Using Ontologies for Enterprise Architecture and COBIT 5 analysis

Ana Margarida Fernandes Silva
Instituto Superior Técnico
Universidade de Lisboa
Av. Rovisco Pais, 1049-001 Lisboa, Portugal
Email: ana.f.silva@tecnico.ulisboa.pt

***Abstract*—COBIT 5 recommends the Enterprise Architecture (EA) practice as a support to guide the creation and maintenance of its governance and management enablers. These enablers are a set of components that, individually or collectively, influence the success of governance and management over enterprise IT. In particular, process capability assessment is acknowledged as an arguably condition for a successful ITG implementation.**

The EA practice delivers the analysis and planning support that is indispensable for effective ITG, and in which models play a central role to fulfill such activities. EA model analysis supports the assessment, optimization and adaptation of organizational systems. Hence, EA model analysis can be used to support COBIT 5 process assessments.

The analysis of an EA model to support COBIT 5 process assessment must be able to answer questions regarding such activity. However, typical EA modeling languages, such as Archimate, are built to address general EA concerns. Hence, the Archimate language by itself does not suffice to address COBIT 5 assessment analysis needs.

In order to overcome this issue we propose an ontology-based approach to extend the Archimate's meta-model with the COBIT 5 concepts and properties required to address the earlier referred analysis needs, and to automatically analyze models resulting from the instantiation of the resulting ontology. Such proposal was demonstrated by applying the solution artifact to the Archisurance case study. This research work was guided by DSR principles and guidelines, and according to which we have evaluated the proposed solution artifact's efficacy and utility.

***Index Terms*—COBIT 5 PRM, COBIT 5 PAM, Archimate, Ontology, OWL, SPARQL**

I. Introduction

Enterprises increasingly recognize the practical relevance of good-practice frameworks, such as COBIT 5, as means to guide the implementation of IT governance (ITG), maximizing the value of IT investments while reducing its inherent risk [1], [2], [3]. COBIT 5 advocates that efficient and effective ITG initiatives require a holistic approach [4], [5], taking into account a set of interacting components, denominated as "Enablers" that influence the success of governance and management over enterprise IT [1], [4]. Among them is a set of processes described in a Process Reference Model (PRM), accompanied with an assessment model described in Process Assessment Model (PAM) [6], [7]. Assessing processes capability is arguably a necessary condition for the successful implementation of ITG [1], [4], [6], [7], [5].

COBIT 5 recommends the Enterprise Architecture (EA) practice as a support to guide the creation and maintenance of its enablers [5]. Well defined architectures provide an information basis that is crucial for an effective ITG implementation, and act as a complement to the COBIT 5 framework in the context of an ITG initiative [2], [3]. EA model analysis supports the assessment, optimization and adaptation of organizational systems [2], [3]. Therefore, EA model analysis can be used to support COBIT 5 process assessments, as it provides the required information to perform such activity.

A model is built to fulfill a purpose; when building a model, its imperative characteristic is to be able to answer a set of questions that address its stakeholders concerns [2], [8]. For instance, if an EA model serves the purpose of supporting the evidence collection for COBIT 5 process assessment, such a model must be able to answer questions regarding this activity.

However, typical EA modeling languages, such as Archimate, are built to address general EA purposes [2]. Concepts pertaining to the specific COBIT 5 domain are not be found in a domain-independent language such as Archimate. As such the Archimate language meta-model does not suffice to address COBIT 5 process assessment analysis needs.

Therefore we propose *an ontology based approach for the representation and integration of Archimate's meta-model and COBIT 5 concepts and properties, and for the automatic analysis of the resulting ontology, as means to systematically answer a set of questions regarding the evidence collection for COBIT 5 process capability level 1 assessments*.

This research followed the DSRM guidelines and principles. Each of the DSRM steps is mapped to the document sections; section II presents a literature review for this research scope, section III presents the design and development of our solution artifact, sections IV and V demonstrate and evaluate the utility of the solution artifact in practical use, and finally section VI concludes the research and presents possible future work.

II. Background

A. Ontologies

The term ontology first appeared in philosophy, as an attempt to classify things in the world, described as a systematic account of Existence [9], [10]. Computer Science researchers adopted the term ontology to refer to a computational artifact that formally models the relevant entities and relations of some system of interest [11]. Studer et al. , based on the merging of previous definitions, stated that an ontology, in computer science, is "a formal, explicit specification of a shared conceptualization" [10].

A **"conceptualization"** refers to an abstract world view, with respect to a given domain, conceived as a set of concepts, their definitions and the relationships among them [9], [12]. **"Explicit"** means that the concepts used and the constraints on their relationships should be explicitly defined [13]. **"Formal"** implies that the ontology should be represented in a machine-readable format. And at last, **"shared"** means that the ontology should result from an agreement, capturing consensual knowledge, and be accepted by a group [13].

An ontology can be built from scratch, or it can be constructed by means of existing ontologies integration. *Ontology Integration* has had different meanings in the ontology field. Shortly, it consists of identifying common concepts and relations between two or more ontologies [14]. This process can be subdivided into three techniques [15]:

- **Ontology Mapping:** Consists of a process to build a new ontology by describing the semantic relationship between two (or more) concepts from two (or more) ontologies.
- **Ontology Alignment:** Consists of a process to build a new ontology by describing the correspondences between all the equivalent concepts of two ontologies.
- **Ontology Merging:** Consists of a process to build a new ontology by describing the semantic relationship between concepts of several ontologies, and merging them into a single one.

The term "Reasoning" refers to any mechanism/procedure for making explicit a set of facts that are implicit in an ontology. According to Lenzerini [16], there are two main purposes for one to reason over an ontology, namely Validation and Analysis. Performing Validation of an ontology ensures that the ontology is a good representation of the domain of discourse that one is aiming at modeling. By reasoning over an ontology, it is possible to verify if the axioms defined on it are actually being fulfilled. Analysis consists of deducing facts about the domain by reasoning over the ontology, assuming that its representation is a faithful one. It allows, for example, the discovery of new relations between individuals of the ontology [16]. The Web Ontology Language (OWL) [17] provides an efficient representation of ontologies, supporting various types of inference, according to a subset of labguage profiles that differ in the provided expressive power. An OWL ontology can be represented using the Resource Description Framework (RDF). RDF is data model to describe resources, in the form of triples. The SPARQL language is the W3C standard for accessing and querying the RDF graphs. It was designed to query data conforming to the RDF data model.

B. Archimate and Ontologies

Work towards the ontological representation of the Archimate meta-model has already been performed [18]. The use of ontologies was explored has means to provide a formal computable representation for EA models, in this particular case for Archimate models, enabling the analysis of their consistency and completeness against the represented meta-model. Such representation was performed by extracting all the concepts and relations pertaining to the Archimate language meta-model, and transforming them into an OWL classes and properties . Also, axioms were added to reinforce the meta-model restrictions imposed by the language.

It is also acknowledged by other authors that in order to address domain-specific analysis needs for stakeholders, the Archimate language by itself does not suffice [19], [20], [21], [22], [23]. To overcome this issue, Antunes et al. explored the federation of multiple models or domain-specific languages [21], [20], [19]. Such federation was proposed by means of ontology integration, more precisely by means of ontology mappings.

A federated schema can have multiple configurations, for instance an upper ontology (UO) can be specified containing the core concepts and relations and being domain-independent in the context of EA, and domain-specific ontologies (DSO), containing the concepts and relations pertaining to a specific domain, can mapped to such ontology via ontology mappings. Or a federated schema can have multiple domain-specific ontologies mapped to each other; either ways the ontology mappings must be traceable to the stakeholder concerns [21], [22], [21].

C. COBIT 5 and Ontologies

Textor and Geihs [24] explored an ontology-based approach for the calculation of metrics, contained in COBIT PRM, as means to attain a better informed IT management, which in turn enables a better IT governance. As a result from this research work, an OWL representation of COBIT 5 ontology was provided. This ontology contains a set of concepts and relations that belong to an earlier specified COBIT 5 meta-model. Such meta-model was specified based on already existing works towards its definition. Namely, Goeken and Alter [25] and Souza Neto and Ferreira Neto [26] who have specified an ontological meta-model for COBIT 4.1, and Moeller et al. [27] who has specified an ontological meta-model for COBIT 5 PRM.

D. Archimate and COBIT 5

Cadete has proposed to integrate the COBIT 5 process rationale in EA representations, using the standard Achimate extensions in order to promote easy adoption [28]. Such work relies on an ontological mapping between COBIT 5 process performance assessment concepts and the ArchiMate concepts the author found suitable, based on COBIT 5 documentation, Archimate's standard and the purpose the mappings must fulfill [28]. These mappings were performed using concepts from both Archimate's Business Layer and the Motivation extension, although with a major focus on the latter.

Also Almeida et al. has mapped COBIT 5 to Archimate concepts, in this particular case as means to map, model and integrate both COBIT 5 and COSO frameworks, providing a visual representation to help organizations better understand, implement and assess these frameworks simultaneously [29].

Almeida et al. proposed an ontological mapping between COBIT 5 and Archimate meta-models, with a special focus on COBIT 5 PRM and using Archimate concepts also from both Business layer and the Motivation aspect. These mappings were based on Cadete work, and based on the author's interpretation of COBIT 5 documentation and Archimate's specification [29]. Differences between Almeida et al. and Cadete's work rely on the Archimate concepts mapped to COBIT 5 concepts such as *Practice* and *Work Product*. Almeida et al. has mapped these performance indicators to concepts pertaining to the Archimate's Business Layer, while Cadete has mapped them to concepts pertaining to the Archimate's Motivation extension.

III. Research Proposal

Based on the literature review, and on the acknowledged problem, we established our solution objective as:

To provide an ontology-based systematic approach capable of answering questions that support evidence collection for COBIT 5 process capability level 1 assessment, based on the automatic analysis of Archimate models enhanced with COBIT 5 properties.

The evidence collection for COBIT 5 process capability level 1 can be subdivided into three possible approaches: (1) *search for evidence that the intent of the process base practices is being performed*, (2) *verify if process work products are produced that provide evidence of process outcomes*, and additionally (3) *search for evidence that process goals are achieved based on metrics*. Following this, and as we aim to use EA models conformity analysis as means to answer questions to support such approaches, we now explain how such analysis shall be performed.

In order to fulfill such approaches we propose:

a) For (1) and (2), the *search for architectural elements, represented on the organization's Archimate model, that realize COBIT 5 practices and work products*.

b) For (3), the *search for non-compliant architectural elements, represented on the organization's Archimate model, according to a set of COBIT 5 metrics*.

Based on the established solution objective and on the above described approaches, the following requirements are raised:

- **R1** - The solution must represent Archimate, COBIT 5 PRM and Metrics ontologies using the OWL language.
- **R2** - The solution must provide means to integrate the Archimate ontology with COBIT 5 PRM and Metrics ontologies.
- **R3** - The solution must provide means to automatically analyze the integrated ontology based on SPARQL queries, providing answers to questions that support evidence collection for COBIT 5 process capability level 1 assessment.

In order to attain the defined solution objective and requirements, we propose:

An approach based on the application of a set of ontology-based techniques to represent and integrate Archimate's meta-model with COBIT 5 concepts and properties, and also to automatically analyze the resulting integrated ontology.

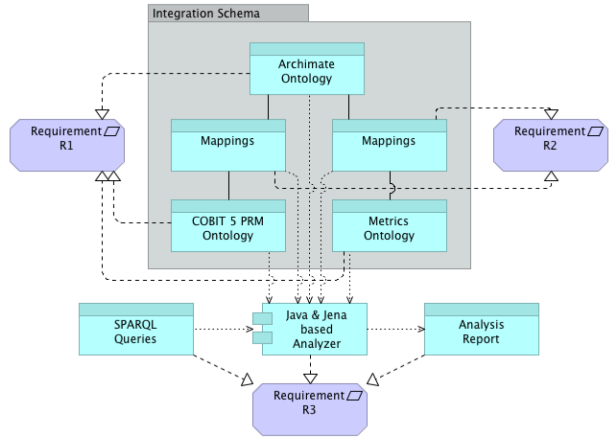

Fig. 1. An overview of the proposed solution.

The figure 1 gives an overview of our proposed approach. In the next three sections we explain how we have realized each of the above defined solution requirements, namely how the ontologies were represented (section III-A), how the ontologies were integrated (section III-B), and how SPARQL queries were used to analyze the integrated ontologies (section III-C).

A. Ontology Representation

The steps towards the representation of ontologies comprised: a careful analysis of the domains and the extraction of its main concepts and relations, followed by its representation in a meta-model; a transformation of its concepts and relations to OWL classes and properties; and the definition of a set of axioms restricting the ontology elements was also performed. The next two sections describe these steps applied to the representation of COBIT 5 PRM and Metrics, respectively.

1) COBIT 5 PRM: Following the steps described previously, based on COBIT 5 documentation [4], [6], [7], and based on previous work towards the specification of COBIT PRM meta-model [25], [26], [27], [24], we extracted the main concepts and relations which are summarized in table I and represented on the UML class diagram depicted in figure 2.

Having the meta-model specified, its concepts and relations were transformed into OWL classes and properties. We used the standard language OWL 2, profile DL, in order to perform such activities. Regarding the depicted meta-model diagram, classes contained in it representing COBIT 5 PRM concepts were transformed into OWL 2 *classes*. Relations between such concepts were transformed into *ObjectProperties*, and each diagram class attribute was transformed to a *DataTypeProperty* whose literal type corresponds to the respective one contemplated on the diagram. Such transformations were performed using the framework *Protégé*.

2) COBIT 5 Metrics: Each metric measures the achievement of a goal, in particular the achievement of a process goal.

TABLE I
COBIT 5 PRM CONCEPTS AND DEFINITIONS.

Process	Generally, a collection of practices influenced by the enterprise's policies and procedures that takes inputs from a number of sources (including other processes), manipulates the inputs and produces outputs.
Practice	High level requirements for effective and practical management and governance of enterprise IT. The tasks and activities needed to accomplish the process purpose and fulfill the process outcomes. Each BP is explicitly associated to a process outcome.
Activity	The main actions to operate the process . They are defined as 'guidance to achieve management practices for successful governance and management of enterprise IT'.
Process Purpose	A description of the overall purpose of the process
Process Goal	A statement describing the desired outcome of a process. An outcome can be an artifact, a significant change of a state or a significant capability improvement of other processes
IT Related Goal	A statement describing a desired outcome of enterprise IT in support of enterprise goals. An outcome can be an artifact, a significant change of a state or a significant capability improvement.
Enterprise Goal	The translation of the enterprise's mission from a statement of intention into performance targets and results.
Stakeholder Need	Stakeholder needs are influenced by a number of drivers, and can be related to a set of generic enterprise goals.
Metric	Metrics can be defined as 'a quantifiable entity that allows the measurement of the achievement of a goal. Metrics should be SMART— specific, measurable, actionable, relevant and timely'.
Work Product	Artifacts considered necessary to support operation of the process. They enable key decisions, provide a record and audit trail of process activities, and enable follow-up in the event of an incident. They are defined at the governance/management practice level.
Stakeholder	Anyone who has a responsibility for, an expectation from or some other interest in the enterprise. Stakeholders and their responsibility levels are documented in RACI charts.

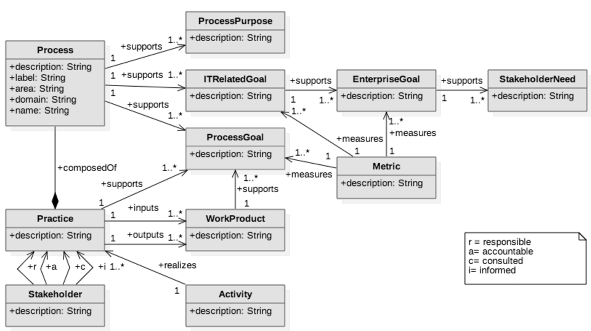

Fig. 2. COBIT 5 PRM meta-model UML class diagram.

In order to verify such achievement we propose to decompose each metric into a set of properties represented as attributes of an organization's Archimate model's elements. Following this, a metric can be a single property or a combination of multiple properties.

COBIT 5 provides a very large set of example metrics, from which we have only chosen a subset containing **ten**. These metrics belong to processes ***DSS02 - Manage Service Requests and Incidents*** and ***DSS05 - Manage Security Services***. The criteria to choose these metrics was to extend as much different Archimate concepts from the three different layers as possible, and to choose at least one by process goal. Therefore the chosen metrics were:

1) Mean elapsed time for handling each type of service request.
2) Level of user satisfaction with service request fulfillment.
3) Number and percent of incidents causing disruption to business-critical processes.
4) Percent of incidents resolved within an agreed-on/acceptable period of time.
5) Number of vulnerabilities discovered.
6) Number of Firewall breaches.
7) Number of incidents involving endpoint devices.
8) Number of Accounts vs Number of Authorized Users.
9) Number of physical security-related incidents.
10) Number of incidents relating to unauthorized access to information.

Having this set, based on the analysis of each of the metrics contained in the chosen subset, we decomposed each metric into a set of possible properties. The majority of the metrics is represented using only a property, but there are some that result from the combination of multiple properties. For example for metric *"3 - Number and percent of incidents causing disruption to business-critical processes"*, the notion of a ***critical*** process as well as the possible types of ***incidents registered by the organization that cause disruption*** were specified as properties to represent such metric.

Each property was represented using the OWL 2 *DataType-Property* primitive, and range axioms were also added so that the type of each property was restricted. Most of the

TABLE II
COBIT 5 DSS02 METRICS AND PROPERTIES.

Metric	Property
Number and percent of incidents causing disruption to business-critical processes.	*critical* *noOfIncidentsUAccessInfoDisr* *noOfIncidentsPhysicalSecDisr* *noOfIncidentsEDevicesDisr*
Percent of incidents resolved within an agreed-on/acceptable period of time.	*noOfIncidentsResolvedUAccessInfo* *noOfIncidentsResolvedPhysicalSec* *noOfIncidentsResolvedEDevices*
Level of user satisfaction with service request fulfillment.	*levelOfSatisfactionWithSR*
Mean elapsed time for handling each type of service request.	*meanTimeSR*

represented properties have a range of the literal type *int*, but for example *"critical"* property has a range of the literal type *boolean*, as it is supposed to tag if a business process is critical or not. Tables II and III show for process DSS02 and DSS05 each metric and respective properties, respectively .

TABLE III
COBIT 5 DSS05 METRICS AND PROPERTIES.

Metric	Property
Number of vulnerabilities discovered.	*noOfVulnerabilities*
Number of firewall breaches.	*noOfBreaches*
Number of accounts (vs. number of authorised users/staff).	*noOfAccounts* *noOfUsers*
Number of incidents involving endpoint devices.	*noOfIncidentsEDevices*
Number of physical security-related incidents.	*noOfIncidentsPhysicalSec*
Number of incidents relating to unauthorised access to information.	*noOfIncidentsUAccessInfo*

B. Ontology Integration

In order to fulfill requirement **R2**, the integration of the existing Archimate ontology with COBIT 5 PRM and Metrics ontologies was performed. In this particular case, we aim to extend the Archimate's meta-model with necessary concepts to address COBIT 5 assessment concerns. As such, we adopted a similar approach to Antunes et al.[21], in order to address such specific analysis concerns, by defining Archimate as an Upper Ontology (UO) containing general EA concepts and relations, and COBIT 5 PRM and Metrics ontologies as Domain Specific Ontologies (DSO) representing the concepts pertaining to these specific domains. Each DSO was integrated with the UO by specifying a set of ontology mappings, which were then represented using OWL 2 language primitives, in a new ontology. In the next two sections we describe how we have integrated Archimate with COBIT 5 PRM and Metrics ontologies, respectively.

1) Archimate and COBIT 5 PRM: COBIT 5 PRM, as its name suggests, is a reference model. The COBIT 5 framework's documentation provides the *"what"* should be done in order to achieve certain goals, but not the *"how"* to be done. Therefor, we find that enterprise elements contained in Archimate models are realizations of COBIT 5 performance indicators, contained in COBIT 5 PRM. Note also, that the purpose of integrating the Archimate ontology with COBIT 5 PRM ontology is to able to answer COBIT 5 evidence collection questions regarding the existence of Archimate elements realizing COBIT 5 ***Practices***, ***Work Products*** and ***Process Goals***.

The Archimate language already delivers a *realization* relationship which fits such purpose, as it models that some end is realized by some means [30]. However, as it is a relationship specific of Archimate, and its use constrained to Archimate concepts only, such a relationship does not contemplate in its range concepts pertaining to COBIT 5 domain. As such, we explored the Archimate's meta-model extension as means to reuse the *realization* relationship covering also COBIT 5 concepts. By meta-model extension we mean the specialization of concepts pertaining to the Archimate's meta-model with concepts pertaining to COBIT 5 PRM meta-model, so that the latter inherits Archimate's restrictions and *realization* mappings are possible.

In order to specify the ontology mappings at the meta-model level we analyzed both Archimate and COBIT 5 PRM meta-models, and based on existing work towards such integration, we extracted semantic relationships between both meta-models concepts represented as classes. As these mappings purpose is to be able to detect which Archimate elements realize which COBIT 5 performance indicators, we constrained the COBIT 5 PRM concepts to map to a subset containing: *Process Goal*, *Practice*, *Work Product*.

We opted to use the ontological mapping of author Cadete [28], as it better fits our analysis needs. According to its work, performance indicators are mapped as Archimate concepts from the Motivation extension, more precisely as goals and requirements, while according to Almeida et al. [29], the indicators are mapped as core and motivational concepts: business process, business object and goal. Based on the definitions provided in subsection III-A1, we find that practices and work products are requirements, rather than business processes and business objects, respectively. Following this, we have transformed the chosen ontology mappings to OWL 2 subClassOf primitives.

2) Archimate and Metrics: We now describe how we mapped the properties represented on the Metrics ontology to Archimate concepts. Note that the primary purpose of representing such properties was to be able to verify the conformity of architectural elements, so that COBIT 5 process goals fulfillment could be checked. Hence, in order to extend the Archimate's meta-model with the represented properties, we first analyzed the metrics that such properties compose along with the two processes they belong to, and the concepts pertaining to the Archimate's meta-model.

Based on such analysis we formulated the mappings between Archimate's meta-model concepts and Metrics properties, which can be found in tables IV and V, for metrics pertaining to process *DSS02* and *DSS05*, respectively. In order to represent such mappings, we have used OWL 2 domain restriction axioms, so that the Archimate's meta-model extension could be performed. Meaning that for each OWL *DataTypeProperty* representing a given Metric's property, we constrained its domain to the Archimate's meta-model concept according to aforementioned mappings.

TABLE IV
COBIT 5 DSS02 METRICS PROPERTIES MAPPINGS TO ARCHIMATE CONCEPTS.

Property	Archimate Concept	Archimate Concept Definition
critical	*Business Process*	"A behavior element that groups behavior based on an ordering of activities. It is intended to produce a defined set of products or business services."
noOfIncidentsUAccessInfoDisr noOfIncidentsPhysicalSecDisr noOfIncidentsEDevicesDisr	*Node*	"A computational resource upon which artifacts may be stored or deployed for execution."
	Device	"A hardware resource upon which artifacts may be stored or deployed for execution."
noOfIncidentsResolvedUAccessInfo noOfIncidentsResolvedPhysicalSec noOfIncidentsResolvedEDevices	*System Software*	"A software environment for specific types of components and objects that are deployed on it in the form of artifacts"
levelOfSatisfactionWithSR	*Business Actor*	"An organizational entity that is capable of performing behavior."
meanTimeSR	*Application Service*	"A service that exposes automated behavior."
	Infrastructure Service	"An externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment."

TABLE V
COBIT 5 DSS05 METRICS PROPERTIES MAPPINGS TO ARCHIMATE CONCEPTS.

Property	Archimate Concept	Archimate Concept Definition
noOfVulnerabilities	*Network*	"A communication medium between two or more devices."
noOfBreaches	*Node*	"A computational resource upon which artifacts may be stored or deployed for execution."
noOfIncidentsEDevices		
noOfIncidentsPhysicalSec	*Device*	"A hardware resource upon which artifacts may be stored or deployed for execution."
noOfIncidentsUAccessInfo	*System Software*	"A software environment for specific types of components and objects that are deployed on it in the form of artifacts."
noOfAccounts *noOfUsers*	*Application Component*	"A modular, deployable, and replaceable part of a software system that encapsulates its behavior and data and exposes these through a set of interfaces."

C. Ontology Analysis

In order to fulfill requirement **R3**, we describe a set of questions and SPARQL queries. The questions should reflect the approaches described in section III, and answers to such questions should be delivered by generating views over the earlier described integrated ontology. Hence, viewpoints to generate these views are required, and were defined by means of SPARQL queries. In the subsequent subsections we present a set of possible stakeholder questions that map to the above mentioned approaches, and the queries that provide answers to such questions.

1) Stakeholders Questions: Based on the described approaches and on the represented ontologies, we formulated a set of 23 example stakeholders questions. Note that these are not the only possible questions that can be asked. A subset of the defined questions is presented bellow:

1) Which **Network** elements exceed the acceptable **number of vulnerabilities**, and which **Process Goal** do they **realize**? Which **critical Business Process** elements have a **number of incidents causing disruption** exceeding the acceptable, and which **Process Goal** do they **realize**?
2) For a given **Process**, which **Business Process realize Practices** that **compose** it?
3) For a given **Process**, which **Business Object realize Work Products** that are **input/output** for the **Practice** that **compose** it?
4) For a given **Process** ,which **Practices** are not **realized** by a **Business Process**, and which **Process Goals** do they **support**?
5) For a given **Process**, which **Work Products** are not **realized** by a **Business Object**, and which **Process Goals** do they **support**?

2) SPARQL Queries: Each of the defined questions was translated into a SPARQL query based on the resulting integrated ontology. The SPARQL query for question 4 can be seen in 1. Along with the queries, we have also built an application written in Java and which uses the Jena framework. This application loads the ontologies using the Jena Ontology API, and runs the defined queries using the ARQ engine. The results of the queries execution are written as a report.

```
SELECT ?practice ?processGoal
WHERE {
        ?p rdf:type cobit:Practice.
        ?pg rdf:type cobit:ProcessGoal.
        ?p cobit:supports ?pg.
        ?pg cobit:description ?processGoal.
        ?p cobit:name ?practice.
        ?process rdf:type cobit:Process.
        ?p cobit:composes ?process.
        FILTER (?process = cobit:DSS02)
        MINUS {
                ?bp rdf:type uo:BusinessProcess.
                ?bp uo:realizes ?p.
                }
        }
```

Listing 1. SPARQL query for question 4

IV. Demonstration

We now demonstrate the utility of our Solution artifact applied to a particular scenario. Such demonstration was performed by executing the described set of SPARQL queries over the defined (populated) ontology resulting from the integration of Archimate, COBIT 5 PRM and Metrics ontologies. We chose to use the Archisurance case study [31] because of its relevance to the Enterprise Architecture community. Although, the Archisurance case study does not contemplate any adoption of Governance frameworks, such as COBIT 5, its authors encourage its extension by adding new views or change scenarios, as long as these are coherent with the initial case study description. As such we have extended the scenario with an Archimate view containing a set of business processess and business objects regarding the subject of incident management. We have also performed the required mappings, as well as defined arbitrary values to populate the metrics.

Based on COBIT 5 documentation [6], we represented a set of business processes meant to fulfill a set of required activities to implement incident management, as well as a set of business objects produced and used in such activities. Figure 3 depicts the resulting Archimate view. Note that, the COBIT 5 process for incident management is *DSS02 - Manage service requests and incidents*. We have only represented business processes and objects regarding the part of incident management, discarding service request management. Hence such a process is expected, according to our representation, never to achieve capability level 1. Each of the represented ontologies was populated with individuals, classified according to the classes defined in such ontologies. For instance, the Archimate metamodel ontology was populated with Archisurance elements,

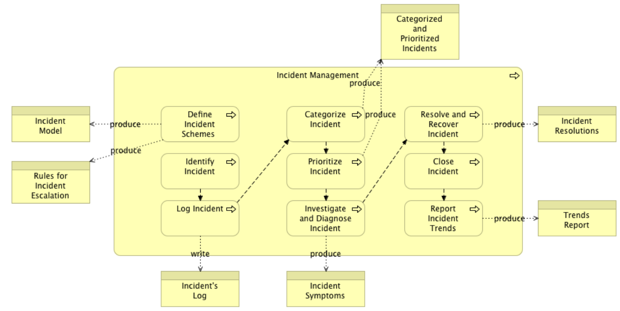

Fig. 3. Archimate view to address Incident Management concerns.

the COBIT 5 PRM meta-model ontology was populated with COBIT PRM elements, and the Metrics ontology properties were populated with arbitrary values. In order to fulfill the analysis needs, a set of instance-level mappings was also performed, namely between individuals from the Archimate and the COBIT 5 PRM populated ontologies.

Having the populated ontologies, each of the SPARQL queries was executed over the resulting populated ontology. For example, for query translating the question 1) the respective result contains a set of Network elements not compliant according to the used metric; such results can be seen in figure 4.

network	number_of_vulnerabilities	processGoal
"BIBIT"	"23"	"Networks and communications security meet business needs."

Fig. 4. Results from Query 1 execution.

The result from the execution of the query translating the question 5) can also be seen in figure 5. Such result contains a set of work products not realized at all by Archisurance elements.

workProduct	processGoal
"Criteria for problem registration"	"IT-related services are available for use."
"Approved service requests"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."
"Fulfilled service requests"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."
"Problem log"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."
"Closed service requests and incidents"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."
"User confirmation of satisfactory fulfilment or resolution"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."
"Request fulfilment status and trends report"	"Service requests are dealt with according to agreed-on service levels and to the satisfaction of users."

Fig. 5. Results from Query 5 execution.

V. Evaluation

In order to evaluate the proposed solution artifact, we have performed an analysis over the demonstration results. Along with such analysis, we have also used a subset of criteria according to Prat et al. DSR artifact evaluation [32], and we have also used the four principles of Österle et al. [33].

From the analysis of the demonstration results we conclude that:

- For queries concerned with metrics verification, the results from their execution could either be empty or contain the set of architectural elements represented in the organization's Archimate model that are not compliant according to each of the used metrics. The demonstration

results show that for a given query result if it is empty we can affirm that according to the metric checked, its respective process goal is achieved. As well as the contrary, if the result is not empty, according to the metric checked, its respective process goal is not achieved.
- For queries concerned with practices and work products evidences, the results from their execution suggest that process goals supported by the realized practices and work products are achieved, as the business processes and business objects contained in such results provide evidence that the intent of the practices is being performed and that work products are produced, respectively. Also, their results suggest that process goals supported by the non-realized practices and work products are not achieved, since there is no evidence in the organization's EA representation that business processes were implemented or that business objects were produced that realize such indicators. For results obtained from the execution of queries 4) and 5), there is a need of additional human analysis. We are able to signal that the intent of practices contained in such results is not performed, and that the work products also contained in such results are not produced. However, we are not able to affirm with certainty such fact, as according to COBIT 5 documentation practices and work products may not fit the organization's context, meaning that they may not have been adopted, instead of neglected.

In order to evaluate the Solution artifact's efficacy and utility we used the DSR artifact evaluation criteria, more precisely the criteria:

- *Goal* - ***Efficacy** :* The solution provides an integration schema, based on ontology mappings, capable of addressing the analysis concerns related to evidence collection for process capability level 1 assessment. The solution also provides, based on SPARQL queries, means to automatically answer a set of questions to support such activity.
- *Environment - Consistency with people -* ***Utility** :* The solution is useful as it provides automatic means to assist the evidence collection process that supports process assessment, providing meaningful information for assessors, retrieved from an Archimate model represented as an ontology and enhanced with COBIT 5 concepts and properties. The solution also provides a more formal approach, relying on models represented as ontologies, to support such activity.

We have also used Österle et al. principles to the ascertain the rigor of the research. According to the author, DSR in IS must comply with four basic principles [33]:

- **Abstraction :** *Each artifact must be applicable to a class of problems* - In this particular research work abstraction is justified by the possibility of applying the demonstration to any existing Archimate model and the COBIT 5 PRM, transformed into individuals of the proposed integrated ontology.
- **Originality :** *Each artifact must substantially contribute to the advancement of the body of knowledge* - As for as we know, from our literature review, there is no work previously performed towards the ontological integration and analysis of Archimate and COBIT 5 domains as means to support evidence collection for COBIT 5 assessment.
- **Justification :** *Each artifact must be justified in a comprehensible manner and must allow for its validation* - The proposed solution is grounded in previously published work, and on official COBIT 5 documentation. The solution was inferred from the established solution objective, which in turn was inferred based on the research motivation and problem. Every step of the research is described and justified throughout the present document, and it relies on the literature review.
- **Benefit :** *Each artifact must yield benefit, either immediately or in the future, for the respective stakeholder groups* - The proposed solution artifact enables a more formal approach to perform evidence collection for COBIT 5 process assessment, automatically assisting such activity, providing an assessor with relevant information.

VI. Conclusion

In this research work the solution artifact proposed to solve the earlier acknowledged problem was *"an approach based on the application of a set of ontology-based techniques to represent and integrate Archimate's meta-model with COBIT 5 concepts and properties, and also to automatically analyze the resulting integrated ontology"*. The solution artifact comprises an integration schema for Archimate's meta-model ontology and COBIT 5 PRM and Metrics ontologies, based on a set of ontology mappings, along with a set of SPARQL queries to automatically analyze the resulting integrated ontology.

The integration of the ontologies was meant to overcome the lack of COBIT 5 domain-specific concepts and properties on Archimate's meta-model, necessary to perform an effective model analysis capable of answering questions regarding the evidence collection for process assessment. The set of SPARQL queries provided viewpoints based on such concepts, capable of automatically generating views that address the specific assessment concerns. As means to support such an approach, we specified and represented COBIT 5 PRM and a subset of COBIT 5 example-metrics meta-models as OWL ontologies.

Following this, a Java program was written, using the Jena framework, which receives as input a set of ontologies already populated and mapped, and produces a set of answers based on the execution of queries over the input ontologies. The utility of the solution artifact was demonstrated through the Archisurance case study, and evaluated according to DSRM artifact evaluation and Österle et al. principles.

The main contribution of these research work was the proposed ontology-based approach capable of answering questions regarding the evidence collection for COBIT 5 process assessment. To support such approach we have represented an ontology for COBIT PRM meta-model, an ontology for a subset of COBIT 5 example metrics, along with a set of ontology mappings and SPARQL queries.

As a future work it would be interesting to explore automatic tools for mapping discovery, in order to overcome the issue of mappings manual transformation. For example, automatic tools based on the comparison of names or descriptions. Noy presents possible approaches to reduce the bottleneck introduced by such activity, based on machine-learning or graph-comparison among others [34].

Finally, it would be of great value to demonstrate the solution artifact utility and efficacy in a real organization, as in this research we have used a fictitious case study, hence having a controlled environment. The validation of the solution artifact with experts from the COBIT 5 and EA communities through interviews would be also an added value to our research.

REFERENCES

[1] S. De Haes, W. Van Grembergen, and R. S. Debreceny, "Cobit 5 and enterprise governance of information technology: Building blocks and research opportunities," *Journal of Information Systems*, vol. 27, no. 1, pp. 307–324, 2013.

[2] M. Lankhorst, *Enterprise Architecture at Work: Modelling, Communication and Analysis (The Enterprise Engineering Series)*, 2nd ed. Springer, 2009.

[3] K. D. Niemann, *From Enterprise Architecture to IT Governance*. Vieweg, 2006.

[4] ISACA, *COBIT 5: A Business Framework for The Governance and Management of Enterprise IT*. ISACA, 2012.

[5] ——, *COBIT 5: Implementation*. ISACA, 2012.

[6] ——, *COBIT 5: Enabling Processes*. ISACA, 2012.

[7] ——, *COBIT 5 Process Assessment Model (PAM): Using COBIT 5*. ISACA, 2013.

[8] P. Johnson, M. Ekstedt, E. Silva, and L. Plazaola, "Using enterprise architecture for cio decision-making: On the importance of theory," in *Proceedings of the Second Annual Conference on Systems Engineering Research, 2004*, 2004.

[9] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge Acquisition - Special issue: Current issues in knowledge modeling*, vol. 5, no. 2, pp. 199–220, Jun. 1993. [Online]. Available: http://dx.doi.org/10.1006/knac.1993.1008

[10] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data Knowl. Eng.*, vol. 25, no. 1-2, pp. 161–197, Mar. 1998. [Online]. Available: http://dx.doi.org/10.1016/S0169-023X(97)00056-6

[11] N. Guarino, D. Oberle, and S. Staab, *What Is an Ontology?* Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–17. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-92673-3_0

[12] M. Uschold and M. Gruninger, "Ontologies: principles, methods and applications," *The Knowledge Engineering Review*, vol. 11, pp. 93–136, 6 1996. [Online]. Available: http://journals.cambridge.org/article_S0269888900007797

[13] O. Corcho, M. Fernández-López, and A. Gómez-Pérez, "Ontological engineering: Principles, methods, tools and languages," in *Ontologies for Software Engineering and Software Technology*. Springer Berlin Heidelberg, 2006, pp. 1–48.

[14] H. S. Pinto, A. Gómez-Pérez, and J. P. Martins, "Some issues on ontology integration," in *16th International Joint Conference on Artificial Intelligence (IJCAI'99) Workshop: KRR5: Ontologies and Problem-Solving Methods: Lesson Learned and Future Trends*, vol. 18. IJCAI and the Scandinavian AI Societies. CEUR Workshop Proceedings, 1999, ontology Engineering Group - OEG. [Online]. Available: http://oa.upm.es/6478/

[15] S. Amrouch and S. Mostefai, "Survey on the literature of ontology mapping, alignment and merging," in *Information Technology and e-Services (ICITeS), 2012 International Conference on*, March 2012, pp. 1–5.

[16] M. Lenzerini, D. Milano, and A. Poggi, "Ontology representation & reasoning," *Universit di Roma La Sapienza, Roma, Italy*, 2004.

[17] P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, Eds., *OWL 2 Web Ontology Language: Primer*. W3C Recommendation, Dec. 2009, available at http://www.w3.org/TR/owl2-primer/.

[18] M. Bakhshadeh, A. Morais, A. Caetano, and J. Borbinha, *Ontology Transformation of Enterprise Architecture Models*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 55–62. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-54734-8_7

[19] M. Bakhshandeh, G. Antunes, R. Mayer, J. Borbinha, and A. Caetano, "A modular ontology for the enterprise architecture domain," in *2013 17th IEEE International Enterprise Distributed Object Computing Conference Workshops*, Sept 2013, pp. 5–12.

[20] G. Antunes, M. Bakhshandeh, R. Mayer, J. L. Borbinha, and A. Caetano, "Using ontologies for enterprise architecture integration and analysis," *CSIMQ*, vol. 1, pp. 1–23, 2014. [Online]. Available: http://dx.doi.org/10.7250/csimq.2014-1.01

[21] G. Antunes, J. Barateiro, A. Caetano, and J. Borbinha, "Analysis of Federated Enterprise Architecture Models," *ECIS 2015 Completed Research Papers*, vol. 10, pp. 1–14, 2015.

[22] G. Antunes, A. Caetano, and J. Borbinha, "Enterprise architecture model analysis using description logics," in *Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), 2014 IEEE 18th International*, Sept 2014, pp. 237–244.

[23] A. Caetano, G. Antunes, J. Pombinho, M. Bakhshandeh, J. Granjo, J. Borbinha, and M. M. da Silva, "Representation and analysis of enterprise models with semantic techniques: an application to archimate, e3value and business model canvas," *Knowledge and Information Systems*, pp. 1–32, 2016. [Online]. Available: http://dx.doi.org/10.1007/s10115-016-0933-0

[24] A. Textor and K. Geihs, "Calculation of cobit metrics using a formal ontology," in *10th IFIP/IEEE International Workshop on Business-driven IT Management*, Ottawa, Canada, May 2015, pp. 1384–1390.

[25] M. Goeken and S. Alter, "Towards conceptual metamodeling of IT governance frameworks approach - use - benefits," pp. 1–10, 2009.

[26] J. A. Souza Neto and A. N. Ferreira Neto, "Metamodel of the it governance framework COBIT," *JISTEM - Journal of Information Systems and Technology Management*, vol. 10, pp. 521 – 540, 12 2013. [Online]. Available: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1807-17752013000300521&nrm=iso

[27] B. Moeller, K. Erek, F. Loeser, and R. Zarnekow, "How sustainable is cobit 5? insights from theoretical analysis and empirical survey data," 2013.

[28] G. Cadete, "Using Enterprise Architecture for COBIT 5 Process Assessment and Process Improvement," Master's thesis, Instituto Superior Técnico, 2015.

[29] R. Almeida, P. Linares Pinto, and M. Mira da Silva, "Using archimate to integrate cobit 5 and coso metamodels," in *European, Mediterranean & Middle Eastern Conference on Information Systems 2016 (EMCIS2016)*, Poland, 2016.

[30] O. G. Standard and T. O. Group, *Open Group Standard The Open Group*, 2013.

[31] H. Jonkers, I. Band, and D. Quartel, "Archisurance case study," 2012.

[32] N. Prat, I. Comyn-Wattiau, and J. Akoka, "Artifact evaluation in information systems design-science research – a holistic view," *18th Pacific Asia Conference on Information Systems, Chengdu, China.*, p. 16, 2014.

[33] H. Österle, J. Becker, U. Frank, T. Hess, D. Karagiannis, H. Krcmar, P. Loos, P. Mertens, A. Oberweis, and E. J. Sinz, "Memorandum on design-oriented information systems research," *European Journal of Information Systems*, vol. 20, no. 1, pp. 7–10, 2011. [Online]. Available: http://dx.doi.org/10.1057/ejis.2010.55

[34] N. F. Noy, *Ontology Mapping*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 573–590. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-92673-3_26