

Instituto de Ciências Matemáticas e de Computação

ISSN -

Processos para Construção de Arquiteturas de Software de SoS: Uma Revisão Sistemática

Marcelo Benites Gonçalves
Edilson José Davoglio Cândido
Elisa Yumi Nakagawa

Nº

ICMC - RELATÓRIO TÉCNICO

São Carlos, SP, Brazil
Dezembro/2012

Resumo

Arquiteturas de software de Sistemas de Sistemas (do inglês *System of Systems* ou SoS) representam uma peça fundamental para a garantia de qualidade desses sistemas. Dessa forma, é fundamental a construção adequada das arquiteturas de software de SoS. Este trabalho tem o objetivo de verificar quais os processos empregados na construção de arquiteturas de software para SoS e os resultados de efetividade reportados em cada abordagem. Para isso, foi conduzida uma revisão sistemática na qual os estudos primários identificados foram analisados e comparados com base em um modelo de referência que estabelece o conjunto de artefatos e atividades básicos esperados de qualquer processo de construção de arquitetura de software. A seleção final incluiu somente quatro estudos e, além disso, houve baixa aderência às atividades e aos artefatos do modelo de referência adotado. Esses resultados evidenciaram a falta de processos adequados para a construção de arquiteturas de software de SoS e também os desafios nesse campo de pesquisa, ambos os assuntos são tratados neste trabalho. Com isso, este trabalho dá importantes direções para o estabelecimento de melhores estratégias para atacar esse desafio de pesquisa.

Conteúdo

1	Introdução	1
2	Planejamento	3
2.1	Objetivos da Pesquisa	3
2.2	Estratégia de Busca de Estudos Primários	4
2.3	Critérios para Seleção dos Estudos Primários	5
2.4	Processo de Seleção dos Estudos Primários	6
2.4.1	Processo de Seleção Preliminar	6
2.4.2	Processo de Seleção Final	6
2.4.3	Avaliação de Qualidade dos Estudos Primários	6
2.5	Estratégias de Extração de Dados e Sumarização dos Resultados	7
2.6	Considerações Finais	9
3	Condução	11
3.1	Realização das Buscas por Estudos Primários	11
3.1.1	Recuperação Automática dos Estudos Primários nas Bases de Busca	11
3.1.2	Recuperação Manual dos Estudos Primários nas Bases de Busca . .	13
3.2	Resultados das Buscas	14
3.2.1	Seleção dos Estudos Primários	14
3.2.2	Reavaliação de Classificação: Aplicação de VTM	15
3.2.3	Avaliação de Qualidade dos Estudos Incluídos	17
3.3	Extração e Meta-síntese dos Resultados	18
3.4	Considerações Finais	23
4	Conclusões	25
	Referências	29

Lista de Figuras

2.1	Processo de seleção dos estudos primários	7
2.2	Modelo para Comparação entre Processos	9
3.1	Mapa conceitual de estudos primários	17
3.2	Referências das referências entre estudos primários	18
3.3	Referências entre os estudos primários	19

Lista de Tabelas

2.1	Eventos e periódicos considerados na busca manual	5
2.2	Critérios para avaliação de qualidade	8
3.1	Estudos primários recuperados das fontes de pesquisa	14
3.2	Histórico de divergências entre pesquisadores no processo de seleção	15
3.3	Estudos excluídos	15
3.4	Estudos primários incluídos na seleção final	16
3.5	Avaliação final da qualidade	17
3.6	Cobertura das atividades	20
3.7	Cobertura dos artefatos	21
3.8	Aderência geral ao modelo proposto	22

Introdução

Com o avanço das tecnologias computacionais e o crescimento em complexidade e tamanho dos sistemas de software, surgiu o conceito de Sistema de Sistemas (do inglês “*System of Systems*” ou SoS) que pode ser entendido como um sistema construído a partir da integração colaborativa de sistemas menores e operacionalmente independentes (DoD, 2008).

Em outra perspectiva, as arquiteturas de software têm tido um papel fundamental no desenvolvimento de sistemas de software, principalmente no tocante aos atributos de qualidade, tais como a interoperabilidade e a capacidade de evolução dos sistemas (Shaw e Clements, 2006). Segundo (Bass et al., 2012), a qualidade de um sistema raramente pode ser incorporada diretamente ao produto final, uma vez que decisões tomadas nos estágios iniciais do processo de desenvolvimento, inclusive durante a concepção da arquitetura, influenciam diretamente em suas características de qualidade. Nesse contexto, é importante investigar quais as melhores direções para o desenvolvimento de arquiteturas para SoS, uma vez que esses sistemas são altamente complexos com comportamentos, características e aspectos arquiteturais próprios (Firesmith, 2010; Maier, 1998).

O principal objetivo desta revisão sistemática foi a identificação de trabalhos que abordam a construção de arquiteturas de software de SoS e identificar o processo mais adequado para guiar a construção dessas arquiteturas. Essa revisão sistemática envolveu cinco pesquisadores: dois pesquisadores em engenharia de software, dois pesquisadores especialistas em arquitetura de software e um especialista em revisões sistemáticas.

Além dos resultados obtidos ao final da revisão, este trabalho também inclui o detalhamento das atividades intermediárias realizadas, sendo elas: o planejamento da revisão, a estratégia adotada para selecionar e utilizar as máquinas de busca, a seleção de trabalhos, entre outros itens.

Planejamento

O planejamento da revisão sistemática foi realizado de acordo com o modelo de protocolo apresentado por (Kitchenham, 2004). Neste capítulo, são apresentados os principais pontos do protocolo da revisão sistemática.

2.1 Objetivos da Pesquisa

- **Objetivo 1:** Investigar os estudos primários que propõem processos para a construção de arquiteturas de software de SoS;
- **Objetivo 2:** Identificar o processo mais adequado para a construção de arquiteturas de software de SoS.

Questão de Pesquisa: No desenvolvimento de *software-intensive* SoS, qual o processo mais adequado à construção de arquiteturas de software de SoS, considerando como base de comparação o modelo de referência proposto por (Hofmeister et al., 2005)?

Uma questão de pesquisa bem formulada é composta por quatro itens, identificados como PICO (*Population, Intervention, Comparison, Outcome*):

- *População:* desenvolvimento de SoS.
- *Intervenção:* processos de construção de arquitetura de software de SoS.

- *Comparação*: aderência às atividades e aos artefatos básicos para qualquer processo de construção de arquitetura definidos por (Hofmeister et al., 2005).
- *Resultados*: o processo mais adequado à construção de arquiteturas de software de SoS.

2.2 Estratégia de Busca de Estudos Primários

Como estratégia de busca dos estudos primários foi definida a busca manual nos eventos e periódicos mais relevantes da área (apresentados na Tabela 2.1), e a busca automática em bases eletrônicas de busca. Além disso, para a busca automática, foram considerados os elementos descritos a seguir:

- *Seleção das fontes de pesquisa*: a escolha do grupo de bases eletrônicas para as buscas é baseada em (Dieste et al., 2009) que determina os seguintes critérios: (i) disponibilidade dos estudos primários; (ii) cobertura das publicações e conferências relevantes na área; e (iii) a busca por estudos em inglês por ser a língua mais amplamente adotada nos principais eventos e periódicos da área e, além disso, foi incluído o português na tentativa de localizar algum trabalho relevante no Brasil e pesquisadores mais próximos geograficamente. Seguindo esses critérios, as fontes selecionadas foram ACM Digital Library¹, IEEE Xplore², ISI Web of Science³, Science Direct⁴, SCOPUS⁵, e SpringerLink⁶.
- *Palavras-chave*: as palavras-chave foram extraídas da questão de pesquisa, e são: “**system of systems**”, “**process**”, “**software architecture**”, “**design**”.
- *String de busca*: a *string* de busca relacionou as palavras-chave pelo operador lógico AND e, além disso, foram incluídos os possíveis sinônimos de cada palavra-chave relacionados pelo operador lógico OR. Título, resumo e palavras-chave foram adotados como meta-informação para a busca nas bases eletrônicas. A *string* resultante foi (“**software architecture**” AND (“**system of systems**” OR “**system-of-systems**” OR “**systems of systems**” OR **systems-of-systems** OR **SoS**) AND (**method** OR **process**) AND (**design** OR **build** OR **development**)) e;

¹<http://dl.acm.org/>

²<http://ieeexplore.ieee.org/>

³<http://apps.isiknowledge.com/>

⁴<http://www.sciencedirect.com/>

⁵<http://www.scopus.com/>

⁶<http://www.springerlink.com/>

- *Validação da string de busca:* para a validação dessa string, foi utilizado o grupo de controle fornecido pelo especialista da área (Acheson, 2010; Chen e Kazman, 2012; Gagliardi et al., 2009; Rothenhaus et al., 2009).

A busca manual considerou a procura *ad hoc* por estudos em inglês e em português nos sites de eventos e periódicos, listados a seguir, que foram fornecidos pelos especialistas em arquitetura de software:

Tabela 2.1: Eventos e periódicos considerados na busca manual

Evento/Periódico	URL(http://)	Qualis CA-PES
WICSA/ECSA (Working IEEE/IFIP Conference on Software Architecture / European Conference on Software Architecture)	www.wicsa.net/	B1/B3
SBCARS (Simpósio Brasileiro de Componentes, Arquiteturas e Reutilização de Software)	cbsoft2013.cic.unb.br/sbcars	B4
ISARCS (International Symposium on Architecting Critical Systems)	isarcs.comparch2013.org/	B5
CompArch (Federated Events on Component-Based Software Engineering and Software Architecture)	www.comparch2013.org/	B5
QoS (International ACM Sigsoft Conference on the Quality of Software Architectures)	qosa.ipd.kit.edu/qosa_2013/	B5
SoSE (System of Systems Engineering), International Conference on	www.inderscience.com/	-

2.3 Critérios para Seleção dos Estudos Primários

Critério de inclusão

- *Critério de Inclusão:* o estudo primário propõe um processo para construção de arquiteturas de software de SoS.

Critérios de exclusão

- *Critério de Exclusão EC1*: o estudo primário não propõe um processo para a construção de arquiteturas de software;
- *Critério de Exclusão EC2*: o estudo primário não trata de arquiteturas de software de SoS;
- *Critério de Exclusão EC3*: o estudo primário não foi escrito em inglês/português;
- *Critério de Exclusão EC4*: o conteúdo do estudo primário já está presente em outro trabalho mais recente do mesmo autor;
- *Critério de Exclusão EC5*: o texto completo do estudo primário não está disponível;

2.4 Processo de Seleção dos Estudos Primários

A seleção dos estudos primários foi dividida em duas fases principais: o processo de seleção preliminar que inclui a leitura do título, resumo e palavras-chave e o processo de seleção final que inclui a leitura do texto completo dos estudos. Além disso, o processo de seleção envolveu dois pesquisadores que realizaram as atividades de seleção de forma individual e, ao final de cada atividade, confrontaram os resultados alcançados. A Figura 2.1 ilustra como as atividades envolvidas no processo de seleção foram organizadas.

2.4.1 Processo de Seleção Preliminar

Uma vez recuperados os estudos primários das bases de pesquisa são eliminadas as duplicatas e os pesquisadores passam para a atividade de seleção com base nos critérios de inclusão/exclusão em confronto com as informações disponíveis no título, resumo e palavras-chave. Após essa atividade, é realizada uma reunião de consenso na qual as seleções realizadas pelos pesquisadores são confrontadas e as disparidades são discutidas para gerar uma seleção preliminar consensual.

2.4.2 Processo de Seleção Final

Para a inclusão/exclusão final dos estudos primários incluídos na seleção preliminar, considerou-se o confronto dos critérios de inclusão/exclusão com o texto completo desses estudos.

2.4.3 Avaliação de Qualidade dos Estudos Primários

A avaliação de qualidade foi empregada como critério auxiliar de análise, não sendo utilizada para a exclusão de estudos. Os critérios de qualidade adotados e suas respectivas

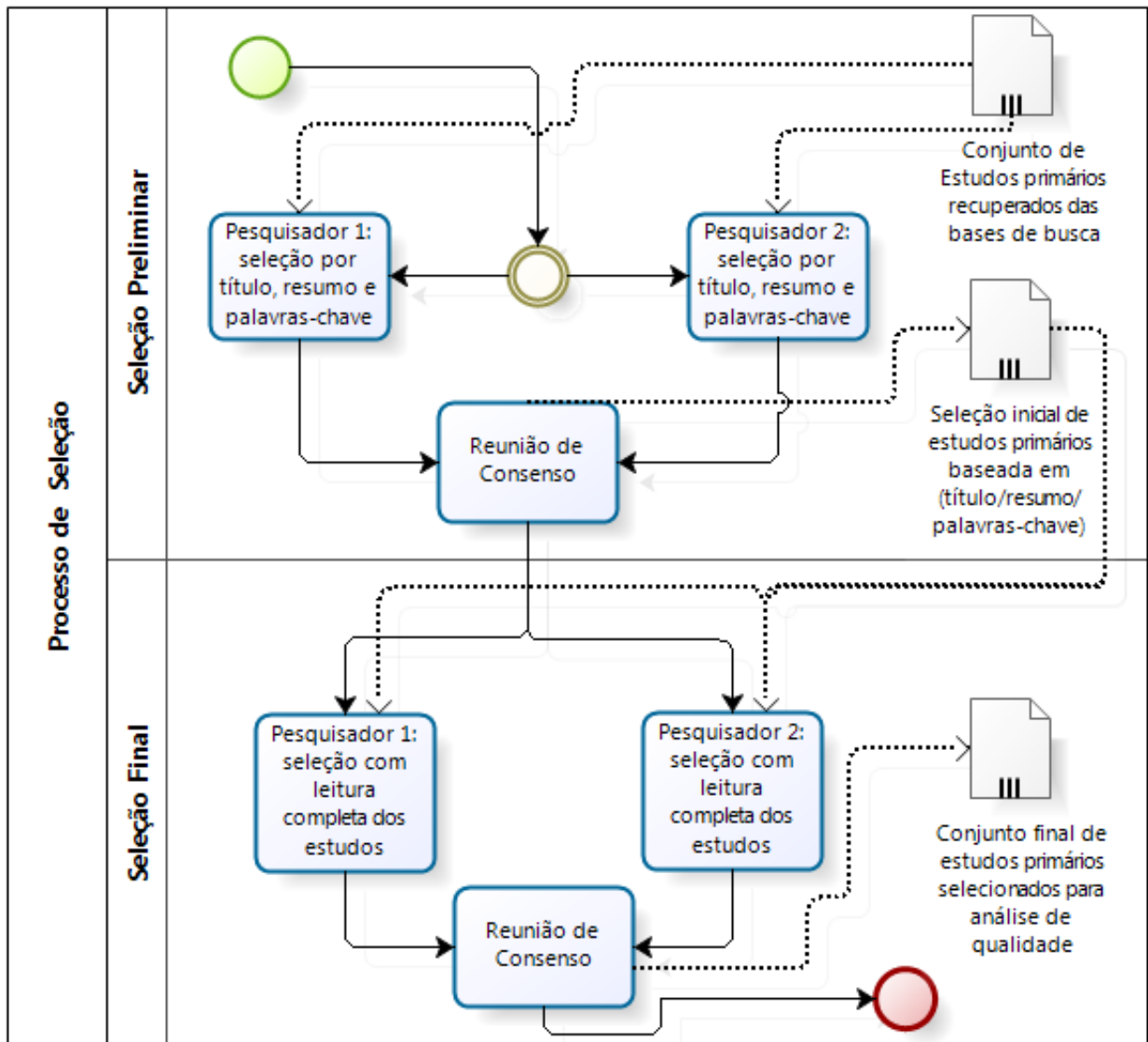


Figura 2.1: Processo de seleção dos estudos primários

escalas de pontuação estão relacionados na Tabela 2.2. Esses critérios consideraram a qualidade de estruturação das propostas dos estudos primários e também seu nível de associação com o tópico central de pesquisa da revisão.

2.5 Estratégias de Extração de Dados e Sumarização dos Resultados

A estratégia de extração de evidências adotada foi a adoção de formulários de extração de dados nos quais, durante a leitura dos textos completos, os pesquisadores alimentam com evidências encontradas nos estudos os campos do formulário. Para dar suporte à execução das estratégias, as seguintes ferramentas foram adotadas:

Tabela 2.2: Critérios para avaliação de qualidade

Critério	Escalas de Pontuação		
	[0,0]	[0,5]	[1,0]
O processo é bem definido em termos de atividades, artefatos e papéis?	Não	Parcialmente	Sim
A proposta central do estudo está claramente definida e fortemente ligada à questão de pesquisa?	Não	Parcialmente	Sim
Em que grau o processo proposto se aplica a SoS?	Aplicação específica	Sub-domínio de SoS	Qualquer tipo de SoS
O processo proposto é validado?	Nenhuma validação	Demonstração teórica ou Estudo de caso na academia	Estudo de caso na indústria

- *JabRef*: organização das referências e dos formulários de pesquisa;
- *Excel*: tabulação de dados numéricos de geração de gráficos.
- *Revis*: auxílio às atividades de seleção com informações sobre tendências para inclusão/exclusão trazidas por meio de VTM (*Visual Text Mining*)⁷.

Quanto à sumarização dos resultados, estabeleceu-se uma estratégia de análise por meio da comparação da aderência do grupo final de estudos incluídos ao modelo geral proposto por (Hofmeister et al., 2005). Esse modelo fornece uma estrutura básica de artefatos e atividades para processos de construção de arquiteturas de software. Ou seja, esse modelo é parte de um *framework* de comparação que determina “o que” qualquer processo para construção de arquitetura de software deve cumprir e, dessa forma, o “como” cada elemento é desenvolvido em cada processo analisado é o objeto de comparação. Os artefatos e atividades desse modelo são listados a seguir e a Figura 2.2 provê uma visão geral dos fluxos e relacionamentos entre esses elementos:

- *Interesse Arquitetural*: interesses relativos ao desenvolvimento do sistema, a sua operação ou qualquer outro aspecto que seja de interesse crítico entre os *stakeholders*;
- *Contexto*: conjunto de circunstâncias e elementos relevantes que definem o ambiente em que o sistema irá operar;
- *Requisitos Arquiteturalmente Significantes (RAS)*: requisito sobre o sistema de software que tem influência na arquitetura e vice-versa;

⁷Para saber mais, acesse ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4343752

- *Soluções Arquiteturais Propostas*: propostas que atendam aos RAS em caráter especulativo;
- *Backlog*: como são registradas as necessidades menores, potenciais ideias futuras;
- *Análise Arquitetural*: como são definidos os problemas que a arquitetura deve resolver (RAS);
- *Síntese Arquitetural*: como são propostas as soluções arquiteturais para um conjunto de RAS;
- *Avaliação Arquitetural*: como é assegurado/avaliado quais as soluções arquiteturais, entre as propostas, devem ser estabelecidas para os RAS existentes;

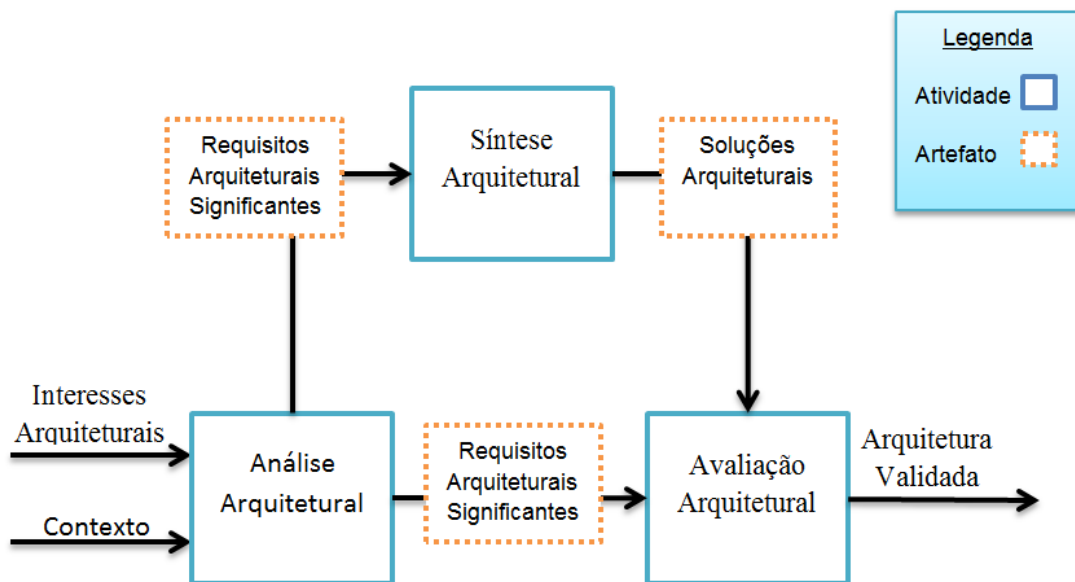


Figura 2.2: Modelo para Comparação entre Processos (Adaptado de (Hofmeister et al., 2005))

2.6 Considerações Finais

Neste capítulo foi apresentado o planejamento da revisão sistemática, abordando as atividades e artefatos utilizados e também pessoas envolvidas e seus papéis. No próximo capítulo será apresentada a condução da revisão sistemática realizada com base nesse planejamento.

Condução

A revisão sistemática foi conduzida por um período de um mês (20 de outubro/2012 a 20 de novembro/2012), de acordo com o planejamento apresentado no capítulo anterior. Neste capítulo é descrita a execução da revisão sistemática de acordo com as atividades estabelecidas no planejamento.

3.1 Realização das Buscas por Estudos Primários

3.1.1 Recuperação Automática dos Estudos Primários nas Bases de Busca

Para avaliar a efetividade da string geral de busca foi utilizado um grupo de controle, fornecido pelos especialistas em arquitetura de software que contribuíram com a revisão sistemática (Acheson, 2010; Chen e Kazman, 2012; Gagliardi et al., 2009; Rothenhaus et al., 2009). Além disso, a efetividade na string adaptada para cada base foi testada verificando-se se os estudos retornados em cada busca continham todos os termos esperados ou seus sinônimos. A string obteve 100% de recuperação dos estudos do grupo de controle na soma dos resultados de avaliações aplicadas nas bases IEEEExplore e Scopus. A seguir, são apresentadas as strings finais adaptadas para cada base utilizada bem como comentários adicionais sobre particularidades e dificuldades de busca em cada base.

ACM Digital Libray

Detalhes: nessa base foi necessário colocar os operadores lógicos (OR/AND) em *lower case*¹. Além disso, a busca foi realizada na seção de busca avançada adotando a estrutura de composição própria da base ((Title:(string)) or (Abstract:(string)) or (Keywords:(string))).

String final adaptada:((Title:(“software architecture” and (“system of systems” or “system-of-systems” or “systems of systems” or sos) and (method or process) and (design or build or development))) or (Abstract:(“software architecture” and (“system of systems” or “system-of-systems” or “systems of systems” or sos) and (method or process) and (design or build or development))) or (Keywords:(“software architecture” and (“system of systems” or “system-of-systems” or “systems of systems” or sos) and (method or process) and (design or build or development))))

IEEEExplore

Detalhes: a string foi aplicada na seção de busca avançada com a opção “Command Search” e seguiu as recomendações de estrutura presentes na base (“Document Title”: (String) OR “Abstract”: (String) OR “Author Keywords”:(String))

String final adaptada:(“Document Title”:“software architecture”AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development) OR “Abstract”: “software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development) OR “Author Keywords”:“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development))

ISI Web of Science

Detalhes: no campo de busca avançada, foi utilizada a estrutura (TI= (String) OR TS=(String)).

String final adaptada:TI=((“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR SoS) AND (method OR process) AND (design OR build OR development))) OR TS=((“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems

¹Mesmo que “caixa baixa”, texto em letras minúsculas.

of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development)))

Science Direct

Detalhes: no modo *expert*, foi utilizada a estrutura (TITLE-ABSTR-KEY(**String**)). String final adaptada: **TITLE-ABSTR-KEY**(“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development))

Scopus

Detalhes: no modo *expert*, foi utilizada a estrutura (TITLE-ABS-KEY(**String**)). Além disso, foi considerada uma recomendação, fornecida na disciplina, em relação a essa base para utilizar chaves ao invés de aspas duplas ao indicar termos compostos que é o indicado nas instruções da base. Contudo, após diversos testes, os resultados indicaram que o problema para utilização de aspas duplas pode ter sido corrigido, uma vez que a base se comportou como esperado ao se utilizar a string com aspas duplas.

String final adaptada:**TITLE-ABS-KEY**(“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR “systems-of-systems” OR SoS) AND (method OR process) AND (design OR build OR development))

SpringerLink

Detalhes: mesmo com o campo de tamanho limitado disponível na busca avançada dessa base, como a string padrão contém um número reduzido de termos, não houve problemas em realizar as buscas. O formato adotado (ab:(**String**)) foi executado no campo regular de buscas.

String final adaptada: **ab**: (“software architecture” AND (“system of systems” OR “system-of-systems” OR “systems of systems” OR SoS) AND (method OR process) AND (design OR build OR development))

3.1.2 Recuperação Manual dos Estudos Primários nas Bases de Busca

Para avaliar a efetividade da busca manual, utilizou-se *websites* de institutos, centros de excelência e empresas que realizam pesquisas associadas com o tema “Sistemas de Sistemas”. Os *websites* escolhidos foram validados pelos especialistas e as buscas basearam-se nos termos definidos e seus sinônimos, utilizados na string de busca automática. A única fonte de recuperação não indexada foi o *website* do SEI (*Software Engineering Institute*). Todos os demais estudos já estavam indexados por meio da busca automática.

3.2 Resultados das Buscas

Desconsiderando estudos duplicados e indisponíveis (10 estudos), foram recuperados ao todo **27** estudos primários para o processo de seleção (inclusão/exclusão) descrito no planejamento. A seguir, são apresentados os resultados do processo de seleção.

Tabela 3.1: Estudos primários recuperados das fontes de pesquisa

Base de Busca	Total	Incluídos Seleção Prelimi- nar	Incluídos Seleção Final	Recuperação
ACM Digital Libray	2	1	0	0%
IEEEExplore	15	4	1	25%
ISI Web of Science	4	1	1	25%
Science Direct	1	1	0	0%
Scopus	15	7	2	50%
SpringerLink	0	0	0	0%
Busca Manual	5	2	1	25%
Totais (sem dupli- catas)	27	12	4	14%

3.2.1 Seleção dos Estudos Primários

Na seleção preliminar, foram incluídos 12 estudos e excluídos 15. Em sequência, na seleção final, restaram quatro estudos primários incluídos para a avaliação de qualidade e comparação. As disparidades entre as seleções dos pesquisadores, bem como a decisão adotada durante as reuniões de consenso são apresentadas na Tabela 3.2. A Tabela 3.3 dá uma visão geral do quanto cada critério de exclusão eliminou estudos e, por fim, a Tabela 3.4 apresenta o grupo final de estudos incluídos. Além dos dados gerais sobre o processo de seleção, a seguir, são apresentadas informações sobre as atividades de reavaliação de classificação e avaliação de qualidade.

Tabela 3.2: Histórico de divergências entre pesquisadores no processo de seleção

Estudo Primário (1º autor)	Fase de consenso	P 1	P 2	Critério de Exclusão	Decisão final
Dobrica (Dobrica e Niemela, 2008)	2	Inclusão	Exclusão	EC1	Excluído
Gagliardi (Gagliardi et al., 2009)	2	Exclusão	Inclusão	EC1	Excluído
Rothenhaus (Rothenhaus et al., 2009)	2	Inclusão	Exclusão	EC1	Excluído
Wang (Wang e Dagli, 2011)	2	Inclusão	Exclusão	EC1	Incluído

Tabela 3.3: Estudos excluídos

Critério de Exclusão	Excluídos Primeira Fase	Excluídos Segunda Fase	Total
EC1	2	1	0
EC2	15	4	1
EC3	4	1	1
EC4	1	1	0
EC5	15	7	2

3.2.2 Reavaliação de Classificação: Aplicação de VTM

O uso da ferramenta Revis(Felizardo et al., 2011) permitiu a verificação de informações sobre o grupo de estudos primários importantes para o processo de revisão sistemática que não são facilmente verificáveis observando apenas os textos dos estudos recuperados. Essas informações foram utilizadas para a auxiliar na escolha de estudos para reavaliação, realizada de forma aleatória no processo original de revisão sistemática proposto por (Kitchenham, 2004). Contudo, mesmo aplicando a atividade de reavaliação da classificação, o grupo de estudos incluídos permaneceu inalterado. A contribuição de cada visualização para a indicação de estudos para reavaliação é apresentada a seguir. Ao final, apenas um estudo foi incluído com base nas informações do VTM, mais três estudos foram seleciona-

Tabela 3.4: Estudos primários incluídos na seleção final

Id	Referência
E1	Acheson, P. Methodology for object-oriented system architecture development Systems Conference, 2010 4th Annual IEEE, pp. 643 -646, 2010.
E2	Humphrey W., "Systems of Systems: Scaling Up the Development Process,"Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical Report CMU/SEI-2006-TR-017, 2006.
E3	Liang, S.; Puett III, J. Luqia Quantifiable software architecture for dependable systems of systems Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3069, 241-265, 2004.
E4	Wang, R. and Dagli, C. Executable system architecting using systems modeling language in conjunction with colored Petri nets in a model-driven systems development process Systems Engineering, 14, 383-409, 2011.

dos de forma aleatória e o grupo foi reavaliado pelos especialistas e não pelos pesquisadores que fizeram a seleção original.

A primeira visualização (Figura 3.1) é um mapa que agrupa os estudos por similaridade de conteúdo considerando título, resumo e palavras-chave. Por meio dessa visualização, foi possível observar que os estudos incluídos estão diretamente ligados por similaridade de conteúdo, o que reforça a confiabilidade nas escolhas realizadas pelos pesquisadores.

A segunda visualização gerada por meio da Revis (Figura 3.2) mostra as referências compartilhadas entre os estudos primários. Essa visualização apontou que os estudos incluídos não tem similaridade de referências e, além disso, que todo o grupo submetido ao processo de seleção tem poucas referências coincidentes. Com isso, não foi possível estabelecer nenhum critério para reavaliação dos estudos incluídos baseado em compartilhamento de referências.

A terceira visualização (Figura 3.3) é o mapa de referências entre estudos, que, dado um estudo primário, verifica quais outros estudos primários do grupo levantado na revisão sistemática ele referencia. Com essa visualização, foi possível observar que os estudos primários não se referenciam mutuamente. Além disso, há um estudo incluído que referencia um excluído. Nesse caso, o estudo excluído poderia ser relevante por estar nas referências de um estudo incluído. Esse estudo foi reavaliado e sua exclusão foi mantida.

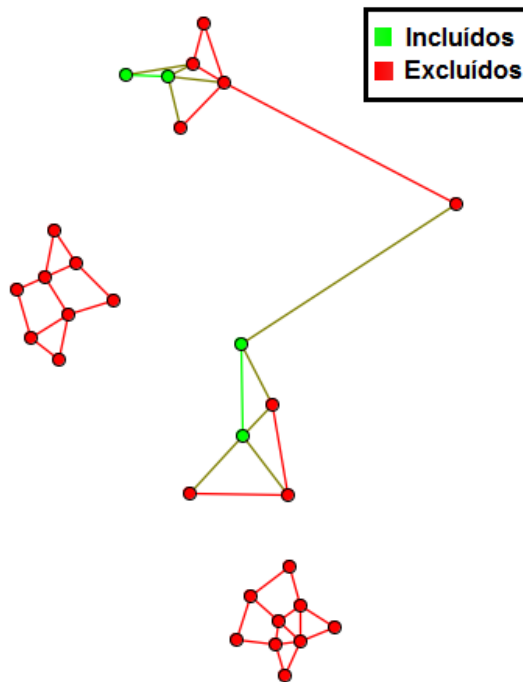


Figura 3.1: Mapa conceitual de estudos primários (ferramenta Revis)

3.2.3 Avaliação de Qualidade dos Estudos Incluídos

A avaliação de qualidade dos estudos incluídos foi realizada independentemente pelos dois pesquisadores que fizeram a classificação dos estudos com base nos critérios estabelecidos no planejamento. Após a avaliação individual, os pesquisadores fizeram uma reunião de consenso sobre a qualidade e ficou estabelecida a avaliação apresentada na Tabela 3.5.

Tabela 3.5: Avaliação final da qualidade

Estudo	O processo é bem definido em termos de atividades, artefatos e papéis?	A proposta central do estudo está claramente definida?	Em que grau o processo proposto se aplica a SoS?	O processo proposto é validado?	Qualidade final
E1	1,0	1,0	1,0	0,5	3,5
E2	0,5	1,0	0,5	0,0	2,0
E3	0,5	0,5	0,5	0,5	2,0
E4	0,5	0,5	0,5	1,0	2,5

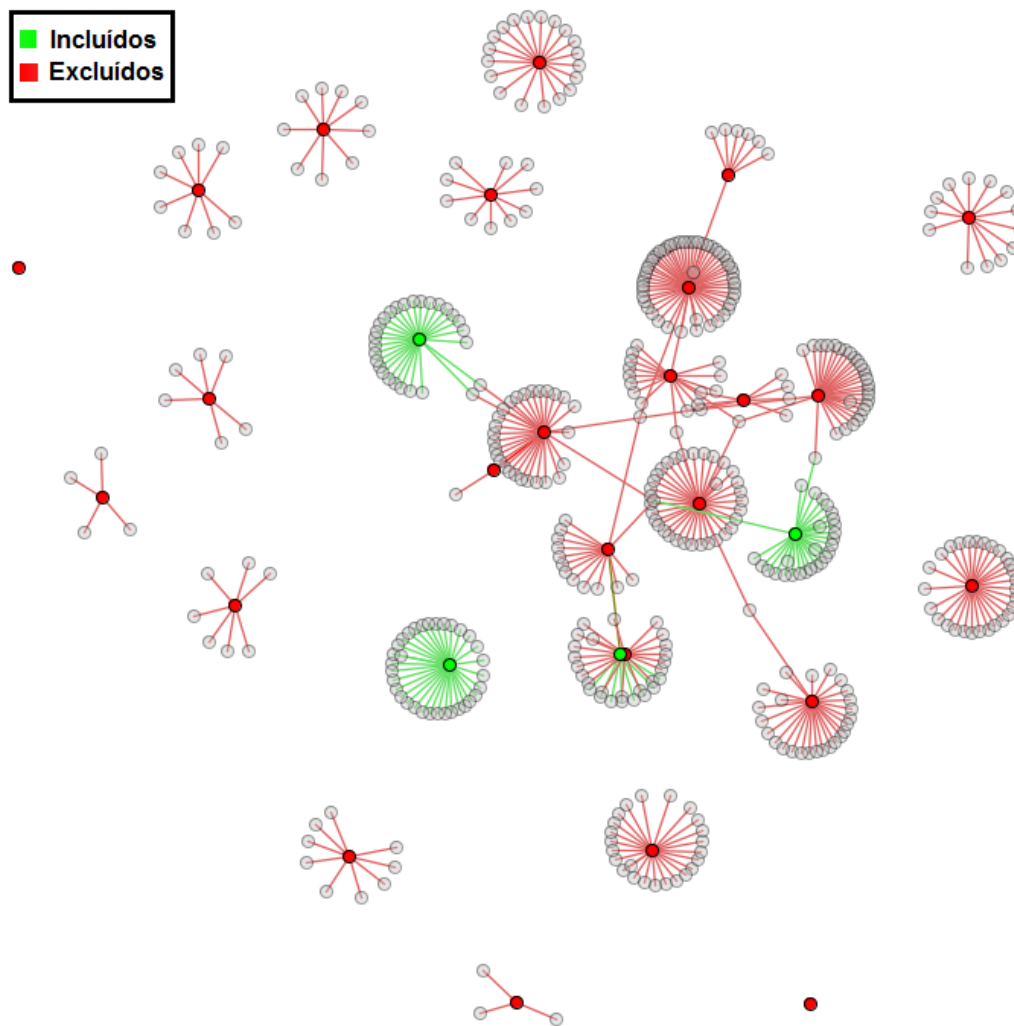


Figura 3.2: Referências das referências (ferramenta Revis)

3.3 Extração e Meta-síntese dos Resultados

Nesta seção, são apresentados os resultados obtidos com a condução da revisão sistemática. De acordo com o planejamento descrito no Capítulo 2, a extração de informações e comparação dos estudos foi organizada baseando-se no modelo de atividades e artefatos básicos esperados para qualquer processo de construção de arquitetura. Além disso, foi considerada a meta-síntese como estratégia para síntese das informações, uma vez que os dados extraídos são qualitativos. Assim, o tratamento estatístico proposto na estratégia de meta-análise não é viável no caso dessa revisão (Bass et al., 2012; Biolchini et al., 2005).

Dessa forma, seguindo o modelo de (Hofmeister et al., 2005), a Tabela 3.6 relaciona, para cada estudo incluído, como são cobertas as atividades esperadas em qualquer processo para construção de arquiteturas de software e a Tabela 3.7 relaciona, para cada estudo incluído, como são cobertos os artefatos esperados em qualquer processo para construção

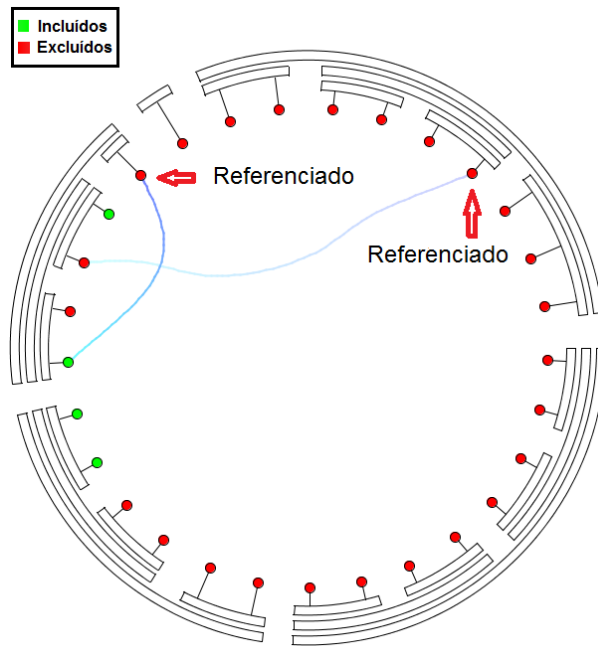


Figura 3.3: Referências entre os estudos primários (*Edge Bundles*, ferramenta Revis) de arquiteturas de software. A Tabela 3.8 apresenta o resultado geral do cálculo da aderência do processo presente em cada estudo incluído ao modelo de referência adotado para comparação.

Tabela 3.6: Cobertura das atividades

Atividade	E1	E2	E3	E4
Análise Arquitetural	Identificação de objetos (OI): levantamento do diagrama AV-1, derivação do OV-1. Definição dos objetivos e expectativas. Concentração nos conceitos centrais do sistema	Definição precisa, simples e concisa dos objetivos e expectativas com o sistema. Definição dos conceitos arquiteturais chave (<i>core system concepts</i>).	<i>Rapid system prototyping</i> aplicada para capturar requisitos computacionais e identificar elementos relevantes à arquitetura.	Representação de ações, fluxos de controle e objetos, além das interfaces entre os componentes do sistema utilizando-se diagramas SysML.
Síntese Arquitetural	Metodologia do quadro negro: um quadro negro vazio inicial é preenchido com todos os elementos do sistema que possam ser possivelmente um objeto. Esse preenchimento é realizado pelo arquiteto juntamente com os <i>experts</i> de domínio.	Empregando múltiplos esforços independentes em paralelo: manter os esforços consistentes com os conceitos arquiteturais chave e construir produtos iniciais.	As atividades são associadas a um conjunto de heurísticas e julgamentos de especialistas, resultando no desenvolvimento e avaliação quantitativa da arquitetura.	<i>Loop</i> de análise estática de requisitos até que todos os <i>stakeholders</i> estejam satisfeitos com o conjunto de soluções propostas.
Avaliação Arquitetural	Os objetos, seus atributos e relacionamentos são iterativamente revisados e refinados após sua elaboração em confronto com os diagramas AV-1 e OV-1 e o quadro negro é reestruturado até que o arquiteto e os <i>experts</i> de domínio estejam satisfeitos com a configuração.	Monitoramento e controle arquitetural constante durante todo o ciclo de vida do SoS.	Verificação e validação sob perspectiva quantitativa. Essa atividade refere-se à inclusão dos atributos de dependência, o alinhamento de restrições com artefatos arquiteturais, a aplicação de restrições ao <i>design</i> e a construção/evolução do sistema.	Comparação das condições e eventos especificados de acordo com o modelo CPN (<i>colored Petri net</i>) por meio de MSCs (<i>Message Sequence Charts</i>).

Tabela 3.7: Cobertura dos artefatos

Artefato	E1	E2	E3	E4
Interesses Arquiteturais	Modelo Formal: Diagramas SysML / Modelo Executável: CPN. Diagramas DoDAF AV-1 e OV-1.	Não atende.	CC (Componentes Conceituais): a representação das perspectivas computacionais dos usuários, bem como as atividades computacionais envolvidas.	Não atende.
Contexto	Diagramas DoDAF AV-1 e OV-1.	Não atende.	Contexto arquitetural definido por regras de interação, estilo arquitetural e protocolos de comunicação.	Não atende.
Requisitos Arquiteturalmente Significantes	Não atende	Conceitos arquiteturais chave (<i>Systems core architectural concepts</i>).	Não atende	Não atende
Soluções Arquiteturais	Quadro negro de objetos	Não atende	Não atende	Modelos semi-formais em SysML, modelos executáveis em redes de Petri
Arquitetura Validada	OR (diagrama de relacionamento entre objetos) e IRD (documento de requisitos de interface) são os últimos artefatos construídos quando a arquitetura já está validada e evoluem com o sistema	Não atende	Framework arquitetural quantificável	Modelos formais em SysML e CPN
Backlog	Não atende	Não atende	Não atende	Não atende

Tabela 3.8: Aderência geral ao modelo proposto

Atividades e artefatos básicos	E1	E2	E3	E4
Análise Arquitetural	1,0	0,5	0,5	0,5
Síntese Arquitetural	0,5	0,5	0,5	0,5
Avaliação Arquitetural	1,0	1,0	1,0	1,0
Interesses Arquiteturais	0,5	0,0	1,0	0,0
Contexto	1,0	0,0	1,0	0,0
Requisitos Arquiteturalmente Significantes	0,0	0,5	0,0	0,0
Soluções Arquiteturais	0,5	0,0	0,0	0,5
<i>Backlog</i>	0,0	0,0	0,0	0,0
Cobertura final	55,5%	27,7%	50%	33,3%

A seguir, é apresentada uma visão geral de cada estudo incluído:

- **E1:** (Acheson, 2010) estabelece um método iterativo e orientado a objetos que se utiliza de elementos do *framework* arquitetural DoDAF (do inglês, *Department of Defense Architectural Framework*) para constituir a base de artefatos que guia o desenvolvimento da arquitetura. Sua abordagem une os conceitos de subsistema e objeto, dessa forma, é proposta uma arquitetura dinâmica e evolutiva para SoS na qual objetos/subsistemas possam ser removidos e adicionados conforme for conveniente;
- **E2:** (Humphrey, 2006) defende que a construção da arquitetura deve seguir uma característica de "comportamento emergente" (Firesmith, 2010; Maier, 1998), na qual as funcionalidades devem emergir na medida em que o SoS amadurece. Dessa forma, o autor propõe que a arquitetura inicial só contemple uma base fundamental e as funcionalidades devem ser estabelecidas conforme o sistema evolui;
- **E3:** (Liang et al., 2004) propõe um processo de construção de arquiteturas de SoS no domínio de sistemas dependentes (DSoS). A proposta central é estabelecer um conjunto de atributos associados à arquitetura e traduzi-los em características que possam ser quantificáveis. Além disso, fornece meios para se avaliar essas características, ao longo do processo de desenvolvimento do DSoS e;
- **E4:** (Wang e Dagli, 2011) propõem um processo para construção de arquiteturas de software baseado em modelos. O passo essencial da proposta é converter especificações arquiteturais construídas em SysML para redes de Petri que atuam complementando as possibilidades de análise e validação da arquitetura, fornecendo

informações dinâmicas sobre estados de transição e comportamentos interativos do sistema enquanto a SysML fornece os diagramas formais para análise estática.

3.4 Considerações Finais

Com os resultados obtidos, a resposta à questão de pesquisa inicial é de que o processo proposto por (Acheson, 2010) é o processo proposto na literatura mais adequado para a construção de arquiteturas de software para SoS, já que este estudo (E1) obteve melhor avaliação de qualidade e maior aderência ao modelo geral proposto por (Hofmeister et al., 2005). Contudo, há fatores relevantes a se considerar. Mais especificamente, foram encontradas poucas iniciativas de processos para construção de arquiteturas de software para SoS, que são geralmente descrições de métodos pouco formalizados e que não contemplam as atividades e artefatos básicos esperados em um processo de construção de arquiteturas de software. Dessa forma, é possível observar que o campo de pesquisa em arquitetura de software para SoS é ainda bastante novo e muito pouco explorado.

Outras observações importantes podem ser listadas considerando os resultados dessa revisão:

- Dos quatro estudos incluídos, dois (E1 e E4) propõem o uso, em algum nível, do *framework* DoDAF (DoD, 2010). O DoDAF já é amplamente aceito, principalmente em aplicações no domínio militar. Ele inclui, mas não se restringe a SoS e fornece uma base para o desenvolvimento de sistemas com ênfase no domínio militar (Richards, 2007). Todas as abordagens encontradas propõem processos incrementais de refinamento arquitetural, revelando, assim como no caso do uso do DoDAF, tendências já presentes em processos para sistemas em larga escala;
- O processo de construção de arquiteturas para SoS é complexo e ainda não foi estabelecido um consenso sobre as formas mais adequadas de realizar sua construção (DoD, 2008; Maier, 1998). Sequer o conceito e caracterização de SoS é consensual entre os pesquisadores (Firesmith, 2010); e
- Nenhum dos estudos incluídos apresenta similaridades entre os processos propostos, o que confirma a pouca colaboração verificada na visualização das referências entre os estudos mostrada na seção que apresenta a estratégia VTM aplicada nessa revisão.

Conclusões

Este trabalho apresentou os resultados de uma revisão sistemática que comparou processos para a construção de arquiteturas de software para SoS. A principal tendência observada foi a adoção de *frameworks* advindos do domínio militar e de estratégias semi-formais (UML/SysML) para a representação dessas arquiteturas. Além dos resultados alcançados com a condução da revisão sistemática, é importante expor as dificuldades percebidas durante a realização desta revisão. Essas dificuldades podem ser agrupadas sob dois contextos principais:

- **Dificuldades gerais para revisões sistemáticas em ES:** (*i*) na busca de estudos primários, a falta de padrão para validação de strings de busca nas bases de busca automática, o que demanda um esforço muito maior para formatar a string original para as particularidades de cada base; (*ii*) no processo de seleção, a baixa qualidade dos resumos dos estudos primários na área de Engenharia de Software aumentam o esforço de seleção, uma vez que estudos que deveriam ser excluídos já na fase de leitura de título, resumo e palavras-chave, acabam avançando para a fase de leitura completa, por apresentarem informações insuficientes nesses elementos e causarem dúvida sobre a exclusão; (*iii*) a dificuldade em realizar a validação estatística requerida na metanálise proposta por (Kitchenham, 2004), uma vez que as informações de interesse na Engenharia de Software tem geralmente um caráter subjetivo e os estudos apresentam poucas evidências experimentais para a validação de suas propostas; e

- **Dificuldades específicas relacionadas ao tópico de pesquisa:** falta de consenso para caracterização de SoS e SoS são geralmente sistemas de larga-escala, o que dificulta a validação em ambiente real de desenvolvimento. Essa falta de validação dificulta a avaliação da efetividade das propostas, considerando que nenhuma apresenta resultados de utilização para serem comparados.

Por fim, a despeito das limitações e dificuldades verificadas, os resultados trazem respostas para um tópico ainda pouco explorado no domínio de SoS. Portanto, vale a pena destacar a necessidade de mais pesquisas na área de arquitetura de software de SoS, o que pode de alguma forma contribuir para o desenvolvimento desses sistemas e, como consequência, contribuir também para aumentar sua agilidade.

Referências

- Acheson, P. Methodology for object-oriented system architecture development. In: *Systems Conference, 2010 4th Annual IEEE*, 2010, p. 643–646.
- Bass, L.; Clements, P.; Kazman, R. *Software architecture in practice*. 3 ed. Addison-Wesley, 2012.
- Biolchini, J.; Mian, P.; Natali, A.; Travassos, G. *System engineering and computer science*. Relatório Técnico 5, COPPE/UFRJ, 2005.
- Chen, H.; Kazman, R. Architecting ultra-large-scale green information systems. In: *2012 1st International Workshop on Green and Sustainable Software, GREENS 2012*, Zurich, 2012, p. 69–75.
- Dieste, O.; Grimán, A.; Juristo, N. Developing search strategies for detecting relevant experiments. *Empirical Software Engineering*, v. 14, n. 5, p. 513–539, 2009.
- Dobrica, L.; Niemela, E. An approach to reference architecture design for different domains of embedded systems. In: *Proceedings of the 2008 International Conference on Software Engineering Research and Practice, SERP 2008*, Las Vegas, NV, 2008, p. 287–293.
- DoD *System engineering guide for systems of systems*. Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering, version 1.0, 2008.
- DoD The dodaf architecture framework version 2.02. 2010.
Disponível em <http://dodcio.defense.gov/>

- Felizardo, K. R.; Salleh, N.; Martins, R. M.; Mendes, E.; MacDonell, S. G.; Maldonado, J. C. Using visual text mining to support the study selection activity in systematic literature reviews. In: *V International Software Engineering and Measurement (ESEM)*, Banff, Alberta, Canada, 2011.
- Firesmith, D. *Profiling systems using the defining characteristics of systems of systems (sos)*. Relatório Técnico, Software Engineering Institute (SEI), 2010.
- Gagliardi, M.; Wood, W.; Klein, J.; Morley, J. A uniform approach for system of systems architecture evaluation. *CrossTalk*, v. 22, n. 3-4, p. 12–15, cited By (since 1996) 0, 2009.
- Hofmeister, C.; Kruchten, P.; Nord, R.; Obbink, H.; Ran, A.; America, P. Generalizing a model of software architecture design from five industrial approaches. In: *WICSA 5th Working IEEE/IFIP Conference*, 2005, p. 77–88.
- Humphrey, W. *Systems of systems: Scaling up the development process*. Relatório Técnico, Software Engineering Institute (SEI), Carnegie Mellon University, Pittsburgh, Pennsylvania, 2006.
Disponível em <http://www.sei.cmu.edu/library/abstracts/reports/06tr017.cfm>
- Kitchenham, B. *Procedures for performing systematic reviews*. Relatório Técnico, Keele University and NICTA, 2004.
- Liang, S.; Puett III, J.; Luqia. Quantifiable software architecture for dependable systems of systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3069, p. 241–265, 2004.
- Maier, M. W. Architecting principles for systems-of-systems. *Systems Engineering*, v. 1, n. 4, p. 267–284, 1998.
- Richards, M. *Managing complexity with the department of defense architecture framework: Development of a dynamic system architecture model*. Tese de Doutorado, Massachusetts Institute of Technology, 2007.
- Rothenhaus, K.; Michael, J.; Shing, M.-T. Architectural patterns and auto-fusion process for automated multisensor fusion in soa system-of-systems. *IEEE Systems Journal*, v. 3, n. 3, p. 304–316, 2009.
- Shaw, M.; Clements, P. The golden age of software architecture. *IEEE Software*, v. 23, n. 2, p. 31–39, 2006.

Wang, R.; Dagli, C. Executable system architecting using systems modeling language in conjunction with colored petri nets in a model-driven systems development process. *Systems Engineering*, v. 14, n. 4, p. 383–409, 2011.