University of Groningen

Architecting systems of systems

Cadavid Rengifo, Hector; Andrikopoulos, Vasilios; Avgeriou, Paris

Published in:
Information and Software Technology

DOI:
[10.1016/j.infsof.2019.106202](https://doi.org/10.1016/j.infsof.2019.106202)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Final author's version (accepted by publisher, after peer review)

Publication date:
2020

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):
Cadavid Rengifo, H., Andrikopoulos, V., & Avgeriou, P. (2020). Architecting systems of systems: A tertiary study. *Information and Software Technology*, 118, [106202]. <https://doi.org/10.1016/j.infsof.2019.106202>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Architecting Systems of Systems: A Tertiary Study

Héctor Cadavid^{a,*}, Vasilios Andrikopoulos^a, Paris Avgeriou^a

^a*Department of Computer Science, Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, the Netherlands*

Abstract

Context: The term System of Systems (SoS) has increasingly been used in a wide variety of domains to describe those systems composed of independent constituent systems that collaborate towards a mission that they could not accomplish on their own. There is a significant volume of research by the software architecture community that aims to overcome the challenges involved in architecting SoS, as evidenced by the number of secondary studies in the field published so far. However, the boundaries of such research do not seem to be well defined, at least partially, due to the emergence of SoS-adjacent areas of interest like the Internet of Things. **Objective:** This paper aims to investigate the current state of research on SoS architecting by synthesizing the demographic data, assessing the quality and the coverage of architecting activities and software quality attributes by the research, and distilling a concept map that reflects a community-wide understanding of the concept of SoS. **Method:** We conduct what is, to the best of our understanding, the first tertiary study on SoS architecting. Such tertiary study was based on five research questions, and was performed by following the guidelines of Kitchenham et al. In all, 19 secondary studies were evaluated, which is comparable to other tertiary studies. **Results:** The study illustrates a state of disconnection in the research community, with research gaps in the coverage of particular phases and quality attributes. Furthermore, a more effective approach in classifying systems as

*Corresponding author

Email addresses: h.f.cadavid.rengifo@rug.nl (Héctor Cadavid),
v.andrikopoulos@rug.nl (Vasilios Andrikopoulos), p.avgeriou@rug.nl (Paris Avgeriou)

SoS is required, as the means of resolving conceptual and terminological overlaps with the related domains. **Conclusions:** Despite the amount of research in the area of SoS architecting, more coordinated and systematic targeted efforts are required in order to address the identified issues with the current state of research.

Keywords: Systems of Systems, SoS Architecting, Tertiary Study, Systematic Literature Review

1. Introduction

A System of Systems is composed of constituent systems that cooperate towards a mission that they cannot accomplish on their own. Although the term System of Systems (*SoS*) does not have a widely accepted definition yet, there is a significant consensus on the characteristics, proposed by Maier [1], that distinguish an SoS from other systems: the *operational* and *managerial independence* of the constituents. Based on these characteristics, this concept has been applied in complex systems from a wide variety of domains, including defense [2, 3], manufacturing [4, 5], emergency systems [6, 7], energy [8, 9], and health care [10, 11]. The main motivation to use a SoS approach in all these domains, is having high-level goals that cannot be tackled separately by the constituents, but could eventually be fulfilled by their synergy. This could be illustrated within a manufacturing SoS, comprised of a set of independent systems including raw material suppliers, factories and distributors. In this system, even though each constituent has a different goal, their cooperation could lead to the *emergence* of higher-level goals, such as energy efficiency or inventory optimization [12].

Interestingly, the same characteristics that make SoS such a powerful concept also pose significant challenges to their architecting process. On the one hand, the dynamic nature of an SoS makes it difficult to anticipate its behavior at design time, and therefore quality requirements are difficult to address [13, 14]. On the other hand, SoS constituent systems are maintained independently and

concurrently by different organizations, in some cases following poorly documented and *ad-hoc* architectural styles [13]. For example, in the manufacturing SoS mentioned above, the architect has to (1) make design decisions considering partially unknown constituent systems (e.g. information systems from factories and distributors), and (2) evaluate such decisions regarding a high-level goal (e.g. inventory optimization), which would be difficult to predict with precision, as it will emerge from the cooperation between the constituents.

As a response to this challenge, the software architecture community has been working on a significant number of approaches to support the architecting process of SoS; this is evidenced by the increasing amount of secondary studies in the domain of SoS published in the recent years. Such secondary studies include general reviews in the domain of SoS architecting [15, 16]; discovery and composition mechanisms for SoS [17]; model-driven approaches for SoS [18]; and software architecture description languages [19], just to mention a few.

However, at this point, we lack an overview of the research related to architecting SoS; this causes both terminological confusion and under-utilization of research results. For example, despite the community-wide consensus on the characterization of SoS [12, 1], there are industrial sectors where architectures with characteristics that resemble an SoS have been created without adopting the term or using established SoS approaches [16]. Furthermore, the diverse nature of the information collected and disseminated by different industrial sectors (e.g. *Embedded Systems* and *Networks/Communications*) that make use of SoS and other overlapping concepts (e.g. *Cyber-Physical Systems* and the *Internet of Things*) is creating a lack of clarity in the terminology used in the field [20, 21].

In order to address these concerns and gain a better understanding of the challenges and boundaries involved in SoS architecting, we chose to conduct what is, to the best of our knowledge, the first *tertiary study* in the area. A tertiary study [22] refers to a systematic review of secondary studies (i.e. a Systematic Literature Review of Systematic Literature Reviews and Mapping Studies), aiming to answer wider research questions. In the case of this tertiary

study, the main research question is:

What is the current state of research in Systems of Systems architecting as evidenced by the secondary studies in the field?

The aforementioned availability of a growing number of secondary studies in the area covering different aspects of SoS architecting justifies our choice of study towards answering this question.

As a result, this study contributes to the field with *a better understanding of the state of the research efforts in SoS architecting*. More specifically, it *describes and categorizes the research* that has been published so far, and to what extent it has *covered the various phases of the architecting process and has addressed software quality attributes*. In addition, it provides the context where such research efforts has been applied by means of a *concept map* distilled from the secondary studies. This map, which is based on the interpretations of SoS concept, and their relations with other commonly related concepts, is also used to discuss further issues such as *potential research overlaps*, and *inconsistencies* not only with well-established concepts, but also with the emergent terminology in the field. These contributions, on the other hand, present a software engineering viewpoint of the topic given the nature of SLRs and SMSs as research instruments of the *evidence-based software engineering* paradigm.

The rest of this paper is structured as follows. Section 2 reports the tertiary study design, including the method followed to conduct the study. Section 3 discusses our answers to the specific research questions identified in the previous section. In Section 4, the most important findings of the study and their implications for the field are presented. In Section 5 we discuss threats to validity. Finally, our conclusions and future work are presented in Section 6.

2. Study Design

2.1. Other Tertiary Studies in Software Engineering

Systematic Literature Reviews (SLR) and Systematic Mapping Studies (SMS), are the most common types of secondary studies and the predominant instru-

ments for *Evidence-Based Software Engineering* (EBSE). They aim at evaluating and interpreting the available research (primary studies) for a particular research question, topic or phenomenon, in a methodical and repeatable way [23]. A Tertiary study, on the other hand, aims to identify, catalog and synthesize secondary studies [22]; in other words it is a Systematic Literature Review of Systematic Literature Reviews and Mapping Studies. A fairly complete survey of Tertiary Studies in Software engineering published between 2009 and 2017 is included in [24], and we identified five more published afterwards [25, 26, 27, 28, 29]. Based on this, to the best of our knowledge, at this point there is no tertiary study focused either on *Software Architecture* in general, or on *SoS architecting* in particular.

2.2. Research Method

Through an early pilot study we discovered that there is a sufficient number of secondary studies in the area of SoS to be able to perform a tertiary study on SoS architecting. For this purpose, we adopt the guidelines detailed in Kitchenham and Charters [30], and follow examples of highly-cited tertiary studies like [22, 31] in line with most other tertiary studies mentioned above.

The guidelines for tertiary studies group nine activities across three phases: *Planning*, *Conducting* and *Reporting*, as shown in Figure 1. The Planning phase has a two-fold aim: on the one hand to define the justification and scope of the review (steps 1 & 2 in the figure); and on the other to define in detail the protocol to be used for conducting the review (step 3), in order to reduce the possibility of researcher bias [30]. Once a protocol is defined, the conducting phase comprises of searching (step 4), selecting (step 5) and assessing the quality (step 6) of the relevant secondary studies, and then extracting, analyzing (step 7) and synthesizing (step 8) the information provided by them in order to address the research questions.

In order to better address the goals of this work we make the following adaptations to the original protocol, as highlighted in Figure 1:

Figure 1: SLR process based on Kitchenham and Charter [30]. The highlighted steps are adaptations to the original process, following the guidelines from Bandara et al. [32] and Zhang et al. [33]

- Step 4: in order to better guide the search process during the *Identification of Research*, we extend and apply the *Quasi-Gold Standard* (QGS) strategy [33], which aims at the definition of an optimal search string by an iterative evaluation of its performance during the search process.
- Step 7: we perform in parallel with the data extraction step originally defined by Kitchenham’s guidelines, a *qualitative* data analysis process. The latter, based on the guidelines of Bandara et al. [32], is included as a means to address some of the identified research questions, to be discussed below.

In the following section, we elaborate on this (adapted) protocol before presenting the results of this study.

2.3. Research Questions & Protocol

The primary question addressed by this Tertiary Study is, as discussed in the introductory section: *What is the current state of research in Systems of Systems (SoS) Architecting as evidenced by the secondary studies in the field?* The following research questions and sub-questions were derived from it, by considering the challenges that SoS pose to the Software Architecture discipline and the terminology-related problems described in the introductory section.

RQ1 What secondary studies have been published in the area of SoS Architecting?

RQ1.1 Which definition of SoS from the literature is used?

RQ1.2 What is the scope of such secondary studies?

RQ2 Which phases/activities of Software Architecture Design are explicitly addressed by the secondary studies?

RQ3 Which Software Quality Attributes, and to what extent, have been considered by current studies in architecture design for SoS?

RQ4 What are the limitations of the current research in architecture design approaches for SoS?

RQ4.1 Were the research topics limited?

RQ4.2 Is there evidence that the use of SLRs is limited due to lack of primary studies?

RQ4.3 Is the quality of current SLRs appropriate?

RQ5 What are the semantic relations between SoS and other related concepts?

RQ5.1 Are these relations among the concepts consistent across the secondary studies?

RQ5.2 To what extent SoS — as a concept — overlaps with other commonly related terms such as CPS, SoS Engineering (SoSE), Smart-* Systems and Software-Intensive Systems?

As no prior tertiary studies in SoS architecting have been published, the first research question aims to identify the secondary studies that have been published so far in the area (*RQ1*), and in what scope (*RQ1.2*). Given the terminological issues discussed in the introduction, this research question also aims to identify which definitions of SoS concept have been used. This can help to assess to what extent the research in this field shares a common conceptual background (*RQ1.1*).

The second and third research questions (*RQ2*, *RQ3*) aim to identify either potential research gaps to be explored, or areas within SoS architecting that could require further secondary studies to fully understand the state of their research. Specifically, *RQ2* explores to what extent the different activities of SoS architecting have been covered so far; to this end we use as reference the architecting activities of the architecture design process proposed by Hofmeister et al. [34] and extended by Tang et al. [35]. *RQ3*, on the other hand, explores which quality attributes — which constitute the most important architecture drivers [36] — have been addressed in the same context. To that end, we use as reference the software product quality model defined by ISO/IEC 25010 [37].

Research question *RQ4*, and its three sub-questions are adopted from Kitchenham’s SLR in software engineering [22]. These questions seek to assess the limitations of the research in SoS architecting based on the number of primary studies, their scope, and overall quality.

Finally, *RQ5* aims to identify in which context the research on SoS-architecting has been undertaken, by distilling the relations between SoS as a concept, its variants from the literature (e.g. SISoS¹, SoCPS², etc.), and concepts commonly associated with SoS like IoT, CPS, CES (*Critical Embedded Systems*), and SoSE (*SoS Engineering*). This question also seeks to assess if there are discrepancies in the concepts among the different secondary studies e.g. due to bias by the authors (*RQ5.1*), and to assess to what extent such concepts overlap based on

¹Software-Intensive Systems of Systems

²Systems of Cyber-Physical Systems

Figure 2: Search process based on the proposed workflow by Zhang et al. [33], including the PICOC criteria proposed by Kitchenham et al. [30]. Additional steps, decisions, and outcomes included in the model are highlighted with a thick line.

the connections identified by the studies (*RQ5.2*).

2.4. Identification of Research — Search Process

The search process, as described in Figure 2, is an extended version of the workflow built around the concept of the *Quasi-Gold Standard* as proposed by Zhang et al. [33]. The *Quasi-Gold standard* (QGS) is defined as *a collection of relevant studies in the field or topic expected to be found by an ideal search strategy*. The QGS is used to calculate the *Quasi-Sensitivity* (*QS*) and *Quasi-Precision* (*QP*) metrics that refer to the proportion of *relevant studies* identified

by the search strategy, and the proportion of *retrieved studies* that are relevant, respectively:

$$QS = \frac{\text{Number of relevant studies retrieved}}{\text{Number of relevant studies}} \quad (1)$$

$$QP = \frac{\text{Number of relevant studies retrieved}}{\text{Number of studies retrieved}} \quad (2)$$

Zhang et al. provide a rigorous approach for designing search strategies in databases by systematically integrating manual with automated search. The aforementioned QGS is defined through manual search and subsequently used to iteratively improve automated search, based on its *QS* metric. We extend this strategy in the form of a search process by introducing an additional step and decision point to it (Step 4.3 and Decision *D1* in Figure 2, respectively). The additional step considers the refinement of the *PICOC*³ criteria based on the *QP* metric before attempting any further search string refinement. Decision *D1* triggers this refinement if it is found that *QP* is below an acceptable threshold. Our rationale for this extension is two-fold. First, the *PICOC* criteria, according to Kitchenham’s guidelines, are important means to frame the research questions, and therefore, the search strings. We, therefore, believe that they should be included in the iterative refinement of the search string. Second, we argue that a too low *QP* could actually indicate a problem within such *PICOC* definition; for example, a Population or a Context that leads to a search string with massive — and non-relevant — search results. For this reason, decision *D1* was also included as the means to refine the *PICOC* criteria towards a higher precision (smaller, but more representative set of studies), before further refining the search string for higher sensitivity.

2.4.1. Search results

To the best of our knowledge, no QGS for *Secondary Studies in Systems of Systems Architecting* has been defined yet by any published SLR. For this reason, we defined our own QGS following the steps shown in Figure 2 by

³Population, Intervention, Comparison, Outcomes, Context

manually identifying a set of representative secondary studies from the venues listed in Table 1. These venues were selected by searching secondary studies in Google Scholar using variants of the search string "Systems-of-Systems" AND "Systematic Review" (with and without hyphens, and with the acronyms SoS, SLR and SMS), identifying the most relevant ones based on their title and abstract from the returned results, and then identifying the corresponding journals and conferences where they were published.

The resulting QGS contains a total of 9 secondary studies as listed in Table 1:

Venue	Selected Studies
System of Systems Engineering Conference - SoSE	[S2]
Software Engineering and Advanced Applications (SEAA)	[S7]
European Conference on Software Architecture workshops (EC-SAW)	[S8][S12]
International ACM Sigsoft Conference on Quality of Software Architectures	[S10]
International Workshop on Software Engineering for Systems-of-Systems - SESoS	[S14][S3][S17]

Table 1: Selected venues for QGS manual search, and the resulting selected studies for inclusion to the QGS

2.4.2. Search String for Automated Search

Based on the research questions described in Section 2.3, a first version of the PICOC criteria shown in Table 2 was defined to guide the construction of the search string. As it can be seen by the defined Context, the first iterations of the protocol aimed to identify Secondary Studies in Software Architecting related not only to the concept of SoS, but also to other, commonly (but not formally) related terms. For the first iteration of the search process, the following databases were used for the automated search: *SpringerLink*, *Scopus*, *Science@Direct*, *ISI Web of Science*, *IEEE Digital Library*, and *ACM Digital*

Population	Secondary Studies on Systems of Systems Architecting
Intervention	Studies on architecting software systems self-identified as Systems of Systems
Comparison	A holistic comparison among the population to analyze collective quality, scope and impact of existing research in Systems of Systems architecting
Outcome	Integrative and Interpretative synthesis of the secondary studies towards identifying open research questions, gaps in the research domain, and best practices for secondary studies in the domain
Context	Secondary Studies reported between 2007 and 2018, in the domain of Software Engineering, which review either primary studies related to Systems of Systems Architecting, or primary studies on architecting-related concepts in domains commonly related to SoS: Federations of Systems, Ultra Large Scale Systems, Cyber-Physical Systems and Self-Adaptive Systems.

Table 2: First version of the PICOC criteria.

Library.

The search string was then set to⁴:

⁴For the sake of brevity, only the search string used for *IEEEExplore* is shown.

```

("system of systems" OR "systems of systems" OR "system-of-systems"
OR "systems-of-systems" OR "Federations of Systems" OR "Ultra
Large Scale Systems" OR "Cyber-Physical Systems" OR "Self-adaptive
systems")
AND
("systematic literature review" OR "systematic mapping review" OR
"systematic review" OR "systematic mapping study" OR "literature
analysis" OR "literature review" OR "paststudies" OR "subject matter
expert" OR "analysis of research" OR "meta-analysis" OR "structured
review" OR "review of studies" OR "empirical body of knowledge")

```

Given the small size of our QGS, we decided to set the QS threshold to 1.0 for the search process, and treat the QP threshold setting as a heuristic.

Iteration	Relevant Studies	Retrieved Studies	Relevant Studies Retrieved	QS	QP	Action
1	9	377	4	0.44	0.0106	Including Federations of Systems, Ultra Large Scale Systems, Cyber-Physical Systems and Self-Adaptive Systems.
2	9	223	4	0.44	0.0179	Including Architecture / Architecting
3	9	388	4	0.44	0.0103	Including smart-*
4	9	95	4	0.44	0.0421	PICOC's Context updated: Search string using only variants of SoS (smart-* removed).
5	9	106	7	0.78	0.066	Replacing 'systematic mapping review' with 'systematic mapping' (identified by data analysis)
6	9	142	9	1	0.0634	Inclusion of Wiley's database

Table 3: Search string refinement iterations

The results of the six iterations of the search process are shown in Table 3. As it can be seen, in the first three iterations, even after refining the search string, QP tended to be extremely low, as the number of studies that matched it was too high (for a secondary studies search), while QS remained the same. In the fourth iteration, QP was improved (without affecting the QS) by changing the *PICOC Context* to a narrower one:

Context	Secondary Studies reported between 2007 and 2018, in the domain of Software Engineering, which review either primary studies related to Systems of Systems Architecture, or primary studies of architecture-related concepts in other domains explicitly linked to SoS.
---------	---

In the fifth iteration both QP and QS were improved by changing one of the search terms, which was identified by analyzing the keywords of the QGS studies that were missed by the search string. Finally, in the sixth iteration it was possible to reach 100% QS by adding an additional database. Any further adjustment resulted in actually lowering QP without affecting QS ; we therefore set the QP threshold to its value from the last iteration (0.0634), and concluded the search process with an initial data set of 142 studies (July of 2018), as shown in Table 3. The final search string eventually used to retrieve this data set was:

```
( "system of systems" OR "systems of systems" OR "system-of-systems"
OR "systems-of-systems" )
AND
( "architecture" OR "architecting" OR "engineering" OR "development" )
AND
( "systematic literature review" OR "systematic mapping review" OR
"systematic review" OR "systematic mapping" OR "literature analysis"
OR "literature review" OR "past studies" OR "subject matter expert"
OR "analysis of research" OR "meta-analysis" OR "structured review"
OR "review of studies" OR "empirical body of knowledge" )
```

The evolution of the search strings and their adaptation for each database are available online⁵, under Appendix C.

2.5. Primary Studies Selection

A tertiary study, like any SLR, requires the explicit definition of *Inclusion* and *Exclusion* criteria as the means to assess the relevance of the secondary studies identified through the search process, to the research questions. The criteria for this study are defined as follows:

Inclusion Criteria. The retrieved study:

- IC1** Is a peer-reviewed SLR or SMS with clearly defined research questions, search process, data extraction, and data presentation.
- IC2** Is either a secondary study focusing on Architecting in the domain of SoS, or in an area related to SoS like Cyber-Physical SoS, SoS Engineering, etc.

Exclusion Criteria. The retrieved study:

- EC1** Does not meet the inclusion criteria.
- EC2** Appears in multiple reports (only the most recent one is considered).
- EC3** Is a tertiary study itself.

As all the results returned from the previous steps were in English, we did not define an EC to exclude works from other languages, as per usual practice.

Primary Studies Selection and Snowballing Results

As shown in Figure 3, and after automatically removing duplicate publications, the inclusion/exclusion criteria described in Section 2.5 were applied independently by the first two authors to the remaining 126 studies, and the outcome was registered — including the justification of the decision — on two separate spreadsheets (Filtered Dataset 1 & 2 in Figure 3).

⁵<https://github.com/search-rug/SoS-architecting-tertiary-study-protocol>

Figure 3: Filtering process

The outcome was merged and after the 5 disagreements were discussed and resolved, 18 secondary studies were selected for the final data set. Forward and Backwards Snowballing [38] was also applied to the 18 studies and no additional secondary studies were found by the former, but study [S14] was identified with the latter and added to the final dataset, for a total of 19 (the complete listing is available in Appendix A). The size of this final dataset was deemed sufficient for a tertiary study, considering that the similar studies mentioned in Section 2.1 have a median of 24 included secondary studies.

2.6. Quality Assessment

In secondary studies quality assessment serves, among other purposes, as an additional and more detailed criterion for the exclusion of pre-selected primary studies (experiments, case studies, etc) from the review. In contrast, it is common for tertiary studies not to exclude any secondary studies based on their quality, as such studies are instead interested in the overall view of the quality of the research in the area [31].

For this tertiary study, the quality assessment is used as the key element to address research question *RQ4*, related to the limitations of the current research, following the example of [31]. The first four quality assessment criteria (*-QAC4*) used for this study are therefore adopted from Kitchenham et al. “as is”. They are used by the CRD⁶ to assess potential systematic reviews for their inclusion into DARE⁷. Furthermore, *QAC5*, also from the DARE criteria, was included by following Cruze’s et al. [39] rationale regarding the importance of data synthesis as a quality factor of a systematic study:

QAC1 *Are the review’s inclusion and exclusion criteria described and appropriate?*

QAC2 *Is the literature search likely to have covered all relevant studies?*

QAC3 *Did the reviewers assess the quality/validity of the included studies?*

QAC4 *Were the basic data/studies adequately described?*

QAC5 *Was synthesis performed on the reported results?*

The scoring procedure considers a three-levels scale: Y(es) = 1.0, P(artially) = 0.5 and N(o, or unknown) = 0. Table 4 lists the criteria used for the evaluation. The evaluation process was carried by the first and second author independently and the results were compared in order to resolve any points of

⁶Centre for Reviews and Dissemination

⁷Database of Abstracts of Reviews of Effects

disagreement. A total of 24 disagreements, out of the 95 elements evaluated (5 questions for each one of the 19 studies) were identified, discussed and solved (See online⁸, Appendix D for the complete record).

Criterion	(Y)es	(P)artially	(N)o
QAC1	Inclusion and exclusion criteria are explicitly defined	Implicit inclusion/exclusion criteria	Inclusion and exclusion criteria are not defined, and cannot be inferred
QAC2	Searched 4 or more digital libraries and included additional search strategies	Searched 3 or 4 digital libraries with no extra search strategies	Searched up to 2 digital libraries
QAC3	Quality criteria explicitly defined and extracted for each secondary study	Quality issues of primary studies addressed by research questions	No explicit quality assessment
QAC4	Each primary study can clearly be traced from the information provided.	Only summary information is provided.	Results of individual studies are neither specified nor summarized.
QAC5	Data was extracted, summarized and interpreted.	Data was extracted and summarized but not interpreted.	Data was not summarized.

Table 4: Quality Assessment Criteria Score Schema based on Kitchehman et al. [31] and Cruzes et al. [39]

The outcome of this process will be discussed in the following section, in the context of answering *RQ4*.

2.7. Data Extraction and Qualitative Content Analysis

The data extraction step in Kitchenham et al.’s protocol in a tertiary study comprises the design of the forms to accurately record the information obtained from the secondary studies [30]. However, our combination of quantitative and

⁸<https://github.com/search-rug/SoS-architecting-tertiary-study-protocol>

qualitative research questions requires the extraction of concepts and relations out of the selected studies. For this purpose, we perform the qualitative content analysis approach of Bandara et al. [40] in parallel with Kitchenham’s form-based approach, as depicted in Figure 1 (Steps 7a and 7b).

More specifically, the form-based data extraction approach (Step 7b in Figure 1), which is more suitable to extract and encode the data required to address quantitative research questions, is used for *RQ1* and *RQ4*. Table 5 lists the fields used in this study for this purpose.

ID	Data collection variable	Related RQ
F1	Number of primary studies	RQ1
F2	Paper type	RQ1
F3	Systematic Review type (SMS/SLR)	RQ1
F4	Review scope	RQ1 (RQ1.2), RQ4 (RQ4.1, RQ4.2)
F5	Application domain of the study	RQ1 (RQ1.2)
F6	SoS definition used by authors	RQ1 (RQ1.1)
F7	Number of quality criteria evaluated in review’s assessment checklist	RQ4 (RQ4.3)
F8	Is such checklist derived from other previously published?	RQ4 (RQ4.4)
F9	Number of authors	RQ1
F10	Was snowballing performed?	RQ4 (RQ4.2)
F11	Review topic	RQ1
F12	Years covered	RQ1

Table 5: Fields for the data-extraction form

Bandara et al.’s qualitative content analysis approach (Step 7a in Figure 1), on the other hand, is used to address *RQ5*, as this type of question requires to distill words into content-related categories [41]. Likewise, *RQ2* and *RQ3* are addressed using this approach by inducing categories of quality attributes and architecture design activities which are then mapped to the models described in

Section 2.3. This qualitative content analysis approach comprises four phases:

1. Systematic identification of papers,
2. Preparation for Analysis,
3. Coding and Analysis, and
4. Writing and reporting of the findings.

Given that Phase (1) is already covered by Steps 1 to 6 of Kitchenham et al.'s protocol (see Figure 1), and Phase (4) by Steps 8 and 9, only Phases (2) and (3) of Bandara et al.'s protocol (shown as Step 7a in Figure 1) are actually performed. More specifically, Phases (2) and (3) are performed (for *RQ2*, *RQ3*, and *RQ5*) after the studies selection and before the synthesis and reporting phases within the SLR protocol, as described in the following.

2.7.1. Preparation for Analysis Phase

In this phase, the coding approach for the qualitative content analysis is defined, which could be *inductive*, *deductive* or a *combination* of both. Given that a tertiary study aims to explore the state of the research, and not testing a pre-defined hypothesis, an inductive approach is followed. In order to narrow the scope of the qualitative analysis and make the results easier to synthesize, a pre-codification schema is used for research questions *RQ2* and *RQ3*, based on two well-known models of architecture design phases and quality attributes: Hofmeister et al.'s Architectural Design Phases [34], including the extension proposed by Tang et al. [35], and the quality model defined by ISO 25010:2011 [37], respectively.

The coding for research question *RQ5*, despite also being performed using an inductive approach, does not make use of a pre-codified schema. Although there are conceptual models proposed in the domain of SoS, such as the SoS-based extension of the ISO/IEC/IEEE 42010 standard proposed by Gonçalves et al. [42], to the best of our understanding there are no widely accepted conceptual models that include the overlapping terminology *RQ5* aims for. For this reason, the coding process for this question starts with an empty category set.

2.7.2. Coding and Analysis Phase

Once the coding approach is defined, the coding and analysis process uses the secondary studies selected in the previous step as qualitative data through the first two stages of *grounded theory* [32], i.e., open and axial coding. Bandara et al.'s guidelines suggest using qualitative content analysis software packages to systematically perform such steps. For this tertiary study the *Atlas.ti*⁹ tool is selected for two main reasons: first, its concepts (e.g. codes, quotations, code networks, etc.) can be easily adapted to a grounded theory process, and second, its features of code co-occurrence search (e.g. to identify in which paragraphs two or more codes were assigned) bring a powerful tool to perform further quantitative analysis once the studies are coded.

The process of open and axial coding, performed with the support of *Atlas.ti* is as follows:

Open coding: identify and label, inductively, a set of concepts (codes) relevant to the research questions. For this, the excerpts related to the domains of the research questions (quality attributes, architecture design phases, and overlapping concepts) are identified, and then the codes are assigned and iteratively refined.

Axial coding: establish relations between the codes based on the evidence provided by the secondary studies. Such relations are defined in two steps: (1) by marking the quotations where the relation of two or more codes are identified, and (2) by defining explicit code-to-code relations within the *Atlas.ti* codes model, based on the subsequent analysis of the quotations identified in the previous step.

The outcome of the open coding phase comprises of the categories of codes for the concepts related to SoS, research topics, and quality attributes; all of them are assigned to their correspondent text segments within the secondary studies.

⁹Atlas.ti version 8.0: <https://atlasti.com/product/v8-windows/>

The axial coding outcome, on the other hand, comprises a series of semantic relations between *SoS* and other concepts identified in the open coding phase.

The process for addressing the posed research questions, based on the outcome of the Data Extraction and Qualitative Content Analysis phase, i.e. form-based data, codes and relations, is discussed in the following.

2.8. *Synthesis*

After extracting the data in the forms and coding schemata (codes and relations) developed through the open and axial coding processes, the next step is to synthesize them in order to address the research (sub-)questions defined in Section 2.3 — at least for the ones that can not be answered by simple data extraction and quantitative analysis.

More specifically, data for *RQ1.1* (SoS definition) and *RQ4.3* (quality of studies) are synthesized using simple counts and statistics from the form-based data, and the outcome of the quality assessment, respectively.

In order to address *RQ1.2* (scope of studies), and partially address *RQ4.1* (scope of research topics) and *RQ4.2* (completeness of primary studies), a set of hierarchical clusters is synthesized from the research scopes identified by the variable F_4 in Table 5. In order to obtain such hierarchical clusters that aim to provide a higher-level perspective of the different research scopes given to the selected secondary studies, we:

- Elicit a set of scope-categories from the final dataset of paper scopes (*Field F_4*).
- Define whole-to-part relations between categories.
- Assign one or more categories to each secondary study.
- Extract a hierarchy of clusters based on such categories.
- Create sub-clusters within the clusters if a subset of its constituents has a distinctive feature, e.g. a particular quality attribute within a cluster of studies with a scope on quality attributes.

- Visualize the cluster data (e.g. using an *Euler* diagram).

To answer *RQ2* and *RQ3* we follow an integrative synthesis [43] approach by mapping the secondary studies to the *quality attributes* and *architecting process models* described in Section 2.3. This mapping is made by matching the elements of the models (i.e. the architecting phases and quality attributes) with the codes assigned to each secondary study. Furthermore, it includes the context in which the linking code occurs, which could be either a reported work, or a research gap/direction. It is foreseen that by arranging these details in a timeline, not only the coverage of the mapping of the models, but also the evolution of the reported gaps will be revealed in order to address the questions mentioned above.

In order to address *RQ5*, we first do an integrative synthesis in a concept map of the semantic relations identified in the secondary studies through *axial* coding [44]. Then, from this raw relations map, repeating patterns and contradictions are identified and formally described as propositions and inconsistencies, respectively. Finally, these facts are represented and summarized in a higher-level, easier to follow, concept map based on which we provide answers to the sub-questions of *RQ5*.

3. Study Results

In the following section we discuss the results of the study as designed in the previous section. The discussion is organized around answering each of the research questions identified in Section 2.3.

3.1. *RQ1: What secondary studies have been published in the area of SoS architecting?*

In order to answer this RQ we first take a look at the demographic data of the selected secondary studies and their corresponding primary studies. Then we will move to the specific sub-RQs for a clearer image of the research field.

Figure 4: Nationality (based on their affiliation) of the researchers involved in the conduction of secondary studies. The overall number of co-authors, with and without duplicates, are shown side by side as a way to visualize to what extent authors are involved in other secondary studies. Authors with double affiliation in different countries are counted twice.

As described in Section 2.5, we found 19 secondary studies published between 2013 and 2018, comprised of 9 SLR and 10 SMS. Figure 4 shows Brazil as — by far — the most recurrent nationality among the co-authors, as well as a significant number of recurrent authors within the secondary studies published in this country. We also found that the distribution over time of the secondary studies, as illustrated in Figure 5, does not show a transition from SMS to SLR studies in the field with some notable exceptions ([S1] after [S16] and [S18] after [S2]). This signifies a field very much still in development since the focus appears to be still on mapping areas for new research through SMS studies, rather than consolidating research on the same topics by means of SLR studies. It is also worth noting that a large proportion of the secondary studies either does not acknowledge the existence of other related secondary studies (no outgoing edges in Figure 5), or only acknowledges the existence of the most popular (and older) ones. This suggests that although there is evidence of local (nation-wide) focus on the research topic, world-wide there seems to be a state of disconnectedness of the research efforts over time.

Figure 5: Timeline of secondary studies (SLR and SMS) in SoS Architecting. The average number of citations per year (cpy) is represented by the diameter of the nodes, and the citations between secondary studies by directed edges.

Focusing further on the citations, the two most cited secondary studies, as shown in Figure 5, are [S14] (with an average of 8 citations per year and 48 citations in total, as per October 2018 according to Google Scholar) and [S9] (with an average of 8.75 per year and a total of 35). They could be considered as the most influential in two different ways: the former, with a broad overview of the field, acts as a reference point for a number of follow-up secondary studies in Systems of Systems architecting and other related primary studies; the latter is also a reference point for studies that address — to some extent — the problem of describing the architecture of an SoS. At the same time it is worth noting that despite [S9] being the most cited secondary study on average per year, no other secondary studies (at least from our set) cites it. This seems to reinforce the idea that the evidence-based research in the field (by means of secondary studies) is seemingly disconnected, as discussed above. This is an issue that the community should work to address collectively, as we will discuss further in Section 4.

The secondary studies have reported, analyzed, and in most cases synthesized a significant amount of primary studies in the domain of Systems of Sys-

tems Architecting; this is summarized in Figure 6.

Figure 6: Timeline of primary studies covered by the selected secondary studies. [S13] does not provide details of the publication dates of its primary studies.

An approximate of 1739 primary studies¹⁰ published between 1991 and 2017 (not considering an outlier from 1969 included in [S16]) were covered by the selected secondary studies. However, it is important to consider that, in some cases, such primary studies are not from the domain of Systems of Systems architecting per se. For example, although the scope of [S18] is on approaches for requirements monitoring in SoS, the study was carried out by identifying the capabilities of generic (that is, not necessarily SoS-oriented) runtime monitoring solutions reported by primary studies, with respect to the characteristics of SoS. In a similar fashion, the primary studies reviewed by [S19] and [S6], belong to the domain of *Model-based Software Engineering* and *Critical-Embedded Systems*, respectively, and they are ‘imported’ in the domain of SoS under the interpretation given by the authors (e.g. in [S6] is stated that CES are “*among the most common examples of Systems of Systems*”).

¹⁰The exact number cannot be calculated, as some secondary studies either do not provide the full list of primary studies (e.g. [S15]), or the provided URLs with such lists are no longer working (e.g. [S10]), and hence the whole set of duplicate primary studies cannot be identified.

Regarding **RQ1.1**, which is related to the definition of SoS used, and based on our Inclusion/Exclusion criteria, all the selected secondary studies are centered on an architecting-related activity on either SoS, or on a type of system explicitly linked to it. Unsurprisingly, most of the secondary studies (14 out of 19) use the properties proposed by Maier [1] to define the SoS concept. Only one secondary study is based on an alternative set of features, namely *Autonomy*, *Belonging*, *Connectivity*, *Diversity* and *Emergence* proposed by Boardman and Sauser [12], while three actually use both, as shown in Figure 7. It is interesting though that the four remaining secondary studies that do not make any reference to an SoS definition actually belong to the group of studies that are not explicitly SoS-oriented, but rather in a domain linked to SoS: *Smart Cities* [S4], *Critical Embedded Systems* [S6], *Software Ecosystems* [S15] and *Industry 4.0* [S19].

Figure 7: Distribution of SoS term definition source per study

Regarding **RQ1.2** concerning the research scope of the secondary studies, this can be classified at three different layers, each containing the next, as depicted in the Euler diagram of Figure 8: secondary studies on general SoS development process models, which includes the architecting process as a phase; secondary studies on SoS architecting as a whole; and secondary studies focusing on a specific aspect of SoS architecting, namely application domains (as discussed above), architecting activities, quality attributes, development paradigms, ar-

Figure 8: Representation of the diversity of focus of the selected secondary studies, and their intersections. ES=Embedded Systems, I4.0=Industry 4.0, MDE=Model-driven Engineering.

chitecting activities, or an intersection between these. The secondary studies on the outermost level ([S2], [S11]), have *Process Models* for SoS as scope, where the design and evolution of SoS architectures is a phase within a more comprehensive process. Interestingly, these reviews were published after (and refer to) the reviews where the focus is SoS architecting as a whole ([S10, S14]), which are among the most cited studies.

Regarding secondary studies with a narrower scope, and starting with *Quality Attributes* (to be discussed in more depth in Section 3.3), two things become quickly evident: first, these studies are *independent* from application domains, and second, the importance that researchers have attributed to *interoperability* in the domain of SoS architecting. Except for one secondary study with a scope on quality attributes for SoS in general ([S3]), the remaining three secondary studies on Quality Attributes ([S7], [S13], [S17]) address different aspects of interoperability, including discovery, composition and integration. However, it is also somewhat surprising that these studies do not acknowledge the existence of each other, or even that of the earlier study on the same subject ([S7]), as

shown in Figure 5.

Regarding the studies with a scope on *Architecting Activities* (further discussed in Section 3.2), covered activities related to design (only in the context of Critical-Embedded Systems) [S6], architecture description [S9] and requirements. Requirements, in turn, are addressed from four different perspectives: mission specification approaches [S16], translation of capability objectives into requirements [S5], requirements monitoring at runtime [S18], and requirements in a particular application context: Smart Cities [S4].

With respect to application domains, most of the selected secondary studies do not focus on a particular one. In contrast to the long list of application domains identified in [S3] and [S10], only five studies defined, based on their main research question and inclusion/exclusion criteria, an explicit scope on: *Smart Cities* ([S4]), *Industry 4.0* ([S19]) and *Software Ecosystems* ([S15]). Nevertheless, *IoT* ([S12]) and *Embedded Systems* ([S15,S6]) when considered as high-level application domains for SoS, are also explicitly addressed by three of the secondary studies.

Regarding the development paradigms mentioned, it is also noticeable that despite the significant number of development process models for SoS identified by [S11], only *Model-Based Engineering* approaches are covered in greater depth by secondary studies: this is performed by ([S19]) for a specific application domain and by ([S8]) independently of application domains. This points to the potential need for further studies, both primary and secondary, in these areas, as we will discuss further in Section 4.

3.2. RQ2 - Which phases/activities of Software Architecture Design are explicitly addressed by the secondary studies?

To address this research question, we map the selected secondary studies on the architecting activities of *Analysis*, *Synthesis*, *Evaluation*, *Implementation*, and *Maintenance & Evolution* as described in Section 2.3. The coverage of such activities, which are identified within the studies through the open-coding process described in Section 2.7, are marked with the letter *E* in Figure 9. As

Year	Study	Analysis	Synthesis	Evaluation	Implementation	Maintenance/ Evolution
2013	S10	E	IR (AD)	E		
2013	S14		IR (AD)	E		E
2014	S8	IR (R)	IR (AD)		IR (MBD)	E
2014	S12			IR (TE)	IR (DDA)	
2014	S16					
2015	S1					
2015	S2	IR (R)	IR (D)			
2015	S3					
2015	S7				IR (DDA)	
2015	S9		IS (D)			
2015	S15					
2016	S17					
2016	S18					
2017	S5	IS (R)				E
2017	S11		IR (MBD)			
2017	S13					
2017	S19					
2018	S4	IS (R)				
2018	S6	E	IS (D)	E		

Figure 9: Coverage of architecting activities by the secondary studies

Legend: E = Explicit Coverage, IS = Implicit coverage by research scope, IR = Implicit coverage by addressing the activity in a section of the study (R=Requirements, AD = Architecture Description, D = Architecture Design, MBD= Model-Based Design, TE = Testing Environments, DDA = Detailed Design Artifacts), = Research Gap reported for the corresponding activity.

it can be seen, only five of the included secondary studies ([S5], [S6], [S8], [S10], [S14]) make explicit reference to at least one of these architecting activities, while none of them actually covers all the core ones (*Analysis*, *Synthesis* and *Evaluation*) proposed by Hofmeister et al. [34].

This phenomenon, which was also reported by [S14] back in 2013, could mean that the lack of a complete and mature process for software-intensive SoS architecting is still an open issue. However, it is worth mention that there is work towards addressing this issue [45] not included by any of the secondary studies.

In order to provide a more complete, in-depth mapping of the architecting activities, we also assess the coverage of research topics that are *implicitly* related to them based on their definitions [34, 46], e.g. requirements-related topics

for Analysis, and design-related topics for Synthesis. The coverage of such topics, in turn, is identified at two different levels: when used as the main scope of the study (IS) based on its inclusion/exclusion criteria (e.g. requirements for SoS), or when addressed by a section of it (IR), as can be seen in Figure 9. Moreover, we use (●) to denote if the study identifies a research gap that is relevant for the corresponding architecting activity.

Based on the outcome of this mapping, we argue that *Analysis*, with respect to requirements, and *Synthesis* can be considered as sufficiently addressed by the secondary studies. Such phases have been not only continuously addressed through most of the secondary studies time-line, but are the only ones covered as the main scope of secondary studies. As examples, we mention [S5], whose scope are requirements, related to the *Analysis* phase; and [S9], with a scope on SoS software architectures description, which is related to the *Synthesis* activity.

For Evaluation, one of the earliest secondary studies ([S14]), concluded that this activity was an open issue as there was no consensus on what should be considered for it when done in the context of an SoS. Furthermore, the lack of coverage of this activity in 2013 is evidenced in the architecting tasks classification described in [S10]. In this classification, there is a ratio of approximately 1:10 between the primary studies with a focus on *Analysis/Evaluation* category and the primary studies with a focus on *Design/Analysis/Modelling/Documentation* category. Four years later, no further secondary studies reported progress in this respect — and therefore the aforementioned consensus might not be built yet. In fact, [S5] concludes that more techniques for the verification of requirements in SoS (i.e. *Evaluation*) are required.

When it comes to Maintenance/Evolution, only a few primary studies on development approaches, and methodologies are reported by [S5, S8, S14] respectively. Furthermore, the research gap identified by [S14] in 2013, regarding the lack of publications reporting concrete observations and experiences on the evolution of SoS architectures, is not addressed by what has been reported by the following secondary studies. However, it is interesting that [S5] also reports as a challenge for the future the need for more effective ways of achieving

SoS evolution, which could be difficult to address given the lack of studies, and therefore the lack of concrete issues on evolution, as mentioned above.

With respect to Implementation, only two studies ([S12],[S8]) make reference to detailed-design level strategies, particularly on how to address the integration, discovery and interoperability of the constituent systems of an SoS. On the other hand, code generation according to the *Model-Driven Development* (MDD) paradigm, is only discussed in [S8]. However, at this point we need to mention that given that MDD and other *Model-Driven Engineering* approaches are also discussed by other secondary studies ([S10,S11,S14,S19]), the true coverage of this architecting activity might be simply hidden by the lack of details (in the secondary studies) on which model (to code) transformations are actually performed by such approaches in an SoS context.

Overall, and based on the sparsity observed in Figure 9, it is fair to conclude that the architecting activities of *Evaluation*, *Implementation* and *Maintenance/Evolution* are not sufficiently discussed by the literature on the field.

3.3. RQ3 - Which Software Quality Attributes, and to what extent, have been considered by current studies in architecture design for SoS?

Figure 10 summarizes the coverage of the *Quality in Use* and *Product Quality* characteristics from the ISO/IEC 25010:2011 quality model [37] in the selected studies. The two studies with the highest number of quality characteristics, [S14] and [S3] — the latter being an SLR on quality attributes for SoS, have in common a research question that explicitly addresses quality attributes for SoS. Both studies, with a two years difference between them, show consistently how the covered Quality in Use and Product Quality characteristics remained almost the same in this period of time. On the other hand, the lack of references to quality characteristics on [S15] and [S19] could also be interpreted as a potential research gap for SoS architecting in application domains like *Industry 4.0* and *Software Ecosystems*.

Focusing further on the quality characteristics, the most frequently addressed one from the Product Quality model is *Compatibility*, as most of the secondary

YEAR	STUDY	Quality in use model					Product quality model								
		Satisfaction	Effectiveness	Efficiency	Context Coverage	Freedom From Risk	Compatibility	Functional Suitability	Usability	Performance Efficiency	Security	Portability	Reliability	Maintainability	
2013	S10	0%	0%	0%	0%	33%	50%	33%	0%	100%	20%	0%	25%	40%	7
2013	S14	0%	0%	100%	50%	33%	50%	0%	100%	100%	100%	33%	25%	100%	9
2014	S8	0%	0%	0%	0%	33%	50%	0%	100%	0%	100%	33%	0%	0%	5
2014	S12	0%	0%	0%	0%	0%	50%	0%	0%	0%	0%	67%	0%	0%	2
2014	S16	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0
2015	S1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	0%	0%	1
2015	S2	0%	100%	100%	0%	33%	50%	0%	0%	0%	100%	0%	100%	0%	4
2015	S3	0%	100%	0%	50%	33%	100%	0%	17%	100%	20%	33%	25%	100%	9
2015	S7	0%	0%	0%	0%	0%	50%	0%	0%	0%	0%	0%	0%	0%	1
2015	S9	0%	0%	0%	0%	33%	50%	67%	0%	0%	20%	0%	25%	100%	6
2015	S15	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0
2016	S17	0%	0%	0%	0%	0%	50%	0%	0%	0%	0%	0%	0%	0%	1
2016	S18	0%	0%	0%	0%	33%	0%	0%	0%	0%	100%	0%	0%	0%	2
2017	S5	0%	0%	0%	100%	0%	50%	0%	0%	0%	0%	100%	0%	100%	4
2017	S11	0%	0%	0%	0%	0%	50%	0%	0%	0%	0%	0%	0%	0%	1
2017	S13	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	33%	25%	0%	3
2017	S19	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0
2018	S4	0%	0%	0%	0%	0%	50%	0%	0%	0%	0%	0%	0%	0%	1
2018	S6	0%	0%	0%	0%	33%	0%	33%	0%	100%	100%	0%	100%	20%	6
		0	2	2	3	8	14	3	3	4	8	6	7	6	

Figure 10: Quality characteristics from the ISO/IEC 25010:2011 model covered from the selected secondary studies. The heat scale colors correspond to the percentage of sub-characteristics covered by each study.

studies refer to *Interoperability* in the context of SoS. Interestingly, this seems to be consistent with the findings discussed in Section 3.1, where the importance of *Interoperability* is evidenced by the number of secondary studies on topics related to this quality attribute. On the other hand, the most frequently addressed quality from the Quality in Use model, is *Freedom from Risk*, which is not surprising as *Safety* (or as labeled by ISO/IEC 25010, *Health and safety risk mitigation*) is the main goal of most SoS, as stated by [S3].

When it comes to the quality characteristics with the lower coverage, it could be argued that Quality in Use characteristics such as *Satisfaction*, *Effectiveness*

and *Efficiency* are rarely addressed by the secondary studies as they measure the achievement degree of specific user goals, and not system-wide ones. However, on the Product Quality characteristics side, it is surprising the low degree to which *Functional Suitability* — which is related to *Functional Completeness* and *Functional Correctness* [37]— has been covered. Even though three secondary studies do mention *Functional Correctness* ([S6], [S9], [S10]), they are limited to minor occurrences of this quality attribute within the primary studies. We find this very disappointing as such low coverage could mean that the correctness of the emergent capabilities of an SoS, and the fulfillment of its global mission could, are subjects that require further research.

Looking at the quality characteristics mentioned in the context of research directions, two secondary studies discussed *Reliability*, *Adaptability*, and *Flexibility*. Study [S3], pointed out how the concept of *Reliability* cannot be fully applied in the context of SoS, given the complexity of identifying a mission-failure status in a distributed system. However, given the number of references to *Reliability* in subsequent studies, this issue might not have been sufficiently addressed. Study [S5] pointed out the importance of *Adaptability* and *Flexibility* considering the evolutionary nature of an SoS. Likewise, this is an important observation given the lack of coverage on the *Context Coverage* quality characteristic by the rest of the studies.

In summary, these results suggest that when it comes to quality attributes, the literature has been focused mostly on the attributes concerned with the way the constituent systems cooperate and with the risks that emerge from such cooperation. However, quality attributes concerned with the fulfillment of the mission(s) of the SoS and its ability to adapt to new ones seem to require further coverage.

3.4. RQ4 - What are the limitations of the current research in architecture design approaches for SoS?

Our results regarding **RQ4.1** show that the research topics addressed by the secondary studies cover a wide variety of scopes within the domain SoS

architecting, including Application Domains, Quality Attributes, and Architecting Activities, as described in Section 3.1. However, by comparing such research scopes with what has been reported by the earlier (and broader) SLRs ([S10],[S14]), it can be seen that the number of research topics addressed by SLR studies are still limited. Notable examples are the lack of secondary studies on highly relevant quality attributes for SoS like *Flexibility*, *Security* and *Evolution*, the latter being also one of the key characteristics postulated by Maier [1].

Although this limitation could be attributed to the way the SMS and SLR studies have been undertaken in this area, as also discussed in Section 3.1, a limited number of primary studies is another likely reason. Unlike the broader SLRs that covered a significant amount of primary studies (e.g. [S10] with 194 and [S14] with 60), SLRs of important application domains like *IoT* ([S12]) and development paradigms like *MDD* ([S8]) covered only very few primary studies (5 and 12, respectively). This evidence suggests, as an answer to **RQ4.2**, that the use of the SLR study type in more specific topics of SoS architecting might be limited due the lack of primary studies.

Regarding **RQ4.3**, related to the quality of the secondary studies, the DARE scale is applied for SLR and SMS studies separately, as the quality assessment of the primary studies (*QAC3*) is not expected for the latter. As can be seen in Table 6, most of the SMS obtained a high score based on the proposed quality assessment criteria. The most common drawback, according to these scores, is the difficulty or impossibility of tracing back the primary studies from the conclusions. SLRs, on the other hand, although with similar scores for *QAC1*, *QAC2*, *QAC4*, and *QAC5*, on average obtained a relatively low score (Table 7), as most of them (with the exception of [S8] and [S17]) do not perform *any* quality assessment to the primary studies (*QAC3*), to the surprise of these authors. It is also worth noting that this phenomenon does not appear to be related with the number of primary studies selected by each SLR, as the quality assessment was omitted even on SLRs with only a few studies. However, the low scores in *QAC3* could be partially explained by the fact that all studies, with the exception of [S18] have appeared in conference proceedings, where space is

Type	Study	QAC1	QAC2	QAC4	QAC5	Final Score
SMS	S2	0.5	0	0	1	1.5/4
SMS	S6	1	1	0.5	1	3.5/4
SMS	S7	1	1	0.5	1	3.5/4
SMS	S11	1	1	1	1	4/4
SMS	S12	1	1	1	1	4/4
SMS	S15	1	1	0.5	1	3.5/4
SMS	S16	1	1	1	1	4/4
SMS	S19	1	0.5	0.5	1	3/4
<i>Sum:</i>		7.5	6.5	5	8	

Table 6: Quality Assessment scores of selected SMS.

Type	Study	QAC1	QAC2	QAC3	QAC4	QAC5	Final Score
SLR	S1	1	1	0	0.5	1	3.5/5
SLR	S3	1	1	0	0.5	1	3.5/5
SLR	S4	1	0	0	1	1	3/5
SLR	S5	1	0	0	1	0.5	2.5/5
SLR	S8	1	0.5	1	1	1	4.5/5
SLR	S9	1	0.5	0	0.5	1	3/5
SLR	S10	1	1	0	0.5	1	3.5/5
SLR	S13	0	0	0	1	1	2/5
SLR	S14	0	1	0	0.5	1	2.5/5
SLR	S17	0.5	1	0.5	1	1	4/5
SLR	S18	1	1	0	1	1	4/5
<i>Sum:</i>		8.5	7	1.5	8.5	10.5	

Table 7: Quality Assessment scores of selected SLR.

limited and therefore quality assessment is sacrificed for this purpose.

3.5. RQ5 - What are the semantic relations between SoS and other related concepts?

In order to identify the semantic relations between SoS and other related concepts, we performed *axial coding* based on the statements made in the sec-

ondary studies¹¹, as described in Section 2.7. After identifying the relations, we combined them in the raw concept map of Appendix B; subsequently we analyzed them looking for matching or complementary relations, contradictions, or logical inconsistencies. The results of this analysis, i.e. the semantic relations between the different concepts, are illustrated as a UML class diagram in Figure 11.

Figure 11: Concept model derived from the relations identified by the *Axial Coding* process. The (*) in the class name denotes that it represents only *some* instances of it, e.g., the class *Embedded System* and *Cyber-Physical System* only represent the *ES* and *CPS* that are *Constituent Systems* of an *SoS*.

¹¹The listing of the extracted relations is also available in the online repository at <https://github.com/search-rug/SoS-architecting-tertiary-study-protocol>

Based on these semantic relations, we address first **RQ5.1**, which is related to the consistency of such relations across the secondary studies. Specifically, we discuss our findings with respect to the relations between SoS, *SoS Engineering* (*SoSE*), *Critical-Embedded Systems* (*CES*), *Embedded Systems* (*ES*) and *Cyber Physical Systems* (*CPS*), whose relations showed no logical inconsistencies as a series of *propositions*. Consequently we examine the relations between SoS and the concepts of *Adaptive Systems* (*AS*) and *Software-Intensive Systems* (*SiS*) separately, as they exhibited a number of *inconsistencies*.

Proposition 1. *An ES, and by extension a CES, can be seen as a constituent system of an SoS, and as a whole SoS.* [S8] and [S14] state that both *ES* and *CES* are common constituent systems of SoS in *Defense*, *Avionics*, *Automotive* and *Military* application domains. In fact, seen as SoS *constituents*, there are no conceptual differences between *CES* and *ES* beyond the critical constraints of the environment they are part of. On the other hand, [S6] states that *CES* are common examples of SoS; this is further illustrated by [S3], which states that SoS is an architectural approach for developing *Automotive Embedded Systems*.

Proposition 2. *A Cyber-Physical System, depending on its scale, could be an SoS (macro-scale) or an Embedded System (micro-scale). A macro-scale CPS, in turn, can be made out of other CPS.* [S12] states that, at micro-level scale, a *CPS* can be an *ES*. On the other hand, both [S12] and [S3] state that the *CPS* are potential SoS — when created at a large scale — which, in turn, could also be made out of other *CPS* ([S12]). Furthermore, the fact that *Automotive*, *Airport* and *Robotic Systems* are mentioned as examples of both *CPS* ([S3]) and SoS ([S14]) corroborate the classification of *CPS* as a type of SoS.

Proposition 3. *SoSE is a holistic approach for the development of SoS, which implicitly considers the architecting process.* The definitions of *SoSE* given by [S5] and [S11] describe it as an area that deals with the application of *Systems Engineering* processes for *SoS*, comprised of a series of core elements (process, methods and techniques). Some of these elements, in turn, are re-

lated with the development and evolution of functional architectures for SoS, as pointed out by [S5] and [S9].

Inconsistency 1. *Adaptive Systems-of-Systems are proposed as a particular type of SoS even though SoS are considered as inherently Adaptive.* On the one hand, [S11] states that an SoS is inherently an *Adaptive System*, whereas [S19] proposes the concept of *Adaptive Systems of Systems* in the context of *Smart-Manufacturing Systems*.

Inconsistency 2. *Software-Intensive Systems of Systems (SiSoS) are proposed as particular types of SoS even though SoS are considered as inherently Software-Intensive, or as an approach for creating Software-Intensive Systems.* [S11] describes SoS as large, complex software-intensive systems, whereas [S14] presents it as an approach for the creation of *SiS*; on the other hand, [S3] and [S8] make use of the concept *Software-Intensive System of Systems (SiSoS)*, as a particular type of SoS.

Regarding **RQ5.2**, and based on the propositions previously discussed and the proposed concept map, SoS, as a research field, seems to overlap with CPS and ES research, to a different extent. As described in *Proposition 2*, the same examples are used to illustrate both a CPS and an SoS: *Robotic, Automotive* and *Airport Systems* are mentioned for both cases.

Furthermore, it is interesting that like CPS, the concept of SoS has been used in systems within two different scales: both at a macro scale, in application domains like smart cities (a common example of SoS), and at a micro-scale, as an approach for architecting ES, which based on *Proposition 1* in some cases can be seen as SoS. To a lesser extent — for systems at a micro-scale level, an overlap between SoS and ES research might exist too. In general ES are *constituents* of an SoS, and are only considered as SoS for special types of ES such as *Embedded Automotive Systems*. In the case of SoSE, there is no conceptual overlap with SoS, as the former is an area that addresses the latter, which is a type of system. However, given *Proposition 3*, when it comes to SoS *architecting* as a research topic it clearly overlaps with SoSE, the former being part of the latter.

Overall, these results suggest that most of the research in SoS architecting

has consistently used the concept of SoS, but the openness of its interpretation has led to its application in a broad spectrum of areas, ranging from micro- to macro-scale systems. This might be actually limiting the development of general architecting approaches for SoS. Furthermore, it may be promoting the emergence of possibly redundant terminology in the field as evidenced by the inconsistencies identified above.

4. Discussion

In this tertiary study, we reviewed 19 secondary studies in *SoS architecting*, in order to address the question: *what is the state of the research in systems of systems architecting as evidenced by secondary studies?* Our tertiary study revealed a fair number of secondary studies on the field addressing a wide variety of topics, including high-level, holistic SoSE approaches, architecting activities, development paradigms and particular application domains. Most of the selected secondary studies (14) are using Maier’s definition [1] of the concept: a system, regardless of its complexity or geographic distribution (i.e. its scale), can be a SoS as long as its constituents are *operationally* and *managerially independent*. Out of the remaining secondary studies (5), all of which are centered on concepts explicitly linked to SoS such as *Smart Cities*, *Critical-Embedded Systems*, *Software Ecosystems* and *Industry 4.0*, most cases (4 out of 5) do not adopt any formal definition of the concept.

One of the most striking findings is that *a significant number of secondary studies*, despite having similar or related research scopes, *does not acknowledge the existence of the other secondary studies in the same area as related works*. This suggests that the SLRs in the field, for the most part, are not emerging as a consequence of preceding mapping studies, as prescribed by the *EBSE paradigm* [22]. Moreover, the uneven coverage of areas reported by the secondary studies, suggest that *the mapping studies are not sufficiently guiding the development of the primary research* either. Especially the uneven coverage of the architecting phases (**RQ2**) and quality attributes (**RQ3**) unveiled by

this tertiary study, could be directly attributed to the under-usage of mapping studies as a resource to identify areas with little research.

However, the broad variety of topics that still need to be addressed indicates that these secondary studies, and especially the mapping studies, could still have a key role in the development of the field as starting points of future research initiatives. This could be a further motivation for strengthening the community towards a more organized development of the field.

Based on our findings, the phases of *Evaluation*, *Implementation* and *Maintenance of SoS*, and the quality attributes of *Adaptability* and *Flexibility* would be important topics to be addressed in the future due to their insufficient coverage in the literature. Moreover, the fact that *the quality of the research* (primary studies) in the area is *effectively still unknown* (**RQ4**) creates the need for further systematic studies, and especially SLRs in the area.

At the same time, it is probably equally, if not more important to have a discussion, as a community, on how the terminology is defining the research boundaries in the field. In this tertiary study we discovered that a set of recursive part-whole relations arise by combining the semantic relations between SoS and other concepts like CPS and ES, as extracted from the secondary studies. That is to say, concepts like CPS and ES can be considered (depending on their scale) both as SoS and as parts of SoS. For example (given the concept map of Figure 11), a hypothetical Airport System could be considered as both a (macro-scale) CPS and a SoS. This system, in turn, could be made out by other micro-scale CPS, each an ES, that could be also considered, in turn, as another SoS (e.g. an Automotive Embedded System). The aforementioned overlapping relations identified in the domain of SoS could be interpreted, as Henshaw suggested [21], by the concept of SoS not representing a particular class of System on its own right, but rather another viewpoint of a wide variety of complex systems with a common set of characteristics. However, such variety may in practice represent complications for effective research in the field: *SoS is potentially a rather too abstract and broad concept to be approached in a systematic and holistic manner.*

When it comes to SoS architecting in particular, we argue that *the broadness of the SoS term may actually be preventing the development of general-purpose SoS architecting approaches*. For instance, it is hard to define a single architecting process suitable for both a macro-scale software-intensive SoS (e.g. a Smart City) and a micro-scale one (e.g. an Automotive Embedded System) regardless of their common characteristics. At the same time, the way the concept of SoS is being used by researchers, e.g. with the use of *ad-hoc* and sometimes redundant derivative terms, as discussed in Section 3.5, may also be contributing to overlapping and scattered research efforts. The overlap between SoS and CPS discussed in **RQ5**, suggests that there may be researchers actually contributing to the problem of architecting SoS, but keeping their work under the umbrella of narrower terms like CPS or ES.

On the other hand, the openness of SoS as a term, and the bias induced by the discipline it is used from (e.g. Systems Engineering or Software Engineering), seems to be inducing the emergence of non standardized terminology of the form *X-SoS*, like *IoT-based SoS*, *SiSoS*, *CP-SoS*¹² and *Adaptive-SoS*, also in an attempt of narrowing it down as a concept. Moreover, even though these emergent terms make perfect sense in a primary study, they may be creating terminological inconsistencies from a community-wide perspective like the ones discussed in **RQ5.2** due to undesirable cluttering of the field. Consequently, we argue that when it comes to SoS architecting-related research, *a standardized, fine-grained classification of SoS based on the nature of their constituents, could lead to a more efficient concentration of research efforts*.

Turning now to a different perspective on the SoS-terminology discussion, it is worth asking: to what extent can the systems identified (by primary or secondary studies) as SoS be really classified as such? That is to say, to what extent the constituents of these systems have the characteristics proposed by the most

¹²*CP-SoS*, or *Cyber-Physical Systems of Systems* despite being an actual concept derived from SoS was not included in the concept map as it was only mentioned in [S12] as a project, not as a concept.

commonly used definitions of SoS (**RQ1.1**) like *Autonomy*, *Emergence* [12] and *Operational/Managerial Independence* [1]? In this context, it was a surprising finding that concepts like *Sensor* or *Actuator* are referred to in the literature as constituents of a SoS (see Figure 11). Such concepts, that are not likely to be independent under any sensible interpretation of the term¹³ stress the importance of addressing this question. More importantly though, this suggests that Maier’s concern about *misclassification* [1] as the origin of many problems in SoS development may still be very well relevant. Given that assessing this misclassification problem goes beyond the scope of this tertiary study, we propose two questions to be addressed in future surveys of the field: (1) *which properties are being considered by researchers and practitioners as essential (i.e. sufficient and necessary) to classify a system as a SoS?*, and (2) *which SoS features are motivating the adoption of SoS as an architecture style?* The answers to these questions could provide further insights on the findings of this tertiary study.

5. Threats to validity

In this section we present potential threats to the validity of this study, and our actions towards their mitigation. The discussion is based on the guidelines for addressing threats to the validity of secondary studies provided by Ampatzoglou et al. [47]. Threats in this context can be classified as *Study Selection Validity*, *Data Validity* and *Research Validity*, as follows:

Study Selection Validity. Threats in this category concern the initial steps of the review, and may lead to either the omission of relevant studies, or the inclusion of irrelevant ones. These potential threats include a limited selection of the digital libraries, ineffective search strings, and bias in the selection of the studies that

¹³Although sensors and actuators may arguably be managerially independent, in general they are not operationally independent as they do not have a purpose (mission) on their own apart from reporting their sensed data to/acting on instructions from a “master” system, respectively.

are included in the review. In order to mitigate the first two, we performed the search in seven of the most important digital libraries in scientific research, including SCOPUS which indexes papers from several publishers. Furthermore, we systematically refined not only the search string, but also the *PICOC* criteria it was based on (as discussed in Section 2). During this systematic refinement process, we assessed the performance of the search strings in terms of *precision* and *sensitivity* using a *Quasi-Gold Standard* built through a manual search. Moreover, we performed forward and backward snowballing on the secondary studies selected after applying the inclusion/exclusion criteria. To address the third type of potential threat (author bias during the selection process), the inclusion/exclusion criteria were applied independently by the first two authors, and then any disagreements were resolved by consensus.

Data Validity. *Unverified data extraction* and *author bias* are among the most important threats to the validity of the data, as these could lead to unreliable results and conclusions. To mitigate author bias in the quality assessment, the first two authors performed this step independently, following the standardized evaluation criteria (DARE) selected for the review protocol. Subsequently, any discrepancies in the score were further discussed and then re-calibrated by consensus. Data Extraction and Qualitative Data Analysis, on the other hand, were performed by the first author and verified by the second. In this regard, we documented every data transformation, so it is possible to trace it back and forth from the synthesis to the corresponding secondary study. As a notable example, the concept map used to address *RQ5* can be traced back to the statements it is made of, which, in turn, can be traced to their corresponding quotes within their origin secondary study. All the relevant material are available online¹⁴. On the other hand, as SLR and SMS are state-of-the-art snapshots from different periods, the inclusion of studies with *Outdated facts* could be a threat to the validity of the data, and hence to the validity of the results of the synthesis

¹⁴<https://github.com/search-rug/SoS-architecting-tertiary-study-protocol>

step. To mitigate this threat, which is inherent to any tertiary study, we are considering time as a dimension within our Synthesis process, as can be seen on *Section 2.8*. For instance, the codes extracted for *RQ2* and *RQ3* in the *Qualitative Data Analysis* are arranged as a timeline so the evolution of the coverage of the architecting activities and quality attributes, and their reported gaps through time, can be traced for synthesis purposes.

Research validity. Given the combination of qualitative and quantitative data analysis and synthesis required to answer the proposed research questions, *repeatability* is of particular concern for this study. To mitigate it, we defined a detailed protocol (see *Section 2*) based on well-established guidelines [48, 32] for systematic literature reviews. Furthermore, we made all the intermediate outcomes of the process publicly available¹⁵: search process history (search strings and their results), pre- and post-consensus data (studies selection and quality assessment), form-based data, and quotations-codes tuples, among others.

6. Conclusion

The term *Systems of Systems (SoS)* has already been applied to a wide range of application domains like defense, energy systems, manufacturing and health-care. In absence of a standardized definition for this concept, the research community has been working with it based on a widely accepted set of distinctive characteristics proposed in literature [1, 12]: *operational* and *managerial independence*, *diversity*, *emergence*, *connectivity*, and *belonging*.

Under the perspective of *SoS* framed by these features, a significant amount of research related to *SoS architecting* has been reported by a series of systematic literature reviews and mapping studies. This work builds on the findings of these secondary studies in order to contribute *a tertiary study on SoS architecting*. Our work therefore aims to address the research question *what is the*

¹⁵ibid

state of the research in SoS architecting? by systematically reviewing available secondary studies in the field.

An important element considered by this tertiary study is the variety of industrial sectors and research areas addressed by the included secondary studies. Despite the wide adoption of characteristics like the ones of Maier or Boardman and Sauser, such variety has led to an increasing emergence of new, and sometimes unclear, terminology in the field; this makes it difficult to assess the field's boundaries. For this reason, in the included secondary studies we looked not only for evidence about what research in the field has been covered so far, but also how the terminology related or derived from *SoS* is guiding or limiting further developments. More specifically, this tertiary study addressed its main research question by:

1. Analyzing and synthesizing the demographic data of the selected secondary studies, including the categorization of their scopes, the references between SMS and SLR through time, and the definition of the *SoS* concept used by their authors (**RQ1**).
2. Assessing the overall coverage of existing research reported by the secondary studies on two of the most important elements of an architecting process: architecting activities and quality attributes (**RQ2** and **RQ3**).
3. Assessing the limitations of the current research, particularly regarding the overall quality of the studies (**RQ4**).
4. Distilling a concept map (Figure 11) that reflects the community-wide understanding of the concept of *SoS* and its relations with other commonly used terms (**RQ5**).

We summarize further the most important findings of this tertiary study. Given the evidence provided by the secondary studies, in this study we find that the architecting activities related to *Evaluation*, *Implementation* and *Evolution* of *SoS* architectures are by large the least covered ones, and therefore require

further research. Regarding quality attributes, the same applies to the ones concerned with the fulfillment of the mission of an *SoS* (e.g. *Reliability* and *Functional Suitability*), and its ability to adapt to new ones (e.g. *Flexibility*). More importantly though, this uneven coverage of topics in combination with our findings on the disconnectedness between secondary studies, lead us to argue that the research in the field would have benefited (and could benefit in the future) by a more extensive use of the mapping studies to direct the research in the field, as suggested by [22].

Regarding our terminology-related findings, the most important suggestions of the reported concept map are: (1) the concepts of *SoS* and CPS seem to overlap significantly as they are used in the same types of micro- and macro-scale systems and (2) the *ad-hoc* terminology that is emerging in the field, arguably driven by the openness of the concept of *SoS*, is in some cases inconsistent or redundant, from a community-wide perspective. Given these findings, we argue that in order to concentrate the research efforts more efficiently in the field, it is necessary to define a standardized, fine-grained classification of *SoS* based on the nature of their constituents. Such classification, which would be complementary to the one of Maier, based on the management approaches of SoS [1], would ideally include details on how the properties of managerial/operational independence and emergence are applicable to each type of system under consideration.

Future work will be oriented towards two complementary directions. The first will be the identification of more up-to-date trends, insights and correlations between the findings of this study, regarding the research limitations, architecting activities and quality attributes. The second will be an evidence-based distillation of a classification schema for *SoS*, including the elicitation of the interpretation of concepts like ‘independence’ and ‘emergence’ by the community. These research directions will be followed by means of empirical research based on primary studies in the field, with either scoping studies, or SLR, as applicable.

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

The authors would like to thank Anja Reuter for her feedback on the manuscript.

References

- [1] M. W. Maier, Architecting principles for systems-of-systems, *Systems Engineering: The Journal of the International Council on Systems Engineering* 1 (4) (1998) 267–284.
- [2] S. Y. Diallo, C. J. Lynch, R. Gore, J. J. Padilla, Emergent behavior identification within an agent-based model of the ballistic missile defense system using statistical debugging, *The Journal of Defense Modeling and Simulation* 13 (3) (2016) 275–289.
- [3] T. Peugeot, N. Dupin, M.-J. Sembely, C. Dubecq, Mbse, plm, mip and robust optimization for system of systems management, application to sccoa french air defense program, in: *Complex Systems Design & Management*, Springer, 2017, pp. 29–40.
- [4] A. Alfieri, M. Cantamessa, F. Montagna, The sos approach for lean manufacturing systems, *International journal of technology management* 57 (1/2/3) (2012) 149–165.
- [5] U. Rauschecker, S. J. Ford, N. Athanssopoulou, Developing a vision for multi-site manufacturing system of systems, in: *Enabling Manufacturing Competitiveness and Economic Sustainability*, Springer, 2014, pp. 79–84.
- [6] D. Ki-Aries, S. Faily, H. Dogan, C. Williams, System of systems characterisation assisting security risk assessment.
- [7] A. M. Zachry, C. Fan, Establishing a framework for disaster management system-of-systems, in: *Systems Conference (SysCon), 2018 Annual IEEE International, IEEE*, 2018, pp. 1–7.

- [8] J.-P. Steghöfer, G. Anders, F. Siefert, W. Reif, A system of systems approach to the evolutionary transformation of power management systems., in: GI-Jahrestagung, 2013, pp. 1500–1515.
- [9] A. J. Lopes, R. Lezama, R. Pineda, Model based systems engineering for smart grids as systems of systems, *Procedia Computer Science* 6 (2011) 441–450.
- [10] A. Gorod, S. Merchant, L. Hallo, Toward systemic governance of cancer treatment as a system of systems, in: 2018 13th Annual Conference on System of Systems Engineering (SoSE), IEEE, 2018, pp. 556–560.
- [11] S. Okami, N. Kohtake, Transitional complexity of health information system of systems: Managing by the engineering systems multiple-domain modeling approach, *IEEE Systems Journal*.
- [12] J. Boardman, B. Sauser, System of systems-the meaning of of, in: *System of Systems Engineering*, 2006 IEEE/SMC International Conference on, IEEE, 2006, pp. 6–pp.
- [13] A. Ceccarelli, A. Bondavalli, B. Froemel, O. Hoeffberger, H. Kopetz, Basic Concepts on Systems of Systems, in: *Cyber-Physical Systems of Systems*, Lecture Notes in Computer Science, Springer, Cham, 2016, pp. 1–39. doi : 10.1007/978-3-319-47590-5_1.
- [14] M. Gagliardi, W. Wood, J. Klein, J. Morley, A uniform approach for system of systems architecture evaluation, *CrossTalk* 22 (3-4) (2009) 12–15.
- [15] E. Y. Nakagawa, M. Gonçalves, M. Guessi, L. B. Oliveira, F. Oquendo, The state of the art and future perspectives in systems of systems software architectures, in: *Proceedings of the First International Workshop on Software Engineering for Systems-of-Systems*, ACM, 2013, pp. 13–20.
- [16] J. Klein, H. van Vliet, A Systematic Review of System-of-systems Architecture Research, in: *Proceedings of the 9th International ACM Sigsoft*

- Conference on Quality of Software Architectures, QoSA '13, ACM, New York, NY, USA, 2013, pp. 13–22. doi:10.1145/2465478.2465490.
- [17] P. Gomes, E. Cavalcante, P. Maia, T. Batista, K. Oliveira, A systematic mapping on discovery and composition mechanisms for systems-of-systems, in: 2015 41st Euromicro Conference on Software Engineering and Advanced Applications, IEEE, 2015, pp. 191–198. doi:10.1109/SEAA.2015.51.
 - [18] V. V. Graciano Neto, M. Guessi, L. B. R. Oliveira, F. Oquendo, E. Y. Nakagawa, Investigating the model-driven development for systems-of-systems, in: Proceedings of the 2014 European Conference on software architecture workshops, ACM, 2014, p. 22.
 - [19] M. Guessi, V. V. G. Neto, T. Bianchi, K. R. Felizardo, F. Oquendo, E. Y. Nakagawa, A systematic literature review on the description of software architectures for systems of systems, in: Proceedings of the 30th Annual ACM Symposium on Applied Computing, ACM Press, 2015, pp. 1433–1440. doi:10.1145/2695664.2695795.
 - [20] V. Barot, M. Henshaw, C. Siemieniuch, H. Dogan, Design of a web-based thesaurus for systems of systems engineering, in: System of Systems Engineering (SoSE), 2013 8th International Conference on, IEEE, 2013, pp. 7–12.
 - [21] M. J. de C Henshaw, Systems of systems, cyber-physical systems, the internet-of-things... whatever next?, *Insight* 19 (3) (2016) 51–54.
 - [22] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, S. Linkman, Systematic literature reviews in software engineering—a systematic literature review, *Information and software technology* 51 (1) (2009) 7–15.
 - [23] B. Kitchenham, Procedures for performing systematic reviews, Keele, UK, Keele University 33 (2004) (2004) 1–26.

- [24] M. U. Khan, S. Sherin, M. Z. Iqbal, R. Zahid, Landscaping systematic mapping studies in software engineering: A tertiary study, *Journal of Systems and Software* 149 (2019) 396–436.
- [25] D. Budgen, P. Brereton, S. Drummond, N. Williams, Reporting systematic reviews: Some lessons from a tertiary study, *Information and Software Technology* 95 (2018) 62–74. doi:10.1016/j.infsof.2017.10.017.
- [26] M. Raatikainen, J. Tiihonen, T. Männistö, Software product lines and variability modeling: A tertiary study, *Journal of Systems and Software* 149 (2019) 485–510. doi:10.1016/j.jss.2018.12.027.
- [27] K. Curcio, R. Santana, S. Reinehr, A. Malucelli, Usability in agile software development: A tertiary study, *Computer Standards and Interfaces* doi:10.1016/j.csi.2018.12.003.
- [28] L. Villalobos-Arias, C. Quesada-López, A. Martinez, M. Jenkins, A tertiary study on model-based testing areas, tools and challenges: Preliminary results, 2018, pp. 15–28.
- [29] N. Rios, M. Mendonça Neto, R. Spínola, A tertiary study on technical debt: Types, management strategies, research trends, and base information for practitioners, *Information and Software Technology* 102 (2018) 117–145. doi:10.1016/j.infsof.2018.05.010.
- [30] B. Kitchenham, S. Charters, Guidelines for performing structural literature reviews in software engineering, *Empirical Software Engineering, National ICT* (2007) 45–56.
- [31] B. Kitchenham, R. Pretorius, D. Budgen, O. Pearl Brereton, M. Turner, M. Niazi, S. Linkman, Systematic literature reviews in software engineering – A tertiary study, *Information and Software Technology* 52 (8) (2010) 792–805. doi:10.1016/j.infsof.2010.03.006.
- [32] W. Bandara, E. Furtmueller, E. Gorbacheva, S. Miskon, J. Beekhuizen, Achieving rigor in literature reviews: Insights from qualitative data anal-

- ysis and tool-support, *Communications of the Association for Information Systems* 37 (2015) 154–204.
- [33] H. Zhang, M. A. Babar, P. Tell, Identifying relevant studies in software engineering, *Information and Software Technology* 53 (6) (2011) 625–637. doi:10.1016/j.infsof.2010.12.010.
 - [34] C. Hofmeister, P. Kruchten, R. L. Nord, H. Obbink, A. Ran, P. America, A general model of software architecture design derived from five industrial approaches, *Journal of Systems and Software* 80 (1) (2007) 106–126. doi:10.1016/j.jss.2006.05.024.
 - [35] A. Tang, P. Avgeriou, A. Jansen, R. Capilla, M. A. Babar, A comparative study of architecture knowledge management tools, *Journal of Systems and Software* 83 (3) (2010) 352–370.
 - [36] N. B. Harrison, P. Avgeriou, Leveraging architecture patterns to satisfy quality attributes, in: *European conference on software architecture*, Springer, 2007, pp. 263–270.
 - [37] I. Iso, Iec25010: 2011 systems and software engineering—systems and software quality requirements and evaluation (square)—system and software quality models, *International Organization for Standardization* 34 (2011) 2910.
 - [38] C. Wohlin, 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*, ACM, 2014, p. 38.
 - [39] D. S. Cruzes, T. Dybå, Research synthesis in software engineering: A tertiary study, *Information and Software Technology* 53 (5) (2011) 440–455. doi:10.1016/j.infsof.2011.01.004.

- [40] W. Bandara, S. Miskon, E. Felt, A systematic, tool-supported method for conducting literature reviews in information systems, in: Proceedings of the 19th European Conference on Information Systems (ECIS 2011), 2011.
- [41] S. Cavanagh, Content analysis: concepts, methods and applications., *Nurse researcher* 4 (3) (1997) 5–16.
- [42] M. B. Gonçalves, E. Cavalcante, T. Batista, F. Oquendo, E. Y. Nakagawa, Towards a conceptual model for software-intensive system-of-systems, in: 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2014, pp. 1605–1610.
- [43] M. Dixon-Woods, S. Agarwal, D. Jones, B. Young, A. Sutton, Synthesising qualitative and quantitative evidence: a review of possible methods, *Journal of health services research & policy* 10 (1) (2005) 45–53.
- [44] J. D. Novak, Concept mapping: A strategy for organizing knowledge, *Learning science in the schools: Research reforming practice* (1995) 229–245.
- [45] M. B. Gonçalves, F. Oquendo, E. Y. Nakagawa, A meta-process to construct software architectures for system of systems, in: Proceedings of the 30th Annual ACM Symposium on Applied Computing, ACM, 2015, pp. 1411–1416.
- [46] C. Yang, P. Liang, P. Avgeriou, A systematic mapping study on the combination of software architecture and agile development, *Journal of Systems and Software* 111 (2016) 157–184.
- [47] A. Ampatzoglou, S. Bibi, P. Avgeriou, M. Verbeek, A. Chatzigeorgiou, Identifying, categorizing and mitigating threats to validity in software engineering secondary studies, *Information and Software Technology*.
- [48] B. A. Kitchenham, D. Budgen, P. Brereton, Evidence-based software engineering and systematic reviews, Vol. 4, CRC Press, 2015.

Appendix A Secondary Studies

- [S1] G. Abdalla, C. D. N. Damasceno, M. Guessi, F. Oquendo, E. Y. Nakagawa, A systematic literature review on knowledge representation approaches for systems-of-systems, in: 2015 IX Brazilian Symposium on Components, Architectures and Reuse Software, IEEE, 2015, pp. 70-79. doi: 10.1109/SBCARS.2015.18
- [S2] J. Axelsson, A systematic mapping of the research literature on system-of-systems engineering, in: 2015 10th System of Systems Engineering Conference (SoSE), IEEE, 2015, pp. 18-23. doi: 10.1109/SYSESE.2015.7151918
- [S3] T. Bianchi, D. S. Santos, K. R. Felizardo, Quality Attributes of Systemsof-Systems: A Systematic Literature Review, in: 2015 IEEE/ACM 3rd International Workshop on Software Engineering for Systems-of-Systems, IEEE, 2015, pp. 23-30. doi: 10.1109/SESOS.2015.12
- [S4] M. Daneva, B. Lazarov, Requirements for smart cities: Results from a systematic review of literature, in: 2018 12th International Conference on Research Challenges in Information Science (RCIS), IEEE, 2018, pp. 1-6. doi: 10.1109/RCIS.2018.8406655
- [S5] R. De Lima, D. De Vargas, L. Fontoura, System of systems requirements: A systematic literature review using snowballing, in: Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, 2017, pp. 97-100. doi: 10.18293/SEKE2017-114
- [S6] D. Feitosa, A. Ampatzoglou, P. Avgeriou, F. J. Affonso, H. Andrade, K. R. Felizardo, E. Y. Nakagawa, Design Approaches for Critical Embedded Systems: A Systematic Mapping Study, in: E. Damiani, G. Spanoudakis, L. Maciaszek (Eds.), Evaluation of Novel Approaches to Software Engineering, Vol. 866, Springer International Publishing, Cham, 2018, pp. 243-274. doi: 10.1007/978-3-319-94135-6_12

- [S7] P. Gomes, E. Cavalcante, P. Maia, T. Batista, K. Oliveira, A systematic mapping on discovery and composition mechanisms for systems-of-systems, in: 2015 41st Euromicro Conference on Software Engineering and Advanced Applications, IEEE, 2015, pp. 191-198. doi: 10.1109/SEAA.2015.51
- [S8] V. V. Graciano Neto, M. Guessi, L. B. R. Oliveira, F. Oquendo, E. Y. Nakagawa, Investigating the model-driven development for systems-of-systems, in: Proceedings of the 2014 European Conference on software architecture workshops, ACM, 2014, p. 22. doi: 10.1145/2642803.2642825
- [S9] M. Guessi, V. V. G. Neto, T. Bianchi, K. R. Felizardo, F. Oquendo, E. Y. Nakagawa, A systematic literature review on the description of software architectures for systems of systems, in: Proceedings of the 30th Annual ACM Symposium on Applied Computing, ACM Press, 2015, pp. 1433-1440. doi: 10.1145/2695664.2695795
- [S10] J. Klein, H. van Vliet, A Systematic Review of System-of-systems Architecture Research, in: Proceedings of the 9th International ACM Sigsoft Conference on Quality of Software Architectures, QoSA '13, ACM, New York, NY, USA, 2013, pp. 13-22. doi: 10.1145/2465478.2465490
- [S11] C. A. Lana, N. M. Souza, M. E. Delamaro, E. Y. Nakagawa, F. Oquendo, J. C. Maldonado, Systems-of-systems development: Initiatives, trends, and challenges, in: 2016 XLII Latin American Computing Conference (CLEI), IEEE, 2016, pp. 1-12. doi: 10.1109/CLEI.2016.7833329
- [S12] P. Maia, E. Cavalcante, P. Gomes, T. Batista, F. C. Delicato, P. F. Pires, On the development of systems-of-systems based on the internet of things: a systematic mapping, in: Proceedings of the 2014 European Conference on Software Architecture Workshops, ACM, 2014, p. 23. doi: 10.1145/2642803.2642828
- [S13] R. C. Motta, K. M. D. Oliveira, G. H. Travassos, Rethinking Interoperability in Contemporary Software Systems, in: 2017 IEEE/ACM Joint 5th

International Workshop on Software Engineering for Systems-of-Systems and 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems (JSOS), IEEE, 2017, pp. 9-15. doi: 10.1109/JSOS.2017.5

- [S14] E. Y. Nakagawa, M. Gonçalves, M. Guessi, L. B. Oliveira, F. Oquendo, The state of the art and future perspectives in systems of systems software architectures, in: Proceedings of the First International Workshop on Software Engineering for Systems-of-Systems, ACM, 2013, pp. 13-20. doi: 10.1145/2489850.2489853
- [S15] E. Papatheocharous, J. Andersson, J. Axelsson, Ecosystems and Open Innovation for Embedded Systems: A Systematic Mapping Study, in: J. a. M. Fernandes, R. J. Machado, K. Wnuk (Eds.), Software Business, Vol. 210, Springer International Publishing, Cham, 2015, pp. 81-95. doi: 10.1007/978-3-319-19593-3_7
- [S16] E. Silva, E. Cavalcante, T. Batista, F. Oquendo, F. C. Delicato, P. F. Pires, On the characterization of missions of systems-of-systems, in: Proceedings of the 2014 European Conference on Software Architecture Workshops, ACM, 2014, p. 26. doi: 10.1145/2642803.2642829
- [S17] I. G. Vargas, T. Gottardi, R. T. V. Braga, Approaches for integration in system of systems: A systematic review, in: 2016 IEEE/ACM 4th International Workshop on Software Engineering for Systems-of-Systems (SESoS), ACM Press, 2016, pp. 32-38. doi: 10.1145/2897829.2897835
- [S18] M. Vierhauser, R. Rabiser, P. Grünbacher, Requirements monitoring frameworks: A systematic review, Information and Software Technology 80 (2016) 89-109. doi: 10.1016/j.infsof.2016.08.005
- [S19] A. Wortmann, B. Combemale, O. Barais, A Systematic Mapping Study on Modeling for Industry 4.0, in: 2017 ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems (MODELS), IEEE, 2017, pp. 281-291. doi: 10.1109/MODELS.2017.14

Appendix B Axial Coding - raw concept map

Figure 12: Concept map of concept related to SoS based on statements extracted from the selected studies. Base relations coming e.g. from Maier's definition are denoted with (*).