremost, the reference architecture must be flexible, a product in continuous development, and regular updates. This requirement is a lesson learned from the triple–A usage since triple–A was used less and less through time. Moreover, the reference architecture must be updated with scenarios and use cases of the latest issues, for example, the centralisation of the student information system. Secondly, one should work from the top–down, such that most stakeholder groups will be able to follow the development process comprehensibly. The reasoning here is that stakeholder groups such as education and management have more process affinity than, information or infrastructure affinity. Lastly, the stakeholder groups are satisfied when they can use the reference architecture to explain to a layperson how the MBO operates, which should entail that the reference architecture is useable to aid in the decision–making processes, such as centralisation of the MBO sector or procurement.

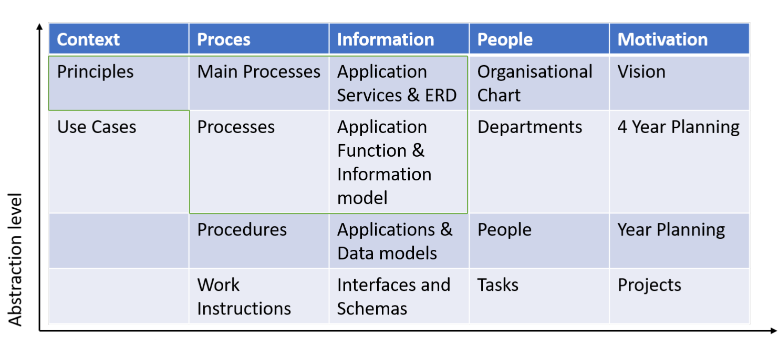

Fig. 19. A framework used in the project.

When deciding what should be modelled, the workgroups used the presented metamodels and the ADL as reference. Furthermore, the already established work, such as triple–A, TPO, HORA, FORA and RIO, was used as a reference as well. From this, they created their table, presented in figure 19. Besides main–processes, other pragmatic choices were made as well, such as striving for maximal alignment with the secondary and the higher–education, by reusing their work when applicable. Combining this with lessons learned and the created work from the older architecture descriptions, meant that the development time was shorter, due to the abundance of available information. Moreover, the stakeholder groups would get an expected result.

The result was that the agreed–upon rudimentary framework in figure 20. In which six main–viewpoints were distinguished, the main–processes, principles, the processes, the application landscape, the logical information model and several scenarios' given form in use cases. This framework, save the logical information model, was also used in the past during the previous development of architectural reference description.

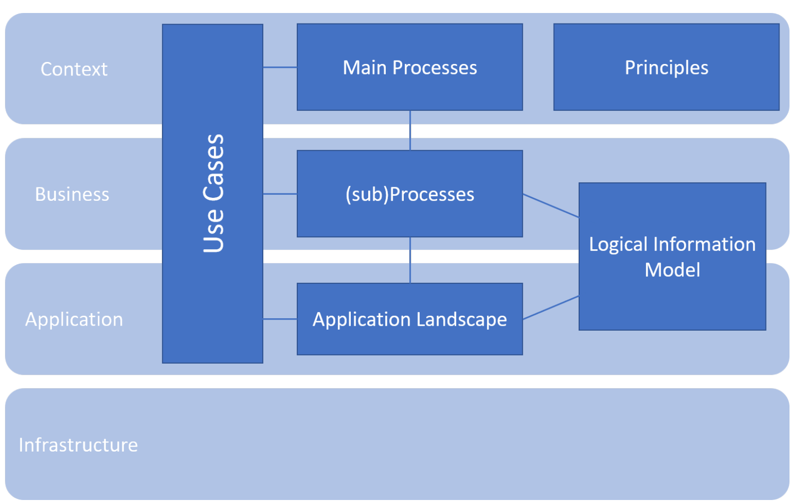

Fig. 20. Used primary viewpoints.

Furthermore, this framework is something which is utilised in the other educative domains as well. Note how the infrastructure layer is empty. The individual MBO institutions deviate considerably in ICT maturity and automation, underlined in the prior conducted research. Consequently, if the reference architecture were to prescribe the infrastructure layer, one feared that it would alienate several institutions from the newly developed reference architecture. This fear also meant that the application landscape was only to name application services and not functions or components. Another principal reason not to model the infrastructure and application components was that one believed most stakeholder groups were not interested in this information, at least not yet.

Based on the created framework, the architecture description language was simplified into the meta–model visualised in figure 21. Creating the meta–model, in which stakeholders, concerns and main viewpoints were defined as well, ended at the beginning of January 2020.

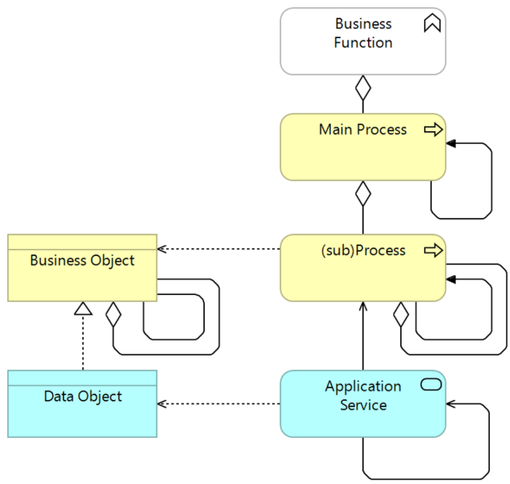

Fig. 21. Initial meta–model.

According to the initial planning, the development of the main processes should have started in November. However, the previous step was delayed, and December is a month with many holidays. Hence, the main–processes' development was postponed until February of 2020. During 2020 the project would have a different team and no longer the conference group of delegated information managers which was used until now. The development process from February 2020 until December 2020 is described in the next section.

6.3 PROJECT DEVELOPMENT ITERATIONS

In January of 2020, the process of the meta–model creation was finished, as well as another sambo–ICT conference in which Route21 was actively promoted towards information managers and business managers alike. During this month, covid–19 became prevalent within the Dutch borders. While the government's initial response was that of caution, in February it became clear that the initial planning with physical meetings, numerous conferences, and direct stakeholder group involvement was no longer possible. The reason for this was the inevitable lockdown of the country. During February, a new project planning was created, and the start of the development was postponed. In March, the country went into lockdown. Furthermore, the development process started, albeit entirely online and no longer with the Leusden group as a sounding board every sixth week, as this required physical contact to function correctly. Subsequently, the Leusden group's frequency became every six months, combined with the already planned conferences of sambo–ICT, which occurs once every half year.

The first thing the new development team did, was to take the earlier inventoried architectural descriptions and create building blocks from them. The inventoried architectural descriptions included:

- triple–A [78]
- TPO [27]
- ROSA [11]
- FORA [46]
- NORA [36]
- MBO–raad taxonomy [54]
- Process architecture examination [61]
- TIER [39]
- CAUDIT Capability Model [19]
- Several internal architecture descriptions from MBO institutions
- RIO [45]
- Amigo [25]
- HKS
- Edustandaard Documents and policies [24]
- A scientific paper [79]

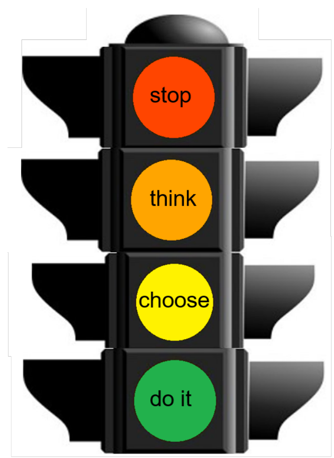

Fig. 22. Traffic light method.

In order to turn these architecture descriptions into useful building blocks, the project team used what they called the 'traffic light' method. The traffic light, visualised in figure 22, distinguishes four colours:

- **Red** Stop: This hopelessly dated, untrue or for any other reason not to be used.
- **Orange** Think: This is useable but requires considerable (re)work.
- **Yellow** Choose: This is useable, but there are other equally valid options as well
- **Green** Do it: This is something we should strive for or something which is directly adoptable.

In the traffic light method, one first defines the business domain on which there is unanimous agreement and subdivide them into architectural layers. These domains were education, examination, education administration and supporting, visualised in figure 23.

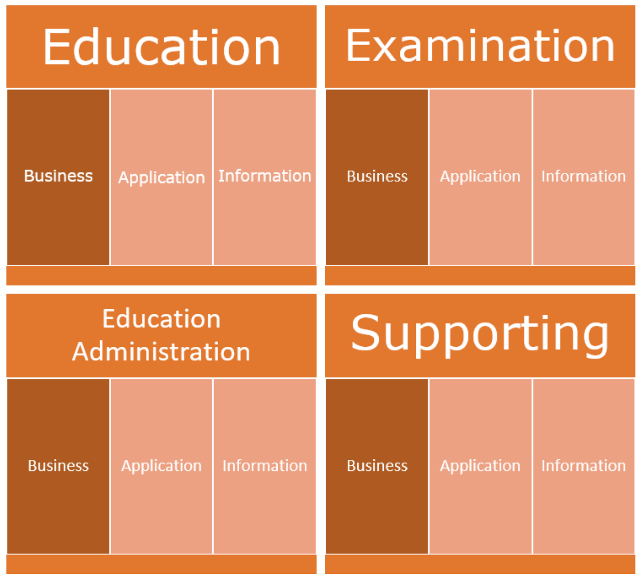

Fig. 23. Used business domains with architectural layers.

Next, a moderator takes the project group through all the inventoried architectural descriptions, based on the defined business domains and the architectural layers (if they exist in the presented architectural description). The project group is then allowed to react to the illustrated architectural description if and only if it has any direct relation, in their opinion, with the current treated business domain. A reaction must be a colour of the traffic light, visualised in figure 23, with a rationale. A response can be on an element, a relation or a pattern. For example, the pattern of the process 'conducting examination' following the process of 'creating an exam' might be flagged as red, since any exam must first be evaluated. Note that not everything in an architectural description needs to be discussed, being indifferent is a valid option as well. During this process, a dedicated stenographer is to document what is being said and which colours are being awarded. The template used by the project team to document this process can be found in the appendix, table 12.

From the resulting documentation of the traffic light session, one starts the following process. First, for each domain and architectural layer, one sets the architecture description with the most points, where green is two points and yellow one point, as the starting point. All other objects, relations, and patterns evaluated from other architectural descriptions are related to the elements, even none evaluated ones, of the starting point. As a result, one now has a work product with lays one architectural description on top of another, pinned on shared topics. Consequently, the work product shows similarities and discrepancies; a visual aid is provided in figure 24.

From the work product, one can start combining found objects, relations and patterns into building blocks which can be used later in the process. The colours of the traffic light give the priority in connecting.

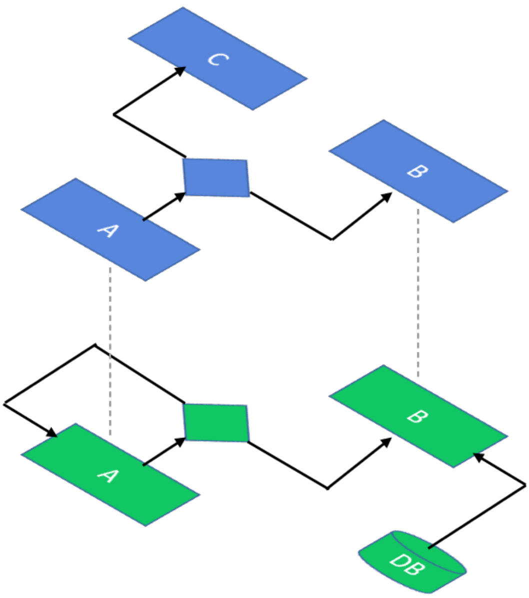

Fig. 24. Example of laying models on top of each other, pinning on element similarities.

For the project, this meant that they pinned other architecture descriptions on the triple–A for processes in the supporting and administration domain and TPO for the education and examination domain. For applications, the higher–education (HORA) architecture description was used as the starting point and the reference architecture of Edustandaard (RIO & Amigo) as the starting point for the information model for all domains. The resulting building blocks were saved in a spreadsheet since the tooling with repository was not yet functional.

THE CREATION OF THE MAIN–PROCESS MODEL FRAMEWORK The creation of the main–process model started with a call to all institutions to send in their preferred model, which they used for business functions or main–processes. Moreover, the said description should be accompanied by information such as the primary users, the goal, which abstraction level was used, which processes were considered on a 'must–have' status, which relation types were used and other relevant rationales for the send model. The period of awaiting and gathering main–process models lasted for two weeks.

The next interaction of the project team consisted of a small ISO–42010 workshop and a recap of the developed building blocks. After the workshop, an interactive session was planned in which the send main–process models were to be evaluated. The session was organised as follows: a moderator would then present each sent in main–process model, including the rationale. After each proposed model, the remaining project team members were asked to fill in a questionnaire evaluating the presented model on presentation and content. A translated version without mark–up is provided in the appendix. This questionnaire was to input the next phase, creating a framework for the main–processes model.

The main–process framework was created rapidly as many sources were already available, such as the generic building blocks and the questionnaires. There were also a set of principles which guided the development of the framework. The main–process model was used to aid in determining ownership, setting boundaries of projects, aid process optimisation, giving insight into processes dependencies, the student's education journey, and providing insight into business domains. Furthermore, the stakeholder group management was to be the leading stakeholder group of the main–process model.

The main–process framework was created in an interactive session with the entire project team. An English reconstruction of the framework is provided in figure 25. The main–process framework consists of business domains, which were to be filled with the main–processes. Notice the similarity with Porter's value chain. This similarity is done for several reasons. First, a value chain is a simple model which is known by most, especially with management. Secondly, the student central approach entailed a natural value chain since the education sector adds value to students. An interviewee noted that: "When one asks why do we do it? Then the answer is for the students. When one sets the (student) stream central, it is only logical that a value chain emerges." Finally, the project team felt that the value chain creates natural process domains, handy for process optimisation or delegating ownership. This reasoning from chains is common practice in the MBO sector. All with all, this decision further enhanced the main–process model's recognisability in stakeholder groups' eyes.

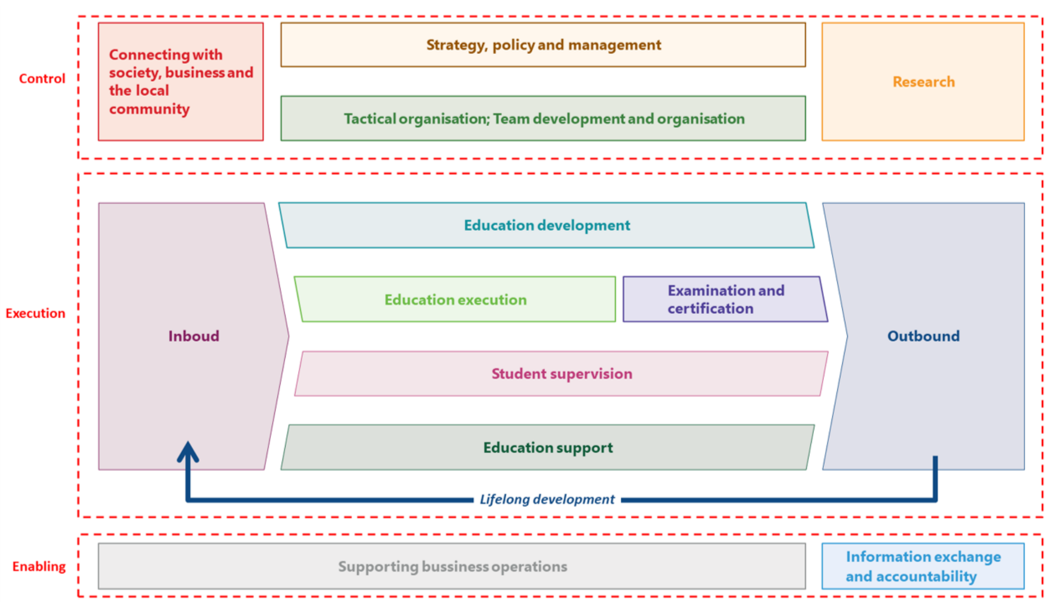

Fig. 25. Main process framework [75].

ORGANISING THE WORKGROUPS After creating the first draft of the main–process framework, the project group felt that there were too many members, to keep developing the main–process architecture centrally. Consequently, they opted to start dividing into workgroups in April of 2020. There were several ways to form workgroups, based on domains, based on architectural layers or based on stakeholder groups & concerns.

Dividing the project team into workgroups based on domains linked to the foremost stakeholder was the concept phase's original plan. However, the number of business domains in the main–process framework proved to be higher than first believed. Consequently, workgroups were to be formed based on a set of domains. However, the project team members admitted that they found it quite challenging to think of processes inside one chunk of the total picture. The same problem applied to the division based on stakeholder groups and their concerns. Additionally, there was difficulty maintaining correspondence between the different architectural descriptions. The latter was made more difficult as the concerns of the stakeholders were varied and far apart. An interviewee noted that "… limiting oneself to one domain while being on different abstraction levels is difficult. Since a process on some levels will never be limited to one domain. Since it is part of a chain of activities that cuts through several domains. In architecture, that is exactly what you want to make insightful". The project manager noted that this approach has been successful in the past. The workgroups would create part of a description in the morning for their stakeholder. In the afternoon, all the workgroups met and combined their models to honour correspondence and learn from each other. However, he noted that this approach was made very difficult, as it requires physical contact to properly work as people need to anticipate with each other and work around one large whiteboard, something that does not work well with online sessions. Consequently, the workgroups were divided based on architectural layers, namely:

- principles, business functions and main processes,
- (sub)processes,
- applications and information.

Another reason for this basis of the division was that each category requires different affinity and skills. The high over modelling of the main processes requires other people than those who model applications and logical information models. Furthermore, the reason why applications and information were not divided further into separate categories was due to the number of people in the project team. The further division would result in a too–small workgroup. Lastly, an added benefit of having one group is responsible for one architectural layer is that descriptions on that layer are concise and have implicit correspondence. To summarise, there were three workgroups created based on architectural layers; these were:

- Workgroup A: is responsible for the context. (Business functions, main–processes and principles).
- Workgroup B: is responsible for the business (Processes).
- Workgroup C: is responsible for business support (Applications and information).

The workgroups were to work in sprints lasting two weeks. At the beginning of each sprint, at the same time the end of the previous sprint, a meeting was planned lasting three hours. The first hour was a plenary session of all workgroups together. During this plenary session, every workgroup presented their work and their plans for the next sprint. At the same time, other workgroups gave feedback and pointers.

Furthermore, the core group gave announcements and updates on topics related to the project, such as stakeholder group interest, questions or developments in the strategic agenda. The last two hours were for the workgroups themselves, to plan, organise and work. Note that the workgroup was to decide their planning. In the planning and working methods, the workgroups were entirely free to choose. As a result, the taken approaches of each workgroup could differ. The working methods of each workgroup are presented in dedicated sections.

Moreover, the workgroups did not get any strict planning stating what should be achieved in each sprint. Instead, they were given a deadline; in December 2020 we want to have the architectural layer you are responsible for filled with architectural descriptions, fit for purpose, and aligned with the other workgroups' products. As a result, the workgroups themselves planned the sprints and what should be done to achieve the goal. The project's coordination and management were not on time nor the products themselves, but on the project team and their purpose. This management approach has two reasons. First, there is no functional design, especially in a creative process. The project manager noted: '… No one knows what they want and when they want it nor why.' Secondly, one should maximise group efficiency, not time efficiency, nor cost efficiency. The goal was to deliver qualitative architectural descriptions fit for purpose, not to be complete or cheap.

One of the ways this group efficiency was maximised was in recruitment. The project recruited members during the entire course of the project. One was free to join if they wanted too, money was not a primary concern, willingness was. By actively marketing on many occasions and selling a clear vision, the project tried to recruit enthusiastic members about sharing the same vision. The project manager noted that 'You should sketch an image, a vision, something to strive for. That point on the horizon you should hold onto, to guide and steer the group towards that point'. This shared vision maximises group efficiency, the goal of the management team is not to dictate, but to keep the spirit alive through guidance, aid and clearing hurdles. The group themselves are capable enough set intermediate goals and organise, but this requires trust from management, as well as from other group members. The faith that everyone does their best to reach the goals and only have good intentions. In other words, make sure that productive workers keep productive. That is not to say that there will not be project members who perhaps struggle to find their role or do not know what to do, then the core team steps in and offers guidance or whatever else is necessary to get the member to be productive. If that fails, then one should be pragmatic and kindly ask the member to take a step back. The result is that the workgroups were free in Route21.

While the workgroups were free, there were some agreements and advice given. First, it was agreed that TOGAF was to be used as something to fall back upon when arguments or confusion would arise. Something which was referred to as 'TOGAF light'. Secondly, the main–process framework was to be used as the starting point to aid correspondence between workgroup products. From that point on the primary process model framework was informally referred to as the coat rack, a manner to hang and relate the different architectural descriptions. Third, it was emphasised that the architectural descriptions are not only diagrams and visualisations, but spreadsheets, matrices, and information crosses as well. Lastly, the final word lies with the core group, as they were the once responsible for guarding correspondence and ensuring insights were shared. To achieve this correspondence, every workgroup should at least have one member in the core group.

During the workgroups' organisation, the discussion and procurement around tooling were still ongoing since the project team could not wait any longer; it was decided that cooperation would occur via Microsoft's office365. The development would take place in the open–source tool called Archi, which would use a git repository to easily share the architectural storage with version control. Furthermore, the final product's publication was to be on a semantic wiki, provided by ArchiXL. This semantic wiki was something which was done in the past and was received very well with stakeholders.

Another topic which arose during the organisation of workgroups was that of stakeholder groups and concerns. The project team felt it was not enough to rely on expert opinion and existing documentation. They reasoned that an investigation should take place into the stakeholder groups and their concerns, as well as more direct stakeholder involvement during the development, such that representatives of stakeholder groups could provide early feedback on the main–viewpoint and views.

The workgroups began in May of 2020. While not everything was arranged yet, the workgroups started under the rule that one will tackle the problems when they arise and improve the next sprint's process.

INVESTIGATING STAKEHOLDER GROUPS AND CONCERNS Recall that the workgroups wanted to investigate the stakeholder groups and their concerns more closely. This investigation was conducted by the core group in parallel to the workgroups and was finished in November 2020. In this investigation, the following approach was taken:

(1) Gather all information on stakeholders already documented.
(2) Gather organisational charts and actor descriptions.
(3) Categorise stakeholders into primary functions.
(4) Group categories into formal stakeholder groups.
(5) Prioritise stakeholder groups through ranking.
(6) Investigate the concerns of the top five stakeholder groups through questionnaires and interviews.
(7) Present the results in the workgroups.

This approach started in May, during June, the ranking of stakeholder groups was finished, appendix 14. Below, in table 2, the top five stakeholder ranking is presented. Each project team member was asked to provide the contact details of at least one person who wished to represent a stakeholder group in their institutions for each stakeholder group. This pool of contact details was sent a questionnaire to elicit their concerns and numbered fifty–six people. The primary purpose was to produce goals about possible usage, areas of interest and preferred presentation. There was, however, a problem. While the sector already had a reference architecture description for several years, many individual MBO institutions only began working under architecture a short while back. Consequently, it is challenging to ask people with no architectural affinity what they desire, since they have no frame of reference nor any grounded expectations.

This confusion was seen as an opportunity to promote architecture as a discipline while also gathering concerns. Hence, the questionnaire had the following structure:

- Introduce the project Route21 and architecture.
- Ask respondents to introduce themselves by stating their profession and their most important responsibilities and in which business domain they are most active.

Rank	Stakeholder group	Stakeholder examples
1	Information management	Think of the architectural Board, CIO, management team, saMBO–ICT, data officer, security officer, project managers, etc.
2	Business management	Think of the governance, strategy policy officers, project managers, innovation change management, executive board, MBO–raad, CEO ...
3	Education management	Think of the education (operation) manager, Educative teams, Education coordinator, principals, etc.
4	IT–infrastructure management	Think of the CTO, Management team, project manager, security officer, administrators, etc.
5	Quality Assurance	Think of the accreditation Organisation, Dutch Data Protection Authority, Ministry of Education (OCW), Government Education Inspectorate, Quality Assurance (QA) Compliance Department, Accountancy Bureau, etc.

Table 2. Top five stakeholder groups

- Show several architectural descriptions and how they help people.
- Ask the respondents to give a star rating for the shown architectural descriptions on the descriptions' content and presentation based on its purpose.
  - One of the shown descriptions was the intermediary architectural description of the main–process model developed by workgroup A.
- Ask what kind of the main viewpoint sounds most interesting to aid them in their work (processes, applications, information, standards, actors & responsibilities, interfaces, etc.)
- Ask the intended purpose of the resulting view.
- Ask to react to a set of statements on the use of architectural descriptions, which can be used in their work.
- Ask where they expect to find architectural descriptions.
- Show several different presentations of a process and chain and state their preferred presentation.
- Ask if they use model languages in their work if any.
- Ask if they are willing to be interviewed in a later stadium.

The questionnaire was received rather well by the fifty invited stakeholder group representatives. Furthermore, the response rate is shown in table 3. The lower response rate of the IT–infrastructure management can be explained by the fact that most MBO institutions have outsourced their IT–infrastructure, thus request to respond to the questionnaire was most likely met with indifference, by the representatives of the external companies. However, the low response rate of education management is more interesting. Although the number of finished responses is low, the number of education managers who started the questionnaire was much higher. They broke off during the questionnaire when the first architectural descriptions were shown. Although they were not reachable for a response, the assumption is that architecture was too alien. Although education management does use process descriptions, it is in a much more confined scope. To help with a reference architecture might have been too broad a scope, with a too high an abstraction level. The questionnaire was followed up with interviews, with at least one interview per stakeholder group.

Stakeholder group	Number of people with finished responses	Useable responses
Information management	12	12
Business management	9	8
Education management	5	2
IT–infrastructure management	4	2
Quality Assurance	5	5
Tot.	**35/56**	**29/56**

Table 3. Response rate questionnaire stakeholder groups' concerns

The results of the questionnaire and interviews can be summarised as follows. The stakeholder groups are interested in the main viewpoints:

- Processes (main–processes and sub–processes)
- Application landscape and their interdependencies
- Information structures
- Information flows
- Interfaces
- Delegation of responsibilities and ownership

The stakeholders wish to use the views derived from these viewpoints to gain insight, in both structure and interdependencies. As well, to communicate responsibilities and standards. Moreover, the reference architecture description is a blueprint that suppliers can use to deliver custom–made solutions, act as an authority on MBO operations, and as a completeness check for internal architecture descriptions.

On the other hand, the stakeholders prefer presentations which honour the local reading direction, which means arrows should go from left to right or from up to down. This concern also entails that time sequence should be honoured; activities left from another activity happens earlier in the process chain's execution. Another request was that the result should be 'clickable', which means that when someone clicks on an element in the publication environment, one zooms into a more detailed view of that element. Finally, the stakeholder requested that the published views provide source code, rationales, design decisions, and text–based explanations since they wished to adopt the architectural descriptions in their institutions and expand them.

THE DEVELOPMENT OF THE MAIN–PROCESS ARCHITECTURE DESCRIPTION & PRINCIPLES

In this section, the process of workgroup A is discussed. Workgroup A was the group with the least number of professional architects, mainly consisting of information managers or people from the business with process modelling affinity. Furthermore, a workgroup member noted how there were many different people from different backgrounds in workgroup A, but could cooperate & communicate since they all shared the same affinity for process modelling.

Workgroup A started with defining the short– and long–term goals of the workgroup in May. One such goal was the management game in September. In September, a conference of different managers was to take place to discuss the topic of centralisation. One of the agenda items was the discussion in which regard individual MBO institutions were the same, and could therefore collaborate, and in what regard they were unique. To aid this discussion, the project manager had arranged the management game. In this game, each manager would receive a copy of the developed main–process model description and four differently coloured pencils, red, blue, green and black. The managers would then be asked to colour elements according to their uniqueness. The colour coding was:

- Red: data are the same.
- Blue: processes or systems are the same.
- Green: policies and executions are the same.
- Black: unique process, no shared properties.

This management game would not only aid the discussion but would also give the model a trial by fire, as well as showing management the benefit of architectural descriptions.

During the goal–setting phase, a new project and workgroup member joined the group. During one of the many presentations the core group gave about Route21, he became inspired and wished to join. He noted how open the group was and how quickly he could enter during an ongoing process. His addition to the team was received well, as he was knowledgeable on the education operations topic, such as the actual teaching. This knowledge component was underrepresented in the group at the time.

After the goals were set, workgroup A started the development of a model describing the main processes. The group started individually sketching a main–process model based on the earlier developed main–process framework. Note that no viewpoints were developed, which expressed model kind, addressed concerns or stakeholders. The resulting sketches were combined & consolidated into one model through discussions and workshops amongst workgroup A, resulting in the first version of the main–process description. This description was enriched with earlier developed building blocks and documentation. From that point, the descriptions were finely tuned for two iterations, based on feedback from other workgroups.

The description was presented to several stakeholders within the institutions from which the group members stemmed. The resulting feedback and newly gained insights meant that group A came at an impasse in July. They could not reach an agreement on the next version of the main–process description. To break the deadlock, the group proposed that a select few would go to a physical location with members of the core group to hold a pressure cooker session. Note that the group was small not only for the sake of the process but also due to covid19 procedures. The pressure cooker session meant that this smaller group would go to a physical location for one day, lock the door, grab a whiteboard, and come out with a final solution, which they did. Thereafter, in the iterations, more detailed descriptions of the processes were added, the documentation was further enriched, and the presentation was fine–tuned. During this time, some group members used the intermediate description of the main processes in some internal projects, to test the description's practicality as a communication tool. The results of this test were incorporated into the main–process model.

In August, the description was finished and sent to a professional designer. Since the result had to appeal to the stakeholder groups, particularly those from the business. During the iterations, the main–process model description was compared to the process models from the higher–education and secondary education, to keep correspondence with the other education sectors.

In September, the management game was held, and the main–process description was received with praise from management. After some adjustments from the received feedback, workgroup A disbanded for the time being and joined up with workgroup B. Later in December, workgroup A would come back to document the project's principles. The principles had remained implicit or unaltered from the concept phase until this point. This activity is ongoing at the time of writing.

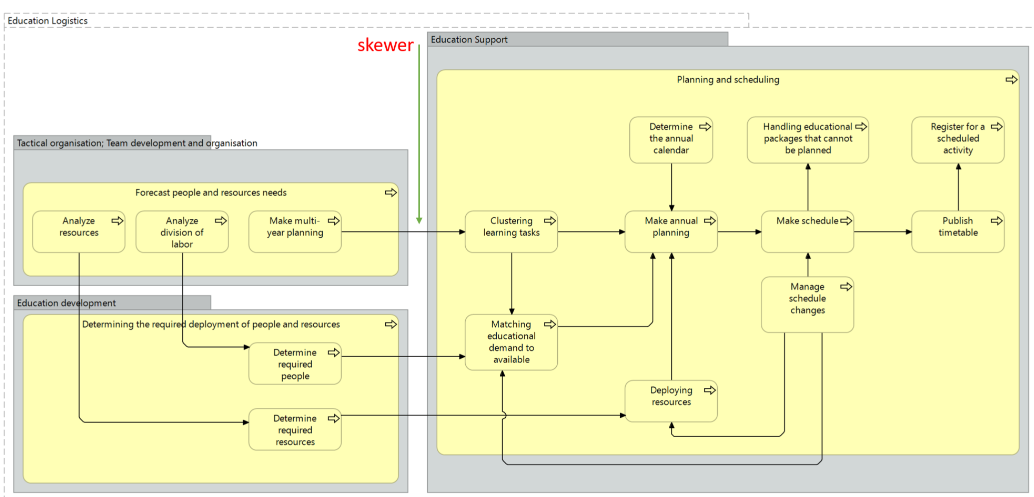

Fig. 26. The figure shows (sub)processes of workgroup B inside A's main–processes, which relates several main–processes. These inter–relations are referred to as skewers in the project; skewers can be relations, as depicted, or other elements that relate main processes.

THE DEVELOPMENT OF THE (SUB)PROCESS ARCHITECTURE DESCRIPTIONS Workgroup B consisted out of the most professional architects when compared to other groups, who were focused on enterprise and business architecture. This workgroup had a troubled start, as the initial assumption was that workgroup B was to specify main processes into subprocesses further. However, the main processes were still to be developed by workgroup A since the workgroups developed parallel. Consequently, workgroup B started by making assumptions based on experience and the old architectural description triple–A as to which main–processes would likely appear.

The next step in the process was to divide the assumed main processes amongst the workgroup members. The division was based on background and experience. For example, someone who had more experience with examination would gain the main–process examination to sketch sub–processes. Each member would then create rudimentary sketches of the underlying processes. Each sketch was to be accompanied by an analysis made by the sketch's author, which describes his process and reasoning on how he came up with the created sketch. All illustrations and accompanying reports were handed in with the core group member active in workgroup B. He would then improve and transform the result into an Archimate model description.

A pair of workgroup members would then review each Archimate based architectural description. The original author could not check or defend his work, as the analysis and description should suffice. If this was not the case, then it was assumed then the model or the analysis was flawed and required improvement. After the peer review, the models were sent to several departments, inside the institutions the workgroup members belonged to, to provide feedback. This feedback would then be incorporated in the next iteration.

During June, both the main–process description and the (sub)process description took a more permanent shape. While there are no significant discrepancies between the main–process model and the assumptions made by workgroup B, which was a validation of completed work for workgroup A. The initial idea that the (sub)processes were the less abstract specification of one and only one main process proved untrue. In other words, when one zooms into a main–process one would assume to see a chain of (sub)processes honouring the borders of the main–process. This assumption proved to be false. What had happened was the (sub)processes moved through several main processes at the same time. For example, workgroup A had defined the main–process 'planning and scheduling' in the domain of 'education support', and the main–process 'forecasting people and resource need' in the 'tactical organisation' domain.

In contrast, workgroup B had modelled a process 'analysing required materials' to trigger the 'matching of educative need to available employees' in their assumed main process 'education logistics'. The analysis process belonged to the main process forecasting, whereas the latter belonged to 'planning and scheduling', note figure 26 for visualisation. Both workgroups felt that this was not a fault in the resulting product, but rather the initial believe that main–processes and (sub)processes were 1–1 mapping. In hindsight, the workgroups stated that (sub)processes should weave different main–processes together as one (sub)process in one main–process, might trigger another (sub)process in another main–process. This weaving of main processes using (sub)processes got the nickname 'wooden skewer', since a skewer strings pieces of meat together for a BBQ.

It was decided that the assumed main processes of workgroup B would be deleted. Subsequently, the subprocesses should be related to workgroup A's intermediate product. Whilst honouring the initial design of workgroup B. As a result, the subprocesses would connect different main processes with 'skewers.' Rather than correcting the descriptions with the entirety of workgroup B and A, it was decided that a subset of workgroup B would execute this plan. This division was because one believed that a smaller group would work much faster and reach consensus quicker.

Thereafter, the iterations were straightforward. The workgroup would take their models to stakeholders and resident experts in the MBO institutions they were affiliated with and incorporate the feedback into the process descriptions. During these iterations' definitions and documentation of the processes was created as well. These iterations continued until October. In October, the results from the stakeholder group concerns investigation were presented. These results were incorporated into the descriptions, which resulted in the first published version of the architectural descriptions of processes. Workgroup B's final activities consisted of peer–reviewing workgroup C's work and fine–tuning the process descriptions. As well, as aiding workgroup A with the development of the principles. At the time of writing, these activities continue.

THE DEVELOPMENT OF THE APPLICATION AND INFORMATION ARCHITECTURE DESCRIPTIONS
Workgroup C consisted out of members from a more core IT background. This workgroup was responsible for developing architectural descriptions on business support, in terms of applications and information. While it was agreed that the main–process description was to be the coat hanger model, the model to which all other descriptions would relate, workgroup C did not have the same troubled start as workgroup B. The reason for this was that information and processes are two sides of the same coin.

Processes follow information, and information follows a process. Therefore, it was decided to begin with the development of the information descriptions, as this would not require the main–process description as the correspondence reference point. This decision would also result in some other benefits. First, this would create the opportunity to validate the work of workgroup A and B. Secondly, information streams can be measured with clear boundaries, where processes cannot.

Sadly, workgroup C founded the created building blocks from the earlier phase lacking information and application elements. Consequently, the first iterations of workgroup C were to gather all available information sources and create from them building blocks. These building blocks would form the initial architectural descriptions. To make the building blocks, one must first gather all relevant information from the various sources, the same as previously stated external sources, combined with the architecture descriptions from some MBO institutions. Each found element was given a definition relevant for the MBO; for example, details found in the HORA were tweaked to apply to the MBO. In the next iterations, the two most senior members of the group decided which information elements would be relevant for an MBO reference architecture. The other group members started the process over again, this time for application service elements. In June, workgroup C had their generic building blocks ready. However, the building blocks' creation took longer than expected, and workgroups A and B had already functional intermediate architecture descriptions of the primary and (sub) processes. Hence, it was decided not to continue using the information description model as the base correspondence, but to use the intermediate process description.

Consequently, application and information architecture descriptions could be developed in parallel. Subsequently, the group divided further into pairs, and each couple would either work on application or information descriptions. This approach was taken as one feared; one of the main viewpoints would not be finished in time if the group continued to work as a whole. Moreover, working in pairs was chosen over an individual approach. One of the project principles was that of 'four eyes', which states no work may enter the central repository without having been evaluated by at least two people.

There was always a pair either reviewing and fine–tuning conducted work, developing application descriptions, or developing the information descriptions. In June, it was decided to expand the meta–model with application components. This expansion was done, since one believed that the stakeholder group would find it challenging to discuss application services, without the context of an application component. However, workgroup C was short on time. Therefore, the data objects in the information viewpoint would be dropped and continued only with business objects. The consequence is visualised in figure 27.

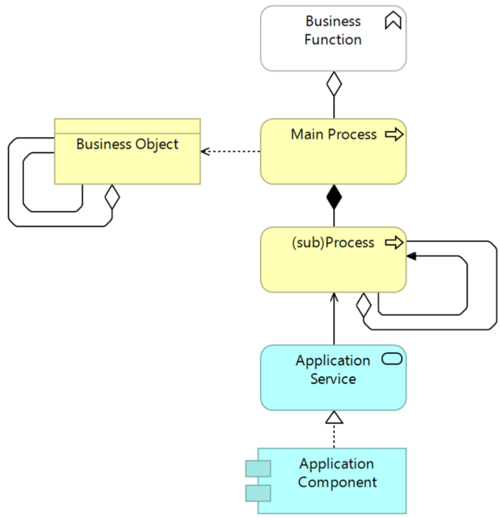

Fig. 27. Adjusted meta–model.

The adjustments to the meta–model started a fierce discussion in workgroup C on formal correctness and informal pragmatism. The other educational domains, the higher–education and secondary schools, did not use application services but instead used application functions. However, relating application function directly to processes, as the higher education did, goes against the Archimate3.0 modelling language, in which only application services expose behaviour to the processes. The reasoning went that the idea of a 'service' was too abstract for stakeholders to understand, and one should align oneself with the other domains within the educational sector. Whereas the other side argued that one should follow the standard and that services were a distinction stakeholder were aware of, but lacked a proper term to distinguish the two. The argument was settled in favour of using application services. This decision was reached after prototypes were shown to stakeholders, who admitted that at first, they did not see any difference between functions or services, but agreed, that there should be a difference between function and service after the explanation provided by an architect.

The iterations of working in alternating pairs on either reviewing or description development continued until both the main–viewpoint and view descriptions were finished. Furthermore, workgroup members occasionally sent out the descriptions to employees within their institutions to gather feedback and support for the created work.

6.4 FUTURE ACTIVITIES IN THE ONGOING DEVELOPMENT PROCESS

In December not all main views were finished. The information model and the project principles were still being developed or documented. This development would continue after the holidays in January. After that is completed, the plan is to publish the created views and start constructing custom viewpoints, that is stakeholder specific. Recall that the project learned from previous projects that stakeholders do not know what they want, and that architecture is a concept far away from most people's frame of reference. This alienation was underlined during the investigation into stakeholders' concerns when the education management group broke off during the questionnaire. Therefore, the plan was to create a baseline architecture description based on experience and main viewpoints. Thereafter, these main views would be shown as prototypes to the stakeholders.

Moreover, stakeholders would be asked to use the descriptions in their work. Consequently, stakeholders can react and state what they like, do not like or miss in the main views. As a result, the architects can create views related to their interest and document them in actual viewpoints since they are directly related to stakeholders and their concerns.

The by example development of viewpoints, in which one start with main views derived from the main viewpoints, resulted in publishing the work in three different forms. First, the project files and source code will be made open–source and available to everyone, such that architects and others affiliated with Archimate3.0 models can use them and provide feedback by raising issues on the git platform. Secondly, the views are published on a semantic wiki using the professionally visualised models. This format is meant for those not comfortable with Archimate3.0, such as management. That is not to say that the Archimate3.0 imagined views are not published on this wiki. Still, it does require specific procedures to switch the visualisation from the informally professionally visualised views to the formal Archimate3.0 views. Additionally, a FAQ and a feedback form are provided on the site as well, such that one can give feedback or raise questions concerning the reference architecture descriptions.

Furthermore, from the investigation into the stakeholder groups' concerns, it was elicited that stakeholders were also interested in information flows, standardised interfaces and actor responsibilities. These main viewpoints were not developed and are registered for the future to enrich the now developed reference architecture description for the MBO.

PROCESS RETROSPECTIVE In December, a retrospective of the development process of 2020 was held. During this retrospective, several interesting remarks were made. The first notable remark was that of the group diversity on both background and knowledge. Moreover, the interviewees unanimously agreed and communicated that the collaboration between architect and non–architects, ICT related functions and business–related functions was positive. This diversity was a team structure which they wished would happen more often. Non–architects felt that they learned a lot from the architects, such as how the sector operated, was interdependent and how one could logically organise the industry. Simultaneously, many architects felt that the non–architects supplied vital information on the processes and other architecture descriptions elements and ensured the architectural descriptions' recognisability and practicality. The project leader noted that the non–architects admonished architects when they dived into rabbit holes about various aspects of architecture fundamentals or one of the description elements.

In contrast, architects ensured that the descriptions were feasible and aligned. One architect noted how the project should not try to be formaly correct but aim to affect the sector desirably. Another way the group was diverse, was in terms of knowledge domains. The core group consisted of specialising generalists, whereas the workgroup consisted of specialists in various education operation domains. One interviewee noted how essential this group formation was. Since the project heavily relied on expert opinion and experience.

Additionally, one interviewee felt that maybe there should have been more project members specialising in the underrepresented domains, such as the research domain. Given the group diversity, one project member noted that the group was diverse. Still, they did share a common ground, namely the affinity of process thinking, the project vision and experience with process modelling in the education sector. He reasoned that if it were not for this common ground, the project would most likely have failed due to language barriers and misunderstandings.

Many group members agreed that the two weekly sprints with the corresponding sprint retrospective, stand–up, closing and joined development start of the workgroups worked well. One member noted that it felt like doing a PDCA cycle every iteration, as conducted in LEAN. Another member pointed out that the shared vision and strong group mentality made that the freedom given was not abused and led to high–quality results in the creative process of architecture description development. Despite this, he did wish there were a bit more time planning, such that his workgroup was better able to measure progress.

Another topic was that of top management and their commitment to the project. This commitment was a requirement, especially for the project's freedom, for example, to allow people to join the project midway. This commitment was enhanced by the management game, which acted as a catalyst. Due to the management game, many managers found out first–hand how architecture is beneficial, especially in fundamental discussions, such as centralisation. Many managers either reconnected with their architects or reinforced in their belief in architecture.

Moreover, the project's interactivity with stakeholder groups, in general, was met with enthusiasm and a newfound understanding of architecture as a discipline. One stakeholder representative from the investigation into stakeholder concerns noted how nice it was that architects asked and sought employees' participation down the ladder. Since they, too, could help improve the sector.

Further, the abundance of information was noted several times as well. Not only had the project several members from the previous reference development project, but there was also well–maintained documentation from the last reference architecture. Additionally, the maturity of architecture in the Dutch public sector meant that several related reference architecture descriptions existed, from which content could be derived. Moreover, one domain within the education sector, namely the primary and secondary schools, was undergoing a project to enrich their reference architecture descriptions. Naturally, collaborative efforts were undertaken, resulting in more feedback and information. One project member noted how this multitude of information sped the initial phase of the project development considerably. As a result, one could start sketching nearly complete pictures early on due to the provided information and building blocks beforehand. A core group member noted how this abundance of information, combined with the project group's broad expertise, mitigated the project's most considerable risk, namely relying on expert opinion to develop the first main–viewpoints and views, with limited stakeholder involvement. Despite this risk, he admitted that an alternative course was unlikely, due to the fact many stakeholders do not have an exact frame of reference, what architecture does or what their concerns should be. It was still one of the most considerable risks of the project.

When asked whether starting with the main–processes instead of the information model was a good thing, since data and processes are two sides of the same coin, many noted that it was a correct decision to start with the main–processes. First and foremost, processes were something that was understood by most identified stakeholders and something that was common ground in terms of knowledge in the project team, whereas data was not. Secondly, the main processes were described in the previous reference architecture, where data and information were not. One project member noted that he would probably not have joined the project, if they began with data, since he became filled with enthusiasm when he saw the framework for the main processes.

Additionally, some group members were asked about the top–down approach of the architecture process in the MBO sector. One begins with a central reference architecture in a top–down approach, which trickles down into architecture descriptions in the institutions. In contrast, in the bottom–up approach, one begins with institutional architecture descriptions, which are abstracted and consolidated into a reference architecture. The group members agreed that the top–down approach was better than the bottom–up approach. One interviewee noted how well public sector institutions, including education institutions, excel at 'nit–picking and discussing topics without agreeing'. Stating a bottom–up approach would have led to a discussion in which once's description is defended as the only truth and other descriptions would be nitpicked upon. Whereas when developed centrally, this would be much less of an issue since its beginning was more abstract and developed collaboratively.

Additionally, smaller institutions have no architects and would, therefore have no input. In contrast, by developing top–down, they have a voice, due to the interactivity moments in route21 and the ability to join the project team as non–architects. Consequently, due to the central reference architecture, the smaller institutions start at a higher level of architecture maturity, when they deploy architecting in the institution.

Lastly, the topics of tooling and covid–19 are discussed. First, the project lead was somewhat relieved that the negations with suppliers for a professional architect modelling tool seized. Stating that he feared that if a commercial tool were procured, that the goal of creating a reference architecture description for the MBO sector would have become secondary to the pursuit of filling the new procured tool with architecture descriptions. Consequently, the project stayed using open–source tools, supporting standardised exports of the source code, and git integration.

On covid–19, there were negative remarks, such a decrease in efficiency compared to the achieved efficiency of the previous development project, with physical attendance. Additionally, the increased effort necessary to keep stakeholders and project members enthusiastic and committed to the project. Surprisingly, there were positive remarks, such as increased attendance due to working from home, and the increased digitalisation and documentation of the development process and its artefacts.

7 DISCUSSION

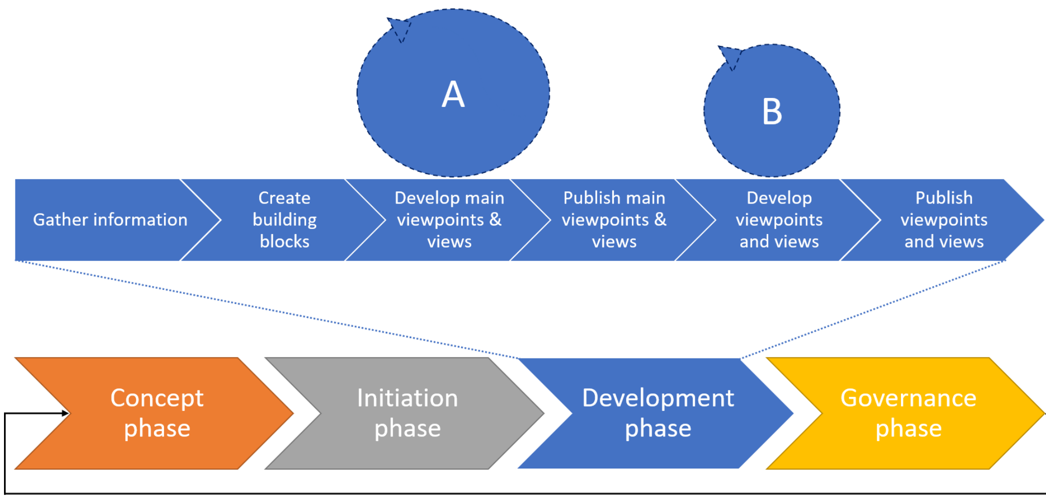

Fig. 28. The observed process.

In this section, the perceived process is generalised to a practical process. Subsequently, this process is compared to the earlier introduced theory. Additionally, best practices from the theory are compared to the case, and any additional lessons learned from the case are stated.

The empirical process can be divided into four phases, the conception phase, the initiation phase, the development phase and the governance phase.

In the concept phase, the project budget, planning, method and scope are formalised. Furthermore, the project's position within the Organisation and a project environment analysis would be conducted as well. More importantly, the main principles which guide the project are defined as well, which can be derived from the enterprise–wide principles. These principles describe the context, scope and rationale of the project—for example, the modularisation principle of the education sector in the case. The resulting artefacts from this phase are the initiation document and set of governing project principles.

In the phase after that, the initiation phase, the project vision, stakeholders, concerns, architecture development language, frameworks, architectural methods and resources are established. In the case, this phase was completed through several conferences and meetings, which first found identified stakeholders based on the project concept and environment analysis. From experience and expert opinion, likely concerns of the stakeholders were identified. After this, the main viewpoints were selected, which was believed to tackle assumed stakeholder concerns. Later the ADL and frameworks, Archimate3.0 & TOGAF 'light', were chosen for the development process of the selected views derived from the main viewpoints. Note that every project has a framework. This framework can be derived from a standard, such as TOGAF, or an in–house solution. The chosen framework is then enriched with identified stakeholders, likely concerns, main–viewpoints and an ADL. Consequently, every project has its framework, which is the resulting artefact of the initiation phase.

The development phase, the phase thereafter, will be discussed later in more detail. The resulting artefact is the architecture description, which consists of rationales, decisions, viewpoints, views and AD–elements.

The last phase, the governance phase, has not been reached at the time of writing. In this phase, the architectural descriptions and artefacts are maintained and receive further iterations. Possible artefacts can include a maintenance plan & schedule, as well as evaluation planning. The evaluation meetings should decide whether the AD should receive more development iterations. For example, when IT centralisation has occurred or when stakeholder groups require new viewpoints & views.

To return to the subject of the development phase. This phase consists of six steps, one perceived iteration set, and one intended iteration set. The latter is intended since the research stopped after developing the main–viewpoints and views in December of 2020. More on the six process steps, these include gathering information, creating building blocks, developing main–viewpoints and views, the publication of main–viewpoints and views, the development of viewpoints and views, and finally the publication of said viewpoints and views. The abstracted process of the project is visualised in figure 28.

When gathering information, one indexes available information and its sources. This information can be internal documents, memos, previous projects, data from competitors, records from the sector or available research papers. In the case, the information was internal, such as the triple–A and TPO, or from the industry, such as the HORA, FORA and NORA, and research material. The resulting artefact is either a document or other information container, which identifies, relates, and describes information and sources. We refer to this as the copy & paste source.

The copy & paste source is then analysed and transformed into building blocks during the next step. This creation of building blocks can be achieved by several methods, such as the earlier discussed traffic light method, which was applied during the copy & paste session. Note that the building blocks' granularity can differ greatly, based on the project's overall vision and purpose. The main idea of these building blocks is that one does not start from scratch or reinvent the wheel, but instead use what is available and copy & paste what is applicable in the situation. Depending on the project and architecture maturity, the resulting artefact is either an information container describing building blocks, such as a spreadsheet, or a filled central repository with developed building blocks.

The next step is the development of main–viewpoints and views. This step is incremental and iterative. In the case, this step was conducted by three different workgroups A through C, and each had their architectural layer. Moreover, the workgroups used different approaches to the development of their architectural descriptions. However, the overall process shared many similarities that can be abstracted to a more general process description, as shown in figure 29.

In the first step of iteration A, analyse & sketch, the group sets their goals and planning of the iteration. Each group member or pair is assigned a particular area of the main viewpoint to develop. In larger groups, multiple couples or individuals may get assigned the same area or have overlapping regions. Each group member or pair analyses the available material and information, resulting in a short analysis report on the designated main–viewpoint area. Using this report, the individual or pair sketches an architectural description. In the first few iterations, the sketches and analysis will be built mostly based on the building blocks and experience, where later iterations will rely more on gathered feedback. Artefacts of this iteration step are several sketches coinciding with analysis reports about a specified area of the main viewpoint. Furthermore, the analysis report's primary purpose is to document used information sources, explain rationales, and explain decisions made in the sketch.

In the next step, the set of sketches and analyses are reviewed by other individuals or pairs from the same group while noting their findings. According to the comments made, the sketches' original authors then meet with the reviewers and refine them. The refined sketches are then consolidated into one or more views, depending on the chosen main viewpoints' areas' granularity. The most important aspect of this step is the correspondence when consolidating. The consolidation can be done by an individual, a pair, a subset of the group, or the entire group. Note that the rule of thumb, in this case, was that the larger the number of consolidators, the longer the discussions, resulting in a longer consolidation process. The resulting artefacts of the review & consolidate step are one or more intermediate views.

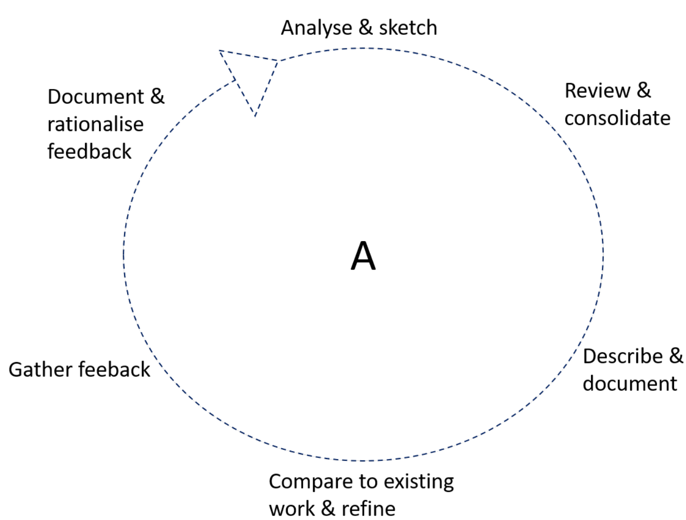

Fig. 29. Observed development iteration.

During the describe & document, the description is added to AD–elements, for example, a process description. As well as any modification to the viewpoint or the view documentation based on new insights. From the analysis reports of step one coinciding with the resulting views, one can formalise any architecture rationales and decisions as well. In the case, this step was skipped in the first few iterations. Furthermore, this step can be conducted with a subset of the group, while the rest continues with the next step. Resulting artefacts include modified views, viewpoints and AD–elements, and documents formalising decisions and corresponding rationales.

In the 'Compare the existing work & refine' step, the correspondence with other workgroups' work and existing architectural description, internal and external, is ensured. When discrepancies are found with other workgroups' work, a meeting should be held, or an issue created to align said discrepancy. When differences are found with descriptions outside of the project, one faces the choice, to either adopt & align to the outside source's solution or keep the discrepancy. When choosing the latter, the decision & rationale should be documented. As a result, one explicitly answers why the project deviates from the outside source. For example, in the case, application services instead of functions were chosen, since that was the adopted standard, despite other sources using application functions. Although no correspondence or correspondence rules were documented in the perceived process, keeping it implicit. Artefacts in this step can include written correspondence and correspondence rules, either separately or in the corresponding viewpoints.

In the step after that, one gathers feedback from one or more sources. This feedback can come from other workgroups, colleagues who are not on the project, stakeholder representatives, focus groups, advocacy groups, researchers, product owners or management. In the earlier iterations, the feedback source will be most likely other workgroups. As the iterations progress and repeat, the sources become more and more the identified stakeholders themselves.

The gathered feedback is then formalised and documented. This formalisation could take forms of user stories, adjustments to the viewpoints, memos or reports. The information from this step as input for the first step of the iteration, analyse & sketch. The iteration begins anew until a satisfactory and accepted result is reached.

Furthermore, the iteration can last any number of weeks. In the case, a sprint lasted two weeks. Although it did occur that workgroups did not finish an entire iteration as described within those two weeks. This iteration debt was either done in free time or accepted and moved back to the next sprint backlog. Furthermore, every two weeks there will be a meeting lasting a morning. The first half will be with the entire project team. In this half, the core group reports progress, pointers and what is new. Subsequently, someone from the workgroup, most often a member of the core group participating in the workgroup, reports on the workgroup's progress, plans, and setbacks. Subsequently, a small show & tell of the resulting work of the previous iteration is held. In the other half of the morning, the project team divides themselves into the workgroups again, to make their plans more concrete, separate & assign the work and if time permits start the work as a group. After this session, the core group meets again and discusses what happened in the groups. In this meeting, decisions were made whether management interference is necessary and if action is needed for the sake of the correspondence. The step after this iteration set A, is the publication of the main–viewpoints and views.

Admittedly, drawbacks of iteration set A include the reliance on experience in the field and architecture as a discipline. The project member must rely on available information and expertise to determine what information is relevant, what building blocks can be created, and create main–viewpoints and views in parallel with little to nonmajor discrepancies between workgroups.

During the iteration set A, feedback is gathered, but the overall involvement of stakeholders is limited. While requesting feedback from stakeholders' representatives inside the project members' institutions frequently happened, gathering feedback from stakeholder group representative outside said institutions was rarer. This more uniformly stakeholder involvement happens in intended iteration set B. In iterations set A, the main–viewpoints and views are mainly developed based on experience and available information. The result from numerous repeats of iteration set A has adjusted main–viewpoints and the views themselves. The involvement of stakeholders, in which they dictate the viewpoints, and the project group creates the corresponding views, is conducted after the main–viewpoints and views are published. That is not to say that there is no involvement, as stated stakeholders can still be asked for feedback during iteration set A.

Moreover, one can investigate stakeholders and their concerns in parallel with iteration set A, as done in the case. The results of this investigation can be used for some quick wins. For example, the research in the case suggested that stakeholders have trouble understanding if the relationship did not honour the reading direction, incorporating this resulted in a quick win. Additionally, the project group became aware that stakeholders' interest lied in more main viewpoints than just the ones they were developing. However, viewpoints about a set of stakeholders and concerns, where said stakeholders are product owners of the to be created views do not occur during the development of main–viewpoints and views in iteration set A, but in the development of viewpoints and views is intended iteration set B.

Recall that one first develops a baseline architecture description consisting of main viewpoints and views developed on experience and available information. This limited initial stakeholder involvement is because stakeholders or rather those outside of the architecture department, arguable inside the architecture department, have no idea what architecture is and what they can expect from it, let alone define requirements and interest to create viewpoints. Recall that this was also evident during the investigation into stakeholder concerns in the case, where the education management broke off the questionnaire because it was too much. It is easier to show stakeholders baseline architecture descriptions, take the time to explain and discuss said description and then ask what 'do you like, hate, what should be different' and so on. Rather than asking 'what do you want from the architecture discipline', through business analysis with no examples. In other words, one first develops the baseline, such that it is possible to create the viewpoints and views later by example or prototyping. In a way, the step which produces the main–viewpoints and views can be seen as a long sprint zero. In sprint zero, a prototype which can be used in practice is developed and published, from which custom viewpoint and views for the stakeholders can be created, through interaction. Although we have no evidence for this, we assume that iteration set B is similar to that of A. The most considerable difference would be the more direct stakeholder involvement as product owners of the resulting architectural descriptions.

Phase	Activity	Step	Artefacts
Concept			Initiation document and project principles
Initiation			Project framework, initial repository
Development			Architecture Description
	Gather Information		Copy paste source
	Create building blocks		Building blocks (can be inside repository)
	Develop main–viewpoints and views		Main viewpoints views
		Analyse sketch	Sprint plan goals, sketches analysis reports
		Review consolidate	Results are one or more (intermediate) main views. Main viewpoints are updated where necessary.
		Describe document	Adjusted main–viewpoints and views, as well as documented decisions rationales
		Compare to existing work	Results are refined main–viewpoints and views as well as documented correspondence and correspondence rules.
		Gather feedback	
		Document rationalise feedback	Documented evaluation report
	Publish the main–viewpoints and views		Published main–viewpoints and views
	Develop viewpoints and views		(Custom) viewpoints and views
		B1	
		...	
		Bx	
	Publish viewpoints and views		Published viewpoints and views
Governance			Maintenance plan evaluation planning

Table 4. Perceived process with artefacts

When comparing the perceived process, see table 4, to the theory, one can note similarities and differences. First, we compare the process with doDAF, then TOGAF and finally NATO–AFv4. When comparing the process to the doDAF method for architectural development one can note that the first two steps in doDAF's development method, namely 'determine the intended use of architecture' and 'determine the scope of architecture', is not a part of the development phase in the process. These two steps in doDAF methods are similar to the concept and initiation phase, rather than being part of the development phase. Although doDAF does not name an artefact during the 'determine intended use of architecture' step. The named topics, such as project methods and impact, are all part of the initiation document from the concept phase. Whereas the 'determine the scope of architecture' has its place in the initiation phase and the 'gathering of information' step in the development phase.

Steps three through five of doDAF's method are similar to the development phase, whereas stage 6 can be compared with the two publication steps. Furthermore, doDAF does state more artefacts, such as a taxonomy for the integration of views and metadata registries. However, if necessary one can enrich this perceived process with these artefacts, recall that it is often good practice to mix and weave several methods and frameworks. This process does not state, such a taxonomy or registries, because they were not necessary for the project's scale conducted in the case. Furthermore, the same step that states the taxonomy, also states that a plan on recording data and filling architectural tools should be delivered. Although it is a good thing to have a plan, recall the project lead during the process retrospective, feared that such an approach would take away from the overarching vision and project goals and make filling the tool or repository a goal unto itself, rather than a consequence of the project. Moreover, too much documentation and extra artefacts may dampen the creative process or are challenging to combine with agile working methods.

When comparing the TOGAF's ADM process, one can note the following similarities and differences, as visualised in figure 30. The first thing one notices is that the TOGAF defined 'migration planning' and 'opportunities & solutions' are absent in the empirical process. These missing steps have to do with the architectural product which the project was to deliver, namely a reference architecture. A reference architecture has no migration planning, nor has it a phase where the architectural description is transformed into working products for a roadmap. On the similarity side of things, the concept phase and preliminary step into TOGAF have a near–complete overlap. The initiation plan artefact has a 1–1 relation to the Organisational model for EA into TOGAF. The most considerable difference is the tailored architecture framework and repository artefacts of TOGAF in the preliminary phase, which are artefacts of the initiation in the perceived process. While on the initiation phase, one addition TOGAF notes for the initiation plan is selecting a management framework.

Furthermore, the architecture vision step into TOGAF and initiation phase are similar as well, and the difference lies in one of the artefacts, namely the project framework. TOGAF creates frameworks in the preliminary step and creates drafts in the vision step. Whereas in the empirical process, the framework is made in the initiation phase, consisting of stakeholders, concerns, viewpoints, meta–models (possible drafts) and ADL. Moreover, TOGAF's preliminary phase has a general framework of the enterprise or enterprise projects, not a unique framework for the project, as is the case in the perceived process.

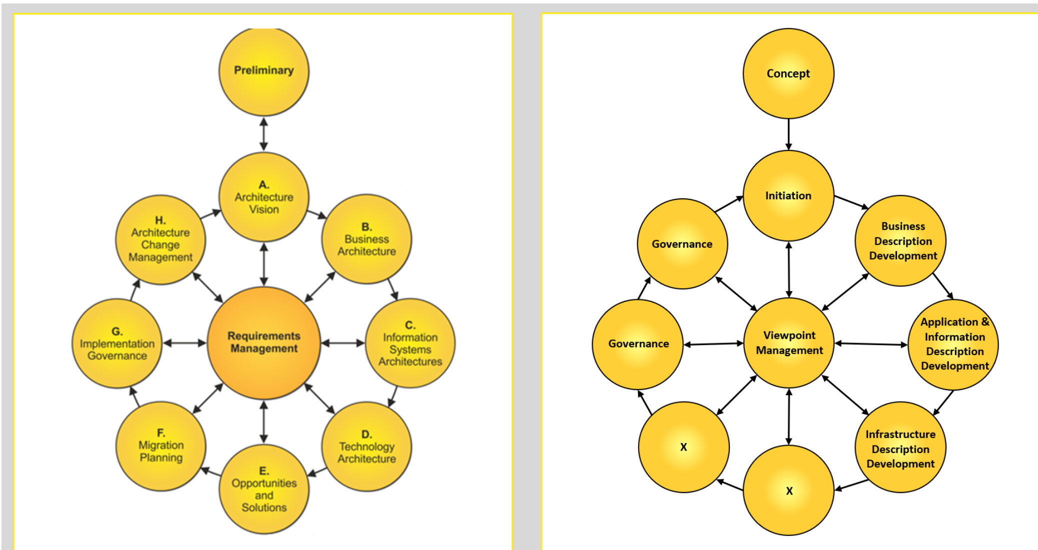

Fig. 30. Observed process compared to TOGAF's ADM. X, a similarity did not occur.

TOGAF's ADM steps B, C & D, corresponds to the development phase of the perceived process. While the perceived process did B, C & D parallel, TOGAF models it serial. The difference in scope can explain this seriality. As TOGAF has abstracted the process and by modelling it serial, the aforementioned top–down approach is modelled. This distinction in the perceived process is not necessary as the top–down approach was a project principle. Consequently, TOGAF starts with the process before data as well.

Another difference includes viewpoint management. Recall that a viewpoint, and to an extent the main viewpoint, frames the stakeholders and their concerns. In other words, the viewpoints govern the views, those combined with AD–elements, rationales and decisions make up for the overall architectural description. Consequently, there is no requirement management, such as into TOGAF, but viewpoint management.

In table 5, the development iteration set A is compared with that of TOGAF. Notice that step 3 and 4, the roadmap components and impact analysis, are absent. This absence has again to do with that the empirical process created a reference architecture. The last step of TOGAF's development process, "create an architecture definition document", compared to the third step of the observed process: "describe & document", may seem like the same activity. However, there are not the same activity; the architecture definition document provides a qualitative view of the created architectural description and its elements. On the other hand, The "describe & document" iteration step in the empirical process illustrates the qualitative view of only the AD–elements, such as element descriptions. In other words, TOGAF's architecture definition document is much broader describing facets, as design decisions, rationales and footprints. Within the empirical process, the respective viewpoints describe these facets.

Moreover, there is a reason why the 'describe & document' step comes before the feedback step since the descriptions are necessary to convey semantics (interpretation) and are a part of the architectural description. For example, the AD process element 'billing' might include creating the bill in one Organisation or just sending a bill in another Organisation. Thus, descriptions are necessary to convey the right interpretation.

Lastly, is the implementation of governance and the architecture change management of TOGAF in relation to the governance phase in the empirical process. Since a reference architecture was developed the practical approach did not need a finer granularity between change management and governance implementation. Both steps into TOGAF were combined into the governance phase.

The comparison with NATO–AFv4 is a short one, in that there are no notable discrepancies. Both the empirical method description and that of NATO–AFv4 can work side by side.

To conclude the empirical method has similarities with all three methods proposed, namely doDAF, TOGAF and NATO–AFv4, but knows fewer artefacts or architecting steps. These missing artefacts can be added to the empirical process if need be, but they were not needed when creating the reference architecture description of the education sector. However, one should be careful with adding more artefacts, as even more documentation may deafen the creative process or may lead to become incompatible with an agile mindset. More on the reference architecture description development, due to the reference type architecture, several critical areas in the architecting process were not perceived, which are described in doDAF, TOGAF and NATO–AFv4, being the migration and transition planning. This research is limited to reference architecture description development and requires further investigation into more common architecting processes.

In the remainder of the discussion section, the perceived process is compared to other case studies and reported best practices.

Step in empirical iteration	Name in empirical iteration	Step into TOGAF's ADM	Name into TOGAF's ADM
1	Analyse sketch	1	Develop architecture description
2	Review consolidate	1	Develop architecture description
3	Describe document	6	Create the architecture definition document
4	Compare to existing work refine	2	Perform gap analysis
5	Gather feedback	5	Conduct a formal stakeholder acceptance review finalise
6	Document rationalise feedback	5	Conduct a formal stakeholder acceptance review finalise

Table 5. Comparing TOGAF's development iteration to the perceived iteration

Recall that similar research in Norway's education sector underlined the importance of committed management to enterprise architecture. One of the reasons why commitment is more problematic than usual is because of the unclear benefit and definition of architecture as a discipline. In the case of this research, management was committed and stayed committed. The commitment started by making the project a part of the strategic agenda and it stayed committed by showing the benefit of architecture, through the management game. Another vital facet was that of a shared vision. This importance is underlined in this case, as a shared vision leads to enthusiasm in the group. Consequently, the core group had only to manage by exception, as the workgroups themselves were able and willing to organise and go the extra mile.

One of the best practices was that of right tooling, while it is essential to have adequate tools, such as editing the file as a group at the same time, it should not become a goal to fill and use the tool with architecture descriptions. The purpose and the visions should remain with the stakeholders and their concerns.

Another best practice that was named was that fancy architectural description does not make good architecture. Despite this, a lack of fancy descriptions might bore or lead to misunderstandings by none IT–stakeholders. In other words, some degree of pretty and informal descriptions is necessary to engage with specific stakeholders.

Lastly, was the rule of thumb that architectural development is a full–time job. The reasoning was that part–timers or those indifferent to the discipline would devalue the resulting products. In the case, the project members participated in the project in parallel with their regular duties and tasks. While individual project team members indeed stated that they sometimes were irritated, when other members did not always have the time. It was not a real issue within the project. There was no perceived devaluation in the product. A possible explanation is that all project members were enthusiastic and shared a vision and, more importantly, believed in that vision. Whereas, some other part timers might be more committed to their every day tasks. However, project member would sometimes continue working on the project in their free time. Although, this shows commitment, it also shows that they required more hours to work on the project than originally scheduled. All with all, this rule of thumb did not apply to the full extent, and perhaps the problem is not part–time versus full–time, but the degree in commitment and enthusiasm.

Additionally, there are some lessons learned from this project, which can be added to the list. Do note that these are lessons learned from the case and are at best, best practices, lacking any serious scientific underpinning.

- Use a diverse project team of specialists in terms of knowledge, sharing a common ground or affinity, such as process modelling.
- The core group or steering group should be specialising generalists, such that they can effectively communicate with project members and partake in the workgroups.
- Let the steering group members participate in the work.
- There should be clear borders to the project, such that the project team is guided. However, there should not be any strict planning, which can kill the creative process.
- Manage by exception.
- Actively engage with stakeholders and through 'show & tell' convince them of architecting and architecture descriptions' benefits.

- Develop by prototyping. Architecture as a discipline is young and ill–defined. Stakeholders do not know what they want and what to expect of architecture. Reacting to something concrete is easier for stakeholders.
- When using workgroups, allow them to diverge into a working method which best suits them.
- Start developing the most common denominating viewpoint of stakeholders, in the case; this was the main processes. This base can function as a reference and fixing point and serve to create recognition between stakeholders and the project.
- Start centralised and top–down. Smaller organisations do not have the resources needed, and a bottom–up approach can lead to undesirable human behaviour.

7.1 LIMITATIONS AND THREATS TO VALIDITY

The research is limited in several facets. First is that the case is about developing a reference architecture description within the Dutch educational sector. As stated earlier, the development process of a reference architecture description misses some common steps or phases often described in an architecting process. Secondly, the Dutch educational sector has a higher level of EA maturity due to government regulations and investments. Consequently, there is a vast pool of EA information available, such as the NORA, RIO & Edustandaard. The perceived development process relied on this information availability and experience. Hence, applicability in other educational or public sectors outside the Netherlands may vary widely. Moreover, some MBO institutions already had some architecture descriptions and established experience with architecture. Another limitation was the original plan, where one first develops the (main) viewpoints and after that the (main) views. This initial plan was not possible in practice since stakeholders did not know what they desired or were interested in. Subsequently, creating the main viewpoint with a view and using that view as a prototype to elicit and develop viewpoints proved to be more practical. This deviation in planning combined with covid–19 outbreak and lockdowns meant that the actual thought–out process would never see the light of day. Consequently, the perceived process was different and amid rare circumstances. As a result, this case has limited repeatability qualities.

Although participating in the architecture development process has delivered a much more in-depth understanding of the said process, active participation has influenced the process from taking its otherwise natural outcome. Additionally, the nature of the single case study makes generalisation and applicability to other situations hard. Moreover, the dataset is limited in terms of conducted interviews.

8 CONCLUSION

Enterprise architecting is the process to establish the enterprise's long–term vision and structure. Additionally, any methods, activities, or motivations revolving around the enterprise architectural descriptions, such as its governance, maintenance and development. This architecting process is undertaken to achieve IT and business alignment as well as other benefits. The enterprise architecture description describes the enterprise through AD–elements, views and viewpoints. In this description, viewpoints frame concerns from stakeholders on systems of interest, whereas the views answer these concerns according to the viewpoint. Enterprise architecting knowledge today mainly exist out of best practices rather than any foundational science. Examples of these best practices include many frameworks. These frameworks exist to help enterprise architecting, such as doDAF, TOGAF and NATO–AFv4. Within the scientific circles, most works are on the topic of frameworks or best practices. Consequently, the architecture description development is a phenomenon that is poorly understood and under–reported.

Whilst studying a real–world case in the Dutch educational sector, in which we describe an observed enterprise architecture description development process, we find a process that differs from theory. Although bearing similarities to TOGAF, the perceived process emphasises pragmatism, freedom and less documentation. The first contribution this study makes is to add a new case to the literature.

The second contribution is identifying a practical EA development process in the public sector. Four phases distinguish this perceived process. The process starts with a concept phase resulting in an initiation plan and the project's governing principles. Within the initiation phase after, likely stakeholders and concerns are identified through experience and available enterprise information. Based on this information, a selection of main–viewpoints, ADL and meta–models are selected or developed. The result of this phase is the project framework. The project framework is the input of the development phase, where available information is transformed into building blocks used to develop main views corresponding to the main viewpoint through an iteration–set. This iteration–set is characterised by the analysis –> sketch –> compare –> consolidate –> document –> correspond –> and gather feedback loop. The resulting main–viewpoints and –views act as prototypes when developing viewpoint and views for the specific stakeholders.

These stakeholders are identified through experience, available information and expert opinion in the concept phase. These assumptions can be confirmed through elicitation techniques. This grouping of stakeholders is not definitive and can expand or shrink during the project's subsequent phases due to more experience or new information. The identified stakeholders are formalised in the project framework.

On the subject of stakeholders, the architecture description must be recognisable and appealing. Recognisability can be achieved by using known formats, for example, a value chain. This recognisability is crucial as it can aid in understanding descriptions. Secondly, appearance is essential as well. This statement might sound unlogical. One could argue that the functionality of the descriptions are more important than appearance. However, if the description is unappealing, it is not used by the stakeholders. The same holds for formality; adopting a standard and following definitions is good, but making sure that the descriptions will be used naturally is better. In other words, it should come secondary to pragmatism. Using definitions or standards to deviate from stakeholders' experience might render the description useless since stakeholders will not use it.

Speaking of stakeholder concerns, the 'by example' development, using main viewpoints and views, is done since stakeholders often have varied and unfounded architecture expectations. When providing stakeholders with a prototype or example, the elicitation of viewpoint and views becomes achievable in practice. This development process is limited in the reliance on project member experience, enthusiasm, and overall information availability concerning enterprise architecture. Additionally, testing the work with the actual users and stakeholders based on both content and design is beneficial. First, it can test the functionality and the degree to which the stakeholders are inclined to use the descriptions. Secondly, it actively markets the project products and architecture as a discipline within the enterprise.

Additional ways to elicit stakeholder concerns include project members experience & expertise, requirement management elicitation techniques, public information, and industry specific information.

The case suggests that the need for an architecture description comes from the stakeholders themselves or market demand, not a pure management or department decision. Additionally, it confirms the need for management commitment and involving management as a stakeholder in the architecture descriptions. Moreover, stakeholders should be directly involved with the development process; this can be achieved in multiple ways, such as testing intermediate descriptions in other projects, acceptance testing, questionnaires, interviews, or a game. The latter is an excellent tool to involve management, as it can show the usefulness of EA, can function as a functionality test and market the project towards management.

When comparing the theoretical process to the perceived process based on architectural artefacts, the project–specific framework springs to mind. In the project framework, viewpoint management is central as opposed to requirement management, as requirements are a concern framed by the viewpoint. This project framework can be derived from the enterprise framework in combination with a standard such as TOGAF. This intermediate framework is then enriched with stakeholders, concerns and viewpoints. Each project should have its framework since each project caters to an arguably unique set of stakeholders and concerns. Additionally, it is essential to remember that all artefacts, including the framework, are dynamic and not set in stone. During the continuation, the framework evolves with the project.

On the process of creating views, there are some things to conclude. One of these things is the use of a diverse project team in terms of knowledge. A project team with different expertise in operational domains can use its combined expertise to create EA descriptions. However, these team members should share a common ground in order to prevent language confusion. This diverse team should divide into smaller, more manageable workgroups than gain a specific subset of the architectural description to develop. As they are responsible, they organise themselves, the steering group manages by exception, when either the architectural description's correspondence is in danger, or the group productiveness is declining. All workgroups do share the same repository and can see each other's works. Moreover, groups actively challenge each other to review intermediate products. On the steering group, the steering group needs to be actively involved with the production, but as team members instead of managers. As shown in other works, a distant steering group leads to mismanagement. By letting the steering group participate in the development cycle, one eliminates this distance. Moreover, this steering group can then aid the correspondence of the EA descriptions.

Additionally, one should consider that EA development is a creative process and benefits from treating the process as a creative one. This creativity requires freedom for the development team and management by exception. Despite being a creative process, structured working methods need to be used, and documentation is necessary. However, documenting can kill the creative spirit; there is a need to find a balance between what to document and what not to document.

All in all, this study can be used to make known approaches more practical by using an agile development approach. This approach should be characterised by direct stakeholder involvement. In that case, one can test the intermediate products and aid the stakeholder decision–making by using prototypes; this is the idea of main viewpoint and stakeholder specific viewpoints. Agility is necessary, do not try to create an EA description in one go; start small and grow big. Do this iteratively, first analyse known information and sketch the knowledge into a description. Subsequently, consolidate and review the sketches into a more formal description. As is agreed upon in the project framework, document the description to compare the description to other relative descriptions and refine. Finally, gather & document feedback, test the work and begin a new iteration.

Further, the stakeholders and their concerns are structured in viewpoints embedded in the project framework's architectural artefact. When creating the project framework, decide what to document and what not to. Trying to document everything kills the creative process. During the process, one should not blindly follow standards or methods but instead use pragmatism. It is better to bring in motion the desired effects on the enterprise than to be correct. When considering the EA development process, one should consider the framework or methods and the project team's organisation. The team composition should be diverse in terms of knowledge on the enterprise with a common ground and gain the freedom in organisation and budget to carry out the creative process of developing EA descriptions. On the other hand, the steering should manage by exception when group productivity is declining or description correspondence needs to be ensured. This form of management is best achieved by minimising the distance between steering and producing.

8.1 FUTURE WORK

This research was limited to only one possible domain within the public sector, namely education. It would be insightful if other disciplines, such as healthcare, ministries and police, could be investigated. As a result, one can compare the different sectors and investigate the found similarities further. Consequently, a development approach born out of practice could be formalised for the public sector. On the other hand, it would also be interesting to investigate the private sector, which has a much stricter regulation on sharing information. Recall that the found development approach relied on information availability. The same research could also be conducted within the education sector, when developing non–reference architecture, investigating similarities and discrepancies. Moreover, the missing architecting steps, such as migration planning, would become insightful due to the nature of non–reference architecture. Additionally, further studies in viewpoint management, requirement management and architecting styles with their benefits and drawbacks when compared to each other can help enterprises select their perfect fit for EA management.

Finally, since the perceived process relied on available information, it would be interesting to perceive an architecting process in an organisation with a large architecture debt. In other words, an organisation new to architecting, which is just beginning to describe and align its architecture.

Acknowledgments

I want to thank Drs. Bas Kruiswijk and Dr. Werner Heijstek (Leiden University) for supervising this study. I thank saMBO–ICT for cooperating with this study and allowing an internship to conduct the study. My thanks go out to Drs. Frans van Neerbos for supervising the internship, his helpful council and his assistance in this study. Finally, I would like to thank the entire project team of route 21 for their active participation, time and helpful suggestions.

References

[1] 1996. The MEMA-model: towards a new approach for method engineering. *Information and Software Technology* 38, 4 (1996), 295–305. https://doi.org/10.1016/0950-5849(95)01087-4

[2] 2011. ISO/IEC/IEEE Systems and software engineering – Architecture description. *ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000)* (2011), 1–46.

[3] Nur Azaliah Abu Bakar, Nazri Kama, and Harihoddin Selamat. 2016. Enterprise architecture development and implementation in public sector: The Malaysian perspective. 88 (06 2016), 176–188. https://www.researchgate.net/publication/304942589_Enterprise_architecture_development_and_implementation_in_public_sector_The_Malaysian_perspective

[4] Ibrahim Abunadi. 2019. Enterprise Architecture Best Practices in Large Corporations. *Information* 10 (09 2019), 293. https://doi.org/10.3390/info10100293

[5] Hamood Al-Kharusi, Suraya Miskon, and Mahadi Bahari. 2018. Enterprise Architecture Development Approach in the Public Sector. *International Journal of Enterprise Information Systems* 14 (10 2018), 124–141. https://doi.org/10.4018/IJEIS.2018100109

[6] Samar Alamri, Manal Abdullah, and Adnan Albar. 2018. Enterprise Architecture Adoption for Higher Education Institutions. *International Journal of Simulation: Systems, Science and Technology* 19 (10 2018), 16.1–16.8. https://doi.org/10.5013/IJSSST.a.19.05.16

[7] Ayed Alwadain, Erwin Fielt, Axel Korthaus, and Michael Rosemann. 2016. Empirical insights into the development of a service-oriented enterprise architecture. *Data & Knowledge Engineering* 105 (2016), 39–52. https://doi.org/10.1016/j.datak.2015.09.004

[8] Gonçalo Antunes, Artur Caetano, Marzieh Bakhshandeh, Rudolf Mayer, and José Borbinha. 2013. Using Ontologies to Integrate Multiple Enterprise Architecture Domains, Vol. 160. 61–72. https://doi.org/10.1007/978-3-642-41687-3_8

[9] Farhad Arbab, F. Boer, Marcello Bonsangue, Marc Lankhorst, and Leon van der Torre. 2007. Integrating Architectural Models. (01 2007).

[10] Command & Control Board Architecture Capability TeamConsultation and Command & Control Board Architecture Capability Team Consultation. 2020. . nato.int. https://doi.org/nato_static_fl2014/assets/pdf/2020/9/pdf/NAFv4_2020.07.pdf

[11] Bureau Edustandaard Architectuurraad. 2019. Hoofdpagina. https://www.wikixl.nl/wiki/rosa/index.php/Hoofdpagina
[12] Frank Armour and Stephen Kaisler. 2002. *Enterprise Systems Architecting.* https://doi.org/10.1002/0471028959.sof104
[13] F.j. Armour, S.h. Kaisler, and S.y. Liu. 1999. A big-picture look at enterprise architectures. *IT Professional* 1, 1 (1999), 35–42. https://doi.org/10.1109/6294.774792
[14] Frank Armour, Stephen Kaisler, and S.Y. Liu. 1999. Building an Enterprise Architecture Step-by-Step. *IT Professional* 1 (08 1999), 31 – 39. https://doi.org/10.1109/6294.781623
[15] Muhammad Ali Babar, Alan W. Brown, and Ivan Mistrik. 2014. *Agile software architecture aligning agile processes and software architectures.* Elsevier/MK, Morgan Kaufmann.
[16] Martin van den Berg and Marlies van. Steenbergen. 2004. *DYA: stap voor stap naar professionele enterprise-architectuur.* Soveti Nederland B.V. en ten Hagen Stam b.v.
[17] Harry Bouwman, Harrie Van Houtum, Marijn Janssen, and Gerrit Versteeg. 2011. Business Architectures in the Public Sector: Experiences from Practice. *Communications of the Association for Information Systems* 29 (2011). https://doi.org/10.17705/1cais.02923
[18] Ruth Breu, Matthias Farwick, Matheus Hauder, Florian Matthes, and Sascha Roth. 2013. Enterprise Architecture Documentation: Current Practices and Future Directions. https://www.researchgate.net/publication/257383200_Enterprise_Architecture_Documentation_Current_Practices_and_Future_Directions
[19] CAUDIT. 2021. https://www.caudit.edu.au/
[20] Gabriel Cepeda-Carrion. 2010. Competitive Advantage of Knowledge Management. *Knowledge Management* (Jan 2010), 2646–2659. https://doi.org/10.4018/978-1-59904-933-5.ch216
[21] Duong Dang and Samuli Pekkola. 2016. Systematic Literature Review on Enterprise Architecture in the Public Sector. *Electronic Journal of e-Government* (12 2016).
[22] Bert Dingemans. 2011. Stakeholder behoeften beschrijven binnen Togaf 9. *VIA NOVA ARCHITECTURA* (Sep 2011). http://www.interactory.nl/documenten/stakeholderbehoeften.pdf
[23] DoDAF. [n.d.]. *6-steps_process.* U.S. Department of Defense. https://dodcio.defense.gov/Portals/0/Images/dodaf20/6-step_process.jpg
[24] Bureau Edustandaard. [n.d.]. Afspraken Archive. https://www.edustandaard.nl/standaard_afspraken/
[25] Bureau Edustandaard. 2020. AMIGO: methodiek om te komen tot bouwbare uitwisselspecificaties. https://www.edustandaard.nl/amigo/
[26] David Emery and Rich Hilliard. 2009. Every Architecture Description Needs a Framework: Expressing Architecture Frameworks Using ISO/IEC 42010. 31 – 40. https://doi.org/10.1109/WICSA.2009.5290789
[27] Kennispunt MBO Onderwijs en Examinering. 2019. Hoofdpagina TPO. https://tpo-mbo.nl//tpo/index.php?title=Hoofdpagina
[28] Ahmad Fajar, Aldian Nurcahyo, and Septia Redisa. 2018. SOA SYSTEM ARCHITECTURE FOR INTERCONECTED MODERN HIGHER EDUCATION IN INDONESIA. *Procedia Computer Science* 135, 354–360. https://doi.org/10.1016/j.procs.2018.08.184
[29] D Garlan, Robert T. Monroe, and D Wile. 1997. Acme: an architecture description interchange language. *CASCON* (1997). https://doi.org/10.1145/782010.782017
[30] Frank Goethals, Monique Snoeck, Wilfried Lemahieu, and Jacques Vandenbulcke. 2006. Management and enterprise architecture click: The FAD(E)E framework. *Information Systems Frontiers* 8 (02 2006), 67–79. https://doi.org/10.1007/s10796-006-7971-1
[31] Richard Hilliard. 2012. Knowledge mechanisms in IEEE 1471/ISO 42010.
[32] Rich Hilliard. 2014. All About ISO/IEC/IEEE 42010 version 2014-r5 (PPT).
[33] Tl Hope. 2015. The critical success factors of enterprise architecture.
[34] S.J.B.A. Hoppenbrouwers, A.I. Bleeker, and H.A. Proper. 2004. .
[35] S.J.B.A. Hoppenbrouwers, Henderik Alex Proper, and Victor van Reijswoud. 2005. Navigating the Methodology Jungle: The communicative role of modelling techniques in information system development. *Computing Letters* 1 (2005), 97–106.
[36] ICTU. [n.d.]. NORA online. https://www.noraonline.nl/wiki/NORA_online
[37] ICTU. 2001. Over ons. https://www.ictu.nl/over-ons
[38] ICTU. 2018. NORA. https://www.digitaleoverheid.nl/overzicht-van-alle-onderwerpen/basisregistraties-en-stelselafspraken/nora/
[39] Internet2.edu. 2020. Internet2 Wiki. https://spaces.at.internet2.edu/pages/viewpage.action?pageId=98306902
[40] Mistrik Ivan. 2014. *Economics-driven software architecture.* Elsevier, Morgan Kaufmann.
[41] Chanki Jeong. 2012. EA as an Actionable Architecture. *Journal of Information Technology and Architecture* 9, 2 (Jun 2012), 133–142.
[42] Stephen Kaisler and Frank Armour. 2017. 15 Years of Enterprise Architecting at HICSS: Revisiting the Critical Problems. https://doi.org/10.24251/HICSS.2017.585

[43] Stephen Kaisler, Frank Armour, and Michael Valivullah. 2005. Enterprise Architecting: Critical Problems. (01 2005). https://doi.org/10.1109/HICSS.2005.241
[44] Anjali Kaushik and Aparna Raman. 2015. The new data-driven enterprise architecture for e-healthcare: Lessons from the Indian public sector. *Government Information Quarterly* 32, 1 (2015), 63–74. https://doi.org/10.1016/j.giq.2014.11.002
[45] Kennisnet. [n.d.]. Informatieobjecten. https://www.rio-onderwijs.nl/informatieobjecten-2
[46] Kennisnet. 2021. Leeswijzer. https://www.wikixl.nl/wiki/fora/index.php/Leeswijzer
[47] Philippe Kruchten. 1995. Architectural Blueprints—The "4+1" ViewModel of Software Architecture. *IEEE Software* 12, 6 (Nov 1995), 42–50. https://doi.org/~gregor/teaching/papers/4+1view-architecture.pdf
[48] Philippe Kruchten, Rafael Capilla, and Juan Dueñas. 2009. The Decision View's Role in Software Architecture Practice. *IEEE Software* 26 (03 2009), 36–42. https://doi.org/10.1109/MS.2009.52
[49] MARC LANKHORST. 2019. *ENTERPRISE ARCHITECTURE AT WORK: modelling, communication and analysis*. SPRINGER.
[50] Åsa Lindström, Pontus Johnson, Erik Johansson, Mathias Ekstedt, and Mårten Simonsson. 2006. A survey on CIO concerns-do enterprise architecture frameworks support them? *Information Systems Frontiers* 8 (02 2006), 81–90. https://doi.org/10.1007/s10796-006-7972-0
[51] Shun Chi Ryan Lo and Ning Chen. 2017. IEEE 42010 and Agile Process- Create Architecture Description through Agile Architecture Framework.
[52] Rik Maes. 1999. Informatiemanagement in kaart gebracht. *Intertax* 99 (01 1999). https://doi.org/10.5117/mab.77.12766
[53] Judy Matthews and Arthur Shulman. 2005. Competitive advantage in public-sector organizations: Explaining the public good/sustainable competitive advantage paradox. *Journal of Business Research* 58 (02 2005), 232–240. https://doi.org/10.1016/S0148-2963(02)00498-8
[54] MBO-Raad. 2021. Taxonomie documenten MBO gereed - saMBO-ICT. https://www.sambo-ict.nl/2012/07/taxonomie-documenten-mbo-gereed/
[55] Nenad Medvidovic, David Rosenblum, David Redmiles, and Jason Robbins. 2002. Modeling software architectures in the Unified Modeling Language. *Medvidovic, N. and Rosenblum, D.S. and Redmiles, D.G. and Robbins, J.E. (2002) Modeling software architectures in the unified modeling language. ACM Transactions on Software Engineering and Methodology, 11 (1). pp. 2-57. ISSN 1049331X* 11 (01 2002). https://doi.org/10.1145/504087.504088
[56] Ivan Mistrik. 2014. *Relating system quality and software architecture*. Elsevier, Morgan Kaufmann is an imprint of Elsevier.
[57] Klaus Niemann. 2008. *Enterprise Architecture Management and its Role in IT Governance and IT Investment Planning*. 208–228. https://doi.org/10.4018/978-1-60566-068-4.ch010
[58] Naval Aviation Enterprise: Chief Technology Office. 2015. ISO/IEC/IEEE 42010.
[59] DOD Deputy Chief Information Officer. 2010. DODAF - DOD Architecture Framework Version 2.02 - DOD Deputy Chief Information Officer. https://dodcio.defense.gov/Library/DoD-Architecture-Framework/
[60] Dag H. Olsen and Kjersti Trelsgård. 2016. Enterprise Architecture Adoption Challenges: An exploratory Case Study of the Norwegian Higher Education Sector. *Procedia Computer Science* 100 (2016), 804 – 811. https://doi.org/10.1016/j.procs.2016.09.228 International Conference on ENTERprise Information Systems/International Conference on Project MANagement/International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2016.
[61] Kennispunt MBO onderwijs & examinering. [n.d.]. De Procesarchitectuur Examinering (PE). https://pe.onderwijsenexaminering.nl/
[62] The open group. 2014. *Business Scenario: Open Public Sector Data Reference: K140* (k140 ed.). The Open group. https://publications.opengroup.org/togaf-library/industry-specific/k140
[63] The open group. 2018. *34_contentfwk5*. The open group. https://pubs.opengroup.org/architecture/togaf9-doc/m/Figures/34_contentfwk5.png
[64] The open group. 2018. *adm*. The open group. https://pubs.opengroup.org/architecture/togaf9-doc/m/Figures/adm.png
[65] The open group. 2018. *Structure*. The open group. https://pubs.opengroup.org/architecture/togaf9-doc/m/Figures/01_structure.png
[66] The open group. 2018. *The TOGAF standard, version 9.2*. Van Haren Publishing.
[67] The open group. 2019. *ArchiMate® 3.1 specification*. The open group. https://pubs.opengroup.org/architecture/archimate31-doc/toc.html
[68] The open group and The open group. 2018. *2. Core Concepts*. Van Haren Publishing, 9–23. https://pubs.opengroup.org/architecture/togaf9-doc/m/chap02.html
[69] The open group and The open group. 2018. *20. Architecture Principles*. Van Haren Publishing, 223–265. https://pubs.opengroup.org/architecture/togaf9-doc/m/chap20.html
[70] Practoraten.nl. 2017. Onderzoek. https://www.practoraten.nl/onderzoek/
[71] Rijksoverheid. 2021. https://www.digitaleoverheid.nl/

[72] Álvaro Rocha and Filipe Sá. 2013. Planning the information architecture in a local public administration organization. *Information Development* 30 (07 2013), 223–234. https://doi.org/10.1177/0266666913489841

[73] Jeanne W. Ross, Peter Weill, and David C. Robertson. 2006. *Enterprise architecture as strategy: creating a foundation for business execution*. Harvard Business School Press.

[74] Babak Darvish Rouhani, Mohd Naz'ri Mahrin, Fatemeh Nikpay, Rodina Binti Ahmad, and Pourya Nikfard. 2015. A systematic literature review on Enterprise Architecture Implementation Methodologies. *Information and Software Technology* 62 (2015), 1 – 20. https://doi.org/10.1016/j.infsof.2015.01.012

[75] Route21. 2021. *Domeinen Hoofdprocesmodel*. Sambo-ICT. https://route21.wikixl.nl/index.php/Hoofdpagina

[76] Jungwoo Ryoo and Hossein Saiedian. 2006. A framework for classifying and developing extensible architectural views. *Information and Software Technology* 48, 7 (2006), 456 – 470. https://doi.org/10.1016/j.infsof.2005.05.006

[77] Patrick Saint-Louis and James Lapalme. 2016. Investigation of the lack of common understanding in the discipline of enterprise architecture : A systematic mapping study. *2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW)* (2016). https://doi.org/10.1109/edocw.2016.7584364

[78] Sambo-ICT. 2007. Triple A. https://triplea.sambo-ict.nl/processen_binnen_triple_a.html

[79] Felix Sanchez-Puchol, Joan A. Pastor-Collado, and Baptista Borrell. 2017. Towards an Unified Information Systems Reference Model for Higher Education Institutions. *Procedia Computer Science* 121 (2017), 542 – 553. https://doi.org/10.1016/j.procs.2017.11.072 CENTERIS 2017 - International Conference on ENTERprise Information Systems / ProjMAN 2017 - International Conference on Project MANagement / HCist 2017 - International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN/HCist 2017.

[80] SBB. 2019. Het Nederlandse onderwijssysteem. https://www.s-bb.nl/studenten/diplomawaardering-en-onderwijsvergelijking/het-nederlandse-onderwijssysteem

[81] Daniel Simon, Kai Fischbach, and Detlef Schoder. 2013. An Exploration of Enterprise Architecture Research. *CAIS* 32 (2013), 1.

[82] Rob Stovers, Jules de Ruijter, and R. Wagter. 2019. *GEA enterprisearchitectuur in de praktijk: betere prestaties door sturen op samenhang*. Dialoog.

[83] SURF. 2018. Hoofdpagina. https://hora.surf.nl/index.php/Hoofdpagina

[84] UWV. 2020. Homepage - STAP-budget. https://www.stap-budget.nl/

[85] Katharina Winter, Sabine Buckl, Florian Matthes, and Christian Schweda. 2010. Investigating the State-of-the-Art in Enterprise Architecture Management Methods in literature and Practice. 90.

[86] Robert K. Yin. 2014. *Case study research: design and methods*. Sage Publication.

[87] John Zachman. 1999. A Framework for Information Systems Architecture. *IBM Systems Journal* 38 (01 1999), 454–470. https://doi.org/10.1147/sj.382.0454

[88] Zuoqin Zhang. 27-03-2020. *MASTER'S THESIS OF ICT in Business and the Public Sector: University of Leiden* (27-03-2020). https://theses.liacs.nl/pdf/2019-2020-ZhangZuoqin.pdf

A Project Timeline

Project Timeline

Date	Description
May 2019	• A report on the analysis of currently used architectural descriptions in the MBO sector concludes that current descriptions need to be harmonised on the short term. On the long term, a new architecture description needs to be developed. • The Leusden Group has its first meeting.
June 2019	• Harmonisation starts. • Project definition, planning and budget for the development of a new architecture reference description is created. The project is to be named Route21.
July 2019	• The Project is introduced to the IT-panel. • Several kick-offs presentations are held through the MBO sector.
September 2019	• The information Manager network is invited to help Route21, such as identifying stakeholders, tooling, metamodels, suggestions and possible team members. • A saMBO-ICT conference is held, route21 is actively marketed. • The relation of the reference architecture of higher-education and route21 is decided.
November 2019	• Results of research to the architectural maturity of MBO institutions are presented. • Deliverables and the definition of done are made more concrete. • The meta–model and framework are chosen.
December 2019	• All known internal and external architecture is gathered into a repository. • Active cooperation is sought with OCW, SURF and Kennisnet.
January 2020	• The first activity in the development process starts. • A presentation is given to several layers of management on the need of architecture, the purpose of route21 and how it will aid them. Representatives for the active participation of the managers in the development process are sought. • Covid-19 Virus enters Dutch Borders.
March 2020	• The country goes into lockdown due to Covid-19. • The copy and paste session are held with the light traffic model. • The framework for the main–process model is developed. • The questionnaire is held for evaluation of known architecture descriptions on the main–process model. • A saMBO-ICT conference is held, route21 is actively marketed.
April 2020	• Workgroups A, B and C are formed based on Archimate 3.0 layers. • Earlier identified stakeholders are formalised and prioritised. • Working agreements of workgroups are formalised. • Lockdown due to Covid-19 is partially lifted.
May 2020	• The relevance of custom viewpoints based on personas is discussed. • The centralisation discussion has become more prominent. • A recap of ISO 42020 is given, and propositions for viewpoint templates are held.
August 2020	• Subprocesses become officially skewers.
September 2020	• A saMBO-ICT conference is held, route21 is actively marketed, feedback is requested. • Research into the concerns of stakeholders start. • Project architecture principles are made explicit.
November 2020	• Findings of stakeholder concerns are presented. • The country goes into lockdown again due to Covid-19.
December 2020	• All architectural descriptions are brought to version 0.9.
January 2021	• Steering group consolidates and harmonises architectural descriptions and publicises version 1.0. • The next development cycle starts for version 1.1 – 2.0.

Project Timeline

B DODAF

View-point	Models	General Description
All View-point	AV-1 Overview and Summary Information	This model describes a Project's Visions, Goals, Objectives, Plans, Activities, Events, Conditions, Measures, Effects (Outcomes), and produced objects.
	AV-2 Integrated Dictionary	This model is an architectural data repository with definitions of all terms used throughout the architectural data and presentations.
Capa-bility View-point	CV-1: Vision	Addresses the enterprise concerns associated with the overall vision for transformational endeavours and thus defines the strategic context for a group of capabilities.
	CV-2: Capability Taxonomy	This model captures capability taxonomies. The model presents a hierarchy of capabilities. These capabilities may be presented in the context of a timeline - i.e., it can show the required qualifications for current and future capabilities.

	CV-3: Capability Phasing	The planned achievement of capability at different points in time or during specific periods. The CV-3 shows the capability phasing in terms of the activities, conditions, desired effects, rules complied with, resource consumption and production, and measures, without regard to the performer and location solutions
	CV-4: Capability Dependencies	Here the dependencies between planned capabilities and the definition of logical groupings of capabilities are discussed.
	CV-5: Capability to Organisational Development Mapping	The fulfilment of capability requirements shows the planned capability deployment and interconnection for a particular Capability Phase. The CV-5 offers the intended solution for the phase in terms of performers and locations and their associated concepts.
	CV-6: Capability to Operational Activities Mapping	A mapping between the capabilities required and the operational activities that those capabilities support.
	CV-7: Capability to Services Mapping	This model is a mapping between the capabilities and the services that these capabilities enable.
Data and Information	DIV-1: Conceptual Data Model	Results are the required high-level data concepts and their relationships.
	DIV-2: Logical Data Model	The documentation of the data requirements and structural business process (activity) rules. In DoDAF V1.5, this was the OV-7.
	DIV-3: Physical Data Model	Resulting in the physical implementation format of the Logical Data Model entities, e.g., message formats, file structures, physical schema. In DoDAF V1.5, this was the SV-11.
Operational Viewpoint	OV-1: High-Level Operational Concept Graphic	The high-level graphical/textual description of the operational concept.
	OV-2: Operational Resource Flow Description	A description of the Resource Flows exchanged between operational activities.
	OV-3: Operational Resource Flow Matrix	A description of the resources exchanged and the relevant attributes of the exchanges.
	OV-4: Organisational Relationships Chart	This model describes the Organisational context, role or other relationships among Organisations.
	OV-5a-b: Operational Activity Decomposition Tree	This model describes the capabilities and activities (operational activities) organized in a hierarchal structure.

	OV-6a-c: Operational Rules Model	This model describes one of three models used to describe the activity (operational activity).
Project Viewpoint	PV-1: Project Portfolio Relationships	It describes the dependency relationships between the Organisations and projects and the Organisational structures needed to manage a portfolio of projects.
	PV-2: Project Timelines	This model describes a timeline perspective on programs or projects, with the key milestones and interdependencies.
	PV-3: Project to Capability Mapping	A mapping of programs and projects to capabilities shows how the specific tasks and program elements help achieve a capability.
Services Viewpoint	SvcV-1 Services Context Description	This model describes the identification of services, service items, and their interconnections.
	SvcV-2 Services Resource Flow Description	A description of Resource Flows exchanged between services.
	SvcV-3a-b Systems-Services Matrix	This model describes the relationships among or between systems and services in each Architectural Description.
	SvcV-4 Services Functionality Description	The functions performed by services and the service data flows among service functions (activities).
	SvcV-5 Operational Activity to Services Traceability Matrix	Mapping of services (activities) back to operational activities (actions).
	SvcV-6 Services Resource Flow Matrix	It provides details of service Resource Flow elements being exchanged between services and the attributes of that exchange.
	SvcV-7 Services Measures Matrix	This model describes the measures (metrics) of Services Model elements for the appropriate timeframe(s).
	SvcV-8 Services Evolution Description	This model describes the planned incremental steps toward migrating a suite of services to a more efficient suite or toward evolving current services to future implementation.
	SvcV-9 Services Technology Skills Forecast	The emerging technologies, software/hardware products, and skills expected to be available in each set of time frames will affect future service development.
	SvcV-10a-c Services Rules Model	This model describes one of three models used to describe service functionality.

Standards Viewpoint	StdV-1 Standards Profile	This model describes the listing of standards that apply to solution elements.
	StdV-2 Standards Forecast	This model describes emerging standards and potential impact on current solution elements, within a set of time frames.
Systems Viewpoint	SV-1 Systems Interface Description	This model describes the identification of systems, system items, and their interconnections.
	SV-2 Systems Resource Flow Description	A description of Resource Flows exchanged between systems.
	SV-3 Systems-Systems Matrix	This model describes the relationships among systems in each Architectural Description. It can be designed to show interest relationships, (e.g., system-type interfaces, planned vs existing interfaces).
	SV-4 Systems Functionality Description	The functions (activities) performed by systems and the system data flows among system functions (exercises).
	SV-5a Operational Activity to Systems Function Traceability Matrix	A mapping of system functions (activities) back to operational activities (actions).
	SV-5b Operational Activity to Systems Traceability Matrix	A mapping of systems back to capabilities or operational activities (activities).
	SV-6 Systems Resource Flow Matrix	This model provides details of system resource flow elements being exchanged between systems and the attributes of that exchange.
	SV-7 Systems Measures Matrix	This model describes the measures (metrics) of Systems Model elements for the appropriate timeframe(s).
	SV-8 Systems Evolution Description	This model describes the planned incremental steps toward migrating a system suite to a more efficient suite of evolving a current system to future implementation.
	SV-9 Systems Technology Skills Forecast	The emerging technologies, software/hardware products, and skills expected to be available in each set of time frames will affect future system development.
	SV-10a-c Systems Rules Model	This model describes one of three models used to define system functionality.

Table 8. DoDAF's main viewpoints and models (views) described [59].

C TOGAF'S DELIVERABLES

Deliverable	Output from	Input to
Architecture Building Blocks	F, H	A, B, C, D, E
Architecture Contract	-	-
Architecture Definition Document	B, C, D, E, F	C, D, E, F, G, H
Architecture Principles	Preliminary, A, B, C, D	Preliminary, A, B, C, D, E, F, G, H
Architecture Repository	Preliminary	Preliminary, A, B, C, D, E, F, G, H, Requirements Management
Architecture Requirements Specification	B, C, D, E, F, Requirements Management	C, D, Requirements Management
Architecture Roadmap	B, C, D, E, F	B, C, D, E, F
Architecture Vision	A, E	B, C, D, E, F, G, H, Requirements Management
Business Principles, Business Goals, and Business Drivers	Preliminary, A, B	A, B
Capability Assessment	A, E	B, C, D, E, F
Change Request	F, G, H	-
Communications Plan	A	B, C, D, E, F
Compliance Assessment	G	H
Implementation and Migration Plan	E, F	F
Implementation Governance Model	F	G, H
Organisational Model for Enterprise Architecture	Preliminary	Preliminary, A, B, C, D, E, F, G, H, Requirements Management
Request for Architecture Work	Preliminary, F, H	A, G
Requirements Impact Assessment	Requirements Management	Requirements Management
Solution Building Blocks	G	A, B, C, D, E, F, G
Statement of Architecture Work	A, B, C, D, E, F, G, H	B, C, D, E, F, G, H, Requirements Management
Tailored Architecture Framework	Preliminary, A	Preliminary, A, B, C, D, E, F, G, H, Requirements Management

Table 9. TOGAF's Deliverables [66].

D DATA

ID	Source	Description
D01	Interview	This document is an interview with a stakeholder, security architect, of the new reference architecture description.
D02	Interview	This document is an interview with a stakeholder, Information manager, of the new reference architecture description.
D03	Interview	This document is an interview with a stakeholder, business manager, of the new reference architecture description.

D04	Interview	This document is an interview with a stakeholder, quality assurance manager, of the new reference architecture description.
D05	Interview	This document is an interview and retrospective of the architectural development process with a member from the steering group, coordinating workgroup B (H02)
D06	Interview	This document is an interview and retrospective of the architectural development process with a member of workgroup A (H07)
D07	Interview	This document is an interview and retrospective of the architectural development process with a member of workgroup A B (H06)
D08	Interview	This document is an interview and retrospective of the architectural development process with the head of the steering group (H01)
D09	Interview	This document is an interview and retrospective of the architectural development process with a member of workgroup A (H11)
D10	Interview	This document is an interview and retrospective of the architectural development process with a member of workgroup C (H08)
D11	Questionnaire	This document is the results of evaluating several business architectural descriptions send in from MBO institutions.
D12	Questionnaire	This document is the results of a questionnaire identifying stakeholder group concerns and suggestions aggregated into their respective groups
D13	Questionnaire	This document is the results of a questionnaire identifying stakeholder group concerns and suggestions.
D14	Questionnaire	This document is the results of a questionnaire prioritising stakeholder groups
D15	Questionnaire	This document is the raw data of the concern questionnaire
D16	Internal Presentation	This document is a presentation presenting stakeholder concerns and suggestions
D17	Internal Presentation	This document is a presentation about the different option to model the information architecture description
D18	Internal Document	This document tries to classify different stakeholders found within the MBO environment into stakeholder groups
D19	Internal Document	This document is a document which provides a rudimentary sketch of MBO environment
D20	Internal Presentation	This document is a presentation provided to a select few which presents version 0.9 of the business architecture description
D21	Presentation	This document is a presentation which explains the main–process (business function) architecture description
D22	Internal Document	These are two documents which describe the project and its planning
D23	Internal Document	These are two reports on the first meeting of the Leusden group
D23	Presentation	The presentation was given with the project kick-off
D24	Presentation	This document is the presentation was given at the September conference of 2019

D25	Internal Research	This document is created by an external researcher on the maturity level of individual MBO institutions
D26	Internal Document	This document is a report on the meeting which dedicated the meta–model to be used
D27	Internal Presentation	This document is the presentation governing the copy and pastes session
D28	Internal Presentation	This document is the presentation used with D11; it contains all the internal business architectural descriptions.
D29	Internal Presentation	The presentation used to introduce the workgroups and the goals for the next months
D30	Internal Presentation	This document is a presentation used as a reference during the work. It summarises ISO 42020
D31	Internal Template	This document is a template which was meant to be used for the creation of viewpoints
D32	Internal Document	This document describes agreements, workgroups and the tooling to be used
D33	Internal Document	This document describes the relationship between the distinguished process business function groups and the most important stakeholder group
D34	Internal Document	This document presents the result of the copy and pastes session
D35	Internal Document	These documents present the questionnaires' results about making architectural decisions and evaluating the different business architecture descriptions of MBO institutions.
D36	Internal Presentation	This document is the presentation used in the meeting about the meta–model (D26)
D37	Internal Presentation	This document is the presentation of SURF which describes the envisioned relation between HORA and the MBO
D38	Internal Presentation	This document is the standard presentation used to introduce the project
D39	Internal Presentation	This document is the internal presentation to management describing the relation between EA and their goals
D40	Internal Document	This document is the results of the 'management game.'
D41	Internal Archive	These are several documents describing the architectural principles of the project
D42	Internal Memo	This email describes the sprint planning of a sprint of workgroup B
D43	Internal Memo	This memo describes the current situation of the project and the sprint planning of workgroup B

D44	Internal Archive	This series of documents describe the feedback of a stakeholder group on the architecture descriptions
D45	Internal Document	This document describes the relation between business-main–process (function) and sub-processes.
D46	Internal Archive	These are several documents which report on the meetings during all sprints of workgroup C
D47	Internal Document	This document was used to identify business and information objects
D48	Internal Document	This document was used to identify application services and components
D49	Internal Archive	This document is the source code of the architectural descriptions
D50	Online Publication	This source is the publicized reference architecture descriptions (still in development)
D51	Internal Document	This document describes a proposed approach for the year 2020
D52	Internal Document	This document is the presentation of the sambo conference in Jan 2020

Table 10. Gathered project documents.

E PROJECT TEAM COMPOSITION

Position	**Started in Workgroup**	**Affiliation**	**Relevant Years of experience**	**Previous experience**	**Reference**
Change Manager	Core group	saMBO-ICT	28	Strategic Planner Program manager processes and innovation ICT Manager Interim Consultant	H01
External Consultant	Core group	External Consultancy Company	30	Software Engineer Lecturer ICT Architecture Academic University	H02
Researcher	Core Group	Academic University	2	Cybersecurity employee	H03
ICT architect	Workgroup B	An MBO institution	24	Senior Engineer Coordinator Technical Innovation	H04

Domain manager education standards and chain reference architecture (ROSA FORA)	Workgroup C	Kennisnet	25	Network Engineer Product Manager Manager education standards Head Architectural board	H05
Information Architect	Workgroup A B	An MBO institution	23	Head IT Project Manager IT Information Manager	H06
Information manager	Workgroup A	An MBO institution	40	Information Manager Program Manager ICT	H07
Security Officer	Workgroup C	An MBO institution	39	Software Analyst IT Consultant ICT Architect Business Policy Officer	H08
Data Architect	-	An MBO institution	32	Lecturer IT logistics at the University of applied sciences Enterprise Architect	H09
Information Manager	Workgroup B	An MBO institution	37		H10
Information Manager	Workgroup A	An MBO institution	30	Team leader of the economics education program Program manager investments	H11
Line Manager Information Management	Workgroup A	An MBO institution	28	Enterprise Architect Information Manager CEO	H12
Information manager	Workgroup B	An MBO institution	25	Lecturer Physics at the MBO Lecturer Chemistry at the MBO Application Functionality Manager	H13
Information manager	Workgroup C	An MBO institution	40	Innovation consultant Consultant education ICT development Senior consultant education	H14
Enterprise Architect	Workgroup C A	An MBO institution	28	Project Manager Transition manager Innovation manager Security Architect ICT Consultant	H15

Business Policy Officer	Workgroup C	An MBO institution	30	Team leader chemistry education program Infrastructure manager Team leader ICT education program Team Leader system and network education program Senior Consultant business and ICT Enterprise Architect	H16
Information manager	Workgroup C	An MBO institution	9	Information architect Information policy officer	H17
IT Architect	Workgroup B	An MBO institution	16	ICT Engineer Technical Project Lead IT Consultant	H18
Process Analyst	Workgroup B	An MBO institution	24	Business Architect Senior Consultant process management Process Controller	H19

Table 11. Team composition.

F TEMPLATE TRAFFIC LIGHT

AD name	**Domain**	**Architectural Layer**	**Type (object, relation or pattern)**	**Name**	**Evaluation**	**Rationale**	**Notes**

Table 12. Traffic light template.

G QUESTIONNAIRE MAIN PROCESS MODEL EVALUATION

- What is the most significant difference in this description when compared to the default reference architecture (triple–A & TPO), in your opinion?
- What are the strong points?
- What are the weak points?
- Any other remarks?

Evaluation Criteria	**Abysmal**	**Bad**	**Neutral**	**Good**	**Very Good**	**Do not know**
Chosen abstraction level						
Used selection of elements (processes functions)						
Used selection of domains (process or function groups)						
Used relation between elements						
Used relation between domains						

The general selection of content						
Content alignment on the proposed stakeholder group (management)						
Used visualisation of elements						
Used visualisation of domains						
Used visualisation of relations						
The used colour palette						
General visualisation						
Visual alignment on the proposed stakeholder group (management)						
Clarity and delimitations of responsibilities						
Relationship to data models						
Relationship to application models						
Chosen model kind (diagram, spreadsheet, a text document, etc.)						

Table 13. Questionnaire template

H RANKING STAKEHOLDER GROUPS

Rank	**Stakeholder group**	**Stakeholder examples**
1	Information management	Architectural Board, CIO, management team, saMBO-ICT, data officer, security officer, project managers, …
2	Business management	Governance, strategy policy officers, project managers, innovation change management, executive board, MBO-raad, …
3	Educative management	Education (operation) manager, Educative teams, Education coordinator, principals, …
4	IT-infrastructure management	CTO, Management team, project manager, security officer, administrators…
5	Quality Assurance	Accreditation Organisation, Dutch Data Protection Authority, Ministry of Education (OCW), Government Education Inspectorate, Quality Assurance (QA) Compliance Department, Accountancy Bureau, …
6	Information communication operations (functional management)	Cyber Security Privacy Department, ICT Department, Service Management Department, Architect, Data Officer, Functional Management, Information Manager / Knowledge Transfer Coordinator, Business Intelligence Department, Access Management, …
7	Education Administration Scheduling (Education Logistics)	Student Resources Department (SR), DUO, Education Administration, Teacher, Education Team, Registrar, Scheduling Planning Office, Student, Examination Group, Education Support, …

8	Suppliers	Supplier, xAAS (PAAS, SAAS, ...), Software Supplier, Publisher Distributor (Educational Materials), …
9	Education operations	Prospective students, internships Group, Foundation Vocational Education, MBO, Teachers, Education team, Student, Examination group, Adult education (VAVO), …
10	ICT innovation	Prospective Students, Teacher, Education Team, Student, Examination Group, Service Management, …
11	Consulting	Advisory body, Chain standard Organisation, Ministry of Education (OCW), Standard advisor, Pedagogue/psychologist, (Internal) Consultancy, …
12	External partners	Third parties, National educational support Organisations, local municipality, MBO support Organisations, Relationships companies authorities, Sectoral educational support Organisations, Internship places, …
13	Supporting business operations	Facilities Department, Finance Department, Human Resources (HR) Department, Purchasing Contract Management Department, Legal Affairs Department, Real Estate Department, … .
14	IT-infrastructure operations (realisation maintenance)	ICT department, Architect, Maintenance Operator, Technical management, …
15	Inbound and outbound education institutions	The inbound institution (Secondary schools), Outbound institutions (Higher education), …
16	Facilitate, guidance and care	Facilities Department, Real Estate Department, Education Team, Student Teacher guidance group, …
17	Funders	DUO, Finance department, local municipality, Ministry of Education (OCW), …
18	Recruitment of Students Teachers	Prospective students, DUO, Teacher, Education team, Procurement (Recruitment of students and teachers), …
19	Marketing	Marketing department, PR communication department, Strategy policy department, …
20	Other relations	Alumni, Advocacy groups, higher-education support Organisations, Other government services, primary education council, Association of Universities of Applied Sciences, caregiver, secondary support Organisations, secondary education council, VSNU, VVE, …

Table 14. Stakeholder groups ranked according to the questionnaire's result.