Automation of Enterprise Architecture Discovery based on Event Mining from API Gateway logs: State of the Art

Carlos Roberto Pinheiro
Universidade de Trás-os-Montes e Alto Douro
Vila Real, Portugal
carlos.guti@gmail.com

Sérgio Guerreiro
INESC-ID, Instituto Superior Técnico
Universidade de Lisboa
Lisboa, Portugal
sergio.guerreiro@tecnico.ulisboa.pt

Henrique São Mamede
INESC TEC
Universidade Aberta
Lisboa, Portugal
hsmamede@gmail.com

2021 IEEE 23rd Conference on Business Informatics (CBI) | 978-1-6654-2069-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/CBI52690.2021.10062

***Abstract*——Enterprise Architecture (EA) is defined as a coherent set of principles, methods, and models used to design an organizational structure, containing business processes, information systems (IS), IT infrastructure, and other artefacts aiming the alignment of business, IT, and other organizational dimensions with the strategic objectives of a company. One of the most critical in Enterprise Architecture Management (EAM) is creating EA models representing different viewpoints for managing various company concerns on its IT landscape. At the same time, the speed of changes pressures EAM to automate modeling activities. In this context, architects need adequate tools to discover the current state of EA, enabling analyzing improvement opportunities and support architectural decisions making in a fast and agile way with more precision about the real conditions. EA Mining is the use of data mining techniques to automate the creation or update of EA models with data collected from different data sources. This work presents an exploratory review of the literature to gather the state of art on EA mining models from applications logs pursuing to automate the architecture modeling. Through this literature review, we identified the main aspects, techniques, and challenges of EA modeling automation.**

Keywords— Enterprise Architecture Management, Enterprise Architecture, Architecture Mining, Predictive Analysis, Automatic Architecture Modeling.

I. INTRODUCTION

Enterprise Architecture (EA) is a coherent set of principles, methods, and models useful to design an organizational structure. It contains business processes, information systems (IS), and IT infrastructure that aligns business and IT initiatives with the strategic objectives of an organization [1].

Information systems are defined in [1] as the computer-based portion of a business system, involving applications and data over an IT infrastructure. The business capabilities of modern companies are highly dependent on the information systems and its underlying IT infrastructure [2]. From this perspective, it is notorious the importance of optimizing the components of the EA, planning changes, and ensuring alignment with strategic and business objectives, reducing risks and costs. At the same time, the speed we have seen in business changes implies the need to respond faster and faster to scenarios in which threats and opportunities arise at all the time. Hence companies are continuously redefining its business goals, and it requires continuously review and adapting its process. In this context, managing outdated models could lead architects to make wrong decisions with harmful effects for organizations, at the same time that manual EA modeling incurs additional time to deliver EA models, besides being prone to errors and poor understanding [3]. Consequently, we need to evolve support tools that help make architecture decision-making more agile and driven by accurate data.

EAM creates, maintains, and analyzes models of Enterprise Architecture. These models cover different concepts that reflect the perspective of the business and IT and must be maintained constantly in response to the continuous transformations of the company. As EA models grow, it became harder to maintain them since many stakeholders from many sources need to contribute with relevant information to the architecture. According to Farwick *et al.* [2], the literature about EAM and two practice surveys indicate that maintaining the EA model, particularly manual documentation activities, represents one of the most significant challenges for EAM in practice.

Based on Perez-Castillo *et al.* [4] study, we define EA mining as the usage of data mining techniques to obtain accurate current views of EA models. One of the most critical issues in EAM is creating EA models representing different viewpoints for managing different concerns of the company its IT landscape (e.g., processes, services, applications, data, and infrastructure). In an EAM context, there are at least two primary purposes to create enterprise models. First, EA models can be used to describe the baseline / current situation of the enterprise. Second, EA models can also be employed to describe the (desired) target/future condition. However, in other research, Pérez-Castillo *et al.* [3] affirm that the EA modeling still is done with a low degree of automation, and the current research approaches are focusing on automating EA documentation only based on specific data. This raises the importance of mapping the EA mining techniques, because will probably not be easy find a unique solution for any scenario.

There are scenarios where event logs record events from multiple organizations that can collaborate working together to handle process instances, or where different organizations are essentially running the same process while sharing common experience, knowledge, or infrastructure [5]. These cross-organizational process models describe processes crossing the boundaries of a single organization, spanning over two or more organizations.

An API Management tool monitors all data traffic that crosses the boundary of the company network and its systems.

978-1-6654-2069-3/21/$31.00 ©2021 IEEE
DOI 10.1109/CBI52690.2021.10062

It seems to be a great place to mining cross-organizational interactions, considering that the external systems of business partners will call these APIs to interact with the company process, and the API Gateway will mediate calls among different systems of an organization. Thus, API gateway may play an important role, at least to complement some event logs, not only at enterprise level, but also, in process mining.

Thus, the problem that we intend to investigate is how to make AS-IS architecture modeling more agile and intelligent by using API Gateway logs to support better architectural decisions at the enterprise level.

As EA covers a wide range of issues and views on these issues for different stakeholders, we chose to focus preferably on the application services and process architectures layers, with a focus on the API Gateway component and issues related to business values, architectural principles, cost reduction and processes efficiency. Other aspects are equally important. However, we chose these aspects to reduce the scope of the research. In our opinion, it will be useful for the industry application and enough to prove the general idea of extract enterprise architectural views from API Gateway logs, besides enabling it for generalizations in the future. For processes efficiency in this context, we are considering the perspective of the execution time and execution cost, which apparently is not a huge challenge to measure.

The following sections briefly describe the research method, followed by the related works found, where we describe the main concepts and summarize the main contributions to the focus of research. Then, we follow a brief discussion about the findings and insights, and finally, we present a conclusion wrapping the result of this literature review and future work.

II. METHODOLOGY

According to Kiteley & Stogdon [6], a literature review can be carried out as a research methodology with the objective of gathering what is known about a specific subject or problem in a way that has not been previously reported. Therefore, among other possible functions, a literature review can serve to consolidate the understanding, gather findings from multiple sources, map the terrain of the evidence about a given problem or highlight the most convincing proposals in the published literature so far. Thus, in this context, we present the process used in this research, which has an exploratory objective of gathering the current knowledge about techniques and challenges to automatic modeling of EA models. Although our main research focuses on API Gateway as a data source, to know the mining approaches and techniques available in the state of art, we have not limited this literature review to just API Gateway. To carry out this review, we followed the procedures recommended by Kitchenham [7], which consists of three phases as illustrated in Fig. 1.

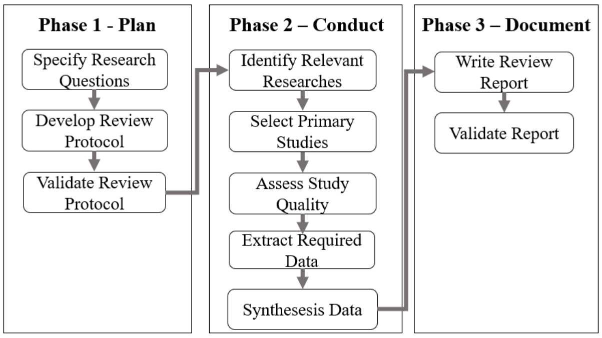


Fig. 1 Kitchenham Systematic Review Process

In the planning phase, given the context, problem, and motivations presented before as the driver for this research, we defined as the objective answer the research questions below to gather the state of art in the field.

RQ - How to capture and automatically generate an EA model by mapping business processes and applications that support these processes based on the relevant events captured from API Gateway to support real data-driven architectural decisions?

This research focus on enterprise architect's population, intent to help them through automate EA modelling based on data mining techniques and to provide a tool for simulation of TO-BE models. It probably will help to create more realistic, reliable, and cheaper models comparing to manual modeling. The research outcome is the current state of the art on automatic modeling and simulation of enterprise architecture and the context is the scenario in which the speed of business and IT change pressures EAM to respond quickly, when, automating modelling becomes a critical task.

To define the keywords and search string, we initially identified through ad hoc search on Google Scholar the most promising places to start the research. After some reading, we noticed some sources that seemed to have a good number of articles in the area. We also checked that they have good score in SCImago and CORE/ERA. At this point, we searched with focus on these sources: IEEE EDOC, IEEE EDOCW, CEUR, AIS CAIS and AIS JAIS. After selecting and read some papers from these sites, the keywords and query were revised and resulted in the final version, shown in the square below.

("Enterprise Architecture*" OR "Enterprise Architecture Management" OR "Enterprise Information System*") AND (("Architecture Mining" OR "Process Mining" OR "Event Mining" OR "Architecture Extract*" OR "Reverse Engineering" OR "Automatic Architecture Modeling") OR (Simulation OR "Impact Analysis") OR ("Deep Learning" OR "Artificial Intelligence" OR "Machine Learning" OR "Neural Network*"))

We also decided to apply the inclusion e exclusion criteria described below, for papers selection.

Inclusion Criteria:

- Main objective or paper is to discuss or investigate methods for automating architecture, process or event mining, or architecture simulation;
- Paper focuses on EA Modelling; Publish after 2015;
- Publish with peer review.

Exclusion Criteria:

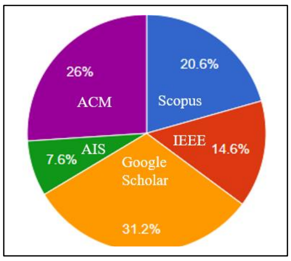
- Source incomplete, not found, or not accessible;
- Not a paper, i.e. Books and theses;
- Before 2019 without citation;
- Not written in English or Portuguese;
- Predatory suspect by Beall´s List or Stop Predatory Journal;
- Title out of the research objectives;
- Abstract, introduction and conclusion out of the research objectives.

We applied the search string to ACM Digital Library, AISeL, IEEE Explorer, Google Scholar and Scopus. The total number of papers resulting from the search is distributed as shown in TABLE I.

Particularly on Google Scholar, the search returned 996 works in total. As the results of Google Scholar was very comprehensive, we executed some additional steps in the protocol. Firstly, we eliminated all publications before 2019 without citation, then those where the origin was outside the field, such as congresses on medicine, law, logistics and others. The result was reduced to 256 papers, which removed 16 books and 22 entries that did not link to any article resulted in 218 papers that were then joined to the papers from the other bases to follow the rest of the protocol. The total of papers found for selection is shown in Papers Found per Source.

At the end of the protocol, the process resulted in 39 accepted papers, distributed according to Fig. 2. At the end of the process of reading the 36 accepted papers, added to 8 foundation works which did not come from the search in the bases [1], [6]–[10], and discard those who have not made significant contributions to the topic we obtained the 27 contributions in this paper.

TABLE I. PAPERS FOUND PER SOURCE

Base	Quantity	Distribution Percentage
Google Scholar	218	
ACM Digital Library	182	
IEEE Explorer	102	
Scopus	144	
AISeL	53	
Total	699	

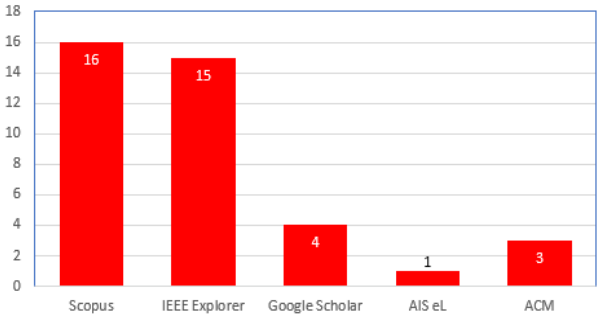


Fig. 2 Papers Accepted per Source

III. STATE OF ART

Here we present a general overview of the knowledge base available in this investigation, which compiles the main references that support the problem definition and feasible solutions. We depict the main concepts, definitions, frameworks, methods, techniques, and research results related to automatic modeling or discovery in EA models.

A. *Enterprise Architecture Management*

Enterprise Architecture (EA) usually deals with the general view and not with details. It seeks to understand and communicate how each of the business and IT components of an organization relates to the others [1]. The Zachman Framework presents a structural ontological model for describing an organization and that provides a logical framework for classifying and organizing the descriptive representations of an organization that are important for business management and for the development of enterprise systems in a graphical structure that allow for different perspectives on the construction of a complex product and its abstractions. This classification allows us to focus on selected aspects of an architectural object without losing the holistic view [10]. TOGAF defines a method of building enterprise architectures by identifying relevant building blocks in four different domains [1]:

- **Business architecture**: describes the business strategy, governance, organization, and key business processes.
- **Data architecture**: describes the structure of an organization's logical and physical data assets and its management capabilities.
- **Application architecture:** provides blueprints for applications, their interactions and their relationships with the organizations business processes.
- **Technology architecture**: describes the logical software and hardware resources needed to support the deployment of business services, data, and applications, including IT infrastructure, middleware, networks, communications, processing, and standards.

Winter & Fischer [9] extended the TOGAF domain layers, adding two more, integration architecture and process architecture. One benefit of this expanded model is clearly positioning the scope of EA concerns that touch each different point of these architectures in each EA domain layer, which in this model, details other aspects while keeps the link with EA concerns on the top of them. It describes possible interfaces between EA and specialized architectures. For this reason, we have decided to use this structure as a reference for our work.

Business architecture: The business architecture represents the fundamental organization of the corporation from a business strategy viewpoint. Product/service categories, product/service groups, and products/services should comprise EA, more detailed representations of product sub-architecture are managed by a product management tool.

Process architecture: The process architecture represents the fundamental organization of service development, creation, and distribution in the enterprise context. Detailed process descriptions, including specifications of activities and work steps, are out of EA scope and should be maintained by using specialized business process modeling tools.

Integration architecture: The integration architecture represents the fundamental organization of information system components in the enterprise context. Design and evolution principles for this layer focus on agility, cost efficiency, integration, and service speed. Aggregating dependencies and data flows between applications or application components belong to EA. Detailed interface descriptions for data exchange, remote procedure calls, etc., should be maintained using tools like integration repositories of Enterprise Application Integration (EAI) tools.

Software architecture: The software architecture represents the fundamental organization of software artifacts. Detailed descriptions of data objects are not essential for EA purposes and should be managed by using a data modeling tool. In addition, structural and behavioral aspects of single software components are not covered by EA and should be managed by using software design tools.

Technology architecture: The technology architecture represents the fundamental organization of computing components, hardware, and networks. Detail specifications of IT components are not essential for EA. Asset management tools should be used for managing such metadata, and appropriate interfaces must maintain consistency between the different repositories.

B. Modeling Automatization for Enterprise Architecture

Farwick *et al.* [2], presents a situational method for EA documentation considering the context of the enterprise documentation and the variability of data sources such as network scanners, Enterprise Service Buses (ESBs), and others. They seek to deal with the challenge of modeling and maintaining models of heterogeneous, dynamic, and complex social systems to facilitate documentation of the current state of architecture (AS-IS). Their approach consists of using multiple documentation techniques, a method to the assembly process and a metamodel that contains the metadata to perform the data import and modeling process. Furthermore, they reported some publications that reinforce the importance of monitoring relevant changes for EA by gathering events from information systems. Some challenges they described are related to when collecting data. It may be necessary to resolve possible conflicts or deal with orphan elements, deal with unstructured data, and deal with the granularity of the data that can increase the effort to maintain the model quality. Another relevant point is that despite data from more technical EA layers can be easily collected, data relevant to automation may not be available in an automated way. Therefore, to achieve the objective of modelling automatization, it will be essential to observe the quality and granularity of the data collected.

Lagerström *et al.* [11] propose a foundation for a tool of automatic designer for secure enterprise architecture modeling. That is a tool that searches for designs that better fulfil requirements considering an architecture model, a specification of stakeholder requirements, and a limited design space. To advance their proposition, they pointed as future work to develop a Doman Specific Language (DSL), Define a Markov Decision Process and implement reinforcement learning algorithms. They focus exclusively on modelling enterprise security concerns, and this is a position paper. Although we have not found their final framework or tool at this time, this research reinforces the importance of the theme under analysis.

C. Event Mining for Architecture Modelling

Perez-Castillo *et al.* [4], presents a systematic review in EA Mining processes in which they state that the research field to the automation of EA modeling is not fully mature. There is much research without sufficient empirical evidence of its feasibility and applicability in the industry. They advocate that systematic and faster ways to collect data about architecture may save costs, since companies will not have to employ so many hours of experts to discover the EA current state. They conclude that it is possible to use different sources of knowledge to model different elements of EA model. Among the recognized patterns, the most common is to apply process mining techniques on process event logs. Nevertheless, they indicate that some automatic mining techniques still require a combination of expert knowledge, mainly due to the difficulty of simply automatic extracting some EA descriptions. Furthermore, it appears to suggest as a future work the creation of a common repository of knowledge for different mining techniques to feed EA models. Although we have not found the repository, our research would contribute to this type of repository with new techniques discussion. Later Pérez - Castillo *et al.* [3] argued that the continuous adaptation of EA in a context of constant changes and high volatility is not trivial. Then, they developed a reverse engineering approach for AchiMate model based on several techniques, static analysis of source code, process mining, database schema and textual analysis.

Välja *et al.* [12], [13] developed an approach to create EA models from heterogeneous and fragmented data sources. According to them, modeling IT architecture is a complex, time-consuming, and error-prone task, but despite automating modeling is feasible, it has certain obstacles be overcome. The use of multiple sources means that heterogeneous data needs to be merged. Moreover, the same data collection might be useful for creating more than one architecture model. The availability of vast amount of data sources within organizations and mining techniques makes it possible to capture and evaluate dynamic changes in enterprise IT architecture with less effort. In this context, they proposed a framework for automatically creating holistic IT architecture models from machine-readable in heterogeneous data sources, like network scanners, system management, events, and security monitors. Then, these data are merged, and the trustworthiness of different sources is calculated. However, highly disputed data is excluded from the final model. Also, they highlight some problems related to varying accuracy of actuality, coverage, reorientation of data syntax, file format, and inconsistent semantics. Nevertheless, they indicate that these problems can be handled by using probabilistic models.

Harzenetter *et al.* [14] present a data mining approach to generate workflows for managing executables based on declarative deployment models. They note that many current application technologies support cloud deployments for distributed applications by declarative deployment models that describe the application structure, including all application components and their relationships, and deployment tasks. The proposed approach focuses on automatic workflow generation for holistic management functionalities that affect multiple components which are distributed across cloud environments but have limited support of automatization. In our view, despite the focus on deployment workflow, this paper presents an interesting idea that seems to be also applied to generate structural EA views by aggregation of these structures.

Simović *et al.* [15] bring an approach to extract business information from enterprise application logs based on a domain-specific language definition. The interesting lesson from this work is to define as a premise for the model that the records must be registered according to the IEEE eXtensible Event Stream (XES) standard. XES defines a grammar of tags and provides a unified and extensible methodology to capture systems behavior through logs and event flows [8].

1) Probabilistic and Predictive Analysis

Christ *et al.* [16] propose a reference architecture to combine Complex Event Processing (CEP) mechanisms with predictive analysis using the Conditional Event Occurrence Density Estimation (CEODE) concept that uses probabilistic laws to calculate the probability of event combinations. CEP is a technology for analyzing process event flows, detecting defined event patterns, and responding to them. However, the classical CEP is not able to consider events that have not yet happened. For this reason, the authors introduced the CEODE to CEP, pursuing to create proactive and predictive event processing rules. In their proposal, events are captured through sensors and applied to the business process management (BPM) context to anticipate and mitigate undesirable future events. As the main contribution, we see the conceptual use of the conditional density function to calculate uncertainties, forecasts, and probability patterns of events. Furthermore, the approach proposed seems to be a good candidate for adaptation to the context of web services and where sensors could be replaced by events in logs files.

Roudjane *et al.* [17] claim that log analysis has begun to include elements of data mining and machine learning to find patterns that emerge from data sets and that can provide insight into the process in progress or determine the course of future actions. The research focuses on a specific log mining and trend analysis problem by calculating trends in a sequential flow of events and detecting deviations based on a given reference trend. In the sequence of this work, Roudjane *et al.* [18] combined the concepts of event flow processing and machine learning to create a structure that allows the calculation of various types of predictions in a business process. Due to the sequential nature of their analysis of the events in these studies, their predictions are more suitable for business process analyses. For our study, the method to finding patterns and trends sounds useful for simulating and comparing trends when applying or simulating each change.

Boer *et al.* [19] sought to provide an overview of the mechanisms for IT Architectural resilience by combining infrastructure and process mining techniques for EA. They then present a resilient EA reference model with support for predictive IT maintenance, in which they addressed some mechanisms on how to protect the state of resilience at various levels of abstraction and provide the link between the selected resilience attributes that guarantee predictive maintenance. The proposal needs validation. However, they summarized an interesting set of resiliency attributes, such as flexibility, agility, adaptive capacity, disruptive events, vulnerability, and redundancy. These attributes may derive indicators that probably could be considered to simulate changes in the EA Model.

Johnson, Lagerström, *et al.* [20] present a special language for architectural prediction based on multiple attributes, the Multi-Attribute Prediction Language (MAPL). It consists of an analysis metamodel for unexpected qualities of a system of systems, which considers five assigned areas: cost of service; Service availability, data accuracy, coupling, and application size. The article demonstrates how these five areas are modeled and analyzed quantitatively based on the ArchiMate standard for enterprise architecture modeling and on the Predictive, Probabilistic Architecture Modeling Framework (P^{2}AMF) previously published by Johnson *et al.*[21]. The main contribution of this work is a language and metamodel for predictive analysis of multiple attributes of application service quality. Furthermore, P^{2}AMF can provide an interesting basis for modeling simulation and probabilistic analysis.

In another research Hacks & Lichter [22] developed an approach to deal with the collection of contradictory data. This approach is also based on the P^{2}AMF previously presented by Johnson *et al.* [21]. However, the original Johson´s model uses UML / OCL notation, while these authors applied it to an ArchiMate model. In addition, alternative scenarios in different versions are added to the model over a time series considering a distributed EA evolution. However, to estimate the probabilities of the existence of elements in architecture, their approach uses expert's opinion. Here, it seems for us that the proposal presents a limitation regarding the analysis work of architects and experts to assign as probabilities rationally. Thus, only after these configurations is it possible to obtain any architecture view. Even so, this view continues to be prone to misinterpretation and highly dependent on the knowledge of experts.

Johnson, Ekstedt, *et al.* [23] have seen the task of automatically create a current and future EA model as a probabilistic state estimation problem, which they propose address using Dynamic Bayesian Networks (DBN) based on techniques of machine learning to the EA model state estimation. They collect data from many sensors, such as network sniffers, directory services, and others. Accord to them, the DBN is useful for dealing with irregular data, where there may be strong evidence of some aspects but scarce of others. The DBN can merge the available data into a coherent and probabilistic model that represents the world as it is known, with a degree of confidence attributed to various parts of the current and future EA model. Furthermore, DBN can capture causal dependencies by conditioning the probability of a phenomenon in another. An important point is related to the challenge to filter the data to reduce noise and avoid irrelevant information.

Going further in the literature, it is also possible to find the many deep learning usages to predict architectural aspects, mostly using recurrent neural networks (RNNs). Most of them are applied to processing input sequences one step at a time and maintaining a state throughout to predict the process behavior and time to finish [24]–[27]. Their most significant contribution here is the conceptual idea of applying deep learning technology to obtain behavioral and structural viewpoints of the architecture.

D. Literature Analysis

To provide a consolidated view of the literature presented so far, the TABLE II groups and summarizes the works and their contributions to the research subject under analysis.

TABLE II. MAIN CONTRIBUTION OF PRIMARY STUDIES

Topic	Work	Contribution
EA modeling automatization These works focus on automatically generate EA models from different kind of data sources, like network scanners, events logs, security monitors, and others	[2]	Developed an approach using multiple documentation techniques, a method to the assembly process and a metamodel that contains the metadata to perform the data import and modeling process
	[3], [4]	Developed a reverse engineering for AchiMate based on static analysis of source code, process mining, database schema and textual analysis.
	[11]	Propose automatic modeling based on Markov Decision Process.
	[12], [13]	Developed an approach to create EA models from heterogeneous and fragmented data sources like network scanners, system management, events, and security monitors. Aftter merge data, thei calculate the trustworthiness of different sources.
Enterprise Process Mining These works focus specifically on modelling enterprise process based on events collected from logs and sensors.	[14]	Present a data mining approach to generate workflows based on declarative deployment models of cloud distributed applications.
	[15]	Extract business information from enterprise application logs based on a domain-specific language definition in accord to the IEEE eXtensible Event Stream (XES) standard.
Predictive Analysis These works also collect data from logs, sensors, and other data sources, but they focus on predict the future events, trends, and impacts on architecture	[16]	These works also collect data from logs, sensors, and other data sources, but they focus on predict the future events, trends, and impacts on architecture.
	[17], [18]	Mining and trend analysis by calculating trends in a sequential flow of events and detecting deviations based on a given reference trend. They combined event flow processing and machine learning to create a structure to calculate predictions in a business process.
	[19]	They provide an overview of the mechanisms combining infrastructure and process mining techniques for EA that supports predictive IT maintenance. Based on several attibutes which are candidates for monitoring.
	[20]	Present a language for architectural prediction based on multiple attributes (MAPL).
	[22]	Developed an approach to deal with the collection of contradictory data based on the P^2AMF.
	[23]	Create a current and future EA model as a probabilistic state estimation using a Dynamic Bayesian Networks (DBN) and machine learning techniques to the EA model state estimation.

We noticed here a good amount of works that provide important bases for the automation of modeling in the scope of EAM. They also present several promising techniques for collecting and analyzing events recorded in logs files and other data sources capable of providing relevant information for predictive analyses based on statistical models and machine learning. From another perspective, when classifying the relationship between the focus of the works and the architectural layers proposed by Winter & Fischer [9], we observed the predominance of applications of these techniques in the business processes architecture, as shown in TABLE III.

TABLE III. MAIN FOCUS OF RESEARCHES IN RELATION TO ARCHITECTURAL LAYERS

Architectural Layer	Works	total
Process Architecture	[14]–[16], [18], [24]	10
Integration Architecture	[3], [20], [22]	3
Technology Architecture	[11], [19]	2
Transversal to Layers	[2], [13], [23]	3

IV. RESEARCH DISCUSSION

We observed that logs recorded by enterprise applications may serve as a powerful source of knowledge and may contribute to raising the level of automation of EA modeling. It can provide information related to dynamic information to EA models, and combined with mining techniques, may promote these logs as an enabler for agile EA modeling [4]. Moreover, we identified different techniques to predict results of enterprise architectural concerns, which may reduce many manuals and time-consuming tasks in EAM modeling and lead to a better return on investment in EAM, as well as supporting better decision making for future architecture when providing an accurate current condition view.

It is clear that manual modeling is an error-prone and time-consuming activity that requires specialized skills [20], [23]. This research revealed that several investigations had been carried out aiming to automate the modeling of EA. These works identified some issues and challenges that need to be overcome to extract data and automatically build a reliable EA model. It is still hard to guarantee data and model accuracy and solve conflicts, deal with missing elements, false negatives, false positives, and correlate independent events.

EAM provides decision support based on organization-wide models. However, creating these models is complex due to the multiple aspects of an organization that need to be considered, making manual efforts time-consuming and error prone. Although, we have seen some studies seeking to address issues related to automatization of architecture modeling by data mining and predictive analyses. The link among these studies sounds open opportunities for improvements, especially to describe how effectively use an API gateway as a data source for this context. In this literature review, we have not found any work that explicitly addresses the gathering and usage of API Gateway data for mining and discovery of enterprise architectures models. Moreover, some proposals were designed for specific domains, but have opportunities to be generalized to other domains, e.g. the proposed DSL by Lagerström *et al.* [11] which was focused on security, but probably may be adapted and tested to business process.

Some research report challenges in dealing with heterogeneity, fragment, and truthfulness of data that may be addressed by new research. From the API management perspective, based on our empirical experience, we observe additional challenges to correlating different API calls, which is not trivial, as each call is completely different from the others. Usually, they are open for consumers to implement the processes as their need. Therefore, they do not have an explicit calls sequence. Moreover, the entire process by the consumer application before and after the call, including all backend processing between its services call and return are unknown by the API Gateway. For these reasons, it sounds not be adequate for fine-grained process details. However, it may contribute to some specific architectural views at the enterprise level.

We can consider that the traceable points in an API Gateway go from the moment when the consumer application of the service made request until the moment that the API Gateway calls the interface of the service provider (backend API), and when the backend service returns the response until the delivery of the API Gateway response to the consumer application.

We see automatic architecture modeling as the capability to generate architectural models directly from data and events captured from different informalized data sources. However, we need to agree with Johnson, Ekstedt, *et al.* [23] when they argue that automatic modeling is not an all-or-nothing proposition. This view is corroborated by a recent literature review carried out by Pérez-Castillo *et al.* [4], which reinforces that automatic modeling techniques to extract EA descriptions still require combining expert knowledge. We note that a totally automatized process should be hard to achieve due to the complexity of EA, the diversity in enterprise objectives, and stakeholder's needs. Thus, semi-automated modeling and partially architectural views still can bring significant value, while still, there are challenges to be solved. The TABLE IV lists the main challenges and attention points related to data and event mining.

Lastly, we may note that the weakness of part of publications in the field is the absence of strong empirical evidence and validation supported by study cases. This aspect may indicate attractive opportunities for research to evolve some aspects of these studies and validate their theories.

TABLE IV. ARCHITECTURE MINING CHALLENGES

Challenges in Data Mining in General	Deal with data duplication and conflicts
	Deal with orphan events or elements
	Deal with unstructured data
	Deal with different granularities and levels of abstraction of data
	Deal with quality and the trustworthiness of data
	Deal with high volatility and constant changes in data structures
	Deal with data heterogeneity and inconsistent semantics
	Deal with noise and irrelevant data
Especific Challenges in Data Mining from API Gateway logs	API call is completely independent and different from the others
	APIs calls do not have a explicit sequence
	API calls are just partially observable related to the complete processing of services

V. Conclusions and Future Work

This work presented the state of art in automating the modeling of EA. This field of investigation involves practically all layers of the EA, which deals with vast subjects in terms of modeling. In specific, we are narrowing in the context of events produced and consumed by software applications corresponding to the ArchiMate application layer. An API Management tool monitors all traffic across the boundary of the company and its systems. IT will mediate service calls among different systems of an organization. Thus, it is a great place to mining cross-organizational interactions. However, it seems to be few explored for this context.

This literature review revealed some methods for the discovery and automatic modeling of architectures based on data collection from different types of logs and data sources. Despite these methods, no specific works were found for the use of API Gateway. Furthermore, several techniques for predictive analytics were raised that can be applied to the model discovered through an API Gateway log. From these references it seems to be feasible to develop a new approach and at the same time point out some challenges that must be resolved. Especially regarding the diversity of information that may be present in the log, the need to deal with data quality, inconsistency, duplicity, and especially in the cross-organizational processes discovery, relating events whose process is not explicit.

One weakness of this work is its scope, since it had the purpose of obtaining a more global view regarding mining techniques to extract models in enterprise architecture models, and not only based on API Gateway, neither on log files. Future work should deepen in the consolidation of a theoretical framework based on the balance of benefits of each model or technique identified in this research to apply specifically to mining from API Gateway logs. Nonetheless, for the purpose of this research, we understand that this work achieved the objective of providing an overview of the current state of enterprise architecture mining and capturing insight for future developments.

It was possible to observe from these recent studies that the field is open for investigation and present some challenges related to the difficulty in deal with multiple and heterogeneous data source in architecture mining. Validating consistently scientific proposals is notoriously important. However, the Perez-Castillo *et al.* [4] review suggests a lack of scientifically robust validations. This point must be considered in any future development, while alert to finding ways to test the theories presented before any adoption.

Acknowledgment

This work was supported by the European Commission program H2020 under the grant agreement 822404 (project QualiChain) and by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UIDB/50021/2020 (INESC-ID).

References

[1] The Open Group, *The TOGAF® Standard, Version 9.2*. 2018. [Em linha]. Disponível em: https://pubs.opengroup.org/architecture/togaf9-doc/arch/index.html

[2] M. Farwick, C. M. Schweda, R. Breu, e I. Hanschke, «A situational method for semi-automated Enterprise Architecture Documentation», *Softw. Syst. Model.*, vol. 15, n. 2, pp. 397–426, Mai. 2016, doi: 10.1007/s10270-014-0407-3.

[3] R. Pérez-Castillo, D. Caivano, F. Ruiz, e M. Piattini, «ArchiRev—Reverse engineering of information systems toward ArchiMate models. An industrial case study», *J. Softw. Evol. Process*, vol. 33, n. 2, p. e2314, 2021, doi: https://doi.org/10.1002/smr.2314.

[4] R. Perez-Castillo, F. Ruiz-Gonzalez, M. Genero, e M. Piattini, «A systematic mapping study on enterprise architecture mining», *Enterp. Inf. Syst.*, vol. 13, n. 5, pp. 675–718, Mai. 2019, doi: 10.1080/17517575.2019.1590859.

[5] W. van der Aalst *et al.*, «Process Mining Manifesto», em *Business Process Management Workshops*, Berlin, Heidelberg, 2012, pp. 169–194. doi: 10.1007/978-3-642-28108-2_19.

[6] R. Kiteley e C. Stogdon, *Literature Reviews in Social Work*. 1 Oliver's Yard, 55 City Road, London EC1Y 1SP United Kingdom: SAGE Publications Ltd, 2014. doi: 10.4135/9781473957756.

[7] B. Kitchenham, «Procedures for Performing Systematic Reviews», vol. 33, pp. 1–26, 2004.

[8] IEEE, «IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams». Nov. 2016.

[9] R. Winter e R. Fischer, «Essential Layers, Artifacts, and Dependencies of Enterprise Architecture», em *2006 10th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW'06)*, Out. 2006, pp. 30–30. doi: 10.1109/EDOCW.2006.33.

[10] J. Zachman, «The concise definition of the zachman framework by: John A. Zachman», *Zachman International Enterprise Architecture*, 2008. https://www.zachman.com/16-zachman/the-zachman-framework/35-the-concise-definition (acedido Abr. 28, 2021).

[11] R. Lagerström, P. Johnson, e M. Ekstedt, «Automatic Design of Secure Enterprise Architecture: Work in Progress Paper», em *2017 IEEE 21st International Enterprise Distributed Object Computing Workshop (EDOCW)*, Out. 2017, pp. 65–70. doi: 10.1109/EDOCW.2017.19.

[12] M. Välja, M. Korman, R. Lagerström, U. Franke, e M. Ekstedt, «Automated Architecture Modeling for Enterprise Technology Management Using Principles from Data Fusion : A Security Analysis Case», 2016, pp. 14–22. Acedido: Abr. 20, 2021. [Em linha]. Disponível em: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-242720

[13] M. Välja, R. Lagerström, U. Franke, e G. Ericsson, «A Framework for Automatic IT Architecture Modeling: Applying Truth Discovery», *Complex Syst. Inform. Model. Q.*, vol. 0, n. 20, Art. n. 20, Out. 2019, doi: 10.7250/csimq.2019-20.02.

[14] L. Harzenetter, U. Breitenbücher, F. Leymann, K. Saatkamp, B. Weder, e M. Wurster, «Automated Generation of Management Workflows for Applications Based on Deployment Models», em *2019 IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)*, Out. 2019, pp. 216–225. doi: 10.1109/EDOC.2019.00034.

[15] A. P. Simović, S. Babarogić, e O. Pantelić, «A domain-specific language for supporting event log extraction from ERP systems», em *2018 7th International Conference on Computers Communications and Control (ICCCC)*, Mai. 2018, pp. 12–16. doi: 10.1109/ICCCC.2018.8390430.

[16] M. Christ, J. Krumeich, e A. W. Kempa-Liehr, «Integrating Predictive Analytics into Complex Event Processing by Using Conditional Density Estimations», em *2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW)*, Set. 2016, pp. 1–8. doi: 10.1109/EDOCW.2016.7584363.

[17] M. Roudjane, D. Rebaïne, R. Khoury, e S. Hallé, «Real-Time Data Mining for Event Streams», em *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*, Out. 2018, pp. 123–134. doi: 10.1109/EDOC.2018.00025.

[18] M. Roudjane, D. Rebaïne, R. Khoury, e S. Hallé, «Predictive Analytics for Event Stream Processing», em *2019 IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)*, Out. 2019, pp. 171–182. doi: 10.1109/EDOC.2019.00029.

[19] M. Boer, M. Friedrich, M. Krämer, P. Noack, J. N. Weiss, e A. Zimmermann, «Towards Resilient Enterprise Architecture for Predictive Maintenance», em *Innovation in Medicine and Healthcare Systems, and Multimedia*, Singapore, 2019, pp. 381–391. doi: 10.1007/978-981-13-8566-7_36.

[20] P. Johnson, R. Lagerström, U. Franke, e M. Ekstedt, «Modeling and analyzing systems-of-systems in the Multi-Attribute Prediction Language (MAPL)», 2016, pp. 1–7. Acedido: Abr. 20, 2021. [Em linha]. Disponível em: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-195039

[21] P. Johnson, J. Ullberg, M. Buschle, U. Franke, e K. Shahzad, «An architecture modeling framework for probabilistic prediction», *Inf. Syst. E-Bus. Manag.*, vol. 12, n. 4, pp. 595–622, Nov. 2014, doi: 10.1007/s10257-014-0241-8.

[22] S. Hacks e H. Lichter, «A Probabilistic Enterprise Architecture Model Evolution», em *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*, Out. 2018, pp. 51–57. doi: 10.1109/EDOC.2018.00017.

[23] P. Johnson, M. Ekstedt, e R. Lagerstrom, «Automatic Probabilistic Enterprise IT Architecture Modeling: A Dynamic Bayesian Networks Approach», em *2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW)*, Set. 2016, pp. 1–8. doi: 10.1109/EDOCW.2016.7584351.

[24] Q. Duan, J. Zeng, K. Chakrabarty, e G. Dispoto, «Accurate Predictions of Process-Execution Time and Process Status Based on Support-Vector Regression for Enterprise Information Systems», *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 34, n. 3, pp. 354–366, Mar. 2015, doi: 10.1109/TCAD.2014.2387831.

[25] K. M. Hanga, Y. Kovalchuk, e M. M. Gaber, «A Graph-Based Approach to Interpreting Recurrent Neural Networks in Process Mining», *IEEE Access*, vol. 8, pp. 172923–172938, 2020, doi: 10.1109/ACCESS.2020.3025999.

[26] N. Mehdiyev, J. Evermann, e P. Fettke, «A Novel Business Process Prediction Model Using a Deep Learning Method», *Bus. Inf. Syst. Eng.*, vol. 62, n. 2, pp. 143–157, Abr. 2020, doi: 10.1007/s12599-018-0551-3.

[27] N. Mehdiyev e P. Fettke, «Manuscript submitted (10.07.2020) to the edited volume "Interpretable Artifi-cial Intelligence: A perspective of Granular Computing" (published by Springer). Explainable Artificial Intelligence for Process Mining: A General Overview and Application of a N», 2020.