Accepted manuscript. This is the author's version of the work. It is posted here for your personal use. Not for redistribution.
The definitive version is published by ELSEVIER in the "Journal of Systems and Software".
The final publication can be accessed under the DOI: https://doi.org/10.1016/j.jss.2021.111004

Three Decades of Software Reference Architectures: A Systematic Mapping Study

Lina Garcés[a], Silverio Martínez-Fernández[b], Lucas Oliveira[c], Pedro Valle[d], Claudia Ayala[b], Xavier Franch[b] and Elisa Yumi Nakagawa[d]

[a] *Institute of Mathematics and Computing, Federal University of Itajubá, Itajubá, Brazil*
[b] *Universitat Politécnica de Catalunya, BarcelonaTech, Barcelona, Spain*
[c] *Federal Institute of São Paulo, São Carlos, Brazil*
[d] *Department of Computer Systems, University of São Paulo, São Carlos, Brazil*

ARTICLE INFO

Keywords:
Software architecture
Reference architecture
Systematic mapping
Secondary study

ABSTRACT

Software reference architectures have played an essential role in software systems development due to the possibility of knowledge reuse. Although increasingly adopted by industry, these architectures are not yet completely understood. This work presents a panorama on existing software reference architectures, characterizing them according to their context, goals, perspectives, application domains, design approaches, and maturity, as well as the industry involvement for their construction. For this, we planned and conducted a systematic mapping study. During last decade, the number of reference architectures in very diverse application domains has increased, resulting from efforts of industry, academia, and through their collaborations. Academic reference architectures are oriented to facilitate the reuse of architectural and domain knowledge. The industry has focused on architectures for standardization with certain maturity level. However, the great amount of architectures studied in this work have been designed without following a systematic process, and they lack the maturity to be used in real software projects. Further investigations can be oriented to gathering empirical evidences, from different sources than academic data libraries, that allow to understand how references architectures have been constructed, utilized, and maintained during the whole software life-cycle.

1. Introduction

Software architectures are widely recognized as a backbone for any successful software-intensive system, and therefore, they play a fundamental role in determining the software quality of such systems [57]. In this scenario, reference architectures capture the essence of the architectures of a collection of systems in a given domain [14]. The purpose of reference architectures is to provide guidance for the development, standardization, and evolution of architectures in a specific application domain [19]. Given the above benefits, many reference architectures have been proposed in the last decades, and some of them have gained wide acceptance in their domains over the years. Well-known examples are CORBA for objects integration through brokers [94], AUTOSAR for the automotive sector [9, 91], S3 to design service-oriented solutions [7], EIRA, a European reference architecture to build interoperable e-Government systems [25], and the NIST Big Data Interoperability Framework [98]. These examples have in common being developed by consortia involving major industrial players (such as manufacturers and suppliers) and researchers.

Besides the practical use of reference architectures, theoretical foundations were developed along the years, aiming at establishing a definition to these architectures, characterizing them, proposing processes and methods to design them, including their architectural analysis, synthesis, and evaluation [4, 19, 23, 31, 33, 36, 73], as well as finding benefits and drawbacks of such architectures aiming at supporting organizations' decision making to their adoption [6, 63]. The main benefits from both industry and academy perspectives, reported in literature, include increased interoperability among systems/subsystems, reduction of development costs/time by making possible the reuse, reduction of risks in software projects, improvement in communication, and adoption of the best practices. On the other hand, drawbacks include high learning curve, lack of documentation, and/or an inadequate representation [64].

Even with a number of reference architectures being proposed in the last decades, these architectures are not yet completely understood regarding their characteristics, purposes, application domains, and maturity. Furthermore, it is also unclear the process that yielded to their construction and if they are aligned to the theoretical foundations mentioned above. Motivated by this lacuna, the main objective of this work is to provide a survey on the current state of the art of software reference architectures. In order to ensure its rigor, this survey was undertaken as a Systematic Mapping Study (SMS) based on original guidelines proposed in [53, 78]. We have identified up to 162 reference architectures reported in the scientific literature, and we have characterized them based on significant traits, like their application domains, context, purpose, perspective, design approach, and maturity. In addition, we have identified gaps and possible future work.

This article is organized as follows. Section 2 presents

lina@unifei.edu.br (L. Garcés); silverio.martinez@upc.edu (S. Martínez-Fernández); lucas.oliveira@ifsp.edu.br (L. Oliveira); pedrohenriquevalle@usp.br (P. Valle); cayala@essi.upc.edu (C. Ayala); franch@essi.upc.edu (X. Franch); elisa@icmc.usp.br (E.Y. Nakagawa)
ORCID(s):

© 2021 Elsevier. This manuscript version is made available under the CC-BY-NC-ND 4.0 license http://creativecommons.org/licenses/by-nc-nd/4.0/

a brief background on the main theory on reference architectures that is important to better understand the analysis developed in this work. It also presents related work. Section 3 details the planning and execution procedure of our SMS. Section 4 shows a detailed panorama of the existing reference architectures, answering the research questions of our SMS. Section 5 discusses the main findings on the results. Section 6 reports the threats of validity of this study. Finally, Section 7 concludes this work summarizing how this work advances the state-of-the-art on the topic of reference architectures.

2. Background and Related Work

Different definitions of reference architecture have been proposed over the years [14, 38, 56, 70, 81]; in general, they have focused on different aspects of systems, e.g., business, software, technology, domain, hardware, or network. All these definitions have in common the reuse of knowledge for the development and evolution of a set of systems in a given application or technology domain. As this study focused on the *software* aspect, we considered a **software reference architecture** as an abstraction of software elements, together with the main responsibilities and interactions of such elements, capturing the essentials of existing software systems in a domain and serving as a guide for the architectural design of new software systems (or versions of them) in the domain.

From an engineering perspective, building reference architectures for modern software systems is a challenging task. Therefore, several authors have proposed guidelines, principles, recommendations, and processes [4, 19, 23, 33, 73, 36, 31] to support such a challenge. For instance, [73] presented a process to design reference architectures, based on the general model for software architecture design proposed in [48], applying the following steps:

- Establish their scope, defined as the set of systems intended to be produced based on these architectures.
- Select and investigate as many information sources as needed to gain all required domain knowledge.
- Perform an architectural analysis, referring to the identification of architecturally significant requirements.
- Carry out the synthesis of the reference architecture, representing it by using some architectural description language.
- Evaluate the reference architecture through surveys and even their instantiation and use.

This process has been successfully applied to establish different reference architectures for both academia [72] and industry [84].

With regard to related work, a number of literature reviews (systematic or not) were already conducted to identify reference architectures specifically in different domains, including cyber-defense [86], Internet of Things (IoT) [67], service-oriented systems [22], self-adaptive systems [1, 34], ambient assisted living [37], industry 4.0 [71, 79, 90], big data [88, 95], and smart grids [3], and for investigating their application in agile projects [31, 102]. In general, existing literature reviews are interested in obtaining an overview in target domains. However, reference architectures research lacks of a comprehensive understanding on how these architectures have been created, their purposes, and maturity, which is the main goal of this work.

3. Research Method

Having the identification and analysis of existing software reference architectures as the main goal, this SMS was planned, conducted, and reported following well-known, experimented guidelines found in [53, 78]. Planning and conduction are described below.

3.1. Planning

This study addressed five research questions (RQ), which are presented below together with their metrics to quantify evidence and support results analysis:

- **RQ1 -** Which are the software reference architectures proposed and/or used over the years?

  Rationale: To obtain a comprehensive panorama of existing software reference architectures and which of them had involvement of academia and industry in their design.

  Metrics: Set of all software reference architectures found in scientific literature, besides the amount of architectures by year and by involvement (industry or academia).

- **RQ2 -** For which types of systems and/or application domains have software reference architectures been proposed?

  Rationale: To identify types of systems, such as service-oriented systems and self-adaptive systems, or domains (e.g., transportation, health-care) that have already benefited from reference architectures.

  Metrics: Types of software systems or application domains of the architectures, besides the amount of architectures by type of systems or domain.

- **RQ3 -** How are the existing software reference architectures characterized?

  Rationale: To understand the context (single or multiple organizations), purpose (facilitation or standardization), and perspective (classical or preliminary) of such architectures.

  Metrics: The amount of software architectures in each context, purpose, and perspective, besides the relationships among them.

- **RQ4 -** How have the existing software reference architectures been designed?

  Rationale: To find out if these architectures are designed using some systematic process.

Metrics: Amount of software reference architectures using available design approaches (processes, methods, steps, guidelines, among others).

- **RQ5 -** Which is the maturity of existing software reference architectures?

  Rationale: To find the maturity level of these architectures, which can indicate their feasibility and also the impact in their domain.

  Metrics: Amount of software reference architectures in each maturity level, *i.e.,* their evaluation level [61, 89]: E0 (not evaluated), E1 (assessed using toy examples), E2 (evaluated by experts), E3 (assessed through controlled experiments), E4 (applied in industrial case studies); and E5 (used in industry). We believe architectures with a good evaluation such as their use in industry projects are certainly more mature if compared with others with only an academic toy example.

We considered as the population of our SMS all existing reference architectures for software systems. Therefore, we defined the search string based on two main keywords: (i) *reference architecture* as the broader term to obtain as most studies as possible in the researched topic, and (ii) *software architecture* to constraint our search to software systems, and avoiding studies dealing with enterprise architectures, hardware architectures, or building architectures. Moreover, we decided to avoid generic terms as framework, model, or blueprint, since by using them in the search string, most results obtained from data libraries were out of this SMS scope.

As *reference architecture* is a well-known, disseminated term, we did not consider other related terms; otherwise, we considered the following similar terms of *software architecture* as listed by [80]: software structure, software design, system architecture, system structure, and system design. Hence, final search string[1] was: *("reference architecture" and ("software architecture" or "software structure" or "software design" or "system architecture" or "system structure" or "system design")).* With regard to the publication databases, we selected those ones according to [24, 53]: Scopus[2], Web of Science[3], IEEE Xplore[4], ACM Digital Library[5], ScienceDirect[6], and SpringerLink[7]. Scopus, ScienceDirect, and Web of Science are general indexing systems and allow us to cover a broader scope for our search. IEEE Xplore, ACM Digital Library, and SpringerLink publish works of the most important venues (conferences and journals) related to software architectures. The search string and the search procedure was tailored to the specific characteristics of every digital library.

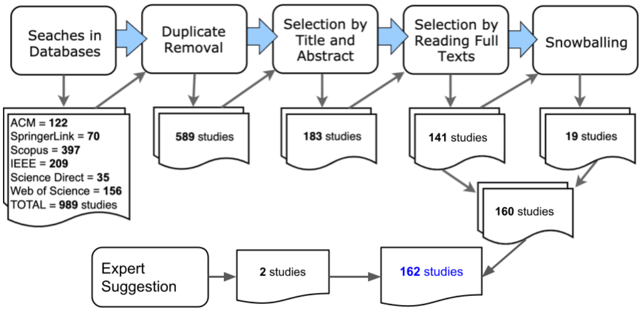


Figure 1: Process to the selection of relevant primary studies

We also defined two inclusion criteria (IC) and five exclusion criteria (EC):

- IC1: The study proposes a reference architecture for software systems.
- IC2: The study reports the use of a reference architecture for software systems.
- EC1: The study does not address a reference architecture for software systems, but for other areas (*e.g.,* enterprise architectures or hardware architectures).
- EC2: The study addresses (proposes or uses) a reference architecture for software systems, but another more complete study related to the same architecture exists.
- EC3: The study does not provide detailed information about a reference architecture, because it is a table of contents, short course description, invited talk of events, summary of events, among others.
- EC4: The study was not peer reviewed.
- EC5: The study was written in other language than English.

3.2. Conducting the Search

This SMS was conducted from January to July, 2018 by seven researchers with experience in reference architectures and software architectures. They also have experience in researching, conducting, and updating several SMS and systematic literature reviews.

Figure 1 depicts steps of the selection process. By adapting the search string for each database and considering the search on title, abstract, and keywords, we obtained a total of 989 studies and removing duplicates, 589 studies remained. After the first selection where we applied the selection criteria on title, abstract, and keywords, 183 studies were selected. After reading the full text of these studies and applying the selection criteria again, 141 primary studies were selected. Besides that, a backward and forward snowballing inspection [83, 99] on the list of references of each selected study made us possible to include other 19 relevant studies,

[1] For this search string, we also consider the plural form of all terms, but for simplification, only the singular terms are shown here.

[2] http://scopus.com

[3] http://isiknowledge.com

[4] https://ieeexplore.ieee.org

[5] http://dl.acm.org

[6] http://sciencedirect.com

[7] http://link.springer.com

totaling 160 studies. As a final step, we considered 2 additional studies identified by an expert and that were not recovered in previous stages. In cases where we found more than one study detailing the same architecture we only included the most complete one. As a result, we identified 162 unique reference architectures, which are listed in Appendix A (Table 5) together with their ID (RA1 to RA162), proponent, purpose, perspective, maturity, and domain.

To support this selection process, we used JabRef[8]. To ensure the results reliability and mitigate researcher bias in our mapping, the selection process was carried out by four researchers as recommended by [100]. Every study was reviewed exactly by two researchers. In case of conflict between the two reviewers, a third researcher analyzed the study and solved the disagreement.

3.3. Data extraction, Analysis, and Synthesis

To obtain evidence to answer our RQs, we used the data extraction form, which is available as a supplementary material of this work in [35]. The data extraction form contains the properties addressing our five research questions: bibliometrics, application domains, context, design, and maturity.

To share a common understanding of the data extraction form and mitigate researcher bias, three researchers piloted the data extraction form with 15 studies. Minor changes were made to the initial version of the data extraction form, consisting on descriptions for each classification criteria as described below. This helped to clarify our understanding on the classification criteria. We then performed the data extraction of all studies.

To characterize the context (RQ3) and maturity (RQ5) of the selected reference architectures, we analyzed them using the three main aspects of the framework proposed by Angelov et al. [4], and an adjusted evaluation level from [61, 89]. It is worth remarking that a composition of these aspects results in types of reference architecture, which, according to [4], could categorize most of the existing reference architectures. These classification criteria consisted of the following pre-defined fields:

- Context refers to the organizations that can be the intended recipients of the reference architectures, i.e., an architecture could benefit only a *single organization* or be addressed to *multiple organizations*, which share a certain property (a market domain or a geographical property such as a region or country).

- There are two main purposes for reference architectures. Firstly, architectures for *facilitation*, which essentially promote sharing and reuse of large knowledge and experience accumulated for years, related to how better and more easily architect software systems in their domains. Secondly, those for *standardization* mainly aim at improving interoperability among components in a system, or among collaborating systems in such domains.

- With respect to the perspective, *classical* architectures aggregate best practical experience from the use of preferably a number of different software systems. On the other hand, *preliminary* architectures are defined when there is not yet a large set of software systems in a given domain or type of systems, but only as research experiments, or even they can provide an innovative architectural design with respect to the existing state of the art.

- The maturity level of reference architectures is measured in a scale from 0 to 5: E0 (not evaluated), E1 (assessed using toy examples), E2 (evaluated by experts), E3 (assessed through controlled experiments), E4 (applied in industrial case studies); and E5 (used in industry).

In a further step, data extracted was crosschecked by the researchers. This step helped ensure correct extraction of data and minimize human error.

Extracted data from the final set of included studies was stored in a database and managed using Python libraries Plotly[9] and Pandas[10]. Such extracted data is also available in the open science package of this study [35].

For data analysis, we used qualitative and narrative synthesis methods as recommended in [29]. The results are presented in the next section.

Also, to analyze fields of the data extraction form with open text (e.g., application domain of RQ2), we used thematic analysis [21]. We first applied open coding to extracted application domains (e.g., SOA, transportation), then categorized related codes to form themes, and further refined the themes to identify higher-order themes (i.e., categories) of application domains (e.g., software environments).

4. Results

This section offers a detailed panorama about the existing reference architectures answering each RQ proposed in our SMS.

4.1. Existing reference architectures

With regard to RQ1, Figure 2 summarizes the 162 reference architectures found by year[11], highlighting the amount of them that were designed in the academic context (in gray color), industry context (in dark-blue color), and collaboration between industry-academy context (in light-blue color). Table 5 in Appendix A, lists the final set of studies reporting reference architectures and that were analyzed in this SMS.

We considered as academic architectures those that have been reported only by authors from academia and/or research centers. We found that 61.1% (99/162) of architectures are from academy. Industry reference architectures refer to those

[8] http://www.jabref.org

[9] https://plot.ly/python

[10] https://plot.ly/pandas

[11] In case of reference architectures in several studies, as the result of applying EC2, this refers to the year when the most complete study of a given reference architecture was published.

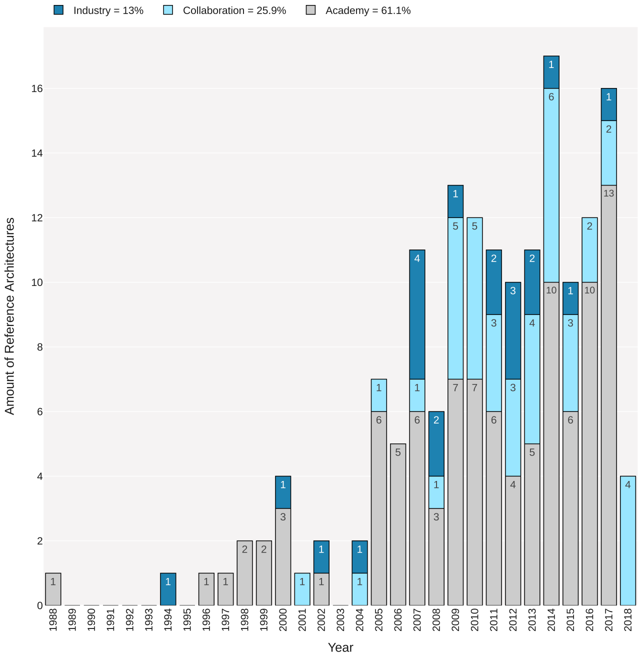
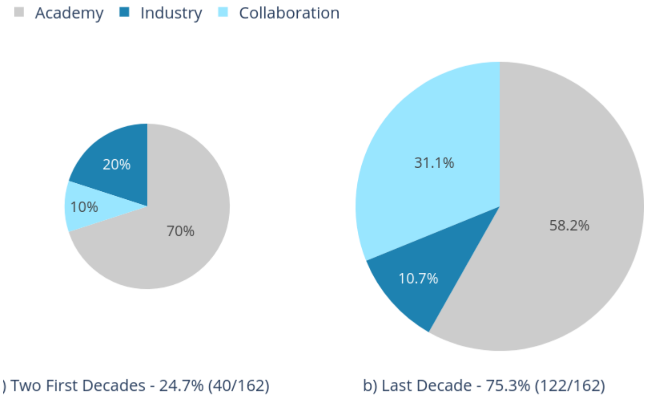


Figure 2: Distribution of reference architectures over the years.



) Two First Decades - 24.7% (40/162)

b) Last Decade - 75.3% (122/162)

Figure 3: Changes on Participation of Academy, Industry, and Collaborations During Three Decades.

with all authors from industry. 13% (21/162) of these architectures have been proposed in industry context. The remaining 25.9% (42/162) of reference architectures were constructed through collaborations between practitioners and academics.

The first reference architecture retrieved in this study was published in 1988, *i.e.,* around 30 years ago, and it was defined by academics for software development environments. There was a gap until middle of 90's when such architectures started to be more researched and used. The first industry architecture, found in this SMS, was published in 1994 for the transportation domain. The first architecture, recovered in this study, designed through collaboration between industry-academy was reported in 2001 for the healthcare domain. The last decade (i.e., from 2008 to 2018) concentrates an increasingly number of architectures, (75.3%, or 122/162). Hence, we can say that reference architecture is still an active research area.

Figure 3 compares the proportion of reference architectures proposed by academy, industry, and through their collaboration over the first twenty-years and the last decade. The 24.7% (40/162) of reference architectures were conceived during the first two decades. By that time, this research area was mainly led by academia, since the 70% (28/40) of these architectures were made only by academics. In addition, considering their collaboration with industry (i.e., with whom academics proposed 4/40, or the 10% of the architectures), academia accumulated the 80% (32/40) of participation in this area until the year of 2007.

A great amount, i.e., 75.3% (122/162) of reference architectures were published in scientific data libraries during the last decade. As expected, academia has maintained its leadership, contributing with the construction of 89.3% of reference architectures (i.e., 71/122 made only by academics, plus 38/122 made with industry collaboration). Industry involvement (considering its collaboration with academia) s-trengthened during the last years, increasing their proportion from 30% (12/40) in the first two decades to 41.8% (51/122) in the last decade. In other words, a great amount of architectures with industry participation (i.e., 81% or 51/63) were proposed since 2008, as depicted in Figures 2 and 3. Moreover, the dissemination of architectures made only by industry decreased during the last 10 years; however, its collaboration with academia increased, participating on the establishment of 31.1% (38/122) of the newest reference architectures. The increasingly involvement of industry could evidence that software companies have started to trust on such architectures to bring benefits to their projects. Additionally, the increasing collaboration between industry and academia could mean that more robust initiatives are being put forward in industry to reinforce knowledge reuse.

4.2. Types of systems and application domains of the reference architectures

With respect to the types of systems or domains targeted by the reference architectures aiming at answering RQ2, we found that reference architectures were designed for 40 different types of systems or domains, as depicted in Figure 4. It is worth highlighting that there exist architectures that could be classified in two or more domains/types; however, in these cases, we classified them according to the main target of the architecture. For instance, RA88 addresses both cloud computing and software environments/tools as a service, and was classified as cloud computing because it mainly focuses on cloud infrastructure issues than software environments/tools.

We found fifteen types of systems and application domains that have received more attention from reference architectures creators (i.e., with at least five reference architecture), namely, software environments, business management, transportation, Service-Oriented Architecture (SOA),

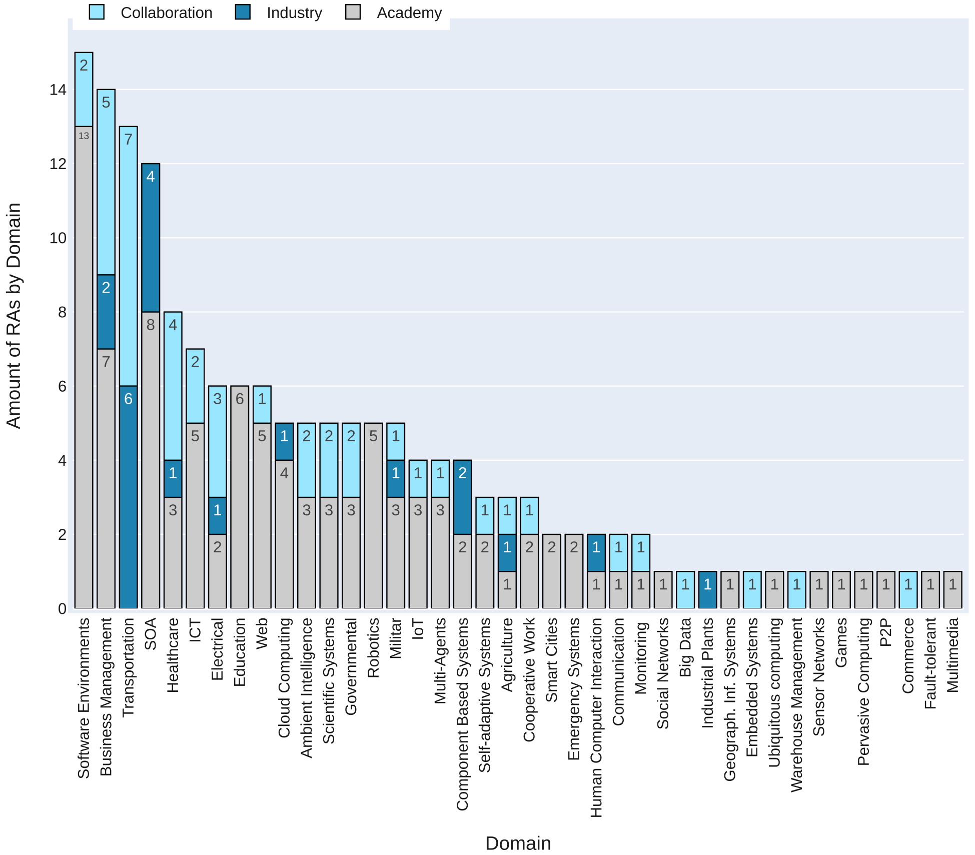


Figure 4: Domains and types of software systems of the existing reference architectures.

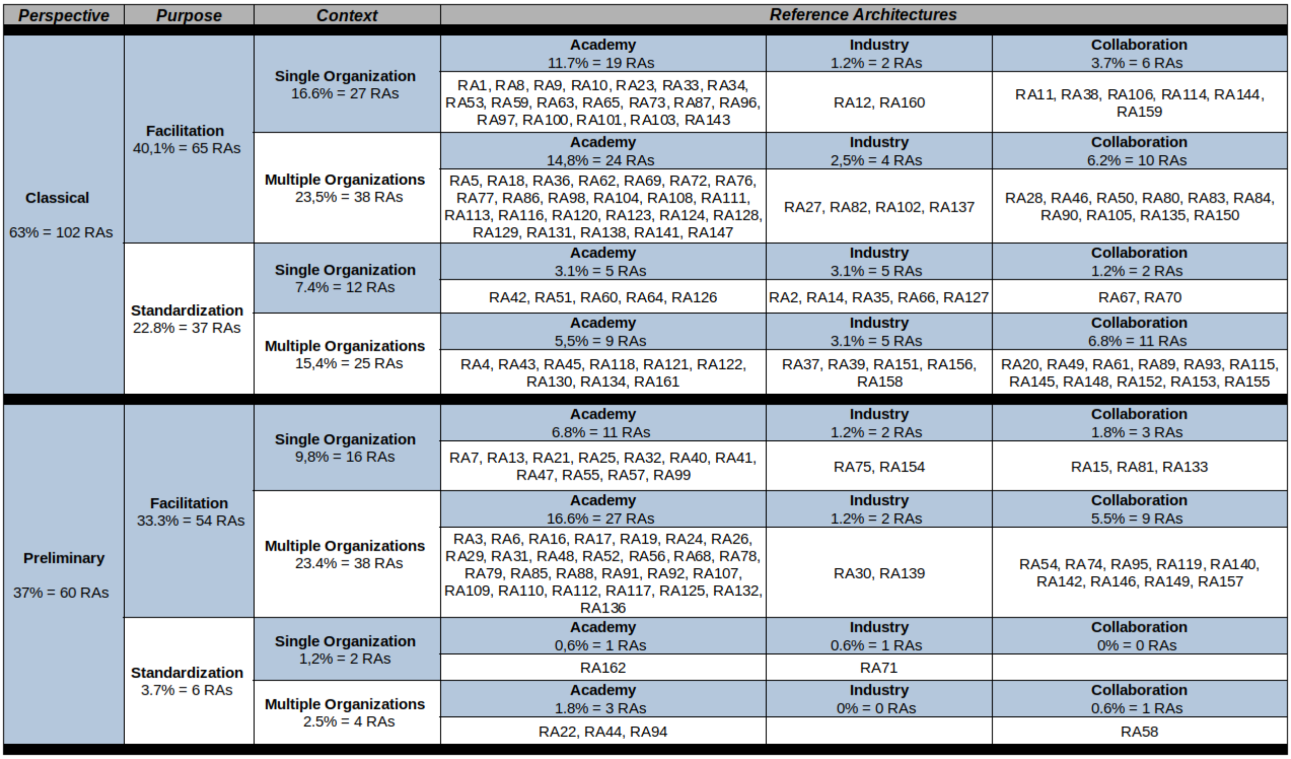
healthcare, Information and Communication Technologies (ICT), electrical, education, web applications, cloud computing, ambient intelligence, scientific, governmental, robotics, and military. The 72.2% (117/162) of existing reference architectures have been focused on any of those domains. We have observed an interesting trend in the transportation domain, which has a high number of architectures (13 in total), all of them with industry involvement. In fact, this group has important architectures as AUTOSAR (RA156) for automotive sector and ARC-IT (RA152) for cooperative and intelligent transportation. Collaboration between industry and academia has been more common (i.e., with more than 3 architectures by domain) in domains of business management, transportation, healthcare, and electrical systems. Academia has led reference architectures in domains of software environments, business management, SOA, ICT, Web, education, and robotic systems, proposing at least 5 architectures for each of these types of systems. Existing reference architectures in domains of education, robotics, smart cities, and emergency systems have been defined only by academics.

4.3. Classification of reference architectures

To characterize the reference architectures to answer RQ3, we followed the classification proposed in [5] and gathered the perspective (classical or preliminary), purpose (facilitation or standardization), and context (single or multiple organizations) of these architectures. This information is presented in Figure 5 and listed in columns three to five of Table 5 (in Appendix A).

4.3.1. Perspective of Reference Architectures

Analyzing the perspective of all reference architectures, whose distribution is shown in Figure 5, we found that 63% (102/162) of them are *classical*, i.e., authors recovered/mined knowledge from existing software systems in the domain to learn from them and define the reference architecture. A representative example of a classical architecture is S3 (RA27) destined to service orientation that has already an amount of service-oriented systems implemented and running. To design *preliminary* architectures, authors considered not only existing software systems, but also proposed how future systems could be created from general architectures. The 37% (60/162) of existing reference architectures are preliminary and representative examples are those for unmanned air-crafts



Perspective	Purpose	Context	Reference Architectures		
Classical 63% = 102 RAs	Facilitation 40,1% = 65 RAs	Single Organization 16.6% = 27 RAs	Academy 11.7% = 19 RAs	Industry 1.2% = 2 RAs	Collaboration 3.7% = 6 RAs
			RA1, RA8, RA9, RA10, RA23, RA33, RA34, RA53, RA59, RA63, RA65, RA73, RA87, RA96, RA97, RA100, RA101, RA103, RA143	RA12, RA160	RA11, RA38, RA106, RA114, RA144, RA159
		Multiple Organizations 23,5% = 38 RAs	Academy 14,8% = 24 RAs	Industry 2,5% = 4 RAs	Collaboration 6.2% = 10 RAs
			RA5, RA18, RA36, RA62, RA69, RA72, RA76, RA77, RA86, RA98, RA104, RA108, RA111, RA113, RA116, RA120, RA123, RA124, RA128, RA129, RA131, RA138, RA141, RA147	RA27, RA82, RA102, RA137	RA28, RA46, RA50, RA80, RA83, RA84, RA90, RA105, RA135, RA150
	Standardization 22.8% = 37 RAs	Single Organization 7.4% = 12 RAs	Academy 3.1% = 5 RAs	Industry 3.1% = 5 RAs	Collaboration 1.2% = 2 RAs
			RA42, RA51, RA60, RA64, RA126	RA2, RA14, RA35, RA66, RA127	RA67, RA70
		Multiple Organizations 15,4% = 25 RAs	Academy 5,5% = 9 RAs	Industry 3.1% = 5 RAs	Collaboration 6.8% = 11 RAs
			RA4, RA43, RA45, RA118, RA121, RA122, RA130, RA134, RA161	RA37, RA39, RA151, RA156, RA158	RA20, RA49, RA61, RA89, RA93, RA115, RA145, RA148, RA152, RA153, RA155
Preliminary 37% = 60 RAs	Facilitation 33.3% = 54 RAs	Single Organization 9,8% = 16 RAs	Academy 6.8% = 11 RAs	Industry 1.2% = 2 RAs	Collaboration 1.8% = 3 RAs
			RA7, RA13, RA21, RA25, RA32, RA40, RA41, RA47, RA55, RA57, RA99	RA75, RA154	RA15, RA81, RA133
		Multiple Organizations 23.4% = 38 RAs	Academy 16.6% = 27 RAs	Industry 1.2% = 2 RAs	Collaboration 5.5% = 9 RAs
			RA3, RA6, RA16, RA17, RA19, RA24, RA26, RA29, RA31, RA48, RA52, RA56, RA68, RA78, RA79, RA85, RA88, RA91, RA92, RA107, RA109, RA110, RA112, RA117, RA125, RA132, RA136	RA30, RA139	RA54, RA74, RA95, RA119, RA140, RA142, RA146, RA149, RA157
	Standardization 3.7% = 6 RAs	Single Organization 1,2% = 2 RAs	Academy 0,6% = 1 RAs	Industry 0.6% = 1 RAs	Collaboration 0% = 0 RAs
			RA162	RA71	
		Multiple Organizations 2.5% = 4 RAs	Academy 1.8% = 3 RAs	Industry 0% = 0 RAs	Collaboration 0.6% = 1 RAs
			RA22, RA44, RA94		RA58

Figure 5: Reference architectures classified according to their perspective, purpose, and context

(RA81) and space on-board software (RA58), which were proposed with few experience or amount of real systems implemented and deployed. Academia had invested more efforts, proposing more than the half (i.e., 55.8%, or 57/102) of classical reference architectures. Industry has been responsible for creating the 15.6% (16/102) of these architectures, and also has participated, collaborating with academia, of the construction of the 28.4% (29/102) of classical architectures. Preliminary reference architectures have also been mostly proposed by academia, which has participated with 70% (42/60) architectures. The remaining 30% (18/60) of preliminary architectures have been defined in industry context, from which, 13 of them had academia support.

4.3.2. Purpose of Reference Architectures

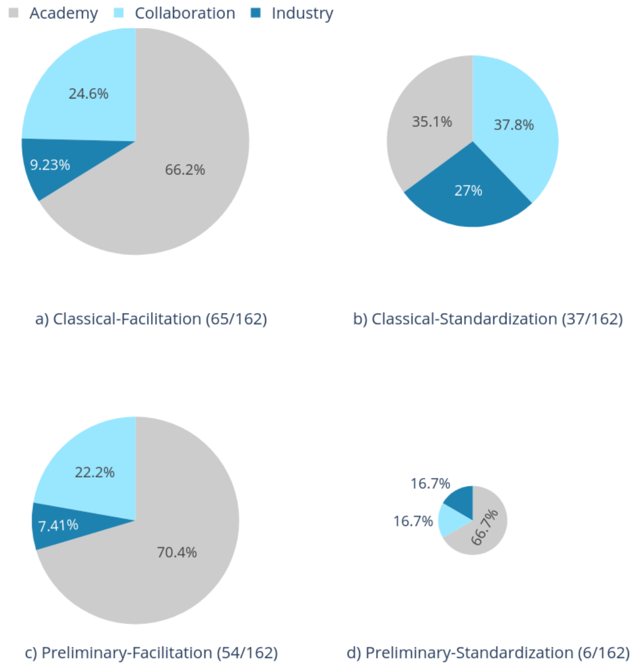
With regard to their purpose, also shown in Figure 5, 73.5% (119/162) of the reference architectures were conceived for *facilitation*, supporting the understanding of the main building blocks of concrete architectures and serving as guidelines to design systems in the domain. The remaining 26.5% (43/162) of the architectures were destined to *standardization*, promoting the unification of strategies to normalize the way that software systems in the domain could be developed. Some architectures have both purposes, then we decided to classify them for standardization purpose, since they also could be used to facilitate the understanding on how to create new architectures. This is the case of ARC-IT (RA152), which is essentially for standardization, although it has facilitation as a secondary purpose.

Figure 6 illustrates the participation of academy, industry, and through their collaboration, to propose reference architectures of different purposes and perspectives. At first glance, it is possible to see in Figures 6a) and 6c) that, architectures for facilitation (i.e., with classical or preliminary perspective) have been proposed mostly by academia. Industry participation (with or without collaboration with academia) is few comparing with academia efforts for this type of architectures. However, industry involvement has augmented by creating architectures for standardization. For instance, in Figure 6b), industry has participated of the construction of the 62.1% (23/37) of classical architectures and the 33.3% (2/6) of the preliminary architectures for standardization.

The 68.1% (81/119) of architectures for facilitation have been idealized in the academic context. Industry has been involved in the construction of 31.9% (38/119) of these architectures, from which the 23.5% (28/119) of them had the collaboration of academia. The 58.1% (25/43) of reference architectures for standardization counted with industry contributions. Henceforth, architectures for standardization have obtained more interest from industry (i.e., which contributed with 25.6% or 11/43 of these types of architectures) comparing with those architectures with facilitation purposes (i.e., for which industry proposed 8.4% or 10/119 of them).

4.3.3. Context of Reference Architectures

Considering the context where reference architectures will be used, *i.e.,* single organization or multiple organizations, we found that 35.2% (57/162) were destined to single institu-



a) Classical-Facilitation (65/162)

b) Classical-Standardization (37/162)

c) Preliminary-Facilitation (54/162)

d) Preliminary-Standardization (6/162)

Figure 6: Proportion of Reference Architectures by Perspectives (classical or preliminary) and Purposes (facilitation or standardization) proposed by Industry, Academia, or both.

tions, while 64.8% (105/162) were built to be used by multiple organizations. Representative example for single organization is the Mars Design Reference Architecture (DRA 5.0) (RA154) that describes the systems and operations used for missions to explore the Mars surface by NASA. An example for multiple organizations is the reference architecture (RA70) that can be adopted to the development of warehouse management and control systems by any organization. Academia have contributed in more proportion to both contexts, comparing with industry, at creating 64.9% (36/57) and 60% (62/105) of architectures for both single and multiple organizations, respectively. Though, the 64.3% (63/98) of academic architectures are oriented to multiple organizations. Similarly, the 73.8% (31/42) of architectures made through industry-academia collaboration are intended for multiple organizations. On the other hand, architectures made only by industry have been oriented, in similar proportions, to both single (with the 47.6% or 10/21 architectures) and multiple (with the 47.6% or 11/21) organizations.

Putting together information about the perspective, purpose, and context, Figure 5 lists the classification of each reference architecture. We can observe that classical architectures for facilitation are more commonly found totaling 40.1% (65/162), while preliminary architectures for standardization are less numerous, *i.e.*, only six architectures (or 3.7%). We also observed that 46.9% (76/162) of reference architectures were defined with facilitation purposes and oriented to multiple organizations with more participation of academia than industry.

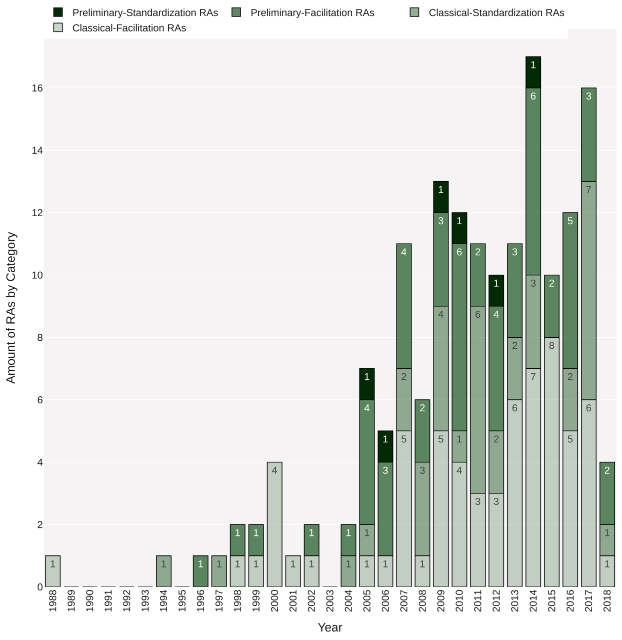


Figure 7: Classification of Reference Architectures by Purposes (facilitation or standardization) and Perspectives (classical or preliminary) Over Years.

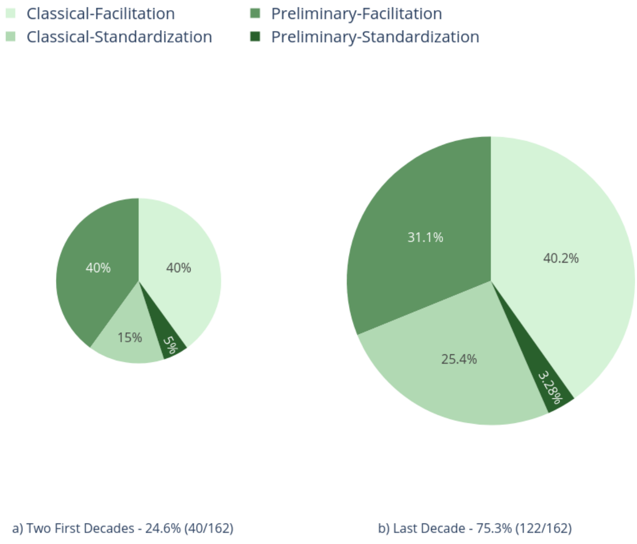


Figure 8: Classification of Reference Architectures by Purposes (facilitation or standardization) and Perspectives (classical or preliminary) Over Decades.

Based on information presented in Figures 7 and 8, we observe that during the first two decades (from 1988 to 2007), classical and preliminary architectures with facilitation purposes were equally predominant, each type reported in 40% (16/40) of analyzed studies. In the last decade, classical architectures for facilitation had maintained their predomi-

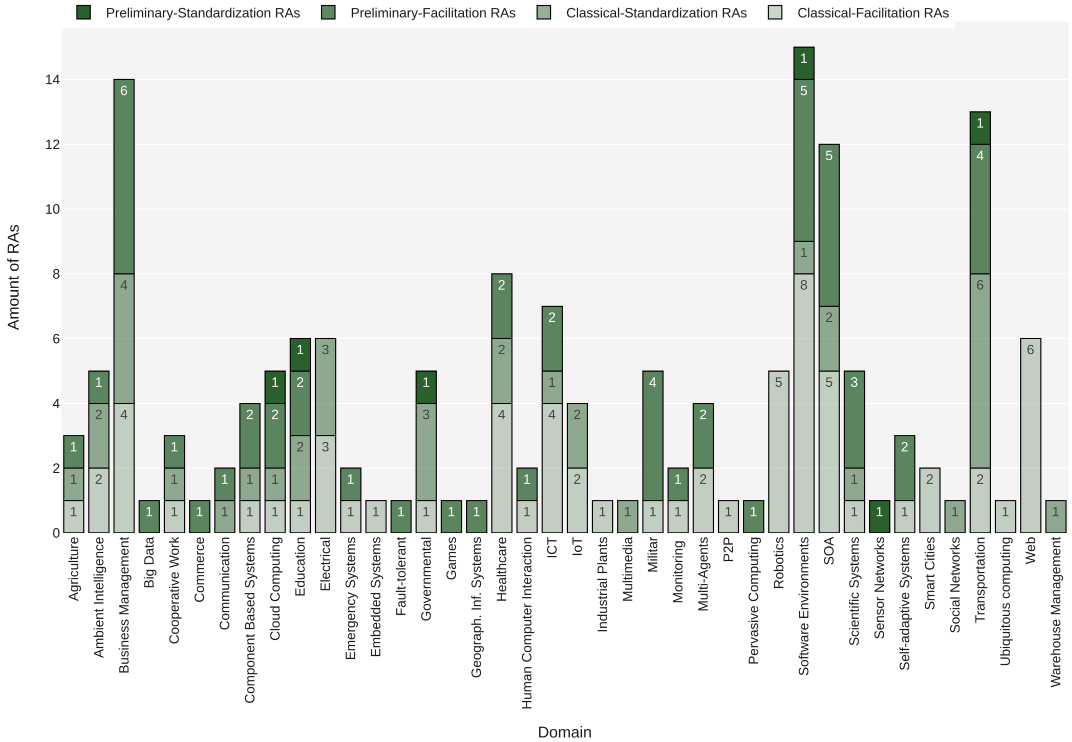


Figure 9: RAs for Different Domains and Type of Systems Classified by Purposes (facilitation or standardization) and perspectives (classical or preliminary).

nance with 40.2% (49/122) of published architectures, whilst the amount of classical and preliminary architectures for standardization had increased from 15% (6/40) to 25.4% (31/122) and from 5% (2/40) to 3.3% (4/122), respectively. Preliminary architectures for facilitation had decreased passed from 40% (16/40) in the two first decades to 31.1% (38/122) in last years. Moreover, the great amount (81.4%, 35/43) of architectures for standardization were proposed in the last decade.

Figure 9 classifies reference architectures regarding their domain, proponent (i.e., academia, industry, or both), and their types (i.e., classical for facilitation, classical for standardization, preliminary for facilitation, and preliminary for standardization). We identified that, independently of the purpose (facilitation or standardization) and the perspective (classical or preliminary), it is possible to find at least one architecture for software environments, transportation, and cloud computing domains. Additionally, it was possible to see that the 48.8% (21/43) of architectures for standardization are concentrated in only the 12.5% (5/40) of domains, namely, transportation, business management, governmental, education, and electrical systems. Moreover, for domains of web, robot, and military systems (i.e., each domain with at least five reference architectures) were not found any architecture with standardization purposes. The 53.3% (32/60) of preliminary architectures are focused on the 17.5% (7/40) of domains, such as, software environments, business management, transportation, SOA, military, scientific, and cloud computing systems. Classical and facilitation architectures are more dispersed among all domains.

4.4. Design of existing reference architectures

To answer RQ4, we looked for approaches (including processes, methods, guidelines, among others) used to design the 162 reference architectures, considering also any additional material available (*e.g.,* websites, technical reports, and other related studies). It is important to highlight that only 14.8% (24/162) of the primary studies reported to have adopted one of such approaches, which are listed in Table 1. For the remaining 85% (138/162), we consider that they were created through an *ad hoc* way, since it was not possible to find in their documentation any means used to create them.

For each approach found in this SMS, Table 1 presents a short description, year of publication, the context where it was proposed (Industry (I), Academia (A), or through Collaboration (C)), the ID and context (Industry (I), Academia (A), or through Collaboration (C)) of reference architectures that adopted the approach, and for which types of systems or domains such architectures were destined. The first 13 approaches listed in this table were specifically proposed to reference architectures engineering. They were also used in the design of 17 architectures (in which RA16 used two approaches [50, 13]). The last 6 approaches, used to create 7 architectures, are not specific to engineer reference architectures, but they were proposed for different architectural purposes.

We observed that the first approach for engineering reference architectures, we found in this SMS, was proposed in 1994, six years after the publication of the first reference architecture (i.e., in 1988) recovered in this study. Since then and till the conduction of this SMS, several approaches continuously appeared until 2015. We also identified that most approaches were used once, except for ProSA-RA [70] and ARM IoT [15] that were applied in six and two reference architectures, respectively.

We distinguished that approaches oriented to engineer reference architectures are focused in three main perspectives namely, derivation, application of styles, and application of empirical methods.

Four approaches propose derivation of reference architectures from existing architectures [55], from requirements and domain knowledge [70, 66], and from existing systems and products [70, 60]. Two of these approaches were defined only in academic contexts [70, 60]. Derivation has mainly been used to create classical architectures for facilitation. Ten reference architectures have been constructed through derivation and the majority (9/10) have been created by academia.

Five approaches suggest the application of architectural styles [20, 13, 66, 70, 60]. Ten reference architectures have applied architectural styles for orienting their design. Two of these approaches were proposed by industry [20, 66] for creating preliminary architectures with facilitation goals (e.g., RA29, RA74). The remaining three approaches were made by academics and have been mostly used to design classical architectures for facilitation.

Empirical methods were the basis of two approaches proposed in academia [32] and with industry collaboration [87]. Both approaches were used by industry to facilitate the design of software systems in domains of big data (RA119) and transportation (RA106).

Moreover, 4.3% (7/162) architectures adopted other approaches not specific to reference architectures engineering: the RM-ODP (Reference Model - Open Distributed Processing) [28], the QADA (Quality-Driven Architecture Design and Quality Analysis Method) [65], the 4 + 1 view model [58], a combination of TOGAF (The Open Group Architecture Framework) and Archimate (i.e., an architecture modeling language) [49], the union of SAAM (Software Architecture Analysis Method) [51] and the commonality analysis method [97], and the ARM IoT (A Reference Model to construct IoT architectures) [15].

We also found that 4 (of all 19 approaches) were proposed only by industry, 7 (out of 19) only by academia, and 8 (out of 19) through industry-academia collaboration. Approaches have been more applied in academia than in industry, *i.e.,* 17 (out of 24) architectures created by following an approach were proposed by academics, whilst 7 (out of 24) were built with industry collaboration. In a more detailed analysis, the 12 approaches defined with industry involvement were applied in equal proportions to establish architectures in both academy and with industry collaboration. From the 7 approaches proposed only in academic context, most of them (5/7) were used to design academic architectures, 2 were applied with industry collaboration, and none of them was used to create industry architectures. Hence, while approaches defined with industry involvement have similarly impacted architecture from both only academic and with industry collaborations, approaches from academy have mostly reached academic architectures. An example is the architectural process presented in [70], which is the most used (in 6 reference architectures), but all those architectures were designed in academia.

4.5. Maturity of reference architectures

To answer RQ5 and determine the maturity of the 162 architectures found in this SMS, we collected how they were evaluated, classifying them in one of the six evaluation levels: **E0 -** reference architectures not assessed, **E1 -** reference architectures evaluated using toy examples; **E2 -** reference architectures qualitatively evaluated by experts; **E3 -** reference architectures assessed through controlled experiments in laboratory; **E4 -** reference architectures applied in industrial case studies; and **E5 -** reference architectures being used in industry. Maturity levels are presented in Figure 10, and listed in the last column of Table 5 (in Appendix A). The levels adopted in this work are similar to the technology assessment standards, such as Technology Readiness Levels (TRL) [61, 89] and SOA Maturity Model[12]; the former refers to a method by NASA and US Department of Defense (DoD) to assess the maturity level of a particular technology, while the latter provides guidance to measure progress and adoption of SOA. Hence, we believe that these evaluation levels could provide us an understanding of the level of maturity of reference architectures.

Classified as **E0**, 24% (39/162) of the reference architectures have no evaluation, being only conceptual proposals without application. The most of them (71.8%, 28/39) were designed only in academic context.

Almost the half part of the architectures (54.3%, 88/162) were minimally assessed. Toy examples, expert opinions, or controlled experiments were applied respectively in 32% (52/162), 16% (26/162), and 6.2% (10/162) of the reference architectures for evaluating their contained knowledge. These architectures were classified in the lowest maturity levels, namely **E1**, **E2**, and **E3**, and have been mainly proposed by academia (71.5%, 63/88) or through academia-industry collaborations (23.8%, 21/88).

The 21.6% (35/162) of reference architectures were classified in the two highest maturity levels. Specifically, 6.2% (10/162) and 15.4% (25/162) of architectures present, respectively, maturity levels **E4** and **E5**. The 77% (27/35) of mature architectures have been defined with industry involvement, i.e., 13 of them were proposed in only industry context while the 14 have been specified through collaboration with academia.

The great proportion of reference architectures proposed by academy (91.9%, 91/99) and through academy-industry

[12] http://soa.omg.org/Uploaded%20Docs/SOA/SOA_Maturity.pdf

Table 1
Approaches used to design reference architectures and their proponents (I = Industry, A = Academia, C = Academy-Industry Collaboration)

Approach	Year	Proponent	RA ID	RA context	RA type of system/domain
Approaches to design reference architectures					
Architecture derivation from other reference architectures [55]	1994	C	RA9	A	Robotics
FORM - Feature Oriented Reuse Method [50]	1998	C	RA16	A	SOA
Application of UML metamodel [12]	1999	C	RA11	C	Healthcare
Reference architecture derivation process based on existing systems [44]	2000	A	RA18	A	Web
Application of architectural style [20]	2000	I	RA29	A	Games
Application of architectural styles and description using ADL (Architectural Description Languages) [13]	2005	A	RA16	A	SOA
Empirically-grounded reference architectures [32]	2011	A	RA119	C	Big Data
Derivation of reference architecture from requirements, domain knowledge, and patterns [66]	2011	I	RA74	C	Business management
Angelov et al.'s framework [4]	2012	C	RA88	A	Cloud computing
HyperFlex Toolchain [40]	2014	A	RA97	A	Robotics
ProSA-RA - Process for reference architectures [70]	2014	A	RA33, RA64, RA65, RA76, RA87, RA131	A, A, A, A, A, A	Software environments, SOA, Self-adaptive systems, Ubiquitous computing, ICT
Bottom-up process from existing products in a domain and architectural styles application [60]	2015	A	RA101	A	Healthcare
Application of empirical methods [87]	2015	C	RA106	C	Transportation
Other approaches					
RM-ODP - The ISO Reference Model for Open Distributed Processing [28]	1995	C	RA15	C	Commerce
4+1 view model [58]	1995	I	RA69	A	Web
SAAM (Software Architecture Analysis Method) + commonality analysis [51, 97]	1998	C	RA104	A	Robotics
QADA - Quality-Driven Architecture Design and Quality Analysis Method [65]	2002	A	RA20	C	Communication
TOGAF (The Open Group Architecture Framework) + Archimate [49]	2009	I	RA83	C	Business management
ARM IoT - Architectural Reference Model for IoT [15]	2013	C	RA118	A	IoT
			RA124	A	Smart Cities

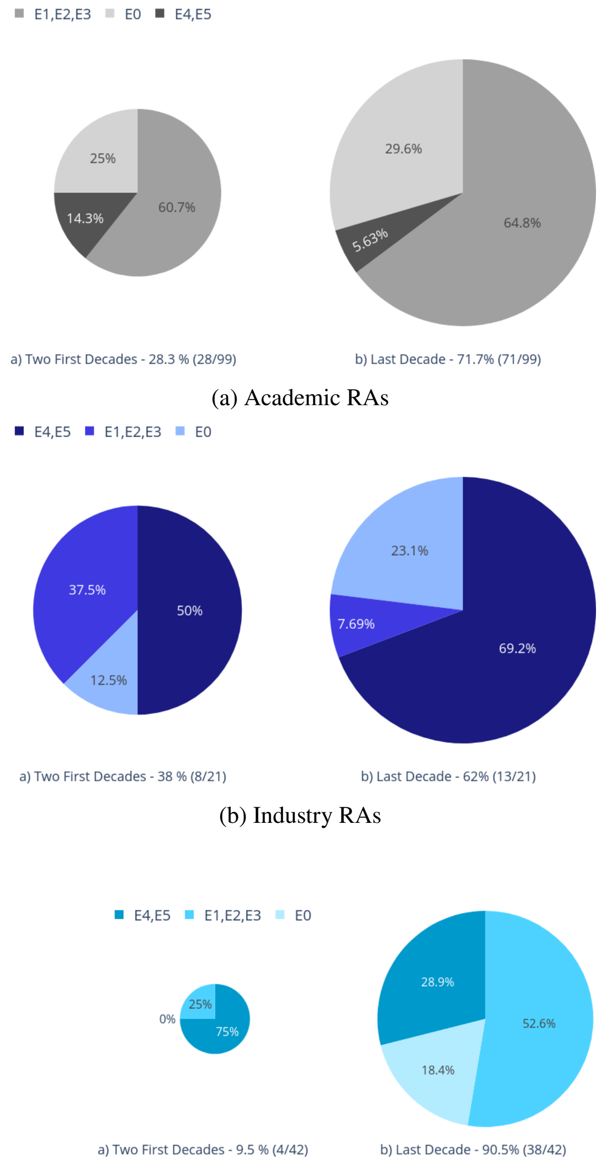
collaboration (66.7%, 28/42) were classified in the lowest maturity levels, i.e., E0 to E3. In comparison, the 62% (13/21) of those architectures made in only industry context were categorized in the highest levels of maturity, i.e., E4 and E5.

As listed in Tables 2, 3, and 4, some representative domains that have mature architectures are transportation, business management, healthcare, SOA, scientific systems, software environments, electrical, web, and robotics, which have at least two mature references architectures. Designed to different domains, mature architectures have mostly the perspective of being classical (indicated as C in Tables 2, 3, and 4) than preliminary (P). Industry architectures tend to provide more mature architectures for standardization (S) than facilitation (F) compared to academic ones.

We can observe in Figure 10 that 11 mature reference architectures existed until 2007 (i.e., in levels E4 and E5), from which 63.6% (7/11) were specified with industry involvement, the remaining 36.5% (4/11) architectures were defined only in academia. It is interesting to see that, during the decade from 2008 to 2018, industry has increased their contribution at proposing 83.3% (20/24) of mature architectures, whilst academia has decreased their participation with only 16.7% (4/24) mature architectures. Additionally, mature architectures made by academy-industry collaboration decreased during the last decade. Therefore, industry has been most focused than academia at proposing mature reference architectures. However, in a more general view, the proportion of all mature architectures (from industry, academia and through their collaboration) has decreased from 27.5% (11/40) in the two first decades to 19.7% (21/122) in the last decade. It seems that in last years both academia and industry have invested more efforts to pro-

Table 2
Mature reference architectures from industry.

Ref.	ID	Purpose	Perspective	Domain
[17]	RA12	F	C	Health-care
[7]	RA27	F	C	SOA
[85]	RA75	F	C	Militar
[46]	RA82	F	C	Human Computer Interaction
[101]	RA102	F	C	Industrial Plants
[77]	RA137	F	C	SOA
[8]	RA140	F	P	Scientific Systems
[30]	RA14	S	C	Transportation
[92]	RA35	S	C	Transportation
[10]	RA37	S	C	Business Management
[54]	RA39	S	C	Electrical
[82]	RA66	S	C	Component Based Systems
[59]	RA71	S	C	Cloud Computing
[9]	RA156	S	C	Transportation



(a) Academic RAs

(b) Industry RAs

(c) RAs from Academy-Industry Collaboration

Figure 10: Maturity of reference architectures over time

pose architectures with no assessment (*i.e.,* the proportion of level E0 architectures increased from 20%, 8/40, to 26.2%, 32/122) and with low maturity (*i.e.,* the amount of architectures of levels E1, E2, and E3 augmented from 52.5%, 21/40, to 55%, 67/122) than to assess the feasibility of such architectures in real scenarios.

5. Discussions of Results

This section highlights our main findings based on evidences obtained from answering the research questions provided in section 4.

Table 3
Mature reference architectures from academy-industry collaboration.

Ref.	ID	Purpose	Perspective	Domain
[11]	RA11	F	C	Health-care
[18]	RA28	F	C	Scientific Systems
[96]	RA50	F	C	Ambient Intelligence
[43]	RA80	F	C	Software Environments
[45]	RA81	F	C	Transportation
[16]	RA84	F	C	IoT
[26]	RA90	F	C	Embedded Systems
[87]	RA106	F	C	Transportation
[68]	RA119	F	P	Big Data
[74]	RA20	S	C	Communication
[93]	RA58	S	C	Transportation
[76]	RA67	S	C	Electrical
[75]	RA93	S	C	Business Management

Table 4
Mature reference architectures from academia

Ref.	ID	Purpose	Perspective	Domain
[44]	RA141	F	C	Web
[103]	RA09	F	C	Robotics
[41]	RA18	F	C	Web
[42]	RA19	F	P	Business Management
[62]	RA53	F	C	Robotics
[69]	RA64	S	C	Software Environments
[47]	RA134	S	C	Governmental
[27]	RA111	F	C	Business Management

5.1. Characterization of Reference Architectures from Academia

Reference architectures proposed entirely by academics represent the great proportion (i.e., 61% or 99/162) of architectures found in this study. The 71.7% (71/99) were constructed during the last decade, mainly for software environments, business management, and SOA systems. The 57,6% (57/99) of academic architectures were defined following a classical perspective. These architectures have mostly been proposed for facilitation (i.e., 81.8% or 81/99) than for standardization (i.e., 18.2% or 18/99) purposes. Moreover, they had more tended to support multiple organization (i.e., 63.6% or 63/99) than single organizations (i.e., 36.4% or 36/99). The proportion of academic architectures has decreased from 70% (28/40) to 58.2% (71/122) in the last decade.

The great proportion (i.e., 92% or 91/99) of academic architectures can be characterized as having low maturity, being assessed through experiments or toy examples. Only the 8% (8/99) of them have been evaluated in real scenarios. From the mature architectures defined by academics, the 75% (6/8) have facilitation purposes, and the 50% (4/8) were constructed by following architectural approaches. Moreover, the proportion of mature architectures made by academics have decreased during the last decade (i.e., 5.6% or 4/71) comparing with the two-first decades (i.e., 14.2% or 4/28). Mature architectures from academics were mainly proposed for web, robotics, and business management systems.

A few amount (i.e., 17.1% or 17/99) of academic architectures have followed approaches for constructing software architectures. The 88.2% (15/17) of them were defined for facilitation purposes. The application of approaches by academics have decreased over years. Until 2007 the 21.4%

(6/28) of academic reference architectures were defined through derivation from other more abstract architectural solutions or by applying approaches presented in [58, 97, 51]. During the decade from 2008 to 2018, only the 15.5% (11/71) of works followed approaches, which were more oriented to reference architectures engineering, as those detailed in [5, 70, 60, 15]. Additionally, from the 17 architectures made by using some approach only the 23.5% (4/17) of them are considered mature.

5.2. Characterization of Reference Architectures from Industry

Industry has contributed with the 12.9% (21/162) of reference architectures found in this mapping study. Most of them, i.e., the 66.6% (14/21), were proposed in last decade mainly for SOA, transportation, and healthcare systems. A vast amount (i.e., 76.2% or 16/21) of industrial architectures were defined following a classical perspective. An interesting finding is that industrial architectures have been unbiased regarding their purposes (i.e., facilitation or standardization) or context (i.e., single or multiple organizations). This is, we found quite the same amount of industrial reference architectures (i.e., 52.4% or 11/21) for each purpose and context. The architectures proportion from industry has decreased over years, passing from 20% (8/40), in the two-first decades, to 10.7% (13/122), in last decade.

It is possible to say that industry reference architectures can be characterized by its maturity, since more than the half (i.e., 62% or 13/21) were classified in the highest levels of maturity (i.e., E4 and E5). Mature architectures have been mostly defined from a classical perspective (i.e., 84.6% or 11/13). Moreover, the 63.6% (7/11) of architectures for standardization and the 54.6% (6/11) for facilitation purposes are considered mature. Similarly, the 60% (6/10) or these architectures for single organization and the 58.3% (7/12) for multiple organizations are also mature. The proportion of mature architectures from industry has increased from 50% (4/8), in two-first decades, to 69.2% (9/13) in last decade. Mature architectures from industry are more frequent to be found for transportation and SOA-based systems.

An unexpected result of this study is that all architectures made only by industry practitioners have been constructed in an ad-hoc way. After analysing these architectures we did not find evidence indicating the use of any approach for their construction.

5.3. Characterization of Reference Architectures from Industry-Academia Collaboration

Collaborations between industry and academia has contributed with the 25.9% (42/162) of reference architectures found in this systematic mapping study. Similarly, to architectures made by only academics or practitioners, most reference architectures from collaborations (i.e., 69% or 29/42) have been conceived followed a classical perspective. Practitioners and academics have joint efforts to propose architectures mostly for business management and transportation systems. The 66.7% (28/42) of these architectures have been defined for facilitation purposes. The 73.8% (31/42) of architectures from cooperation are oriented to be used in multiple organizations context. Collaborations between industry and academia for proposing reference architectures were scarce during the two-first decades, contributing with the 10% (4/40) of existing architectures by that time. Collaborations have significantly increased during last years, providing the 31.1% (38/122) of architectures published in last decade.

Reference architectures from collaborations present most maturity than academic architectures, but are less mature than industrial architectures. This is, the proportion of architectures from cooperation (i.e., 33.3% or 14/42) that were classified in highest maturity levels (i.e., E4 and E5) is greater if compared with the 8% (8/99) from academia; however, it is minor if compared with the proportion of mature architectures from industry (i.e., 62% or 13/21). Therefore, the industry involvement in reference architectures construction through collaboration did not guarantee the assessment of such architectures in industrial case studies or real scenarios. Mature architectures from those collaborations are frequently focused on transportation systems.

The 71.4% (10/14) of mature architectures from collaboration were defined from a classical perspective. We also observed that the 50% (7/14) of architectures for standardization purposes and the 54.55% (6/11) for single organization context are considered mature. Moreover, the 78.5% (11/14) of all mature architectures resulted form practitioners and academics cooperation were published in the last decade.

The tendency of industry-academia cooperation to apply approaches for constructing reference architectures is more similar to the tendency found in academic architectures than in industrial ones. For instance, whilst all industrial architectures were defined without using any approach, the 17.1% (17/99) of academic architectures used some approach for their construction. This proportion is quite similar to the 16.7% (7/42) of architectures from cooperation which were built making use of some approach, as those detailed in [12, 32, 66, 87, 28, 65, 49]. The proportion of approaches application has been decreased over decades. During the first-two decades the 75% (3/4) of architectures were defined by using some approach; however, this proportion has declined to 10.5% (4/38) in last decade.

5.4. Domains and Types of Systems Over Time

As showed in Section 4.2, the great amount of RAs has been focused on fifteen domains from a total of forty domains and systems found in this study. We compared the proportion of RAs for the most common domais and type of systems. It was possible to identify that during the first two decades SOA, Transportation, Component Based Systems, Software Environments, Healthcare, and Web were the domains for which more reference architectures were proposed, i.e., representing more than the half of existing architectures by that time. During the last decade, interesting changes regarding priority domains can be found. For instance, component based systems have not been anymore the focus for reference architectures and new relevant domains and types of systems have appeared into scene, namely, business management, electrical, ICT, ambient intelligence, cloud computing, governmental, scientific, military, and robotics systems. Therefore, SOA, transportation, software environments, healthcare, and web have been maintained as priority domains independently of time.

Moreover, we observed the interest on types of domains and systems are quite similar in industry, academia, and their collaborations. We found discrepancies only in two domains, i.e., transportation and software environments. During all time periods, industry has been involved in the construction of all reference architectures for transportation, whilst academia has had strong involvement in all architectures for software environments.

5.5. Characterization of Reference Architectures by Perspective

It was found a prevalence of classical reference architectures (i.e., 63% or 102/162) over preliminary ones (i.e., 37% or 60/162).

The 63.7% (65/102) of classical architectures have facilitation purposes and the 66.1% (43/65) of them have been proposed only by academics. This trend is similar in preliminary architectures, for which we found a predominance for architectures with facilitation purpose defined in academia (i.e., 90% or 54/60).

The 36.3% (37/102) of classical architectures have standardization purposes, from which the 62.1% (23/37) have been mainly defined by industry, or with practitioners involvement. This finding differs from preliminary architectures for standardization (i.e., 6/60), which were mainly proposed by academics.

The 73.5% (75/102) of classical architectures were classified in lowest levels of maturity (i.e., E0 to E3). The 78.6% (22/28) of classical and mature architectures were defined by practitioners or through their collaboration with academia. This scenario is similar for preliminary and mature architectures (i.e., 11.8% or 7/59), for which the 85.7% (6/7) had practitioners contributions.

Only the 17.6% (18/102) of classical architectures reported approaches adoption, mostly for facilitation purposes. This result is analogous to the 12.9% (7/54) of preliminary architectures constructed using some approach, all of them with facilitation goals. In both, classical and preliminary architectures, academics have led the approaches utilization.

Finally, no correlation was found between the reference architectures perspective, the approach used for their elaboration, and their maturity levels.

5.6. Characterization of Reference Architectures by Purpose

Reference architectures for facilitation (i.e., 73.4% or 119/162) have primed over those architectures for standardization (i.e., 26.6% or 43/162).

The proportion of facilitation architectures that follow classical (i.e., 54.6% or 65/119) or preliminary (i.e., 45.3% or 54/119) perspectives is quite similar, and both types have been defined in its majority by academics. The great amount of architectures for standardization (i.e., 83.7% or 36/43) were defined in a classic way, being that the 53.5% (23/43) of them were created through industry participation.

The 88% (22/25) of reference architectures built following an approach have facilitation purposes. Architectures for standardization, i.e., RA64, RA20, and RA118, were defined following ProSA-RA [73], QADA [65], and the ARM IoT [15], respectively.

The 64.7% (77/119) of architectures for facilitation have been built considering multiple organizations context. Such architectures have been frequently proposed for business management (i.e., 11.6% or 9/77), SOA (i.e., 9% or 7/77), software environments (7.8% or 6/77), ICT (7.8% or 6/77), and scientific systems (5.2% or 4/77), among others. The 67.4% (29/43) of architectures for standardization also support multiple organizations context. The 41.3% (12/29) of such architectures are focused on transportation, business management, and governmental systems, in equal proportions. Therefore, it is possible to observe that architectures for standardization are concentrated in a small set of domains, differently form facilitation architectures which are dispersed in a broader group of systems.

Finally, architectures for facilitation can be considered narrowly more mature than architectures for standardization, i.e., the 18.5% (22/119) of facilitation architectures are mature compared with the 7% (3/43) of mature architectures with standardization goals.

5.7. Characterization of Mature Reference Architectures

Only the 21.6% (35/162) of architectures found in this study were considered mature, i.e., they were assessed through industrial case studies or application in real industry scenarios. Mature architectures can be found for different perspectives (i.e., classical or preliminary) and purposes (i.e., facilitation or standardization). The 27.4% (28/102) of classical architectures are mature and most of them (i.e., 75% or 21/28) had industry involvement. The 11.6% (7/60) of preliminary architectures were considered as mature and the great amount of them (i.e., 85.7% or 6/7) had a strong participation of industry practitioners. The 30.2% (13/43) architectures for standardization are mature, being that the 84.6% (11/13) of them were developed making use of practitioners experience. On the contrary, for constructing the 18.33% (22/120) of mature architectures for facilitation, academia and industry had invested the same efforts, i.e., each part had contributed with the 27.3% (6/22) of these architectures, and the remaining 45.4% (10/22) was made through their collaboration.

Independently of mature reference architectures characterization all them have in common a strong cooperation with industry. In this perspective, practitioners involvement can bring important practical experience from the domain to orient the reference architecture development. Therefore, it is expected that industry participation enables the architectural teams to assess reference architectures in real industrial scenarios.

5.8. Approaches Impact on Reference Architectures Maturity

The 78.3% (127/162) of reference architectures found in this study have been created without following a systematic process or methods for their construction, despite the existence of approaches for this purpose, which have been mainly proposed by academics. This is also true in the majority of mature architectures, which the 77.1% (27/35) of them were designed through an *ad hoc* way. This scenario reveals a gap between researchers who have proposed theoretical contributions with design approaches and the practitioners audience.

Only the 22.8% (8/35) of mature reference architectures have applied some approach for their construction. Classical and mature architectures for facilitation have been defined through the use of derivation methods [55, 44], empirical methods [87], and metamodels [12]. Preliminary and mature architectures for facilitation followed empirical-grounded methods for reference architectures [33]. Classical and mature architectures for standardization have applied more systematic architecture-oriented processes and methods [65, 73].

One interesting finding of this mapping is the decline of approaches application for creating reference architectures by academia, industry, and their coalitions. It is important to conduct more empirical studies to find evidences for explaining this situation. Even though, considering we did not find evidences that support the use of approaches by industry, it is necessary to investigate how reference architectures are being designed by software industry practitioners.

It was not found any correlation between the use of methods and highest maturity levels. Therefore, it is not possible to define that the application of certain approach will support the achievement of upper maturity levels in reference architectures. In this perspective, the low maturity of architectures found in this study, can be more related with the lack of industry participation than by following systematic approaches for its construction.

Therefore, we can conclude there are still open issues regarding consolidating the area of reference architecture engineering with regard to diverse aspects and activities (e.g., analysis, design, evaluation, conformance checking, evolution) and, in particular, the practical adoption of such contributions to widely systematize the design, use, and evolution of these architectures mainly in the industry context.

6. Threats to Validity

The main threats to validity identified for this SMS, as a secondary study [2], are study selection validity, data validity, and research validity. In this section we explain how such threats were mitigated.

Study Selection Validity: The search string construction might impose threats to the relevancy of the primary studies identified To mitigate this threat we followed the guidelines in [52] to define the search string for secondary studies.

The search for reference architectures was conducted in six publication databases, namely IEEE Xplore, ACM DL, ScienceDirect, SpringerLink, Scopus, and Web of Science. According to [24] and [52], these publication databases are the most relevant sources in software engineering area. In addition, we wanted to be as inclusive as possible; thus, no limits were placed on date of publication and we avoided imposing restrictions (i.e., filters by title, abstract, and keywords) on the primary study selection. Aiming at not missing any important evidence, we also conducted the *snowball* technique [83] using the reference list of the selected primary studies. During the search, conference papers, journals articles, technical reports, and chapter of books were considered. However, despite of our effort to include all relevant evidence in this mapping, it is possible that academic reference architectures were missed, additionally to those propose in industry scenarios and that were not published in academic libraries.

Regarding *study inclusion/exclusion*, we aimed at ensuring an unbiased selection process, we defined research questions in advance, and devised inclusion and exclusion criteria. The protocol of this SMS was reviewed by all authors with experience at conducting secondary studies. We believe that the questions and criteria are detailed enough to provide an assessment of how the final set of primary studies was obtained. Moreover, aiming to increase the reliability of our study, each study was read by at least two researchers. When conflicts in the application of inclusion and exclusion criteria were presented, the study was reviewed by a third participant to make the final decision. However, it might be possible that studies proposing reference architectures were excluded in first stage they lack of important information in the title, abstract, keywords, introduction, and conclusions sections.

Data Validity: Another threat to this mapping refers to *data extracted* from the primary studies, since not all the information were obvious to answer the research questions and some data had to be interpreted. Moreover, in the event of a disagreement between reviewers, a discussion was conducted to ensure that a full agreement was reached. Moreover, data extracted was crosschecked to minimize researchers bias.

Regarding *bias of the classification schema*, to characterize the context and maturity of the selected reference architectures, we analyzed them based on aspects from existing frameworks such as the one proposed by Angelov et al. [4], and an updated them when necessary as the evaluation level from [61, 89].

Research Validity: To address *research method bias*, SMS was purposely selected as the research method in our investigation, and reasons for that are presented as follows. Secondary studies as systematic mapping studies and systematic literature reviews (SLRs) are well-know evidence-based research methods proposed by the Empirical Software Engineering area. The execution of an SMS allows researchers to identify and classify research related to a given software engineering topic, providing an overview of that topic [78, 2]. In particular, an SLR enhances the research synthesis making use of data from previously published studies in a research topic [2, 52, 53]. In our study, we selected SMS over SLR since we intended to provide an overview of existing reference architectures rather than comparing them regarding their relative merits, which is a goal of SLR [2].

Multivocal literature reviews (MLR) are a special type of SLR that includes as primary studies from both academic literature and gray literature (e.g., technical reports, preprints, and practitioners blogs). MLR make it possible to understand the state-of-the-art and practice by including evidence from practitioners that was not published in academic libraries. Considering the broader scope of MLRs comparing with SMSs and SLRs, recent guidelines reported that planning and conducting an SMS or SLR is be a fundamental step of the overall MLR process [39]. The main reason is because planning an MLR is not so different from planning SMS or SLR that follow well-known guidelines in [52, 78].

Concerning *generalizability* of this study, we are aware that, by considering studies published only in academic libraries or those recovered from snowballing implies a threat on results generalization, since we did not consider all possible reference architectures created by industry or those published in the gray literature, e.g., standards repositories. To mitigate this threat, we applied the snowballing technique [83, 99] on reference lists of included studies. With this, we identified ten reference architectures published in technical reports or gray literature. Moreover, all results reported in this SMS related with industry participation were based exclusively on evidences obtained from industrial architectures retrieved in academic data libraries. Henceforth, our intention is not to use the SMS results to generalize the software industry participation for creating reference architectures. To do so, this study's results could be further extended with evidence obtained from gray literature through the conduction of an MLR.

7. Conclusions

We performed an SMS on existing software reference architectures, published in scientific data libraries, to understand their application domains, main characteristics (i.e., perspective, purpose, and context), design issues, and maturity. Additionally, we investigated the involvement of academia and the software industry and their collaborations at constructing these architectures.

Academia first started to invest in designing reference architectures. Academic reference architectures have appeared in 34 out of the 40 domains and types of systems identified in this study. Academic architectures have predominated for software environments, SOA, education, and robotics systems. Almost two third of the reference architectures proposed by academics are characterized by their classical perspective and facilitation purpose, since they have promoted the sharing of concepts, software elements and configurations, and decisions (e.g., architectural patterns, styles, tactics, and technologies) that can be reused to design new software systems in a specific domain. Academic architectures have facilitated such reuse in existing (or consolidated) domains, as well as in new domains in which many uncertainties exist. Considering the perspective of reference architectures, those preliminary ones are mostly led by academia, with 70% (42/60) of participation, offering the fundamental building blocks for innovative software systems, for instance, RA88 for cloud computing systems. Only a small part of academic architectures (*i.e.,* 8% or 8/99) can be considered as mature and the remainder ones possibly could present difficulties to be used as an asset in real software projects. Most of them were mainly assessed through the use of toy examples (or prototyping), expert opinions (using surveys), and controlled experiments in laboratory contexts. Considering that academia had invested important efforts during three decades to construct reference architectures, it was expected a more impact of their architectures in real projects; however, most academic architectures lack of evidences to ensure their viability in software development projects. Therefore, in practice, academic architectures are more adequate to obtain a first understanding about domain terminology, possible operations of software systems, and (if available) reasonable arrangements for systems' internal structures.

Industry has increasingly involved in the creation and dissemination of reference architectures, with main focus on application domains where large, complex systems are required, such as transportation, health-care, electrical, and agriculture. Industry has focused on classical architectures for standardization, abstracting best architectural practices from existing software systems in a domain, and standardizing such practices to guide architectures of new systems in the sector. Differently from academia, a good part of industry reference architectures (i.e., 62% or 13/21) can be considered mature, since many of them were defined to enhance software development processes in real organizations.

Whilst the proportion of academic and industrial architectures has decreased in last decade, the proportion of architectures from their collaboration has augmented, which can demonstrate the necessity of forming multidisciplinary teams to address challenges associated to the software industry market. Despite the increase of architectures from cooperation, their maturity is minor compared with industrial architectures. Moreover, the proportion of mature architectures from collaboration has decreased in last decade whilst the proportion of mature architectures from industry has augmented. Architectures from collaboration have tend to be constructed (by using approaches for it) and assessed (by not fully evaluating them in industry case studies or real scenarios) in similar ways than academics architectures, but application domains are more related with industry needs, i.e., transportation.

Based on evidences obtained from studies analyzed in this mapping, it was not possible to identify practices for constructing mature reference architectures for specific purposes (i.e., facilitation or standardization) and perspectives (e.g., classical or preliminary). Obtaining empirical evidences to understand, for instance, how reference architectures are being engineered by the software industry could be of great benefit to the software architecture community. Such investigation could be inspired by the work in [48] that proposes a generic process to analyse, synthesize, and evaluate software architectures. In the same line of research, it could be interesting to gather evidence from industry to better understand factors associated to reference architectures maturity, longevity, and sustainability. For instance, AUTOSAR (RA156), IIRA (RA153), S3 (RA27), and Mars Design Reference Architecture (RA154) could be used as case study to investigate how such architectures have been managed to deal with modifications throughout software development projects life cycle.

We are aware of the necessity to complement this investigation with evidences from gray literature, i.e., studies published in different sources than academic data libraries. Possible information sources could be standards repositories of organizations as NIST, IEEE SA, or ISO/IEC, which contain important reference architectures resulting from consortium collaboration, for instance, [98]. Based on such evidence, we consider it could be possible to better characterize, for instance, standardization and the industry impact on mature architectures, as well the approaches used by industry, which were not identified in this SMS.

Although we found that there are several references architectures in a plethora of diverse domains, we consider there is still an amount of application domains and types of systems that do not present reference architectures. Some examples are those that require software systems with dynamic architectures, what could include domains as smart-* (building, grid, farms), autonomous vehicles, airport systems, and so on. Even for domains or types of systems that have benefited from reference architectures, there are subdomains still not covered. An example is the software engineering area with 12 architectures, but lacking of architectures to support the design and integration of tools in many software engineering activities, such as evolution, maintainability, project management, among others. For designing all these new architectures, derivation from existing ones could be adopted, intending to reuse well established experience even from other domains or types of systems.

Acknowledgments

The work herein presented has been supported by the Brazilian funding agencies FAPESP (São Paulo Research Foundation) Grants No. 2018/07437-9 and 2017/06195-9 and the CNPq (National Council for Scientific and Technological Development) Grants No. 312634/ 2018-8 and 141602/2017-1, the GENESIS project under the National Spanish Program for Research Aimed at the Challenges of Society (RETOS 2016, contract TIN2016-79269-R), and the "Beatriz Galindo" Spanish Program BEAGAL18/00064.

Supplementary Material

Supplementary material associated with this article can be found in [35].

CRediT authorship contribution statement

Lina Garcés: Conceptualization, Data curation, Methodology, Formal analysis, Validation, Investigation, Visualization, Writing - original draft, Writing - review & editing.. **Silverio Martínez-Fernández:** Conceptualization, Methodology, Formal analysis, Validation, Investigation, Writing - review & editing. **Lucas Oliveira:** Conceptualization, Methodology, Formal analysis, Validation, Investigation, Writing - review. **Pedro Valle:** Investigation, Formal analysis, Validation. **Claudia Ayala:** Methodology, Writing - review. **Xavier Franch:** Conceptualization, Methodology, Writing - review. **Elisa Yumi Nakagawa:** Conceptualization, Methodology, Writing - review & editing.

References

[1] Affonso, F., Scannavino, K., Oliveira, L., Nakagawa, E., 2014. Reference architectures for self-managed software systems: A systematic literature review, pp. 21–31. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84946689432&doi=10.1109%2fSBCARS.2014.18&partnerID=40&md5=6aff0b1ae488bbd38a7ad35762cee941, doi:10.1109/SBCARS.2014.18. cited By 4.

[2] Ampatzoglou, A., Bibi, S., Avgeriou, P., Verbeek, M., Chatzigeorgiou, A., 2019. Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. Information and Software Technology 106, 201 – 230. URL: http://www.sciencedirect.com/science/article/pii/S0950584918302106, doi:https://doi.org/10.1016/j.infsof.2018.10.006.

[3] Ananthavijayan, R., Shanmugam, P., Padmanaban, S., Holm-Nielsen, J., Blaabjerg, F., Fedak, V., 2019. Software architectures for smart grid system-a bibliographical survey. Energies 12. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85063749412&doi=10.3390%2fen12061183&partnerID=40&md5=2dbb2e66a87704b51130f22e0a2ba9f2, doi:10.3390/en12061183. cited By 1.

[4] Angelov, S., Grefen, P., Greefhorst, D., 2012a. A framework for analysis and design of software reference architectures. Information and Software Technology 54, 417–431. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84856028851&partnerID=40&md5=28a47975f8cd8ffbdb8b57787164db8f.

[5] Angelov, S., Grefen, P., Greefhorst, D., 2012b. A framework for analysis and design of software reference architectures. Information and Software Technology 54, 417–431.

[6] Angelov, S., Trienekens, J., Kusters, R., 2013. Software reference architectures - exploring their usage and design in practice. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7957 LNCS, 17–24. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84879862842&partnerID=40&md5=99adc6633c279f90bf46921b6f881400.

[7] Arsanjani, A., Zhang, L.J., Ellis, M., Allam, A., Channabasavaiah, K., 2007. S3: A service-oriented reference architecture. IT Professional 9, 10–17. doi:10.1109/MITP.2007.53.

[8] Astekin, M., Sozer, H., 2012. Utilizing clone detection for domain analysis of simulation systems, in: Joint Working IEEE/IFIP Conference on Software Architecture (WICSA) and European Conference on Software Architecture (ECSA), pp. 287–291.

[9] AUTOSAR, 2018. Automotive open system architecture, in: In http://www.autosar.org/, Access in 08/10/2018.

[10] Bahuguna, A., 2008. Reference architecture for enterprise batch processing of information, in: Third International Conference on Digital Information Management (ICDIM), pp. 8–17. doi:10.1109/ICDIM.2008.4746803.

[11] Barber, K., Graser, T., Silva, J., 2001. Developing a traceable

domain reference architecture to support clinical trials at the national cancer institute an experience report, in: Proceedings of Eighth Annual IEEE International Conference and Workshop On the Engineering of Computer-Based Systems (ECBS), pp. 144–151. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-0034970227&partnerID=40&md5=b605012047f1c9afe8cfb74a7c307a86. cited By (since 1996)2.

[12] Barber, K.S., Graser, T.J., 1999. Representing Domain Reference Architectures by Extending the UML Metamodel. Technical Report TR99-UT-LIPS-SEPA-01. The University of Texas at Austin.

[13] Bashroush, R., Spence, I., Kilpatrick, P., Brown, T., 2005. A generic reference software architecture for load balancing over mirrored web servers: Nasr case study, in: 3rd ACS/IEEE International Conference on Computer Systems and Applications, 2005, pp. 743–746. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-11144233361&partnerID=40&md5=5ac5aa4bde7ec669c4744482f3bb9ac4. cited By (since 1996)1.

[14] Bass, L., Clements, P., Kazman, R., 2003. Software Architecture in Practice. 2 ed., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[15] Bassi, A., Bauer, M., Fiedler, M., Kramp, T., van Kranenburg, R., Lange, S., Meissner, S. (Eds.), 2013. Enabling Things to Talk Designing IoT solutions with the IoT Architectural Reference Model. Springer.

[16] Bauer, M., Boussard, M., Bui, N., Loof, J., Magerkurth, C., Meissner, S., Nettsträter, A., Stefa, J., Thoma, M., Walewski, J., 2013. Iot reference architecture, in: Enabling Things to Talk. URL: http://link.springer.com/chapter/10.1007/978-3-642-40403-0_8, doi:10.1007/978-3-642-40403-0_8.

[17] Campos, F., Mills III, W., Graves, M., 2002. A reference architecture for remote diagnostics and prognostics applications, in: Proceedings of IEEE AUTOTESTCON, pp. 842–854. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-0036443164&partnerID=40&md5=ce62eab8a26944c8d16b493e298bc1c8. cited By (since 1996)5.

[18] Candela, L., Castelli, D., Pagano, P., 2007. A reference architecture for digital library systems: Principles and applications, in: Proceedings of the 1st International Conference on Digital Libraries: Research and Development, Springer-Verlag, Berlin, Heidelberg. pp. 22–35. URL: http://dl.acm.org/citation.cfm?id=1782334.1782339.

[19] Cloutier, R., Muller, G., Verma, D., Nilchiani, R., Hole, E., Bone, M., 2010. The concept of reference architectures. Systems Engineering 13, 14–27.

[20] Collins-Cope, M., Matthews, H., 2001. A reference architecture for component based development, in: Patel, D., Choudhury, I., Patel, S., de Cesare, S. (Eds.), OOIS 2000, Springer London, London. pp. 225–237.

[21] Cruzes, D.S., Dyba, T., 2011. Recommended steps for thematic synthesis in software engineering, in: 2011 international symposium on empirical software engineering and measurement, IEEE. pp. 275–284.

[22] De Oliveira, L., Romero Felizardo, K., Feitosa, D., Nakagawa, E., 2010. Reference models and reference architectures based on service-oriented architecture: A systematic review. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 6285 LNCS, 360–367. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-78049330800&doi=10.1007%2f978-3-642-15114-9_29&partnerID=40&md5=5fdaa3b90f510b83fb92c12b439847f2, doi:10.1007/978-3-642-15114-9_29. cited By 12.

[23] Dobrica, L., Niemela, E., 2008. An approach to reference architecture design for different domains of embedded systems, in: SERP 2008, Las Vegas, USA. pp. 287–293.

[24] Dyba, T., Dingsoyr, T., Hanssen, G.K., 2007. Applying systematic reviews to diverse study types: An experience report, in: First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007), pp. 225–234. doi:10.1109/ESEM.2007.59.

[25] EIRA, 2018. European interoperability reference architecture, in: https://joinup.ec.europa.eu/solution/eira/, Access in 08/10/2018.

[26] Eklund, U., Bosch, J., 2014. Architecture for embedded open software ecosystems. Journal of Systems and Software 92, 128–142. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84898803901&partnerID=40&md5=59a86edc1ffd1212224e0f752557b204. cited By (since 1996)0.

[27] Evers, S., Ernsting, J., Majchrzak, T.A., 2016. Towards a reference architecture for model-driven business apps, in: 2016 49th Hawaii International Conference on System Sciences (HICSS), pp. 5731–5740. doi:10.1109/HICSS.2016.708. iEEE.

[28] Farooqui, K., Logrippo, L., de Meer, J., 1995. The iso reference model for open distributed processing: an introduction. Computer Networks and ISDN Systems 27, 1215 – 1229. URL: http://www.sciencedirect.com/science/article/pii/016975529500087N, doi:https://doi.org/10.1016/0169-7552(95)00087-N. iSO Reference Model for Open Distributed Processing.

[29] Felizardo, K., Nakagawa, E., Fabbri, S., Ferrari, F., 2017. Revisão sistemática da literatura em Engenharia de Software: teoria e prática. 1 ed., Elsevier.

[30] Followell, D., Gilbertson, D., Keller, K., 2004. Implications of an open system approach to vehicle health management, in: Proceedings of IEEE Aerospace Conference, pp. 3717–3724 Vol.6. doi:10.1109/AERO.2004.1368189.

[31] Galster, M., Angelov, S., Martínez-Fernández, S., Tofan, D., 2017. Reference architectures and scrum: friends or foes?, in: Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ACM. pp. 896–901.

[32] Galster, M., Avgeriou, P., 2011. Empirically-grounded reference architectures: A proposal, pp. 153–157. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-79960522929&partnerID=40&md5=8a758acbf58b5fbfaa431a508b2ac67b.

[33] Galster, M., Avgeriou, P., Weyns, D., Mannisto, T., 2011. Empirically-grounded reference architectures: A proposal, in: QoSA 2011, Boulder, USA. pp. 153–157.

[34] Garcés;, L., Martínez-Fernández;, S., Neto;, V., Nakagawa, E.Y., 2020. Architectural solutions for self-adaptive systems. Computer , 1–9doi:10.1109/MC.2020.3017574.

[35] Garcés, L., Martínez-Fernández, S., Oliveira, L., Valle, P., Ayala, C., Franch, X., Nakagawa, E., 2021. Dataset on existing software reference architectures. datacd. Mendeley Data, v1. URL: https://data.mendeley.com/datasets/7tv49bs4kf/draft?a=0e5b2276-570c-407f-b77e-8c4d5eb231b0, doi:10.17632/7tv49bs4kf.1[DOIisreservedbutnotactive].

[36] Garcés, L., Nakagawa, E.Y., 2017. A process to establish, model and validate missions of systems-of-systems in reference architectures, in: Proceedings of the Symposium on Applied Computing, ACM, New York, NY, USA. pp. 1765–1772. URL: http://doi.acm.org/10.1145/3019612.3019799, doi:10.1145/3019612.3019799.

[37] Garcés, L., Oquendo, F., Nakagawa, E., 2020. Assessment of reference architectures and reference models for ambient assisted living systems: Results of a systematic literature review. International Journal of E-Health and Medical Communications 11, 17–36. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85074350385&doi=10.4018%2fIJEHMC.2020010102&partnerID=40&md5=27e12ba6eb8ca985ebdbbc6b4c0eb6de, doi:10.4018/IJEHMC.2020010102. cited By 0.

[38] Garland, J., Anthony, R., 2003. Large-Scale Software Architecture. A practical guide using UML. 1 ed., John Wiley and Sons Ltd., England.

[39] Garousi, V., Felderer, M., Mäntylä, M.V., 2019. Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. Information and Software Technology 106, 101 – 121. URL: http://www.sciencedirect.com/science/article/pii/S0950584918301939, doi:https://doi.org/10.1016/j.infsof.2018.09.006.

[40] Gherardi, L., Brugali, D., 2014. Modeling and reusing robotic software architectures: The hyperflex toolchain, pp. 6414–6420. Cited

By 1.

[41] Grosskurth, A., Godfrey, M., 2005. A reference architecture for web browsers, pp. 661–664. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-33646906217&partnerID=40&md5=fb63dc9e087283ca33508a0b923a4dc2. cited By (since 1996)18.

[42] Gruhn, V., Weber, T., 2005. From an e-business revenue model to its software reference architecture. IFIP Advances in Information and Communication Technology 189, 33–47. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84884397907&partnerID=40&md5=99c018687a6d86bbd93435812148b7ed. cited By (since 1996)0.

[43] Guerra, E., Alves, F., Kulesza, U., Fernandes, C., 2013. A reference architecture for organizing the internal structure of metadata-based frameworks. Journal of Systems and Software 86, 1239–1256. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84875235796&partnerID=40&md5=aed766d68a9330fe2ff22cd52cdf3893. cited By (since 1996)1.

[44] Hassan, A., Holt, R., 2000. A reference architecture for web servers, in: Reverse Engineering, 2000. Proceedings. Seventh Working Conference on, pp. 150–159. doi:10.1109/WCRE.2000.891462.

[45] Heisey, C., Hendrickson, A., Chludzinski, B., Cole, R., Ford, M., Herbek, L., Ljungberg, M., Magdum, Z., Marquis, D., Mezhirov, A., Pennell, J., Roe, T., Weinert, A., 2013. A reference software architecture to support unmanned aircraft integration in the national airspace system. Journal of Intelligent and Robotic Systems: Theory and Applications 69, 41–55. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84893682314&partnerID=40&md5=197f37a17e92d313e1d513cbdb6caa2e. cited By (since 1996)0.

[46] Heredero, G., Penmetsa, H., Agrawal, V., Shastri, L., 2013. Activity context-aware system architecture for intelligent natural speech based interfaces, pp. 21–35. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84898880934&partnerID=40&md5=53b71536c4b749d89ab306b97022e17f. cited By (since 1996)0.

[47] Herold, S., Mair, M., Rausch, A., Schindler, I., 2013. Checking conformance with reference architectures: A case study, in: 17th IEEE International Enterprise Distributed Object Computing Conference, pp. 71–80. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84892538751&partnerID=40&md5=80e43e761ca856d6939a1942d9e639d9.

[48] Hofmeister, C., Kruchten, P., Nord, R.L., Obbink, H., Ran, A., America, P., 2007. A general model of software architecture design derived from five industrial approaches. Journal of Systems and Software 80, 106–126.

[49] Jonkers;, H., Proper;, E., Turner, M., 2009. TOGAF and ArchiMate: A Future Together. A Vision for Convergence and Co-Existence. Technical Report. The Open Group.

[50] Kang, K., Kim, S., Lee, J., Kim, K., Shin, E., Huh, M., 1998. Form: A feature-oriented reuse method with domain-specific reference architectures. Annals of Software Engineering 5, 143–168. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-0032343158&partnerID=40&md5=25911c71c1005ebd05e8284a4a79806d.

[51] Kazman;, R., Bass;, L., Abowd;, G., Webb, M., 1994. Saam: a method for analyzing the properties of software architectures, in: Proceedings of 16th International Conference on Software Engineering, pp. 81–90. doi:10.1109/ICSE.1994.296768.

[52] Kitchenham, B., Charters, S., 2007. Guidelines for performing systematic literature reviews in software engineering.

[53] Kitchenham, B.A., Budgen, D., Brereton, P., 2015. Evidence-based software engineering and systematic reviews. volume 4. CRC press.

[54] Kling, M., Mitchener, M., Meyer, B., Jones, D., 2008. An implementation of the government reference architecture waveform developer and system integrator roles, in: IEEE Military Communications Conference (MILCOM) 2008. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-62349104806&partnerID=40&md5=79e4ceac55fddb386def40f292895c3b. cited By (since 1996)0.

[55] Kramer, T., Senehi, M.K., Michaloski, J., Ray, S., Rippey, W., Wallace, S., Quintero, R., Albus, J., 1994. A reference architecture for control of mechanical systems, in: Tutorial and Workshop on Systems Engineering of Computer-Based Systems, 1994., Proceedings., 1994, pp. 104–110.

[56] Kruchten, P., 2000. The Rational Unified Process: An Introduction. 2 ed., The Addison- Wesley Object Technology Series, Addison-Wesley.

[57] Kruchten, P., Obbink, H., Stafford, J., 2006. The past, present, and future for software architecture 23, 22–30.

[58] Kruchten, P.B., 1995. The 4+1 view model of architecture. IEEE Software 12, 42–50. doi:10.1109/52.469759.

[59] Liu, J., Zhang, L.J., Hu, B., He, K., 2012. Ccra: Cloud computing reference architecture, in: EEE Ninth International Conference on Services Computing (SCC), pp. 657–665. doi:10.1109/SCC.2012.110.

[60] Losavio, F., Ordaz, O., 2015. Quality-based heuristic for optimal product derivation in software product lines, in: Internet Technologies and Applications (ITA), 2015, pp. 125–131.

[61] Mankins, J.C., 2009. Technology readiness assessments: A retrospective. Acta Astronautica 65, 1216–1223.

[62] Martinez-Barbera, H., Herrero-Perez, D., 2010. Programming multirobot applications using the thinkingcap-ii java framework. Advanced Engineering Informatics 24, 62 – 75. URL: http://www.sciencedirect.com/science/article/pii/S1474034609000536, doi:http://dx.doi.org/10.1016/j.aei.2009.08.011. informatics for cognitive robots.

[63] Martínez-Fernández, S., Ayala, C.P., Franch, X., Marques, H.M., 2017. Benefits and drawbacks of software reference architectures: A case study. Information and software technology 88, 37–52.

[64] Martinez-Fernandez, S., Medeiros Dos Santos, P.S., Ayala, C.P., Franch, X., Travassos, G.H., 2015. Aggregating empirical evidence about the benefits and drawbacks of software reference architectures, in: 2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 1–10. doi:10.1109/ESEM.2015.7321184.

[65] Matinlassi;, M., Niemela;, E., Dobrica, L., 2002. Quality-Driven Architecture Design and Quality Analysis Method, A Revolutionary Initiation Approach to a Product Line Architecture. Technical Report. VTT Publications 456. VTT Technical Research Centre of Finland, Espoo.

[66] Miksovic, C., Zimmermann, O., 2011. Architecturally significant requirements, reference architecture, and metamodel for knowledge management in information technology services, in: 9th Working IEEE/IFIP Conference on Software Architecture (WICSA), pp. 270–279.

[67] Muccini, H., Moghaddam, M., 2018. Iot architectural styles: A systematic mapping study. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 11048 LNCS, 68–85. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85057261227&doi=10.1007%2f978-3-030-00761-4_5&partnerID=40&md5=cc58e9a786537b1c2e6057a69e2cfa2b, doi:10.1007/978-3-030-00761-4_5. cited By 3.

[68] Nadal, S., Herrero, V., Romero, O., Abell, A., Franch, X., Vansummeren, S., Valerio, D., 2017. A software reference architecture for semantic-aware big data systems. Information and Software Technology 90, 75 – 92. Science Direct.

[69] Nakagawa, E., Ferrari, F., Sasaki, M., Maldonado, J., 2011a. An aspect-oriented reference architecture for software engineering environments. Journal of Systems and Software 84, 1670–1684. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-79960844661&partnerID=40&md5=944e0d3f4b3f2dbc60378c1c9f270ed7. cited By (since 1996)9.

[70] Nakagawa, E., Guessi, M., Maldonado, J., Feitosa, D., Oquendo, F., 2014a. Consolidating a process for the design, representation, and evaluation of reference architectures, in: Software Architecture (WICSA), 2014 IEEE/IFIP Conference on, pp. 143–152. doi:10.

1109/WICSA.2014.25.

[71] Nakagawa, E.Y., Antonino, P.O., Schnicke, F., Capilla, R., Kuhn, T., Liggesmeyer, P., 2021. Industry 4.0 reference architectures: State of the art and future trends. Computers & Industrial Engineering 156, 107241. URL: https://www.sciencedirect.com/science/article/pii/S0360835221001455, doi:https://doi.org/10.1016/j.cie.2021.107241.

[72] Nakagawa, E.Y., Ferrari, F., Sasaki, M., Maldonado, J.C., 2011b. An aspect-oriented reference architecture for software engineering environments. Journal of Systems and Software 84, 1670–1684.

[73] Nakagawa, E.Y., Guessi, M., Feitosa, F., Oquendo, F., Maldonado, J.C., 2014b. Consolidating a process for the design, representation, and evaluation of reference architectures, in: WICSA 2014, Sydney, Australia. pp. 143–152.

[74] Niemela, E., Kalaoja, J., Lago, P., 2005. Toward an architectural knowledge base for wireless service engineering. IEEE Transactions on Software Engineering 31, 361–379.

[75] Norta, A., Grefen, P., Narendra, N., 2014. A reference architecture for managing dynamic inter-organizational business processes. Data and Knowledge Engineering 91, 52–89. Cited By (since 1996)0.

[76] Nuseibeh, T.T.G.B.B., 2011. A reference architecture for consumer electronics products and its application in requirements engineering, in: Relating Software Requirements and Architectures. URL: http://link.springer.com/chapter/10.1007/978-3-642-21001-3_13, doi:10.1007/978-3-642-21001-3_13.

[77] Olaf Zimmermann, Petra Kopp, S.P., 2009. Architectural knowledge in an soa infrastructure reference architecture, in: Software Architecture Knowledge Management. URL: http://link.springer.com/chapter/10.1007/978-3-642-02374-3_12, doi:10.1007/978-3-642-02374-3_12.

[78] Petersen, K., Vakkalanka, S., Kuzniarz, L., 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology 64, 1 – 18. URL: http://www.sciencedirect.com/science/article/pii/S0950584915000646, doi:https://doi.org/10.1016/j.infsof.2015.03.007.

[79] Qin, W., Chen, S., Peng, M., 2019. Recent advances in industrial internet: Insights and challenges. Digital Communications and Networks URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85070560118&doi=10.1016%2fj.dcan.2019.07.001&partnerID=40&md5=b0ee4d45fca6cc53a7c2fe4268f7f5c8, doi:10.1016/j.dcan.2019.07.001. cited By 0.

[80] Qureshi, N., Usman, M., Ikram, N., 2013. Evidence in software architecture, a systematic literature review, in: Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, ACM, New York, NY, USA. pp. 97–106. URL: http://doi.acm.org/10.1145/2460999.2461014, doi:10.1145/2460999.2461014.

[81] Reed, P., 2002. Reference Architecture: The best of best practices. Online. http://www.ibm.com/developerworks/rational/library/2774.html. Accessed in May 14th, 2014.

[82] Riddick, F., Kibira, D., Tina Lee, Y., Balakirsky, S., 2011. A component-based approach for manufacturing simulation, pp. 54–61. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-84878025789&partnerID=40&md5=cf43813ae773cd89d5fbc286d7e712bc. cited By (since 1996)0.

[83] Ridley, D., 2012. The Literature Review: A Step-by-Step Guide for Students. SAGE Study Skills Series. 2nd edition. ed., SAGE Publications Ltd.

[84] Rohling, A., Graciano Neto, V., Ferreira, M., Santos, W., Nakagawa, E.Y., 2019. A reference architecture for satellite control systems. Innovations in Systems and Software Engineering 15, 139–153.

[85] Satake, H., Skutt, T., Sherman, M., Eagleson, W., Rittenbach, T., Sepka, T., 2013. Government reference architecture extensions for application to base stations, in: Military Communications Conference, MILCOM 2013 - 2013 IEEE, pp. 1762–1767. doi:10.1109/MILCOM.2013.298.

[86] Savold, R., Dagher, N., Frazier, P., McCallam, D., 2017. Architecting cyber defense: A survey of the leading cyber reference architectures and frameworks, pp. 127–138. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85028679819&doi=10.1109%2fCSCloud.2017.37&partnerID=40&md5=210989bd7b4600b2017254febd8f24e8, doi:10.1109/CSCloud.2017.37. cited By 1.

[87] Schroeder, J., Holzner, D., Berger, C., Hoel, C.J., Laine, L., Magnusson, A., 2015. Design and evaluation of a customizable multi-domain reference architecture on top of product lines of self-driving heavy vehicles: An industrial case study, in: Proceedings of the 37th International Conference on Software Engineering - Volume 2, IEEE Press, Piscataway, NJ, USA. pp. 189–198.

[88] Sena, B., Allian, A., Nakagawa, E., 2017. Characterizing big data sofware architectures: A systematic mapping study. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85037811319&doi=10.1145%2f3132498.3132510&partnerID=40&md5=4c3cf49851259886bf48b59c058ffef7, doi:10.1145/3132498.3132510. cited By 0.

[89] Shishko, R., E.D.H., Fox, G., 2004. Nasa technology assessment using real options valuation. Systems Engineering 7, 1–13.

[90] Sittón-Candanedo, I., Alonso, R., Rodríguez-González, S., García Coria, J., De La Prieta, F., 2020. Edge computing architectures in industry 4.0: A general survey and comparison. Advances in Intelligent Systems and Computing 950, 121–131. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85065921255&doi=10.1007%2f978-3-030-20055-8_12&partnerID=40&md5=90b2d6ea943b97751b56a6d752a36cc3, doi:10.1007/978-3-030-20055-8_12. cited By 1.

[91] Springer Fachmedien Wiesbaden, 2013. Autosar — the worldwide automotive standard for e/e systems. ATZextra worldwide 18, 5–12. URL: https://doi.org/10.1007/s40111-013-0003-5, doi:10.1007/s40111-013-0003-5.

[92] Tamblyn, S., Hinkel, H., Saley, D., 2007. National aeronautics and space administration (nasa) crew exploration vehicle (cev) reference guidance, navigation, and control (gn&c) architecture, pp. 587–605. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-38549111873&partnerID=40&md5=0bbdeef7946652617983d62c49376cdd. cited By (since 1996)0.

[93] Terraillon, J.L., Jung, A., Arberet, P., Montenegro, S., Rossignol, A., Garcia, G., Li, J., Rodriguez, A., Mazzini, S., Hougaard, P., Fowell, S., Ferraguto, M., Panunzio, M., 2010. Space on-board software reference architecture. URL: http://www.scopus.com/inward/record.url?eid=2-s2.0-79955390509&partnerID=40&md5=13067fcf89bfabcdd698e0acdf5c8193. cited By (since 1996)0.

[94] The Object Management Group, 2021. Common Object Request Broker Architecture (CORBA). techreport 3.4. URL: https://www.omg.org/spec/CORBA/.

[95] Volk, M., Bosse, S., Bischoff, D., Turowski, K., 2019. Decision-support for selecting big data reference architectures. Lecture Notes in Business Information Processing 353, 3–17. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85068105027&doi=10.1007%2f978-3-030-20485-3_1&partnerID=40&md5=1c49daf615790f598d368e0ee740d7e8, doi:10.1007/978-3-030-20485-3_1. cited By 0.

[96] Waibel, A., Stiefelhagen, R., 2009. Computers in the Human Interaction Loop. 1st ed., Springer Publishing Company, Incorporated. URL: http://link.springer.com/chapter/10.1007/978-1-84882-054-8_22.

[97] Weiss, D.M., 1998. Commonality analysis: A systematic process for defining families, in: van der Linden, F. (Ed.), Development and Evolution of Software Architectures for Product Families, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 214–222.

[98] Wo L. Chang, David Boyd, O.L., 2019. NIST Big Data Interoperability Framework: Volume 6, Reference Architecture. techreport Special Publication (NIST SP) - 1500-6r2. NIST. doi:https://doi.org/10.6028/NIST.SP.1500-6r2.

[99] Wohlin, C., 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering, in: Proceedings of

the 18th International Conference on Evaluation and Assessment in Software Engineering, Association for Computing Machinery, New York, NY, USA. URL: https://doi.org/10.1145/2601248.2601268, doi:10.1145/2601248.2601268.

[100] Wohlin, C., Runeson, P., da Mota Silveira Neto, P.A., Engström, E., do Carmo Machado, I., de Almeida, E.S., 2013. On the reliability of mapping studies in software engineering. Journal of Systems and Software 86, 2594 – 2610. URL: http://www.sciencedirect.com/science/article/pii/S0164121213001234, doi:https://doi.org/10.1016/j.jss.2013.04.076.

[101] Wollschlaeger, M., Theurich, S., Winter, A., Lubnau, F., Paulitsch, C., 2015. A reference architecture for condition monitoring, in: IEEE World Conference on Factory Communication Systems (WFCS), pp. 1–8.

[102] Zani, V., Feitosa, D., Nakagawa, E., 2011. Current state of reference architectures in the context of agile methodologies, pp. 590–595. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84855560170&partnerID=40&md5=6346b90c933471b092d3d69eacb3cf02. cited By 1.

[103] Álvarez, B., Iborra, A., Alonso, A., de la Puente, J.A., 2001. Reference architecture for robot teleoperation:: development details and practical use. Control Engineering Practice 9, 395 – 402. URL: http://www.sciencedirect.com/science/article/pii/S0967066100001210, doi:https://doi.org/10.1016/S0967-0661(00)00121-0.

Lina Garcés is an Assistant Professor at the Institute of Mathematics and Computing, Federal University of Itajubá (IMC/UNIFEI), Brazil. She was a postdoctoral research fellow (2019-2020) at University of São Paulo (USP), Brazil. She received her BEng (2009) and MSc (2012) from the Industrial University of Santander (UIS), Colombia, and her PhD (2018) from USP, and University of Southern Brittany (UBS), France. Her main research interests are software architectures, software engineering, empirical software engineering, interoperability, systems-of-systems, e-Health, and ambient assisted living. She is a member of the IEEE, SBC (Brazilian Computer Society), and SBIS (Brazilian Society of Health Informatics) acting in the consolidation of technical standards for Clinical IoT. Contact her at *lina@unifei.edu.br*

Silverio Martínez-Fernández is an Assistant Professor at UPC-BarcelonaTech since January 2020. He was a Post-Doctoral Fellow of the European Research Consortium for Informatics and Mathematics (2016-2018) and operative project manager (2018-2019) in Fraunhofer IESE (Germany). His interests include Empirical Software Engineering, Reference Architectures, Software Analytics, Data-driven Development, and AI/ML-based systems. In EU framework programmes, he acted as Evaluation WP leader in Q-Rapids (H2020, RIA). He has participated in the organization of several conferences and workshops (e.g., co-chair of PROFES 2019, CESI@ICSE 2018). He is Editorial Board Member of the SCI-indexed journal IET Software (IEEE). He has also been reviewer of multiple journals (e.g. IST, JSS, IJCIS) and PC member of international conferences (e.g. ESEM, ICSME, ECSA, CIbSE). Contact him at *silverio.martinez@upc.edu.*

Lucas Bueno Ruas de Oliveira is an Assistant Professor at the Federal Institute of São Paulo (IFSP), Brazil. He received his double PhD degree in Computer Science from University of São Paulo (USP) and University of Southern Brittany (UBS), in 2015. He conducted a research internship in the ARCHWARE team at IRISA, France, and worked as postdoctoral fellow at USP. His main research interests are related to software architecture, reference architecture, service-oriented architecture, and robotic systems. He is a member of the Brazilian Computer Society (SBC). Contact him at *lucas.oliveira@ifsp.edu.br*.

Pedro Valle is a doctorate student in computer sciences at the Institute of Mathematic and Computer Sciences (ICMC) - University of São Paulo (USP), Brazil. He received his BSc (2013) from the Federal University of Goiás (UFG) and his MSc (2016) in computer science from USP. Currently, his research interests are software architecture, reference architectures, interoperability, and Systems-of-Systems. Contact him at *pedrohenriquevalle@usp.br*

Claudia Ayala is an Associate Professor at the Universitat Politècnica de Catalunya (UPC). She received her PhD degree in Informatics from UPC. She was a postdoctoral ERCIM research fellow at the Norwegian University of Science and Technology (NTNU), Norway. Her current research interests include empirical software engineering, open source software engineering, requirements engineering, software architecture and quality. She serves as reviewer of several journals and is a member of several program committees in the area. Contact her at *cayala@essi.upc.edu.*

Xavier Franch is a full professor of informatics at Universitat Politècnica de Catalunya (UPC). His main interests are in Requirements Engineering, Software Architecture, and Empirical Software Engineering. He belongs to the Editorial Board of IST, REJ, Computing, and IJCIS. He is Journal First chair at JSS and Editor-in-Chief at IET Software. He has been chair and has organized more than fifty international scientific events in the area of software engineering. Contact him at *franch@essi.upc.edu*.

Elisa Yumi Nakagawa is a MS (1998) and PhD (2006) in Computer Science from the Institute of Mathematic and Computer Sciences (ICMC) - University of São Paulo (USP), Brazil. She conducted her Post-Doctoral in 2011–2012 in Fraunhofer IESE, Germany, and in 2014–2015 at University of Southern Brittany (UBS), France. She is associate professor in the Department of Computer Systems at USP. Her main research interests are software architecture, reference architectures, systems-of-systems, software testing, and evidence-based software engineering. She is a member of the IEEE and SBC. Contact her at *elisa@icmc.usp.br*

A. List of primary studies

Table 5: Primary studies addressing reference architectures (Proponent (A=Academia, I=Industry, C = Academy-Industry Collaboration), Purpose (F=Facilitation, S=Standardization), Perspective (C=Classical, P=Preliminary), Maturity, and Domain as presented in Section 3)

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA1	Mulazzani, M., 1988. An open layered architecture for dependability analysis and its application, FTCS, pp. 96-101.	A	F	C	E1	Software Environments
RA2	Kramer, T., Senehi, M.K., Michaloski, J., Ray, S., Rippey, W., Wal-lace, S., Quintero, R., Albus, J., 1994. A reference architecture for control of mechanical systems, in: Tutorial and Workshop on Systems Engineering of Computer-Based Systems, 1994., Proceedings.,1994, pp. 104–110.	I	S	C	E0	Transportation
RA3	Hayes-Roth, B., Larsson, J., 1996. A domain-specific software architecture for a class of intelligent patient monitoring agents. Journal of Experimental and Theoretical Artificial Intelligence 8, 149–171.	A	F	P	E2	Healthcare
RA4	Gronbaek, K., Wiil, U., 1997. Towards a common reference architecture for open hypermedia. Journal of Digital Information, v. 1, n. 2.	A	S	C	E1	Multimedia
RA5	Grefen, P., Remmerts De Vries, R., 1998. A reference architecture for workflow management systems. Data and Knowledge Engineering 27, 31–57.	A	F	C	E0	Business Management
RA6	Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters,P., 1998. Reference architecture for holonic manufacturing systems: Prosa. Computers in Industry 37, 255 –274.	A	F	P	E2	ICT
RA7	Welch, L., Masters, M., Madden, L., Marlow, D., Irey_IV, P.,Werme, P., Shirazi, B., 1999. A distributed system reference architecture for adaptive qos and resource management. In: International Parallel Processing Symposium. Springer, Berlin, Heidelberg, p. 1316-1326.	A	F	P	E0	Militar
RA8	Savigni, A., Tisato, F., 1999. Kaleidoscope - a reference architecture for monitoring and control systems, in: Donohoe, P (Ed.), Software Architecture, pp. 369–388.	A	F	C	E0	Monitoring
RA9	Alvarez, B., Iborra, A., Alonso, A., de la Puente, J.A., 2001. Reference architecture for robot teleoperation:: development details and practical use. Control Engineering Practice 9, 395 – 402.	A	F	C	E5	Robotics
RA10	Sandkuhl, K., Messer, B., 2000. Towards reference architectures for distributed groupware applications, in: Proceedings of 8th Euromicro Workshop on Parallel and Distributed Processing, pp. 135–141.	A	F	C	E1	Cooperative Work
RA11	Barber, K., Graser, T., Silva, J., 2001. Developing a traceable do-main reference architecture to support clinical trials at the national cancer institute an experience report, in: Proceedings of Eighth Annual IEEE International Conference and Workshop On the Engineering of Computer-Based Systems (ECBS), pp. 144–151.	C	F	C	E5	Healthcare
RA12	Campos, F., Mills III, W., Graves, M., 2002. A reference architecture for remote diagnostics and prognostics applications, in: Proceedings of IEEE AUTOTESTCON, pp. 842–854.	I	F	C	E5	Healthcare
RA13	Xu, J., Randell, B., Romanovsky, A., 2002. A generic approach to structuring and implementing complex fault-tolerant software, in: Proceedings Fifth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISIRC), pp.207–214.	A	F	P	E1	Fault-tolerant
RA14	Followell, D., Gilbertson, D., Keller, K., 2004. Implications of an open system approach to vehicle health management, in: Proceedings of IEEE Aerospace Conference. 2004, pp. 3717–3724, Vol.6	I	S	C	E5	Transportation
RA15	Myrhaug, H., Whitehead, N., Goker, A., Erlend, T., Till, F., Lech,C., 2004. Ambiesense â a system and reference architecture for personalised context-sensitive information services for mobile users, in:European Symposium on Ambient Intelligence. Springer, Berlin, Heidelberg, p. 327-338.	C	F	P	E1	Commerce
RA16	Bashroush, R., Spence, I., Kilpatrick, P., Brown, T., 2005.A generic reference software architecture for load balancing over mirrored web servers: Nasr case study, in:3rd ACS/IEEE International Conference on Computer Systems and Applications, 2005, pp. 743–746.	A	F	P	E1	SOA
RA17	o, L., Junde, S., 2005. Design and implementation of a new application based on lbs over 3g service layer. In: 2nd Asia Pacific Conference on Mobile Technology, Applications and Systems. IEEE, p. 1-4.	A	F	P	E1	Communication
RA18	Governor, J., Hinchcliffe, D., Nickull, D., 2009. Web 2.0 Architectures: What entrepreneurs and information architects need to know." O'Reilly Media, Inc."	A	F	C	E5	Web
RA19	Gruhn, V., Weber, T., 2005. From an e-business revenue model to its software reference architecture. IFIP Advances in Information and Communication Technology 189, 33–47.	A	F	P	E5	Business Management
RA20	Niemela, E., Kalaoja, J., Lago, P., 2005. Toward an architectural knowledge base for wireless service engineering. IEEE Transactions on Software Engineering, 31, 361–379.	C	S	C	E4	Communication
RA21	ollard, J., Duke, R., 2005. A reference architecture for instructional educational software, in: Proceedings of the 2005 South East Asia Regional Computer Science Confederation (SEARCC), Vol. 46, pp. 43–52.	A	F	P	E2	Education
RA22	Beer, D., Kunis, R., Runger, G., 2006. A component based software architecture for e-government applications, in: The First International Conference on Avail-ability, Reliability and Security (ARES). pp. 1004–1011.	A	S	P	E0	Governmental
RA23	Singh, A., Haahr, M., 2006. A peer-to-peer reference architecture. In: 1st International Conference on Communication Systems Software & Middleware. Vols 1 and 2, pp. 109–118.	A	F	C	E1	P2P
RA24	Vorobiev, A., Han, J., 2006. Secrobat: Secure and robust component-based architectures. In: 13th Asia Pacific Software Engineering Conference (APSEC), pp. 3–10.	A	F	P	E1	Component Based Systems
RA25	Liu, L., Thanheiser, S., Schmeck, H., 2008. A reference architecture for self-organizing service-oriented computing. In: International Conference on Architecture of Computing Systems. Springer, Berlin, Heidelberg, 2008. p. 205-219.	A	F	P	E0	Pervasive Computing
RA26	Zewdie, B., Carlson, C., 2006. Adaptive component paradigm for highly configurable business components. In: IEEE International Conference on Electro/information Technology, pp. 185–190.	A	F	P	E1	Component Based Systems
RA27	Arsanjani, A., Zhang, L.J., Ellis, M., Allam, A., Channabasavaiah,K., 2007. S3: A service-oriented reference architecture. IT Professional 9, pp. 10–17.	I	F	C	E5	SOA

Continued on next page

Table 5 – *Continued from previous page*

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA28	Candela, L., Castelli, D., Pagano, P., 2007. A reference architecture for digital library systems: Principles and applications, in: Proceedings of the 1st International Conference on Digital Libraries: Re-search and Development, Springer-Verlag, Berlin, Heidelberg. pp.22–35.	C	F	C	E5	Scientific Systems
RA29	Folmer, E., 2007. Component based game development a solution to escalating costs and expanding dead-lines? Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 4608 LNCS, 66–73.	A	F	P	E0	Games
RA30	Hadar, E., Perreira, M., 2007. Web services variation façade - domain specific reference architecture for increasing integration usability. In: IEEE International Conference on Web Services (ICWS), pp. 1207–1211.	I	F	P	E1	SOA
RA31	peng Chen, H., Zhang, C., 2007. A queueing-theory-based fault detection mechanism for soa-based applications, in: The 9th IEEE International Conference on E-Commerce Technology and the 4th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, 2007. (CEC/EEE), pp. 157–166.	A	F	P	E0	SOA
RA32	Murakami, E., Saraiva, A., Ribeiro Jr., L., Cugnasca, C., Hirakawa,A., Correa, P., 2007. An infrastructure for the development of distributed service-oriented information systems for precision agriculture. Computers and Electronics in Agriculture, Vol. 58, pag. 37–48.	A	F	P	E1	Agriculture
RA33	Nakagawa, E., Guessi, M., Maldonado, J., Feitosa, D., Oquendo,F., 2014. Consolidating a process for the design, representation,and evaluation of reference architectures. In: IEEE/IFIP International Conference on Software Architecture (WICSA), pp. 143–152	A	F	C	E2	Software Environments
RA34	Oliveira, M., Pereira, J., 2007. Extensible virtual environment systems using system of systems engineering approach. In: 17th International Conference on Artificial Reality and Telexistence, pp. 89–96.	A	F	C	E2	Software Environments
RA35	Tamblyn, S., Hinkel, H., Saley, D., 2007. National aeronautics and space administration (nasa) crew exploration vehicle (cev) reference guidance, navigation, and control (gn&c) architecture, pp. 587–605.	I	S	C	E5	Transportation
RA36	WEYNS, D.,HOLVOET, T., 2006. A reference architecture for situated multiagent systems. In: International Workshop on Environments for Multi-Agent Systems. Springer, Berlin, Heidelberg, p. 1-40.	A	F	C	E3	Multi-Agents
RA37	Bahuguna, A., 2008. Reference architecture for enterprise batch processing of information. In: Third International Conference on Digital Information Management (ICDIM), pp. 8–17.	I	S	C	E5	Business Management
RA38	Carbon, R., Johann, G., Keuler, T., Muthig, D., Naab, M., Zilch,S., 2008. Mobility in the virtual office - a document-centric workflow approach. In: International Workshop on Software Architectures and Mobility, pp. 21–26.	C	F	C	E1	Business Management
RA39	Kling, M., Mitchener, M., Meyer, B., Jones, D., 2008. An implementation of the government reference architecture wave-form developer and system integrator roles. In: IEEE Military Communications Conference (MILCOM), IEEE, p. 1-5.	I	S	C	E5	Electrical
RA40	Liu, L., Thanheiser, S., Schmeck, H., 2008. A reference architecture for self-organizing service-oriented computing. In: International Conference on Architecture of Computing Systems. Springer, Berlin, pp. 205–219.	A	F	P	E0	SOA
RA41	Resinas, M., Fernandez, P., Corchuelo, R., 2008. A reference architecture for automated negotiations of service agreements in open and dynamic environments. In: On the Move to Meaningful Internet Systems: OTM Workshop. p. 463-472.	A	F	P	E1	Software Environments
RA42	Zheng, Q., Dong, B., Tian, F., Chen, W., 2008. A service-oriented approach to integration of e-learning information and resource management systems, in: 12th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 1047–1052.	A	S	C	E1	Education
RA43	Bitzer, S., Ramroth, S., Schumann, M., 2009. Mashups as an architecture for knowledge management systems. In: 42nd Hawaii International Conference on System Sciences (HICSS), pp. 1–10.	A	S	C	E0	Business Management
RA44	Casola, V., Gaglione, A., Mazzeo, A., 2009.A reference architecture for sensor networks integration and management.Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 5659 LNCS, pp. 158–168.	A	S	P	E1	Sensor Networks
RA45	Choi, H., Lim, C., Kim, J., 2009. Defining reference architecture for NTIS development. In: 11th International Conference on Advanced Communication Technology. IEEE, p. 284-287.	A	S	C	E0	Governmental
RA46	Yu, L., Zhang, L., Xiang, H., Su, Y., Zhao, W., Zhu, J., 2009. A framework of testing as a service. In: International Conference on Management and Service Science (MASS), pp. 1–4.	C	F	C	E1	Software Environments
RA47	Lockemann, P., Nimis, J., 2009. Dependable multi-agent systems:Layered reference architecture and representative mechanisms. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 4324 LNAI, 27–48.	A	F	P	E2	Multi-Agents
RA48	Meland, P., Ardi, S., Jensen, J., Rios, E., Sanchez, T., Shahmehri,N., Tondel, I., 2009. An architectural foundation for security model sharing and reuse; In: International Conference on Availability, Reliability and Security (ARES), pp. 823–828.	A	F	P	E2	Software Environments
RA49	Peristeras, V., Fradinho, M., Lee, D., Prinz, W., Ruland, R., Iqbal,K., Decker, S., 2009. Cera: A collaborative environment reference architecture for interoperable cwe systems. Service Oriented Computing and Applications, vol. 3, pp. 3–23.	C	S	C	E3	Cooperative Work
RA50	Waibel, A., Stiefelhagen, R., 2009. Computers in the Human Interaction Loop. 1st ed., Springer Publishing Company, pp. 3-6.	C	F	C	E5	Ambient Intelligence
RA51	Lin, C., Lu, S., Fei, X., Chebotko, A., Pai, D., Lai, Z., Fotouhi, F.,Hua, J., 2009. A reference architecture for scientific workflow management systems and the view soa solution. IEEE Transactions on Services Computing, Vol. 2, pp. 79–92.	A	S	C	E3	Scientific Systems
RA52	Ceri, S., Brambilla, M., 2010.Search computing sys-tems. Lecture Notes in Computer Science (including sub-series Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 6051 LNCS, 1–6.	A	F	P	E0	Scientific Systems
RA53	Martinez-Barbera, H., Herrero-Perez, D., 2010. Programming multirobot applications using the thinking cap-ii java framework. Advanced Engineering Informatics, Vol. 24, pp. 62 – 75.	A	F	C	E4	Robotics

Continued on next page

Table 5 – *Continued from previous page*

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA54	Kanstren, T., Savola, R., 2010. Definition of core requirement sand a reference architecture for a dependable, secure and adaptive distributed monitoring framework. In: Third International Conference on Dependability (DEPEND), pp. 154–163.	C	F	P	E2	Monitoring
RA55	Macedo, H., 2010. Model driven development approach to natural language generation systems. ACM SIGSOFT Software Engineering Notes, v. 35, n. 4, pp. 1-7.	A	F	P	E1	Human Computer Interaction
RA56	Meissen, U., Voisard, A., 2010. Towards a reference architecture for early warning systems. In: 2nd International Conference on Intelligent Networking and Collaborative Systems (INCOS), pp. 513–518.	A	F	P	E3	Emergency Systems
RA57	Resinas, M., Fernández, P., Corchuelo, R., 2010. Automatic service agreement negotiators in open commerce environments. International Journal of Electronic Commerce v. 14, pp. 93–128.	A	F	C	E3	Software Environments
RA58	Terraillon, J.L., Jung, A., Arberet, P., Montenegro, S.,Rossignol, A., Garcia, G., Li, J., Rodriguez, A., Mazzini,S., Hougaard, P., Fowell, S., Ferraguto, M., Panunzio, M., 2010. Space on-board software reference architecture, Vol. 682.	C	S	P	E5	Transportation
RA59	ing, J., Zhang, J., 2010. Research on open SaaS software architecture based on SOA. In: International Symposium on Computational Intelligence and Design (ISCID), IEEE, pp. 144–147.	A	F	C	E0	SOA
RA60	Rivero, C.R., HernÃ¡ndez, I., Ruiz, D., Corchuelo, R., 2011. A reference architecture for building semantic-web mediators. International Conference on Advanced Information Systems Engineering. Springer, Berlin, Heidelberg, p. 330-341.	A	S	C	E1	SOA
RA61	De La Cruz, E., Lopez, D.b., Uribe, G., Gonzalez, C.b., Blo-bel, B., 2011. A reference architecture for integrated ehrin colombia, v. 169, pp. 305-309.	C	S	C	E1	Healthcare
RA62	Krellner, B., Kunis, R., Runger, G., 2011. Modeling of energy-sensitive manufacturing processes. In: 9th International Conference on Industrial Informatics. IEEE, 2011. p. 334-340.	A	F	C	E0	Electrical
RA63	Mikkonen, T., Salminen, A., 2011. Towards a reference architecture for mashups. Lecture Notes in Computer Science(including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7046 LNCS, pp. 647–656.	A	F	C	E1	Web
RA64	Machado, C. A., Silva, E., Batista, T., Leite, J., & Nakagawa, E., 2014. RA-Ubi: A Reference Architecture for Ubiquitous Computing. In European Conference on Software Architecture. Springer, pp. 98-105.	A	S	C	E4	Software Environments
RA65	Oliveira, L., Nakagawa, E., 2011. A service-oriented reference architecture for software testing tools. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 6903 LNCS, pp. 405–421.	A	F	C	E3	SOA
RA66	Riddick, F., Kibira, D., Tina Lee, Y., Balakirsky, S., 2011. A component-based approach for manufacturing simulation. In: Proceedings of the 2011 Emerging M&S Applications in Industry and Academia Symposium. Society for Computer Simulation International, pp. 54-61.	I	S	C	E5	Component Based Systems
RA67	Trew, T., Botterweck, G., & Nuseibeh, B., 2011. A Reference Architecture for Consumer Electronics Products and Its Application in Requirements Engineering. In Relating Software Requirements and Architectures, Springer, Berlin, Heidelberg, pp. 203-231.	C	S	C	E5	Electrical
RA68	Astudillo, H., Becerra, C.b., 2012. A digital library built around the information-content model duality. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7634 LNCS, pp. 299–308.	A	F	P	E1	Scientific Systems
RA69	Sleiman, H.A., Corchuelo, R., 2012. A reference architecture to devise web information extractors. In: International Conference on Advanced Information Systems Engineering. Springer, Berlin, Heidelberg, pp. 235-248.	A	F	C	E1	Web
RA70	Verriet, J., & van Wijngaarden, B., 2012. A reference architecture capturing structure and behaviour of warehouse control. In Automation in Warehouse Development. Springer, London, pp. 17-32.	C	S	C	E1	Warehouse Management
RA71	Liu, J., Zhang, L.J., Hu, B., He, K., 2012. Ccra: Cloud computing reference architecture. In: IEEE Ninth International Conference on Services Computing (SCC), pp. 657–665.	I	S	C	E5	Cloud Computing
RA72	Lewis, G., Novakouski, M., Sânchez, E., 2013. A reference architecture for group-context-aware mobile applications. Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering 110 LNICST, pp. 44–63.	A	F	C	E1	Ambient Intelligence
RA73	Simanta, S., Lewis, G., Morris, E., Ha, K., Satyanarayanan, M., 2012. A reference architecture for mobile code offload in hostile environments. In: IEEE/IFIP International Conference on Software Architecture (WICSA), pp. 282–286.	A	F	C	E1	Militar
RA74	Zimmermann, O., Miksovic, C., Küster, J., 2012. Reference architecture, metamodel, and modeling principles for architectural knowledge management in information technology services. Journal of Systems and Software, v. 85, n. 9, p. 2014-2033.	C	F	P	E2	Business Management
RA75	Satake, H., Skutt, T., Sherman, M., Eagleson, W., Rittenbach, T.,Sepka, T., 2013. Government reference architecture extensions for application to base stations, in: Military Communications Conference, IEEE, pp. 1762–1767.	I	F	C	E4	Militar
RA76	Affonso, F., Nakagawa, E., 2013. A reference architecture based on reflection for self-adaptive software. In: VII Brazilian Symposium on Software Components, Architectures and Reuse (SBCARS), pp. 129–138.	A	F	C	E1	Self-adaptive Systems
RA77	Rocha, H., Costa, E., Brito, H., Tenorio, M., 2013. Recommending software architecture for customized learning management system. In: 8th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1–6.	A	F	C	E0	Education
RA78	Bruns, R., Dunkel, J., 2014. Towards pattern-based architectures for event processing systems. Software - Practice and Experience 44, pp. 1395–1416.	A	F	P	E2	Software Environments
RA79	Fitzpatrick, D., Coallier, F., & Ratté, S., 2013. A reference architecture for an enterprise knowledge infrastructure. In: IFIP International Conference on Product Life-cycle Management. Springer, Berlin, Heidelberg, pp. 326-335.	A	F	P	E0	Business Management
RA80	Guerra, E., Alves, F., Kulesza, U., Fernandes, C., 2013. A reference architecture for organizing the internal structure of metadata-based frameworks. Journal of Systems and Software 86, pp. 1239–1256.	C	F	C	E5	Software Environments
RA81	Heisey, C., Hendrickson, A., Chludzinski, B., Cole, R., Ford, M.,Herbek, L., Ljungberg, M., Magdum, Z., Marquis, D., Mezhi-rov, A., Pennell, J., Roe, T., Weinert, A., 2013. A reference software architecture to support unmanned aircraft integration in the national airspace system. Journal of Intelligent and Robotic Systems: Theory and Applications, v. 69, pp. 41–55.	C	F	P	E5	Transportation

Continued on next page

Table 5 – *Continued from previous page*

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA82	Heredero, G., Penmetsa, H., Agrawal, V., Shastri, L., 2013. Activity context-aware system architecture for intelligent natural speech based interfaces. In: Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence, pp. 21–35.	I	F	C	E5	Human Computer Interaction
RA83	van Sinderen; M. Steenwijk; P. Verkroost, M.E.I.M.J., 2013. Towards a reference architecture for fuel-based carbon management systems in the logistics industry. In: Information Systems Frontier, v. 15, n. 5, pp. 725-745.	C	F	C	E1	Business Management
RA84	Bauer, M., Boussard, M., Bui, N., De Loof, J., Magerkurth, C., Meissner, S., & Walewski, J. W., 2013. IoT reference architecture. In: Enabling Things to Talk. Springer, Berlin, Heidelberg. pp. 163-211.	C	F	C	E5	IoT
RA85	ernandez-Montes, A., Ortega, J., Sanchez-Venzala, J., Gonzalez-Abril, L., 2014. Software reference architecture for smart environments: Perception. Computer Standards & Interfaces, v. 36, pp. 928 – 940.	A	F	P	E1	Ambient Intelligence
RA86	Addo, I., Ahamed, S., Yau, S., Buduru, A., 2014. A reference architecture for improving security and privacy in internet of things applications. In: International Conference on Mobile Services. IEEE, p. 108-115.	A	F	C	E2	IoT
RA87	Machado, C. A., Silva, E., Batista, T., Leite, J., & Nakagawa, E., 2014. RA-Ubi: A Reference Architecture for Ubiquitous Computing. In: European Conference on Software Architecture. Springer. pp. 98-105.	A	F	C	E1	Ubiquitous computing
RA88	Chauhan, M., Babar, M., 2014. Towards a reference architecture to provision tools as a service for global software development. In: IEEE/IFIP Conference on Software Architecture (WICSA), pp. 167–170.	A	F	P	E0	Cloud Computing
RA89	Jaramillo, D., Furht, B., Agarwal, A., 2014. Mobile virtualization reference architecture. In: Virtualization Techniques for Mobile Systems. Springer, 2014. pp. 37-53.	C	S	C	E2	Business Management
RA90	Eklund, U., Bosch, J., 2014. Architecture for embedded open software ecosystems. Journal of Systems and Software, v. 92, pp. 128–142.	C	F	C	E5	Embedded Systems
RA91	Hildebrandt, D., 2014. A software reference architecture for service-oriented 3d geovisualization systems. ISPRS International Journal of Geoinformation, v. 3, pp. 1445–1490.	A	F	P	E1	Geograph. Inf. Systems
RA92	Portocarrero, J., Delicato, F., Pires, P., Batista, T., 2014. Reference architecture for self-adaptive management in wireless sensor networks. In: International Conference on Adaptive and Intelligent Systems. Springer, Cham, pp. 110-120.	A	F	P	E2	Self-adaptive Systems
RA93	Norta, A., Grefen, P., Narendra, N., 2014. A reference architecture for managing dynamic inter-organizational business processes. Data and Knowledge Engineering, v. 91, pp. 52–89.	I	S	C	E4	Business Management
RA94	Tajalli, H., MedvidovicÌ, N., 2014. Idare - a reference architecture for integrated software environments. Software - Practice and Experience, v. 44, pp. 299–316.	A	S	P	E2	Software Environments
RA95	Aßmann, U., Götz, S., Jézéquel, J. M., Morin, B., & Trapp, M., 2014. A reference architecture and roadmap for models@ run. time systems. In Models@ run. time, Springer, Cham, pp. 1-18.	C	F	P	E2	Self-adaptive Systems
RA96	Bruns, R., Dunkel, J., 2014. Towards pattern-based architectures for event processing systems. Software - Practice and Experience, v. 44, pp. 1395–1416.	A	F	C	E0	Software Environments
RA97	Gherardi, L., & Brugali, D. (2014, May). Modeling and reusing robotic software architectures: the hyperflex toolchain. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE. pp. 6414–6420.	A	F	C	E1	Robotics
RA98	Fernandez, E. B., & Monge, R., 2014. A security reference architecture for cloud systems. In Proceedings of the WICSA 2014 Companion Volume (p. 3). ACM.	A	F	C	E2	Cloud Computing
RA99	Braberman, V., D&039; Ippolito, N., Kramer, J., Sykes, D., Uchitel, S., 2015. Morph: A reference architecture for configuration and behaviour self-adaptation. In: Proceedings of the 1st International Workshop on Control Theory for Software Engineering, pp. 9–16.	A	F	P	E1	Militar
RA100	Castelli, G., Mamei, M., Rosi, A., & Zambonelli, F., 2015. Engineering pervasive service ecosystems: the sapere approach. In: ACM Transactions on Autonomous and Adaptive Systems (TAAS), v. 10, pp. 10.	A	F	C	E2	Electrical
RA101	Losavio, F., Ordaz, O., 2015. Quality-based heuristic for optimal product derivation in software product lines. In: Internet Technologies and Applications (ITA), 2015, pp. 125–131.	A	F	C	E1	Healthcare
RA102	Wollschlaeger, M., Theurich, S., Winter, A., Lubnau, F., Paulitsch, C., 2015. A reference architecture for condition monitoring. In: IEEE World Conference on Factory Communication Systems (WFCS), pp. 1–8.	I	F	C	E4	Industrial Plants
RA103	Rossiter, S., 2015. Simulation Design: Trans-Paradigm Best-Practice from Software Engineering. Journal of Artificial Societies and Social Simulation, pp. 1-47.	A	F	C	E1	Software Environments
RA104	Srinivasan, V., Murphy, R.R., Bethel, C.L., 2015. A Reference Architecture for Social Head Gaze Generation in Social Robotics. In: International Journal of Social Robotics, v. 7, n. 5, p. 601-616.	A	F	C	E3	Robotics
RA105	erez, J., Diaz, J., Garbajosa, J., Yague, A., Gonzalez, E., Lopez-Perea, M., 2015. Towards a reference architecture for large-scale smart grids system of systems. In: Proceedings of the Third International Workshop on Software Engineering for Systems-of-Systems, IEEE Press, Piscataway, NJ, USA. pp. 5–11.	C	F	C	E2	Electrical
RA106	Schroeder, J., Holzner, D., Berger, C., Hoel, C.J., Laine, L., Magnusson, A., 2015. Design and evaluation of a customizable multi-domain reference architecture on top of product lines of self-driving heavy vehicles: An industrial case study, in: Proceedings of the 37th International Conference on Software Engineering - Volume 2, IEEEPress, Piscataway, NJ, USA. pp. 189–198.	C	F	C	E4	Transportation
RA107	De Sutter, B., Falcarin, P., Wyseur, B., Basile, C., Ceccato, M.,d'Annoville, J., Zunke, M., 2016. A Reference Architecture for Software Protection. In: 13th Working IEEE/IFIP Conference on Software Architecture (WICSA). IEEE, pp. 291-294.	A	F	P	E0	Software Environments
RA108	Diniz, H., Silva, E., Nogueira, T., Gama, K., 2016. A Reference Architecture for Mobile Crowdsensing Platforms. In: International Conference on Enterprise Information Systems (ICEIS), V. 2, pp. 600–607.	A	F	C	E0	Smart Cities
RA109	Leite, A., Girardi, R., 2016. A Reference Architecture of a Hybrid Learning Agent. In: IEEE/WIC/ACM International Conference on Web Intelligence (WI). IEEE, pp. 421-424.	A	F	P	E1	Education
RA110	Gadea, C., Trifan, M., Ionescu, D., Ionescu, B., 2016. A reference architecture for real-time microservice api consumption. In: Proceedings of the 3rd Workshop on Cross Cloud Infrastructures & Platforms, ACM, New York, NY, USA. pp. 21–26.	A	F	P	E0	SOA

Continued on next page

Table 5 – *Continued from previous page*

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA111	Evers, S., Ernsting, J., Majchrzak, T.A., 2016. Towards a reference architecture for model-driven business apps. In: 49th Hawaii International Conference on System Sciences (HICSS), pp. 5731–5740.	A	F	C	E4	Business Management
RA112	Garcia-Rodriguez, S., Sleiman, H.A., Nguyen, V.Q.A., 2016. A multi-agent system architecture for microgrid management. In International Conference on Practical Applications of Agents and Multi-Agent Systems, Springer, Cham, pp. 55-67.	A	F	C	E1	Multi-Agents
RA113	Spinner, S., Walter, J., Kounev, S., 2016. A reference architecture for online performance model extraction in virtualized environments. In: Companion Publication for ACM/SPEC on International Conference on Performance Engineering, ACM, New York, NY, USA.pp. 57–62.	A	F	C	E0	Software Environments
RA114	Kassahun, A., Hartog, R., Tekinerdogan, B., 2016. Realizing chain-wide transparency in meat supply chains based on global standards and a reference architecture. In: Computers and Electronics in Agriculture, v. 123, pp. 275 – 291.	C	F	C	E1	Agriculture
RA115	eureiter, C., Uslar, M., Engel, D., Lastro, G., 2016. A Standards-based Approach for Domain Specific Modelling of Smart Grid System Architectures. In: 11TH System of Systems Engineering Conference (SOSE), IEEE, pp. 1-6.	C	S	C	E0	Electrical
RA116	Park, S., Park, Y.B., 2016. Ite arbitrator: A reference architecture framework for sustainable it ecosystems. In: IEEE/ACM 4th International Workshop on Software Engineering for Systems-of-Systems (SESoS), pp. 25–31.	A	F	C	E3	ICT
RA117	Vasilecas, O., Kalibatiene, D., LavbiÄ, D., 2016. Rule- and context-based dynamic business process modelling and simulation. Journal of Systems and Software (JSS), v. 122, pp. 1–15.	A	F	P	E1	Business Management
RA118	Carrez, F., Elsaleh, T., Gómez, D., Sánchez, L., Lanza, J., Grace,P., 2017. A reference architecture for federating iot infrastructures supporting semantic interoperability. In: 2017 European Conference on Networks and Communications (EuCNC). IEEE, pp. 1-6.	A	S	C	E0	IoT
RA119	Nadal, S., Herrero, V., Romero, O., Abell, A., Franch, X., Vansummeren, S., Valerio, D., 2017. A software reference architecture for semantic-aware big data systems. In: Information and Software Technology (IST), v. 90, pp. 75–92.	C	F	P	E4	Big Data
RA120	Chauhan, M.A., Babar, M.A., Sheng, Q.Z., 2017. A reference architecture for provisioning of tools as a service: Meta-model, ontologies and design elements. In: Future Generation Computer Systems, v. 69, pp. 41–65.	A	F	C	E2	SOA
RA121	Zbick, J., 2017. A Web-based Reference Architecture for Mobile Learning: Its Quality Aspects and Evaluation. In: IEEE International Conference on Software Architecture Workshops (ICSAW). IEEE, pp. 230-235.	A	S	C	E1	Education
RA122	Roy, B., Mondal, A.K., Roy, C.K., Schneider, K.A., Wazed, K., 2017. Towards a Reference Architecture for Cloud-based Plant Genotyping and Phenotyping Analysis Frameworks. In: International Conference on Software Architecture (ICSA). IEEE, pp. 41-50.	A	S	C	E0	Cloud Computing
RA123	Balis, B., Bartynski, T., Bubak, M., Harezlak, D., Kasztelnik, M.,Malawski, M., Nowakowski, P., Pawlik, M., Wilk, B., 2017. Smart levee monitoring and flood decision support system: reference architecture and urgent computing management. In: Procedia Computer Science, v. 108, pp.2220–2229.	A	F	C	E3	Emergency Systems
RA124	Beltran, V., Skarmeta, A., Ruiz, P., 2017. An arm-compliant architecture for user privacy in smart cities: Smartie quality by design in the iot. In: Wireless Communications and Mobile Computing, pp.1-13.	A	F	C	E1	Smart Cities
RA125	Geisriegler, M., Kolodiy, M., Stani, S., Singer, R., 2017. Actor based business process modeling and execution: A reference implementation based on ontology models and microservices. In: 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA). IEEE, pp. 359–362.	A	F	P	E0	Business Management
RA126	Hernandez-Mendez, A., Braun, D., Matthes, F., Langen, M., 2017.Towards a context-aware vertical social software ecosystem. In: IEEE 19th Conference on Business Informatics (CBI), pp. 76–82.	A	S	C	E0	Ambient Intelligence
RA127	Huh, M., Park, J., 2017. Design of interfaces among functionalities for smart greenhouse. In: 19th International Conference on Advanced Communication Technology (ICACT). IEEE, pp. 649–652.	I	S	C	E0	Agriculture
RA128	Male, G., Shaghouei, E., Pattinson, C., 2017. Towards the development of a culture aware reference architecture for mobile technology: A tools and methodology perspective. In: IEEE AFRICON, pp.968–973.	A	F	C	E2	ICT
RA129	Jazayeri, B., Schwichtenberg, S., 2017. On-the-fly computing meets iot markets - towards a reference architecture. In: IEEE International Conference on Software Architecture Workshops (IC-SAW). IEEE, pp. 120–127.	A	F	C	E0	ICT
RA130	Mao, Z., Jiang, Y., Min, G., Leng, S., Jin, X., Yang, K., 2017. Mobile social networks: Design requirements, architecture, and state-of-the-art technology. In: Computer Communications, v. 100, pp. 1–19.	A	S	C	E0	Social Networks
RA131	Nortocarrero, J.M., Delicato, F.C., Pires, P.F., Costa, B., Li, W., Si,W., Zomaya, A.Y., 2017. Ramses: A new reference architecturefor self-adaptive middleware in wireless sensor networks. In: Ad HocNetworks, v. 55, pp. 3–27.	A	F	C	E2	ICT
RA132	Salah, H., Eltoweissy, M., 2017. Towards collaborative trust man-agement. In: IEEE 3rd International Conference on Collaboration and Internet Computing (CIC), pp. 198–208.	A	F	P	E2	Cooperative Work
RA133	Kott, A., Mancini, L.V., Théron, P., Drašar, M., Dushku, E., Günther, H., Kont, M., LeBlanc, B., Panico, A., Pihelgas, M., 2018. Initial reference architecture of an intelligent autonomous agent for cyber defense. In: arXiv preprint arXiv, pp. 1803-10664.	C	F	C	E1	Militar
RA134	Herold, S., Mair, M., Rausch, A., Schindler, I., 2013. Checking conformance with reference architectures: A case study. In: 17th IEEE International Enterprise Distributed Object Computing Conference. IEEE, pp. 71-80.	A	S	C	E4	Governmental
RA135	Regli, W., Mayk, I., Cannon, C., Kopena, J., Lass, R., Mongan, W.,Nguyen, D., Salvage, J., Sultanik, E., Usbeck, K., 2014. Development and specification of a reference architecture for agent-based systems. In: IEEE Transactions on Systems, Man, and Cybernetics: Systems, v. 44, pp. 146–161.	C	F	C	E1	Multi-Agents
RA136	Tekinerdogan, B., OÌzturk, K., Dogru, A., 2011. Modeling and reasoning about design alternatives of software as a service architectures. In: 9th Working IEEE/IFIP Conference on Software Architecture (WICSA), pp. 312–319.	A	F	P	E1	Cloud Computing

Continued on next page

Table 5 – *Continued from previous page*

ID	Reference	Proponent	Purpose	Perspective	Maturity	Domain
RA137	Olaf Zimmermann, Petra Kopp, S.P., 2009. Architectural knowledge in an soa infrastructure reference architecture. In: Software Architecture Knowledge Management. Springer, Berlin, Heidelberg, pp. 217-241.	I	F	C	E5	SOA
RA138	Nakagawa, E., Guessi, M., Maldonado, J., Feitosa, D., Oquendo,F., 2014a. Consolidating a process for the design, representation,and evaluation of reference architectures. In: IEEE/IFIP Conference on Software Architecture (WICSA), pp. 143–152.	A	F	C	E3	Robotics
RA139	Miksovic, C., Zimmermann, O., 2011. Architecturally significant requirements, reference architecture, and metamodel for knowledge management in information technology services. In: 9th Working IEEE/IFIP Conference on Software Architecture (WICSA), pp.270–279.	I	F	P	E2	Business Management
RA140	Astekin, M., Sozer, H., 2012. Utilizing clone detection for domain analysis of simulation systems, in: Joint Working IEEE/IFIP Conference on Software Architecture (WICSA) and European Conference on Software Architecture (ECSA), pp. 287–291.	C	F	P	E5	Scientific Systems
RA141	Hassan, A., Holt, R., 2000. A reference architecture for web servers. In: Proceedings Seventh Working Conference on Reverse Engineering. IEEE, pp. 150–159.	A	F	C	E5	Web
RA142	Rittenbach, T., Kovarik Jr., V., Krause-Aiguier, R., Stewart, C., 2010. Complex terminal systems design: Minimizing time to deployment. Proceedings of IEEE Military Communications Conference (MILCOM), pp. 656–661.	C	F	P	E0	ICT
RA143	Reitmann, B., Kinsella, S., Hayes, C., Decker, S., 2009. Implementing semantic web applications: Reference architecture and challenges. In: 5th International Workshop on Semantic Web Enabled Software Engineering, pp. 16–30.	A	F	C	E0	Web
RA144	Chen, J., Wu, D., Zhang, J., Oquendo, F., 2010. Formal modelling and analysis of hla architectural style. In: International Journal of Modelling, Identification and Control, v. 9, pp. 71–82.	C	F	C	E1	Web
RA145	Hanke, S., Mayer, C., Hoeftberger, O., Boos, H., Wichert, R., Tazari,M.R., Wolf, P., Furfari, F., 2011. universAAL – An Open and Consolidated AAL Platform. In: Ambient assisted living. Springer, Berlin, Heidelberg, pp. 127–140.	C	S	C	E2	Ambient Intelligence
RA146	Hietala, H., Ikonen, V., Korhonen, I., Lahteenmaki, J., Maksimainen,A., Pakarinen, V., Parkka, J., Saranummi, N., 2009. Feelgood- Ecosystem of PHR based products and services. In: Research report VTT-R-07000–09, VTT Technical Research Centre of Finland, pp. 1-69.	C	F	P	E1	Healthcare
RA147	Tazari, S., Valero, A.F., Dommarco, R., Lazaro Ramos, J.P., Furfari,F., 2010. PERSONA Perceptive Spaces Promoting Independent Aging. TechnicalReport D3.1.3. Fraunhofer IGD, ITACA UPV, and CNR-ISTI.	A	F	C	E1	Healthcare
RA148	Wartena, F., Muskens, J., Schmitt, L., Petkovic, M., 2010. Continua: The reference architecture of a personal telehealth ecosystem. In: 12th IEEE International Conference on e-Health Networking Applications and Services (Healthcom). Lyon, France. pp. 1–6.	C	S	C	E1	Healthcare
RA149	GIRA. Geospatial interoperability reference architecture (gira). In: www.dni.gov/files/ISE/documents/ DocumentLibrary/GIRA.pdf., pp. 1-2012.	C	F	P	E1	Transportation
RA150	NHSIA, 2018. National human services interoperability architecture (nhsia) - overview. In: www.acf.hhs.gov/sites/default/files/assets/w0120_overview.pdf. Accessed in 08/10/2018. pp. 1-51.	C	F	C	E0	Governmental
RA151	Laskey, K., Estefan, J.A., McCabe, F.G., Thornton, D., 2012. Reference architecture foundation for service oriented architecture. In: Oasis, Committee Draft, v. 2, pp. 1-26.	I	S	C	E0	SOA
RA152	ARC-IT, 2018. Arc-it - architecture reference for cooperative and intelligent transportation. In: https://local.iteris.com/arc-it/. Accessed in 08/10/2018.	C	S	C	E0	Transportation
RA153	Lin, S., Miller, B., Durand, J., Bleakley, G., Chigani, A., Martin, R., Crawford, M., 2017. The industrial internet of things volume G1: reference architecture. In: Industrial Internet Consortium, pp. 10-46.	C	S	C	E0	ICT
RA154	Drake, B.G., Hoffman, S.J., Beaty, D.W., 2014. Human exploration of mars, design reference architecture 5.0. In: IEEE Aerospace Conference. IEEE, pp. 1-24.	I	F	C	E0	Transportation
RA155	EIRA, 2018. European interoperability reference architecture. In: https://joinup.ec.europa.eu/solution/eira/. Accessed in 08/10/2018.	C	S	C	E0	Governmental
RA156	AUTOSAR, 2018. Automotive open system architecture. In: http://www.autosar.org/. Accessed in 08/10/2018.	I	S	C	E5	Transportation
RA157	Pääkkönen, P., Pakkala, D., 2015. Reference architecture and classification of technologies, products and services for big data systems. In: Big Data Research, v. 2, pp. 166–186.	C	F	P	E1	Transportation
RA158	OATH, 2007. Oath reference architecture Release 2.0. In: technical report. pp. 1-50.	I	S	C	E1	Transportation
RA159	Governor, J., Hinchcliffe, D., Nickull, D., 2009. Web 2.0 Architectures: What entrepreneurs and information architects need to know. "O'Reilly Media, Inc.".	C	F	C	E0	Transportation
RA160	Collins-Cope, M., Matthews, H., 2001. A reference architecture for component based development. In: Patel, D., Choudhury, I., Patel,S., de Cesare, S. (Eds.), OOIS 2000, Springer London, London. pp.225–237.	I	F	C	E1	Component Based Systems
RA161	J. Guth, J., Breitenbücher, U., Falkenthal, M., Leymann, F. and Reinfurt, L. Comparison of IoT platform architectures: A field study based on a reference architecture, 2016 Cloudification of the Internet of Things (CIoT), 2016, pp. 1-6.	A	S	C	E2	IoT
RA162	N. Arch-int, C. Lursinsup and P. Sophatsathit, A Reference Architecture for interoperating existing e-Learning Systems using Metadata and Web Services Model, In CIMCA-IAWTIC'06), 2005, pp. 891-896	A	S	P	E2	Education