IT University
of Copenhagen

Research Protocol of

Software Architectures for Robotics Systems: A Systematic Mapping Study

Aakash Ahmad
Software and Systems Section
aahm@itu.dk

Muhammad Ali Babar
Software and Systems Section
maba@itu.dk

IT University Technical Report Series TR-[2016-]

ISSN [March 2016]

ISSN
ISBN

Copies may be obtained by contacting:

IT University of Copenhagen
Rued Langgaards Vej 7
DK – 2300 Copenhagen S
Denmark

Telephone: +45 72 18 50 00
Telefax: +45 72 18 50 01

Contents

Research Protocol of

Software Architectures for Robotics Systems: A Systematic Mapping Study

Aakash Ahmad[1], Muhammad Ali Babar[2, 1]

CREST - Centre for Research on Engineering Software Technologies

[1]Software and Systems Section, IT University of Copenhagen, Denmark

[2]School of Computer Science, University of Adelaide, Australia

[1]aahm@itu.dk, [2]ali.babar@adelaide.edu.au

Overview

Context: Several research efforts have been targeted to support architecture centric development and evolution of software for robotic systems for the last two decades.

Objective: We aimed to systematically *identify* and *classify* the existing solutions, research progress and trends that influence architecture-driven modeling, development and evolution of robotic software.

Research Method: We have used Systematic Mapping Study (SMS) method for identifying and analyzing 56 peer-reviewed papers. Our review has (i) taxonomically classified the existing research and (ii) systematically mapped the solutions. To conduct the SMS, we document the methodological details and necessary steps for the study as the research protocol.

Conclusions: In this report, we provide the details of the research protocol that identifies and documents the necessary data and steps for the SMS. The details of the protocol help to understand the methodology, reproducing the results and helps to interpret the study findings more objectively.

Report Organization: We have organized this report as follows. Section 1 defines the research protocol for mapping study. Section 2 presents the identification and qualitative assessment of the primary studies. Section 3 concludes the report with discussion about evaluating the protocol. Section 4 presents some validity threats to the mapping study.

Keywords: Research Protocol, Research Methodology, Systematic Mapping Study

1. Defining the Research Protocol

We used Systematic Mapping Study (SMS) method [2] that involves a three step process (in Figure 1): (i) planning a study, (ii) data collection and synthesis, and (iii) mapping and documenting results. A systematic approach for a review reduces bias in identifying, selecting, synthesizing the data and reporting results. Following sub-sections provide the details of research methodology guided by Figure 1

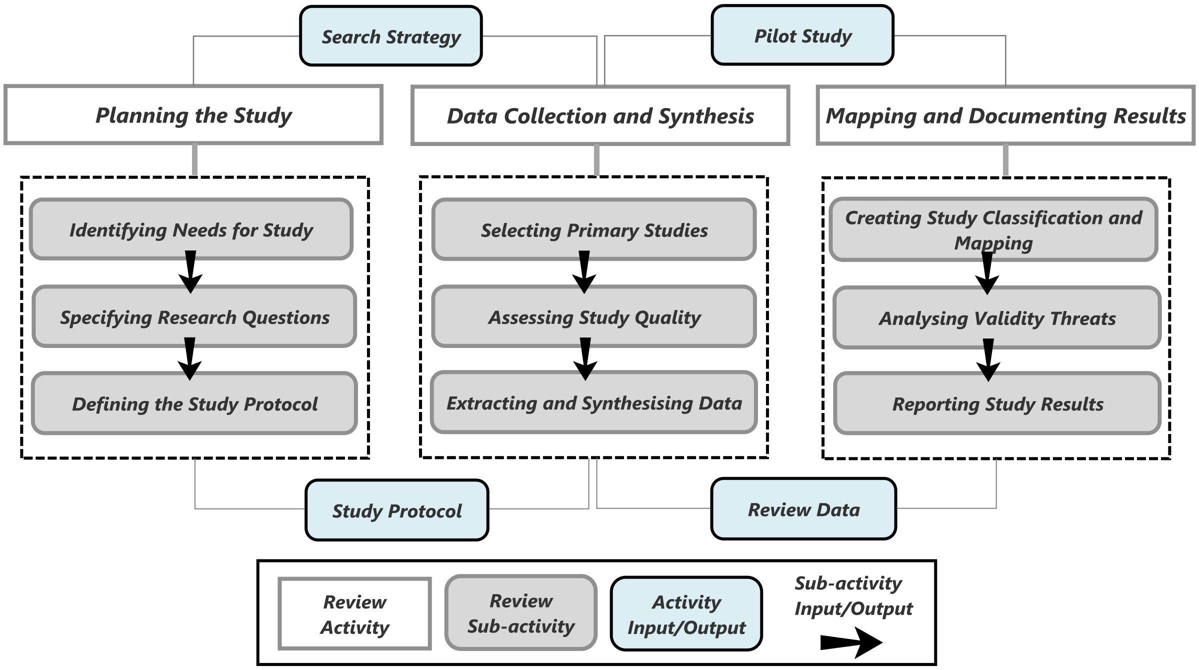


Figure 1. An Overview of the Methodology for Mapping Study

1.1 Defining the Protocol for Systematic Mapping Study

According to the guidelines for conducting the systematic review and mapping studies in [1, 2], the research protocol includes, (i) identification of the needs to conduct the mapping study, (ii) defining study's scope based on research questions (RQs) to be answered, and (iii) formulating the search string and search strategies (based on RQs) to identify, include/exclude and qualitatively analyze the relevant literature. A study's protocol documents methodological details and logistical procedures for a research study [1]. The protocol also allows us to evaluate different activities and their outcomes. We discuss the individual steps involved in the protocol.

1.1.1 Identify the Needs for Mapping Study

Despite a multi-disciplinary and continuously growing research for more than twenty years, there was no effort to systematically select, analyze, and report the peer-reviewed research on the progression, maturation and emerging trends of architectural solutions for robotic software. In contrast to the existing research in [3, 4], that are focused on service-orientation or in general software solutions for robotics; our proposed study aims to focus on generic architectural solutions for robotic software. Specifically, in comparison to [18]; our proposed study is aimed at going beyond SOA and providing a more comprehensive mapping and review of other architectural solutions. Before conducting the SMS, we must ensure that a similar study to our review has not been conducted or published. Therefore, we searched the IEEE Xplore, ACM Digital Library, Springer Link and Science Direct (on 04/03/2016) with the following search string in Listing 1 to identify the relevant secondary studies. Specifically, Listing 1 presents the search string to identify any relevant (survey/review-based) secondary studies on software architecture for robotic systems. Based on the literature identification with the following search string, none of the publications that we retrieved were aimed at answering the outlined research questions below that had motivated our mapping study.

(“Systematic Literature Review” **OR** *“Systematic Mapping”* **OR** *“Study”* **OR** *“Survey”)*
AND
(“Software Architecture” **OR** *“Software Component”* **OR** *“Software Framework”* **OR** *“Software Engineering”)*
AND
(“Robot” **OR** *“Robotic”* **OR** *“Humanoid”)*

Listing 1. Search String to Identify the Relevant Secondary Studies.

1.1.2 Specifying the Research Questions

We formulated three research questions to be answered by our study. Each of the main research questions has sub-questions for fine-grained investigation and presentation of the results.

RQ 1: What is the state-of-research on software architecture-based solutions for robotic systems?

The objectives of this question can be met by answering the following sub-questions.

- ***RQ 1.1: What research themes have been identified and how they can be classified?***
- ***RQ 1.2: What types of architectural solutions have been reported for robotic software?***
- ***RQ 1.3: What architectural frameworks have been provided to support the solutions?***
- ***RQ 1.4: What architectural notations have been exploited for solution representation?***
- ***RQ 1.5: What validation methods have been employed to evaluate the solutions?***
- ***RQ 1.6: What were the application domains for architectural solutions?***

RQ 2 – What are the demographic details of research in terms of publication years, sources and active communities?

To answer different aspects of research demography, we decompose the questions as follows.

- ***RQ 2.1: What is the publication frequency and fora of research over the years?***
- ***RQ 2.2: What are the prominent venues of publication and the types of published research?***
- ***RQ 2.3: What research communities are active on software architectural solutions for robotic systems?***

RQ 3 - What were the past trends and what types of existing and emerging trends could be identified for architecture-based solutions for robotic systems?

To discuss the past, present and possible future research trends, we answer the following questions.

- ***RQ 3.1: What are the past trends of research on architecting robotic software?***
- ***RQ 3.2: What are the emerging trends of research on software architecture for robotics?***

1.1.3 Searching the Relevant Literature – Primary Studies

After defining the protocol, we followed the steps to collect and synthesize the data (from Figure 1). Figure 2 shows the search process used for this study. The above-mentioned research questions helped us to identify a set of keywords that were used to build a search string that was applied to five databases shown in Figure 2. We limited our search to the peer-reviewed literature from years 1991 to 2015 (15/09/2015). The year 1991 was chosen as the initial search found no earlier results related to any of the research questions with 23380 hits. In the primary search process (detailed next), we focused on title and abstract, therefore, it resulted in a high number of studies that were not relevant, which we refined with secondary search process - limiting the extracted studies to 308 in total. In order to identify and select primary studies, the search string was customized as per individual databases for effective search [1]. Please note that the first search string (cf. **Section 1.1.1**) helped us to identify the relevant secondary studies on software architecture for robotic systems.

In contrast, the search string in Figure 2 aims to identify the primary studies that focus on methods and techniques for architecture-driven robotics software. Based on screening and qualitative assessment of the extracted studies out of 97 a total of 56 studies were selected for inclusion in this study.

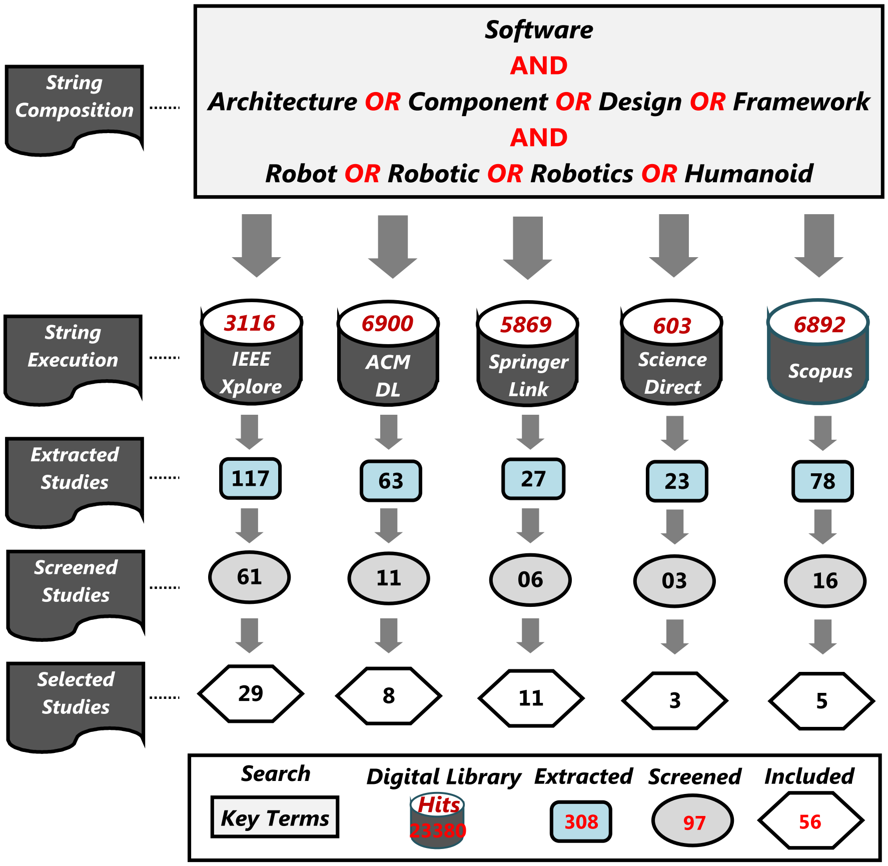


Figure 2. Summary of the Literature Search Process with Search String

2 Identification, Qualitative Assessment and Data Extraction of Primary Studies

2.1 Selection of Primary Studies

Step I – Primary Search is further decomposed into four tasks to identify the relevant literature. We provide a summary of each task involved in the primary search process in Table 1.

Table 1. A Summary of Steps for Primary Search of Relevant Literature.

Search Step	Description
***A*. Deriving Search Terms**	We derived search terms from RQs in Section 1.1.2
***B*. Considering Synonyms and Alternatives for Search Terms**	We considered the alternative keywords for deriving the literature search stings: **- Software Architecture** as [*Software Design*, *Software Component*, *Software Framework*] The other relevant terms for architecture like 'Software Structure' or 'Software Styles' were excluded as they resulted in a large number of irrelevant literature hits. **- Robots** as [Robotic, Robotics, Humanoid]. The other relevant terms for robots like 'Agent' or 'Android' were also excluded to avoid a large number of irrelevant hits.
***C*. Combining Search Terms to Compose Search Strings**	We combined the search terms to compose the search strings: - Boolean ***OR*** operators were used to incorporate alternative spellings and synonyms - Boolean ***AND*** operators is used to link the search terms.
***D*. Dividing and Customizing Search Strings**	We divided and customized the search strings so that they could be applied to different databases (digital libraries) containing the literature. We assigned the unique IDs to every (sub-) search string.

Step II – Customized Search Strings

As per the step D in Table 1, we derived the following customized search strings as per the individual digital libraries. Google Scholar have been used as a primary source for the identification of relevant literature in systematic reviews. However, among others the critical factors such as the (i) frequency of changes (tweaks) to Google search algorithm**[1]**, (ii) personalized and context-driven searching capabilities, and (iii) a significant amount of gray literature determined our decision about not including Google Scholar as one of the primary literature search source. We only used Google Scholar as an auxiliary search engine to ensure if some relevant literature may not have been missed by our selected search engines/digital libraries (cf. Figure 2)

---

[1] Google Algorithm Change History: http://www.seomoz.org/google-algorithm-change

to randomly cross-check the search results. The cross check did not reveal any additional relevant study.

- ***IEEE eXplore*** (www.ieeexplore.ieee.org)

Search String 1 for IEEE eXplore

("Document Title": Software ***OR*** "Abstract": Software) ***AND***("Document Title": Architecture ***OR*** "Document Title": Component ***OR*** "Document Title": Design ***OR*** "Document Title": Framework ***OR*** "Abstract": Architecture ***OR*** "Abstract": Component ***OR*** "Abstract": Design ***OR*** "Abstract": Framework) ***AND*** ("Document Title": Robot ***OR*** "Document Title": Robotic ***OR*** "Document Title": Humanoid ***OR*** "Abstract": Robot ***OR*** "Abstract": Robotic ***OR*** "Abstract": Humanoid)

- ***ACM Digital Library*** (www.dl.acm.org)

Search String 2 for ACM Digital Library

((Owner:ACM) ***AND***(Abstract "Software" ***OR*** Title "Software") ***AND*** (Abstract "Architecture" ***OR*** Abstract "Component" ***OR*** Abstract "Design" ***OR*** Abstract "Framework" ***OR*** Title "Architecture" ***OR*** Title "Component" ***OR*** Title "Design" ***OR*** Title "Framework") ***AND*** (Abstract "Robot" ***OR*** Abstract "Robotic" ***OR*** Abstract "Humanoid" ***OR*** Title "Robot" ***OR*** Title "Robotic" ***OR*** Title "Humanoid"))

- ***Springer Link*** (www.link.springer.com)

Search String 3 for Springer Link

(Software) ***AND*** (Architecture ***OR*** Design ***OR*** Component ***OR*** Framework) ***AND*** (Robot ***OR*** Robotic ***OR*** Humanoid)

- ***Science Direct*** (www.sciencedirect.com)

Search String 4 for Science Direct

TITLE-ABSTR-KEY((Software) ***AND*** (Architecture ***OR*** Design ***OR*** Component ***OR*** Framework) ***AND*** (Robot ***OR*** Robotic ***OR*** Humanoid))

- ***Scopus*** (www.scopus.com)

We have refined the following search string with the exclusion of the term 'KEY' (keyword-based selection), that helped us eliminating a significant number of irrelevant studies. In doing so, we may have overlooked some relevant literature (cf. Section 9, *Validity Threats to the Identification of Primary Studies*). However, we believe the refinement of the search strings has helped us to minimize the irrelevant literature.

Search String 5 for Scopus

TITLE-ABS((Software) ***AND*** (Architecture ***OR*** Design ***OR*** Component ***OR*** Framework) ***AND*** (Robot ***OR*** Robotic ***OR*** Humanoid))

Step III – Secondary Search a two phase process consisting of the *primary* and *secondary* search is based on the guidelines and empirical comparison of literature search using *digital database*/libraries and the *snowballing process* [5]. As detailed earlier, the primary search (cf. Section 1.1.3, Figure 2) identified the relevant studies for SMS by executing the search strings on digital databases/libraries. After screening and qualitative assessment, we selected 56 studies for review and analyzed the references/bibliography section for each of the 56 selected studies to find other relevant studies for their possible inclusion – the snowballing process. We limited the snowballing process to the selected primary (56) studies to avoid any exhaustive search. A detailed comparison and the relative benefits and limitations of literature search using digital databases vs snowballing is detailed in [5]. As a results of snowballing process, we did not identify any study to be included in the review. In particular, during the snowballing process; we found studies that were either (i) already included in the list of primary studies, (ii) not explicitly relevant to RQs or (iii) eliminated during the qualitative assessment process.

2.2 Screening and Qualitative Assessment of Studies

The study selection comprises of a two-step process that includes screening and qualitative assessment as presented in Tables 2 and Table 3. In Table 3, the qualitative assessment helps us to include/exclude studies and rank the selected studies based on their quality score.

Step I – Screening of Studies

The study screening is a two-step process consisting of the *Generic Screening* (having five sub-steps, **I-A** to **I-E**) and *Specific Screening* as outlined in Table 2. During the Generic Screening step (review of study titles), first we need to screen the 308 studies to ensure that (i) *duplicate studies* (Step I-A), (ii) non *English language* literature (Step I-B), (iii) *non peer-reviewed* and *non-published research* (Step-I-C) and, (iv) any study representing an *entire book* (Step I-D) are removed first. Furthermore, any *secondary study* (such as a systematic review or survey-based literature) is eliminated from primary studies to be reviewed – any secondary study represents related research to our SMS. The Generic Screening step helped us to remove a significant number of studies (211) with total remaining studies 97.

Table 2. Summary of the Study Selection Process (without qualitative assessment)

	Step I – Generic Screening		
I-A	*Is the study a duplicate?*	**YES**	**NO**
I-B	*Is the study in English language?*	**YES**	**NO**
I-C	*Is the study a scientific peer-reviewed published research (no white papers or technical reports)?*	**YES**	**NO**
I-D	*Is the study not a secondary study?*	**YES**	**NO**
I-E	*Is the study not a book?*	**YES**	**NO**
	If [YES] to all four criteria then go to Step II, otherwise exclude study		
	Step II – Specific Screening		
II-A	RQ1, RQ2 *Does the study presents an architectural method, technique or a solution for robotic software systems?* If [**YES**] go to Table B.2 (qualitative assessment), otherwise exclude study		

During the Step II - Specific Screening, we performed a preliminary review to analyze the relevance of the studies to RQs (method, technique or a solution for robotic software systems). During Step II, the decision to exclude ([NO]) or proceeding to the final selection ([YES]) was based on an examination of study *titles* and a preliminary review of the *abstracts*, *conclusions* and any other relevant part of the remaining studies. Based on the screening the number of studies was reduced to 97.

Step II – Qualitative Assessment of Studies

During qualitative assessment of 97 included studies, we focused on assessing the technical rigor of contents presented in the study. The qualitative assessment is based on two factors as general assessment (G) and specific assessment (S), in Table 3. We adopted the guidelines of qualitative synthesis of research evidence [6], tailored them as per the needs for our study as presented in Table 3.

Table 3. Summary of Quality Assessment Checklist

General Items for Quality Assessment (G)				
Score for General Items		Yes = 1	Partial = 0.5	No = 0
G1	Are *problem definition* and *motivation* of the study clearly presented?			
G2	Is the *research environment* in which the study was carried out properly explained?			
G3	Are *research methodology* and its *organization* clearly stated?			
G4	Are the *contributions* of the in-line with presented *results*?			
G5	Are the *insights* and *lessons learnt* from the study explicitly mentioned?			
Specific Items for Quality Assessment (S)				
Score for Specific Items		Yes = 1	Partial = 0.5	No = 0
S1	Is the research clearly *focused* on *software architecture solutions for robotic systems*?			
S2	Are the details about *related research* clearly addressing *architectural solutions*?			
S3	Is the *research validation* clearly illustrates the evaluation of architectural solutions?			
S4	Are the results *clearly validated* in a real (industrial case study) evaluation context?			
S5	Are limitations and future research clearly positioned?			

Quality assessment represents 5 factors criteria, providing a maximum score of 1. In the assessment formula below, S and G each represent a total of five factors as ***Specific and Generic Items*** with S having a maximum score of 3 and G with a maximum score 1. S contributes three times more than G (75% weight) as specific contributions of a study are more important than general factors for assessment. Based on the consensus among the researchers the maximum score was decided as G + S = 4, where a 3 – 4 score represented quality papers, a score less than 3 and greater than or equal to 1.5 was acceptable and a score less than 1.5 resulted in study exclusion.

$$\boldsymbol{Quality\ Score} = \left[\frac{\sum_{G=1}^{5}}{5} + \left(\frac{\sum_{S=1}^{5}}{5} \times 3 \right) \right]$$

Based on qualitative assessment of *97* studies, we excluded a total of *41* (quality score less than 1.5) studies to finally select *56* primary studies for the review.

2.3 Data Extraction for Synthesis

To collect and record the format in Table 4 collects two types of data (i) *generic and study demographic data items* (D01 – D06) and ii) *classification and mapping specific data items* (M01 – M10). The latter category helps us to answer the RQs (cf. Section 1.1.2).

Table 4. Template for Extraction of Data from Primary Studies

ID	*Data Item*	*Aim*
Generic and Study Demographic Data Items		
D01	*Study ID*	Unique id of study ________
D02	*Study Title*	The title of Study ________________________
D03	*Bibliography*	a) List of Author(s) ________________________ b) Year of Publication ______________________ c) Source of Publication *Journal* *Conference* *Symposium* *or Workshop* *Other* ________________________
D04	*Citation Count*	Total number of citations _______________
D05	*Quality Score*	Quality score of study _______________
D06	*Additional Information*	Any additional or study specific information ________________________
Classification and Mapping Specific Data Items		
M01	*Research Problem*	Overview of research problems addressed ______________________
M02	*Architectural Solution*	Overview of solution to address the problem ______________________
M03	*Research Context*	Context and application domain: *Academic* *Industrial* *Both* *Other*
M04	*Framework Support*	Architectural Framework for Development. *Yes* *No* *Description* ________
M05	*Modeling Notations*	*UML* *ADL* *Graph Models* *Ontologies* *Other*
M06	*Architecture Model Type*	*Component* *Service* *Object* *Other*
M07	*Validation Method*	*Design and Evaluation* *Case Study* *Survey* *Experiments* *Other*
M08	*Architecture Evaluations*	*Yes* *No* *Description*
M09	*Research Trends*	The identified research trends ______________________

M10	*Future Dimensions*	The identified future research

The protocol was internally reviewed a few times for improvement. We also have our study's protocol externally evaluated for refinements and reducing the bias. We performed a pilot study by reviewing 15 (more than 25%) of the included studies. The pilot study purported to reduce the study identification bias and to refine the process for (i) identification of primary studies, (ii) extraction of data from these studies and (iii) synthesizing the results. Based on the protocol review and pilot study, we expanded the review scope in the context of relevant studies [1, 2], improved search strategies and refined the study inclusion/exclusion and qualitative assessment criteria [6]. We also evaluated the study protocol with details provided later in this section.

3. Evaluating the Protocol for SMS

Once the protocol is defined, the guidelines for conducting the SLR and SMS [1, 2, 5, 6] suggest the needs to internally and externally evaluate the study protocol before its execution. We performed both the internal and external evaluation of the protocol to eliminate or minimize the possible bias. We focused on specifically evaluating steps that included (i) *identification and qualitative assessment of the primary studies*, (ii) *consistency of data extraction and reporting*, and finally (iii) *data synthesis and results reporting*.

3.1 Internal Evaluation

As a team of two researchers, we focused on a structured representation of the information (e.g.; Qualitative assessment of studies – Table 3, Data extraction template – Table 4) for an objective interpretation and evaluation of the methodology steps. First of all, both the researchers executed the search string on the selected digital libraries (cf. Figure 2) individually and then shared the search results. We also conducted a pilot search first to refine the search strings. For example, the terms *software architecture* and *software design* are complementary and virtually synonymous. We also included the term software design in the search string that led to a significant increase in the number of identified studies and the efforts to retrieve the most relevant. We excluded the terms like 'Software Structure' or 'Software Styles' in order to minimize the irrelevant literature. Once the consistency of search results was ensured, the first researcher identified the relevant studies, maintaining their references and derived a multi-criteria assessment of the study quality (cf. Table 3). The second researcher cross-checked the results of literature identification (randomly selecting and checking results with 2/5 of the digital libraries) and assessing studies against the qualitative assessment checklist.

3.2 External Evaluation

As a two-step process was performed by a researcher external to our team, who had expertise in conducting SLRs and whose research interests lied in the area of software architecture. In the first step, we shared the RQs, identified studies and the data extraction form. Based on his feedback and recommendations, we refined the research questions and made necessary adjustments to the data extraction template (cf. Table 4). For example, a suggestion was to distinguish between the presentation of solution validation and architecture evaluations (attributes M07, M08 in Table 4 –

Data Extraction Form). In the second phase, due to time constraints instead of sharing the detailed results, we only shared the data extraction form and research questions. Based on the external feedback, we refined and finalized the data extraction form before capturing data and synthesizing results. Some possible threats to the validity of research are detailed later after discussion of the results.

4. Validity Threats

This study provides a classification to map the reported solutions by reviewing and analyzing peer-reviewed literature. We followed the guidelines for conducting systematic mapping studies reported in [1, 2, 5] based on a defined and - internally and externally - evaluated protocol for SMS (cf. Section 1). Like any other empirical study, systematic mapping studies can also have limitations that must be considered for analyzing the potential impact of the validity threats to the findings of SMS [6]. We discuss three types of validity threats associated with different activities of this SMS.

4.1 Threats to Identification of Primary Studies.

In the literature search strategy (cf. Section 1, Section 2), we aimed to retrieve as many relevant studies as possible to avoid any possible literature selection bias and to accommodate all the available evidence. We faced a challenge in determining the scope of our study as the notion of architecture means different things to different research communities including software engineering, robotics, artificial intelligence and others. Therefore, to cover them all and avoid any bias, we searched the literature based on relevant terms (cf. Table 1) and combined them in our search string (cf. Figure 2). While this search strategy and search string composition significantly increases the search work, however, it enabled us to find a comprehensive set of the relevant study. We also developed and evaluated a review protocol (cf. Section 1 - Section 3). The protocol provides a replicable blue-print to derive the search strategies, literature identification and selection.

4.2 Threats to Quality of Studies and Data Extraction Consistency

The results of this study and their quality are based on the quality of the studies that have been reviewed. This means if the quality of the primary studies is low, the claims and their supporting evidence derived from these studies are unlikely to be strong and reliable. Therefore, it is vital to (i) minimize the threats regarding the quality of selected studies and to ensure (ii) a consistent representation of data extracted from these studies. It is of central importance to qualitatively analyze and synthesize of the data extracted from the selected studies. As described in section 1, we followed a multi-step process and explicitly assessed the quality of each individual study to ensure that a lack of quality results in an exclusion of the study. The ideal scenarios may strictly adhere to the guidelines in [1, 2], however, the quality metric (Table 2, Table 3) can be subjective based on the objectives of SMS and the consensus among researchers.

Moreover, we derived a structured template (cf. Table 4) to ensure consistency in data extraction and capturing as per the needs of the study's RQs. For a fine-grained representation of the extracted data, we have defined a generic and mapping specific attributes to capture data for detailed synthesis.

4.3 Threats to Data synthesis and Results Reporting

The final type of threat corresponds to the bias or a lack of systematic approach to synthesize and report the results. We tried to mitigate this threat by conducting a pilot study. A limited number of researchers and their expertise (software engineering and software architecture) may have an internal bias on the style and reporting of results. The threat to the reliability of data synthesis and reporting has been mitigated based on discussion and peer review of the extracted data by the researchers, having a structured template for data synthesis, and several steps where the scheme and process were refined and evaluated. Whilst we followed the guidelines from [1] to conduct the study, we had deviations from the ideal approaches based on the requirements of this study detailed in Section 1. We believe that the validity of the study is high, given the use of a systematic and recommended procedure, the extensive discussions and evaluation of the protocol and a pilot study to refine the scope of review.

5. Demography of Published Research

We now discuss the demography of the published research in terms of (i) main sources of publication (in Section 5.1), and (ii) various research communities actively pursuing research on software architecture for robotics (in Section 5.1).

5.1 Main Sources by Publication Frequency

We report the main sources (publication venue/sources) of the reviewed studies based on publication frequency as presented in Table 5. The information in Table 5 is expected to help us to identify the representative research communities and an evidence of robotics or specifically software architecture for robotics as a multi-disciplinary research. We present the prominent publications venues in terms of journals, conference series or symposium/workshop along with the type of research that has been published in those venues. We only highlight the publication sources that have at-least two or more studies published as in Table 5. We do not provide extensive details of the published studies. Table 5 indicates the studies (*Study ID*) that provide a reference to consult previous sections. Table 5 highlights a mix of robotics and software engineering research venues and their research focus.

Among others, the main sources are *RAS, T-ASE*, *IROS* as some of the premier journals and conference series respectively for the robotic research community, whose studies are focused on supporting design and reconfiguration activities with a temporal distribution between 1998 to 2015. The top venues in software engineering community are *SEAMS* (a community of self-adaptation software research) and *ICSE* that represents the premier software engineering conference. The publications at these two venues are focused on runtime evolution (architectural reconfiguration) and design-time evolution (architectural reengineering) of robotic systems respectively with publication coverage from 2005 to 2009 with no publication identified at these two venues in the last five years. It is vital to mention about *STAR* with a specific book titled *Software Engineering for Experimental Robotics* [7] published in 2007 mainly addressing software engineering issues for robotic systems. The other two are *SIMPAR* and *ICAR* as robotics research venues with published studies focused on supporting robotics development and coordination related activities published between 2008 to 2012.

Table 5. Overview of Main Sources by Publication Frequency

Study ID	**Publication Source**	**Type**	**Acronym**	**Community**
[S47, S48, S49, S56]	Robotics and Autonomous Systems	Journal	RAS	Robotics/System and Control Engineering
[S22, S25, S32]	IEEE/RSJ International Conference on Intelligent Robots and Systems	Conference	IROS	Robotics
[S5, S36, S41]	International Workshop on Software Engineering for Adaptive and Self-Managing systems	Workshop	*SEAMS	Software Engineering
[S8, S28, S31]	IEEE Transactions on Automation Science and Engineering	Journal	**T-ASE	Robotics/System and Control Engineering
[S14, S37]	International Conference on Software Engineering	Conference	ICSE	Software Engineering
[S50, S54]	Journal of Software Engineering for Robotics	Journal	JOSER	Software Engineering/ Robotics
[S45, S46]	Springer Tracts in Advanced Robotics	Book Chapter	STAR	Robotics
[S33, S35]	International Conference on Simulation, Modeling, and Programming for Autonomous Robots	Conference	SIMPAR	Robotics
[S51, S55]	Workshop on Model-Driven Robot Software Engineering	Workshop	***MORSE	Software Engineering/ Robotics
[S34, S38]	International Conference on Advanced Robotics	Conference	ICAR	Robotics

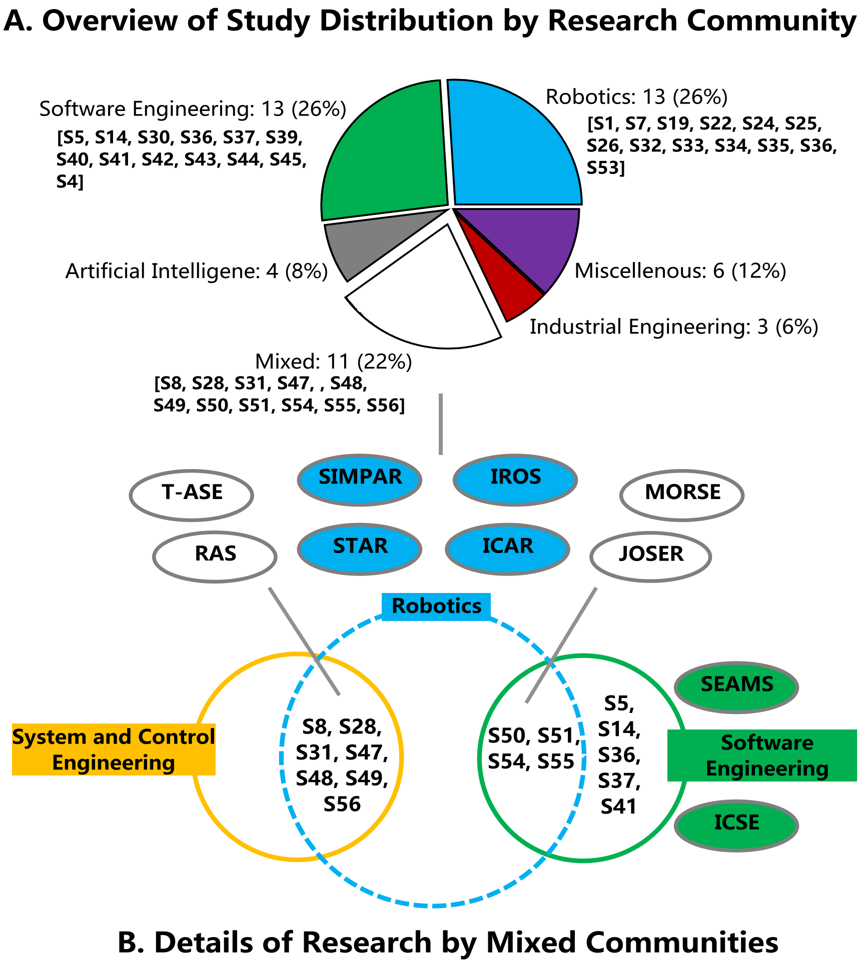
*****Prior to 2011 **SEAMS** was considered a workshop titled *Workshop on Software Engineering for Adaptive and Self-Managing Systems.* Now the title remains same but SEAMS is a symposium. The studies [S5, S36, S41] were all published before 2011 so we classify them as workshop papers.
******Prior to 2004 *IEEE Transactions on Automation Science and Engineering (***T-ASE***)* was titled as *IEEE Transactions on Robotics and Automation* with two included studies [S8, S28]. Now these two studies are organized under the new title.
*******The first edition of this workshop was organized in 2014 promoting the application of software engineering and specifically the application of model-driven techniques to robotic systems

It is vital to mention that in recent years there is a focus on synergizing the research and practices between software engineering and robotics. A specific example is *JOSER* (first issue published in 2010) that aims at providing a platform where the existing software engineering approaches and methods can be leveraged for the development of robotic software systems. Moreover, *MORSE* (first edition in 2014) reflects an effort of software engineering community to support publishable and applicable results of model-driven engineering and its application to robotic systems. The sources and frequency of publication in Table 5 suggests that both the robotics and software engineering research communities are working to synergize their efforts by exploiting the models, languages, tools, infrastructures, patterns, principles and ecosystems etc. for the development, evolution and operations of robotic software. A recent special issue of Autonomous Robots reported the trends on the use of open source software for design and development of robotic systems [8].

5.2 Publications Distribution by Active Research Community

Now we discuss the active communities of research on architecture-driven software for robotics. Figure 3 A) presents the relative distribution of the reviewed studies published by various communities. Moreover, Figure 3 B) specifically highlights a mix of the research communities with 11 studies (e.g., JOSER – a journal focused on the unification of software engineering and robotics research) addressing architectural solutions for robotic systems. From Figure 3 A), it is clear that software architecture-driven robotics is multi-disciplinary research that ranges from *artificial intelligence* to *software*, *industrial engineering*.

Figure 3 B) provides expanded details of the mixed community as overlapping research as Venn Diagram from Figure 3 A). Specifically, Figure 3 B) an overlap of various research communities. For example, the studies [S50, S54] published in JOSER and [S51, S55] published in MORSE represent one of the most recent Journal and Workshop respectively that aims to unify the software engineering and robotics research. Moreover, the studies [S8, S28, S31, S47, S48, S49, S56] published in T-ASE and RAS highlight unified research from robotics and systems and control engineering.


A. Overview of Study Distribution by Research Community

B. Details of Research by Mixed Communities

Figure 3. Active Communities of Published Research.

Figure 3 A) complements the findings presented in Table 5. The findings reveal that software engineering and robotics communities represent a combined 52% (26% and 26% respectively) of the reviewed studies. The distribution of the studies in terms of active research communities is based on the guidelines of ACM classification scheme and also on the analysis of the research focus of the publication venues where both communities have been publishing. For example, in Table 5 the study [S14] in Figure 3 is classified under computing and software engineering (publication source: *ICSE – International Conference on Software Engineering*).

As a possible interpretation of Figure 3, the studies [S9, S12, S13] represent research on architecture for robotics in the context of *artificially intelligent systems to support* the coordination and development of robotics. One study [S9] exploits service-oriented robotics to develop a team of intelligent robots that collect and share mission specific information to support intelligent and autonomous first responder robots. The miscellaneous category refers to the studies published in other communities that are disjoint to the ones in Figure 3. For example, one study [S2] has been published in *IEEE Aerospace Conference* and proposes an architecture named CLARAty developed by NASA to support space control missions. Figure 3 B) provides extended details of the mixed community as overlapping research as Venn Diagram from Figure 3 A). Specifically, Figure 3 B) an overlap of various research communities. For example, the studies [S50, S54] published in JOSER and [S51, S55] published in MORSE represent one of the most recent Journal and Workshop respectively that aims to unify the software engineering and robotics research. Moreover, the studies [S8, S28, S31, S47] highlight unified research from robotics and systems and control engineering.

The research demographics details in this section go beyond the mapping and classification of the existing research to present the publication frequency, sources and active communities. The demographic results suggest a multi-disciplinary research (cf. Figure 3) and possible collaborations among different research communities can further benefit the state-of-the-research. The results support the recent efforts that have been promoting cross-fertilization of research and practices from different communities to engineer robotic software [57, 58].

The findings reveal that software engineering and robotics communities represent a combined 52% (26% and 26% respectively) of the reviewed studies. The distribution of the studies in terms of active research communities is based on the guidelines of ACM classification scheme and also on the analysis of the research focus of the publication venues where both communities have been publishing. For example, in Table 5 the study [S14] in Figure 3 is classified under computing and software engineering (publication source: *ICSE – International Conference on Software Engineering*). We have classified [S50, S54] under software engineering as *JOSER – Journal of Software Engineering for Robotics* (cf. Table 5) is a source of research that aims to support the application of software engineering methodologies to robotic systems.

References

[1] P. Brereton, B. Kitchenham, D. Budgen, M. Turner, M. Khalil. *Lessons from Applying the Systematic Literature Review Process within the Software Engineering Domain*. In Journal of Systems and Software, Elsevier, vol 80, no 4, pp: 571– 583, 2007.

[2] K. Petersen, R. Feldt, S. Mujtaba, M. Mattsson. *Systematic Mapping Studies in Software Engineering*. In In 12th International Conference on Evaluation and Assessment in Software Engineering (EASE 2008), ACM, pp: 68 – 77, 2008.

[3] L. B. R. Oliveira, F. Osorio, E.Y. Nakagawa. An Investigation into the Development of Service-oriented Robotic Systems. In 28th Annual ACM Symposium on Applied Computing (SAC 2013), ACM, pp: 223 - 228 2013.

[4] C. Pons, R. Giandini, G. Arévalo. A Systematic Review of Applying Modern Software Engineering Techniques to Developing Robotic Systems. Revista Ingeniería e Investigación vol. 32, no. 1, pp: 58 - 63, 2012.

[5] C. Wohlin, Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. In 18th international Conference on Evaluation and Assessment in Software Engineering (EASE 2014), 2014.

[6] D.S. Cruzes, T. Dybå. Synthesizing Evidence in Software Engineering Research. In Proceedings of the 4th International Symposium on Empirical Software Engineering and Measurement (ESEM 2010), ACM, 2010.

[7] D. Brugali, A. Brooks, A. Cowley, C. Côté, A. Domínguez-Brito, D. Létourneau, F. Michaud, and C. Schlegel. Trends in Component-Based Robotics. In Software Engineering for Experimental Robotics. Springer Tracts in Advanced Robotics, Springer, vol 30, pp: 135 - 142. Springer Berlin Heidelberg, 2007.

[8] C. Pantofaru, S. Chitta, B. Gerkey, R. Rusu, W. D. Smart, R. Vaughan. Special Issue on Open Source Software-Supported Robotics Research. In Autonomous Robots, Springer, vol 34, no 3, pp: 129 – 131, 2013.

[57] IEEE Robotics and Automation Society: Technical Committee on Software Engineering for Robotics and Automation (TC-SOFT) [Online:] http://robotics.unibg.it/tcsoft/, [Access:] 08/03/2015.

[58] D. Brugali. From the Editor-in-Chief: A New Research Community, a New Journal. In Journal of Software Engineering for Robotics, ISSN: 2035-3928, vol 1, no 1, pp, 01 - 02, 2010.

List of Selected Studies for Systematic Mapping

Study ID	Author(s), Title, Channel of publication	Publication Year	Citation Count	Quality Score
[S1]	T. Kaupp, A. Brooks, B. Upcroft and A. Makarenko. ***Building a Software Architecture for a Human-Robot Team Using the Orca Framework***. In IEEE International Conference on Robotics and Automation	2007	07	**3.5**
[S2]	R. Volpe, I.A.D. Nesnas, T. Estlin, D. Mutz, R. Petras, H. Das. ***The CLARAty Architecture for Robotic Autonomy***. In IEEE Aerospace Conference	2001	232	**2.5**
[S3]	T. Baier, M. Hiiser, D. Westhoff, J. Zhang. ***A Flexible Software Architecture for Multi-modal Service Robots***. In Multiconference on Computational Engineering in Systems Applications	2006	09	**2.5**
[S4]	H. Ahn, D-S. Lee, S. C. Ahn. ***A Hierarchical Fault Tolerant Architecture For Component-based Service Robots***. In IEEE International Conference on Industrial Informatics.	2010	06	**2.5**
[S5]	G. Edwards, J. Garcia, H. Tajalli, D. Popescu, N. Medvidovic, G. Sukhatme. ***Architecture-driven Self-adaptation and Self-management in Robotics Systems***. In ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems.	2009	32	**3.0**
[S6]	X. Ma, K. Qian, X. Dai, F. Fang, Y. Xing. ***Framework Design for Distributed Service Robotic Systems***. 5th IEEE Conference on Industrial Electronics and Applications.	2010	0	**2.5**
[S7]	T. Maenad, A. Tikanmäki , J. Riekki , J. Röning***. A Distributed Architecture for Executing Complex Tasks with Multiple Robots*** In IEEE International Conference on Robotics and Automation	2004	16	**3.0**
[S8]	L. E. Parker. ***ALLIANCE: An Architecture for Fault Tolerant Multirobot Cooperation***. In IEEE Transactions on Robotics and Automation	1998	1206	**4.5**
[S9]	J. S. Cepeda, R. Soto, J. L. Gordillo, L. Chaimowicz. ***Towards a Service-Oriented Architecture for Teams of Heterogeneous Autonomous Robots***. In 10th Mexican International Conference on Artificial Intelligence	2011	01	**3.0**
[S10]	D. J. Miller, R. C. Lennox. ***An Object-oriented Environment for Robot System Architectures***. In IEEE Control Systems	1991	136	**4.0**
[S11]	J. F. Inglés-Romero, C. Vicente-Chicote, B. Morin, O. Barais. ***Towards the Automatic Generation of Self-Adaptive Robotics Software: An Experience Report***. In 20th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises.	2011	03	**2.5**
[S12]	J. Kwak, J. Y. Yoon, and R. H. Shinn. ***An Intelligent Robot Architecture based on Robot Mark-up Languages***. In IEEE International Conference on Engineering of Intelligent Systems.	2006	07	**2.5**
[S13]	P. Kazanzides, J. Zuhars, B. Mittelstadt, B. Williamson, P. Cain, F. Smith, L. Rose, B. Musits. ***Architecture of a Surgical Robot***. In IEEE International Conference on Systems, Man and Cybernetics.	1992	19	**3.0**
[S14]	M. Kim, J. Lee, K. C. Kang, Y. Hong, S. Bang. ***Re-engineering Software Architecture of Home Service Robots: A Case Study***. In 27th International Conference on Software Engineering.	2005	16	**4.0**
[S15]	J. Yool, S. Kim, and S. Hong. ***The Robot Software Communications Architecture (RSCA): QoS-Aware Middleware for Networked Service Robots***. In SICE-ICASE International Joint Conference	2006	11	**3.0**
[S16]	W. Hongxing, L. Shiyi, Z. Ying, Y. Liang, W. Tianmiao. ***A Middleware Based Control Architecture for Modular Robot Systems***. In IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications.	2008	09	**2.5**
[S17]	N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku and W. Yoon. ***Composite Component Framework for RT-Middleware (Robot Technology Middleware)***. In IEEE/ASME International Conference on Advanced Intelligent Mechatronics.	2005	18	**3.0**
[S18]	I. Buzurovic, T. K. Podder, L. Fu, and Y. Yu. ***Modular Software Design for Brachytherapy Image-Guided Robotic Systems***. In 2010 IEEE International Conference on Bioinformatics and Bioengineering.	2010	0	**3.0**
[S19]	S. Limsoonthrakul, M. N. Dailey, M. Srisupundit. ***A Modular System Architecture for Autonomous Robots Based on Blackboard and Publish-Subscribe Mechanisms***. In International Conference on Robotics and Biomimetics.	2008	09	**2.5**
[S20]	G. Hu, W. P. Tay, and Y. Wen. ***Cloud Robotics: Architecture, Challenges and Applications***. In IEEE Network.	2012	30	**3.5**
[S21]	A. Angerer, A. Hoffmann, F. Ortmeier, M. Vistein and W. Reif. ***Object-Centric Programming: A New Modeling Paradigm for Robotic Applications***. In International Conference on Automation and Logistics	2009	01	**2.5**
[S22]	M. Y. Jung, A. Deguet, and P. Kazanzides. ***A Component-based Architecture for Flexible Integration of Robotic Systems***. In IEEE/RSJ International Conference on Intelligent Robots and Systems.	2010	19	**2.5**
[S23]	D. Kim and S. Park. ***Designing Dynamic Software Architecture for Home Service Robot Software***. In International Conference on Embedded and Ubiquitous Computing.	2007	05	**4.0**

[S24]	Y. Park, I. Ko, and S. Park. ***A Task-based Approach to Generate Optimal Software-Architecture for Intelligent Service Robots***. 16th IEEE International Conference on Robot & Human Interactive Communication.	2007	01	**2.5**
[S25]	W. Hongxing, D. Xinming, L. Shiyi, T. Guofeng, W. Tianmiao. ***A Component Based Design Framework for Robot Software Architecture***. IEEE/RSJ International Conference on Intelligent Robots and Systems.	2009	06	**2.5**
[S26]	Lorenzo Flückiger, Hans Utz***, Service Oriented Robotic Architecture for Space Robotics: Design, Testing, and Lessons Learned.*** Journal of Field Robotics	2013	01	**4.0**
[S27]	Anis Koubaa***, A Service-Oriented Architecture for Virtualizing Robots in Robot-as-a-Service Clouds.*** 27th International Conference on Architecture of Computing Systems	2014	0	**3.0**
[S28]	James E. Beck, Michael Reagin, Thomas E. Sweeny, Ronald L. Anderson, Timothy D. Garner. ***Applying a Component-Based Software Architecture to Robotic Workcell Applications.*** In IEEE Transactions on Robotics and Automation.	2000	17	**4.0**
[S29]	Javier Gamez García, Juan Gómez Ortega, Alejandro Sánchez García, and Silvia Satorres Martínez***. Robotic Software Architecture for Multisensor Fusion System.*** IEEE Transactions on Industrial Electronics	2009	32	**3.5**
[S30]	Nenad Medvidovic, Hossein Tajalli, Joshua Garcia, and Ivo Krka, Yuriy Brun, George Edwards. ***Engineering Heterogeneous Robotics Systems: A Software Architecture-Based Approach.*** IEEE Computer.	2011	07	**4.0**
[S31]	Soohee Han, Mi-sook Kim, and Hong Seong Park. ***Development of an Open Software Platform for Robotic Services.*** IEEE Transactions on Automation Science and Engineering	2012	12	**4.0**
[S32]	Alex Brooks, Tobias Kaupp, Alexei Makarenko and Stefan Williams, Anders Oreback***. Towards Component-based Robotics.*** IEEE/RSJ International Conference on Intelligent Robots and Systems	2005	168	**2.5**
[S33]	Li Hsien Yoong, Zeeshan E. Bhatti, and Partha S. Roop***. Combining IEC 61499 Model-Based Design with Component-Based Architecture for Robotics.*** The Third International Conference on Simulation, Modeling, and Programming for Autonomous Robots	2012	01	**2.5**
[S34]	Jonghoon Kim, Mun-Taek Choi, Munsang Kim, Suntae Kim, Minseong Kim, Sooyong Park, Jaeho Lee, ByungKook Kim***. Intelligent Robot Software Architecture.*** 13th International Conference on Advanced Robotics.	2008	07	**2.5**
[S35]	Jonghoon Kim, Mun-Taek Choi, Munsang Kim, Suntae Kim, Minseong Kim, Sooyong Park, Jaeho Lee, ByungKook Kim***. Intelligent Robot Software Architecture.*** 13th International Conference on Advanced Robotics.	2008	07	**2.5**
[S36]	Dongsun Kim, Sooyong Park, Youngkyun Jin, Hyeongsoo Chang, Yu-Sik Park, In-Young Ko, Kwanwoo Lee, Junhee Lee, Yeon-Chool Park, Sukhan Lee***. SHAGE: A Framework for self-Managed Robot Software.*** In 2006 International Workshop on Self-adaptation and Self-managing Systems	2006	22	**2.5**
[S37]	Minseong Kim, Suntae Kim, Sooyong Park, Mun-Taek Choi, Munsang Kim, Hassan Gomaa. ***UML-based Service Robot Software Development: A Case Study.*** 28th International Conference on Software Engineering.	2006	29	**3.0**
[S38]	Christian Schlegel, Thomas Haßler, Alex Lotz and Andreas Steck***. Robotic Software Systems: From Code-Driven to Model-Driven Designs.*** International Conference on Advanced Robotics	2009	39	**3.0**
[S39]	Hossein Tajalli, Joshua Garcia, George Edwards and Nenad Medvidovic***. PLASMA: A Plan-based Layered Architecture for Software Model-driven Adaptation.*** IEEE/ACM International Conference on Automated Software Engineering	2010	27	**3.5**
[S40]	Markus Klotzbücher, Nico Hochgeschwender, Luca Gherardi, Herman Bruyninckx, Gerhard Kraetzschmar, Davide Brugali***. The BRICS Component Model: A Model-based Development Paradigm for Complex Robotics Software Systems.*** 28th Annual ACM Symposium on Applied Computing	2013	09	**2.5**
[S41]	John C. Georgas and Richard N. Taylor***. Policy-Based Self-Adaptive Architectures: A Feasibility Study in the Robotics Domain.*** International Workshop on Software Engineering for Adaptive and Self-managing systems	2008	35	**3.0**
[S42]	Andreas Steck, Alex Lotz and Christian Schlegel***. Model-driven Engineering and Run-time Model-usage in Service Robotics.*** 10th ACM International Conference on Generative Programming and Component Engineering	2012	04	**3.5**
[S43]	Francisco Ortiz, Diego Alonso, Bárbara Álvarez, Juan A. Pastor***. A Reference Control Architecture for Service Robots Implemented on a Climbing Vehicle.*** 10th Ada-Europe International Conference on Reliable Software Technologies.	2005	08	**2.5**
[S44]	Jennifer Pérez, Nour Ali, Jose A. Carsı, Isidro Ramos, Bárbara Álvarez, Pedro Sanchez, Juan A. Pastor ***. Integrating Aspects in Software Architectures: PRISMA Applied to Robotic tele-operated Systems.*** Information and Software Technology.	2007	31	**3.5**
[S45]	Carle Côté, Dominic Létourneau, Clément Raïevsky, Yannick Brosseau, François Michaud. ***Using MARIE for Mobile Robot Software Development and Integration.*** Software Engineering for	2007	12	**3.5**

	Experimental Robotics			
[S46]	Antonio C. Domínguez-Brito, Daniel Hernández-Sosa, José Isern-González, Jorge Cabrera-Gámez***. CoolBOT: A Component Model and Software Infrastructure for Robotics.*** Software Engineering for Experimental Robotics	2007	06	**2.5**
[S47]	José Baca, Prithvi Pagala, Claudio Rossi, Manuel Ferre. ***Modular Robot Systems Towards the Execution of Cooperative Tasks in Large Facilities*****.** Robotics and Autonomous Systems	2015	2	**3.5**
[S48]	Didier Crestani, Karen Godary-Dejean, Lionel Lapierre. ***Enhancing Fault Tolerance of Autonomous Mobile Robots***. Robotics and Autonomous Systems	2015	0	**3.5**
[S49]	Andrea Bonarini, Matteo Matteucci, Martino Migliavacca, Davide Rizzi. ***R2P: An Open Source Hardware and Software Modular Approach to Robot Prototyping.*** Robotics and Autonomous Systems	2014	06	**4.0**
[S50]	Christian Berger. ***From a Competition for Self-Driving Miniature Cars to a Standardized Experimental Platform: Concept, Models, Architecture, and Evaluation.*** Journal of Software Engineering for Robotics	2014	07	**3.5**
[S51]	Jan Oliver Ringert, Alexander Roth, Bernhard Rumpe, Andreas Wortmann**.** ***Code Generator Composition for Model-Driven Engineering of Robotics Component & Connector Systems.*** International Workshop on Model-Driven Robot Software Engineering.	2014	0	**3.0**
[S52]	Peihua Chen, Qixin Cao. ***A Middleware-Based Simulation and Control Framework for Mobile Service Robots.*** Journal of Intelligent & Robotic Systems	2014	0	**3.5**
[S53]	Manja Lohse, Frederic Siepmann, Sven Wachsmuth***. A Modeling Framework for User-Driven Iterative Design of Autonomous Systems.*** International Journal of Social Robotics	2014	2	**3.0**
[S54]	Gianluca Antonelli, Khelifa Baizid, Fabrizio Caccavale, Gerardo Giglio, Giuseppe Muscio, Francesco Pierri. ***Control Software Architecture for Cooperative Multiple Unmanned Aerial Vehicle-Manipulator Systems*****.** Journal of Software Engineering for Robotics	2014	0	**3.5**
[S55]	Ion Mircea Diaconescu, Gerd Wagner. ***Towards a General Framework for Modeling, Simulating and Building Sensor/Actuator Systems and Robots for the Web of Things***. International Workshop on Model-Driven Robot Software Engineering	2014	0	**2.5**
[S56]	L. Riazuelo, Javier Civera, J.M.M. Montiel**.** ***C^2TAM: A Cloud Framework for Cooperative Tracking and Mapping*****.** Robotics and Autonomous Systems	2014	18	**4.0**